mathematics

MDPI

Article

An Algorithm for Numerical Integration of ODE with Sampled Unknown Functional Factors

Y. Villacampa and F. J. Navarro-González *

Department of Applied Mathematics, University of Alicante, 03690 San Vicente del Raspeig, Spain; villacampa@ua.es

* Correspondence: francisco.navarro@ua.es

Abstract: The problem of having ordinary differential equations (ODE) whose coefficients are unknown functions is frequent in several fields. Sometimes, it is possible to obtain samples of the values of these functions in different instants or spatial points. The present paper presents a methodology for the numeric solving of these ODE. There are approximations to the problem for specific cases of equations, especially in the case where the parameters correspond to constants. Other studies focus on the case in which the functions under consideration are linear or meet a certain condition. There are two main advantages of the proposed algorithm. First, it does not impose any condition over the data or the subsequent function from where these sample data are derived. Additionally, the methodology used in the functions modeling can control the possibility of overfitting in the function modeling. This is a crucial point in order to limit the influence of model biases in the numerical solution of the ordinary differential equation under study.

Keywords: ordinary differential equations; unknown functional coefficients; equation coefficient sampling

MSC: 34F05

Citation: Villacampa, Y.; Navarro-González, F.J. An Algorithm for Numerical Integration of ODE with Sampled Unknown Functional Factors. *Mathematics* **2022**, *10*, 1516. https://doi.org/10.3390/math10091516

Academic Editor: Zhisheng Shuai

Received: 31 March 2022
Accepted: 26 April 2022
Published: 2 May 2022

1. Introduction

In the study and modeling of systems, differential equations can be obtained with parameters that depend on observations. That is, they are parameters that can vary throughout the domain of definition and for which experimental data are known. Differential equations can be either ordinary differential equations or partial differential equations. Problems modeled by this type of differential equations are, for example, (see [1–5]): (a) the heat equation with unknown specific heat and heat sources; (b) the diffusion equation of products with non-constant and unknown innovation and imitation coefficients; (c) the study of hydraulic transients; and (d) water flows such as groundwater seepage in anisotropic media. In this sense, it is important to obtain hybrid-modeling methodologies that allow combining parameter estimation methods with algorithms of numerical resolution of differential equations.

When the dynamics of the processes underlying the system under study are known, differential equations can model their behavior, especially in the case of complex dynamic processes. In many cases, these differential equations contain constant parameters or functional dependencies with the independent variable that determine the characteristics of the corresponding solutions. The exact knowledge of its parameters is crucial for predicting the behavior of the solutions. Some concrete examples of the study of the problem considered in specific fields could be, for example, in biology and ecology [6–8], chemistry [9,10], electronics [11] and engineering [12,13].

The problem of determining the constant coefficients present in a differential equation from the study of the solutions of the system has been the subject of different studies, presenting a variety of methods of resolution. First approaches were based on least squares techniques. However, due to the absence of an analytical solution, the need to solve the

ODEs numerically makes this approach computationally expensive. An alternative is the generalized profiling estimation method, where the solution is approximated using a linear combination of basis functions and adjusting the coefficients of the corresponding expansion equation by comparing them with the differential equation (see [14,15]). This method is applied even in the case in which the model is stochastic [16]. A study on the properties of this type of algorithm can be found in [17]. Other investigations are based on the use of non-deterministic methods (for example genetic algorithms [18], particle swarm optimization [10]) or Machine Learning (ML) techniques. Some studies focus on methods applicable to specific types of equations. For example [19], where the problem corresponds to a differential equation associated with the time evolution of a system, and is solved by making a large number of observations of the evolution of the system over short intervals.

However, the application of the techniques discussed in real problems presents a series of difficulties due to the characteristics of the numerical data available in such cases, as discussed in [20]. In addition, the presence of outliers makes it necessary to design and use robust algorithms, see, for example, [21]. There are also some considerations that complicate the efficient use of some of the methods. For example, in stiff problems, the presence of nonlinearity in the solution space [10] or high dimensionality of the parameters space, in partial differential equations (PDE) [12].

The approach taken in the present investigation is in a certain way the inverse of that of the cases previously considered. It is proposed to obtain a solution of the differential equations by previously determining the values of the parameters that best fit the observations made. However, the approach used in [10] has a certain similarity in its first stages to the one presented in this paper.

When the function to model corresponds to a spatial-distributed variable, kriging is usually used as the best linear estimator of the function from a set of random measurements. This can be found in the work of [22,23] for the calculation of unknown coefficients in linear PDE [24,25].

The proposal presented in this research work performs a modeling of the parameters of the differential equation in the case of functional dependence, by means of a local regression method based on the finite element method. Therefore, a brief introduction to the modeling methods will be given below.

The number of techniques available to model a relationship between variables is large. Selecting one or another depends on a variety of factors, from the data itself to the knowledge of the underlying equation involving the variables. The toolbox goes from classical statistical methods as linear or multilinear regression to machine-learning oriented methods as neural networks and related techniques, random trees, support vector regression and others.

The problem under consideration consists of obtaining a model of the relationship between a set of variables that is assumed to be determined by a function $f : \Omega \subset \mathbb{R}^d \to \mathbb{R}$

$$y = f(x^1, \ldots, x^d). \tag{1}$$

All the considered methods start from an experimental dataset Y of P tuples sampled from the relation (1):

$$\mathrm{Y} = \left\{ \left(x^1_{[k]}, x^2_{[k]}, x^3_{[k]}, \ldots.., x^d_{[k]}, y_{[k]} \right) \right\}_{k=1,2,\ldots,P}, \tag{2}$$

where, for every $k = 1, 2, \ldots, P$ is verified $\left(x^1_{[k]}, x^2_{[k]}, x^3_{[k]}, \ldots.., x^d_{[k]}, y_{[k]} \right) \in \Omega \times \mathbb{R}$.

The literature developing the different methods can be easily accessed and they are widely applied to a variety of problems. From these data, each methodology allows us to obtain estimates of the values of $f(x^1, \ldots, x^d)$ at each point of Ω using an algorithm:

$$\begin{array}{c} F : \Omega \subset \mathbb{R}^d \times \mathbb{R}^J \to \mathbb{R} \\ (x^1, \ldots, x^d; \theta^1, \ldots, \theta^J) \longmapsto F(x^1, \ldots, x^d; \theta^1, \ldots, \theta^J) \end{array}, \tag{3}$$

where $(\theta^1, \ldots, \theta^J)$ represents a set of parameters specific to each methodology and which determines the output value of the model. These estimated values will be denoted by:

$$\hat{y} = F(x^1, \ldots, x^d; \theta^1, \ldots, \theta^J). \tag{4}$$

The methods for numeric model estimation are usually stated as a minimization problem over some kind of global error E obtained as a function $H : \mathbb{R}^P \to \mathbb{R}$ of individual errors $\varepsilon_{[k]}$ defined over the sampled and estimated values:

$$E = H(\varepsilon_{[1]}, \varepsilon_{[2]}, \ldots., \varepsilon_{[P]}) \tag{5}$$

with $\varepsilon_{[k]} = y_{[k]} - \hat{y}_{[k]} = y_{[k]} - F(x^1_{[k]}, \ldots.., x^d_{[k]}; \theta^1, \ldots, \theta^J)$.

Thus considered, the function H allows the approach of an optimization problem (see Equation (6)). Then, the optimal values $(\theta^1_*, \ldots, \theta^J_*)$ that minimize the value of H for the available dataset are sought.

$$H\Big(y_{[1]} - F(x^1_{[1]}, \ldots, x^d_{[1]}; \theta^1, \ldots, \theta^J), \ldots, y_{[P]} - F(x^1_{[P]}, \ldots, x^d_{[P]}; \theta^1, \ldots, \theta^J)\Big). \tag{6}$$

These optimum parameters define the final model F_* as:

$$F_*(x^1, \ldots, x^d; \theta^1_*, \ldots, \theta^J_*). \tag{7}$$

The present work follows the previous line of research developed by the authors with methodologies based on the finite elements method (FEM), a method for finding numerical solutions to differential equations with boundary conditions, developed initially to be applied in civil and aeronautical engineering [26–31].

In the following, in order to simplify notation, the criterion of writing the d-dimension points as a single variable will be used. Let us consider a given differential equation defined by a differential operator D, acting on a domain $\Omega \subset \mathbb{R}^d$:

$$D(f) = v, \tag{8}$$

where $f, v \in V(\Omega)$, V being a function space defined over Ω.

The finite element method replaces the domain Ω of V by a collection of N_e closed sets $\{\Omega_1, \ldots, \Omega_{N_e}\}$ called elements, that verify:

$$\begin{array}{c} \mathrm{U}_{j=1}^{N_e} \Omega_j = \Omega \\ \Omega_{j_1} \cap \Omega_{j_2} = \partial\Gamma_{j_1} \cap \partial\Gamma_{j_2} \end{array}, \tag{9}$$

where $\partial\Gamma_j$ is the frontier of the closed Ω_j.

This process is called meshing and it is usually done using sets of specific geometries, for example, in two dimensional triangles or rectangles. The generated mesh has an associated number that is related to the size of the elements. This parameter is usually denoted by h and relates to the order of the error in the interpolations that will be defined in the next paragraph:

$$h = \min_{\rho \epsilon \mathbb{R}} \big\{ B(x_j, \rho) \supset \Omega_j \;\big|\; x_j \in \Omega_j,\ j = 1, \ldots, N_e \big\}, \tag{10}$$

where $B(x_j, \rho)$ is the open ball centred in x_j with radius ρ.

Meshing on a domain also defines a set of Q points called nodes $(\zeta_1, \zeta_2, \ldots\ldots, \zeta_Q)$ that are used as support for the interpolation of any function defined on Ω through a related set of functions called shape functions $(\varphi_1(x), \varphi_2(x), \ldots\ldots, \varphi_Q(x))$ verifying the following conditions at the nodes:

$$\varphi_i(\zeta_j) = \delta_{ij}, \tag{11}$$

for every $i, j = 1, \ldots, Q$.

The linear span defined from the set of functions $(\varphi_1(x), \varphi_2(x), \ldots\ldots, \varphi_Q(x))$ forms a finite dimensional subspace which is composed by continuous piecewise polynomial functions of degree K (Sobolev space). This vector space is associated with a given division of the domain where the problem is set in and it will be denoted by V_h. That is:

$$V_h = span(\varphi_1(x), \varphi_2(x), \ldots\ldots, \varphi_Q(x)). \tag{12}$$

On the space V_h, a derivative operator D_h that generalizes the usual derivative to the functions of V_h that are not derivable at all points of Ω can be defined.

The original problem can now be posed by searching the function $f_h \epsilon V_h$:

$$f_h(x) = \sum_{i=1}^{Q} u_i \cdot \varphi_i(x) \tag{13}$$

that best approximates the equation:

$$D_h(f_h) = v_h, \tag{14}$$

where $v_h \epsilon V_h$ is the function of V_h that corresponds with the interpolation of $v(x)$ defined in Equation (8).

The application of the FEM to the modelling of systems developed by the authors is based on the principle of projection through interpolation of any function $f \in V(\Omega)$ in the space V_h as:

$$f(x) = \sum_{i=1}^{Q} \alpha_i \cdot \varphi_i(x) \quad \begin{array}{c} F : \Omega \subset \mathbb{R}^d \times \mathbb{R}^J \to \mathbb{R} \\ (x^1, \ldots, x^d; \theta^1, \ldots, \theta^J) \longmapsto F(x^1, \ldots, x^d; \theta^1, \ldots, \theta^J) \end{array}, \tag{15}$$

where $\alpha_i = f(\zeta_i)$.

The numerical regression model for the relation Equation (1) will consist of the determination of the values $\{\alpha_1, \ldots, \alpha_Q\}$ that minimize an error function defined in a similar way to that in Equation (5).

The authors have been developing regression techniques based on Equation (15) as a method of approximation [32–35]. The last proposal is given by the octahedric regression methodology that will be presented in the next section.

The rest of the paper is organized as follows: Section 2 presents the basics of the modelling technique used in the proposed methodology, the octahedric regression and a study on the overfitting control and the computational algorithm. Section 3 presents four application examples. The first three correspond to the static heat equation and the last one to the Bass equation. Finally, Section 4 presents an analysis of the results obtained and a description of the computational characteristics of the methodology, together with a reflection on future lines of research.

2. Materials and Methods

The problem under consideration can be divided into two different phases: first, a model for the coefficients of the equation is determined. After this model has been obtained, it can be used to solve the differential equation using a typical numerical method, as those based on Runge–Kutta or multistep approaches.

In this section, we will proceed to the explanation of the modeling method to be used to estimate the coefficient functions at the nodes where the approximate numerical function is calculated. The following section presents the proposed solution to deal with the problem of overfitting, which is common to the techniques of modeling from experimental data. The last subsection develops the algorithm in the form of pseudocode, commenting on its main features.

2.1. Octahedric Regression

The problem of obtaining a model for the functional coefficients will be solved using the technique developed by the authors, called octahedric regression, and is presented in the research paper [36]. This algorithm presents good properties with respect to the following points: easy detection and control of the overfitting problem, efficient parallelization based on its embarrassing parallelism and fast numerical estimation for out of sample points.

Octahedral regression is an evolution of the previous numerical methodologies developed by the authors and corresponds to the fastest and most efficient of them. The next paragraphs provide a brief presentation of the basis of this methodology.

The basis is the use of parameterized radial functions, defined in the following point.

Definition 1. *A radial function is a function* $\Phi : \mathbb{R}_+ x \mathbb{R}_+ \to \mathbb{R}_+$ *that accomplishes the conditions*

$$\left.\begin{array}{l} \forall \omega \neq 0 \ \lim\limits_{r \to \infty} \Phi(r, \omega) = 0 \\ \forall r \neq 0 \ \lim\limits_{\omega \to 0} \Phi(r, \omega) = 0 \end{array}\right\}. \tag{16}$$

The next definition introduces the averaged value estimator for a function at any point. In the following points, it is assumed that a radial function $\Phi(r, \omega)$ is selected and that the conditions on the functions for the existence of the corresponding integrals are satisfied.

Definition 2. *Given a function* $f : \Omega \subset \mathbb{R}^d \to \mathbb{R}$ *, the weighted average regression* $c^*(x_o)$ *of* f *at* $x_o \in \Omega$ *is defined using the implicit definition:*

$$c^*(x_o) \cdot W^0(x_o) - \int_\Omega f(x) \cdot \Phi(\|x - x_o\|, \omega)\, dx = 0 \quad \left.\begin{array}{ll} \forall \omega \neq 0 & \lim\limits_{r \to \infty} \Phi(r, \omega) = 0 \\ \forall r \neq 0 & \lim\limits_{\omega \to 0} \Phi(r, \omega) = 0 \end{array}\right\}, \tag{17}$$

where $W^0(x_o) = \int_\Omega \Phi(\|x - x_o\|, \omega)\, dx$ *and* $\Phi(\|x - x_o\|, \omega)$ *are a function as presented in Definition 1.*

In order to reduce the bias that may exist when calculating the previous weighted average, a set of points is introduced that will be used in the final estimation of the model.

Definition 3. *Given a point* $x_o \epsilon\, \Omega$ *and a parameter h (in the model is called pseudocomplexity), the octahedric support of size h around* x_o *is the set of* $2 \cdot d$ *points:*

$$\left\{ x_o + \frac{h}{2} \cdot e_1, x_o + \frac{h}{2} \cdot e_2, \ldots .., x_o + \frac{h}{2} \cdot e_d, x_o - \frac{h}{2} \cdot e_1, x_o - \frac{h}{2} \cdot e_2, \ldots .., x_o - \frac{h}{2} \cdot e_d \right\}, \tag{18}$$

where $\{e_j\}_{j=1}^d$ *are the vectors of the canonical basis of* $\mathbb{R}^d$.

Definition 4. *The octahedric regression model of* $f(x)$ *at* x_o *is defined as the average of the weighted average regressions* $c^*\left(x_o \pm \frac{h}{2} \cdot e_j\right)$ *calculated at the points of the octahedric support introduced in Definition 3. This estimated value will be denoted by* $\hat{f}(x_o)$ *and is given by:*

$$\hat{f}(x_o) = \frac{1}{2d} \cdot \sum_{j=1}^d \left[c^*\left(x_o + \frac{h}{2} \cdot e_j\right) + c^*\left(x_o - \frac{h}{2} \cdot e_j\right) \right]. \tag{19}$$

The following proposition shows the nature of the proposed estimator.

Proposition 1. *The octahedric estimation* $\hat{f}(x_o)$ *of* $f(x)$ *at* x_o *is a correction of order two with respect to the weighted average regression* $c^*(x_o)$ *at the same point.*

Demonstration. Developing the $c^*\left(x_o \pm \frac{h}{2}\cdot e_j\right)$ functions in Equation (19) and $\Phi\left(\|x - x_o \pm \frac{h}{2}\cdot e_j\|, \omega\right)$ through the implicit definition given in Equation (17) in powers of h, the first order term cancels and the following expression holds:

$$\hat{f}(x_o) = c^*(x_o) + O(h^2). \tag{20}$$

Equation (20) shows that octahedric regression is a correction of order h^2 to the weigthed average for central objective points. Consequently, when $h \to 0$, both values tend to coincide $\hat{f}(x_o) \to c^*(x_o)$, causing overfitting on the points of the sample with respect to the points nearest to x_o. The way to study the existence and importance of this effect is through a second estimation called restricted model, where in the calculation of the model at a point, the corresponding values are removed from the sample in the case of being part of it. The results will be discussed in the next subsection.

When the problem is modelling a function from a discrete sample of size P (see Equation (2)) of points and values of $f(x)$, the integrals must be calculated using finite sums, so:

$$W^0(x_o) = \int_\Omega \Phi(\|x - x_o\|, \omega)\, dx \cong \frac{1}{P}\cdot \sum_{k=1}^{P} \Phi(\|x_k - x_o\|, \omega) \tag{21}$$

$$\int_\Omega f(x)\cdot\Phi(\|x - x_o\|, \omega)\, dx \cong \frac{1}{P}\cdot \sum_{k=1}^{P} y_k\cdot\Phi(\|x_k - x_o\|, \omega). \tag{22}$$

Then, the discrete version of Equation (17) is:

$$\sum_{k=1}^{P}[c^*(x_o) - x_k]\cdot\Phi(\|x_k - x_o\|, \omega) = 0. \tag{23}$$

2.2. Control of Model Overfitting

From the result of Proposition 1, the trend to overfitting has been discussed. In this point, the question will be considered further.

Let us study the problem of estimating the value of an objective function $f(x)$ at the first point of the sample $x_1 \in Y$, and suppose that the rest of the points on the sample are ordered depending on the distance to x_1. Taking Equation (21) at the support points $x_1 \pm \frac{h}{2}\cdot e_j$,

$$W^0\left(x_1 \pm \frac{h}{2}\cdot e_j\right) \approx \Phi\left(\frac{h}{2}, \omega\right)\cdot\left\{1 + \frac{\Phi\left(\|x_2 - x_1 \mp \frac{h}{2}\cdot e_j\|, \omega\right)}{\Phi\left(\frac{h}{2}, \omega\right)} + .. + \frac{\Phi\left(\|x_P - x_1 \mp \frac{h}{2}\cdot e_j\|, \omega\right)}{\Phi\left(\frac{h}{2}, \omega\right)}\right\} \tag{24}$$

The term on brackets represents the relative weight of each sample point in the weighted average. Summing up the contributions on each support point, the result can be written as:

$$1 + \Psi(x_2 - x_1, \omega) + .. + \Psi(x_P - x_1, \omega), \tag{25}$$

where

$$\Psi(x_k - x_l, \omega) = \frac{\sum_{j=1}^{d}\left\{\Phi\left(\|x_k - x_l - \frac{h}{2}\cdot e_j\|, \omega\right) + \Phi\left(\|x_k - x_l + \frac{h}{2}\cdot e_j\|, \omega\right)\right\}}{2\cdot \mathrm{d}\cdot\Phi\left(\frac{h}{2}, \omega\right)}. \tag{26}$$

The value of

$$\Pi(x_r) = 1 + \sum_{k\neq r}^{P} \Psi(x_k - x_r, \omega) \tag{27}$$

represents the weight of all the points in the estimation of the model calculated at x_r. At zero order in h, Equation (27) is approximately:

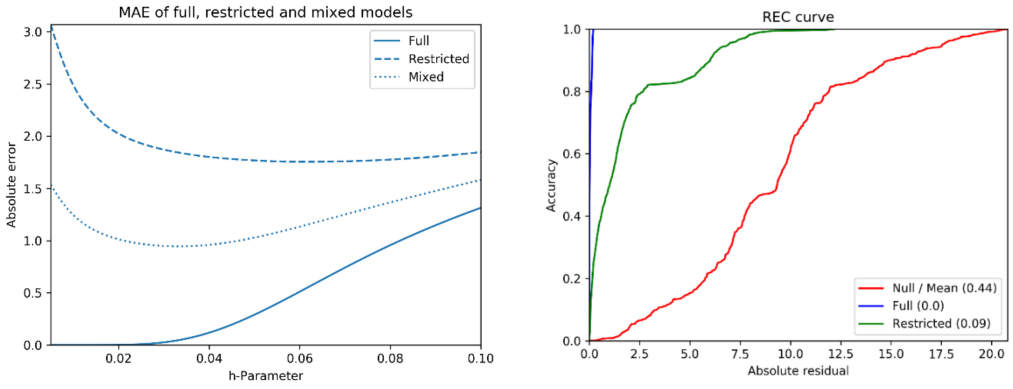
$$\Pi(x_r) \cong 1 + \frac{\sum_{k \neq r}^{P} \Phi(\|x_k - x_r\|, \omega)}{\Phi\left(\frac{h}{2}, \omega\right)}. \tag{28}$$

By Equation (16), the fractions converge very fast to zero as the distance to x_r grows, and the only contribution depends on the points that are at a distance similar to the nearest point, that is, when $\Phi(\|x_k - x_r\|, \omega) \approx \Phi\left(\frac{h}{2}, \omega\right)$. The number $q(\omega)$ of points involved is determined by the value of the parameter ω, and $c^*(x_r)$ is a weighted mean of the q-nearest points. Therefore, the octahedric regression presented in the present paper corresponds to a mean of a simpler estimator calculated on the support of size h defined around x_r. These simple estimators correspond to the weighted average of the $q(x_r, \omega)$-nearest neighbors. In the limit $\omega \to 0$, $c^*\left(x_r \pm h \cdot e_j\right)$ are obtained from the nearest point. In the case of considering all the experimental points, the model is called full. In that case, when $h \ll 1$, the nearest to the support points will very frequently be x_r, and then: $c^*\left(x_r \pm h \cdot e_j\right) \to y_r$, and according to Equation (20), $\hat{f}(x_r) \to y_r$, confirming the trend to overfitting.

The overfitting is caused by the incorporation of noise to the model. If the point that is being calculated is not included in the estimation, the points that are used to obtain the values of $c_i^*\left(x_r \pm h \cdot e_j\right)$ have a greater probability of presenting independent noise influence, diminishing the overfitting in that form. The new model calculated in this form is called restricted and corresponds to having a cross-validation for each experimental point where the test set is formed by itself.

It has been commented previously that the methodology trends to overfitting when the size of the parameter h is small. Now, a real case of the application of octahedric regression will be shown in order to illustrate how this problem can be treated. Different statistical coefficients can be used to quantify the goodness of fit of a model. For example, the mean absolute error or REC curves. The example is a resume of the results presented in [36]. The first step is to calculate the so-called full and restricted models for different values of the parameter h.

As Figure 1a shows, the mixed model, calculated as the average of the errors of the restricted and full models, presents a minimum that can be used to fix the best value of the parameter h in the estimation process.



(**a**) (**b**)

Figure 1. Effect of overfitting and determination of optimum level for parameter h: (**a**) Typical behavior of MAE (Mean Absolute Error) for the full, restricted and mixed models in terms of h; (**b**) Typical REC curve for the null, full and restricted models.

2.3. Computational Algorithm

The second step of the global algorithm corresponds with the numeric integration of the ordinary differential equation. As is well known, there is a variety of general

algorithms and methods and, depending on the concrete ODE under study, some specific methodologies can be considered.

In the present research, the numerical method of integration used in the calculations corresponds to the predictor-corrector method based on the fourth-order multi-step Adams–Bashforth and Adams–Moulton algorithms. In these methods, it is necessary to calculate the unknown functional parameters at the nodes, so the regression model is used to estimate these values.

The exact form for the algorithm depends on the ODE under study. Algorithm 1 corresponds to the differential equation solved presented in Section 3.1:

$$\frac{d}{dx}\left(\alpha(x)\cdot\frac{du}{dx}\right) = f(x). \tag{29}$$

Following Equations (21)–(23), the algorithm can be condensed in the next schema (Algorithm 1), where Euler's method has been used in the numerical solution of the equation:

Algorithm 1. Sampled factors ODE numerical integrator

Input:
N: Number of intervals,
a: Left interval extreme,
b: Right interval extreme,
w: Parameter of radial function,
$x_1^{(s)}, .., x_P^{(s)}$: Sample of independent variable,
$\alpha_1^{(s)}, .., \alpha_P^{(s)}$: Sample of alpha function,
$f_1^{(s)}, .., f_P^{(s)}$: Sample of f function,
Output:
$x_0, .., x_N$: Coordinates of nodes,
$u_0^{(0)}, .., u_N^{(0)}$: Estimated numerical function,
$u_0^{(1)}, .., u_N^{(1)}$: Estimated numerical derivative
Procedure IntegratingODE
$h \leftarrow (b-a)/N$
/* Modeling unknown functions at nodes*/
for i $\leftarrow 0,\ N$ **do**
$x_i \leftarrow x_i + i\cdot h$
$W^0 \leftarrow 0$ See Equation (21)
$S_\alpha \leftarrow 0$ See Equation (22)
$S_f \leftarrow 0$ See Equation (22)
for k $\leftarrow 0,\ P$ **do**
$\rho \leftarrow \Phi\left(x_i, x_k^{(s)}, w\right)$
$W^0 \leftarrow W^0 + \rho$
$S_\alpha \leftarrow S_\alpha + \rho\cdot\alpha_k^{(s)}$
$S_f \leftarrow S_f + \rho\cdot f_k^{(s)}$
$\hat{\alpha}_i \leftarrow S_\alpha / W^0$ See Equation (23)
$\hat{f}_i \leftarrow S_f / W^0$ See Equation (23)
/* Initial conditions */
$u_0^{(1)} \leftarrow y_0'\cdot\hat{\alpha}_0$
$u_0^{(0)} \leftarrow y_0$
/* Calculating numerical solution (Euler's method)*/
for i $\leftarrow 0,\ N-1$ **do**
$u_{i+1}^{(1)} \leftarrow u_i^{(1)} + \hat{f}_i\cdot h$
$u_{i+1}^{(0)} \leftarrow u_i^{(0)} + u_i^{(1)}\cdot h/\hat{\alpha}_i$
return $\{x_0, \ldots, x_N\}, \left\{u_0^{(0)}, \ldots, u_N^{(0)}\right\}, \left\{u_0^{(1)}, \ldots, u_N^{(1)}\right\}$

3. Results

To test the behavior of the proposed methodology, several examples are solved. First, two cases of the one-dimensional static inhomogeneous heat equation will be studied. Later, the Bass equation is analyzed to consider a different example.

Let us denote the solution to the differential equations on an interval $I = [a, b]$ by $u(x)$.

For a given selection of standard deviation (σ) and sample size (P), the functional parameters are sampled and generate the corresponding numerical solutions for the equations. Denoting the set of samples by $S(P, \sigma)$ and each individual sample by $s \in S(P, \sigma)$, the solution can be represented as $u_s(x)$. In the examples, 200 iterations with different samples of the characteristic functions of the equation are run. From this set of solutions, the following statistical indexes are considered:

$$\overline{U}(x, P, \sigma) = \max_{s \in S(P,\sigma)} u_s(x) \tag{30}$$

$$\underline{U}(x, P, \sigma) = \min_{s \in S(P,\sigma)} u_s(x) \tag{31}$$

$$\langle U(P, \sigma) \rangle = u(b) - \langle u_s(b) \rangle_{s \in S(P,\sigma)} \tag{32}$$

$$U_M(P, \sigma) = u(b) - \operatorname*{median}_{s \in S(P,\sigma)} u_s(b). \tag{33}$$

$\overline{U}(x, P, \sigma)$ and $\underline{U}(x, P, \sigma)$ represent the extreme values of the solutions at each point x for a given sample size n and standard deviation σ. The amplitude of the difference $\delta(P, \sigma) = \max_{x \in [a,b]} \left\{ \overline{U}(x, P, \sigma) - \underline{U}(x, P, \sigma) \right\}$, shows the size of variations in the solutions related to a given value of P and σ.

These indexes can be greatly influenced by low frequent extreme values, so additional parameters must be considered. For example, in the case that the exact solution is known, something frequent when studying the behavior of the algorithm, the average (Equation (32)) and median (Equation (33)) of the error calculated over the samples $s \in S(P, \sigma)$ can be a more convenient approach when studying the behavior of the solutions and their dependence on the magnitude of the perturbation.

The testing process followed these steps:

1. Selection of the different standard deviation values $\left\{ \sigma_{[j]} \right\}_{j=1}^{N_\sigma}$ for the perturbation (see next point) with which the study is to be carried out. The following steps should be performed once for each value of $\sigma \in \left\{ \sigma_{[j]} \right\}_{j=1}^{N_\sigma}$ in the above set and as many times as indicated by the number of iterations (see Tables 1–4);
2. Generation of function samples from the uniform distribution $X \sim \mathcal{U}(a, b)$ for the points coordinates and a random normal distribution for the perturbation $\epsilon \sim N(0, \sigma)$ as follows:
$$Y^{(i)} = f^{(i)}(X) + \epsilon, \tag{34}$$
where $f^{(i)}(x)$ represents the i-th functional parameter of the ODE under study and $Y^{(i)}$ is the corresponding sample. The size of each sample is specified in Tables 1–4;
3. Once the number of intervals N has been selected, the model is obtained at the N+1 nodes obtained by dividing the interval $I = [a, b]$ into N sub-intervals;
4. Numerical resolution of the equation using the values of the functional parameters at the nodes obtained in the previous step, by means of an ordinary ODE resolution method, in the present case predictor-corrector;
5. Calculation of statistical values $\overline{U}(x, P, \sigma)$, $\underline{U}(x, P, \sigma)$, $U(P, \sigma)$ and $U_M(P, \sigma)$ from the results obtained in each iteration and their use (in the form of tables or graphs).

3.1. Static Heat Equation for Inhomogeneous Media

The application is based on the one-dimensional case of the second order heat equation, given by:

$$c \cdot \rho \cdot \frac{\partial u}{\partial t} - \frac{\partial}{\partial x}\left(\alpha(x) \cdot \frac{\partial u}{\partial x}\right) = f(x). \tag{35}$$

In steady state, the dependence with time vanishes, obtaining the static heat equation in the form:

$$\frac{d}{dx}\left(\alpha(x) \cdot \frac{du}{dx}\right) = f(x). \tag{36}$$

3.1.1. Example 1

The first example corresponds to the equation:

$$\frac{d}{dx}\left(cos(x) \cdot \frac{du}{dx}\right) = cos(2x), \tag{37}$$

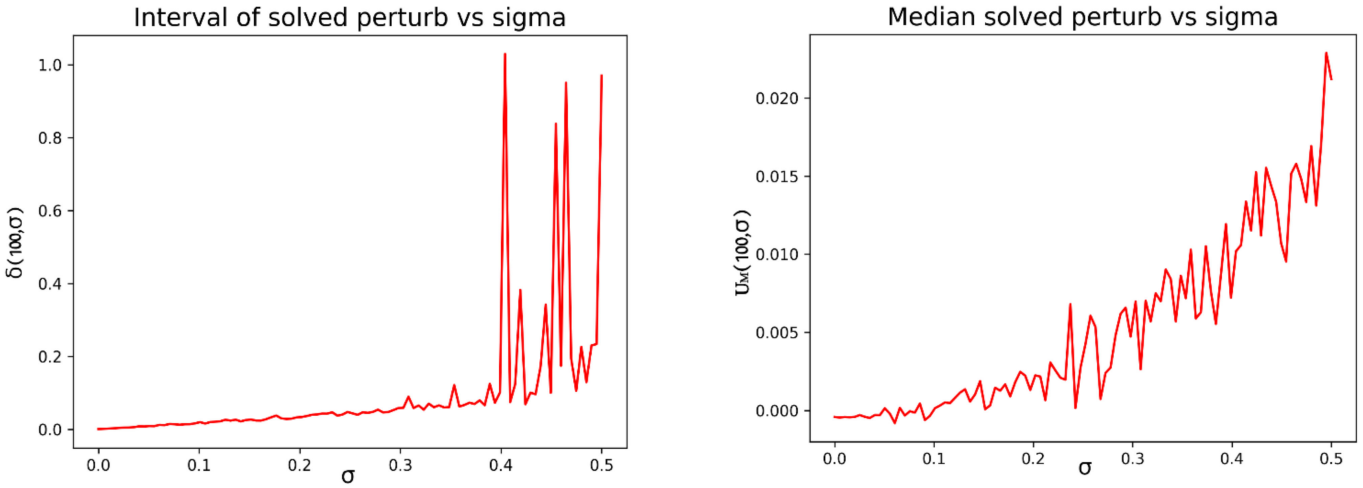
with $u(x)$ defined over the interval $x \in \left[0, \frac{\pi}{4}\right]$ and constrained by the initial conditions $u(0) = 1$ and $u'(0) = 0$.

The functions $\alpha(x)$ and $f(x)$ are perturbed using a random variable following a normal distribution with average zero and standard deviation σ ranging from zero to 0.5. Different samples of these perturbed functions are considered with sizes between 5 and 100.

The equation is solved numerically using a predictor–corrector method (fourth order Adams–Bashforth and Adams–Moulton) in the interval $\left[0, \frac{\pi}{4}\right]$ using a step size of $h = \frac{\pi}{400}$.

The analytical solution for the Equation (37) is $u(x) = 2 - cos(x)$, and is used to compare with the numerical solution $u_s(x)$.

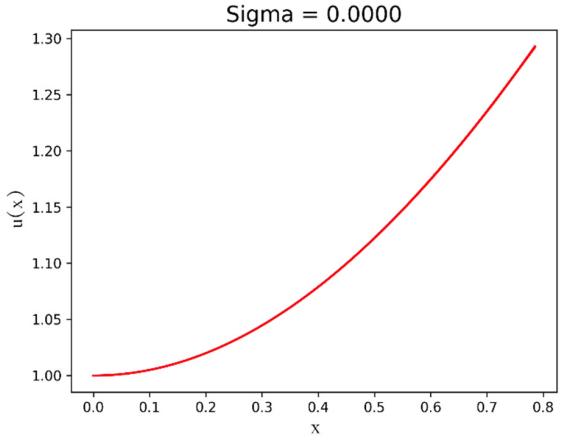
First, we can study the influence of the standard deviation parameter of the perturbation with respect to the differences between the extreme values $\delta(100, \sigma)$. It is also interesting to compare the evolution of $U_M(100, \sigma)$ with respect to the standard deviation. Those results can be seen in Figure 2a,b:



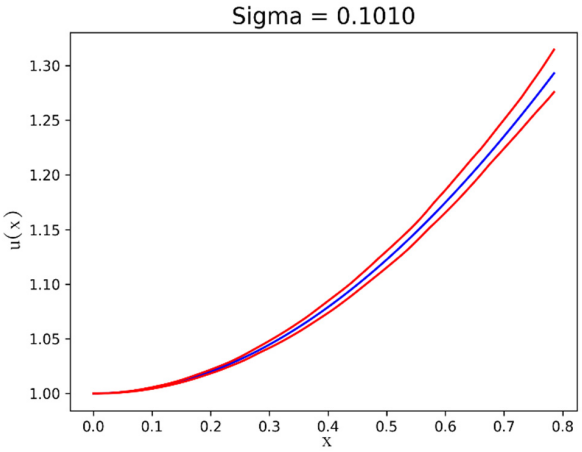
(**a**) (**b**)

Figure 2. Effect on the solution of different standard deviations for 200 samples of the normal perturbation term: (**a**) Range of the differences $\delta(100, \sigma)$; (**b**) Median of the difference between the numerical solutions and the exact solutions at right extreme of the interval $U_M(100, \sigma)$.

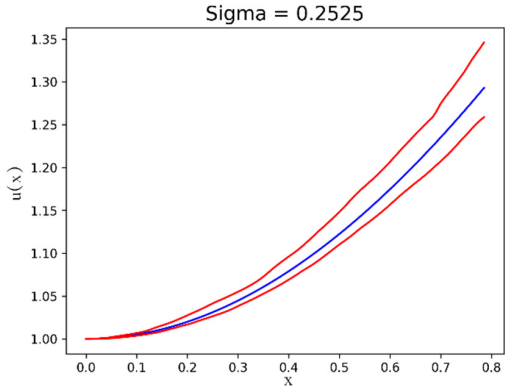
Figure 3 shows the exact $u(x)$ and the maximum $\overline{U}(x,n,\sigma)$ and minimum $\underline{U}(x,n,\sigma)$ of the numerical solutions at each point on for the iterations corresponding to different values of the standard deviation on the normal perturbation term:



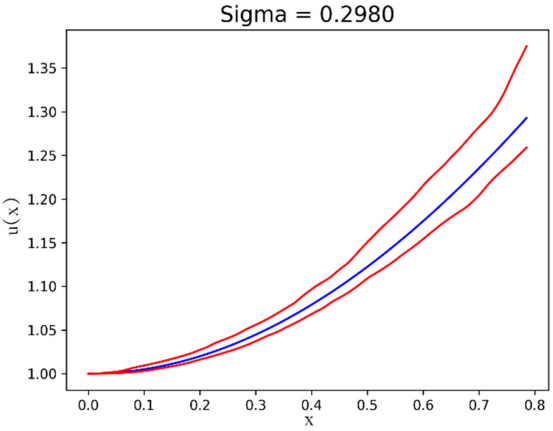
(**a**)



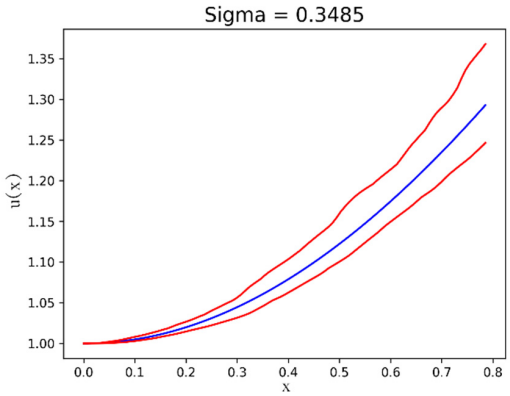
(**b**)



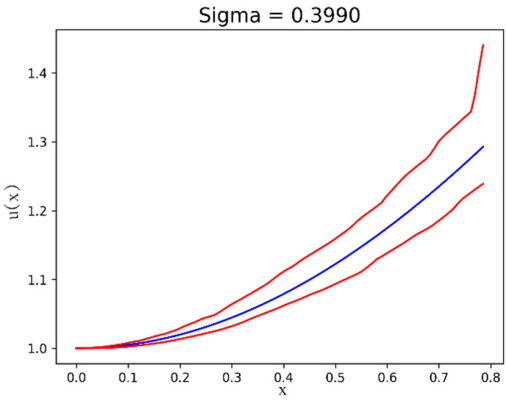
(**c**)



(**d**)

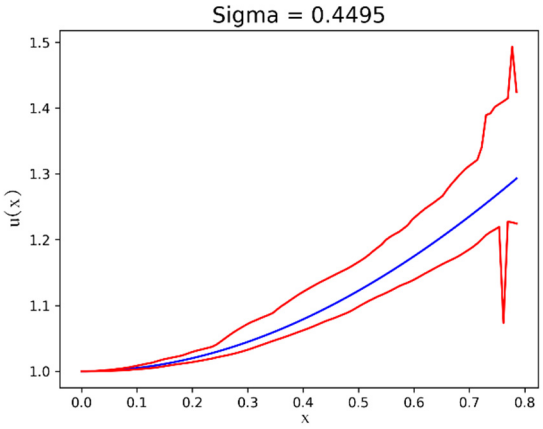


(**e**)

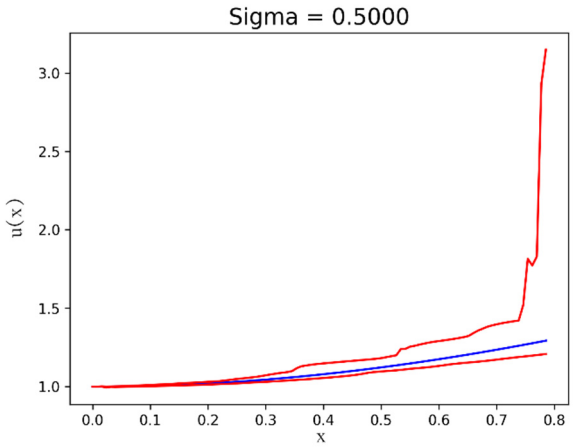


(**f**)

Figure 3. *Cont.*



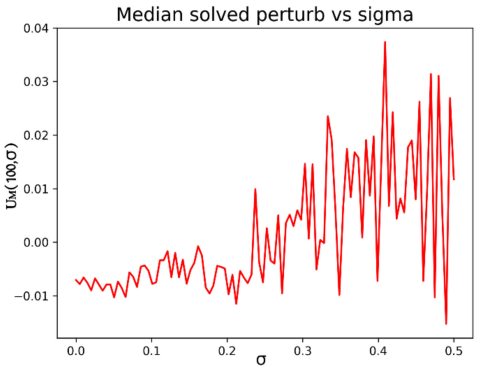
(**g**)



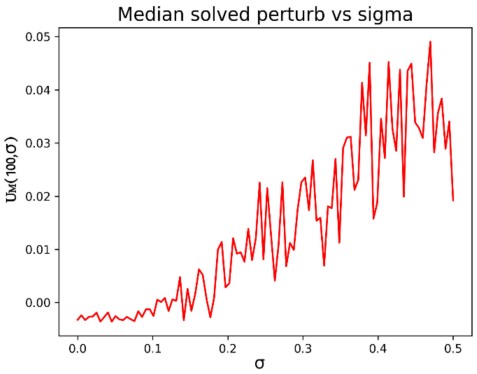
(**h**)

Figure 3. $u(x)$ (blue line), $\overline{U}(x, 100, \sigma)$ (upper red line) and $\underline{U}(x, 100, \sigma)$ (lower red line) after 200 iterations for different values of the standard deviation: (**a**) $\sigma = 0$; (**b**) $\sigma = 0.1010$; (**c**) $\sigma = 0.2525$; (**d**) $\sigma = 0.2980$; (**e**) $\sigma = 0.3485$; (**f**) $\sigma = 0.3990$; (**g**) $\sigma = 0.4495$; (**h**) $\sigma = 0.500$.

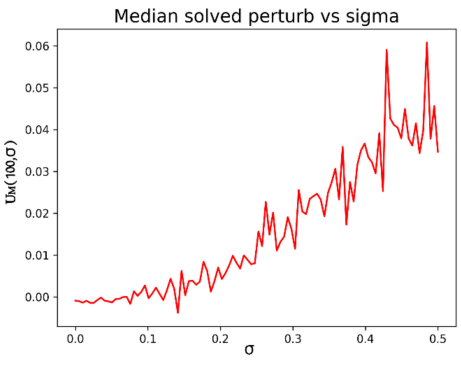
The dependence of the numerical solutions on the sample size can also be considered in order to study the algorithm efficiency. Figure 4 shows the same information that has been presented in Figure 2b for different sample sizes, from 5 to 100.



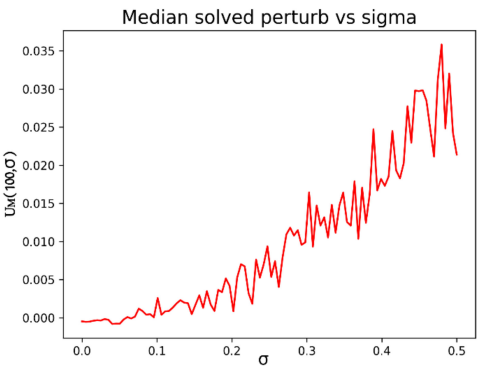
(**a**)



(**b**)

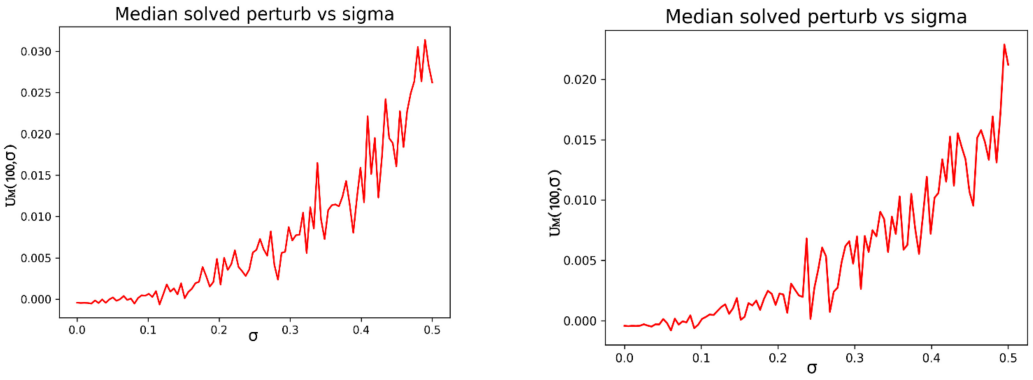


(**c**)



(**d**)

Figure 4. *Cont.*



(**e**) (**f**)

Figure 4. Effect on the parameter $U_M(P,\sigma)$ of different sample sizes for normal perturbation terms with standard deviations from 0 to 0.5: (**a**) P = 5; (**b**) P = 10; (**c**) P = 20; (**d**) P = 50; (**e**) P = 70; (**f**) P = 100.

Table 1. Model parameters for the Example 1.

Sample Size	Std. Dev.	Iterations
100	0–0.5	200
5–100 [1]	0–0.5	200

[1] Results in Figure 4.

In general, the behavior is as expected, obtaining better accuracies as the number of points in the sample increases. However, it can be seen in Figure 4 how, for very small values of P, the result does not present a completely determinate behavior.

3.1.2. Example 2a

Another case of Equation (36) is:

$$\frac{d}{dx}\left(J_n(x)\cdot\frac{du}{dx}\right) = J_m(x), \tag{38}$$

where:

$$J_m(x) = x^m\cdot e^x. \tag{39}$$

The exact solution of this ODE can be obtained as:

$$u(x) = \int \frac{I_m(x)}{J_n(x)}dx = x^{m-n} - m\cdot\int \frac{I_{m-1}(x)}{J_n(x)}dx, \tag{40}$$

where:

$$I_m(x) = \int J_m(x)\,dx. \tag{41}$$

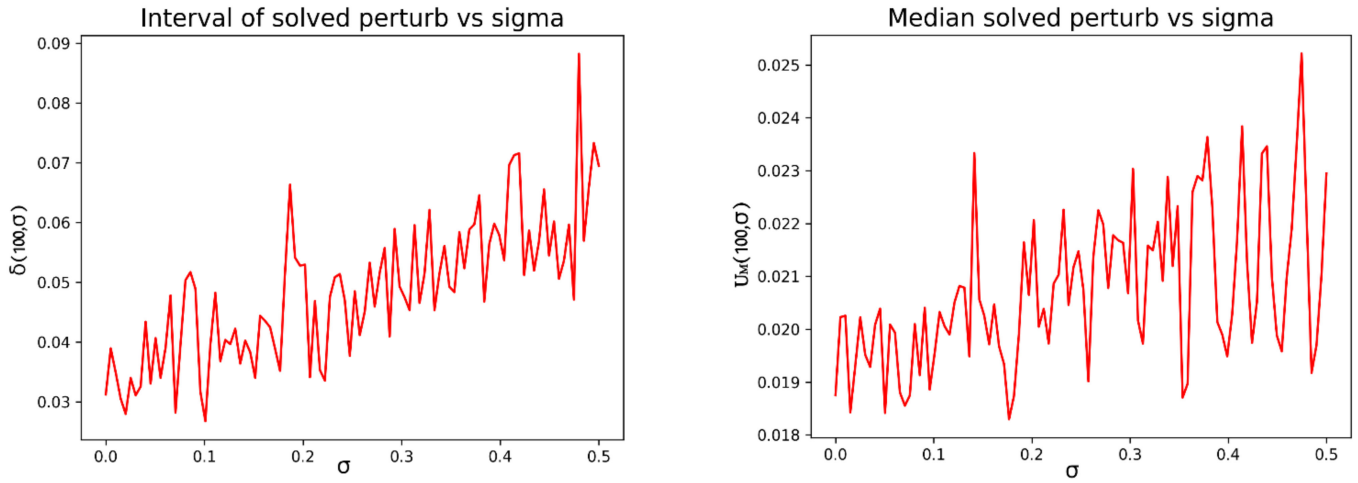
The next two examples will allow comparing the results corresponding to the solutions of the different members of the family of parametric functions $J_n(x)$.

The first case corresponds with the values of the parameters given by $m = 2$ and $n = 4$. Let us study the solution $u(x)$ on the interval $[1,2]$ for the initial conditions $u(1) = -2/3$ and $u'(1) = 1$. The exact solution for the last equation is $u(x) = -\frac{1}{x} + \frac{1}{x^2} - \frac{2}{3}\cdot\frac{1}{x^3}$.

Table 2. Model parameters for the Example 2a.

Sample Size	Std. Dev.	Iterations
100	0–0.5	200

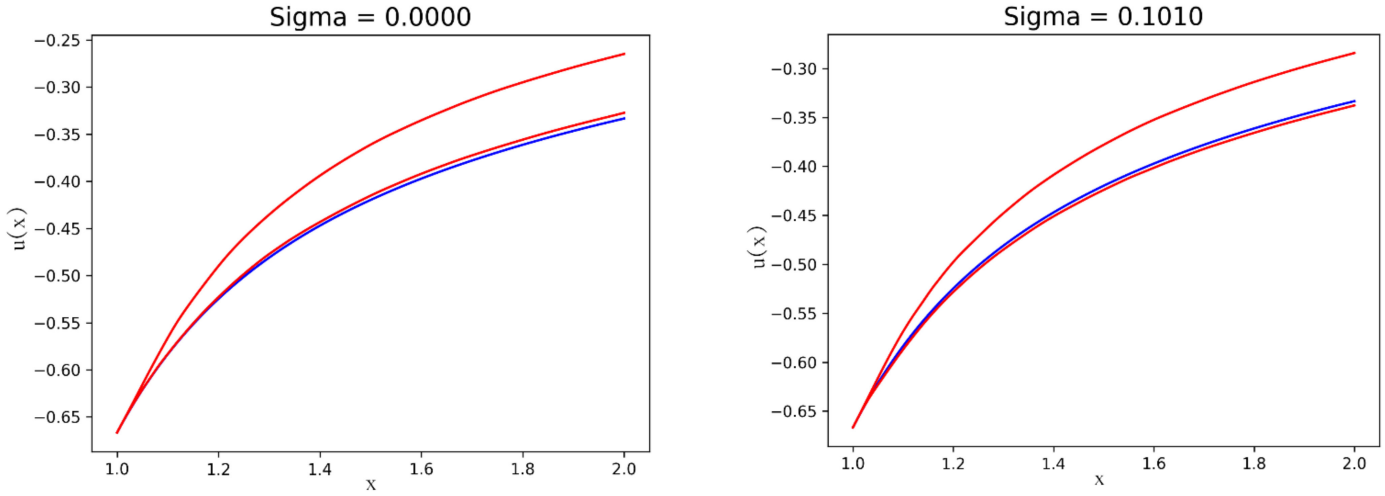
Figure 5 shows the result for the equation given by $m = 2$ and $n = 4$.



(**a**) (**b**)

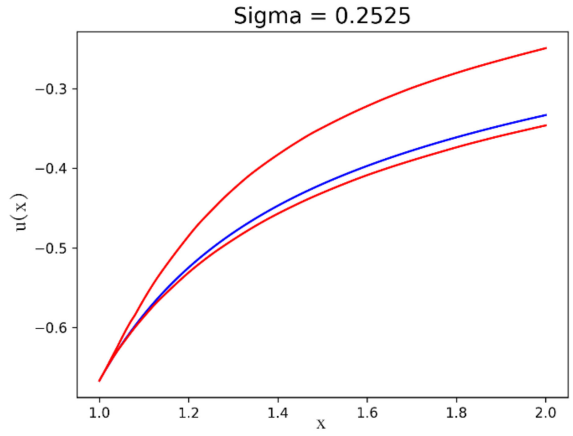
Figure 5. Effect on the solution of different standard deviation for 200 samples of the normal perturbation term: (**a**) Range of the differences $\delta(100, \sigma)$; (**b**) Median of the difference between the numerical solutions and the exact solutions at right extreme of the interval $U_M(100, \sigma)$.

In Figure 6 (x), $\overline{U}(x, 100, \sigma)$ and $\underline{U}(x, 100, \sigma)$ are shown.

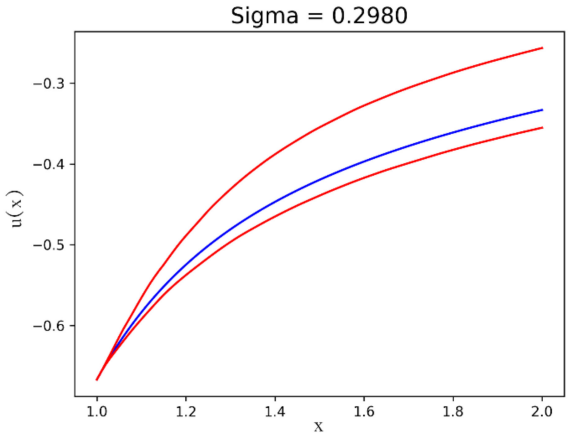


(**a**) (**b**)

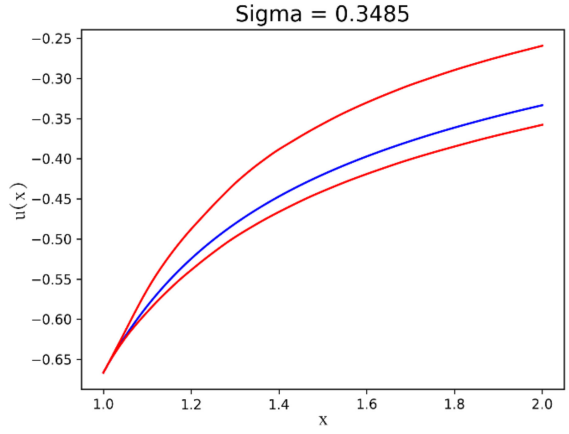
Figure 6. *Cont.*



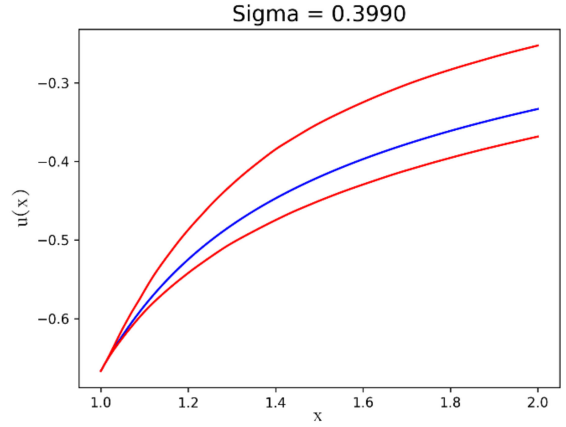
(**c**)



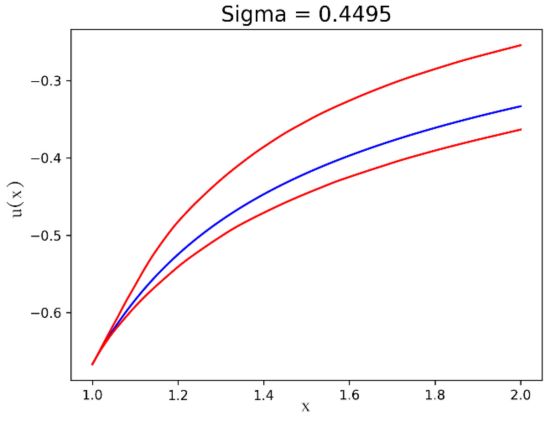
(**d**)



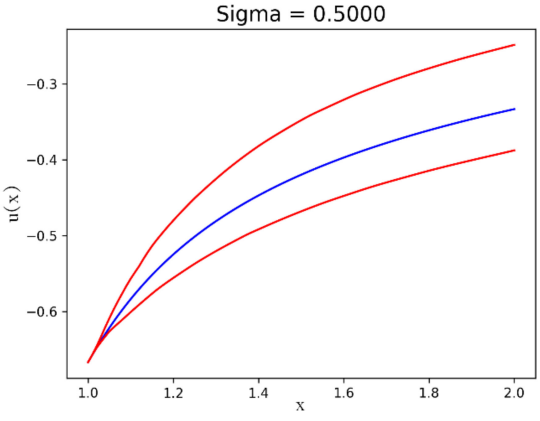
(**e**)



(**f**)



(**g**)

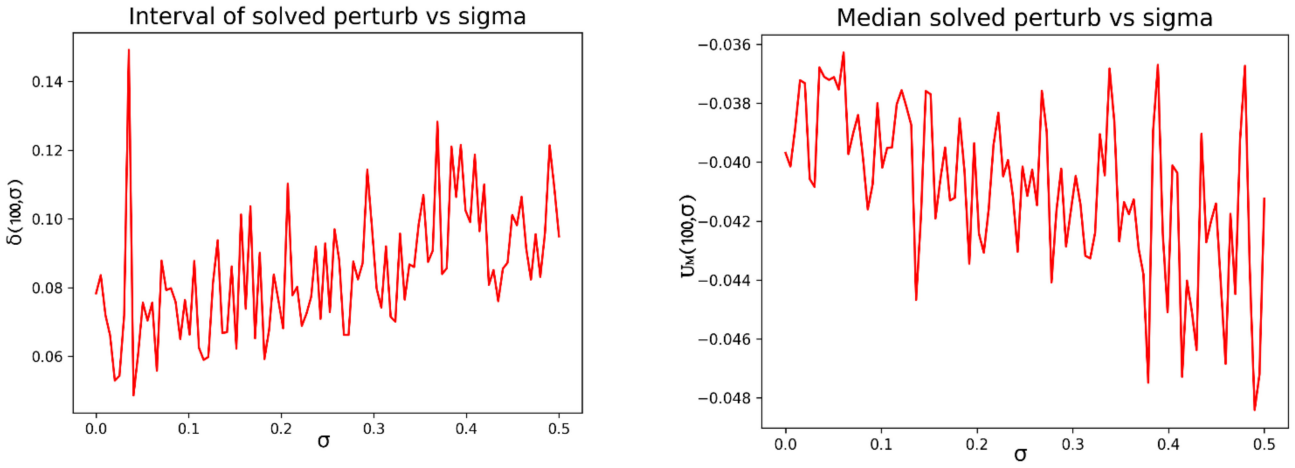


(**h**)

Figure 6. $u(x)$ (blue line), $\overline{U}(x, 100, \sigma)$ (upper red line) and $\underline{U}(x, 100, \sigma)$ (lower red line) after 200 iterations for different values of the standard deviation: (**a**) $\sigma = 0$; (**b**) $\sigma = 0.1010$; (**c**) $\sigma = 0.2525$; (**d**) $\sigma = 0.2980$; (**e**) $\sigma = 0.3485$; (**f**) $\sigma = 0.3990$; (**g**) $\sigma = 0.4495$; (**h**) $\sigma = 0.500$.

3.1.3. Example 2b

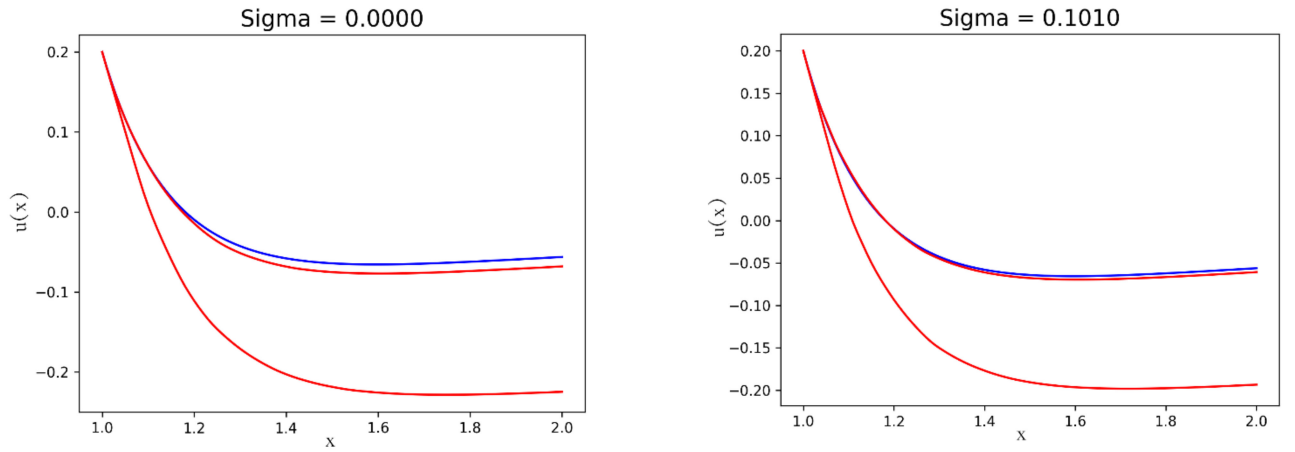
Example 2b corresponds to $m = 3$ and $n = 6$ on the interval $[1, 2]$, with initial conditions $u(1) = 1/5$ and $u'(1) = -2$. In this case, the exact solution is given by the function $u(x) = -\frac{1}{2x^2} + \frac{1}{x^3} - \frac{3}{2x^4} + \frac{6}{5x^5}$. Figure 7 shows the difference $\delta(100, \sigma)$ and the median $U_M(100, \sigma)$.



(**a**) (**b**)

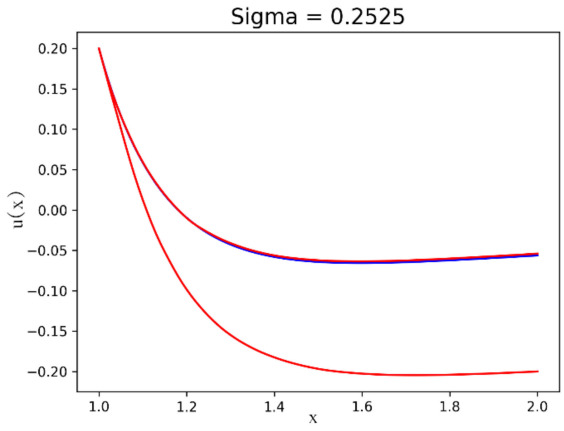
Figure 7. Effect on the solution of different standard deviation for 200 samples of the normal perturbation term: (**a**) Range of the differences $\delta(100, \sigma)$; (**b**) Median of the difference between the numerical solutions and the exact solutions at right extreme of the interval $U_M(100, \sigma)$.

Figure 8 shows the exact and the maximum and minimum numerical solutions for the iterations corresponding to different values of the standard deviation on the normal perturbation term.

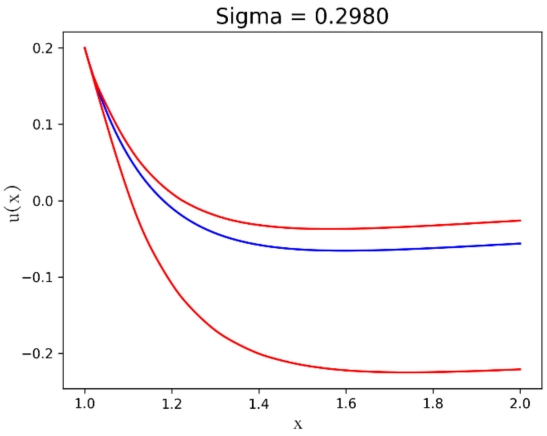


(**a**) (**b**)

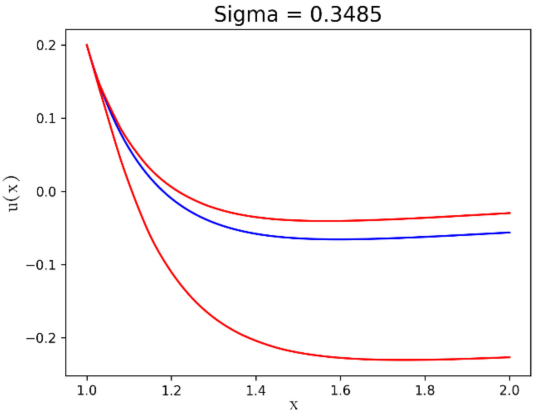
Figure 8. *Cont.*



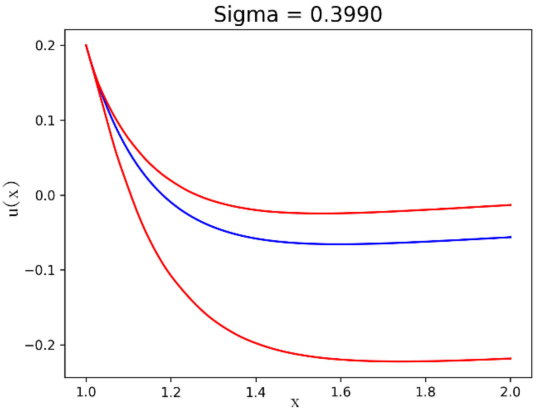
(**c**)



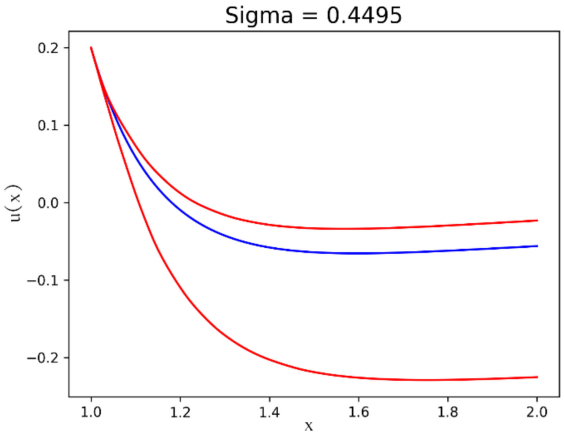
(**d**)



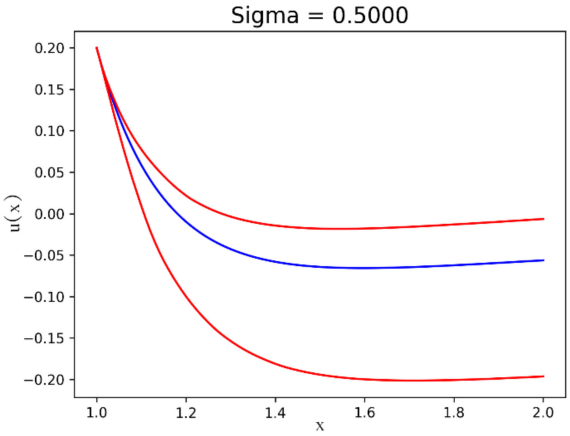
(**e**)



(**f**)



(**g**)



(**h**)

Figure 8. $u(x)$ (blue line), $\overline{U}(x, 100, \sigma)$ (upper red line) and $\underline{U}(x, 100, \sigma)$ (lower red line) after 200 iterations for different values of the standard deviation: (**a**) $\sigma = 0$; (**b**) $\sigma = 0.1010$; (**c**) $\sigma = 0.2525$; (**d**) $\sigma = 0.2980$; (**e**) $\sigma = 0.3485$; (**f**) $\sigma = 0.3990$; (**g**) $\sigma = 0.4495$; (**h**) $\sigma = 0.500$.

Figure 9 presents the effect of sample size on the solution error for sample sizes between 5 and 100 points.

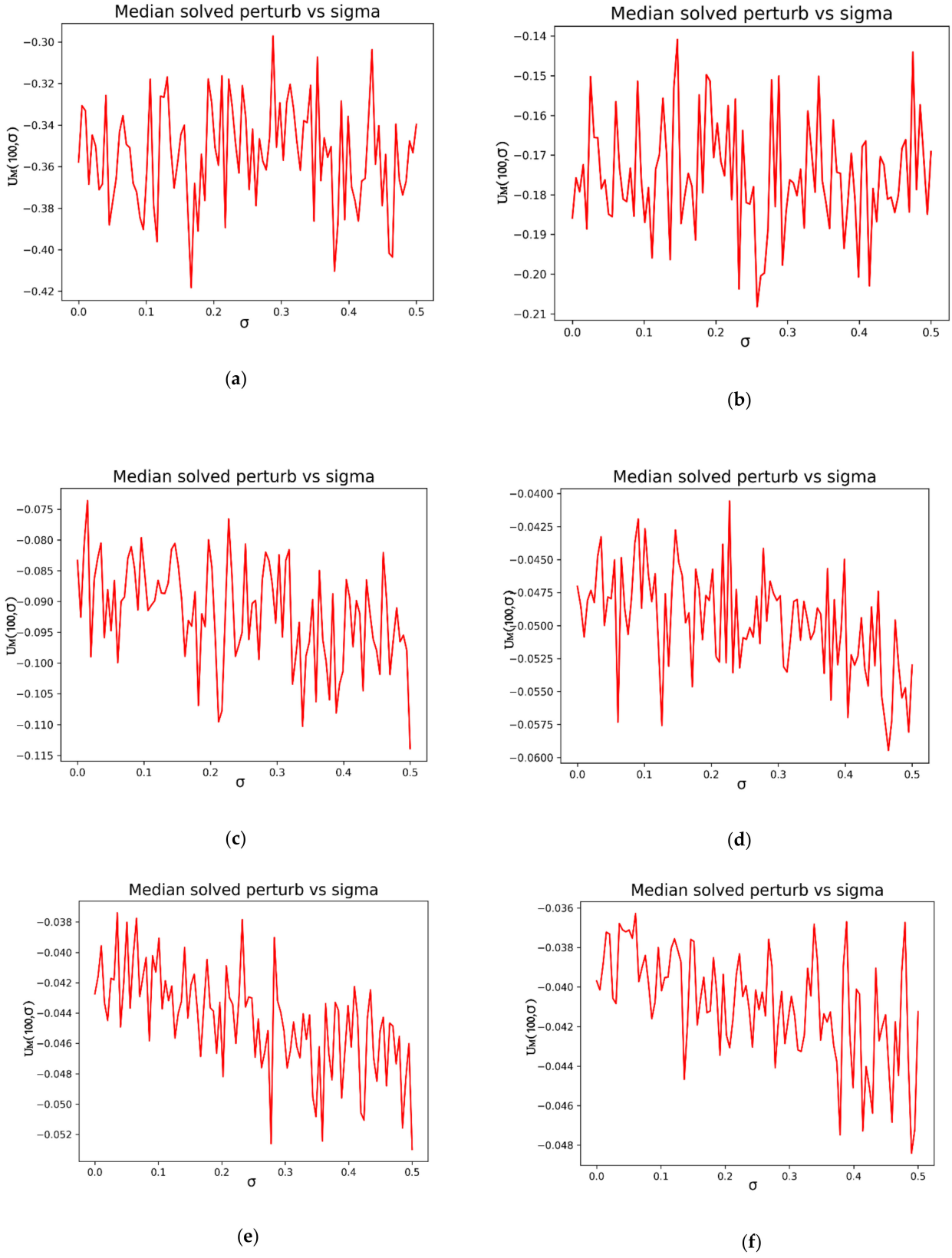


Figure 9. Effect on the parameter $U_M(P, \sigma)$ of different sample sizes for normal perturbation terms with standard deviations from 0 to 0.5: (**a**) P = 5; (**b**) P = 10; (**c**) P = 20; (**d**) P = 50; (**e**) P = 70; (**f**) P = 100.

Table 3. Model parameters for theExample 2b.

Sample Size	Std. Dev.	Iterations
100	0–0.5	200
5–100 [1]	0–0.5	200

[1] Results in Figure 9.

3.2. Bass Equation. Example 3

The third example is a variation of a normalized Bass equation [37] with non-constant coefficients of innovation and imitation:

$$\frac{y'(t)}{1-y(t)} = p(t) + q(t) \cdot y(t), \tag{42}$$

defined over an interval $[0, 5]$ with the initial condition given by $y(0) = 0$. Typical values for the constants in Equation (42), are $p \sim 0.2 \pm 0.1$ and $q \sim 0.4 \pm 0.1$ [38]. In the studied example, two expressions are proposed. For the innovation coefficient:

$$p(t) = 0.2 \cdot e^{-\frac{t}{2}}, \tag{43}$$

and for the imitation coefficient:

$$q(t) = 0.5 \cdot \left[t - \left(\frac{t}{2} \right)^2 \right]. \tag{44}$$

Table 4. Model parameters for the Example 3.

Sample Size	Std. Dev.	Iterations
100	0–0.5	200

Figure 10 represents the parameters $\delta(100, \sigma)$ calculated with 200 different samples of the normal perturbation depending on the values of σ.

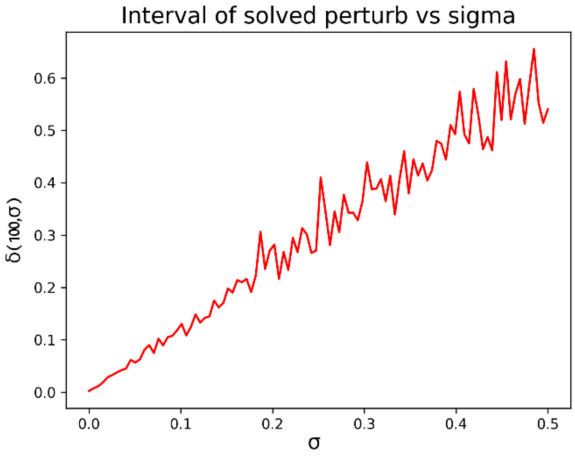
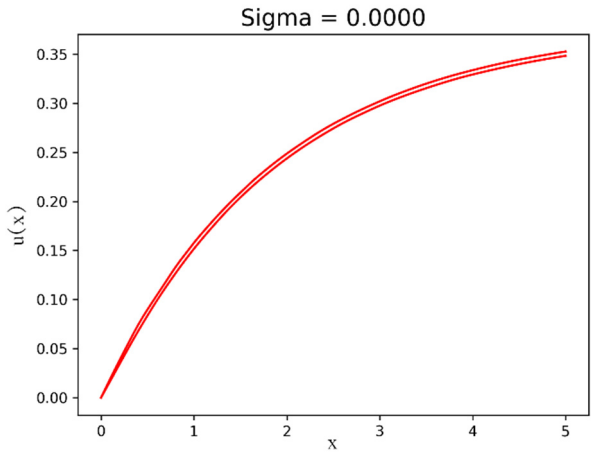
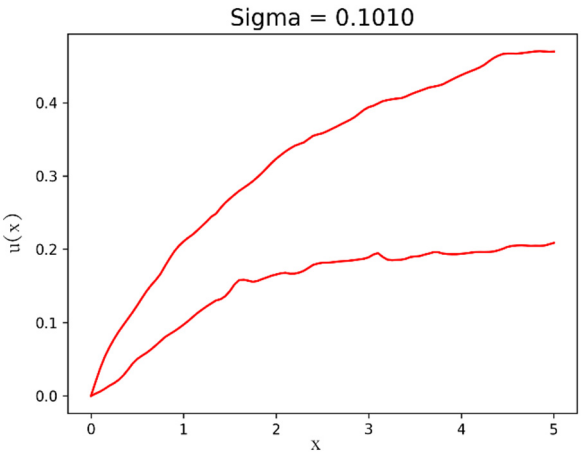


Figure 10. Effect on the solution of different standard deviation for 200 samples of the normal perturbation term for the parameters $\delta(100, \sigma)$.

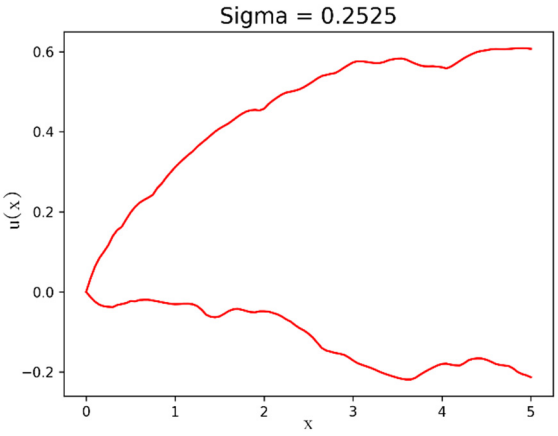
Figure 11 shows the difference between the exact and the maximum and minimum numerical solutions for the iterations corresponding to different values of the standard deviation on the normal perturbation term.



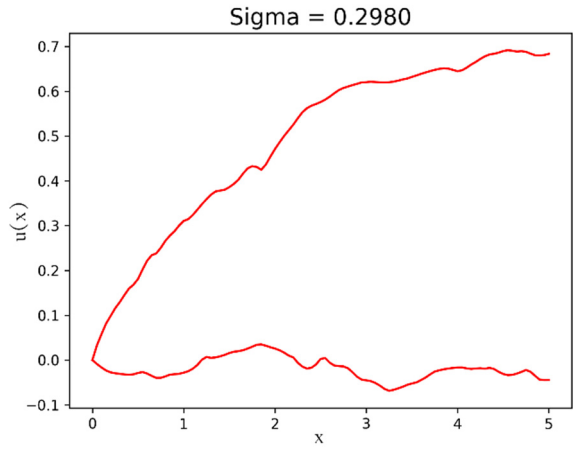
(**a**)



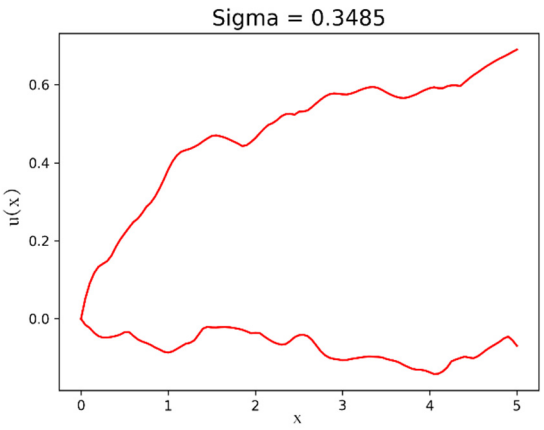
(**b**)



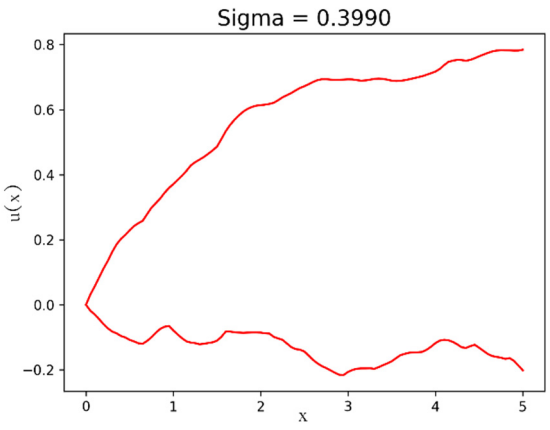
(**c**)



(**d**)

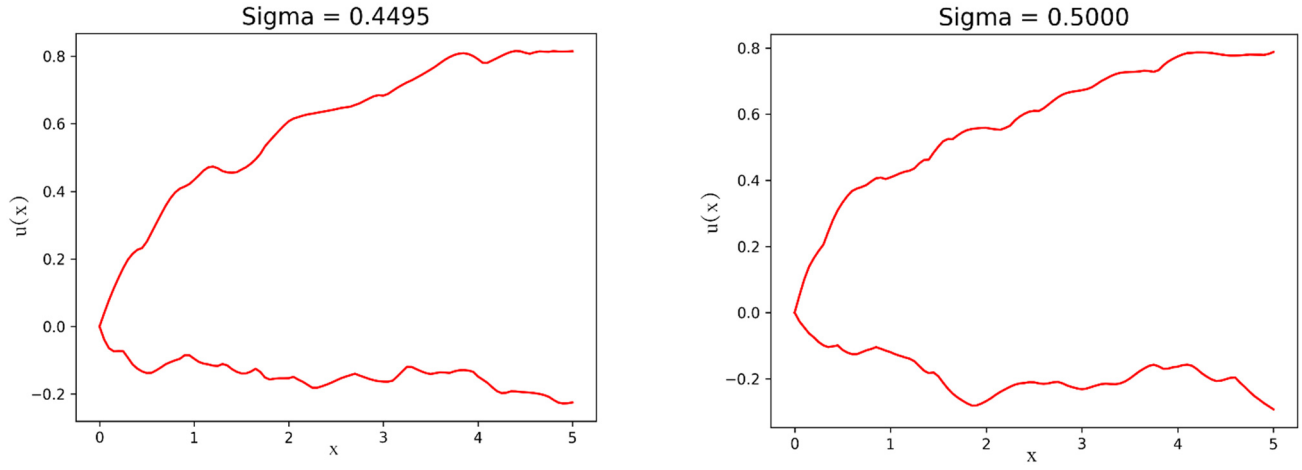


(**e**)



(**f**)

Figure 11. *Cont*.



(**g**) (**h**)

Figure 11. $\overline{U}(x, 100, \sigma)$ and $\underline{U}(x, 100, \sigma)$ after 200 iterations for different values of the standard deviation: (**a**) $\sigma = 0$; (**b**) $\sigma = 0.1010$; (**c**) $\sigma = 0.2525$; (**d**) $\sigma = 0.2980$; (**e**) $\sigma = 0.3485$; (**f**) $\sigma = 0.3990$; (**g**) $\sigma = 0.4495$; (**h**) $\sigma = 0.500$.

In order to facilitate the comparison between the different results, Figure 12 shows together some of the curves represented in Figure 11 for a different execution and function samples.

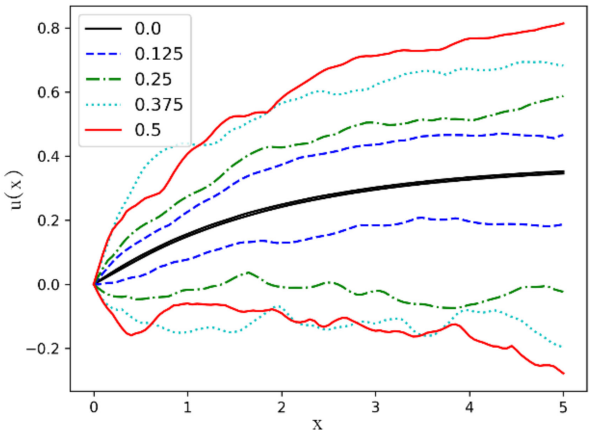


Figure 12. Comparative of $\overline{U}(x, 100, \sigma)$ and $\underline{U}(x, 100, \sigma)$ after 200 iterations for different values of the standard deviation: $\sigma = 0.0$ (black solid), $\sigma = 0.125$ (blue discontinued); $\sigma = 0.25$ (dark green dashed); $\sigma = 0.375$ (light green points) and $\sigma = 0.5$ (red solid).

4. Discussion

In this research, the authors present a methodology for the modeling of differential equations and their resolution in systems where the functional expression of the functional parameters present in these equations is unknown and only an experimental sample of data is available. In the scientific literature, there are references to the determination of parameters of the differential equations governing a system by means of nonlinear regression on experimental data of the solution of the equation under study. Likewise, the field of study of differential equations for which the parameters follow a given distribution constitutes a fertile field of research.

The proposal developed focuses on considering equations whose parameters are unknown functions and whose values are deterministically determined, although affected by some type of error or perturbation. The algorithm allows obtaining numerical solutions

for these equations by performing a previous numerical model of the functions involved from the experimental data.

Three different examples have been considered, two of them with known analytical solutions and another one considering only the numerical solution. From the study of the results obtained, it can be verified that the result of the algorithm shows behavior in accordance with what is expected with respect to the magnitude of the perturbation and the size of the experimental sample used to perform the modeling. On the other hand, the results show how the proposed methodology obtains solutions with a stable response to the corresponding perturbations.

From the present work, the main lines of future research would be two. On the one hand, the modification of the algorithm for the treatment of partial differential equations (PDE). On the other hand, it is worth investigating how to modify the design of the resolution algorithm to make it more computationally efficient. The way to do this would be to study how to combine the modeling algorithm with the resolution algorithm, minimizing the number of operations and the need for storage of the total program.

Author Contributions: Conceptualization, F.J.N.-G. and Y.V.; methodology, F.J.N.-G.; software, F.J.N.-G.; validation, F.J.N.-G. and Y.V.; formal analysis, Y.V.; investigation, F.J.N.-G. and Y.V.; resources, Y.V.; writing—original draft preparation, F.J.N.-G. and Y.V.; writing—review and editing, F.J.N.-G. and Y.V.; visualization, F.J.N.-G.; supervision, Y.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Manteca, I.A., Jr.; Meca, A.S.; Alhama, F. Mathematical characterization of scenarios of fluid flow and solute transport in porous media by discriminated non-dimensionalization. *Int. J. Eng. Sci.* **2012**, *50*, 1–9. [CrossRef]
2. Atkinson, J. *The Mechanics of Soils and Foundations*; Taylor and Francis: New York, NY, USA, 2007.
3. Zienkiewicz, O.C.; Taylor, R.L.; Zhu, J.Z. *The Finite Element Method: Its Basis and Fundamentals*; Elsevier: Amsterdam, The Netherlands, 2005.
4. Graebel, W. *Advanced Fluid Mechanics*; Academic Press: Cambridge, MA, USA, 2007.
5. Cannon, J. *The One-Dimensional Heat Equation, Encyclopedia of Mathematics and Its Applications*; Addison-Wesley: Boston, MA, USA, 1984; ISBN 0-521-30243-9.
6. Rosenbaum, B.; Rall, B.C. Fitting functional responses: Direct parameter estimation by simulating differential equations. *Methods Ecol. Evol.* **2018**, *9*, 2076–2090. [CrossRef]
7. Kalyaev, D.; Zuev, S.M. Algorithms for statistical estimation of coefficients of ordinary differential equations using observational data. *Investig. Antivir. Drug Eff.* **1991**, *6*, 1–23. [CrossRef]
8. Wu, H.; Xue, H.; Kumar, A. Numerical discretization-based estimation methods for ordinary differential equation models via penalized spline smoothing with applications in biomedical research. *Biometrics* **2012**, *68*, 344–352. [CrossRef]
9. Van der Linde, S.C.; Nijhuis, T.A.; Dekker FH, M.; Kapteijn, F.; Moulijn, J.A. Mathematical treatment of transient kinetic data: Combination of parameter estimation with solving the related partial differential equations. *Appl. Catal. A Gen.* **1997**, *151*, 27–57. [CrossRef]
10. Arloff, W.; Schmitt, K.R.; Venstrom, L.J. A parameter estimation method for stiff ordinary differential equations using particle swarm optimization. *Int. J. Comput. Sci. Math.* **2018**, *9*, 419–432. [CrossRef]
11. Abd-Elrady, E.; Schoukens, J. Least squares periodic signal modeling using orbits of nonlinear ODEs and fully automated spectral analysis. *Automatica* **2005**, *41*, 857–862. [CrossRef]
12. Xun, X.; Cao, J.; Mallick, B.; Maity, A.; Carroll, R.J. Parameter estimation of partial differential equation models. *J. Am. Stat. Assoc.* **2013**, *108*, 1009–1020. [CrossRef]
13. Le Cointe, P. Kriging with Partial Differential Equations in Hydrogeology. Master's Thesis, Université Pierre et Marie Curie, Ecole des Mines de Paris & Ecole Nationale du Génie Rural des Eaux et Forêts, Paris, France, 2006.
14. Bochmann, M.; Kämmerer, L.; Potts, D. A sparse FFT approach for ODE with random coefficients. *Adv. Comput. Math.* **2020**, *46*, 1–21. [CrossRef]

15. Poyton, A.A.; Varziri, M.S.; McAuley, K.B.; McLellan, P.J.; Ramsay, J.O. Parameter estimation in continuous-time dynamic models using principal differential analysis. *Comput. Chem. Eng.* **2006**, *30*, 698–708. [CrossRef]
16. Zuev, S.M. Statistical estimation of the coefficients of ordinary differential equations using observational data. *Sov. J. Numer. Anal. Math. Model.* **1986**, *1*, 235–244. [CrossRef]
17. Qi, X.; Zhao, H. Asymptotic efficiency and finite-sample properties of the generalized profiling estimation of parameters in ordinary differential equations. *Ann. Stat.* **2010**, *38*, 435–481. [CrossRef]
18. Tutkun, N. Parameter estimation in mathematical models using the real coded genetic algorithms. *Expert Syst. Appl.* **2009**, *36*, 3342–3345. [CrossRef]
19. Tsitsiashvili, G.; Osipova, M.; Kharchenko, Y. Estimating the coefficients of a system of ordinary differential equations based on inaccurate observations. *Mathematics* **2022**, *10*, 502. [CrossRef]
20. Biegler, L.T.; Damiano, J.J.; Blau, G.E. Nonlinear parameter estimation: A case study comparison. *AIChE J.* **1986**, *32*, 29–45. [CrossRef]
21. Cao, J.; Wang, L.; Xu, J. Robust estimation for ordinary differential equation models. *Biometrics* **2011**, *67*, 1305–1313. [CrossRef]
22. Gulian, M.; Raissi, M.; Perdikaris, P.; Karniadakis, G. Machine learning of space-fractional differential equations. *SIAM J. Sci. Comput.* **2019**, *41*, A2485–A2509. [CrossRef]
23. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Machine learning of linear differential equations using Gaussian processes. *J. Comput. Phys.* **2017**, *348*, 683–693. [CrossRef]
24. Van den Boogaart, K.G. Kriging for processes solving partial differential equations. In Proceedings of the 2001 Annual Conference of the International Association for Mathematical Geology, Cancun, Mexico, 6–12 September 2001.
25. Chen, J.; Chen, Z.; Zhang, C.; Wu, C.F. APIK: Active physics-informed kriging model with partial differential equations. *arXiv* **2020**, arXiv:2012.11798. [CrossRef]
26. Reddy, J.N. *Introduction to the Finite Element Method; Mechanical Engineering*, 4th ed.; McGraw Hill Education: New York, NY, USA, 2019; ISBN 9781259861918 1259861910.
27. Brenner, S.; Scott, R. *The Mathematical Theory of Finite Element Methods*; Springer Science & Business Media: Berlin, Germany, 2007; Volume 15, ISBN 0387759336.
28. Aziz, A.K. *The Mathematical Foundations of the Finite Element Method with Applications to Partial Differential Equations*; Academic Press: Cambridge, MA, USA, 2014; ISBN 1483267989.
29. Babuška, I.; Banerjee, U.; Osborn, J.E. Generalized finite element methods—Main ideas, results and perspective. *Int. J. Comput. Methods* **2004**, *1*, 67–103. [CrossRef]
30. Whiteman, J.R. *The Mathematics of Finite Elements and Applications: Proceedings of the Brunel University Conference of the Institute of Mathematics and Its Applications Held in April 1972*; Academic Press: London, UK, 1973; ISBN 978-0-12-747250-8. Available online: https://ua.on.worldcat.org/oclc/764065 (accessed on 15 December 2021).
31. Zienkiewicz, O.C.; Taylor, R.L.; Taylor, R.L. *The Finite Element Method for Solid and Structural Mechanics*; Elsevier: Amsterdam, The Netherlands, 2005; ISBN 0750663219.
32. Villacampa, Y.; Navarro-González, F.J.; Llorens, J. A geometric model for the generation of models defined in Complex Systems. *WIT Trans. Ecol. Environ.* **2009**, *122*, 71–82.
33. Navarro-González, F.J.; Villacampa, Y. A new methodology for complex systems using n-dimensional finite elements. *Adv. Eng. Softw.* **2012**, *48*, 52–57. [CrossRef]
34. Navarro-González, F.J.; Villacampa, Y. Generation of representation models for complex systems using Lagrangian functions. *Adv. Eng. Softw.* **2013**, *64*, 33–37. [CrossRef]
35. Navarro-González, F.J.; Villacampa, Y. A finite element numerical algorithm for modelling and data fitting in complex systems. *Int. J. Comput. Methods Exp. Meas.* **2016**, *4*, 100. [CrossRef]
36. Navarro-Gonzalez, F.J.; Villacampa, Y. An octahedric regression model of energy efficiency on residential buildings. *Appl. Sci.* **2019**, *9*, 4978. [CrossRef]
37. Bass, F.M. A dynamic model of market share and sales behavior. In Proceedings of the Winter Conference American Marketing Association, Chicago, IL, USA, 27–28 December 1963; pp. 269–275.
38. Massiani, J.; Gohs, A. The choice of Bass model coefficients to forecast diffusion for innovative products: An empirical investigation for new automotive technologies. *Res. Transp. Econ.* **2015**, *50*, 17–28. [CrossRef]