Journal of Computational Electronics (2021) 20:1105–1113
https://doi.org/10.1007/s10825-021-01699-3

The Numerov process over a non-uniform grid

R. Brunetti[1] · N. Speciale[2] · M. Rudan[2]

Received: 10 February 2021 / Accepted: 27 March 2021 / Published online: 13 April 2021

Abstract
The Numerov process is a solution method applicable to some classes of differential equations, that provides an error term of the fifth order in the grid size with a computational cost comparable to that of the finite-difference scheme. In the original formulation of the method, a uniform grid size is required; the paper shows a procedure for extending its applicability to a non-uniform grid in one dimension. The effectiveness of the procedure is tested on a model problem, and comparisons with other methods are carried out. Finally, it is shown how to extend the applicability of the method to a larger class of equations; among these, the mathematical model of semiconductor devices is important in view of its applications to the integrated-circuit technology.

Keywords Numerov process · Non-uniform grid · Differential equations in one dimension · Semiconductor-device model

1 Introduction

The *Numerov Process* (NP [1], cited in [2]), is applicable to the solution of some classes of differential equations; its error term is $O(h^5)$, with h the grid size, much superior to that of the finite-difference scheme, with a comparable computational cost. In the original formulation of NP, a uniform grid is necessary, which is a drawback for many practical applications; more recently, a method for extending the scheme to a variable step in one dimension has been shown [3]. This paper shows an approach, different from that of [3], for extending NP to a non-uniform grid in one dimension; the approach preserves the accuracy of the original NP by combining the latter with a numerical integration having the same order of error.

The interest in highly-accurate schemes derives, among others, from the need of solving the transport model of semiconductor devices, which is fundamental for the integrated-circuit technology. Unfortunately, the form of the semiconductor-device model as it stands is such that NP is not applicable. In the paper, a transformation is shown that makes the semiconductor model tractable with the one-dimensional, variable-step version of NP.

The paper is organized as follows: the extension of NP to a one-dimensional, non-uniform grid is shown in Sect. 2, and the properties of the matrix derived from it are outlined in Sect. 3; the application to a model problem, along with the comparison with the finite-difference scheme, is shown in Sect. 4 using different grids. A comparison with the approach of [3] is carried out in Sect. 5. A more general class of differential equations is considered in Sect. 6; it is shown that, by a suitable transformation of the unknown function, also this class of equations becomes amenable to the application of NP. The example is important because it includes the matematical model of semiconductor devices. Finally, the conclusions are drawn in Sect. 7.

2 The Numerov process over a non-uniform grid

We start by considering the one-dimensional, second-order equation of the form

$$-u'' = F(u, x) , \tag{1}$$

✉ M. Rudan
Massimo.Rudan@unibo.it

R. Brunetti
rossella.brunetti@unimore.it

[1] FIM Department, University of Modena and Reggio Emilia, Via Campi 213/A, I-41125 Modena, Italy

[2] "E. De Castro" Advanced Research Center on Electronic Systems (ARCES) and Department DEI, University of Bologna, Viale del Risorgimento 2, I-40136 Bologna, Italy

where the dependence of F on u may be nonlinear. The standard Numerov Process (NP) requires a uniform grid (the derivation of NP is shown in Appendix I); uniformity is necessary because NP is based on series expansions, and the uniform spacing of the grid points makes the odd powers of the right and left expansions to cancel each other. After eliminating the odd powers, one determines the even powers in terms of u by recursively applying (1); this is also the reason why it is necessary to consider a form like (1) of the equation, where the first derivative of the unknown function does not appear.

To extend NP to a non-uniform grid one must preserve the possibility to cancel, albeit only locally, the odd powers of the expansions. An approach that fulfills this requirement is shown here, which combines a formally exact solution over a non-uniform grid with the improved precision of NP. Consider a one-dimensional, non-uniform grid with N internal nodes, namely $x_0 < x_1 < \cdots < x_N < x_{N+1}$, and let $h_{i+1} = x_{i+1} - x_i$ be the size of one of its elements. With reference to (1), for any $x_{i-1} \le x \le x_i$ it is

$$u'(x) = u'_{i-1} - \int_{x_{i-1}}^{x} F(u,\xi)\,\mathrm{d}\xi\,. \tag{2}$$

Repeating on the next element $x_i \le x \le x_{i+1}$, and integrating again, yields, after some calculations whose details are given in Appendix II,

$$u(x) = u_i + u'_i\,(x - x_i) - \int_{x_i}^{x} (x-\xi)\,F(u,\xi)\,\mathrm{d}\xi\,. \tag{3}$$

Fixing to x_i the second integration limit of (2), and renaming from ξ to x the integration variable transforms (2) into

$$u'_i = u'_{i-1} - \int_{x_{i-1}}^{x_i} F(u,x)\,\mathrm{d}x\,, \qquad i = 1,2,\ldots \tag{4}$$

Similarly, fixing to x_{i+1} the second integration limit of (3), and renaming from ξ to x the integration variable transforms (3) into

$$u_{i+1} = u_i + h_{i+1}\,u'_i - \int_{x_i}^{x_{i+1}} (x_{i+1}-x)\,F(u,x)\,\mathrm{d}x\,, \qquad i = 0,1,\ldots \tag{5}$$

The problem is thus shifted to the calculation of the two integrals

$$G_i = \int_{x_{i-1}}^{x_i} F(u,x)\,\mathrm{d}x\,, \qquad H_{i+1} = \int_{x_i}^{x_{i+1}} (x_{i+1}-x)\,F(u,x)\,\mathrm{d}x\,. \tag{6}$$

Due to the form of F, integrals (6) depend on u, but not on u'. If G_i and H_{i+1} are approximated with formulas where only the nodal values of u appear, namely, $G_i = G_i(u_{i-1}, u_i)$ and $H_{i+1} = H_{i+1}(u_i, u_{i+1})$, the resulting expressions do not introduce extra unknowns with respect to those that explicitly appear in (4) and (5). As shown later, this goal can be accomplished in a manner that exploits the precision of NP.

Assuming that integrals (6) have been calculated, the way in which the nodal values of u are found from (4), (5) depends on the boundary conditions. If the boundary conditions u_0, u'_0 are given, the starting point is calculating u_1 from (5) after letting $i = 0$ there; to calculate u_2 one needs u'_1, which is obtained by letting $i = 1$ in (4); then, one proceeds recursively until the Nth node is reached.

If, instead, the boundary conditions u_0, u_{N+1} are given (which is the typical case in, e.g., the numerical analysis of semiconductor devices), one must eliminate the first derivatives from the system (4), (5). For this, one rearranges (4) to find $G_i = u'_{i-1} - u'_i$, then solves (5) for u'_i to find $u'_i = (u_{i+1} - u_i + H_{i+1})/h_{i+1}$; shifting the index in the latter expression then yields $u'_{i-1} = (u_i - u_{i-1} + H_i)/h_i$ which, combined with the former, provides

$$\frac{u_i - u_{i-1}}{h_i} - \frac{u_{i+1} - u_i}{h_{i+1}} = G_i - \frac{H_i}{h_i} + \frac{H_{i+1}}{h_{i+1}}\,; \tag{7}$$

then, one manipulates the term $G_i - H_i/h_i$ in order to give (7) the more symmetric form

$$\frac{u_i - u_{i-1}}{h_i} - \frac{u_{i+1} - u_i}{h_{i+1}} = \int_{x_{i-1}}^{x_i} \frac{x - x_{i-1}}{h_i} F(u,x)\,\mathrm{d}x + \int_{x_i}^{x_{i+1}} \frac{x_{i+1} - x}{h_{i+1}} F(u,x)\,\mathrm{d}x\,. \tag{8}$$

So far, no approximation has been introduced; to calculate the integrals in (8) one now applies Simpson's rule

$$\int_a^b \omega\,\mathrm{d}x \simeq (b-a)\,\frac{\omega(a) + 4\,\omega(c) + \omega(b)}{6}\,, \tag{9}$$

with c the midpoint of $[a, b]$. The error term of (9) is $\mathrm{O}[(b-a)^5\,\omega^{(4)}]$, with $\omega^{(4)}$ the 4th derivative of ω calculated at some point of the integration interval [4].

Using index $i - 1/2$ to denote the quantities at the midpoint of h_i, index $i + 1/2$ to denote those at the midpoint of h_{i+1}, and applying (9), transforms (8) into

$$-\frac{u_{i-1}}{h_i} + \left(\frac{1}{h_i} + \frac{1}{h_{i+1}}\right) u_i - \frac{u_{i+1}}{h_{i+1}} = \frac{1}{3}\left(h_i\,F_{i-1/2} + \frac{h_i + h_{i+1}}{2}\,F_i + h_{i+1}\,F_{i+1/2}\right). \tag{10}$$

One notes that the left-hand side of (10) corresponds to the discretization of the second derivative obtained from a parabolic interpolation over a non-uniform grid; in turn, the central term of the right-hand side (without the 1/3 factor) is the discretized form of the right-hand side of $-u'' = F$ when the finite-difference method is used (compare with (22)). The improvement inherent in (10) derives from Simpson's rule whose error term, as mentioned above, is $\mathrm{O}(h_i^5)$ in the interval on the left of x_i and $\mathrm{O}(h_{i+1}^5)$ in the interval on the right of it. The issue is to calculate the values $F_{i-1/2}$ and $F_{i+1/2}$, that belong to the midpoint of each interval.

3 The system matrix

From the discussion carried out at the end of Appendix I, one may assume that the equation in hand has been linearized and that the solution is being carried out by iterations; therefore, within the single iteration the dependence of F on u is linear, say, $F = c(x)\,u + s(x)$, whence $F_{i-1/2} = c_{i-1/2}\,u_{i-1/2} + s_{i-1/2}$ and $F_{i+1/2} = c_{i+1/2}\,u_{i+1/2} + s_{i+1/2}$; Eq. (10) then becomes

$$\begin{aligned}
&-\frac{u_{i-1}}{h_i} + \left(\frac{1}{h_i} + \frac{1}{h_{i+1}} - \frac{h_i + h_{i+1}}{6}\,c_i\right) u_i - \frac{u_{i+1}}{h_{i+1}} = \\
&\frac{1}{3}\left(h_i\,c_{i-1/2}\,u_{i-1/2} + h_{i+1}\,c_{i+1/2}\,u_{i+1/2}\right) + \\
&\quad + \frac{1}{3}\left(h_i\,s_{i-1/2} + \frac{h_i + h_{i+1}}{2}\,s_i + h_{i+1}\,s_{i+1/2}\right).
\end{aligned} \tag{11}$$

In (11), the values of c and s at the nodes and at the midpoints of the elements are known, because the two functions are either prescribed or, like in the case considered here, are taken from the previous step of an iterative procedure. In contrast, $u_{i-1/2}$ and $u_{i+1/2}$ are not known; however, u is the solution of (1), for which the NP interpolation (41) holds because the three nodes considered are equally spaced. For $u_{i-1/2}$, replicating (41) over the three nodes in hand yields

$$\begin{aligned}
&-\left(1 + \frac{h_i^2}{48}\,c_{i-1}\right) u_{i-1} + \left(2 - 10\,\frac{h_i^2}{48}\,c_{i-1/2}\right) u_{i-1/2} \\
&\quad - \left(1 + \frac{h_i^2}{48}\,c_i\right) u_i \\
&\quad = \frac{h_i^2}{48}\left(s_{i-1} + 10\,s_{i-1/2} + s_i\right),
\end{aligned} \tag{12}$$

namely, $u_{i-1/2} = a_i^{i-1}\,u_{i-1} + a_i^i\,u_i + h_i^2\,b_i^{i-1/2}$ with

$$\begin{aligned}
&a_i^{i-1} = \frac{48 + h_i^2\,c_{i-1}}{96 - 10\,h_i^2\,c_{i-1/2}}\,, \qquad a_i^i = \frac{48 + h_i^2\,c_i}{96 - 10\,h_i^2\,c_{i-1/2}}\,, \\
&\quad b_i^{i-1/2} = \frac{s_{i-1} + 10\,s_{i-1/2} + s_i}{96 - 10\,h_i^2\,c_{i-1/2}}\,.
\end{aligned} \tag{13}$$

Similarly, $u_{i+1/2} = a_{i+1}^i\,u_i + a_{i+1}^{i+1}\,u_{i+1} + h_{i+1}^2\,b_{i+1}^{i+1/2}$ with

$$\begin{aligned}
&a_{i+1}^i = \frac{48 + h_{i+1}^2\,c_i}{96 - 10\,h_{i+1}^2\,c_{i+1/2}}\,, \qquad a_{i+1}^{i+1} = \frac{48 + h_{i+1}^2\,c_{i+1}}{96 - 10\,h_{i+1}^2\,c_{i+1/2}}\,, \\
&\quad b_{i+1}^{i+1/2} = \frac{s_i + 10\,s_{i+1/2} + s_{i+1}}{96 - 10\,h_{i+1}^2\,c_{i+1/2}}\,.
\end{aligned} \tag{14}$$

Replacing the expressions of $u_{i-1/2}$ and $u_{i+1/2}$ into (11) and rearranging, yields

$$-\beta_i\,u_{i-1} + \alpha_i\,u_i - \gamma_i\,u_{i+1} = \sigma_i + \eta_i\,, \tag{15}$$

with

$$\begin{aligned}
\beta_i &= \frac{1}{h_i} + \frac{a_i^{i-1}\,h_i}{3}\,c_{i-1/2}\,, \\
\gamma_i &= \frac{1}{h_{i+1}} + \frac{a_{i+1}^{i+1}\,h_{i+1}}{3}\,c_{i+1/2}\,,
\end{aligned} \tag{16}$$

$$\begin{aligned}
\alpha_i = &\frac{1}{h_i} + \frac{1}{h_{i+1}} - \frac{a_i^i\,h_i}{3}\,c_{i-1/2} \\
&- \frac{h_i + h_{i+1}}{6}\,c_i - \frac{a_{i+1}^i\,h_{i+1}}{3}\,c_{i+1/2}\,,
\end{aligned} \tag{17}$$

$$\begin{aligned}
\sigma_i &= \frac{h_i}{3}\,s_{i-1/2} + \frac{h_i + h_{i+1}}{6}\,s_i + \frac{h_{i+1}}{3}\,s_{i+1/2}\,, \\
\eta_i &= \frac{b_i^{i-1/2} h_i^3}{3}\,c_{i-1/2} + \frac{b_{i+1}^{i+1/2} h_{i+1}^3}{3}\,c_{i+1/2}\,.
\end{aligned} \tag{18}$$

In conclusion, by combining the Simpson rule with an interpolation of the NP type, integrals (8) have been calculated in terms of the nodal values u_{i-1}, u_i, u_{i+1} only, and the accuracy of NP has been kept. No constraint has been imposed on the elements; as a consequence, the scheme is applicable to a general, non-uniform grid. The outcome of the whole procedure is the $N \times N$ algebraic system (15).

One notes that the system matrix of (15) is tridiagonal like in the finite-difference scheme; however, it is not symmetric because $\gamma_{i-1} \neq \beta_i$ due to the fact that $a_i^{i-1} \neq a_i^i$; the asymmetry is not to be ascribed to the non-uniformity of the grid, nor to the use of NP, but to the presence of term $c\,u$ in the equation to be solved. In fact, even if a uniform grid is used, the matrix resulting from it is still asymmetric (compared with (41)); conversely, if it happens that $c =$ const, then (13) renders $a_i^{i-1} = a_i^i$ and makes the matrix symmetric.

Another observation is that, if one takes the limit of a uniform grid, $h_i \to h$, interpolation (10) becomes

$$-u_{i-1} + 2\,u_i - u_{i+1} = \frac{h^2}{3}\left(F_{i-1/2} + F_i + F_{i+1/2}\right). \tag{19}$$

Apart from the trivial case $F =$ const, limit (19) is different from the interpolation found by directly applying NP to a uniform grid, that is, (35). The reason for the difference is that (35) is obtained by combining two series expansions over adjacent elements, this involving the nodal values of F at x_{i-1}, x_i, and x_{i+1}. In contrast, interpolation (19) is obtained by considering one element at the time and integrating over it a term of the form $(x - x_{i-1})\, F$ or, respectively, $(x_{i+1} - x)\, F$; then, NP is exploited for carrying out the Simpson integration, thanks to its ability to provide the unknown function at the element's midpoint. Combining the results of two adjacent elements finally makes the nodal values of F at $x_{i-1/2}$, x_i, and $x_{i+1/2}$ to appear in (19); the nodal values at x_{i-1} and x_{i+1}, instead, do not enter (19) because the integrands vanish at x_{i-1} or, respectively, at x_{i+1}.

To further compare (19) with (35), one changes the coefficient of the right-hand side of (19) from $h^2/3$ to $h^2/12$; in this way, the weights of the F terms become (4, 4, 4), to be compared with (1, 10, 1) of (35). In both cases, the weights total to 12; however, the lateral weights in (35) are smaller because the lateral nodes of (35) are more distant from the central node x_i than the lateral nodes of (19).

4 Model problem

In this section, we consider a model problem useful for an experimental validation of the procedure outlined in Sects. 2 and 3; we take the interval $0 \le x \le 1$ and assume that the solution of $-u'' = c\, u + s$ is

$$u = \lambda\, \sin(\lambda)\,, \qquad \lambda(x) = k\, \pi\, \frac{1 + p}{1 + p\, x}\,, \tag{20}$$

with k, p two positive integers, so that the boundary conditions are $u(0) = u(1) = 0$. One finds

$$\begin{aligned} c =& (\omega\, \lambda)^2\, (\lambda^2 - 2)\,, \qquad s = -4\, (\omega\, \lambda)^2\, \lambda^2\, \cos(\lambda)\,, \\ \omega =& \frac{p}{k\, \pi\, (1 + p)}\,. \end{aligned} \tag{21}$$

Function (20) is such that the frequency and amplitude of the oscillations increase from right to left; the case corresponding to $k = 2$ and $p = 5$ is considered (function (20) is shown as the green line in Figs. 1 and 2). Equation (20), with coefficients given by (21), is solved on different grids using the generalized NP method of Sects. 2 and 3, or the standard finite-difference scheme (FD); the latter yields

$$\begin{aligned} &- \frac{u_{i-1}}{h_i} + \left(\frac{1}{h_i} + \frac{1}{h_{i+1}} \right) u_i \\ &\quad - \frac{u_{i+1}}{h_{i+1}} = \frac{h_i + h_{i+1}}{2}\, \left(c_i\, u_i + s_i \right)\,. \end{aligned} \tag{22}$$

Each outcome is compared with the true solution (20); more specifically, as error indicators we adopt

$$\epsilon_{\rm FD} = \max_i |u_i^{\rm FD} - u_i|\,, \qquad \epsilon_{\rm NP} = \max_i |u_i^{\rm NP} - u_i|\,, \tag{23}$$

where $u_i^{\rm FD}$ or $u_i^{\rm NP}$ is the value of the solution obtained at the ith node using the FD or NP scheme, and u_i is the value of (20) at the same node.

The first comparison has been carried out by randomly generating the nodal positions within the integration domain; other comparisons have been carried out starting from a uniform grid, whose nodes have subsequently been shifted to obtain a prescribed density of nodes over the integration domain. Specifically, the shift was obtained by defining the new nodal positions y_i, $i = 1, \dots, N$ with the transformation

$$y = \frac{1 + p - \sqrt{1 + p\, (p + 2)\, (1 - x)}}{p}\,, \qquad 0 \le y \le 1\,, \tag{24}$$

where p is the positive integer appearing in the definition (20). Observing that the nodal density of the uniform grid is $Q = 1/(N + 1)$, and that the density $g(y)$ of the shifted grid fulfills the relation $Q\, {\rm d}x = g(y)\, {\rm d}y$, one finds

$$g(y) = \frac{2}{N + 1}\, \frac{1 + p\, (1 - y)}{p + 2}\,. \tag{25}$$

In particular it is $g(0) = (p + 1)\, g(1)$, namely, the ratio $g(0)/g(1)$ is the same as $\lambda(0)/\lambda(1)$, where λ is the parameter appearing in the definition (20) of u. With respect to the original positions pertaining to the uniform grid, transformation (24) shifts the nodes to the left, namely, in the direction where the frequency and amplitude of u increase.

The error indicators (23) are shown in Table 1 for different numbers of grid nodes N; specifically, cols. 1, 2 of the table show the results obtained from the randomly-generated grids, cols. 3, 4 those from the uniform grids, and cols. 5, 6 those from the left-shifted grids (although considering uniform grids is redundant with respect to the scope of this work, the results are shown for comparison with those of the other cases). In each set of simulations, the number of nodes has progressively been reduced until the error indicator (23) reached a value of the same order as the oscillation amplitude of u; after this threshold was reached, the number of nodes was not reduced any more and the simulation was discontinued, which explains the horizontal lines in Table 1.

Considering the general trend of the error indicator, the tables show that, as expected, the error improves from the case of the randomly-generated grids, to that of the uniform grids and, finally, to that of the left-shifted grids. As for the comparison between the two solution methods, it is found in all cases that the non-uniform NP is better by orders of magnitude than the corresponding FD.

Table 1 Error indicators (23) obtained using FD and NP over randomly-generated grids (cols. 1, 2), uniform grids (cols. 3, 4), or left-shifted grids (cols. 5, 6)

	1	2	3	4	5	6
N	ϵ_{FD}	ϵ_{NP}	ϵ_{FD}	ϵ_{NP}	ϵ_{FD}	ϵ_{NP}
5000	1.5	4×10^{-4}	0.4	3.7×10^{-6}	0.2	5×10^{-7}
2500	33	7×10^{-2}	1.7	5.9×10^{-5}	0.6	7×10^{-6}
2000	40	8×10^{-2}	2.7	1.4×10^{-4}	1	2×10^{-5}
1000	–	0.4	12	2.3×10^{-3}	4	3×10^{-4}
750	–	1.0	22	7.2×10^{-3}	7	9×10^{-4}
600	–	3.2	39	1.8×10^{-2}	12	2×10^{-3}
550	–	28	49	2.5×10^{-2}	14	3×10^{-3}
500	–	–	65	3.6×10^{-2}	18	4×10^{-3}
250	–	–	–	0.6	–	7×10^{-2}
150	–	–	–	3.7	–	5×10^{-1}
100	–	–	–	13	–	2.5
75	–	–	–	–	–	7

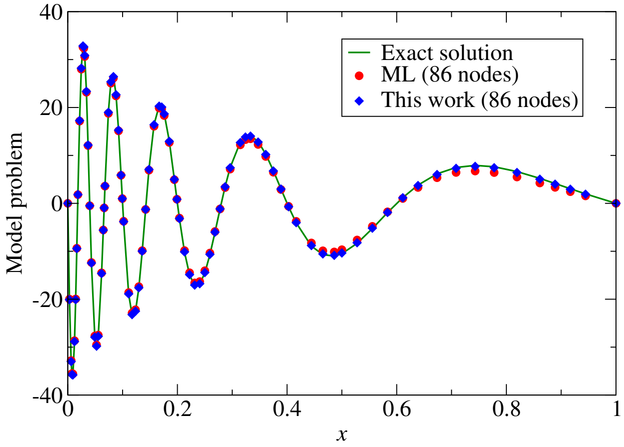

Fig. 1 Comparison between the solution of the model problem obtained over an 86-node grid using ML (red circles) and the method of this work (blue diamonds). The green line shows the exact solution of the same problem (the values of the parameters in (20) are $k = 2$, $p = 5$)

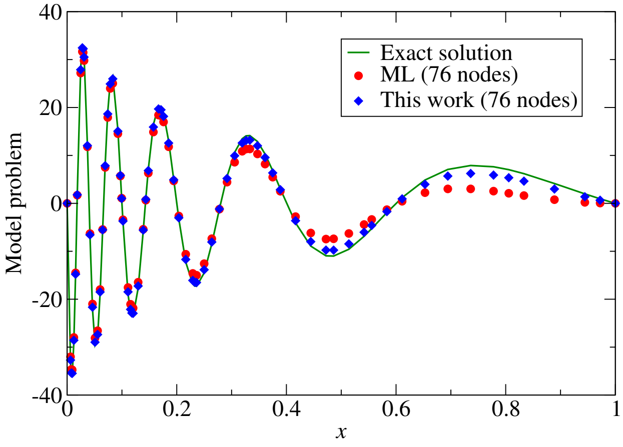

Fig. 2 Comparison between the solution of the model problem obtained over a 76-node grid using ML (red circles) and the method of this work (blue diamonds). The green line shows the exact solution of the same problem (the values of the parameters in (20) are $k = 2$, $p = 5$)

5 Comparison with other approaches

An extension of NP to a non-uniform, one-dimensional grid has been proposed by Vigo-Aguiar and Ramos as a special case of the variable k-step Störmer-Cowell method [3]. One of the outcomes of [3] (indicated here as VR from the authors' initials) is determining a strategy for selecting the length of element h_{i+1}, given $h_1, \dots, h_i$, when solving (1) when the initial conditions $u(x_0) = u_0$, $u'(x_0) = u'_0$ are prescribed; as shown by examples provided in [3], VR is applicable also to boundary-value problems. In the discretization scheme of VR, the left-hand side can be reduced to the form of (10); the right-hand side uses weights in which suitable combinations of the elements appear. Another work showing an extension of NP to a non-uniform, one-dimensional grid is [5], whose scheme is similar to that of VR: the left-hand side is still the same as in (10), whereas the weights at the right-hand side use the single ratio h_{i+1}/h_i; this yields an $\mathrm{O}(h^3)$ error term [6].

In the following, the comparison is carried out using again the model problem $-u'' = c\,u + s$ with coefficients given by (20, 21); the method proposed here is compared with VR and with the boundary-value problem solver of MATLAB® [7] (the latter will be indicated with ML below). In contrast to VR, the scheme of ML considers a system of first-order equations, that are solved by a collocation method with a C^1 piecewise-cubic polynomial; the latter collocates at the ends of each element and at the midpoint, yielding an $\mathrm{O}(h^4)$ error term. It is also worth noting that ML constructs

Table 2 Error indicators (23) obtained with different solution methods (see text)

N	ϵ_{ML}	ϵ_{VR}	ϵ_{VR2}	ϵ_{NP}
986	6.6×10^{-5}	5.9×10^{-1}	1.7×10^{-1}	2.0×10^{-5}
527	1.0×10^{-3}	8.0×10^{-1}	1.0×10^{0}	3.2×10^{-4}
285	1.2×10^{-2}	2.8×10^{0}	4.1×10^{0}	4.1×10^{-3}
153	1.3×10^{-1}	3.0×10^{0}	5.1×10^{0}	5.4×10^{-2}
86	1.2×10^{0}	$2.2 \times 10^{+2}$	$6.0 \times 10^{+1}$	2.1×10^{-1}
76	5.1×10^{0}	$3.3 \times 10^{+2}$	$2.9 \times 10^{+2}$	1.7×10^{0}

Table 3 Simulation time over a uniform grid (units: s)

N	FD	VR	VR2	NP
5, 000	1.6×10^{0}	8.3×10^{-1}	$1.5 \times 10^{+1}$	1.0×10^{0}
2, 500	1.5×10^{-1}	2.0×10^{-1}	1.0×10^{0}	1.8×10^{-1}
2, 000	1.2×10^{-1}	1.0×10^{-1}	4.6×10^{-1}	1.1×10^{-1}
1, 000	4.1×10^{-2}	2.2×10^{-2}	1.1×10^{-1}	2.3×10^{-2}
750	2.1×10^{-2}	1.2×10^{-2}	5.4×10^{-2}	1.2×10^{-2}
500	8.0×10^{-3}	6.0×10^{-3}	2.1×10^{-2}	4.6×10^{-3}
250	1.7×10^{-3}	1.0×10^{-3}	6.0×10^{-3}	1.1×10^{-3}
150	4.7×10^{-4}	4.3×10^{-4}	2.4×10^{-3}	4.9×10^{-4}
100	2.2×10^{-4}	2.2×10^{-4}	2.2×10^{-3}	1.8×10^{-4}
75	1.5×10^{-4}	1.4×10^{-4}	4.3×10^{-4}	1.3×10^{-4}

a non-uniform grid by adding nodes until the specified tolerance is reached; this explains why the numbers of nodes of Table 2 are different from those of Table 1.

Remembering the description of the present method given in Sects. 2, 3, the equation to be solved is recast in integral form and the integral over each element is calculated by the Simpson rule; for the latter, the value of u at the element's midpoint is needed, which is obtained from the NP interpolation (41). It may be argued that the introduction of such a "ghost" point in the middle of each element is equivalent to doubling the number of nodes, which is of advantage when the error indicator (23) is used. For this reason, another set of simulations have been carried out with VR using a double number of nodes with respect to that used with NP; the results of such simulations (which are more expensive in terms of computing time, see Table 3) are labeled VR2. Instead, the doubling of the number of nodes was not used with ML because, as mentioned above, this method uses the elements' midpoint as well.

Table 2 compares the error indicators of type (23) obtained with different solution methods. The table has been obtained by solving (1) using ML with six different values of its internal relative-tolerance parameter. This provided six non-uniform grids, whose numbers of nodes are given in column N; the corresponding error indicators are shown in column ϵ_{ML}. Then, using the same grids, the solutions and the corresponding error indicators have been calculated again using the VR method with N nodes (ϵ_{VR}), the VR method with $2\,N$ nodes (ϵ_{VR2}), and the non-uniform-grid NP method (ϵ_{NP}). The solutions corresponding to the last two lines of Table 2 are shown in Figs. 1 and 2, respectively, for ML and NP, that exhibit the smallest error indicators.

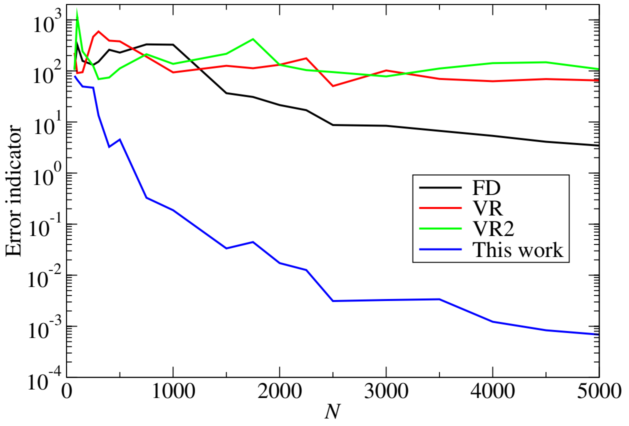

Fig. 3 Error indicators (23) obtained by solving the model problem (20) over randomly-generated grids using the FD (black line), the VR (red line), the VR2 (green line), and the non-uniform NP method (blue line). The horizontal axis shows the number of grid nodes

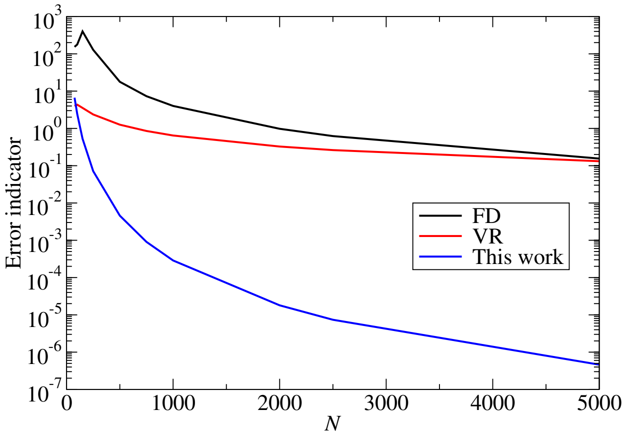

Fig. 4 Error indicators (23) obtained by solving the model problem (20) over initially-uniform grids shifted by applying the (24) scheme, using the FD (black line), the VR (red line), and the non-uniform NP method (blue line). The horizontal axis shows the number of grid nodes

A last set of runs have been carried out to compare ϵ_{FD}, ϵ_{VR}, and ϵ_{NP} over randomly-generated grids (Fig. 3), and over initially-uniform grids shifted by applying the (24) scheme (Fig. 4). Finally, the simulation times of the different methods are compared in Table 3 using a uniform grid by way of example; the calculations have been carried out on a Notebook with an AMD A4-4355N cpu, 1.9 GHz, 4.0 GB RAM.

6 Extension to a larger class of equations

Another important class of second-order equations has the form

$$-g'' + b(x)\, g' = Q(g, x)\,, \tag{26}$$

where b and Q are prescribed. The NP method cannot be applied to (26) as it stands; however, the form of (26) is readily reduced to that of (1) by introducing the auxiliary unknown

$$w = g\, \exp\left[-\frac{1}{2}\int_{x_0}^{x} b(\xi)\,\mathrm{d}\xi\right], \tag{27}$$

with x_0 an arbitrary point; in fact, w fulfills

$$-\,w'' = F(w, x)\,, \qquad F = \left(\frac{b'}{2} - \frac{b^2}{4}\right) w + Q\, \exp\left[-\frac{1}{2}\int_{x_0}^{x} b(\xi)\,\mathrm{d}\xi\right]. \tag{28}$$

An example of equation of the form (26) is given by the transport model of the semiconductor devices, which is important in view of its applications to the integrated-circuit technology; when the transport of electrons is considered, b stands for the normalized electric field and g for the electron concentration. Examples of applications of the NP method to the semiconductor model, over a uniform grid, are given in [8, 9].

7 Conclusions

A method for extending the Numerov process to a non-uniform grid in one dimension has been shown. The result is achieved by acting on each grid element separately: first, the equation to be solved is locally recast in integral form, and the integral over the element is expressed with the Simpson rule; then, the extra unknown allocated at the midpoint of the element is expressed through the Numerov interpolation in terms of the two nodal unknowns belonging to the same element. The error term of the Simpson rule is of the same order as that of the Numerov process, so the accuracy of the latter is preserved. The effectiveness of the scheme has been tested on a model problem, in which both amplitude and oscillation frequency of the solution increase when one of the ends of the integration domain is approached. The method compares favorably with other methods that extend the Numerov Process over a non-uniform grid, and with MATLAB®. In the last part of the paper, it is shown how a suitable transformation of the unknown makes it possible to apply the Numerov process to a larger class of equations, that includes the mathematical model of semiconductor devices.

Appendix I: the Numerov process

An interpolation scheme more refined than the parabolic one is the so-called *Numerov Process* [1] (cited in [2]), which applies to second-order equations of the form

$$-u'' = F(u, x)\,. \tag{29}$$

The essence of the method is that the derivatives of the odd order are canceled, while those of even order are recursively reconstructed in terms of u by means of (29). Considering a uniform grid of size h, the nodal value at u_{i+1} is obtained by means of a Taylor expansion starting from u_i; truncating the expansion to the 5th order yields

$$\begin{aligned} u_{i+1} &\simeq u_i + h\, u_i' + \frac{h^2}{2}\, u_i'' \\ &\quad + \frac{h^3}{6}\, u_i^{(3)} + \frac{h^4}{24}\, u_i^{(4)} + \frac{h^5}{120}\, u_i^{(5)} \end{aligned} \tag{30}$$

and, similarly,

$$\begin{aligned} u_{i-1} &\simeq u_i - h\, u_i' + \frac{h^2}{2}\, u_i'' - \frac{h^3}{6}\, u_i^{(3)} \\ &\quad + \frac{h^4}{24}\, u_i^{(4)} - \frac{h^5}{120}\, u_i^{(5)}\,. \end{aligned} \tag{31}$$

Adding (31) to (30),

$$u_{i-1} - 2\, u_i + u_{i+1} = h^2\, u_i'' + \frac{h^4}{12}\, u_i^{(4)}\,. \tag{32}$$

Now, the same form of the interpolation is obtained if the series expansion is applied to u'' instead of u; one finds in this case

$$u_{i-1}'' - 2\, u_i'' + u_{i+1}'' = h^2\, u_i^{(4)} + \frac{h^4}{12}\, u_i^{(6)}\,. \tag{33}$$

Neglecting the 6th-order term in (33) and eliminating $u_i^{(4)}$ between (32) and (33) yields

$$u_{i-1} - 2\, u_i + u_{i+1} = \frac{h^2}{12}\left(u_{i-1}'' + 10\, u_i'' + u_{i+1}''\right). \tag{34}$$

Finally, letting $F_i = F(u_i, x_i)$ transforms (34) into

$$\begin{aligned} -\, u_{i-1} + 2\, u_i - u_{i+1} &= \frac{h^2}{12}\left(F_{i-1} + 10\, F_i + F_{i+1}\right), \\ i &= 1, 2, \ldots\,. \end{aligned} \tag{35}$$

This interpolation is obtained by keeping terms up to the 5th order; as a consequence, its precision is better than that of the parabolic interpolation. The additional computational cost is due to the right-hand side of (35), whose calculation requires one more multiplication and two more additions with respect to that of the right-hand side of standard finite-difference scheme; the cost of inverting the matrix of the algebraic system (35) is, instead, the same.

As anticipated, the method eliminates (without approximations) the odd-order derivatives and exploits the form of the equation to eliminate the even-order ones. This does not mean that the calculation of the odd-order derivatives is prevented; in fact, subtracting (31) from (30) one obtains

$$u_{i+1} - u_{i-1} = 2\,h\,u_i' + \frac{h^3}{3}\,u_i^{(3)} + \frac{h^5}{60}\,u_i^{(5)} . \tag{36}$$

Neglecting the 5th-order term in (36) and repeating the calculation for u'',

$$u_{i+1}'' - u_{i-1}'' = 2\,h\,u_i^{(3)} + \frac{h^3}{3}\,u_i^{(5)} . \tag{37}$$

Neglecting again the 5th-order term and eliminating $u_i^{(3)}$ between (36) and (37),

$$u_{i+1} - u_{i-1} = 2\,h\,u_i' + \frac{h^2}{6}\left(u_{i+1}'' - u_{i-1}''\right) , \tag{38}$$

which is rearranged as

$$u_i' = \frac{u_{i+1} - u_{i-1}}{2\,h} + \frac{h}{12}\left(F_{i+1} - F_{i-1}\right) . \tag{39}$$

The calculation of the derivative keeps the terms up to the 4th order. It is also important to note that (39) provides the derivative at the ith node, not within one or the other neighboring elements like in the parabolic approximation.

Still considering (29), it is convenient to consider separately the case where the dependence of F on u is linear, namely,

$$F = c\,u + s , \tag{40}$$

where c, s are two given functions of x, from the case where the dependence is nonlinear. In the linear case (35) and (39) become, respectively,

$$\begin{aligned} -\left(1 + \frac{h^2}{12}\,c_{i-1}\right) u_{i-1} + \left(2 - 10\,\frac{h^2}{12}\,c_i\right) u_i \\ - \left(1 + \frac{h^2}{12}\,c_{i+1}\right) u_{i+1} \\ = \frac{h^2}{12}\left(s_{i-1} + 10\,s_i + s_{i+1}\right) , \end{aligned} \tag{41}$$

$$\begin{aligned} u_i' = \left(\frac{1}{2\,h} + \frac{h}{12}\,c_{i+1}\right) u_{i+1} - \left(\frac{1}{2\,h} + \frac{h}{12}\,c_{i-1}\right) \\ u_{i-1} + \frac{h}{12}\left(s_{i+1} - s_{i-1}\right) . \end{aligned} \tag{42}$$

In the nonlinear case, which is typical of, e.g., the mathematical model of semiconductor devices, one must preliminarily linearize (29) starting from a tentative solution $\bar{u}$; specifically, one lets $u = \bar{u} + \delta u$, this transforming (29) into

$$-\bar{u}'' - \delta u'' = F(\bar{u} + \delta u, x) \simeq F(\bar{u}, x) + \left(\frac{\partial F}{\partial u}\right)_{\bar{u}} \delta u , \tag{43}$$

which must then be solved by iterations. When dealing with (43) one may still use (41) and (42) provided the following replacements are used:

$$u \leftarrow \delta u , \qquad c \leftarrow \left(\frac{\partial F}{\partial u}\right)_{\bar{u}} , \qquad s \leftarrow \bar{u}'' + F(\bar{u}, x) . \tag{44}$$

At convergence, $\delta u = 0$ and $-\bar{u}'' = F(\bar{u}, x)$.

Appendix II: calculation of the integrals of (2) and (3)

Integrating (1) from an arbitrary point x_0, and letting $u_0' = u'(0)$, yields

$$u' = u_0' - Y(x) , \qquad Y(x) = \int_{x_0}^{x} F[u(\xi), \xi]\,\mathrm{d}\xi . \tag{45}$$

Integrating again, still from x_0, and letting $u_0 = u(x_0)$, provides

$$u = u_0 + u_0'\,(x - x_0) - \int_{x_0}^{x} Y(\xi)\,\mathrm{d}\xi . \tag{46}$$

On the other hand,

$$\frac{\mathrm{d}[\xi\,Y(\xi)]}{\mathrm{d}\xi} = Y(\xi) + \xi\,\frac{\mathrm{d}Y(\xi)}{\mathrm{d}\xi} = Y(\xi) + \xi\,F[u(\xi), \xi] . \tag{47}$$

Integrating (47) between x_0 and x, and using $Y(x_0) = 0$, yields

$$x\,Y(x) = \int_{x_0}^{x} Y(\xi)\,\mathrm{d}\xi + \int_{x_0}^{x} \xi\,F[u(\xi), \xi]\,\mathrm{d}\xi . \tag{48}$$

Function $Y(x)$ at the left-hand side of (48) is taken from the second of (45), so that (48) becomes

$$\int_{x_0}^{x} Y(\xi)\,\mathrm{d}\xi = \int_{x_0}^{x} (x - \xi)\,F[u(\xi), \xi]\,\mathrm{d}\xi . \tag{49}$$

Combining (46) with (49) provides

$$u = u_0 + u_0'\,(x - x_0) - \int_{x_0}^{x} (x - \xi)\,F[u(\xi), \xi]\,\mathrm{d}\xi . \tag{50}$$

Letting $x_0 = x_{i-1}$ in (45) yields (2); similarly, letting $x_0 = x_i$ in (50) yields (3).

Author Contributions Authors' contributions: Equal contributions.

Funding Open access funding provided by Alma Mater Studiorum - Università di Bologna within the CRUI-CARE Agreement. Not applicable.

Availability of data and material (data transparency): Data and material are available.

Declarations

Conflicts of interest Not applicable.

Code availability: Code is available.

References

1. Numerov, B.: A new method for determining the orbits and calculating the ephemerides accounting for the perturbations. Publ. Obs. Central Astrophys. **2**, 188–288 (1923). **(in Russian)**
2. Chow, P.C.: Computer solutions to the Schrödinger equation. Am. J. Phys. **40**, 730–734 (1972)
3. Vigo-Aguiar, J., Ramos, H.: A variable-step Numerov method for the numerical solution of the Schrödinger equation. J. Math. Chem. **37**(3), 255–262 (2005)
4. Press, W.H., Flannery, B.R., Teukolsky, S.A., Wetterling, W.T.: Numerical recipes: the art of scientific computing. Cambridge University Press, New York (1988)
5. Zlotnik, A.: Čiegis, R.: A compact higher-order finite-difference scheme for the wave equation can be strongly non-dissipative on non-uniform meshes. Appl. Math. Lett. **115**, 106949 (2021)
6. Jain, M.K., Iyengar, S.R.K., Subramanyam, G.S.: Variable mesh methods for the numerical solution of two-point singular perturbation problems. Comput. Methods Appl. Mech. Eng. **42**, 273–286 (1984)
7. Kierzenka, J., Shampine, L.F.: A BVP solver based on residual control and the MATLAB PSE. ACM Trans. Math. Softw. **27**(3), 299–316 (2001)
8. Speciale, N., Brunetti, R., Rudan, M.: Extending the Numerov process to the semiconductor transport equations. In: Esseni, D., Palestri, P., Rideau, D. (eds.) Proceedings of the 2019 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD 2019), pp. 37–40. IEEE, Udine, Italy (2019)
9. Speciale, N., Brunetti, R., Rudan, M.: Solution of the semiconductor-device equations by the Numerov process. Adv. Sci. Technol. Eng. Syst. J. **5**(6), 1414–1421 (2020)