



HAL
open science

Extending nonstandard finite difference schemes rules to systems of nonlinear ODEs with constant coefficients

Marc E. Songolo, Brigitte Bidégaray-Fesquet

► To cite this version:

Marc E. Songolo, Brigitte Bidégaray-Fesquet. Extending nonstandard finite difference schemes rules to systems of nonlinear ODEs with constant coefficients. *Journal of Difference Equations and Applications*, 2024, pp.1-26. 10.1080/10236198.2023.2300726 . hal-03283849

HAL Id: hal-03283849

<https://hal.science/hal-03283849>

Submitted on 12 Jul 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Extending nonstandard finite difference schemes rules to systems of nonlinear ODEs with constant coefficients

Marc E. Songolo^{a**} and Brigitte Bidégaray-Fesquet^b

^aDepartment of Mathematics and Computer Science,
University of Lubumbashi, 1825 Lubumbashi, Republic Democratic of the Congo;
ICAM School of Engineering-Nantes campus, 44470 Carquefou, France;
IETR, Polytech Nantes, University of Nantes, 44306 Nantes, France;
^bUniv. Grenoble Alpes, Grenoble INP^{††} LJK, 38000 Grenoble, France

ABSTRACT. In this paper, we present a reformulation of Mickens' rules for nonstandard finite difference (NSFD) scheme to adapt them to systems of ODEs. This leads to exact schemes in the linear case, and also improve the accuracy in the nonlinear case. In the Hamiltonian nonlinear case, it consists in adding correction terms to schemes derived by Mickens.

KEYWORDS. Ordinary Differential Equations, Matrix exponential, Exact finite difference schemes, Nonstandard finite difference schemes.

1 Introduction

The nonstandard finite difference (NSFD) method was created to overcome the defects of numerical instabilities presented by traditional methods such as Euler, Runge–Kutta, etc. The numerical instabilities indicate that the discrete equations are not able to model correctly the qualitative properties of the solutions to the differential equations [Mic94]. The rules for constructing NSFD models were derived from the construction of exact schemes for certain equations. These rules were then applied to many types of ordinary differential equations (ODE) and to partial derivatives in order to obtain stable schemes which preserve the qualitative properties of the equations. The reference [Pat16] reviews many recent developments and applications of NSFD schemes.

In this paper, we propose to revisit some of Mickens' rules in the light of recent works [Cie13, CR10, CR11, QT18, SBF18, SBF21]. Especially the second rule, that relates to the renormalization of the discretization step-size, does not *a priori* take into account coupling factors between the equations

**Corresponding author. Email: marc.songolo@gmail.com

††Institute of Engineering Univ. Grenoble Alpes

in a differential system. We intend here to extend Mickens' rules to the case of systems of ODEs where we split a linear and a nonlinear part $X' = AX(t) + B(X(t))$.

The paper is organized as follows: we present Mickens' basic rules in Section 2. Section 3 displays the construction of NSFD schemes for systems of ODEs, from a matrix formulation to a derivation of scalar forms. This leads to define two correctors, the effect of which we explore on two examples in Section 4. The discussion then sets out the possible difficulties in developing this strategy and the way to overcome them.

2 The Nonstandard Finite Difference Context

2.1 Nonstandard Finite Difference Rules

In this section, we first give the rules for the construction of NSFD schemes as proposed by Mickens [Mic94].

Rule 1. *The order of the discrete derivatives must be exactly equal to the order of the corresponding derivatives of the differential equations.*

Rule 2. *Denominator functions for the discrete derivatives must, in general, be expressed in terms of more complicated functions of the step-sizes than those conventionally used.*

Rule 3. *Nonlinear terms must, in general, be modeled non-locally on the computational grid or lattice.*

Rule 4. *Special solutions of the differential equations should also be special (discrete) solutions of the finite difference models.*

Rule 5. *The finite difference equations should not have solutions that do not correspond exactly to solutions of the differential equations.*

These rules initially apply to single differential equations. Already in [Mic94] the case of Hamiltonian equations treated as systems of two first order equations makes use of a slightly modified version of Rule 2. Indeed, the derivatives are approximated by

$$\frac{dx}{dt} \simeq \frac{x_{k+1} - \psi(\Delta t)x_k}{\phi(\Delta t)}, \quad (1)$$

where ψ and ϕ are the functions of the step-size Δt and the parameters of the equations. As suggested by Rule 2, the denominator $\phi(\Delta t)$ plays the role of the step size and is such that

$$\phi(\Delta t) = \Delta t + \mathcal{O}(\Delta t^2) \text{ as } \Delta t \rightarrow 0.$$

There is an additional function ψ , which is not mentioned in Rule 2 and is close to identity, namely

$$\psi(\Delta t) = 1 + \mathcal{O}(\Delta t^2) \text{ as } \Delta t \rightarrow 0.$$

2.2 Nonstandard, Exact, and Best Finite Difference Schemes

Definition 1 (Nonstandard finite difference scheme, [Mic00]). *A nonstandard finite difference scheme is any discrete representation of a system of differential equations that is constructed based on the above rules.*

Originally, in [Mic94], this was the definition of a *best finite difference scheme*. Indeed these rules have been defined to have *exact finite difference schemes* thus avoiding the usual questions about consistency, stability and convergence. This more or less involves that we know exact solutions of the equations (see Rules 4 and 5), which is of course not the case in general. It is however expected that schemes constructed with these rules would lead to the "best" schemes.

3 Nonstandard Finite Difference Models

We address systems of ordinary differential equations where we separate a linear and a nonlinear part:

$$X' = AX(t) + B(X(t)), \quad (2)$$

where $t \in [0, T]$, $X(t) \in \mathbb{R}^n$, $A \in \mathcal{M}_{n \times n}(\mathbb{R})$, and $B \in \mathcal{C}^0(\mathbb{R}^n, \mathbb{R}^n)$.

The analytical solution to system (2) can be expressed in integral form by

$$X(t + \Delta t) = e^{\Delta t A} X(t) + \int_t^{t+\Delta t} e^{(t+\Delta t-s)A} B(X(s)) ds. \quad (3)$$

To go further in the explicit computations, we approximate $B(X(s))$ on the time interval $[t, t + \Delta t]$ by a function of $X(t)$ and $X(t + \Delta t)$:

$$B(X(s)) \simeq \mathcal{B}(X(t), X(t + \Delta t)).$$

Inserting this in (3)

$$\begin{aligned} X(t + \Delta t) &\simeq e^{\Delta t A} X(t) + \int_t^{t+\Delta t} e^{(t+\Delta t-s)A} ds \mathcal{B}(X(t), X(t + \Delta t)) \\ &= e^{\Delta t A} X(t) + (e^{\Delta t A} - I)A^{-1} \mathcal{B}(X(t), X(t + \Delta t)), \end{aligned}$$

where I is the identity matrix in $\mathcal{M}_{n \times n}(\mathbb{R})$. The simplest numerical method obtained by this formula is the *exponential Euler approximation* for which $\mathcal{B}(X(t), X(t + \Delta t)) = B(X(t))$:

$$X_{k+1} = e^{\Delta t A} X_k + (e^{\Delta t A} - I)A^{-1} B(X_k),$$

where X_k is approximating $X(t_k)$, $t_k = k\Delta t$ for $k \in \mathbb{N}$. This method makes use of a matrix exponential and is hence called *exponential integrator* [HO10]. Such an approximation for the nonlinear part does however not fulfill Rule 3 which advocates for a nonlocal discretization of the nonlinear part. We will therefore prefer the more general form

$$X_{k+1} = e^{\Delta t A} X_k + (e^{\Delta t A} - I)A^{-1} \mathcal{B}(X_k, X_{k+1}). \quad (4)$$

3.1 Matrix formulation

In view of (4), we define the renormalisation matrix $\Phi(\Delta t) = (e^{\Delta t A} - I)A^{-1}$, and we can replace the exponential $e^{\Delta t A}$ by $I + \Phi(\Delta t)A$ in (4) to obtain

$$X_{k+1} = (I + \Phi(\Delta t)A) X_k + \Phi(\Delta t)\mathcal{B}(X_k, X_{k+1}),$$

or equivalently

$$\Phi^{-1}(\Delta t) (X_{k+1} - X_k) = AX_k + \mathcal{B}(X_k, X_{k+1}), \quad (5)$$

where the renormalization matrix verifies the property

$$\Phi(\Delta t) = \Delta t I + \mathcal{O}(\Delta t^2) \text{ as } \Delta t \rightarrow 0.$$

Let us generalize (1) and hence Rule 2 for systems of ordinary differential equations. The scalar functions ϕ and ψ are then replaced by matrix-valued functions Φ and Ψ .

Rule 2'. *The first order derivatives in a nonstandard scheme for a system of ordinary differential equations should be approximated as*

$$\frac{dX}{dt} \simeq \Phi(\Delta t)^{-1}(X_{k+1} - \Psi(\Delta t)X_k),$$

where

$$\Phi(\Delta t) = \Delta t I + \mathcal{O}(\Delta t^2) \text{ as } \Delta t \rightarrow 0,$$

and

$$\Psi(\Delta t) = I + \mathcal{O}(\Delta t^2) \text{ as } \Delta t \rightarrow 0.$$

Scheme (5) is nonstandard. In particular, the discretization of the first order derivative corresponds to the above generalized rule, with $\Psi \equiv I$. The fact that we are able to write an exact or only a best scheme depends on the nonlinearity.

The major drawback of such a scheme is that we have to evaluate the exponential of matrix $\Delta t A$. This can prove to be an expensive computation [MV03]. This is the reason why we propose in the next section to reformulate scheme (5) in a scalar way.

3.2 Scalar formulation

3.2.1 Construction

To reformulate scheme (5), we consider the Cayley–Hamilton theorem, which implies that the exponential matrix can be rewritten as a finite expansion in powers of A :

$$e^{\Delta t A} = \alpha_0(\Delta t)I + \alpha_1(\Delta t)A + \alpha_2(\Delta t)A^2 + \dots + \alpha_{n-1}(\Delta t)A^{n-1}, \quad (6)$$

where $\alpha_0(\Delta t), \alpha_1(\Delta t), \dots, \alpha_{n-1}(\Delta t) \in \mathbb{R}$. The construction of these coefficients in the general case can be found in [MV03].

Introducing expansion (6) in the exponential integration scheme (4) yields

$$\begin{aligned} X_{k+1} &= \alpha_0(\Delta t)X_k + \alpha_1(\Delta t)[AX_k + \mathcal{B}(X_k, X_{k+1})] \\ &\quad + \sum_{j=2}^{n-1} \alpha_j(\Delta t)A^{j-1}[AX_k + \mathcal{B}(X_k, X_{k+1})] \\ &\quad + (\alpha_0(\Delta t) - 1)A^{-1}\mathcal{B}(X_k, X_{k+1}), \end{aligned}$$

which also reads

$$\begin{aligned} \frac{X_{k+1} - \alpha_0(\Delta t)X_k}{\alpha_1(\Delta t)} &= [I + R_1(\Delta t, A)][AX_k + \mathcal{B}(X_k, X_{k+1})] \\ &\quad + R_0(\Delta t, A)\mathcal{B}(X_k, X_{k+1}), \end{aligned}$$

where we define the two correction factors

$$R_0(\Delta t, A) = \frac{\alpha_0(\Delta t) - 1}{\alpha_1(\Delta t)}A^{-1}, \quad R_1(\Delta t, A) = \sum_{j=2}^{n-1} \frac{\alpha_j(\Delta t)}{\alpha_1(\Delta t)}A^{j-1}. \quad (7)$$

In addition, we also introduce the notion of correction vectors,

$$\begin{aligned} T_0(\Delta t, A, X_k, X_{k+1}) &= R_0(\Delta t, A)\mathcal{B}(X_k, X_{k+1}), \\ T_1(\Delta t, A, X_k, X_{k+1}) &= R_1(\Delta t, A)[AX_k + \mathcal{B}(X_k, X_{k+1})], \end{aligned}$$

to write the NSFD scheme as

$$\begin{aligned} \frac{X_{k+1} - \alpha_0(\Delta t)X_k}{\alpha_1(\Delta t)} &= AX_k + \mathcal{B}(X_k, X_{k+1}) \\ &\quad + T_0(\Delta t, A, X_k, X_{k+1}) + T_1(\Delta t, A, X_k, X_{k+1}). \end{aligned} \quad (8)$$

With regard to Mickens' second rule, we identify $\psi(\Delta t) = \alpha_0(\Delta t)$ and $\phi(\Delta t) = \alpha_1(\Delta t)$.

Remark 1. *If the system dimension is $n = 2$, $R_1(\Delta t, A) = 0$. In the case of a single equation ($n = 1$), the above formulation is not valid since $\alpha_1 \equiv 0$. For linear systems, the correction $T_0(\Delta t, A, X_k, X_{k+1})$ vanishes.*

3.2.2 Order estimate

Proposition 1. *The coefficients $\alpha_j(\Delta t)$ occurring in (6) verify*

$$\alpha_j(\Delta t) = \frac{\Delta t^j}{j!} + \mathcal{O}(\Delta t^n).$$

Proof. Let $S_{n-1}(\Delta tA)$ be the truncated expansion of $\exp(\Delta tA)$ in terms of powers of ΔtA :

$$S_{n-1}(\Delta tA) = \sum_{j=0}^{n-1} \frac{\Delta t^j}{j!} A^j.$$

Then

$$\exp(\Delta t A) - S_{n-1}(\Delta t A) = \sum_{k=n}^{+\infty} \frac{\Delta t^k}{k!} A^k = \Delta t^n A^n \sum_{k=0}^{+\infty} \frac{\Delta t^k}{(n+k)!} A^k,$$

$$\|\exp(\Delta t A) - S_{n-1}(\Delta t A)\| \leq \Delta t^n \|A\|^n \sum_{k=0}^{+\infty} \frac{\Delta t^k}{k!} \|A\|^k = \Delta t^n \|A\|^n \exp(\Delta t \|A\|).$$

For $\Delta t \in]0, \Delta t_0]$, setting $C = \|A\|^n \exp(\Delta t_0 \|A\|)$,

$$\|\exp(\Delta t A) - S_{n-1}(\Delta t A)\| \leq C \Delta t^n.$$

The construction in [MV03] is based on the Cayley–Hamilton theorem. Matrix A^n can be written as a finite expansion in lower powers of A , defining coefficients c_j , $j = 0, \dots, n-1$:

$$A^n = \sum_{j=0}^{n-1} c_j A^j.$$

This allows to define coefficients β_{kj} , $k \geq 0$, $j = 0, \dots, n-1$, such that

$$A^k = \sum_{j=0}^{n-1} \beta_{kj} A^j,$$

and the β_{kj} can be computed iteratively

$$\beta_{kj} = \begin{cases} \delta_{kj} & k < n, \\ c_j & k = n, \\ c_0 \beta_{k-1, n-1} & k > n, j = 0, \\ c_j \beta_{k-1, n-1} + \beta_{k-1, j-1} & k > n, j > 0. \end{cases}$$

Plugging this in the expansion of $\exp(\Delta t A)$ in terms of powers of $\Delta t A$, we obtain coefficients α_j :

$$\alpha_j(\Delta t) = \sum_{k=0}^{n-1} \frac{\Delta t^k}{k!} \beta_{kj} + \frac{\Delta t^n}{n!} \beta_{nj} + \sum_{k=n+1}^{\infty} \frac{\Delta t^k}{k!} \beta_{kj} = \frac{\Delta t^j}{j!} + \frac{\Delta t^n}{n!} c_j + \sum_{k=n+1}^{\infty} \frac{\Delta t^k}{k!} \beta_{kj}.$$

This implies that at the precision $\mathcal{O}(\Delta t^n)$ the expansion (6) is exactly $S_{n-1}(\Delta t A)$. \square

Proposition 2. *The correction factors R_0 and R_1 have the following series expansion*

$$R_0(\Delta t, A) = \frac{\Delta t^{n-1}}{n!} (-1)^{n-1} \det(A) A^{-1} + \mathcal{O}(\Delta t^n),$$

$$R_1(\Delta t, A) = \sum_{j=2}^{n-1} \frac{\Delta t^{j-1}}{j!} A^{j-1} + \mathcal{O}(\Delta t^{n-1}).$$

Proof. We compute

$$\frac{\alpha_0(\Delta t) - 1}{\alpha_1(\Delta t)} = \frac{\frac{\Delta t^n}{n!}c_0 + \mathcal{O}(\Delta t^{n+1})}{\Delta t + \mathcal{O}(\Delta t^n)} = \frac{\Delta t^{n-1}}{n!}c_0 + \mathcal{O}(\Delta t^n).$$

Besides $c_0 = (-1)^{n-1} \det(A)$. For $j \geq 2$,

$$\frac{\alpha_j(\Delta t)}{\alpha_1(\Delta t)} = \frac{\frac{\Delta t^j}{j!} + \mathcal{O}(\Delta t^{n+1})}{\Delta t + \mathcal{O}(\Delta t^n)} = \frac{\Delta t^{j-1}}{j!} + \mathcal{O}(\Delta t^{n-1}).$$

□

3.2.3 Correction of the right-hand side

In Equation (8), Rule 2 in its generalized scalar form (1) is untouched, but now the discretization of the right-hand side is modified.

Rule 3'. *In an NSFD scheme for the nonlinear system (2) satisfying the classical Rule 2 (1), the usual nonlocal discretization of the right-hand side $AX_k + \mathcal{B}(X_k, X_{k+1})$ should be supplemented with correction terms:*

$$AX_k + \mathcal{B}(X_k, X_{k+1}) + R_1(\Delta t, A) [AX_k + \mathcal{B}(X_k, X_{k+1})] + R_0(\Delta t, A) \mathcal{B}(X_k, X_{k+1}),$$

where R_0 and R_1 are given by (7).

4 Numerical tests

We will now study the impact of the corrections in various situations. According to Remark 1 we can find contexts where one or the other correction vanishes.

4.1 Impact of R_0

4.1.1 A quadratic nonlinear oscillator

We first study the impact of R_0 . According to Remark 1, we therefore consider a system of two differential equations ($n = 2$), so that $R_1 \equiv 0$. In [Hu06] the quadratic nonlinear differential equation is presented as a good benchmark for numerical schemes

$$\begin{aligned} x'' + x + x^2 &= 0, \\ x(0) &= x_0 > 0, \\ x'(0) &= 0. \end{aligned} \tag{9}$$

This equation occurs for example in human eardrum oscillation modeling. It has the significant advantage to have a known exact solution for $x_0 < 1/2$, namely

$$x(t) = x_0 + a \operatorname{sn}^2(\omega t, m),$$

where

$$\begin{aligned} a &= \frac{-12x_0(1+x_0)}{\sqrt{3(1-2x_0)(3+2x_0)+3(1+2x_0)}}, \\ \omega &= \frac{1}{2}\sqrt{\frac{1}{2}+x_0} + \frac{1}{6}\sqrt{3(1-2x_0)(3+2x_0)}, \\ m &= \frac{1}{2} + \frac{3(2x_0^2+2x_0-1)}{3+(1+2x_0)\sqrt{3(1-2x_0)(3+2x_0)}}, \end{aligned}$$

and sn is the Jacobi sine function. We write the second order equation (9) in the Hamiltonian form

$$\begin{cases} x' = y, \\ y' = -x - x^2. \end{cases} \quad (10)$$

We thus obtain a system of the form (2), with matrices

$$A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad B(X) = \begin{pmatrix} 0 \\ b(x) \end{pmatrix},$$

where $b(x) = -x^2$ and with initial data $x(0) = x_0$ and $y(0) = 0$.

Besides the comparison with an exact solution, two properties can be used to evaluate the quality of numerical methods. First the exact solutions to (9) are periodic with period

$$P = 4 \int_0^{\pi/2} \frac{d\theta}{\sqrt{1-m^2 \sin^2 \theta}}.$$

Second, the differential equation (9) satisfies a conservation law:

$$E(t) \equiv \frac{1}{2}(x'(t))^2 + \frac{1}{2}x(t)^2 + \frac{1}{3}x(t)^3 = \frac{1}{2}x_0^2 + \frac{1}{3}x_0^3. \quad (11)$$

Several methods exist to obtain efficient schemes (called *best schemes* by Mickens [Mic94]) for a harmonic oscillator. We can cite the Gautschi type method [HL99], the exponential integration method [HO10], the gradient method [Cie11, Cie13, CR10, CR11] and the NSFD method [Mic94, MOR05, MR94]. Here, we are interested in the NSFD method. In [SBF18] the computation of α_0 and α_1 is made explicit for two-dimensional matrices in term of the eigenvalues, namely $\pm i$ for the matrix involved in (10), leading to $\alpha_0(\Delta t) = \cos(\Delta t)$ and $\alpha_1(\Delta t) = \sin(\Delta t)$. This is coherent with the predictions of Proposition 1. Indeed, we have here $c_0 = -1$ and $c_1 = 0$, and this yields $\alpha_0(\Delta t) = 1 - \Delta t^2/2 + \mathcal{O}(\Delta t^3)$ and $\alpha_1(\Delta t) = \Delta t + \mathcal{O}(\Delta t^3)$.

4.1.2 Mickens' scheme for Hamiltonian systems

In [Mic94], Mickens discretized Equation (10) as

$$\begin{cases} \frac{x_{k+1} - \cos(\Delta t)x_k}{\sin(\Delta t)} = y_k, \\ \frac{y_{k+1} - \cos(\Delta t)y_k}{\sin(\Delta t)} = -x_k - (x_{k+1})^2. \end{cases}$$

This scheme has the form (8), with $\mathcal{B}(X_k, X_{k+1}) = B(X_{k+1})$ and no correction term. Eliminating y_k leads to a discretization of (9):

$$\frac{x_{k+1} - 2x_k + x_{k-1}}{\sin^2(\Delta t)} + \frac{2[1 - \cos(\Delta t)]x_k}{\sin^2(\Delta t)} + x_k^2 = 0.$$

The quantity $\frac{2[1 - \cos(\Delta t)]}{\sin^2(\Delta t)}$ tends to 1 as $\Delta t \rightarrow 0$, but we want to have exactly 1 to have an exact computation of the linear term of Equation (9). We can express everything in terms of trigonometric functions of $\Delta t/2$ and write

$$\frac{x_{k+1} - 2x_k + x_{k-1}}{[2\sin(\Delta t/2)]^2} + x_k + \cos^2(\Delta t/2)x_k^2 = 0. \quad (12)$$

One of the consequences of the conservation law (11) is that any periodic solution oscillates with a constant amplitude. The Mickens scheme (12) has this property. It suffices to note that it is invariant for any transformation which swaps x_{k+1} and x_{k-1} .

However, in the case of a harmonic oscillator, it must also be shown that there is a constant first integral. Moreover it was established in [Mic94] that the discretization of the nonlinear term used in (12) does not make it possible to obtain a constant (discrete) integral. It is then advisable [Mic94] to discretize the non-linear term as

$$b(x) \approx -x_k \left(\frac{x_{k+1} + x_{k-1}}{2} \right)$$

and the new scheme is

$$\frac{x_{k+1} - 2x_k + x_{k-1}}{[2\sin(\Delta t/2)]^2} + x_k + \cos^2(\Delta t/2)x_k \frac{x_{k+1} + x_{k-1}}{2} = 0, \quad (13)$$

which conserves all the qualitative properties of the original equation.

This would necessitate to take

$$\frac{y_{k+1} - \cos(\Delta t)y_k}{\sin(\Delta t)} = -x_k - x_{k+1} \frac{x_{k+2} + x_k}{2}$$

as a discretization for the second equation of the Hamiltonian form. The system is then highly implicit and uses time $t + 2\Delta t$ when approximating $B(X(s))$ on the time interval $[t, t + \Delta t]$ in the integral of Equation (3).

4.1.3 Adding a correction term

Let us now see how the NSFD scheme of Section 3.2 reads for Equation (10). Recall that for $n = 2$, $R_1(\Delta t, A) \equiv 0$. We also have already computed $\alpha_0(\Delta t)$ and $\alpha_1(\Delta t)$. The scalar NSFD schemes reads

$$\frac{X_{k+1} - \cos(\Delta t)X_k}{\sin(\Delta t)} = AX_k + \mathcal{B}(X_k, X_{k+1}) - \tan(\Delta t/2)A^{-1}\mathcal{B}(X_k, X_{k+1}). \quad (14)$$

Up to possible choices for $\mathcal{B}(X_k, X_{k+1})$, and noticing that $-A^{-1} = A$, we find the same scheme as Mickens' but with a correction factor $\tan(\Delta t/2)A\mathcal{B}$.

We follow the same steps as in the previous paragraph to obtain a scheme for the initial second order equation.

The scalar NSFDF scheme for equation (10) reads

$$\begin{cases} \frac{x_{k+1} - \cos(\Delta t)x_k}{\sin(\Delta t)} = y_k + \tan(\Delta t/2)b(x_k, x_{k+1}), \\ \frac{y_{k+1} - \cos(\Delta t)y_k}{\sin(\Delta t)} = -x_k + b(x_k, x_{k+1}). \end{cases}$$

Combining these two equations in the same way than for Mickens' scheme first yields

$$\begin{aligned} \frac{x_{k+1} - 2x_k + x_{k-1}}{\sin^2(\Delta t)} &= \frac{2[\cos(\Delta t) - 1]x_k}{\sin^2(\Delta t)} + b(x_{k-1}, x_k) \\ &\quad + \frac{\tan(\Delta t/2)}{\sin(\Delta t)} [b(x_k, x_{k+1}) - \cos(\Delta t)b(x_{k-1}, x_k)]. \end{aligned}$$

With the same transformation that led to (12), we find

$$\frac{x_{k+1} - 2x_k + x_{k-1}}{[2\sin(\Delta t/2)]^2} + x_k = \frac{1}{2}[b(x_{k-1}, x_k) + b(x_k, x_{k+1})]. \quad (15)$$

The $\cos^2(\Delta t/2)$ coefficient of Equations (12) or (13) has disappeared. Besides to obtain $-x_k \frac{x_{k-1} + x_{k+1}}{2}$ in the right-hand side, one has simply to choose $b(x_k, x_{k+1}) = -x_k x_{k+1}$, which is a nonlocal discretization of the nonlinearity (and therefore complies to Rule 3) and is only semi-implicit. It only involves the present and the past but not the future, contrarily to what has been observed for Mickens' scheme (13).

4.1.4 Numerical results

We now compare the previous numerical methods for $x_0 = 0.25$ which lies in the valid interval for initial data, namely $[0, 1/2[$. The exact solution is given by $x_0 + a \operatorname{sn}^2(\omega t, m)$ with $a \simeq -0.55$, $\omega \simeq 0.53$, and $m \simeq 0.33$. Its time evolution over the time interval $[0, 35]$ is displayed in Figure 1.

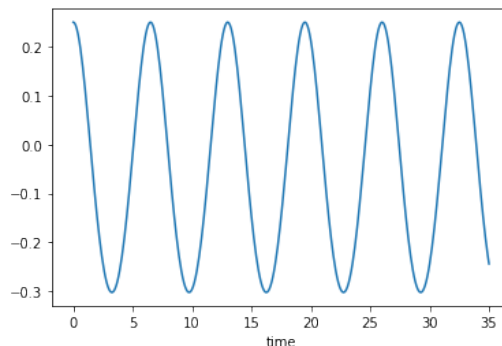


Figure 1: Time evolution of the exact solution of Equation (9) for $x_0 = 0.25$.

As already mentioned, this solution is periodic in time. For the tests we use different values of Δt and compute solutions *via*

- the explicit Euler scheme,
- the implicit Euler scheme,
- Mickens' scheme (12),
- Mickens' scheme (13),
- the scalar scheme with correction R_0 (15) with $b(x_k, x_{k+1}) = -x_k x_{k+1}$.

The advantage of comparing the methods with a quadratic nonlinearity is that we are able to compute explicitly the iterates for all these methods without adding methods to solve nonlinear systems such as predictor–correctors or fixed points. The time evolution of the relative error between the exact solution and the computed solutions are shown in Figure 2 for different values of the time-step.

Of course, the three NSFD schemes, (12), (13), and (15), outperform the Euler schemes, in particular they do not show a deterioration of the error as time evolves.

The correction R_0 does indeed improve Mickens' original schemes, gaining more than two errors of magnitude for small Δt . For large Δt the gain is not so clear, but this is due to the approximation of the nonlinearity, which is the only source of approximation in (15), and which is dominant for $\Delta t = 0.05$.

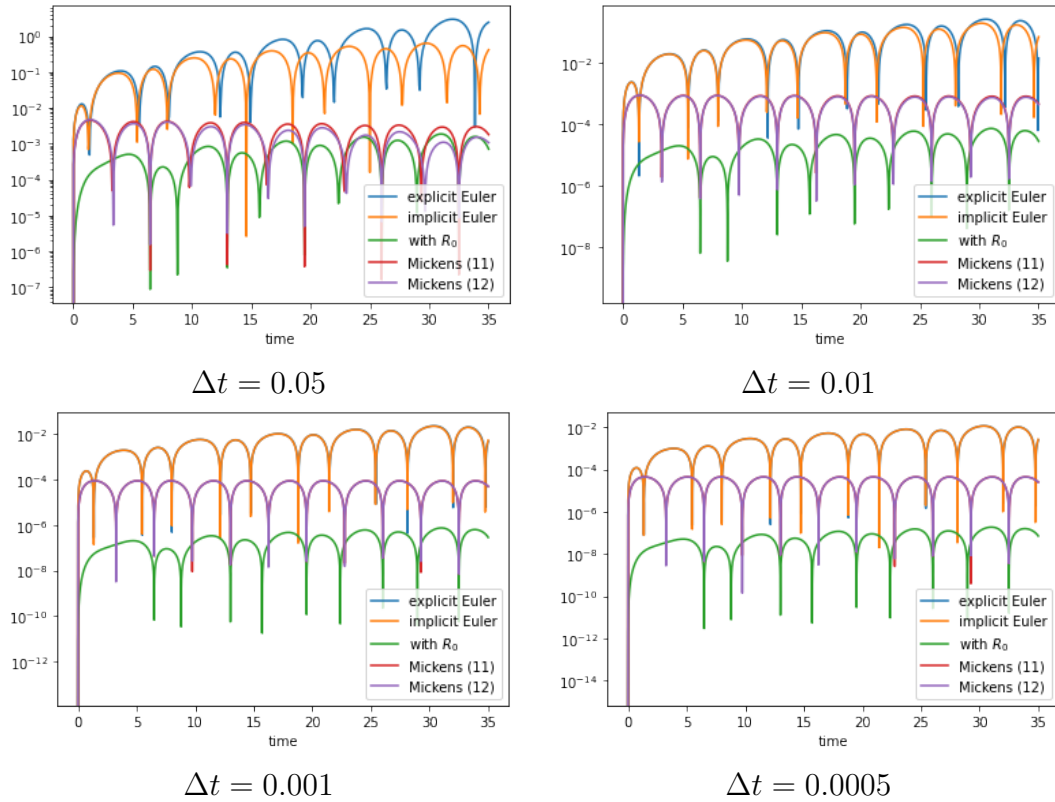


Figure 2: Time evolution of the relative errors for $\Delta t = 0.05, 0.01, 0.001,$ and 0.0005 for Equation (9) and $x_0 = 0.25$.

4.2 Impact of R_1

4.2.1 A forest biomass model

To test the impact of R_1 only, we now consider a linear system with dimension greater than 2, namely $n = 3$. Again, according to Remark 1, $T_0 \equiv 0$ in this linear example. We use a simple example found in [GW98] dealing with the evolution forest biomass. More precisely, we denote $x(t)$ the biomass decayed into humus, $y(t)$ the biomass of dead trees, and $z(t)$ the biomass of living trees. The corresponding evolution equations are

$$\begin{cases} x'(t) = -x(t) + 3y(t), \\ y'(t) = -3y(t) + 5z(t), \\ z'(t) = -5z(t), \end{cases} \quad (16)$$

with an initial data where there are no dead trees and no humus at $t = 0$, namely $x(0) = 0$, $y(0) = 0$, and $z(0) = z_0$. The corresponding matrix A is

$$A = \begin{pmatrix} -1 & 3 & 0 \\ 0 & -3 & 5 \\ 0 & 0 & -5 \end{pmatrix}, \quad (17)$$

yielding the exact solution

$$\begin{cases} x(t) = \frac{15}{8} (e^{-t} - 2e^{-3t} + e^{-5t}) z_0, \\ y(t) = \frac{5}{2} (e^{-3t} - e^{-5t}) z_0, \\ z(t) = e^{-5t} z_0. \end{cases}$$

4.2.2 Derivation of correction terms

For matrix A given by Equation (17), we have $A^3 = -15 - 23A - 9A^2$, i.e. $c_0 = -15$, $c_1 = -23$, and $c_2 = -9$. We therefore predict that $\alpha_0(\Delta t) = 1 - \frac{5}{2}\Delta t^3 + \mathcal{O}(\Delta t^4)$, $\alpha_1(\Delta t) = \Delta t - \frac{23}{6}\Delta t^3 + \mathcal{O}(\Delta t^4)$, and $\alpha_2(\Delta t) = \frac{1}{2}\Delta t^2 - \frac{3}{2}\Delta t^3 + \mathcal{O}(\Delta t^4)$. Writing

$$(\alpha_0(\Delta t)I + \alpha_1(\Delta t)A + \alpha_2(\Delta t)A^2) \begin{pmatrix} 0 \\ 0 \\ z_0 \end{pmatrix} = \begin{pmatrix} x(\Delta t) \\ y(\Delta t) \\ z(\Delta t) \end{pmatrix}$$

yields

$$\begin{aligned} \alpha_0(\Delta t) &= \frac{15}{8}e^{-\Delta t} - \frac{5}{4}e^{-3\Delta t} + \frac{3}{8}e^{-5\Delta t}, \\ \alpha_1(\Delta t) &= e^{-\Delta t} - \frac{3}{2}e^{-3\Delta t} + \frac{1}{2}e^{-5\Delta t}, \\ \alpha_2(\Delta t) &= \frac{1}{8}e^{-\Delta t} - \frac{1}{4}e^{-3\Delta t} + \frac{1}{8}e^{-5\Delta t}. \end{aligned}$$

These values do agree with the predicted expansions at order 3. Since there is no nonlinear part,

$$\begin{aligned} X_{k+1} &= \alpha_0(\Delta t)X_k + \alpha_1(\Delta t)AX_k + \alpha_2(\Delta t)A^2X_k \\ &= \alpha_0(\Delta t)X_k + \alpha_1(\Delta t)AX_k + \alpha_2(\Delta t)A^2X_k. \end{aligned} \quad (18)$$

4.2.3 Numerical results

For the numerical test case, we compare our method (18), which should be exact since no approximation has been done in its derivation, with the Euler explicit and implicit methods. We also compute

$$X_{k+1} = \gamma_0(\Delta t)X_k + \gamma_1(\Delta t)AX_k + \gamma_2(\Delta t)A^2X_k,$$

where the γ_j are the order 3 approximations of α_j , namely $\gamma_0(\Delta t) = 1 - \frac{5}{2}\Delta t^3$, $\gamma_1(\Delta t) = \Delta t - \frac{23}{6}\Delta t^3$, and $\gamma_2(\Delta t) = \frac{1}{2}\Delta t^2 - \frac{3}{2}\Delta t^3$. Finally we derive a NSFD scheme on the above principles but for each equation separately, leading to

$$\begin{cases} \frac{x_{k+1} - x_k}{1 - e^{-\Delta t}} = -x_k + 3y_k, \\ \frac{y_{k+1} - y_k}{(1 - e^{-3\Delta t})/3} = -3y_k + 5z_k, \\ \frac{z_{k+1} - z_k}{(1 - e^{-5\Delta t})/5} = -5z_k. \end{cases} \quad (19)$$

The exact solution is computed over $[0, 10]$, corresponding to ten years of time evolution, and the result is displayed on Figure 3.

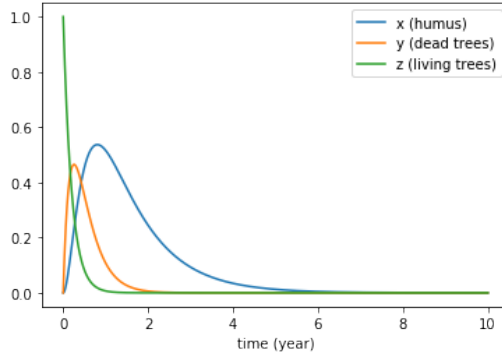


Figure 3: Time evolution of the exact solution of Equation (16) for $z_0 = 1$.

As expected from the equations, the biomass decayed into humus (corresponding to x) has the slowest time evolution and the errors accumulated on x are greater than on the other variables. This is why we will show the errors on this variable. Dealing with a variable that is naturally decaying to zero the relative errors are computed as

$$E_k = \frac{|x_k - x_k^e|}{x_k^e},$$

where x_k^e is the exact value and x_k the computed value. Figure 4 shows the relative errors for the different methods.

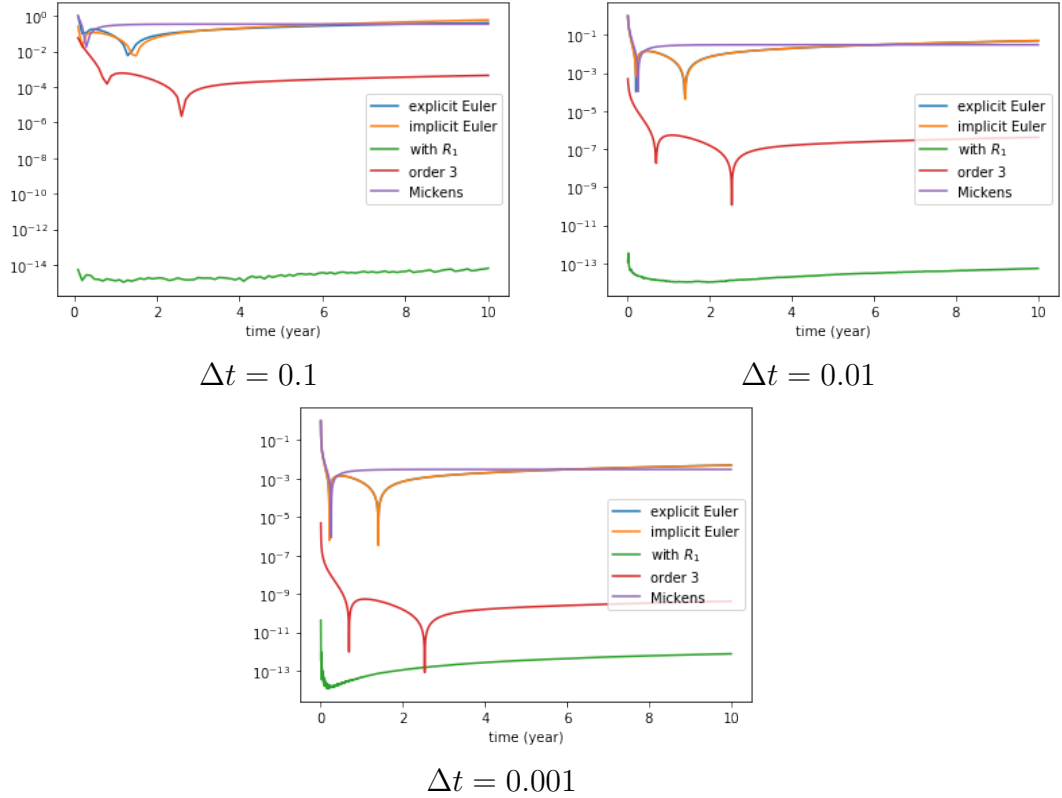


Figure 4: Time evolution of the relative errors for $\Delta t = 0.1, 0.01, 0.001$ for the forest biomass model.

As expected our method is exact. The order 3 method also behaves very well. It has the major advantage to be derived only with the knowledge of the coefficient of the characteristic polynomial of matrix A which is much easier to compute than the α_j . The performance of the traditional NSFD method (19) is comparable to that of the explicit and implicit Euler methods.

4.3 Impact of R_0 and R_1

4.3.1 A forest biomass model with constant force

To test the impact of both R_0 and R_1 , we consider the forest biomass model (16), in which we introduce a constant forcing by planting trees. This corresponds to add a constant z_f in the right-hand side of the last equation, modeling the time evolution of living trees. Hence the system reads

$$\begin{cases} x'(t) = -x(t) + 3y(t), \\ y'(t) = -3y(t) + 5z(t), \\ z'(t) = -5z(t) + z_f, \end{cases} \quad (20)$$

with initial conditions $x(0) = 0$, $y(0) = 0$, and $z(0) = z_0$.

The analytical solution is given by

$$\begin{cases} x(t) = \frac{15}{8} (e^{-t} - 2e^{-3t} + e^{-5t}) z_0 + \frac{1}{8} (8 - 15e^{-t} + 10e^{-3t} - 3e^{-5t}) z_f, \\ y(t) = \frac{5}{2} (e^{-3t} - e^{-5t}) z_0 + \frac{1}{6} (2 - 5e^{-3t} + 3e^{-5t}) z_f, \\ z(t) = e^{-5t} (z_0 - \frac{z_f}{5}) + \frac{z_f}{5}. \end{cases}$$

We display in Figure 5 the time evolution of this analytical solution for $z_0 = 1$ and $z_f = 0.5$. We observe in particular the theoretical long time limits, z_f , $z_f/3$, and $z_f/5$ for x , y , and z respectively.

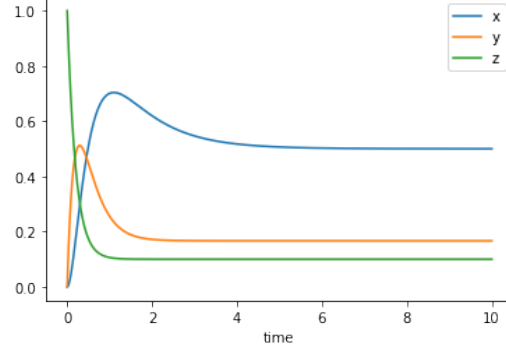


Figure 5: Time evolution of the exact solution of System (20) for $z_0 = 1$ and $z_f = 0.5$.

The NSFD scheme (8) reads

$$\begin{aligned} X_{k+1} = & \alpha_0(\Delta t)X_k + \alpha_1(\Delta t) [AX_k + \mathcal{B}] + \alpha_2(\Delta t)A [AX_k + \mathcal{B}] \\ & + (\alpha_0(\Delta t) - 1)A^{-1}\mathcal{B}, \end{aligned} \quad (21)$$

where the matrix A and coefficients α_j are the same as in (18), but now we have a constant nonlinearity \mathcal{B}

$$\mathcal{B} = \begin{pmatrix} 0 \\ 0 \\ z_f \end{pmatrix}. \quad (22)$$

This test case enables to study the impact of the correction term for the nonlinear part without any approximation on the nonlinearity itself.

We anew compare this method, with the explicit and implicit Euler schemes, the scheme where the coefficients α_j are replaced by the corresponding γ_j , and the traditional NSFD scheme where we replace the last equation in System (19) by

$$\frac{z_{k+1} - z_k}{(1 - e^{-5\Delta t})/5} = -5z_k + z_f.$$

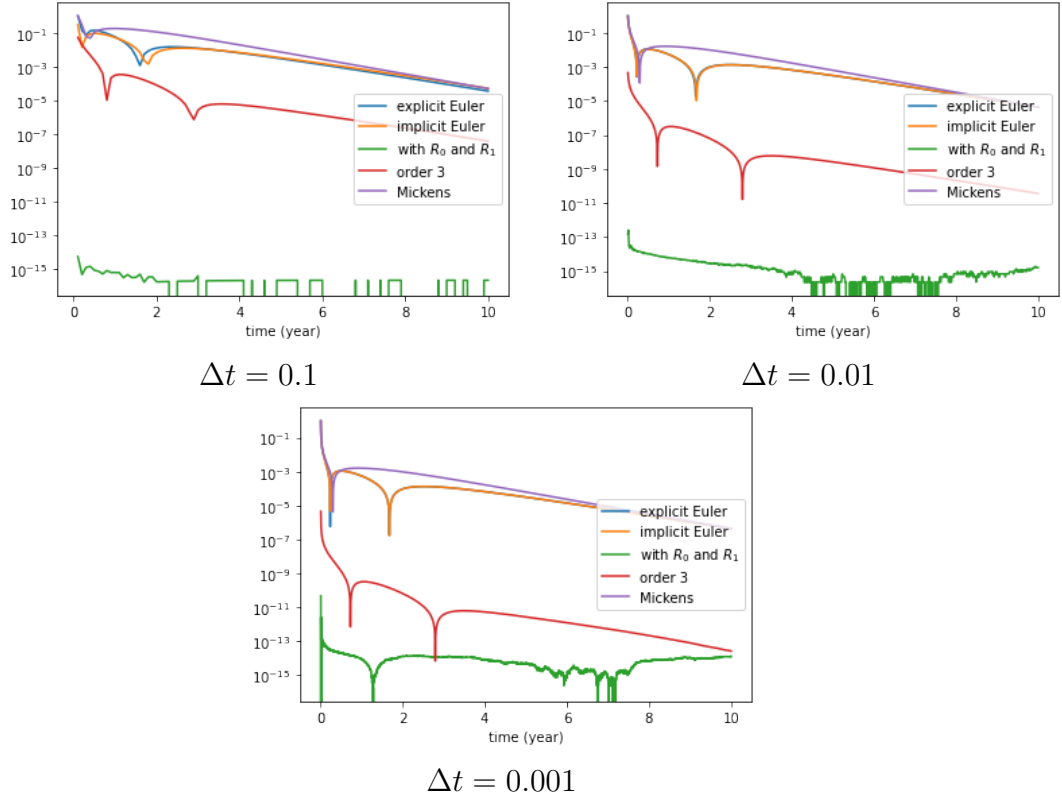


Figure 6: Time evolution of the relative errors for $\Delta t = 0.1, 0.01, 0.001$ for the forest biomass model with constant forcing.

The time evolution of the relative error between the analytical exact solution and the approximated solutions is shown in Figure 6 for different values of the time step. Our method is exact and behaves very well for any time step. Replacing the α_j 's by their third order approximation also yields good results, while the traditional Mickens' NSFD scheme is comparable to the explicit and implicit Euler methods.

4.3.2 A forest biomass model with a seasonal plantation

To continue to explore method errors, we now modify the biomass model (16) to have both corrections R_0 and R_1 and this time a nonlinearity that models seasonal plantations, and which amounts to performing a sinusoidal forcing

$$\begin{cases} x'(t) = -x(t) + 3y(t), \\ y'(t) = -3y(t) + 5z(t), \\ z'(t) = -5z(t) + z_f [1 + \cos(\omega t)], \end{cases} \quad (23)$$

with initial conditions $x(0) = 0$, $y(0) = 0$, and $z(0) = z_0$. Such a time dependent forcing will have to be approximated in the numerical schemes.

The exact analytical solution of the new system is

$$\begin{cases} x(t) = \frac{15}{8} (e^{-t} - 2e^{-3t} + e^{-5t}) z_0 + \frac{1}{8} (8 - 15e^{-t} + 10e^{-3t} - 3e^{-5t}) z_f \\ \quad + 15 \frac{3(5 - 3\omega^2) \cos(\omega t) + \omega(23 - \omega^2) \sin(\omega t)}{(1 + \omega^2)(9 + \omega^2)(25 + \omega^2)} z_f \\ \quad + \frac{15}{8} \left(\frac{-e^{-t}}{1 + \omega^2} + \frac{6e^{-3t}}{9 + \omega^2} + \frac{-5e^{-5t}}{25 + \omega^2} \right) z_f, \\ y(t) = \frac{5}{2} (e^{-3t} - e^{-5t}) z_0 + \frac{1}{6} (2 - 5e^{-3t} + 3e^{-5t}) z_f \\ \quad + 5 \frac{(15 - \omega^2) \cos(\omega t) + 8\omega \sin(\omega t)}{(9 + \omega^2)(25 + \omega^2)} z_f + \frac{5}{2} \left(\frac{-3e^{-3t}}{9 + \omega^2} + \frac{5e^{-5t}}{25 + \omega^2} \right) z_f, \\ z(t) = e^{-5t} z_0 + \frac{1}{5} (1 - e^{-5t}) z_f + \frac{5 \cos(\omega t) + \omega \sin(\omega t)}{25 + \omega^2} z_f + \frac{-5e^{-5t}}{25 + \omega^2} z_f. \end{cases}$$

This solution is displayed in Figure 7. We choose $z_0 = 1$ and $z_f = 0.5$ to have the same mean limits as in the previous simulations. We also choose $\omega = 2\pi$ to have a one year period for the forcing.

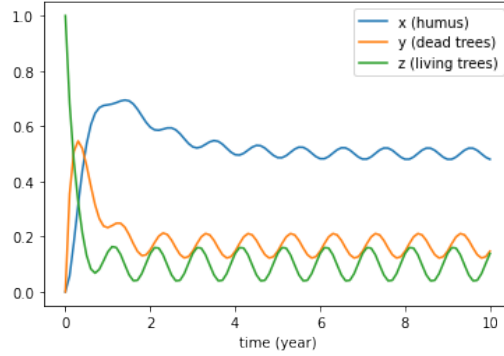


Figure 7: Time evolution of the exact solution of System (23) for $z_0 = 1$, $z_f = 0.5$ and $\omega = 2\pi$.

The numerical schemes we compare are exactly the same as before, except for the treatment of B :

$$B(t) = \begin{pmatrix} 0 \\ 0 \\ z_f [1 + \cos(\omega t)] \end{pmatrix},$$

which is now time-dependent and for which we have to choose an approximation. For the computation of X_{k+1} from X_k , five approximations have been used and compared if relevant, namely

$$\mathcal{B}_{\text{left}} = B(t_k), \quad \mathcal{B}_{\text{right}} = B(t_{k+1}), \quad \mathcal{B}_{\text{middle}} = B((t_k + t_{k+1})/2),$$

$$\mathcal{B}_{\text{half}} = (B(t_k) + B(t_{k+1}))/2, \quad \mathcal{B}_{\text{mean}} = \int_{t_k}^{t_{k+1}} B(t) dt.$$

The explicit and implicit Euler methods clearly use $\mathcal{B}_{\text{left}}$ and $\mathcal{B}_{\text{right}}$ respectively, but the question is open for the other numerical methods. In a first row

of numerical tests we compare the errors when \mathcal{B} is approximated by $\mathcal{B}_{\text{half}}$. We choose this because it is the form which (besides the explicit one) is the easiest to extend when nonlinearities involving X_k are concerned. Figure 8 shows the errors for the five studied schemes. Again our method and its third order approximation outperform the three other schemes.

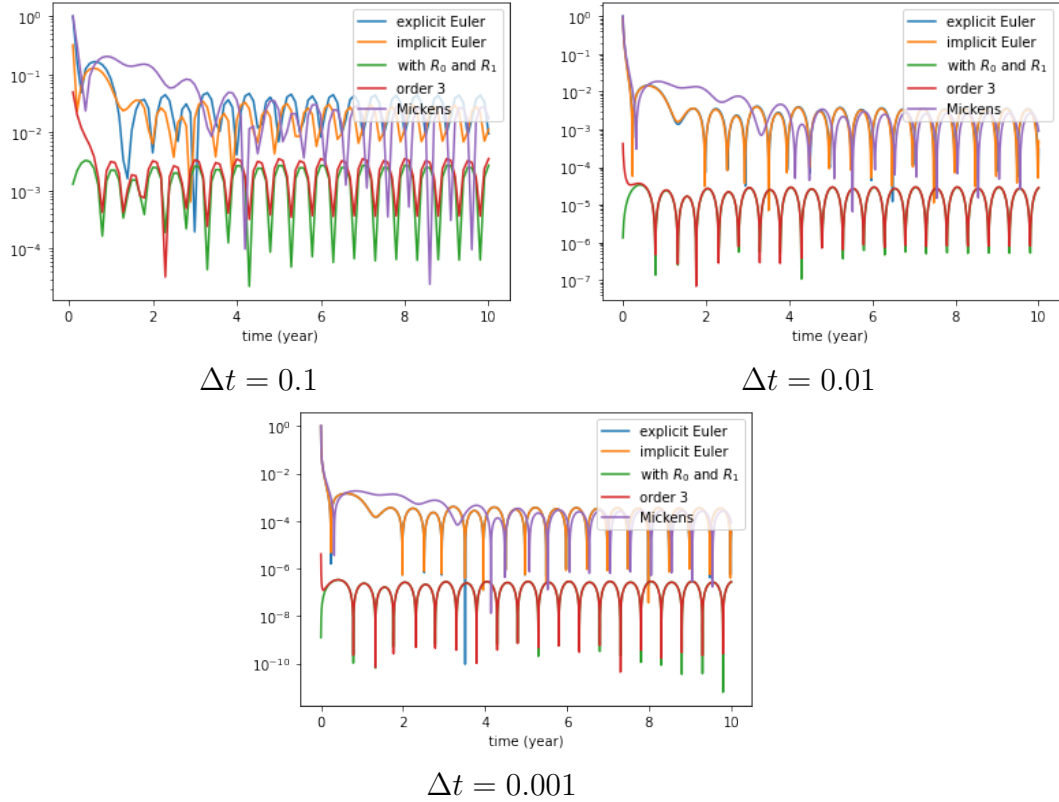


Figure 8: Time evolution of the relative errors for $\Delta t = 0.1, 0.01, 0.001$ for the forest biomass model with time-dependent forcing.

Now we want to discuss the comparison of the two best methods. Since there is an approximation in the estimation of the time-dependent term, and that this approximation is coarser than the approximation in the third order method, both schemes yield very similar results, and the errors are $O(\Delta t^2)$. We can discuss a little further by comparing the use of $\mathcal{B}_{\text{left}}, \mathcal{B}_{\text{middle}}, \mathcal{B}_{\text{half}}$, and $\mathcal{B}_{\text{mean}}$ for $\Delta t = 0.001$. The numerical results are displayed in Figure 9. The computation with $\mathcal{B}_{\text{half}}$ yields the worst results among the other methods but the difference is not significant enough to be worth when dealing with more complex nonlinearities or time dependent forcings.

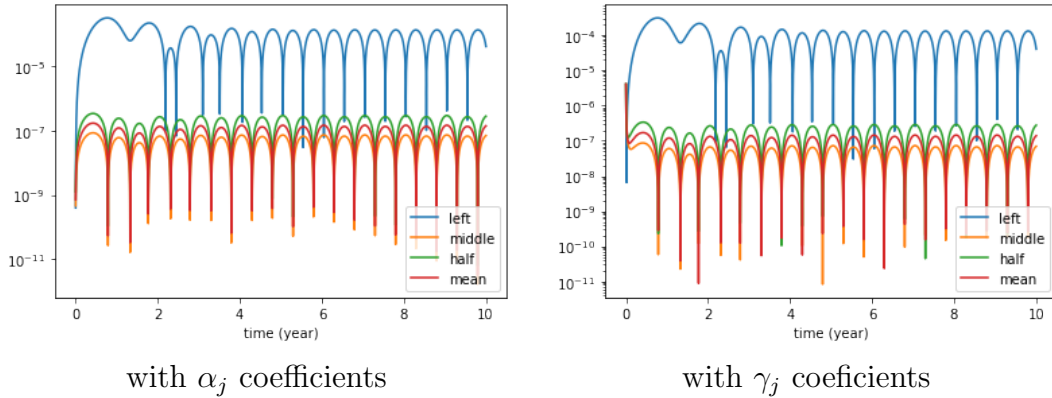


Figure 9: Comparison of the approximations of the time-dependent forcing for our methods and its third order approximation.

5 Discussion

5.1 Extended rules for systems

We have defined two new rules for NSFD schemes for systems of ODEs. These rules stem from a careful derivation when splitting the equation into a linear and a nonlinear part. The only approximations are made on the nonlinear part.

In a first step a matrix formulation is given, leading to a generalization of the second rule (Rule 2'), which addresses the treatment of the first derivative. The usual scalar functions ϕ and ψ , involved in the denominator and the numerator respectively, are then replaced by matrix valued functions. The system is treated as a whole, contrarily to what can usually be done where each equation is taken into account more or less separately. An example of this separate treatment is illustrated by (19).

The matrix formulation is an exponential integrator, and deriving a scalar version of this scheme allows to avoid the possible difficulties in computing the matrix exponentials. This leads to usual scalar coefficients in the discretization of the first order derivative, but they are the same for all the equations, and to correction terms in the right-hand side, which are described by Rule 3'.

In the examples we have separated the effect of the two correction terms on purpose. But of course they are designed also to work together. If the system is linear, or the nonlinearity is a constant forcing term, no approximation is made at any stage of the derivation and the obtained scheme is exact. In the case of a constant forcing term and for at least three coupled equations the two correction terms are nonzero.

5.2 Deriving the scalar coefficients

The derivation of the scalar coefficient is tedious. The examples shown here are quite simple since they deal with very few equations. In our second

example, we computed $\exp(\Delta t A)$ formally and wrote equation (6), which led to solve a three-dimensional linear system in the α_j . Computing $\exp(\Delta t A)$ formally needs to know the eigenvalues and eigenvectors.

Replacing this formal derivation by a numerical determination of the α_j , computing $\exp(\Delta t A)$ numerically and solving the resulting systems numerically can destroy the quality of the method. We have experienced ourselves that even not being careful with the computation of the (scalar) exponentials in the construction of the α_j in Section 4.2 leads to destroy the fine equilibrium that leads to the expansions in Proposition 1 and to a not better scheme than the explicit Euler scheme!

If the formal computation is not possible, we strongly recommend to replace the α_j by their n -th order approximation as done in Section 4.2 with the "order 3" scheme. This approximation has the advantage to only use the knowledge of the coefficients of the characteristic polynomial. This polynomial is easier to compute than the α_j . It is indeed the first step in the computation of the α_j . Taking γ_j simply consists in using the truncated series $S_{n-1}(\Delta t A) = \sum_{j=0}^{n-1} \frac{\Delta t^j}{j!} A^j$ instead of the matrix exponential. For a linear system with $n = 5$, this is equivalent to use the classical order 4 Runge–Kutta method. For other system dimensions, we also have a Runge–Kutta-like method, but with an order that is adapted to n .

5.3 Singular linear part

In the previous discussion, we have used A^{-1} and implicitly have supposed that A was non-singular. If A is singular, the nonlinearity B can be written as $B = AC + K$ where K belongs to the kernel of A . Then

$$\int_0^{\Delta t} e^{(\Delta t-s)A} ds B = (e^{\Delta t A} - I)C = (e^{\Delta t A} - I)A^+ B,$$

where A^+ is the generalized inverse of A . This allows to generalize our approach in the singular case.

6 Conclusion

Having considered the NSFD method as a special class of exponential integrators, we have been able to revisit Mickens's rules to apply to systems of ODEs. When these systems are linear, the method is exact. In the Hamiltonian nonlinear case, it consists in adding to Mickens' schemes a correction term, that has been shown to improve the accuracy.

References

- [Cie11] J.L. Cieśliński, *On the exact discretization of the classical harmonic oscillator equation*. Journal of Difference Equations and Applications, **17**(11), 1673–1694 (2011).

- [Cie13] J.L. Cieśliński, *Locally exact modifications of numerical schemes*. Computers & Mathematics with Applications, **65**(12), 1920–1938 (2013).
- [CR10] J.L. Cieśliński and B. Ratkiewicz, *Improving the accuracy of the discrete gradient method in the one-dimensional case*. Physical Review E, **81**(1), 016704:1–6 (2010).
- [CR11] J.L. Cieśliński and B. Ratkiewicz, *Energy-preserving numerical schemes of high accuracy for one-dimensional Hamiltonian systems*. Journal of Physics A: Mathematical and Theoretical, **44**(15), 155206:1–14 (2011).
- [GW98] Grant B. Gustafson and Calvin H. Wilcox, *Analytical and computational methods of advanced engineering mathematics*, Springer (1998).
- [HL99] M. Hochbruck and C. Lubich, *A Gautschi-type method for oscillatory second-order differential equations*. Numerische Mathematik, **83** 403–426 (1999).
- [HO10] M. Hochbruck and A. Ostermann, *Exponential integrators*. Acta Numerica, **19**, 209–286 (2010).
- [Hu06] H. Hu, *Exact solution of a quadratic nonlinear oscillator*. Journal of Sound and Vibration **295**, 450–457 (2006).
- [Mic94] R. E. Mickens, *Nonstandard finite difference models of differential equations*. World scientific (1994).
- [Mic00] R. E. Mickens, *Nonstandard finite difference schemes*. In R. E. Mickens (ed), Applications of Nonstandard Finite Difference Schemes, World scientific, pp. 1–54 (2000).
- [MOR05] R.E. Mickens, K. Oyediji, and S. Rucker, *Exact finite difference scheme for second-order, linear ODEs having constant coefficients*. Journal of Sound and Vibration, **287**(4–5), 1052–1056 (2005).
- [MR94] R. E. Mickens and I. Ramadhani, *Finite-difference schemes having correct linear stability properties for all step-sizes III*. Computers & Mathematics with Applications, **27**(4), 77–84 (1994).
- [MV03] C. Moler and C. Van Loan, *Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later*. SIAM Review, **45**(1), 3–49 (2003).
- [Pat16] K. C. Patidar, *Nonstandard finite difference methods: recent trends and further developments*. Journal of Difference Equations and Applications, **22**(6), 817–849 (2016).
- [QT18] D. Quang A and H. Manh Tuan. *Exact finite difference schemes for three-dimensional linear systems with constant coefficient*. Vietnam Journal of Mathematics, **46**, 471–492 (2018).

- [SBF18] M. E. Songolo and B. Bidégaray-Fesquet, *Nonstandard finite-difference schemes for the two-level Bloch model*. International Journal of Modeling, Simulation and Scientific Computing, **9**(4), 1850033:1-23 (2018).
- [SBF21] M. E. Songolo and B. Bidégaray-Fesquet, *Strang splitting schemes for N -level Bloch models*. To appear in International Journal of Modeling, Simulation, and Scientific Computing.