

Simulation of harmonic oscillators on the lattice

Michael M. Tung¹ | Javier Ibáñez² | Emilio Defez¹ | Jorge Sastre^{1,3}

¹Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, Camino de Vera s/n, Valencia, 46022, Spain

²Instituto de Instrumentación para Imagen Molecular, Universitat Politècnica de València, Camino de Vera s/n, Valencia, 46022, Spain

³Instituto de Telecomunicaciones y Aplicaciones Multimedia, Universitat Politècnica de València, Camino de Vera s/n, Valencia, 46022, Spain

Correspondence

Javier Ibáñez, Instituto de Instrumentación para Imagen Molecular, Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain.

Email: jjibanez@dsic.upv.es

Communicated by: J. R. Torregrosa

This work deals with the simulation of a two-dimensional ideal lattice having simple tetragonal geometry. The harmonic character of the oscillators give rise to a system of second-order linear differential equations, which can be recast into matrix form. The explicit solutions which govern the dynamics of this system can be expressed in terms of matrix trigonometric functions. For the derivation we employ the Lagrangian formalism to determine the correct solutions, which extremize the underlying action of the system. In the numerical evaluation we develop diverse state-of-the-art algorithms which efficiently tackle equations with matrix sine and cosine functions. For this purpose, we introduce two special series related to trigonometric functions. They provide approximate solutions of the system through a suitable combination. For the final computation an algorithm based on Taylor expansion with forward and backward error analysis for computing those series had to be devised. We also implement several MATLAB programs which simulate and visualize the two-dimensional lattice and check its energy conservation.

KEYWORDS

cosine and sine matrix functions, forward and backward errors, harmonic oscillators, lattice dynamics, lagrangian formalism, matrix differential equations, taylor approximation

MSC CLASSIFICATION

70G99; 70H03; 70-04

1 | INTRODUCTION

In this paper we will consider the classic case of a two-dimensional harmonic lattice with simple tetragonal geometry. The governing equations of motion for this system of harmonic oscillators are derived in matrix form via a variational principle.¹ The underlying multi-particle Lagrangian incorporates the particular geometry of the coupled oscillators and is used to minimize the action comprised of kinetic energy and elastic potential energy. We show that this naturally leads to trigonometric matrix functions describing the dynamics of the system.

After the exposition of the physical model with its underlying Lagrangian dynamics, we continue with the simulation of some especially interesting cases with increasing lattice dimension. For this, we develop new algorithms for the matrix functions arising, i.e, $\cos(\sqrt{X})$ and $(\sqrt{X})^{-1} \sin(\sqrt{X})$ for a real or complex square matrix X are developed, since we will show that the algorithm from Ref.² for $\cos(X)$ cannot be applied to $(\sqrt{X})^{-1} \sin(\sqrt{X})$. The new algorithms are based on Taylor approximations with forward and backward error bounds. The efficiency and performance of these algorithms allow a reliable and stable simulation of lattice models with substantial size. Other, different methods for the evaluation of matrix functions can be found in works by Dehghan & Hajarian.^{3,4} Furthermore, recent works by Kazem & Dehghan^{5,6} demonstrate how matrix functions can be used to approximate solutions in problems of heat conduction.

Throughout this work, we will adopt the usual notation in matrix calculus. Thus, we denote by $\mathbb{R}^{n \times n}$ the set of all real square matrices of order n . Matrices I (or I_n) and $\mathbf{0}$ denote, respectively, the identity matrix (identity matrix of order n) and the null matrix, i.e., the matrix all composed of zeros. In this paper we use the 1-norm of a vector $v = (v_1, v_2, \dots, v_n) \in \mathbb{R}^n$ defined as $\|v\|_1 = \max_{1 \leq i \leq n} \{|v_i|\}$, and the 1-norm of a matrix $A \in \mathbb{R}^{n \times n}$, defined as

$$\|A\|_1 = \sup_{z \neq 0} \frac{\|Az\|_1}{\|z\|_1}.$$

The Kronecker product, denoted $A \otimes B$, of matrices $A = (a_{ij}) \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{r \times s}$ is defined by the following block matrix

$$A \otimes B = \begin{pmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{pmatrix} \in \mathbb{R}^{mr \times ns}.$$

This paper is organized as follows: Section 2 presents the physical problem that will be modeled. Section 3 deals with the fundamental differential equations which determine the model. Sections 4, 5 and 6 explain the MATLAB algorithms developed. Section 7 presents the numerical results, and Section 8 finishes with some conclusions.

2 | LATTICE DYNAMICS

A quadratic lattice of length l is composed of n^2 equidistant nodes ij , where $i, j \in \{1, \dots, n\}$, $n \in \mathbb{N}$. Their spacing is therefore $a = l/(n + 1)$. Each element ij has components x_{ij} and y_{ij} , respectively, which will depend on time $t \geq 0$. If one of the positional indices is either 0 or $n + 1$, the element would lie on the lattice wall and is thus neglected, *viz.* Figure 1. Observe that the geometry of the lattice imposes the symmetry condition that all physical formulas are invariant under the simultaneous change of $x \leftrightarrow y$, $i \leftrightarrow j$, and transposition (interchange of the position of the two indices). We will suppose that the different elements are two-dimensional harmonic oscillators with the same spring constant or stiffness, $\sigma > 0$, while neglecting friction. The Figure 1 illustrates how the element at position ij interacts in all directions with its neighboring elements. We choose as equilibrium length the natural spacing $a > 0$ and suppose that the border elements are also attached with this distance to a fixed suspension. Internal elements have four neighbors at the distance a when in equilibrium. For simplicity, we also assume that the elements have the same mass $m > 0$.

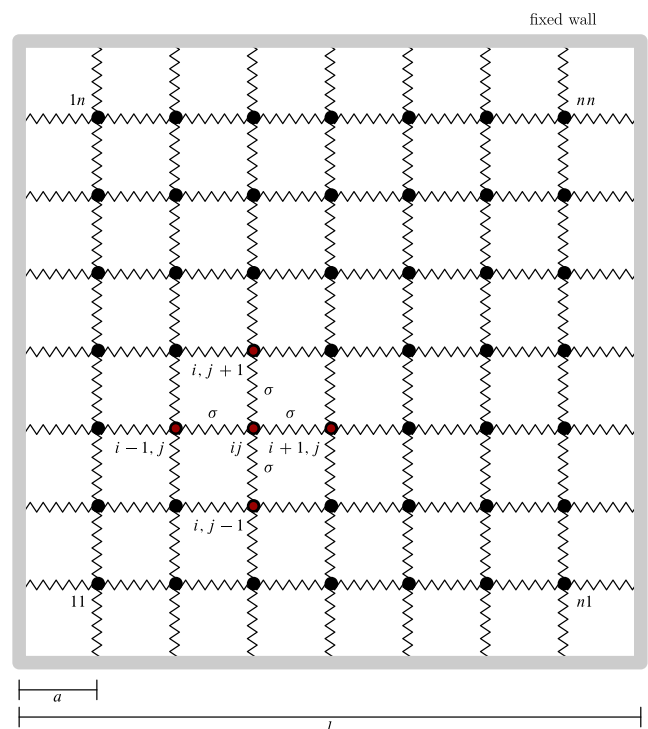


FIGURE 1 Homogeneous $n \times n$ square lattice of simple tetragonal geometry with height and width $l > 0$. All masses $m > 0$ are equal and at equilibrium are situated periodically at the n^2 oscillating nodes with equidistant spacing $a = l/(n + 1)$. The internal elastic interaction is determined by the spring constant $\sigma > 0$. The border masses are attached to a fixed wall. The model neglects friction [Colour figure can be viewed at wileyonlinelibrary.com]

2.1 | Lagrangian formalism

If ξ and η denote, respectively, the displacements in the directions of the x - and y -axis relative to the equilibrium position, then the kinetic energy T_{ij} of each element ij is given by

$$T_{ij} = \frac{1}{2}m \left(\dot{\xi}_{ij}^2 + \dot{\eta}_{ij}^2 \right).$$

Hence, the total kinetic energy of the system is given by

$$T = \sum_{i,j=1}^n T_{ij} = \frac{1}{2} \sum_{i,j=0}^n m \left(\dot{\xi}_{ij}^2 + \dot{\eta}_{ij}^2 \right), \quad (1)$$

where in the last step we included for latter convenience the additional nodes $i, j = 0$ with vanishing velocities $\dot{\xi}_{i0} = \dot{\xi}_{0j} = \dot{\eta}_{i0} = \dot{\eta}_{0j} = 0$.

The potential energy V_{ij} for each element ij is determined by Hooke's Law considering the elastic interaction with all its present neighbors:

$$V_{ij} = \frac{1}{2}\sigma \left[\left(\xi_{i+1,j} - \xi_{i,j} \right)^2 + \left(\eta_{i,j+1} - \eta_{i,j} \right)^2 \right],$$

where $\sigma > 0$ is the same spring constant for all elements for the elastically-homogeneous lattice. Because the walls are fixed, we are required to take $\xi_{0j} = \xi_{n+1,j} = \eta_{i,0} = \eta_{i,n+1} = 0$. Thus, the total potential energy can be expressed as

$$V = \frac{1}{2}\sigma \sum_{i,j=0}^n \left[\left(\xi_{i+1,j} - \xi_{i,j} \right)^2 + \left(\eta_{i,j+1} - \eta_{i,j} \right)^2 \right]. \quad (2)$$

According to the standard procedure of classical mechanics, we are now in the position to construct the Lagrangian function $L = L(\xi_{ij}, \eta_{ij}, \dot{\xi}_{ij}, \dot{\eta}_{ij})$ as follows:

$$L = T - V = \frac{1}{2} \sum_{i,j=0}^n \left[m \left(\dot{\xi}_{ij}^2 + \dot{\eta}_{ij}^2 \right) - \sigma \left(\xi_{i+1,j} - \xi_{i,j} \right)^2 - \sigma \left(\eta_{i,j+1} - \eta_{i,j} \right)^2 \right].$$

Hamilton's Principle readily yields the corresponding equations of motion via a variational principle for the action integral:

$$\delta \int dt L = 0. \quad (3)$$

Equivalently, the solutions of Equation (3) are solutions of the Euler-Lagrange equations:

$$\frac{\delta L}{\delta \xi_{kl}} = \left(\frac{d}{dt} \frac{\partial}{\partial \dot{\xi}_{kl}} - \frac{\partial}{\partial \xi_{kl}} \right) L = 0, \quad (4)$$

$$\frac{\delta L}{\delta \eta_{kl}} = \left(\frac{d}{dt} \frac{\partial}{\partial \dot{\eta}_{kl}} - \frac{\partial}{\partial \eta_{kl}} \right) L = 0, \quad (5)$$

which are decoupled in the x - and y -direction, as expected because of the underlying symmetry property.

Evaluating Equation (4), a detailed computation gives

$$\begin{aligned}
 \frac{1}{2} \left(\frac{d}{dt} \frac{\partial}{\partial \dot{\xi}_{kl}} - \frac{\partial}{\partial \xi_{kl}} \right) \sum_{i,j=0}^n \left[m \dot{\xi}_{ij}^2 - \sigma (\xi_{i+1,j} - \xi_{i,j})^2 \right] &= 0, \\
 m \dot{\xi}_{kl} + \frac{1}{2} \sigma \frac{\partial}{\partial \xi_{kl}} \sum_{i,j=0}^n (\xi_{i+1,j} - \xi_{i,j})^2 &= 0, \\
 m \dot{\xi}_{kl} + \sigma \sum_{i,j=0}^n (\xi_{i+1,j} - \xi_{i,j}) \frac{\partial}{\partial \xi_{kl}} (\xi_{i+1,j} - \xi_{i,j}) &= 0, \\
 m \dot{\xi}_{kl} + \sigma \sum_{i,j=0}^n (\xi_{i+1,j} - \xi_{i,j}) [\delta_{i+1,k} \delta_{j,l} - \delta_{i,k} \delta_{j,l}] &= 0, \\
 m \dot{\xi}_{kl} + \sigma \sum_{i,j=0}^n (\xi_{i+1,j} - \xi_{i,j}) (\delta_{i+1,k} \delta_{j,l} - \delta_{i,k} \delta_{j,l}) &= 0, \\
 m \dot{\xi}_{kl} + \sigma \sum_{i,j=0}^n (\xi_{i+1,j} - \xi_{i,j}) (\delta_{i+1,k} - \delta_{i,k}) \delta_{j,l} &= 0, \\
 m \dot{\xi}_{kl} + \sigma \sum_{i=0}^n (\xi_{i+1,l} - \xi_{i,l}) (\delta_{i+1,k} - \delta_{i,k}) &= 0, \\
 m \dot{\xi}_{kl} + \sigma (\xi_{k,l} - \xi_{k-1,l} - \xi_{k+1,l} + \xi_{k,l}) &= 0, \\
 m \dot{\xi}_{kl} + \sigma (2\xi_{k,l} - \xi_{k-1,l} - \xi_{k+1,l}) &= 0.
 \end{aligned} \tag{6}$$

where $\delta_{i,j}$ denotes the conventional Kronecker symbol.

Similarly to the previous computation of (4), we obtain for (5) the result:

$$m \dot{\eta}_{kl} + \sigma (2\eta_{k,l} - \eta_{k,l-1} - \eta_{k,l+1}) = 0. \tag{7}$$

By comparing (6) and (7), observe that the interchange $\xi \leftrightarrow \eta$ implies simultaneously the transposition of the matrix components so that $(ij) \leftrightarrow (ji)$. This symmetry operation relates to the reflection of the lattice about the diagonal in the first coordinate quadrant.

If we identify $X = (\xi_{kl}), Y = (\eta_{kl}) \in \mathbb{R}^{n \times n}$, then Equations (6) and (7) can be expressed succinctly as

$$\begin{aligned}
 m \ddot{X} + \sigma X A_0 &= 0, \\
 m \ddot{Y} + \sigma A_0 Y &= 0,
 \end{aligned}$$

where $A_0 \in \mathbb{R}^{n \times n}$ is the invertible tridiagonal matrix

$$A_0 = \begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \mathbf{0} \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ \mathbf{0} & & & & -1 & 2 & -1 \\ & & & & & -1 & 2 \end{pmatrix}.$$

If the values of matrices X and Y and their derivatives are known at time $t = 0$, the dynamical evolution of the physical system is fully determined by the following initial value problems:

$$\begin{aligned}
 m \ddot{X}(t) + \sigma X(t) A_0 &= 0, \quad t \in [0, T] \\
 X(0) &= X_0, \quad \dot{X}(0) = \dot{X}_0,
 \end{aligned} \tag{8}$$

$$\begin{aligned} m\ddot{Y}(t) + \sigma A_0 Y(t) &= 0, \quad t \in [0, T] \\ Y(0) &= Y_0, \quad \dot{Y}(0) = \dot{Y}_0. \end{aligned} \quad (9)$$

2.2 | Energy conservation

If we structure the positions x_{ij} and y_{ij} of the nodes in sequential vector order

$$x = \begin{pmatrix} \bar{x}_{11} + \xi_{11} \\ \vdots \\ \bar{x}_{1n} + \xi_{1n} \\ \vdots \\ \bar{x}_{n1} + \xi_{n1} \\ \vdots \\ \bar{x}_{nn} + \xi_{nn} \end{pmatrix} \in \mathbb{R}^{n^2}, \quad y = \begin{pmatrix} \bar{y}_{11} + \eta_{11} \\ \vdots \\ \bar{y}_{1n} + \eta_{1n} \\ \vdots \\ \bar{y}_{n1} + \eta_{n1} \\ \vdots \\ \bar{y}_{nn} + \eta_{nn} \end{pmatrix} \in \mathbb{R}^{n^2},$$

where \bar{x}_{ij} and \bar{y}_{ij} are, respectively, the equilibrium position of the nodes at ij for $i, j \in \{1, 2, \dots, n\}$, then the total energy of the above oscillating system can be expressed as

$$E = \frac{1}{2} m \dot{x}^T \dot{x} + \frac{1}{2} \sigma x^T A_0 \otimes I_n x. \quad (10)$$

We now show that for a certain type of Lagrangian for the lattice the total energy E is conserved:

Theorem 1. Suppose $x(t) \in \mathbb{R}^n$ with derivatives $\dot{x}(t)$ for an open interval $t \in I$. Also let $A, B \in \mathbb{R}^{n \times n}$ be constant square matrices. Then construct the time-independent Lagrangian

$$L(x, \dot{x}) = \dot{x}^T A \dot{x} - x^T B x.$$

The corresponding Euler-Lagrange equations are given by the functional derivative via a variational principle:

$$\delta \int dt L = 0 \rightsquigarrow \frac{\delta L}{\delta x} = 0 \rightsquigarrow A \ddot{x} + Bx = 0.$$

Therefore, since $\partial L / \partial t = 0$, the differential equations

$$A \ddot{x} + Bx = 0$$

will conserve the total energy according to Noether's theorem.⁷ The typical Legendre transformation yields for the total energy in this case

$$E(x, \dot{x}) = \frac{1}{2} \dot{x}^T A \dot{x} + \frac{1}{2} x^T B x.$$

3 | SOLUTION OF THE INITIAL VALUE PROBLEMS

First, we solve the differential matrix equation Equation (8). Dividing the interval $[0, T]$ into r subintervals of lengths equal to $h = T/r$, we will compute the solutions $X_i = X(t_i)$, $i = 1, \dots, r$, by taking into account that $X(0)$ and $\dot{X}(0)$ are known values. For each subinterval we consider the solutions $X(t_i)$ and $\dot{X}(t_i)$ from $X(t_{i-1})$ and $\dot{X}(t_{i-1})$ by solving the differential equation at instant t_i :

$$\begin{aligned} \ddot{X}(t) + X(t)A &= 0, \quad t \in [t_{i-1}, t_i] \\ X(t_{i-1}) &= X_{i-1}, \quad \dot{X}(t_{i-1}) = \dot{X}_{i-1}, \end{aligned} \quad (11)$$

where $A = \frac{\sigma}{m}A_0$. Knowing that X_{i-1} and \dot{X}_{i-1} in general do not commute with $\sin(\sqrt{A}h)$ and $\cos(\sqrt{A}h)$, but \sqrt{A} and $(\sqrt{A})^{-1}$ commute⁸ with $\sin(\sqrt{A}h)$ and $\cos(\sqrt{A}h)$, the solution of (11) and its derivative are, respectively,

$$X_i = X_{i-1} \cos(\sqrt{A}h) + \dot{X}_{i-1} (\sqrt{A})^{-1} \sin(\sqrt{A}h) \quad (12)$$

$$\dot{X}_i = -X_{i-1} \sqrt{A} \sin(\sqrt{A}h) + \dot{X}_{i-1} \cos(\sqrt{A}h), \quad (13)$$

where \sqrt{A} denotes any square root of a non-singular square matrix A .

Indeed, if we consider

$$X(t) = X_{i-1} \cos(\sqrt{A}(t - t_{i-1})) + \dot{X}_{i-1} (\sqrt{A})^{-1} \sin(\sqrt{A}(t - t_{i-1})), \quad (14)$$

then

$$\begin{aligned} \dot{X}(t) &= -X_{i-1} \sqrt{A} \sin(\sqrt{A}(t - t_{i-1})) + \dot{X}_{i-1} \cos(\sqrt{A}(t - t_{i-1})), \\ \ddot{X}(t) &= -X_{i-1} A \cos(\sqrt{A}(t - t_{i-1})) - \dot{X}_{i-1} \sqrt{A} \sin(\sqrt{A}(t - t_{i-1})). \end{aligned} \quad (15)$$

This implies that

$$\begin{aligned} \ddot{X}(t) + X(t)A &= -X_{i-1} A \cos(\sqrt{A}(t - t_{i-1})) - \dot{X}_{i-1} \sqrt{A} \sin(\sqrt{A}(t - t_{i-1})) \\ &\quad + X_{i-1} A \cos(\sqrt{A}(t - t_{i-1})) + \dot{X}_{i-1} \sqrt{A} \sin(\sqrt{A}(t - t_{i-1})) \\ &= 0, \end{aligned}$$

where the following initial conditions are satisfied:

$$\begin{aligned} X(t_{i-1}) &= X_{i-1} \cos(0) + \dot{X}_{i-1} (\sqrt{A})^{-1} \sin(0) = X_{i-1}, \\ \dot{X}(t_{i-1}) &= -X_{i-1} \sqrt{A} \sin(0) + \dot{X}_{i-1} \cos(0) = \dot{X}_{i-1}. \end{aligned}$$

Finally, after substituting $h = t_i - t_{i-1}$ in (14) and (15), we obtain expressions Equations (12) and (13).

On the other hand, since

$$\begin{aligned} f(x) \equiv \cos(\sqrt{x}) &= \sum_{k=0}^{\infty} \frac{(-1)^k (\sqrt{x})^{2k}}{(2k)!} = \sum_{k=0}^{\infty} \frac{(-1)^k x^k}{(2k)!} \equiv T_c(x), \\ g(x) \equiv (\sqrt{x})^{-1} \sin(\sqrt{x}) &= (\sqrt{x})^{-1} \sum_{k=0}^{\infty} \frac{(-1)^k (\sqrt{x})^{2k+1}}{(2k+1)!} = \sum_{k=0}^{\infty} \frac{(-1)^k x^k}{(2k+1)!} \equiv T_s(x), \end{aligned} \quad (16)$$

and the series $T_s(x)$ and $T_c(x)$ both have a radius of convergence $R = \infty$, then $f(A)$ and $g(A)$, $A \in \mathbb{R}^{r \times r}$, are well defined.⁹

Th. 9.1.2

Therefore one arrives at

$$\begin{aligned} \cos(\sqrt{A}h) &= T_c(Ah^2), \\ (\sqrt{A})^{-1} \sin(\sqrt{A}h) &= hT_s(Ah^2), \\ \sqrt{A} \sin(\sqrt{A}h) &= hAT_s(Ah^2), \end{aligned}$$

and expressions (12) and (13) can be written as

$$X_i = X_{i-1} T_c(Ah^2) + h \dot{X}_{i-1} T_s(Ah^2), \quad (17)$$

$$\dot{X}_i = -hX_{i-1}AT_s(Ah^2) + \dot{X}_{i-1}T_c(Ah^2), \quad (18)$$

and it is easy to prove that the solutions at t_i can be calculated as

$$X_i = X_{i-1}T_c(At_i^2) + t_i\dot{X}_{i-1}T_s(At_i^2), \quad (19)$$

$$\dot{X}_i = -t_iX_{i-1}AT_s(At_i^2) + \dot{X}_{i-1}T_c(At_i^2), \quad (20)$$

Similarly, it can be shown that for the differential matrix Equation (9) the solution and its derivative may be written as

$$\begin{aligned} \dot{Y}(t) + AY(t) &= 0, \quad t \in [0, T], \\ Y(0) &= Y_0, \quad \dot{Y}(0) = \dot{Y}_0. \end{aligned} \quad (21)$$

At a particular instant $t_i \in [0, T]$ we have

$$Y_i = T_c(Ah^2)Y_{i-1} + hT_s(Ah^2)\dot{Y}_{i-1}, \quad (22)$$

$$\dot{Y}_i = -hAT_s(Ah^2)Y_{i-1} + T_c(Ah^2)\dot{Y}_{i-1}, \quad (23)$$

and

$$Y_i = T_c(At_i^2)Y_0 + t_iT_s(At_i^2)\dot{Y}_0, \quad (24)$$

$$\dot{Y}_i = -t_iAT_s(At_i^2)Y_0 + T_c(At_i^2)\dot{Y}_0. \quad (25)$$

4 | ALGORITHM FOR COMPUTING $T_C(X)$ AND $T_S(X)$

An efficient method for computing matrix functions is based on obtaining its matrix Taylor approximation and on using the Paterson-Stockmeyer method¹⁰ for the evaluation of the Taylor polynomial considered. This method consists of expressing a polynomial $P_m(X) = \sum_{i=0}^m p_i X^i$, $X \in \mathbb{R}^{n \times n}$ as

$$\begin{aligned} P_m(X) &= ((\dots (X^q p_m + X^{q-1} p_{m-1} + \dots + X p_{m-q+1} + I p_{m-q}) \\ &\quad \times X^q + X^{q-1} p_{m-q-1} + X^{q-2} p_{m-q-2} + \dots + X p_{m-2q+1} + I p_{m-2q}) \\ &\quad \times X^q + X^{q-1} p_{m-2q-1} + X^{q-2} p_{m-2q-2} + \dots + X p_{m-3q+1} + I p_{m-3q}) \\ &\quad \dots \\ &\quad \times X^q + X^{q-1} p_{q-1} + X^{q-2} p_{q-2} + \dots + X p_1 + I p_0. \end{aligned} \quad (26)$$

where the grouping factor q is a positive integer,¹¹ p. 6454,^{9,10} p. 574.¹² p. 72–74 The optimal orders of Taylor approximation are,¹³ p. 1837

$$\mathbb{M} = \{1, 2, 4, 6, 9, 12, 16, 20, 25, 30, \dots\},$$

where we denote the elements \mathbb{M} as m_0, m_1, m_2, \dots , respectively. From work by Higham,¹² p. 74 the optimal values of q for $m_k \in \mathbb{M}$, $k = 0, 1, 2, \dots$, are $\lceil \sqrt{m_k} \rceil$ and $\lfloor \sqrt{m_k} \rfloor$, and both values divide m_k .

Let

$$P_m(X) = \sum_{k=0}^m \frac{(-1)^k X^k}{(2k)!}, \quad (27)$$

$$Q_m(X) = \sum_{k=0}^m \frac{(-1)^k X^k}{(2k+1)!}. \quad (28)$$

be, respectively, the Taylor approximations of order m of $T_c(X)$ and $T_s(X)$. When one assumes,¹² p. 290 $\|X\| \lesssim 64551$, approximations (27) and (28) are accurate. For large $\|X\|$, it is advisable to reduce that norm by using an analogue scaling and doubling method used in the computation of the matrix cosine function.¹⁴ In our case, we select a positive integer s such that $P_m(4^{-s}X)$ and $Q_m(4^{-s}X)$ be good approximations of $T_c(4^{-s}X)$ and $T_s(4^{-s}X)$, respectively. The values m and s

Algorithm 1 Given a matrix $X \in \mathbb{R}^{n \times n}$, this algorithm calculates approximately $T_c(X)$ and $T_s(X)$.

-
- | | |
|--|---|
| 1: Select adequate values of m and s | ▷ Phase I |
| 2: $X = 4^{-s} X$ | |
| 3: Compute $C = P_m(X)$ and $S = Q_m(X)$ (see Expression (26)) | ▷ Phase II |
| 4: for $i = 1 : s$ do | ▷ Phase III: Recovering T_c and T_s |
| 5: $S = SC$ | |
| 6: $C = 2C^2 - I$ | |
| 7: end for | |
-

must be carefully chosen for computing $T_c(X)$ and $T_s(X)$ accurately and efficiently. Since it is well-known,⁹ p. 526 that $\sin(2Y) = 2 \sin(Y) \cos(Y)$ and $\cos(2Y) = 2\cos^2(Y) - I$, $Y \in \mathbb{R}^{n \times n}$, then

$$\begin{aligned} T_s(4X) &= \left(\sqrt{4X}\right)^{-1} \sin\left(\sqrt{4X}\right) = \frac{1}{2} \left(\sqrt{X}\right)^{-1} \sin\left(2\sqrt{X}\right) \\ &= \left(\sqrt{X}\right)^{-1} \sin\left(\sqrt{X}\right) \cos\left(\sqrt{X}\right) = T_s(X)T_c(X), \end{aligned}$$

and

$$T_c(4X) = \cos\left(\sqrt{4X}\right) = \cos\left(2\sqrt{X}\right) = 2\cos^2\sqrt{X} - I = 2T_c^2(X) - I.$$

Hence, $T_s(X)$ and $T_c(X)$ can be recovered by using the following recurrences:

$$\begin{aligned} S_{i+1} &= S_i C_i, \quad i = 0, 1, \dots, s-1, \\ C_{i+1} &= 2C_i^2 - I, \quad i = 0, 1, \dots, s-1. \end{aligned}$$

Algorithm 1 computes $T_c(X)$ and $T_s(X)$ simultaneously by the Taylor and the Paterson-Stockmeyer methods, using the above scaling and doubling method.

5 | COMPUTATION OF THE SCALING FACTOR AND THE ORDER OF TAYLOR APPROXIMATION.

5.1 | Forward errors

The following theorem¹⁵ is used in this subsection:

Theorem 2. Let $h_l(x) = \sum_{i \geq l} p_i x^i$ be a power series with radius of convergence w , $\tilde{h}_l(x) = \sum_{i \geq l} |p_i| x^i$, $X \in \mathbb{C}^{n \times n}$ with $\rho(X) < w$, $l \in \mathbb{N}$ and $t \in \mathbb{N}$ with $1 \leq t \leq l$. If t_0 is the multiple of t such that $l \leq t_0 \leq l + t - 1$ and

$$\beta_t = \max \left\{ d_j^{1/j} : j = t, l, l+1, \dots, t_0-1, t_0+1, t_0+2, \dots, l+t-1 \right\},$$

where d_j is an upper bound for $\|X^j\|$, $d_j \geq \|X^j\|$, then

$$\|h_l(X)\| \leq \tilde{h}_l(\beta_t).$$

If we apply Theorem 2 to the matrix A for $t = l$, then $\|h_l(A)\| \leq \tilde{h}_l(\beta_l)$, where

$$\beta_l = \max \left\{ d_j^{1/j} : j = l, l+1, \dots, 2l-1 \right\}. \quad (29)$$

We use the same approximation of $\beta_{\min} = \min\{\beta_t^{(l)}, 1 \leq t \leq m+1\}$ as the following approximation known from the literature,¹⁶ Sec. 4.1:

$$\beta_{\min} \approx \max \left\{ d_{l+1}^{1/(l+1)}, d_{l+2}^{1/(l+2)} \right\}, \quad (30)$$

corresponding to the two first terms of Equation (29).

	$\Theta_m^{f,c}$	$\Theta_m^{f,s}$	$\Theta_m^{b,c}$	$\Theta_m^{b,s}$
$m = 2$	$4.307691257 \times 10^{-5}$	$8.240339399 \times 10^{-5}$	$3.419034688 \times 10^{-5}$	$4.534846661 \times 10^{-5}$
$m = 4$	$1.319680930 \times 10^{-2}$	$2.133123720 \times 10^{-2}$	$1.149925454 \times 10^{-2}$	$1.491288550 \times 10^{-2}$
$m = 6$	$1.895232414 \times 10^{-1}$	$2.810808759 \times 10^{-1}$	$1.733508225 \times 10^{-1}$	$2.183873949 \times 10^{-1}$
$m = 9$	1.5605489459×10^0	2.2782864730×10^0	1.5886273831×10^0	1.945487807×10^0
$m = 12$	2.7001120248×10^0	7.3622618527×10^0	5.6861650847×10^0	6.975428251×10^1
$m = 16$	3.3335993055×10^0	1.0102246523×10^1	1.0447593515×10^1	1.882060328×10^2

TABLE 1 Values of $\Theta_m^{f,c}$, $\Theta_m^{f,s}$, $\Theta_m^{b,c}$ and $\Theta_m^{b,s}$ for orders $m = 2, 4, 6, 9, 12, 16$.

The development of this subsection is based on a previous study of errors,² where the relative forward and backward errors were obtained for the matrix cosine function. In this paper we use the same methodology for the series $T_c(X)$ and $T_s(X)$.

Let $P_m(X)$ and $Q_m(X)$ be the Taylor approximations, respectively, of $T_c(X)$ and $T_s(X)$, and $E_{f,m}[T_c(X)]$ and $E_{f,m}[T_s(X)]$ their relative forward errors. If $T_c(X)$ is invertible, then it follows that:

$$E_{f,m}[T_c(X)] = \|T_c^{-1}(X) (T_c(X) - P_m(X))\| = \left\| \sum_{i \geq m+1} d_i^{(m)} X^i \right\|,$$

where the coefficients $d_i^{(m)}$ depend on Taylor approximation of order m . If we define $f_{m+1}^{(c)}(x) = \sum_{i \geq m+1} d_i^{(m)} x^i$ and $\tilde{f}_{m+1}^{(c)}(x) = \sum_{i \geq m+1} |d_i^{(m)}| x^i$, and we apply Theorem 2, then

$$E_{f,m}[T_c(X)] = \|f_{m+1}^{(c)}(X)\| \leq \tilde{f}_{m+1}^{(c)}(\beta_{c,t}^{(m)}), \tag{31}$$

for every $t, 1 \leq t \leq m + 1$. Following established notation,¹⁵ Sec. 5.1 in (31) we denote by $\beta_{c,t}^{(m)}$ the corresponding value of β_t from Theorem 2 for order m .

In the same way, if $T_s(X)$ is invertible, we can obtain the relative forward error of Taylor approximation of order m , $E_{f,m}[T_s(X)]$, for $T_s(X)$:

$$E_{f,m}[T_s(X)] = \|T_s^{-1}(X) (T_s(X) - Q_m(X))\| = \left\| \sum_{i \geq m+1} \bar{d}_i^{(m)} X^i \right\|.$$

If we define $f_{m+1}^{(s)}(x) = \sum_{i \geq m+1} \bar{d}_i^{(m)} x^i$ and $\tilde{f}_{m+1}^{(s)}(x) = \sum_{i \geq m+1} |\bar{d}_i^{(m)}| x^i$, and we apply Theorem 2, then

$$E_{f,m}[T_s(X)] = \|f_{m+1}^{(s)}(X)\| \leq \tilde{f}_{m+1}^{(s)}(\beta_{s,t}^{(m)}). \tag{32}$$

Let $\Theta_m^{f,c}$ and $\Theta_m^{f,s}$ be

$$\Theta_m^{f,c} = \max \left\{ \theta \geq 0 : \sum_{i \geq m+1} |d_i^{(m)}| \theta^i \leq u \right\},$$

$$\Theta_m^{f,s} = \max \left\{ \theta \geq 0 : \sum_{i \geq m+1} |\bar{d}_i^{(m)}| \theta^i \leq u \right\},$$

where $u = 2^{-53}$ is the unit roundoff in double precision floating-point arithmetic.

For obtaining $\Theta_m^{f,c}$ and $\Theta_m^{f,s}$ we have used MATLAB Symbolic Math Toolbox to evaluate $\sum_{i \geq m+1} |d_i^{(m)}| \theta^i$ and $\sum_{i \geq m+1} |\bar{d}_i^{(m)}| \theta^i$ for each m in 250-digit decimal arithmetic, adding the first 250 series terms with the coefficients obtained symbolically. Then, a numerical zero-finder is invoked to determine the highest values of $\Theta_m^{f,c}$ and $\Theta_m^{f,s}$ such that $\sum_{i \geq m+1} |d_i^{(m)}| (\Theta_m^{f,c})^i \leq u$ and $\sum_{i \geq m+1} |\bar{d}_i^{(m)}| (\Theta_m^{f,s})^i \leq u$ hold.

Let be $\Theta_m^f = \min \{ \Theta_m^{f,c}, \Theta_m^{f,s} \} = \Theta_m^{f,c}$, $m = 1, 2, 4, 6, 9, 12, 16$. From (31) and (32), if $\beta_t^{(m)} \leq \Theta_m^f$, we obtain

$$E_{f,m}[T_c(X)], \quad E_{f,m}[T_s(X)] \leq u. \tag{33}$$

The values of $\Theta_m^{f,c}$, and $\Theta_m^{f,s}$ are shown in Table 1.

Algorithm 2 Given $X_0, \dot{X}_0, Y_0, \dot{Y}_0 \in \mathbb{R}^{n \times n}$, $h \in \mathbb{R}$ and $A \in \mathbb{R}^{n \times n}$, this algorithm calculates approximately the solutions of differential equations (8) and (9), $X_i, \dot{X}_i, Y_i, \dot{Y}_i$, at $t_i = ih$, $i = 1, 2, \dots, r$.

```

1: for  $i = 1 : r$  do
2:    $t_i = t_{i-1} + h$ 
3:   Compute  $C \cong T_c(Ah^2)$  and  $S \cong T_s(Ah^2)$  by using Algorithm 1
4:    $X_i = X_{i-1}C + h\dot{X}_{i-1}S$  ▷ Expression (17)
5:    $\dot{X}_i = -hX_{i-1}AS + \dot{X}_{i-1}C$  ▷ Expression (18)
6:    $Y_i = CY_{i-1} + hSY_{i-1}$  ▷ Expression (22)
7:    $\dot{Y}_i = -hASY_{i-1} + C\dot{Y}_{i-1}$ . ▷ Expression (23)
8: end for

```

Algorithm 3 Given $X_0, \dot{X}_0, Y_0, \dot{Y}_0 \in \mathbb{R}^{n \times n}$, $A \in \mathbb{R}^{n \times n}$, this algorithm calculates approximately the solutions $X = X(t)$, $\dot{X} = \dot{X}(t)$, $Y = Y(t)$, $\dot{Y} = \dot{Y}(t)$, $t \in [0, T]$ of the differential equations (8) and (9).

```

1: Compute  $C \cong T_c(At^2)$  and  $S \cong T_s(At^2)$  by using Algorithm 1
2:  $X = X_0C + t\dot{X}_0S$  ▷ Expression (19)
3:  $\dot{X} = -tX_0AS + \dot{X}_0C$  ▷ Expression (20)
4:  $Y = CY_0 + tSY_0$  ▷ Expression (24)
5:  $\dot{Y} = -tASY_0 + C\dot{Y}_0$ . ▷ Expression (25)

```

5.2 | Backward errors

The backward errors ΔX and $\Delta \bar{X}$ of approximating $T_c(X)$ and $T_s(X)$, $X \in \mathbb{C}^{n \times n}$, by Taylor approximations $P_m(X)$ and $Q_m(X)$ satisfy

$$P_m(X) = T_c(X + \Delta X), \quad Q_m(X) = T_s(X + \Delta \bar{X}).$$

Using a previously developed approach,² p. 69 for the series $T_c(X)$ and $T_s(X)$, the backward errors ΔX and $\Delta \bar{X}$ can be expressed as the series

$$\Delta X = \sum_{i \geq m+1} e_i^{(m)} X^k, \quad \Delta \bar{X} = \sum_{k \geq m+1} \bar{e}_i^{(m)} X^k.$$

where the coefficients $e_i^{(m)}$ and $\bar{e}_i^{(m)}$ can be computed symbolically by using the MATLAB Symbolic Math Toolbox.

The relative backward error of computing $T_c(X)$ can be calculated as

$$E_{b,m}[T_c(X)] = \frac{\|\Delta X\|}{\|X\|} = \frac{\left\| \sum_{i \geq m} e_i^{(m)} X^{i+1} \right\|}{\|X\|} \leq \left\| \sum_{i \geq m} e_i^{(m)} X^i \right\|.$$

If we define $g_m^{(c)}(x) = \sum_{i \geq m} e_i^{(m)} x^i$, $\tilde{g}_m^{(c)}(x) = \sum_{i \geq m} |e_i^{(m)}| |x|^i$, and we apply Theorem 2, then

$$E_{b,m}[T_c(X)] = \left\| g_m^{(c)}(X) \right\| \leq \tilde{g}_m^{(c)}(\gamma_{c,t}^{(m)}), \quad (34)$$

for every t , $1 \leq t \leq m+1$, where $\gamma_{c,t}^{(m)}$ is the corresponding value of β_t from Theorem 2 for order m .

In the same way, the backward error ΔX of approximating $T_s(X)$ by Taylor approximation $Q_m(X)$ can be calculated as

$$E_{b,m}[T_s(X)] = \frac{\|\Delta X\|}{\|X\|} = \frac{\left\| \sum_{i \geq m} \bar{e}_i^{(m)} X^{i+1} \right\|}{\|X\|} \leq \left\| \sum_{i \geq m} \bar{e}_i^{(m)} X^i \right\|.$$

If we define $g_m^{(s)}(x) = \sum_{i \geq m} \bar{e}_i^{(m)} x^i$ and $\tilde{g}_m^{(s)}(x) = \sum_{i \geq m} |\bar{e}_i^{(m)}| |x|^i$, and we apply Theorem 2, then

$$E_{b,m}[T_s(X)] = \left\| g_m^{(s)}(X) \right\| \leq \tilde{g}_m^{(s)}(\gamma_{s,t}^{(m)}), \quad (35)$$

for every $t, 1 \leq t \leq m + 1$, where $\gamma_{s,t}^{(m)}$ the corresponding value of β_t from Theorem 2 for order m .

Let $\Theta_m^{b,c}$ and $\Theta_m^{b,s}$ be

$$\Theta_m^{b,c} = \max \left\{ \theta \geq 0 : \sum_{i \geq m+1} |e_i^{(m)}| \theta^i \leq u \right\},$$

$$\Theta_m^{b,s} = \max \left\{ \theta \geq 0 : \sum_{i \geq m+1} |\bar{e}_i^{(m)}| \theta^i \leq u \right\},$$

Values $\Theta_m^{b,c}$ and $\Theta_m^{b,s}$ can be calculated in the same way that the values of $\Theta_m^{f,c}$ and $\Theta_m^{f,s}$ were calculated. Those values of $\Theta_m^{b,c}$ and $\Theta_m^{b,s}$ are shown in Table 1. Let be $\Theta_m^b = \min \{ \Theta_m^{b,c}, \Theta_m^{b,s} \} = \Theta_m^{b,c}, m \in \{1, 2, 4, 6, 9, 12, 16\}$. From (34) and (35), if $\beta_t^{(m)} \leq \Theta_m^b$, we obtain

$$E_{b,m}[T_c(X)], \quad E_{b,m}[T_s(X)] \leq u. \tag{36}$$

	Θ_m^f	Θ_m^b	Θ_m
$m = 2$	$4.307691257 \times 10^{-5}$	$3.419034688 \times 10^{-5}$	$4.307691257 \times 10^{-5}$
$m = 4$	$1.319680930 \times 10^{-2}$	$1.149925454 \times 10^{-2}$	$1.319680930 \times 10^{-2}$
$m = 6$	$1.895232414 \times 10^{-1}$	$1.733508225 \times 10^{-1}$	$1.895232414 \times 10^{-1}$
$m = 9$	1.5605489459×10^0	1.5886273831×10^0	1.5886273831×10^0
$m = 12$	2.7001120248×10^0	5.6861650847×10^0	5.6861650847×10^0
$m = 16$	3.3335993055×10^0	1.0447593515×10^1	1.0447593515×10^1

TABLE 2 Values of Θ_m^f, Θ_m^b and $\Theta_m^{b,c}$ and $\Theta(m) = \max \{ \Theta_m^f, \Theta_m^b \}$ for orders $m = 2, 4, 6, 9, 12, 16$.

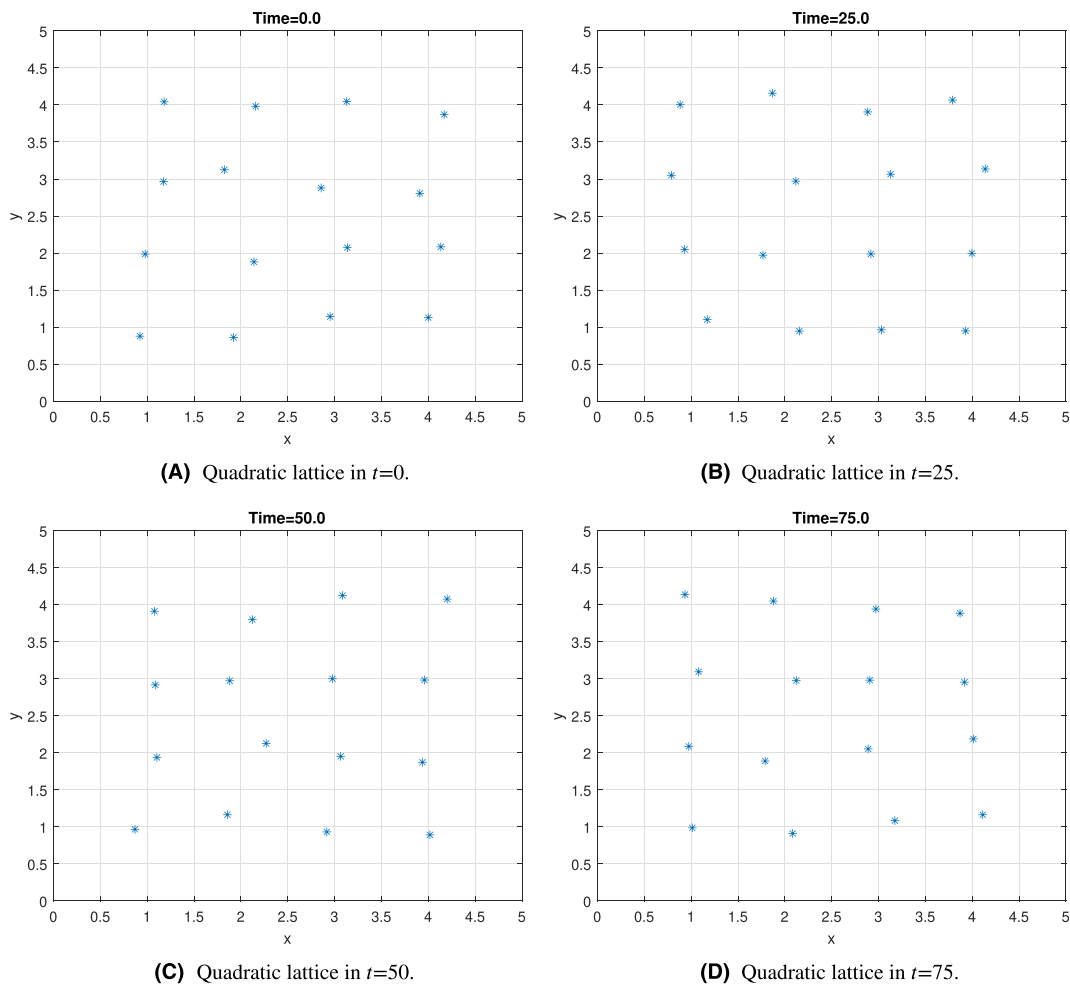


FIGURE 2 Different snapshots of the lattice configuration at instants $t = 0, 25, 50, 75$. The lattice is composed of 4×4 equidistant elements with $\sigma = 1$ and $m = 1$ [Colour figure can be viewed at wileyonlinelibrary.com]

Analogously to previous work,² Sec. 2 to minimize the computational cost, we select $\Theta(m) = \max \{ \Theta_m^f, \Theta_m^b \}$, i.e. $\Theta(m) = \Theta_m^f$ for $m \in \{1, 2, 4, 6\}$, and $\Theta(m) = \Theta_b(m)$ for $m \in \{9, 12, 16\}$. Then, considering (33) and (36) and taking into account the values of Θ_m from Table 2, it follows that if $\beta_t^{(m)} \leq \Theta(m)$, $m \in \{1, 2, 4, 6\}$, then the relative forward errors are lower than or equal to the unit roundoff, and if $\beta_t^{(m)} \leq \Theta(m)$, $m \in \{9, 12, 16\}$, then the relative backwards errors are lower than or equal to the unit roundoff:

$$\begin{aligned} E_{f,m}[T_s(X)], \quad E_{f,m}[T_c(X)] &\leq u, \quad m \in \{1, 2, 4, 6\} \\ E_{b,m}[T_s(X)], \quad E_{b,m}[T_c(X)] &\leq u, \quad m \in \{9, 12, 16\}. \end{aligned} \quad (37)$$

Analogously to previous work,² Prop. 1 it is easy to show that for $\|X\| < \Theta_6$, $T_c(X)$ and $T_s(X)$ are invertible.

5.3 | Computation of the scaling factor and the order of Taylor approximation

For computing the scaling factor s and order m of the Taylor approximation two algorithms have been developed, which are based on Algorithms 2 and 3 of previous work by Sastre et al.² If there exists a value $m_k \leq 16$ such that $\beta_{\min}^{(m_k)} \leq \Theta(m_k)$, then one of the inequalities from (37) is satisfied, and in this case we choose the lower order m_k so that $\beta_{\min}^{(m_k)} \leq \Theta(m_k)$ with a scaling $s = 0$. Otherwise, we choose the Taylor approximation of order 12 or 16 providing the lower cost, with

$$s = \max \left\{ 0, \left\lceil \frac{1}{2} \log \left(\frac{\beta_{\min}^{m_k}}{\Theta(m_k)} \right) \right\rceil \right\}, \quad m_k = 12 \text{ or } 16.$$

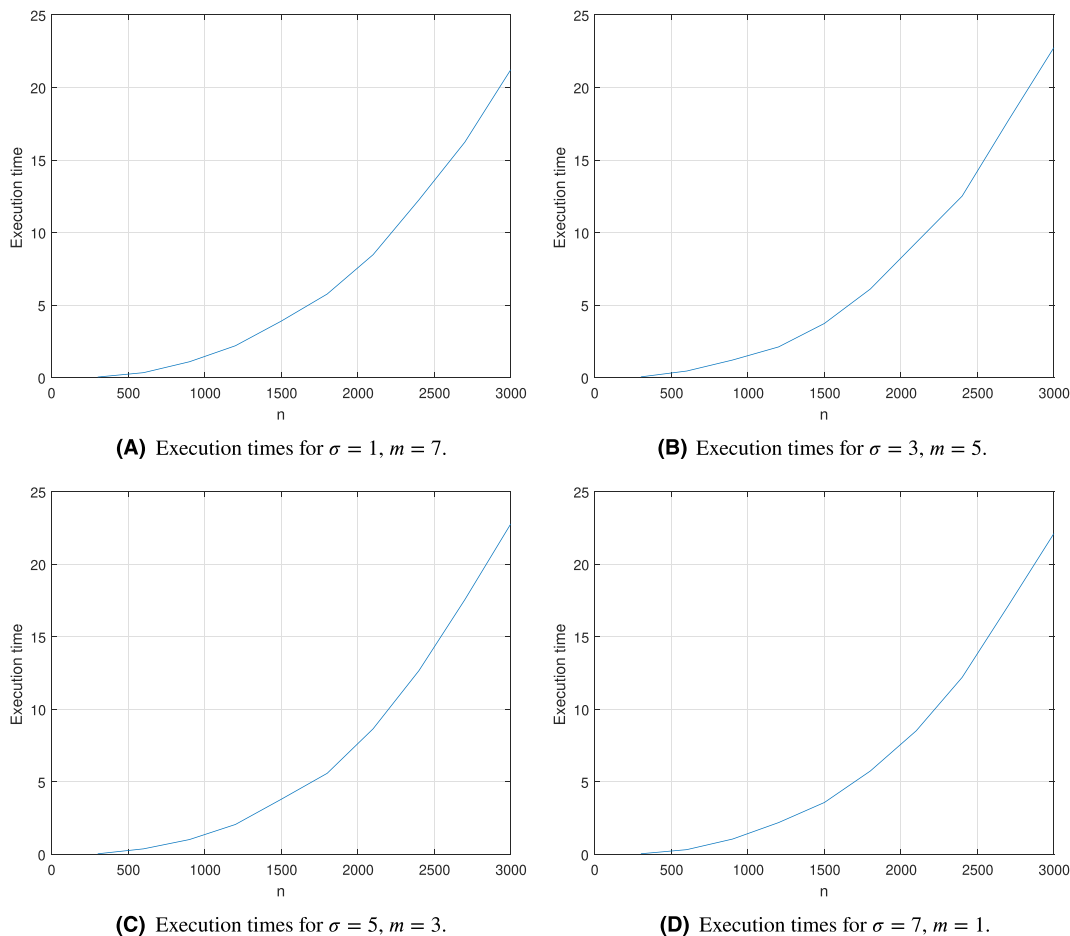


FIGURE 3 Plots of the execution times at instant $t = 1000$ s with varying n , implemented with MATLAB (R2017b) running on an INTEL Core i7-6700HQ 2.60 GHz processor with 16 GB of RAM [Colour figure can be viewed at wileyonlinelibrary.com]

For computing $\beta_{min}^{m_k}$ we use the simplification (30):

$$\beta_{min}^{(m_k)} \approx \max \left\{ d_{m_k+1}^{1/(m_k+1)}, d_{m_k+2}^{1/(m_k+2)} \right\}, \quad m_k \leq 6 \quad (\text{forward bound}),$$

$$\beta_{min}^{(m_k)} \approx \max \left\{ d_{m_k}^{1/m_k}, d_{m_k+1}^{1/(m_k+1)} \right\} \quad m_k \geq 9 \quad (\text{backward bound}),$$

where the bounds $d_l, l = m_k, m_{k+1}, m_{k+2}$, can be computed as detailed below:

- By estimating the required norms of matrix powers using the estimation algorithm from Higham.¹⁷ For example, since for $m_k = 4$ only A and A^2 are available, then we compute approximately $d^5 \cong \|A^5\|, d^6 \cong \|A^6\|$ from A and A^2 using algorithm from Higham.¹⁷ Hence,

$$\beta_{min}^{(4)} = \max \left\{ d_5^{1/5}, d_6^{1/6} \right\}.$$

- Without estimation of norms of matrix powers. In this case, the bounds of matrix powers are computed from products of norms of matrix powers previously computed (see Algorithms 2 and 3 from Sastre et al.²). In this case, $\beta_{min}^{(4)}$ is computed as

$$\beta_{min}^{(4)} = \max \left\{ \left(\|A^2\|^2 \|A\| \right)^{1/5}, \left(\|A^2\|^3 \right)^{1/6} \right\} = \left(\|A^2\|^2 \|A\| \right)^{1/5}.$$

The MATLAB implementations of Algorithms 1 and 2 have been developed previously² and are available to the public at <https://personales.upv.es/jorsasma/software/cosmtay.m>.

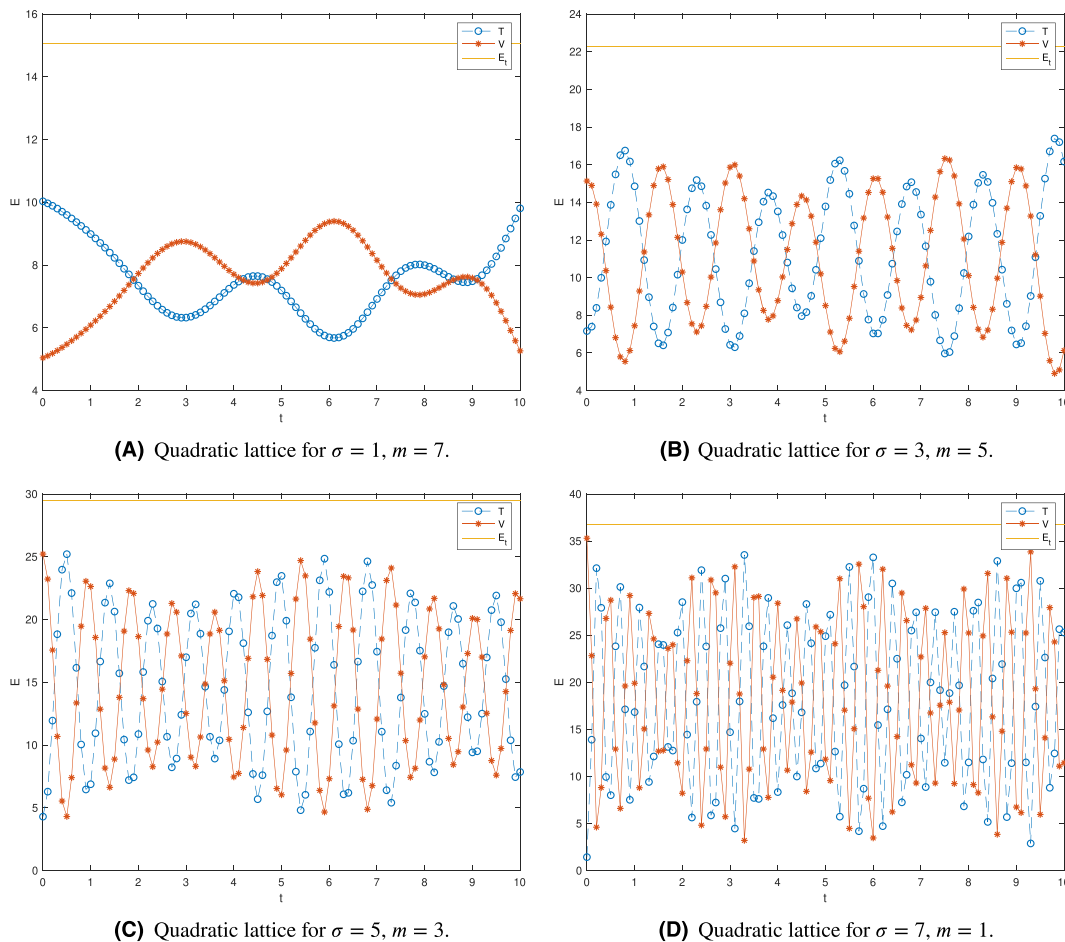


FIGURE 4 Capturing the energy distribution of the homogeneous square lattice for time interval $[0, 10]$. For all oscillators the sum of kinetic energy (T), potential energy (V), and total energy ($E = T + V$) is shown. The values for spring constant σ and mass m vary, whereas the lattice dimension of 4×4 is kept unaltered [Colour figure can be viewed at wileyonlinelibrary.com]

6 | ALGORITHMS FOR COMPUTING THE SOLUTIONS OF THE DYNAMICS OF HARMONIC OSCILLATORS ON THE LATTICE PROBLEM

Algorithm 2 computes the displacements $X(t)$ and $Y(t)$ at $t = h, 2h, \dots, rh$. In each step of the **for** loop the matrices $X(t_i)$, $\dot{X}(t_i)$, $Y(t_i)$ and $\dot{Y}(t_i)$ are obtained from $X(t_{i-1})$, $\dot{X}(t_{i-1})$, $Y(t_{i-1})$ and $\dot{Y}(t_{i-1})$ by using expressions (17)–(18) and (22)–(23).

In some cases it may be convenient to find the solution at a certain instant. Algorithm 3 allows to obtain the solution at a time t from the initial conditions. This algorithm computes the displacements $X(t)$ and $Y(t)$ at t from X_0 , \dot{X}_0 , Y_0 and \dot{Y}_0 by using expressions (19)–(20) and (24)–(25).

7 | NUMERICAL RESULTS

The authors have implemented the MATLAB functions of Algorithms 1, 2 and 3. In addition a MATLAB program has been developed allowing to create an animation with the evolution of the harmonic oscillators. All implementations are available at https://personales.upv.es/jjibanez/software/Lattice_oscilators_programs.zip.

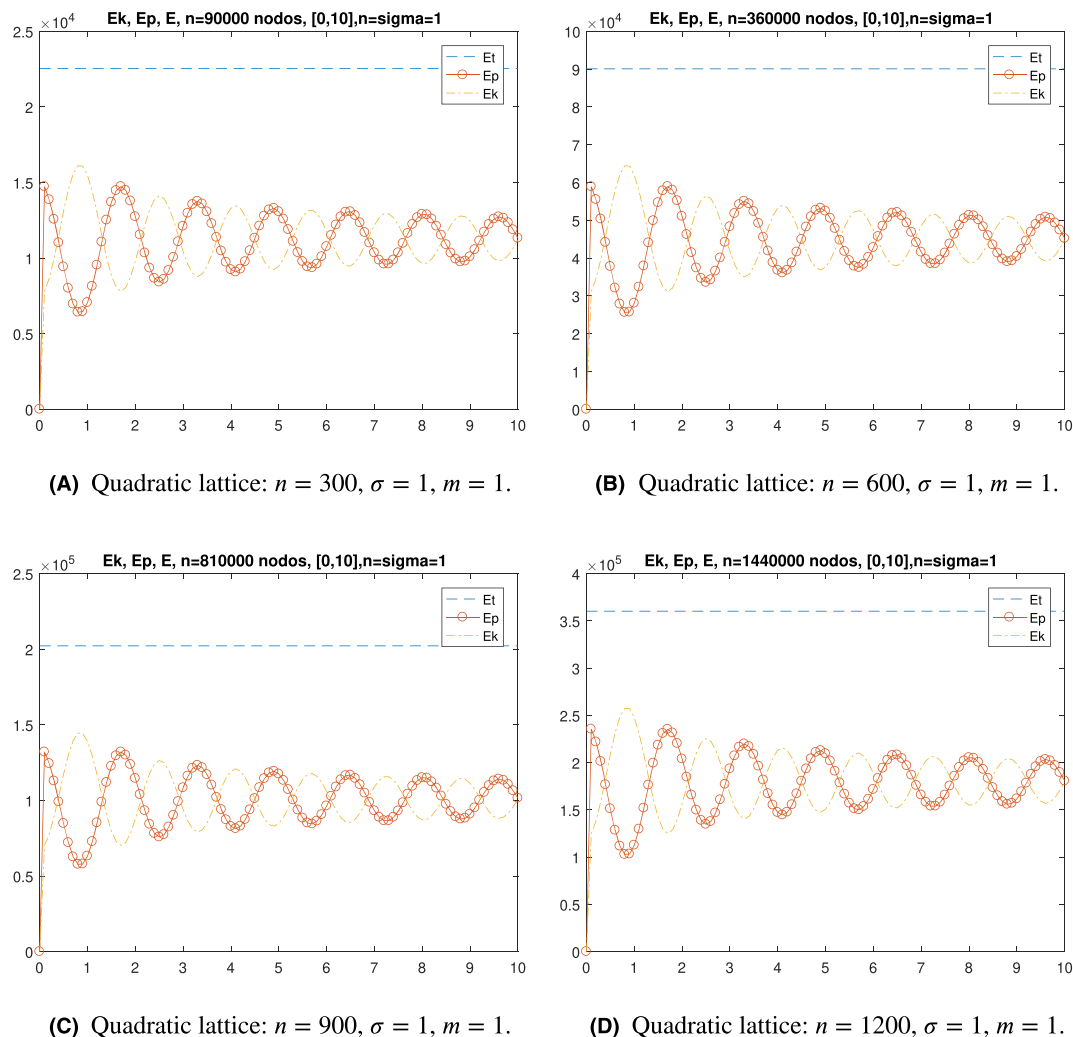


FIGURE 5 Capturing the energy distribution of the homogeneous square lattice for time interval $[0, 10]$. For all oscillators the sum of kinetic energy (T), potential energy (V), and total energy ($E = T + V$) is shown. The values for spring constant σ and mass m are kept unaltered, whereas the lattice dimension varies [Colour figure can be viewed at wileyonlinelibrary.com]

For the simulations, MATLAB (R2017b) running on an INTEL Core i7-6700HQ 2.60 GHz processor with 16 GB of RAM. In all tests we use a quadratic lattice composed of n equidistant nodes with an unit spacing $a = 1$. The initial conditions $X_0, \dot{X}_0, Y_0, \dot{Y}_0 \in \mathbb{R}^{n \times n}$ were randomly generated such that their elements are in the interval $[-0.2, 0.2]$.

By using Algorithm 2, Figure 2 displays several snapshots of the lattice configuration in the time interval $[0, 75]$ for instants $t = 0, 25, 50, 75$ for a lattice composed of 4^2 equidistant nodes.

By using Algorithm 3, Figure 3 shows the execution times in seconds when the solution is obtained at $t = 1000$, varying n , for different values of m and σ . Similar graphs have been obtained with different values of m and σ .

By using Algorithm 2, Figure 4 and 5 illustrate the evolution of the harmonic oscillators in terms of energies. The graphs depict kinetic energy T (see Equation (1)), potential energy V (see Equation (2)), and total energy E for time interval $[0, 10]$, varying the values of the parameters σ and m in Figure 4, and varying the lattice dimension in Figure 5. For all cases the total energy (10) of the system is conserved, as seen by the graph of E being a horizontal line marked in yellow.

8 | CONCLUSIONS

The present work focusses on a new technique to model the dynamics of harmonic oscillators on an ideal 2D lattice. In the Lagrangian framework, the associated equations of motion entail matrix differential equations of second order (and their initial values) of the type Equations (8) and (9). Formally, their solutions can be written in terms of the matrix sine and cosine. For an efficient numerical evaluation of these matrix functions, we have proposed a method based on the convergent series. Equation (16). This allows to obtain the final solutions as a combination of them in a very compact manner.

To carry out the computation of these matrix series, we have used a Taylor approximation with a modified version of the scaling algorithm.² For the simultaneous computation of the two series in Equation (16), new forward and backward error bounds for the Taylor approximation had to be determined, allowing to implement the optimal scaling parameter and the optimal order for the Taylor method.

Finally, several new MATLAB functions have been implemented to create animations of the dynamical evolution of the lattice and to confirm the conservation of energy for the system. Snapshots of these animations help to visualize that indeed for all cases the total energy of the system, Equation (10), is conserved.

It is our hope that the proposed numerical treatment of trigonometric matrix functions will open up new avenues not only for lattice simulations but also for other important applications in physics and engineering.

ACKNOWLEDGEMENTS

This work has been supported by the Spanish *Ministerio de Economía y Competitividad*, the European Regional Development Fund (ERDF) under grant TIN2017-89314-P, and the *Programa de Apoyo a la Investigación y Desarrollo 2018* (PAID-06-18) of the Universitat Politècnica de València under grant SP20180016.

CONFLICTS OF INTEREST

This work does not have any conflicts of interest.

ORCID

Javier Ibáñez  <https://orcid.org/0000-0002-6912-4453>

Jorge Sastre  <https://orcid.org/0000-0002-8612-6717>

REFERENCES

1. Lanczos C. *The Variational Principles of Mechanics*. New York, NY: Dover; 1986.
2. Sastre J, Ibáñez J, Alonso P, Peinado J, Defez E. Two algorithms for computing the matrix cosine function. *Appl Math Comput*. 2017;312:66–77.
3. Dehghan M, Hajarian M. Determination of a matrix function using the divided difference method of newton and the interpolation technique of hermite. *J Comput Appl Math*. 2009;231(1):67–81.
4. Dehghan M, Hajarian M. Computing matrix functions using mixed interpolation methods. *Math Comput Model*. 2010;52(5-6):826–836.

5. Kazem S, Dehghan M. Application of finite difference method of lines on the heat equation. *Numer Methods Partial Differ Equ.* 2018;34(2):626–660.
6. Kazem S, Dehghan M. Semi-analytical solution for time-fractional diffusion equation based on finite difference method of lines (mol). *Eng Comput.* 2019;35(1):229–241.
7. Olver PJ. *Applications of Lie Groups to Differential Equations.* New York, NY: Springer-Verlag; 2008.
8. Dunford N, Schwartz J. *Linear Operators, Part I.* New York, NY: Interscience; 1957.
9. Golub GH, Van Loan C. *Matrix Computations.* Baltimore, MD: The Johns Hopkins University Press; 1996.
10. Paterson MS, Stockmeyer LJ. On the number of nonscalar multiplications necessary to evaluate polynomials. *SIAM J Comput.* 1973;2(1):60–66.
11. Sastre J, Ibáñez JJ, Defez E, Ruiz PA. Efficient orthogonal matrix polynomial based method for computing matrix exponential. *Appl Math Comput.* 2011;217:6451–6463.
12. Higham NJ. *Functions of Matrices: Theory and Computation.* Philadelphia, PA, USA: SIAM; 2008.
13. Sastre J, Ibáñez JJ, Defez E, Ruiz PA. Accurate matrix exponential computation to solve coupled differential models in engineering. *Math Comput Model.* 2011;54:1835–1840.
14. Serbin SM, Blalock SA. An algorithm for computing the matrix cosine. *SIAM J Sci Statist Comput.* 1980;1(2):198–204.
15. Sastre J, Ibáñez J, Ruiz P, Defez E. Efficient computation of the matrix cosine. *Appl Math Comput.* 2013;219:7575–7585.
16. Ruiz P, Sastre J, Ibáñez J, Defez E. High performance computing of the matrix exponential. *J Comput Appl Math.* 2016;291:370–379.
17. Higham NJ. Fortran codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation. *ACM Trans Math Softw.* 1988;14(4):381–396.

How to cite this article: Tung MM, Ibáñez J, Defez E, Sastre J. Simulation of harmonic oscillators on the lattice. *Math Meth Appl Sci.* 2020;43:8237–8252. <https://doi.org/10.1002/mma.6510>