On the reliable and efficient numerical integration of the Kuramoto model and related dynamical systems on graphs

T. Böhle[a], C. Kuehn[a], M. Thalhammer[b]

[a]Technische Universität München, Fakultät für Mathematik,
Boltzmannstraße 3, 85748 Garching, Germany.
[b]Leopold–Franzens Universität Innsbruck, Institut für Mathematik,
Technikerstraße 13/7, 6020 Innsbruck, Austria.

ABSTRACT
In this work, a novel approach for the reliable and efficient numerical integration of the Kuramoto model on graphs is studied. For this purpose, the notion of order parameters is revisited for the classical Kuramoto model describing all-to-all interactions of a set of oscillators. First numerical experiments confirm that the precomputation of certain sums significantly reduces the computational cost for the evaluation of the right-hand side and hence enables the simulation of high-dimensional systems. In order to design numerical integration methods that are favourable in the context of related dynamical systems on network graphs, the concept of localised order parameters is proposed. In addition, the detection of communities for a complex graph and the transformation of the underlying adjacency matrix to block structure is an essential component for further improvement. It is demonstrated that for a submatrix comprising relatively few coefficients equal to zero, the precomputation of sums is advantageous, whereas straightforward summation is appropriate in the complementary case. Concluding theoretical considerations and numerical comparisons show that the strategy of combining effective community detection algorithms with the localisation of order parameters potentially reduces the computation time by several orders of magnitude.

1. Introduction

In the present work, we propose a novel approach for the reliable and efficient numerical integration of nonlinear dynamical systems on network graphs and provide various numerical comparisons confirming its potential. In essence, our objective is to combine effective algorithms for the detection of communities with the concept of localised order parameters based on the precomputation of certain sums. For the sake

---

Dedicated to Jesús María Sanz-Serna due to his seminal contributions in the area of Geometric numerical integration.

M. Thalhammer. Email: mechthild.thalhammer@uibk.ac.at

of concreteness, we focus on Kuramoto-type models for a large set of individual oscillators and graphs that are determined by adjacency matrices with coefficients equal to one and zero, respectively. Generally, the interactions between the oscillators are described by a system of nonlinear ordinary differential equations involving the sines of the associated phases. The classical Kuramoto–Daido model [9, 23] reflects the case of all-to-all coupling and thus corresponds to a complete graph. Realistic extended models incorporate complex networks [32]. Yet, the efficient numerical integration of a Kuramoto-type model on a graph comprising a high number of nodes is a relevant issue, and even the evaluation of the vector field defining the right-hand side of the system poses a major challenge. This problem actually permeates the simulation of complex dynamics on networks [28]. In order to exemplify our strategy and to illustrate its capability, we initially consider the classical Kuramoto–Daido model and revisit the well-known notion of order parameters [2, 9, 23, 26, 37, 38]. An essential component of our procedure for Kuramoto-type models on graphs is the detection of communities [13, 29], since this permits a transformation of the associated adjacency matrix to block structure. We demonstrate that for a submatrix comprising relatively few coefficients equal to zero the precomputation of sums is indeed advantageous and that straightforward summation is suitable in the complementary case. We conclude with theoretical considerations and numerical comparisons, which show that our approach ensures a significant reduction of the required memory capacity as well as the computational cost for the evaluation of the right-hand side. As a consequence, long-term simulations of systems involving a large number of oscillators by geometric integrators [3, 19, 34, 35] are within reach.

Scope of the model. The starting point of our investigations is the Kuramoto–Daido model [9, 23]. We henceforth refer to it as (classical) Kuramoto model. This system of coupled nonlinear ordinary differential equations is a fundamental mathematical model for the dynamical behaviour of a set of weakly coupled, nearly identical oscillators and specifies the time evolution of the associated phases. Despite its simple structure, the Kuramoto model exhibits fascinating phenomena such as synchronisation and phase locking [37]. Originally introduced to describe processes in chemistry and biology [23, 40], it was found to have various applications in other fields such as physics, neuroscience, and engineering [1, 11, 14]. From a theoretical perspective, the Kuramoto model has deep connections to effects present in Hamiltonian systems, particularly to Landau damping [10, 12] and bifurcations from essential spectra [6]. A variety of Kuramoto-type models have a gradient flow structure entailing further interesting mathematical cross-connections [16].

Kuramoto model. The main terms defining the right-hand side of the classical Kuramoto model for M individual oscillators have the form

$$\frac{1}{M} \sum_{\ell=1}^{M} \sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big), \quad t \in [0, T], \quad m \in \{1, 2, \ldots, M\}. \tag{1}$$

Here, ϑ_m denotes the time-dependent phase of the m-th oscillator, which takes values in the circle $\mathbb{S}_1 = \mathbb{R}/(2\pi\mathbb{Z})$. Henceforth, we employ the convenient vector notation

$$\vartheta = (\vartheta_1, \ldots, \vartheta_M)^T : [0, T] \longrightarrow \mathbb{S}_1^M.$$

With regard to the numerical simulation of a high number of oscillators, an essential requirement is the efficient evaluation of these sums at certain time grid points.

Thereby, one issue is the limited memory capacity. As an example, we mention the currently widely used software MATLAB, which has a maximum array size preference of about 10^{10} (74.5 GB) corresponding to a square matrix of dimension 10^5. Consequently, it is desirable to avoid the creation of the matrix

$$\big(\vartheta_\ell(t) - \vartheta_m(t)\big)_{\ell,m\in\{1,2,\ldots,M\}}\,, \quad t \in [0,T]\,,$$

since this would restrict the dimension of the system considerably. An important alternative for the efficient evaluation of (1) relies on the macroscopic order parameters, which are given through

$$r\big(\vartheta(t)\big)\,\mathrm{e}^{\mathrm{i}\,\psi(\vartheta(t))} = \tfrac{1}{M}\sum_{m=1}^{M}\mathrm{e}^{\mathrm{i}\,\vartheta_m(t)}\,, \quad t \in [0,T]\,.$$

More precisely, applying the addition theorem for the sine function to (1), we obtain the following reformulation

$$S_M\big(\vartheta(t)\big) = \tfrac{1}{M}\sum_{m=1}^{M}\sin\big(\vartheta_m(t)\big)\,, \quad C_M\big(\vartheta(t)\big) = \tfrac{1}{M}\sum_{m=1}^{M}\cos\big(\vartheta_m(t)\big)\,,$$

$$\tfrac{1}{M}\sum_{\ell=1}^{M}\sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big) = S_M\big(\vartheta(t)\big)\,\cos\big(\vartheta_m(t)\big) - C_M\big(\vartheta(t)\big)\,\sin\big(\vartheta_m(t)\big)\,, \tag{2}$$

$$t \in [0,T]\,, \quad m \in \{1,2,\ldots,M\}\,.$$

An evident though crucial observation is that precomputing the sums $S_M(\vartheta(t))$ and $C_M(\vartheta(t))$ permits to evaluate $S_M(\vartheta(t))\cos(\vartheta_m(t)) - C_M(\vartheta(t))\sin(\vartheta_m(t))$ for $m \in \{1,2,\ldots,M\}$ in an efficient manner. Numerical comparisons described in Section 2 confirm that this approach reduces the number of function evaluations and thus the computation time considerably.

Kuramoto model on graphs. As indicated above, it is of high relevance to study extensions of the classical Kuramoto model in the context of dynamical networks [32]. We focus on situations, where the sum over all phases is replaced by a sum over certain phases. That is, the right-hand side of the system involves terms of the form

$$\sum_{\ell=1}^{M} A_{m\ell}\,\sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big)\,, \quad A_{m\ell} \in \{0,1\}\,, \quad t \in [0,T]\,, \quad \ell, m \in \{1,2,\ldots,M\}\,.$$

The associated matrix

$$A = \big(A_{m\ell}\big)_{\ell,m\in\{1,\ldots,M\}} \in \mathbb{R}^{M\times M}$$

has the natural interpretation as the adjacency matrix of a graph. In Section 3, we consider the complementary cases of sparse and dense adjacency matrices. Numerical tests for randomly generated matrices show that the precomputation of sums is advantageous whenever the number of coefficients equal to one is larger than the number of coefficients equal to zero, whereas straightforward summation is appropriate otherwise. Subsequently, we examine algorithms detecting communities in a graph, which

3

yield as outputs partitions of the nodes, and extend our approach to block matrices. In our considerations, we do not presume a symmetric adjacency matrix, which would lead to a gradient system.

Generalisations. We point out that our approach applies to the more general case, where the coefficients of the adjacency matrix take values in a finite set, for instance

$$A_{m\ell} \in \{0, 1, \ldots, J\}, \quad \ell, m \in \{1, 2, \ldots, M\}.$$

Provided that a suitable permutation permits the transformation to a block matrix such that a single value is prevalent in each block and the total number of blocks is relatively small, we may expect a significant gain in efficiency by the precomputation of sums. Moreover, it is straightforward to generalise our approach to systems that in addition involve multiple sums such as

$$\sum_{j,k,\ell=1}^{M} A_{mjk\ell} \sin\big(\vartheta_j(t) + \vartheta_k(t) - \vartheta_\ell(t) - \vartheta_m(t)\big), \quad t \in [0, T], \quad m \in \{1, 2, \ldots, M\}.$$

Due to the fact that higher-order Kuramoto models describe interactions beyond usual graph structures, they have recently raised remarkable interest, see [4, 36] and references given therein. However, as their incorporation requires laborious notation and would obstruct a comprehensible presentation of the key idea, we do not include detailed calculations here.

Remarks. We note that suitable reformulations of the classical Kuramoto model and Kuramoto-type models on graphs can also be based on complex exponentials. In view of expedient practical implementations, however, it is preferable to use real-valued quantities. On account of cross-references in captions, figures are included in the end of this manuscript.

2. Kuramoto model

In this section, we state the classical Kuramoto model and study different viewpoints on its reliable and efficient numerical integration.

Original formulation. We consider a set of M limit-cycle oscillators with time-dependent phases

$$\vartheta_m : [0, T] \longrightarrow \mathbb{S}_1, \quad m \in \{1, 2, \ldots, M\}. \tag{3a}$$

In the absence of external driving or damping forces, respectively, the oscillators have the intrinsic frequencies

$$\omega_m \in \mathbb{R}, \quad m \in \{1, 2, \ldots, M\}. \tag{3b}$$

The pairwise interactions between the oscillators are described by the following system

of nonlinear ordinary differential equations

$$\begin{cases} \vartheta'_m(t) = \omega_m + \frac{K}{M} \sum_{\ell=1}^{M} \sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big) \,, \\ \vartheta_m(0) \text{ given} \,, \quad t \in (0, T) \,, \quad m \in \{1, 2, \dots, M\} \,, \end{cases} \tag{3c}$$

where $K > 0$ denotes the coupling constant.

Special choices. In our numerical tests, we consider intrinsic frequencies defined by a single real number $\omega_0 \in \mathbb{R}$ and initial phases of the form

$$\omega_m = 1 + \omega_0 \, \tfrac{(2m - M - 1)}{M - 1} \,, \quad \vartheta_m(0) = \tfrac{2\pi m}{M} \,, \quad m \in \{1, 2, \dots, M\} \,. \tag{3d}$$

Reformulation. Regarding the efficient numerical integration of the classical Kuramoto model, it is to the best advantage to employ a reformulation that relies on elementary addition theorems for trigonometric functions

$$\sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big) = \sin\big(\vartheta_\ell(t)\big) \cos\big(\vartheta_m(t)\big) - \cos\big(\vartheta_\ell(t)\big) \sin\big(\vartheta_m(t)\big) \,, \\ \ell, m \in \{1, 2, \dots, M\} \,.$$

Introducing the abbreviations

$$\begin{aligned} S_M\big(\vartheta(t)\big) &= S_M\big(\vartheta_1(t), \vartheta_2(t), \dots, \vartheta_M(t)\big) = \tfrac{1}{M} \sum_{m=1}^{M} \sin\big(\vartheta_m(t)\big) \in \mathbb{R} \,, \\ C_M\big(\vartheta(t)\big) &= C_M\big(\vartheta_1(t), \vartheta_2(t), \dots, \vartheta_M(t)\big) = \tfrac{1}{M} \sum_{m=1}^{M} \cos\big(\vartheta_m(t)\big) \in \mathbb{R} \,, \end{aligned} \tag{4a}$$

the governing equations (3) read as

$$\begin{cases} \vartheta'_m(t) = \omega_m + K \left(S_M\big(\vartheta(t)\big) \cos\big(\vartheta_m(t)\big) - C_M\big(\vartheta(t)\big) \sin\big(\vartheta_m(t)\big) \right), \\ \vartheta_m(0) \text{ given} \,, \quad t \in (0, T) \,, \quad m \in \{1, 2, \dots, M\} \,. \end{cases} \tag{4b}$$

Employing the standard vector notation

$$\omega = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_M \end{pmatrix} \in \mathbb{R}^M \,, \quad \vartheta(t) = \begin{pmatrix} \vartheta_1(t) \\ \vartheta_2(t) \\ \vdots \\ \vartheta_M(t) \end{pmatrix} \in \mathbb{S}_1^M \,, \quad t \in [0, T] \,, \tag{4c}$$

and setting accordingly

$$\sin\big(\vartheta(t)\big) = \begin{pmatrix} \sin\big(\vartheta_1(t)\big) \\ \sin\big(\vartheta_2(t)\big) \\ \vdots \\ \sin\big(\vartheta_M(t)\big) \end{pmatrix} \,, \quad \cos\big(\vartheta(t)\big) = \begin{pmatrix} \cos\big(\vartheta_1(t)\big) \\ \cos\big(\vartheta_2(t)\big) \\ \vdots \\ \cos\big(\vartheta_M(t)\big) \end{pmatrix} \,, \tag{4d}$$

$$F\big(\vartheta(t)\big) = \omega + K \left(S_M\big(\vartheta(t)\big) \cos\big(\vartheta(t)\big) - C_M\big(\vartheta(t)\big) \sin\big(\vartheta(t)\big) \right), \quad t \in [0, T] \,,$$

5

the system takes the compact form

$$
\begin{cases}
\vartheta'(t) = F\big(\vartheta(t)\big), & t \in (0, T), \\
\vartheta(0) \text{ given}.
\end{cases}
\tag{4e}
$$

Potential. The classical Kuramoto model (3)–(4) has the intrinsic structure of a gradient system. That is, the right-hand side is given by the gradient of a real-valued potential function

$$
-\nabla V(\vartheta) = \left(\omega_m + \tfrac{K}{M} \sum_{\ell=1}^{M} \sin(\vartheta_\ell - \vartheta_m) \right)_{m \in \{1,2,\dots,M\}}
$$
$$
= \omega + K \left(S_M(\vartheta) \cos(\vartheta) - C_M(\vartheta) \sin(\vartheta) \right), \quad \vartheta = (\vartheta_1, \dots, \vartheta_M)^T \in \mathbb{S}_1^M,
$$

such that the governing equations rewrite as

$$
\begin{cases}
\vartheta'(t) = -\nabla V\big(\vartheta(t)\big), & t \in (0, T), \\
\vartheta(0) \text{ given}.
\end{cases}
$$

This in particular implies that the values of the potential decrease when time evolves

$$
\tfrac{\mathrm{d}}{\mathrm{d}t} V\big(\vartheta(t)\big) = \left(\nabla V\big(\vartheta(t)\big) \right)^T \vartheta'(t) = -\left\| \nabla V\big(\vartheta(t)\big) \right\|^2 \le 0,
$$
$$
V\big(\vartheta(t)\big) \le V\big(\vartheta(0)\big), \quad t \in [0, T].
$$

For our purposes, in view of its efficient evaluation, it is advantageous to reformulate the canonical potential as follows

$$
V : \mathbb{R}^M \longrightarrow \mathbb{R} :
$$
$$
\vartheta \longmapsto V(\vartheta) = V(\vartheta_1, \dots, \vartheta_M)
$$
$$
= -\sum_{m=1}^{M} \omega_m \vartheta_m + \tfrac{K}{2M} \sum_{\ell,m=1}^{M} \big(1 - \cos(\vartheta_\ell - \vartheta_m) \big)
\tag{5a}
$$
$$
= -\omega^T \vartheta + \tfrac{KM}{2} \left(1 - \big(C_M(\vartheta)\big)^2 - \big(S_M(\vartheta)\big)^2 \right).
$$

As mentioned below, the choice of the constant of integration is linked to the special case of synchronisation.

Order parameter. The modulus $r : \mathbb{S}_1^M \to \mathbb{R}$ and the angle $\psi : \mathbb{S}_1^M \to \mathbb{R}$ of the complex order parameter are given by

$$
r(\vartheta) \, \mathrm{e}^{\mathrm{i}\,\psi(\vartheta)} = \tfrac{1}{M} \sum_{m=1}^{M} \mathrm{e}^{\mathrm{i}\,\vartheta_m} = C_M(\vartheta) + \mathrm{i}\, S_M(\vartheta), \quad \vartheta \in \mathbb{S}_1^M.
\tag{5b}
$$

Multiplying by $\mathrm{e}^{-\mathrm{i}\,\vartheta_\ell}$ for $\ell \in \{1, 2, \dots, M\}$ and considering the imaginary part of the

resulting relation

$$r(\vartheta)\, e^{i\,(\psi(\vartheta)-\vartheta_\ell)} = \tfrac{1}{M} \sum_{m=1}^{M} e^{i\,(\vartheta_m-\vartheta_\ell)}\,, \quad r(\vartheta)\, \sin\big(\psi(\vartheta)-\vartheta_\ell\big) = \tfrac{1}{M} \sum_{m=1}^{M} \sin\big(\vartheta_m-\vartheta_\ell\big)\,,$$
$$\vartheta \in \mathbb{S}_1^M\,, \quad \ell \in \{1,2,\ldots,M\}\,,$$

the Kuramoto model (3)–(4) rewrites as

$$\begin{cases} \vartheta'(t) = \omega + K\, r\big(\vartheta(t)\big)\, \sin\Big(\psi\big(\vartheta(t)\big) - \vartheta(t)\Big)\,, \\ \vartheta(0) \text{ given}\,, \quad t \in (0,T)\,. \end{cases}$$

We point out that the order parameters contain the entire information about the interactions of the oscillators. Although this reformulation looks like a mean-field equation for a single oscillator, it completely represents the original system.

Indicator for synchronisation. For configurations, where all cosine and sine values are close-by, the modulus of the complex order parameter has values nearly one

$$\cos(\vartheta_m) \approx \cos(\vartheta_1)\,, \quad \sin(\vartheta_m) \approx \sin(\vartheta_1)\,, \quad m \in \{2,3,\ldots,M\}\,,$$
$$C_M(\vartheta) \approx \cos(\vartheta_1)\,, \quad S_M(\vartheta) \approx \sin(\vartheta_1)\,,$$
$$r(\vartheta) = \sqrt{\big(C_M(\vartheta)\big)^2 + \big(S_M(\vartheta)\big)^2} \approx 1\,, \quad \vartheta \in \mathbb{S}_1^M\,.$$

Hence, this quantity indicates synchronisation. Furthermore, in this situation, the above stated choice of the constant of integration in the potential implies

$$V(\vartheta) \approx -\,\omega^T \vartheta\,, \quad \vartheta \in \mathbb{S}_1^M\,.$$

Conserved quantity. A straightforward calculation shows that summation over all governing equations yields the identity

$$\tfrac{1}{M} \sum_{m=1}^{M} \vartheta'_m(t) = \tfrac{1}{M} \sum_{m=1}^{M} \omega_m\,, \quad t \in [0,T]\,,$$

see (3)–(4). Performing integration, this implies that the mean values of the intrinsic frequencies and the initial phases determine the mean values of the phases at later times

$$\tfrac{1}{M} \sum_{m=1}^{M} \vartheta_m(t) = \tfrac{1}{M} \sum_{m=1}^{M} \vartheta_m(0) + t\, \tfrac{1}{M} \sum_{m=1}^{M} \omega_m\,, \quad t \in [0,T]\,.$$

In other words, the dynamics of the classical Kuramoto model is restricted to a time-dependent submanifold defined by the constraint

$$\tfrac{1}{M} \sum_{m=1}^{M} \big(\vartheta_m(t) - \vartheta_m(0) - t\,\omega_m\big) = 0\,, \quad t \in [0,T]\,. \tag{5c}$$

Table 1. Classical Kuramoto model. Computational cost for the evaluation of the right-hand side based on the original formulation (3) and the reformulation (4), respectively. Number of function evaluations (sine, cosine) in dependence of the total number of oscillators.

Original formulation	$M\,(M-1)$
Reformulation	$4\,M$

We anticipate that this characteristic result extends to Kuramoto models on graphs under a certain symmetry condition, see Section 3. However, in general, this conservation property does not hold and then the quantity

$$\frac{1}{M} \sum_{m=1}^{M} \big(\vartheta_m(t) - \vartheta_m(0) - t\,\omega_m\big) \tag{5d}$$

reflects the deviation.

In the limit $M \to \infty$, the sums in (5c)–(5d) are replaced by integrals. In case of randomly chosen problem data, e.g., they are interpreted as expected phases and intrinsic frequency, respectively.

Implementation and computational cost. For systems involving a high number of oscillators $M \gg 1$, the above stated approach leading to a reformulation of the classical Kuramoto model is beneficial in several respects. It permits to reduce the required memory capacity as well as the computational cost for the evaluation of the right-hand side significantly, see Table 1. Moreover, parallelisation techniques can be used.

Exemplification (Euler). In order to exemplify our procedure for the efficient numerical integration of the Kuramoto model, we consider the simplest first-order one-step method, the explicit Euler method. For a time grid with associated stepsizes

$$0 = t_0 < \cdots < t_n < \cdots < t_N = T\,, \quad \tau_n = t_{n+1} - t_n\,, \quad n \in \{0, 1, \ldots, N-1\}\,,$$

and a prescribed initial approximation $\vartheta^{(0)} \approx \vartheta(0)$, the explicit Euler solution is given by the recurrence

$$\vartheta^{(n+1)} = \vartheta^{(n)} + \tau_n\,F\big(\vartheta^{(n)}\big)\,, \quad n \in \{0, 1, \ldots, N-1\}\,,$$

see (4). In each time step, the evaluation of the defining function relies on the precomputation of the sums

$$S_M^{(n)} = S_M\big(\vartheta^{(n)}\big)\,, \quad C_M^{(n)} = C_M\big(\vartheta^{(n)}\big)\,, \tag{6a}$$

and subsequently on the computation of

$$F\big(\vartheta^{(n)}\big) = \omega + K\left(S_M^{(n)} \cos\big(\vartheta^{(n)}\big) - C_M^{(n)} \sin\big(\vartheta^{(n)}\big)\right). \tag{6b}$$

Altogether, this requires $4\,M$ evaluations of sine and cosine functions compared to $M(M-1)$ evaluations of the sine function needed for the original formulation (3). The generalisation to higher-order explicit or implicit time integration methods is

8

straightforward. Long-term simulations are ideally based on geometric integrators. Their benefits over standard methods are demonstrated in [3, 19, 34, 35], e.g.

Remark. In our study, the focus is on the numerical simulation of a high number of oscillators. In this situation, the reduction from M^2 to $M^2 - M$ evaluations of the sine function by taking into account the evident identity for coinciding indices

$$\ell = m: \quad \sin(\vartheta_\ell - \vartheta_m) = 0$$

is of minor relevance and thus will be neglected.

Numerical comparisons. Numerical comparisons of different approaches for the evaluation of the right-hand side of the classical Kuramoto model (3)–(4) and the associated potential (5) are displayed in Figure 1. We focus on an implementation in MATLAB and expect that analogous conclusions hold for other software packages. We vary the total number of oscillators from 10^2 to 10^8, taking into account maximum array size preferences as mentioned in the introduction. A randomly chosen number $\omega_0 \in (0, 1)$ defines the intrinsic frequencies. The phases $\vartheta_1, \vartheta_2, \ldots, \vartheta_M$ are uniformly distributed in $[0, 2\pi]$. We compare

(i) straightforward summation of $\sin(\vartheta_\ell - \vartheta_m)$ for $\ell, m \in \{1, 2, \ldots, M\}$ realised by a double loop over all rows,

(ii) a simple script using $\mathrm{sum}(\sin(\vartheta - \vartheta_m))$ for $m \in \{1, 2, \ldots, M\}$ and thus involving a loop over all rows,

(iii) a script by Cleve Moler that generates the matrix $\sin(\vartheta - \vartheta')$ and then adds up each row,[1]

(iv) the above stated procedure using the precomputation of sums, see (4) and (6).

For a higher number of oscillators, it is probable that the evaluation of functions and the computation of sums are the most time consuming components. The numerical results confirm that our approach (iv) is favourable in this regime and permits the efficient simulation of a high number of oscillators. Similar conclusions hold for the evaluation of the potential.

Numerical integration. The numerical integration of the classical Kuramoto model (3)–(4) based on the precomputation of sums is illustrated in Figures 2–6. Movies showing the time evolution are found at

http://techmath.uibk.ac.at/mecht/MyHomepage/Research/MovieKuramotoClassicalK1.m4v
http://techmath.uibk.ac.at/mecht/MyHomepage/Research/MovieKuramotoClassicalK3.m4v
http://techmath.uibk.ac.at/mecht/MyHomepage/Research/MovieKuramotoClassicalK5.m4v

We consider $M = 10^2$ as well as $M = 10^4$ oscillators and set $T = 200$. The interplay between the coupling constant $K \in \{1, 3, 5\}$ and the constant $\omega_0 = 2$ defining the intrinsic frequencies determines the strength of synchronisation. The points and the arrow reflect the values of the phases and the complex order parameter at the final time

$$\Big(\cos\big(\vartheta_m(t) - \psi(t)\big), \sin\big(\vartheta_m(t) - \psi(t)\big)\Big), \quad m \in \{1, 2, \ldots, M\}, \quad r\big(\vartheta(t)\big), \quad t = T,$$

---

[1] The MATLAB script kuramoto.m by Cleve Moler is available at
https://de.mathworks.com/matlabcentral/fileexchange/72534-kuramoto-s-model-of-synchronizing-oscillators.

see (5b). Moreover, the graphs of the associated quantities

$$V\big(\vartheta(t)\big)\,,\quad \tfrac{1}{M}\sum_{m=1}^{M}\big(\vartheta_m(t)-\vartheta_m(0)-t\,\omega_m\big)\,,\quad t\in[0,T]\,,$$

are shown, see (5a) and (5d). In addition, the computation time CT, measured in seconds, is displayed. For the considered time interval, a variable stepsize fourth-order explicit Runge–Kutta method leads to a reliable result. In particular, it is seen that the values of the potential decrease when time evolves and that the modulus of the complex order parameter is close to one for $K=5$. When applying instead a second-order geometric integrator, we observed an improved behaviour regarding the conserved quantity over long times, see Figure 6. For comprehensive information about the benefits of geometric numerical integrators in comparison with standard integration methods, we refer to [3, 19, 34, 35].

3. Kuramoto models on graphs

In this section, we investigate extensions of the classical Kuramoto model on graph topologies and propose suitable modifications of our approach for their efficient numerical integration.

General formulation. Henceforth, we study the following system of ordinary differential equations

$$\begin{cases} \vartheta'_m(t)=\omega_m+\tfrac{K}{\mathcal{M}_m}\sum_{\ell=1}^{M}A_{m\ell}\,\sin\big(\vartheta_\ell(t)-\vartheta_m(t)\big)\,,\\ \vartheta_m(0)\text{ given}\,,\quad t\in(0,T)\,,\quad m\in\{1,2,\ldots,M\}\,. \end{cases}\tag{7a}$$

In contrast to the classical Kuramoto model (3), the all-to-all coupling is replaced by the interactions of certain communities of oscillators, which are described by the associated adjacency matrix

$$A=\big(A_{m\ell}\big)_{\ell,m\in\{1,\ldots,M\}}\,,\quad A_{m\ell}\in\{0,1\}\,,\quad \ell,m\in\{1,2,\ldots,M\}\,.\tag{7b}$$

We distinguish two kinds of scalings affecting in particular the strength of synchronisation. In case the mth row of A is zero, the corresponding equation reduces to

$$\vartheta'_m(t)=\omega_m\,,\quad t\in(0,T)\,,$$

and hence the scaling is dispensible.

(i) *Uniform scaling.* With regard to the literature, a common choice is

$$\mathcal{M}_m=M\,,\quad m\in\{1,2,\ldots,M\}\,.\tag{7c}$$

10

(ii) *Non-uniform scaling.* As exemplified below, an alternative is

$$\mathcal{M}_m = \sum_{\ell=1}^{M} A_{m\ell}, \quad m \in \{1, 2, \ldots, M\}. \tag{7d}$$

For our considerations, it is convenient to introduce the scaled adjacency matrix

$$\mathcal{A} = \left(\tfrac{1}{\mathcal{M}_m} A_{m\ell}\right)_{\ell, m \in \{1, \ldots, M\}}. \tag{7e}$$

Auxiliary identities. The decisive terms defining the right-hand side of the extended Kuramoto model (7) rewrite as follows

$$\sum_{\ell=1}^{M} \mathcal{A}_{m\ell} \sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big)$$

$$= \sum_{\ell=1}^{M} \mathcal{A}_{m\ell} \sin\big(\vartheta_\ell(t)\big) \cos\big(\vartheta_m(t)\big)$$

$$- \sum_{\ell=1}^{M} \mathcal{A}_{m\ell} \cos\big(\vartheta_\ell(t)\big) \sin\big(\vartheta_m(t)\big), \quad t \in [0, T], \quad m \in \{1, 2, \ldots, M\}.$$

Denoting the componentwise product of two columns by

$$v .* w = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_M \end{pmatrix} .* \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_M \end{pmatrix} = \begin{pmatrix} v_1\, w_1 \\ v_2\, w_2 \\ \vdots \\ v_M\, w_M \end{pmatrix}, \quad v, w \in \mathbb{R}^M,$$

and employing the compact vector notation

$$G\big(\vartheta(t)\big) = \omega + K \left(\big(\mathcal{A} \sin\big(\vartheta(t)\big)\big) .* \cos\big(\vartheta(t)\big) - \big(\mathcal{A} \cos\big(\vartheta(t)\big)\big) .* \sin\big(\vartheta(t)\big) \right),$$

$$t \in [0, T], \tag{8a}$$

we obtain the following reformulation of the governing equations

$$\begin{cases} \vartheta'(t) = G\big(\vartheta(t)\big), \quad t \in (0, T), \\ \vartheta(0) \text{ given}, \end{cases} \tag{8b}$$

11

see also (4). Similar considerations yield the identity

$$\sum_{\ell,m=1}^{M} \mathcal{A}_{m\ell} \cos\big(\vartheta_\ell(t) - \vartheta_m(t)\big)$$

$$= \sum_{\ell,m=1}^{M} \mathcal{A}_{m\ell} \cos\big(\vartheta_\ell(t)\big) \cos\big(\vartheta_m(t)\big) + \sum_{\ell,m=1}^{M} \mathcal{A}_{m\ell} \sin\big(\vartheta_\ell(t)\big) \sin\big(\vartheta_m(t)\big)$$

$$= \big(\cos(\vartheta(t))\big)^T \big(\mathcal{A}\cos(\vartheta(t))\big) + \big(\sin(\vartheta(t))\big)^T \big(\mathcal{A}\sin(\vartheta(t))\big), \quad t \in [0, T].$$

Uniform scaling. In situations, where the adjacency matrix is symmetric and the scaling is uniform

$$A = A^T, \quad \mathcal{M}_m = M, \quad m \in \{1, 2, \ldots, M\}, \quad \mathcal{A} = \tfrac{1}{M}A, \quad \mathcal{A}^T = \mathcal{A},$$

the existence of a potential function is ensured and the sum over all phases leads to a conserved quantity, see also (5). With regard to the classical Kuramoto model and the above stated auxiliary identity, we define the potential through

$$
\begin{aligned}
V : \mathbb{R}^M &\longrightarrow \mathbb{R} : \\
\vartheta &\longmapsto V(\vartheta) = V(\vartheta_1, \ldots, \vartheta_M) \\
&= -\sum_{m=1}^{M} \omega_m \vartheta_m + \tfrac{K}{2} \sum_{\ell,m=1}^{M} \mathcal{A}_{m\ell} \big(1 - \cos(\vartheta_\ell - \vartheta_m)\big) \\
&= -\omega^T \vartheta + \tfrac{K}{2} \bigg(\sum_{\ell,m=1}^{M} \mathcal{A}_{m\ell} - \big(\cos(\vartheta)\big)^T \big(\mathcal{A}\cos(\vartheta)\big) \\
&\qquad\qquad\qquad - \big(\sin(\vartheta)\big)^T \big(\mathcal{A}\sin(\vartheta)\big) \bigg).
\end{aligned}
$$
(9a)

Due to the fact that the relation

$$\mathcal{A}_{m\ell} \sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big) + \mathcal{A}_{\ell m} \sin\big(\vartheta_m(t) - \vartheta_\ell(t)\big) = 0,$$
$$t \in [0, T], \quad \ell, m \in \{1, 2, \ldots, M\},$$

holds, summation and integration with respect to time implies

$$\tfrac{1}{M} \sum_{m=1}^{M} \big(\vartheta_m(t) - \vartheta_m(0) - t\,\omega_m\big) = 0, \quad t \in [0, T].$$
(9b)

Non-uniform scaling. In order to illustrate our motive for the non-uniform scaling, we consider the special case, where the adjacency matrix comprises a square submatrix

with coefficients equal to one and is zero otherwise

$$A = \begin{pmatrix} B^{(11)} & B^{(12)} \\ B^{(21)} & B^{(22)} \end{pmatrix} \in \mathbb{R}^{M \times M},$$

$$B^{(11)} = (1)_{\ell, m \in \{1, \ldots, M_0\}}, \quad B^{(12)} = B^{(21)} = B^{(22)} = 0 \in \mathbb{R}^{M_0 \times M_0},$$

$$M_0 \in \{2, \ldots, M-1\}.$$

This would correspond to a configuration with pairwise interaction of the first part of the oscillators and a decoupling of the second part

$$\begin{cases} \vartheta'_m(t) = \omega_m + \frac{K}{\mathcal{M}_m} \sum_{\ell=1}^{M_0} \sin(\vartheta_\ell(t) - \vartheta_m(t)), & t \in (0,T), \quad m \in \{1, 2, \ldots, M_0\}, \\ \vartheta'_m(t) = \omega_m, & t \in (0,T), \quad m \in \{M_0+1, M_0+2, \ldots, M\}. \end{cases}$$

Here, the non-uniform scaling seems to be more natural

$$\mathcal{M}_m = \sum_{\ell=1}^{M} A_{m\ell} = M_0 \neq M, \quad m \in \{1, 2, \ldots, M_0\}.$$

We point out that a potential function of the form (9) remains valid for non-symmetric scaled adjacency matrices, whereas summation over all governing equations does not lead to a conserved quantity, in general.

Extension of our approach. The extension of our approach for the classical Kuramoto model based on the precomputation of sums to Kuramoto models on graphs (7) requires a careful incorporation of the structure of the associated adjacency matrix. The following considerations prove to be particularly expedient for situations, where the adjacency matrix has a block structure and can be divided into relatively sparse and relatively dense submatrices, respectively. In view of an efficient evaluation of the decisive terms in (7), we consider a submatrix of the form

$$\left(A_{m\ell}\right)_{m \in \{M_1, \ldots, M_2\}, \, \ell \in \{M_3, \ldots, M_4\}},$$

defined by positive integers $M_1, M_2, M_3, M_4 \in \mathbb{N}$ such that $1 \leq M_1 < M_2 \leq M$ as well as $1 \leq M_3 < M_4 \leq M$. Our basic concept is to optimise the required memory capacity and the number of functions evaluations. More precisely, in order to avoid the storage of each submatrix and to accelerate the computation of the sums

$$\sum_{\ell=M_3}^{M_4} A_{m\ell} \sin(\vartheta_\ell(t) - \vartheta_m(t)), \quad m \in \{M_1, \ldots, M_2\},$$

we distinguish two complementary cases. Whenever the number of coefficients equal to zero is relatively high, we store all indices corresponding to non-zero coefficients and sum over these coefficients

$$\sum_{\substack{\ell \in \{M_3, \ldots, M_4\} \\ A_{m\ell}=1}} A_{m\ell} \sin(\vartheta_\ell(t) - \vartheta_m(t)), \quad m \in \{M_1, \ldots, M_2\}. \tag{10a}$$

Whenever the number of non-zero coefficients is relatively high, we instead store all indices corresponding to coefficients equal to zero and make use of the precomputation of sums. That is, where applicable, we employ the reformulation

$$\sum_{\ell=M_3}^{M_4} A_{m\ell} \sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big)$$

$$= \sum_{\ell=M_3}^{M_4} A_{m\ell} \sin\big(\vartheta_\ell(t)\big) \cos\big(\vartheta_m(t)\big) - \sum_{\ell=M_3}^{M_4} A_{m\ell} \cos\big(\vartheta_\ell(t)\big) \sin\big(\vartheta_m(t)\big)$$

$$= \left(\sum_{\ell=M_3}^{M_4} \sin\big(\vartheta_\ell(t)\big) - \sum_{\ell=M_3}^{M_4} \big(1 - A_{m\ell}\big) \sin\big(\vartheta_\ell(t)\big) \right) \cos\big(\vartheta_m(t)\big)$$

$$- \left(\sum_{\ell=M_3}^{M_4} \cos\big(\vartheta_\ell(t)\big) - \sum_{\ell=M_3}^{M_4} \big(1 - A_{m\ell}\big) \cos\big(\vartheta_\ell(t)\big) \right) \sin\big(\vartheta_m(t)\big) ,$$

precompute the sums

$$S_{M_3,M_4}\big(\vartheta(t)\big) = \sum_{\ell=M_3}^{M_4} \sin\big(\vartheta_\ell(t)\big) , \quad C_{M_3,M_4}\big(\vartheta(t)\big) = \sum_{\ell=M_3}^{M_4} \cos\big(\vartheta_\ell(t)\big) , \tag{10b}$$

and then substract the terms that correspond to coefficients equal to zero

$$\sum_{\ell=M_3}^{M_4} A_{m\ell} \sin\big(\vartheta_\ell(t) - \vartheta_m(t)\big)$$

$$= \left(S_{M_3,M_4}\big(\vartheta(t)\big) - \sum_{\substack{\ell\in\{M_3,\ldots,M_4\} \\ A_{m\ell}=0}} \sin\big(\vartheta_\ell(t)\big) \right) \cos\big(\vartheta_m(t)\big)$$

$$- \left(C_{M_3,M_4}\big(\vartheta(t)\big) \right.$$

$$\left. - \sum_{\substack{\ell\in\{M_3,\ldots,M_4\} \\ A_{m\ell}=0}} \cos\big(\vartheta_\ell(t)\big) \right) \sin\big(\vartheta_m(t)\big) , \quad m \in \{M_1, \ldots, M_2\} . \tag{10c}$$

Remark. In situations, where it is desirable to evaluate the right-hand side and the potential in parallel, it is advantageous to modify the above stated approach in view of an efficient computation of

$$\sum_{\ell=M_3}^{M_4} A_{m\ell} \sin\big(\vartheta_\ell(t)\big) = \sum_{\ell=M_3}^{M_4} \sin\big(\vartheta_\ell(t)\big) - \sum_{\ell=M_3}^{M_4} \big(1 - A_{m\ell}\big) \sin\big(\vartheta_\ell(t)\big)$$

$$\sum_{\ell=M_3}^{M_4} A_{m\ell} \cos\big(\vartheta_\ell(t)\big) = \sum_{\ell=M_3}^{M_4} \cos\big(\vartheta_\ell(t)\big) - \sum_{\ell=M_3}^{M_4} \big(1 - A_{m\ell}\big) \cos\big(\vartheta_\ell(t)\big) ,$$

$$m \in \{M_1, \ldots, M_2\} ,$$

either by straightforward summation or based on the precomputation of sums, see also (8)–(9).

Numerical comparisons. In order to test the above described strategy (10) for the efficient computation of the sums

$$\sum_{\ell=1}^{M} A_{m\ell} \sin(\vartheta_\ell - \vartheta_m) \,, \quad \vartheta_m = \tfrac{2\pi m}{M} \,, \quad m \in \{1, 2, \ldots, M\} \,,$$

we study two situations, which we consider to be of practical relevance in view of the numerical integration of Kuramoto models on graphs (3), see also (7).

(i) *Matrices without block structure.* On the one hand, we define well-balanced adjacency matrices and compare the number of function evaluations as well as the computation time of an implementation in MATLAB. For this purpose, we prescribe a threshold $p \in [0, 1]$ and generate row-by-row a sequence of uniformly distributed random numbers $z \in [0, 1]$. Whenever $z > p$, the corresponding coefficient is set to one, otherwise, it is set to zero. For $p = 0.99$, e.g., the resulting adjacency matrix is sparse and straightforward summation is advantageous, see Figure 7. For the complementary case $p = 0.01$, e.g., the adjacency matrix comprises few non-zero coefficients and the precomputation of sums is favourable, see Figure 9. For the threshold $p = 0.5$ both approaches lead to essentially the same counts, see Figure 8.

(ii) *Matrices with block structure.* On the other hand, we generate adjacency matrices that are composed of well-balanced submatrices. Each block is connected with a different threshold. We use
   (a) a double loop over all rows and straightforward summation of $\sin(\vartheta_\ell - \vartheta_m)$ provided that $A_{m\ell} \neq 0$ for $\ell, m \in \{1, 2, \ldots, M\}$,
   (b) a single loop to sum over all indices that correspond to non-zero coefficients, realised by a script of the form $\mathrm{sum}(\sin(\vartheta(\mathrm{NonZero}) - \vartheta_m))$ for $m \in \{1, 2, \ldots, M\}$,
   (c) the precomputation of sums and a single loop to subtract terms that correspond to coefficients equal to zero, applying the script $\mathrm{sum}(\sin(\vartheta(\mathrm{Zero}) - \vartheta_m))$ for $m \in \{1, 2, \ldots, M\}$,
   (d) the procedure in (b) and (c) adapted to each submatrix.
   The obtained results, displayed in Figures 10–12, confirm that the latter approach is beneficial for a higher number of oscillators, where the evaluation of functions and the computation of sums are expected to be the most time consuming components.

Community detection in graphs. Our numerical comparisons show that it is to the best advantage to take the underlying structure of the considered Kuramoto model on a graph (7) into account. A suitable reordering of the governing equations accordingly to the separation of the oscillators into communities corresponds to the transformation of the associated adjacency matrix to a well-balanced block matrix and is a fundamental means in view of their efficient numerical integration, see Figures 13–14. In the following, we study the performance of various algorithms for the detection of communities in graphs. We restrict ourselves to algorithms from the PYTHON packages NETWORKX and CDLIB, which were suitable for our purposes, did not require additional parameters, and took a reasonable computation time, see Table 2. We focus on the relevant case, where interactions primarily take place within certain communities of

15

oscillators. As illustrated in Figure 15, our starting point is a matrix (of suitably chosen dimension) with fully occupied blocks along the diagonal

$$B = (B_{m\ell})_{m,\ell\in\{1,2,\ldots,M\}} = \begin{pmatrix} B^{(11)} & & & 0 \\ & B^{(22)} & & \\ & & B^{(33)} & \\ 0 & & & B^{(44)} \end{pmatrix} \in \mathbb{R}^{M\times M}\,,$$

$$B^{(kk)} = (1)_{i,j\in\{1,2,\ldots,(5-k)M/10\}} \in \mathbb{R}^{(5-k)M/10\times(5-k)M/10}\,, \quad k \in \{1,2,3,4\}\,.$$

Similarly to before, we prescribe a threshold $p \in [0,1]$ that determines the probability that a coefficient of the matrix B is changed from one to zero or from zero to one, respectively. That is, we generate uniformly distributed random numbers $z_{m\ell} \in [0,1]$ for $m, \ell \in \{1, 2, \ldots, M\}$ and define

$$A = (A_{m\ell})_{m,\ell\in\{1,2,\ldots,M\}} \in \mathbb{R}^{M\times M}\,,$$

$$A_{m\ell} = \begin{cases} 1 - B_{m\ell}\,, & z_{m\ell} < p\,, \\ B_{m\ell}\,, & z_{m\ell} \geq p\,, \end{cases} \quad m, \ell \in \{1, 2, \ldots, M\}\,.$$

The higher the threshold, the stronger the deviation of the adjacency matrix from the related block matrix, see Figures 15–16. The community detection algorithms were applied to the randomly permuted adjacency matrix and yield as outputs partitions of the nodes into communities. We associate them with permutations of the sequence $(1, 2, \ldots, M)$ such that nodes in one community are arranged one after the other. These permutations correspond to transformations of the original matrices that can be cast into the form

$$\widetilde{A} = P A P^T$$

with a permutation matrix $P \in \mathbb{R}^{M\times M}$ such that in particular the identity $P^{-1} = P^T$ is valid. For each of these matrices, we identify a matrix with fully occupied blocks along the diagonal reflecting the detected communities

$$\widetilde{B} = \begin{pmatrix} \widetilde{B}^{(11)} & & 0 \\ & \widetilde{B}^{(22)} & \\ 0 & & \ddots \end{pmatrix} \in \mathbb{R}^{M\times M}\,.$$

We recall that the computation of pointwise products such as

$$\left(B \sin(\vartheta)\right).* \cos(\vartheta)\,, \quad \left(\widetilde{B} \sin(\vartheta)\right).* \cos(\vartheta)\,, \quad \vartheta \in \mathbb{R}^M\,,$$

based on the precomputation of sums requires in total $2M$ evaluations of cosine and sine functions and that the number of non-zero coefficients of the matrices

$$C = A - B\,, \quad \widetilde{C} = \widetilde{A} - \widetilde{B}\,,$$

16

Table 2. Algorithms from PYTHON packages applied for community detection in graphs.

NETWORKX [17]	GREEDY_MODULARITY [7]
CDLIB [33]	LOUVAIN [5]
	RBER_POTS [30, 31]
	RB_POTS [24, 31]
	SIGNIFICANCE_COMMUNITIES [39]
	WALKTRAP [27]

match the additional computational costs. For this reason, we determine the quantity

$$\sum_{\ell,m=1}^{M} \left| \widetilde{C}_{m\ell} \right| - \sum_{\ell,m=1}^{M} \left| C_{m\ell} \right|, \tag{11}$$

in order to assess the performance of the algorithms, see [29] for detailed explanations. The numerical results obtained for $M \in \{100, 200, 400, 800, 1600\}$ and thresholds in the range $[0, 0.4]$ are displayed in Figures 15–17. In case $p = 0$, i.e. for a random permutation of the fully occupied block diagonal matrix, each of the tested algorithms detected the four communities. For larger deviations of the adjacency matrices from the underlying block diagonal matrix, however, some algorithms failed. Overall, the community detection algorithm RBER_POTS, available through the PYTHON package CDLIB, provided the most reliable results for a higher number of oscillators.

Favourable community detection algorithm. It is notable that there exist several variants of the algorithm RBER_POTS [30, 31] with the common objective to detect communities in a graph with associated adjacency matrix A. The standard implementation in CDLIB [33] minimises the quantity

$$Q = - \sum_{\ell,m=1}^{M} \left(A_{m\ell} - p \right) \delta(\sigma_m, \sigma_\ell),$$

where $p \in (0, 1)$ represents the mean density of edges in a graph, that is, the ratio between the numbers of actually existing and potential edges. Whenever two nodes $m, \ell \in \{1, 2, \ldots, M\}$ belong to the same community, the quantity $\delta(\sigma_m, \sigma_\ell)$ takes the value one, and it is zero otherwise. Provided that the mean edge density within a community is higher than the mean edge density of the complete network, a node is included in a community. The favourable performance of RBER_POTS in the considered example is explained by the fact that the four communities are well recognisable by the mean edge densities, even for $p = 0.4$.

Numerical integration. Following up the numerical integration of the classical Kuramoto model (3)–(4), we finally study extended Kuramoto models on graphs (7). On the one hand, we consider the situation, where a separation of the oscillators into four communities of the same size is evident. The structure of the associated symmetric adjacency matrix is illustrated in Figure 14 (right). Here, our approach based on the precomputation of sums applies. The numerical results obtained for the common uniform scaling, a total number of $M = 8^4 = 4096$ oscillators, coupling constant $K = 3$, final time $T = 200$, and intrinsic frequencies as well as initial phases

of the form

$$\omega_0 = 2\,, \quad \omega_m = 1 + \omega_0 \, \tfrac{(2m-M-1)}{M-1}\,, \quad \vartheta_m(0) = \tfrac{2\pi m}{M}\,, \quad m \in \{1, 2, \dots, M\}\,,$$

are displayed in Figure 18 (right). On the other hand, we consider the equivalent system without recognisable block structure, see Figure 14 (left), using straightforward summation for the evaluation of the right-hand side. In order to achieve consistency with the previous case, we permute internal frequencies and initial phases accordingly, perform the time integration, and reorder the solution values subsequently. By comparison of the numerical results shown in Figure 18, it is evident that the efficient evaluation of the decisive sums based on the block structure of the adjacency matrix is beneficial and permits a significant reduction of the computation time from approximately 890 seconds to about 54 seconds. For the sake of comparability with the classical Kuramoto, we display the values of the corresponding potentials, which decrease when time evolves, and verify the conservation property, see also (9). The analogous results for the non-uniform scaling are given in Figure 19. As expected, due to the lack of symmetry, the constraint (9b) is not fulfilled.

4. Conclusions and outlook

In summary, we have presented results which confirm that the combination of detecting communities and using localisations of order parameters permits significant improvements regarding the reliable and efficient numerical integration of dynamical systems on graphs. Our approach provides a natural way to exploit the underlying graph structure and local mean-field variables.

Popular models, where the employed concepts apply at once, are given by consensus problems [25]. However, as these systems are linear with known analytical solution representations, we have considered their numerical simulation to be of less importance and have focused on Kuramoto models as relevant instances. Generalisations to other nonlinear dynamical systems on graphs occuring in applications will be the objective of future investigations.

As mentioned in the introduction, it is straightforward to extend our approach to oscillator networks with higher-order interactions included [4, 36]. Suitable modifications have to be developed for Cucker–Smale models describing flocking behaviour [8]. Other types of models, commonly encountered in neuroscience applications, incorporate individual excitable oscillators beyond a phase reduction [20, 22]. A further example is the strategy of combining our approach with micro-macro numerical integration schemes such as projective integration methods [21], where one makes use in the time integration of a macroscopic evolution, e.g., for the order parameter or the probability density of a typical oscillator, in combination with direct microscopic simulation to improve the numerical simulation. In such a context, our improvements directly accelerate the microscopic integrator. Yet, some natural-looking steps are bound to be far more involved. For example, for time-dependent or adaptive network dynamics [15, 18], the recomputation of community structures at every iteration is computationally inefficient and requires novel perspectives. Besides, providing a rigorous numerical analysis to assess the quality of community detection algorithms remains an open question. Furthermore, it is of relevance to study dynamical systems on graphs that incorporate stochastic perturbations.

References

[1] J. A. ACEBRÓN, L. L. BONILLA, C. J. P. VICENTE, F. RITORT, R. SPIGLER. *The Kuramoto model: A simple paradigm for synchronization phenomena.* Rev. Mod. Phys. 77/1 (2005) 137–185.

[2] A. ARENAS, A. DIAZ-GUILERA, J. KURTHS, Y. MORENO, C. ZHOU. *Synchronization in complex networks.* Phys. Rep. 469/3 (2008) 93–153.

[3] S. BLANES, F. CASAS. *A Concise Introduction to Geometric Numerical Integration.* CRC Press, 2016.

[4] C. BICK, T. BÖHLE, C. KUEHN. *Multi-population phase oscillator networks with higher-order interactions.* arXiv:2012.04943 (2020) 1–30.

[5] V. D. BLONDEL, J.-L. GUILLAUME, R. LAMBIOTTE, E. LEFEBVRE. *Fast unfolding of communities in large networks.* Journal of Statistical Mechanics: Theory and Experiment 10 (2008) P10008.

[6] H. CHIBA. *A proof of the Kuramoto conjecture for a bifurcation structure of the infinite-dimensional Kuramoto model.* Ergod. Theory Dyn. Syst. 35/3 (2013) 762–834.

[7] A. CLAUSET, M. E. J. NEWMAN, C. MOORE. *Finding community structure in very large networks.* Physical Review E 70/6 (2004) 066111.

[8] F. CUCKER, S. SMALE. *Emergent Behavior in Flocks.* IEEE Transactions on Automatic Control 52/5 (2007) 852–862.

[9] H. DAIDO. *Onset of cooperative entrainment in limit-cycle oscillators with uniform all-to-all interactions: Bifurcation of the order function.* Physica D: Nonlinear Phenomena 91/1-2 (1996) 24–66.

[10] H. DIETERT. *Stability and bifurcation for the Kuramoto model.* J. Math. Pures Appl. 105/4 (2016) 451–489.

[11] F. DÖRFLER, F. BULLO. *Synchronization in complex networks of phase oscillators: A survey.* Automatica 50/6 (2014) 1539–1564.

[12] B. FERNANDEZ, D. GÉRARD-VARET, G. GIACOMIN. *Landau damping in the Kuramoto model.* Ann. Henri Poincaré 17/7 (2016) 1793–1823.

[13] M. GIRVAN, M. E. NEWMAN. *Community structure in social and biological networks.* Proc. Nat. Acad. Sci. USA 99/12 (2002) 7821–7826.

[14] D. GOLOMB, D. HANSEL, G. MATO. *Mechanisms of synchrony of neural activity in large networks.* Handbook of Biological Physics 4 (2001) 887–968.

[15] T. GROSS, H. SAYAMA. *Adaptive Networks: Theory, Models and Applications.* Springer, 2009.

[16] S. Y. HA, H. K. KIM, S. W. RYOO. *Emergence of phase-locked states for the Kuramoto model in a large coupling regime.* Commun. Math. Sci. 14/4 (2016) 1073–1091.

[17] A. A. HAGBERG, D. A. SCHULT, P. J. SWART. *Exploring network structure, dynamics, and function using NetworkX.* 7th Python in Science Conference (2008) 11–15.

[18] P. HOLME, J. SARAMÄKI. *Temporal networks.* Phys. Rep. 519/3 (2011) 97–125.

[19] A. ISERLES, G. R. W. QUISPEL. *Why Geometric Numerical Integration?* In: K. Ebrahimi-Fard, M. Barbero Liñán (editors). Springer Proceedings in Mathematics & Statistics 267. Discrete Mechanics, Geometric Integration and Lie-Butcher Series, 1–28. Springer, 2018.

[20] E. M. IZHIKEVICH. *Phase equations for relaxation oscillators.* SIAM J. Appl. Math. 60/5 (2000) 1789–1804.

[21] I. G. KEVREKIDIS, C. W. GEAR, J. M. HYMAN, P. G. KEVREKIDIS, O. RUN-

BORG, C. THEODOROPOULOS. *Equation-free, coarse-grained multiscale computation: enabling microscopic simulators to perform system-level analysis.* Comm. Math. Sci. 1/4 (2003) 715–762.

[22] N. KOPELL, D. SOMERS. *Anti-phase solutions in relaxation oscillators coupled through excitatory interactions.* J. Math. Biol. 33/3 (1995) 261–280.

[23] Y. KURAMOTO. *Chemical Oscillations, Waves, and Turbulence.* Springer Series in Synergetics 19. Springer, 1984.

[24] E. A. LEICHT, M. E. J. NEWMAN. *Community Structure in Directed Networks.* Physical Review Letters 100/11 (2008) 118703.

[25] R. OLFATI-SABER, R. M. MURRAY. *Consensus problems in networks of agents with switching topology and time-delays.* IEEE Trans. Aut. Control 49/9 (2004) 1520–1533.

[26] A. S. PIKOVSKY, M. ROSENBLUM, J. KURTHS. *Synchronization.* CUP, 2001.

[27] P. PONS, M. LATAPY. *Computing Communities in Large Networks Using Random Walks.* In: P. Yolum, T. Güngör, F. Gürgen, C. Özturan (editors). Computer and Information Sciences, 284–293. Springer, 2005.

[28] M. A. PORTER, J. P. GLEESON. *Dynamical Systems on Networks: A Tutorial.* Frontiers in Applied Dynamical Systems: Reviews and Tutorials. Springer, 2016.

[29] M. A. PORTER, J. P. ONNELA, P. J. MUCHA. *Communities in networks.* Notices Amer. Math. Soc. 56/9 (2009) 1082–1097.

[30] J. REICHARDT, S. BORNHOLDT. *Detecting fuzzy community structures in complex networks with a potts model.* Physical Review Letters 93/21 (2004) 19–22.

[31] J. REICHARDT, S. BORNHOLDT. *Statistical mechanics of community detection.* Physical Review E 74/1 (2006) 016110.

[32] F. A. RODRIGUES, T. K. D. PERON, P. JI, J. KURTHS. *The Kuramoto model in complex networks.* Phys. Rep. 610 (2016) 1–98.

[33] G. ROSSETTI, L. MILLI, R. CAZABET. *CDLIB: a python library to extract, compare and evaluate communities from complex networks.* Applied Network Science 4/1 (2019).

[34] J. SANZ-SERNA. *Runge-Kutta schemes for Hamiltonian systems.* BIT 28/4 (1988) 877–883.

[35] J. SANZ-SERNA, M. CALVO. *Numerical Hamiltonian Problems.* Chapman & Hall, 1994.

[36] P. S. SKARDAL, A. ARENAS. *Higher-order interactions in complex networks of phase oscillators promote abrupt synchronization switching.* Common. Phys. 3/1 (2020) 1–6.

[37] S. H. STROGATZ. *From Kuramoto to Crawford: Exploring the onset of synchronization in populations of coupled oscillators.* Physica D: Nonlinear Phenomena 143/1-4 (2000) 1–20.

[38] S. H. STROGATZ, R. E. MIROLLO. *Stability of incoherence in a population of coupled oscillators.* Journal of Statistical Physics 63/3-4 (1991) 613–635.

[39] V. A. TRAAG, G. KRINGS, P. VAN DOOREN. *Significant Scales in Community Structure.* Scientific Reports 3/1 (2013) 2930.

[40] A. T. WINFREE. *Biological Rhythms and the Behavior of Populations of Coupled Oscillators.* Journal of Theoretical Biology 16/1 (1967) 15–42.
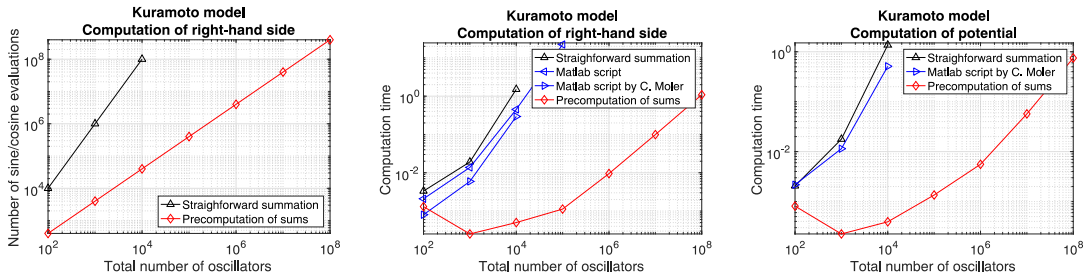
Figure 1. Classical Kuramoto model. Computational cost for the evaluation of the right-hand side and the potential. Left: Number of sine and cosine evaluations versus the total number of oscillators when using straighforward summation and the precomputation of sums, respectively. Middle: Numerical comparison of the computation time for different implementations in MATLAB. Right: Corresponding results for the potential.
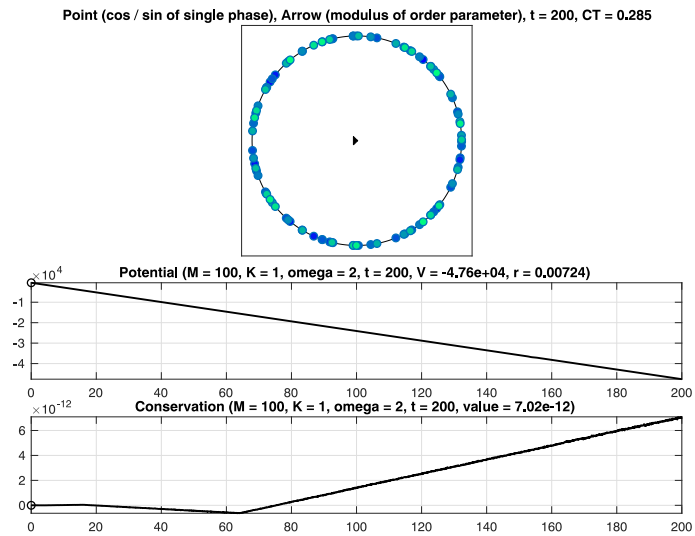

Figure 2. Numerical integration of the classical Kuramoto model involving $M = 10^2$ oscillators. Coupling constant $K = 1$ (no synchronisation). Top: Visualisation of the phases at the final time. Middle: The time series confirms decreasing potential values. Bottom: A conserved quantity is numerically preserved with high accuracy.

21
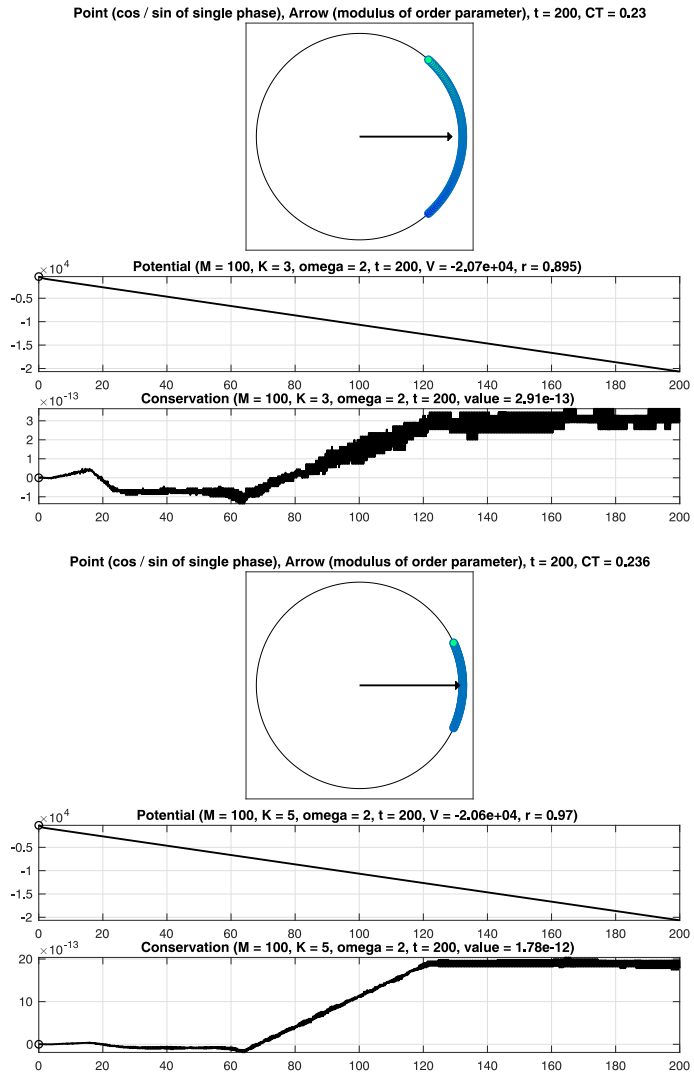
Figure 3. Numerical integration of the classical Kuramoto model involving $M = 10^2$ oscillators. Coupling constant $K \in \{3, 5\}$ (gradual synchronisation).
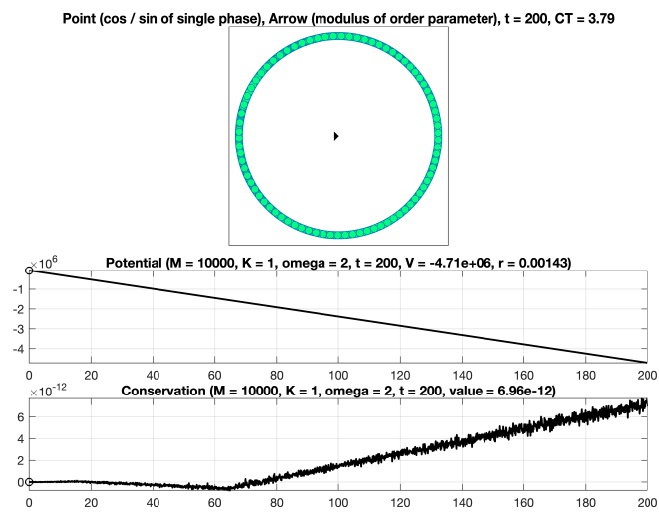
Figure 4. Numerical integration of the classical Kuramoto model involving $M = 10^4$ oscillators. Coupling constant $K = 1$ (no synchronisation).

Point (cos / sin of single phase), Arrow (modulus of order parameter), t = 200, CT = 3.69

Potential (M = 10000, K = 3, omega = 2, t = 200, V = -2.07e+06, r = 0.899)

Conservation (M = 10000, K = 3, omega = 2, t = 200, value = 7.68e-13)

Point (cos / sin of single phase), Arrow (modulus of order parameter), t = 200, CT = 3.82

Potential (M = 10000, K = 5, omega = 2, t = 200, V = -2.06e+06, r = 0.971)

Conservation (M = 10000, K = 5, omega = 2, t = 200, value = 2.65e-12)
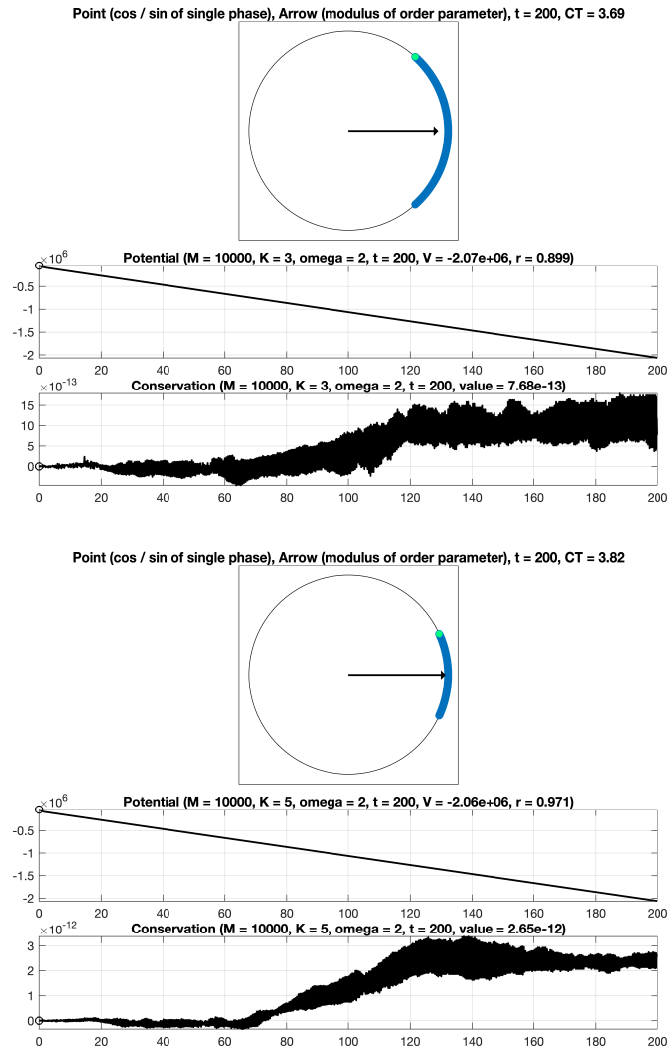
Figure 5. Numerical integration of the classical Kuramoto model involving $M = 10^4$ oscillators. Coupling constant $K \in \{3, 5\}$ (gradual synchronisation).
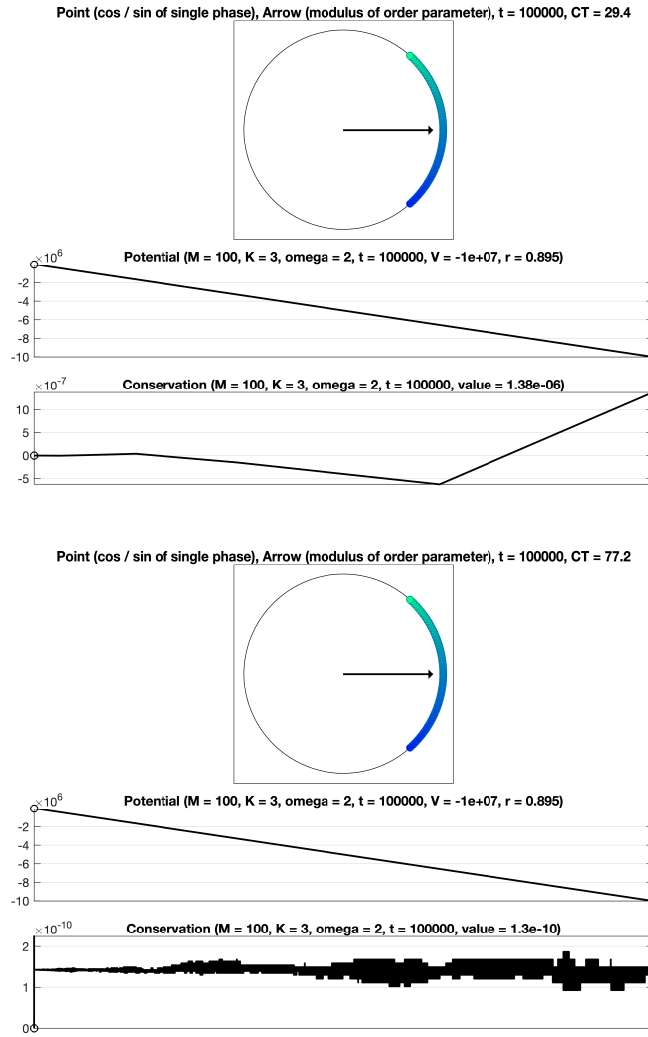
Figure 6. Long-term integration of the classical Kuramoto model based on a second-order explicit Runge–Kutta method (top) and a second-order implicit Runge–Kutta method with improved numerical preservation of a conserved quantity (bottom). To avoid a further diminishment of the relevant vertical axis, ticks along the horizontal axis are omitted.

25
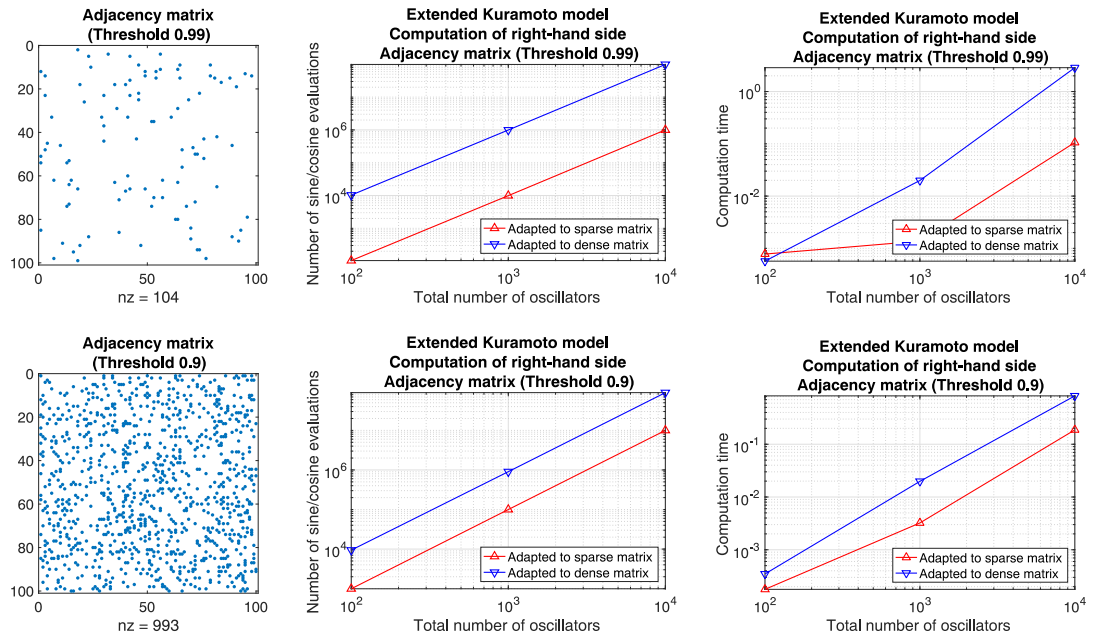
Figure 7. Extended Kuramoto models involving randomly generated adjacency matrices defined through the thresholds $0.99, 0.9$. Evaluation of the right-hand side by means of approaches adapted to sparse matrices (straightforward summation) and dense matrices (precomputation of sums), respectively. Left: Illustration of the adjacency matrix for 100 oscillators. Middle: Number of sine and cosine evaluations versus the total number of oscillators. Right: Comparison of the computation time.
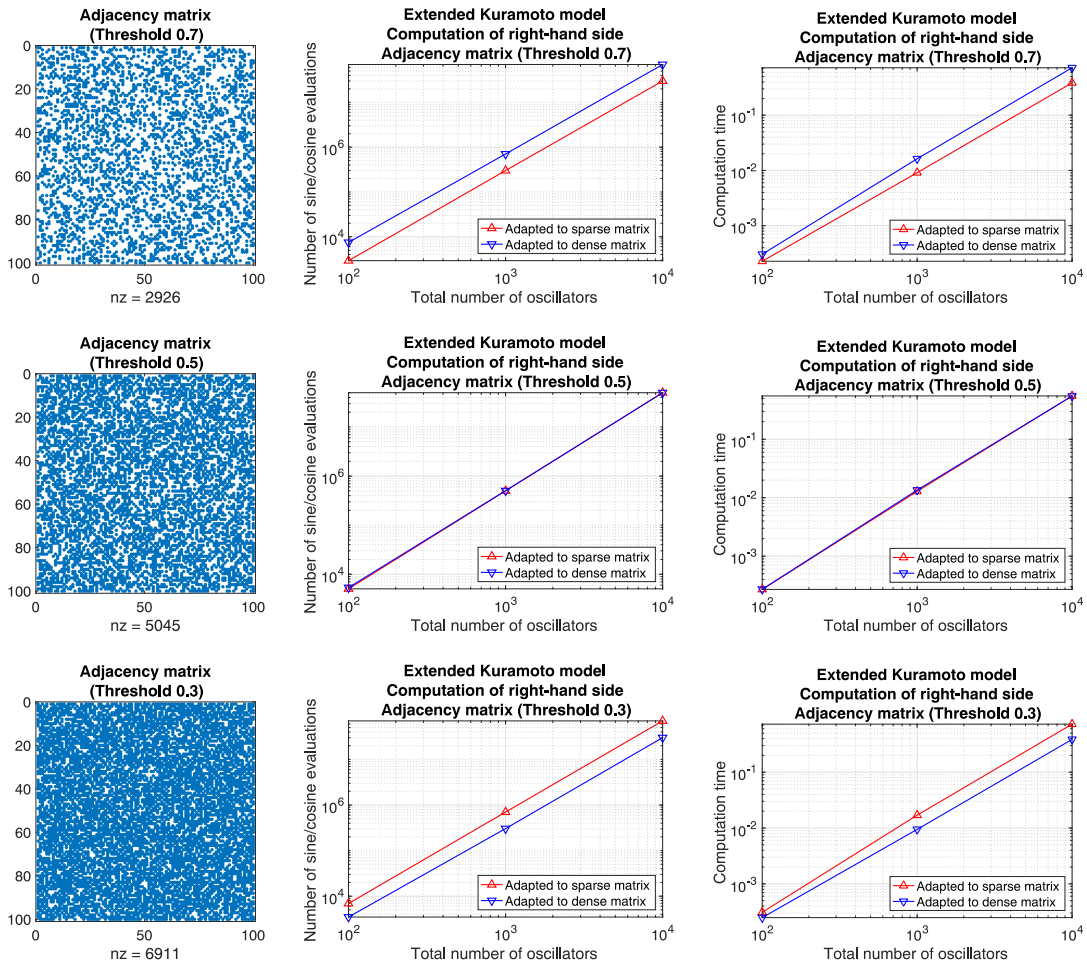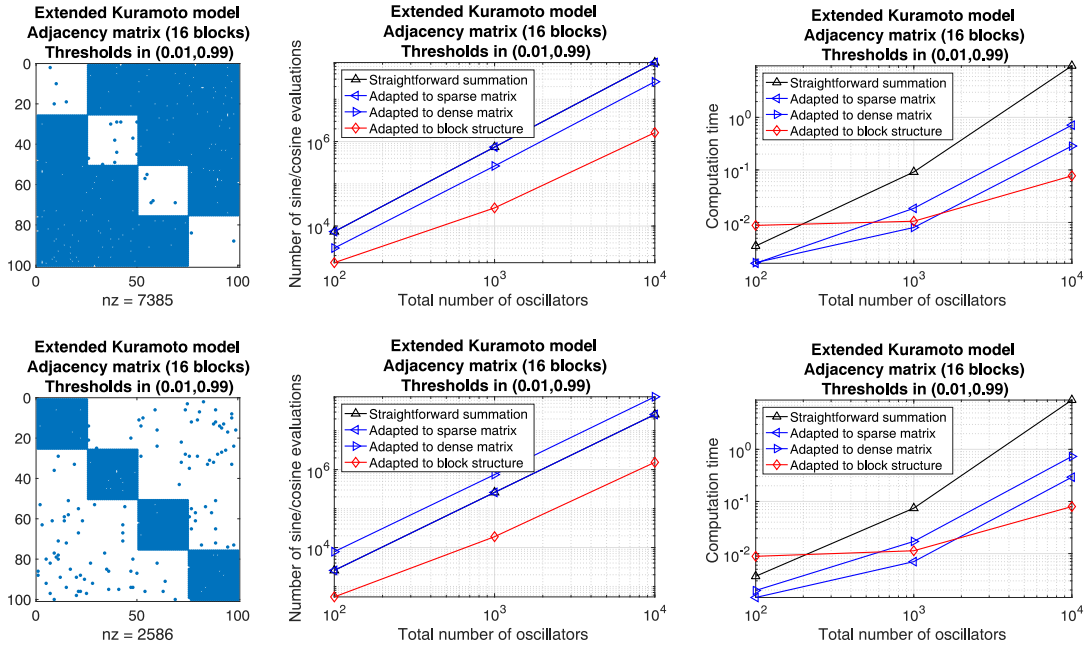
Figure 8. Corresponding results for the thresholds $0.7, 0.5, 0.3$.
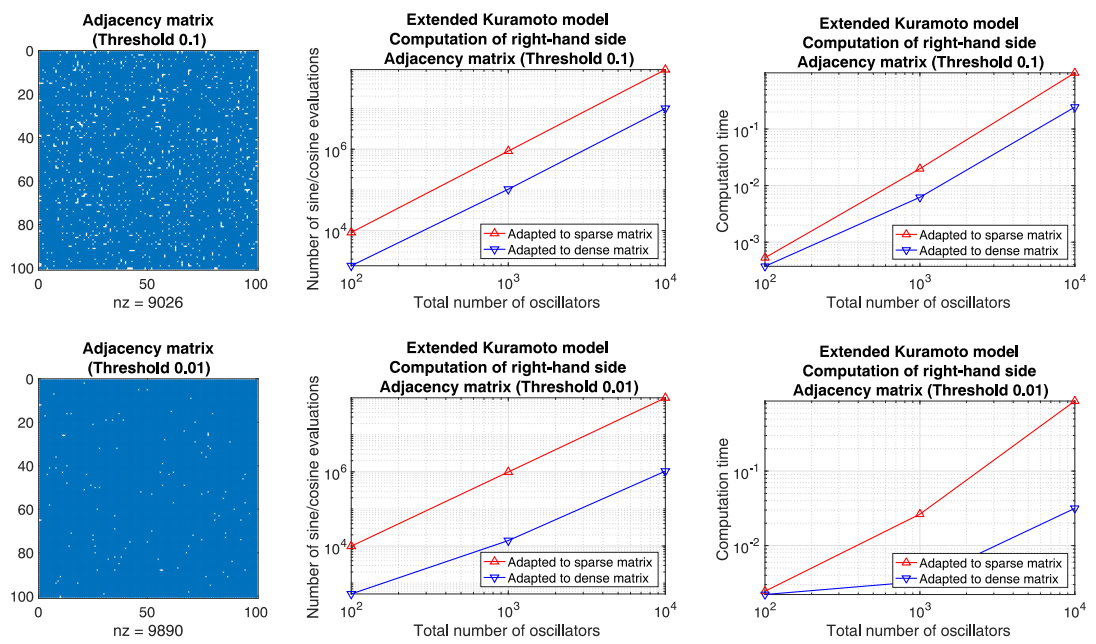
Figure 9. Corresponding results for the thresholds $0.1, 0.01$.
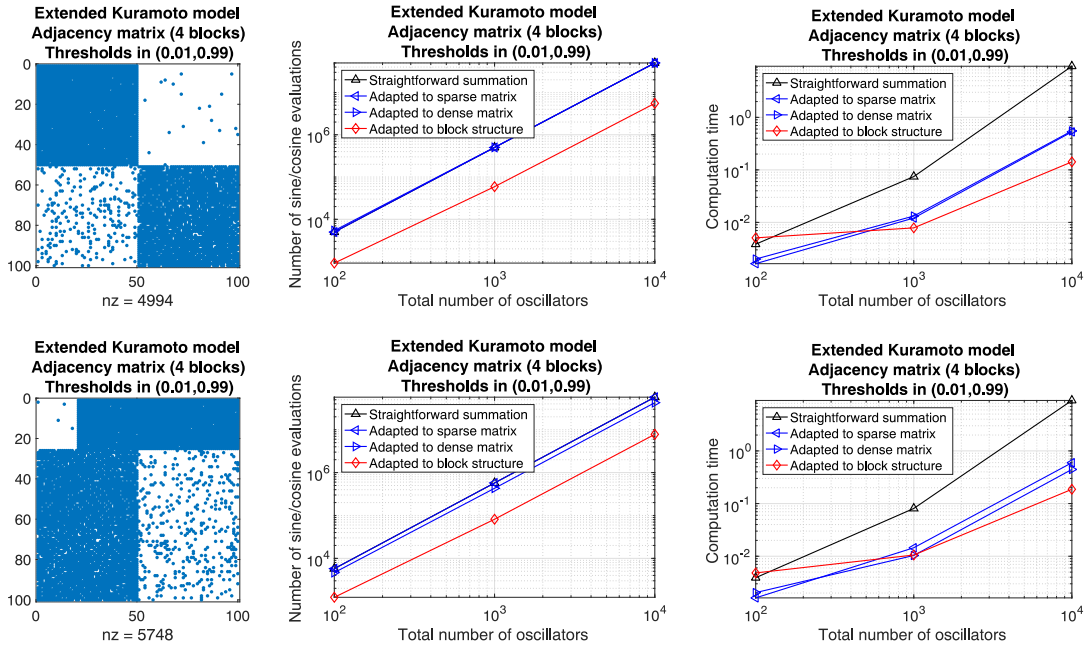
Figure 10. Extended Kuramoto models involving randomly generated adjacency matrices with block structure defined through a threshold per block. Evaluation of the right-hand side by means of different approaches adapted to sparse matrices, dense matrices, and block matrices, respectively. Left: Illustration of the adjacency matrix for 100 oscillators. Middle: Number of sine and cosine evaluations versus the total number of oscillators. Right: Comparison of the computation time.

Figure 11. Corresponding results.
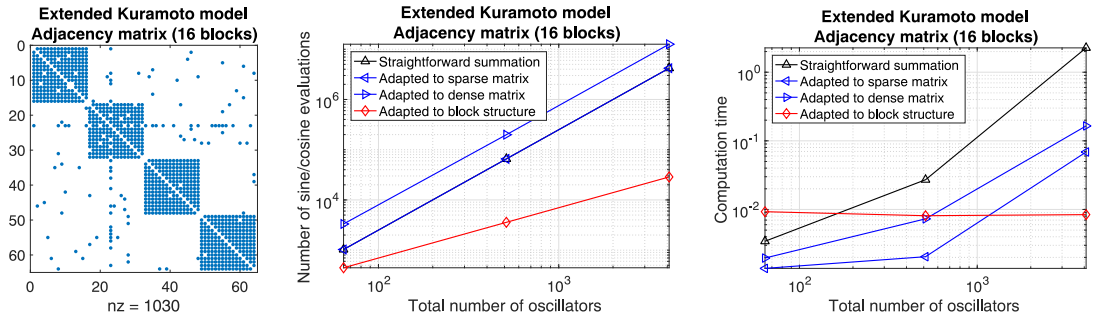
Figure 12. Corresponding results for a more realistic adjacency matrix describing the interactions of four communities of oscillators.
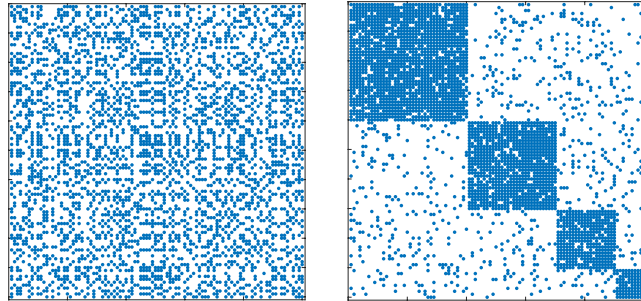
Figure 13. Adjacency matrices A and $P\,A\,P^T$ with suitably chosen permutation matrix P. Left: The structure of the underlying graph is not evident. Right: A separation into communities of oscillators is recognisable. Matrices of this form are used for numerical tests of community detection algorithms.
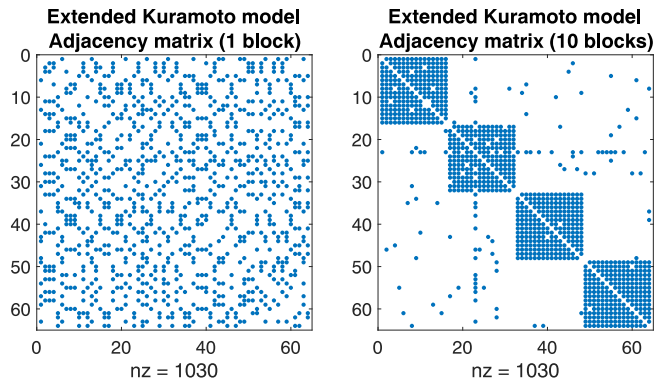

Figure 14. Adjacency matrices A and $P\,A\,P^T$ with suitably chosen permutation matrix P. Left: The structure of the underlying graph is not evident. Right: A separation into communities of oscillators is recognisable. Symmetric adjacency matrices of this form are considered in connection with the numerical integration of Kuramoto models on graphs. See also Figure 12.
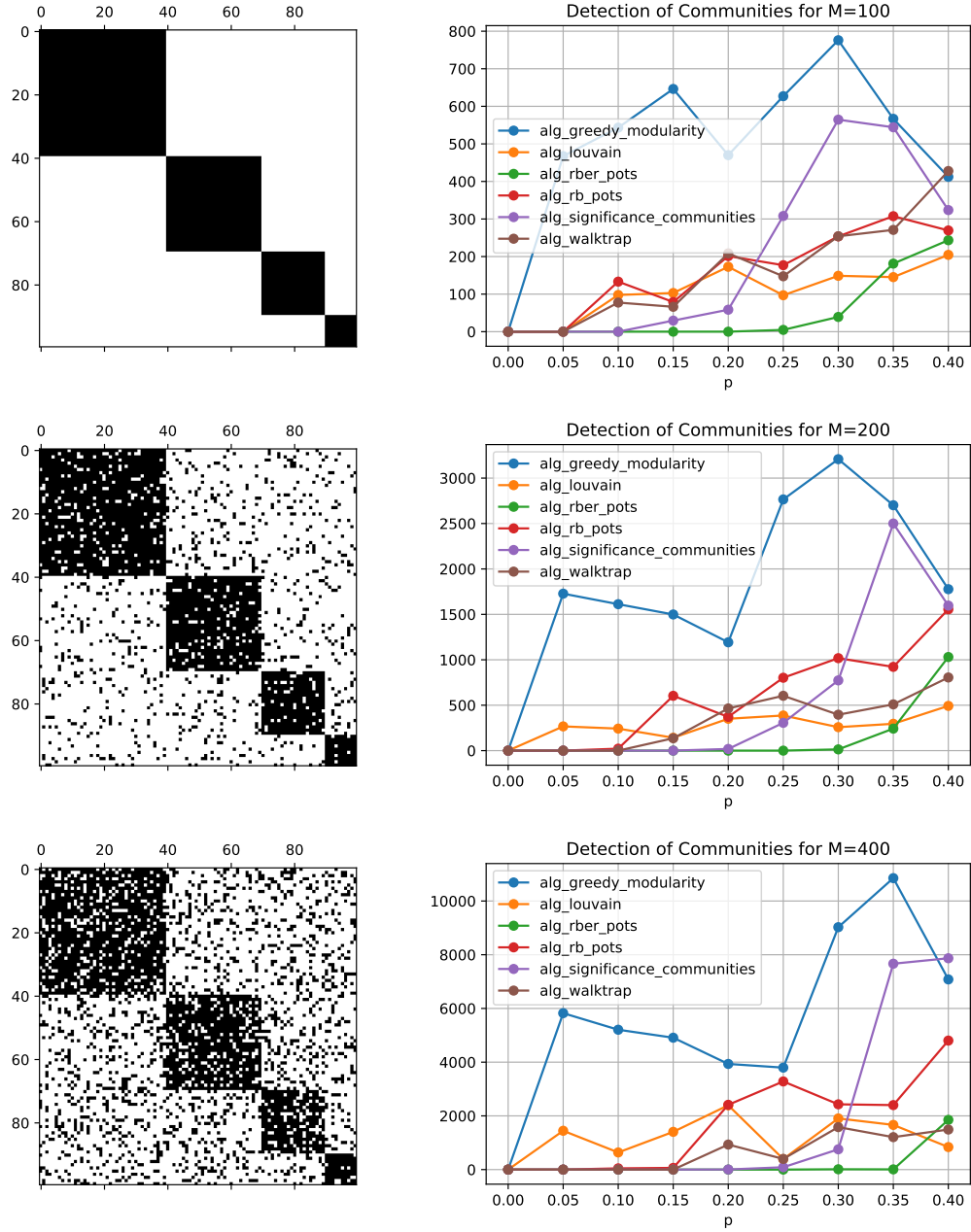
30

Figure 15. Left: Matrix comprising four blocks ($p = 0$, top) and related adjacency matrices for increasing thresholds $p \in \{0.1, 0.2\}$ (middle to bottom). Right: Average of the quantity (11) over eight runs, which is chosen as a measure for the performance of the considered community detection algorithms, see also Table 2. Obtained results for $M = 100$ (top), $M = 200$ (middle), $M = 400$ (bottom).
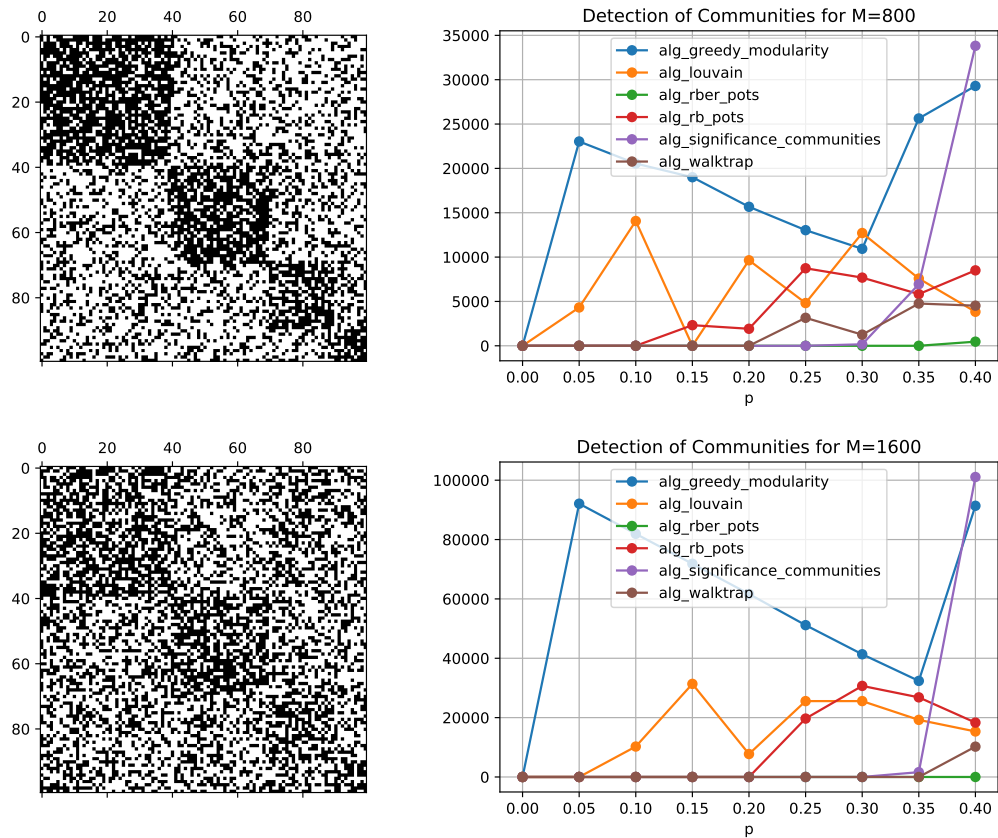
Figure 16. Left: Adjacency matrices for increasing thresholds $p \in \{0.3, 0.4\}$ (top to bottom). Right: Corresponding results for $M = 800$ (top), $M = 1600$ (bottom).
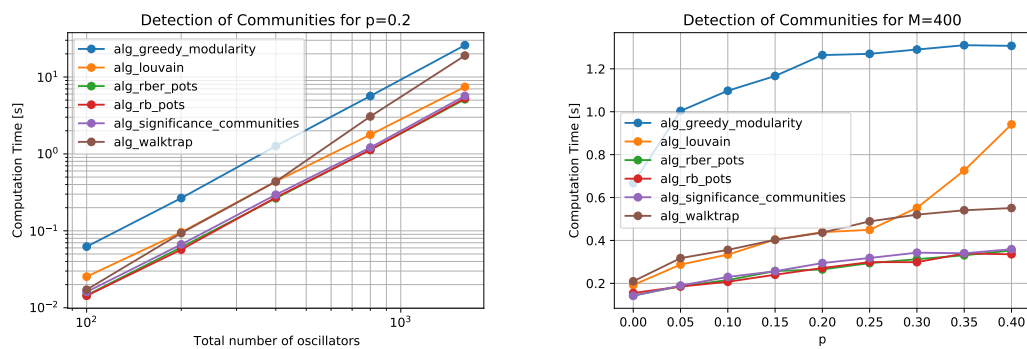

Figure 17. Average computation time of the considered community detection algorithms over eight runs. Left: The displayed numerical results, obtained for the threshold $p = 0.2$, reflect a quadratic scaling with respect to the total number of oscillators. Right: For larger deviations of the adjacency matrices from the underlying block diagonal matrix, related to increasing values of the threshold p, the computation time increases, in general.
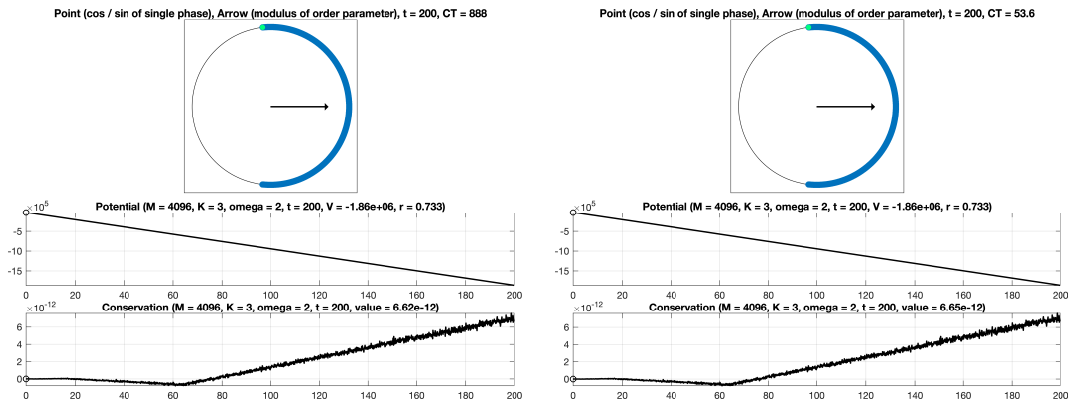
Figure 18. Numerical integration of a Kuramoto model on a graph with the common uniform scaling. The associated adjacency matrix has a structure as illustrated in Figure 14 (left: matrix without recognisable block structure, right: permuted matrix with recognisable block structure). Left: Evaluation of the right-hand side by straightforward summation. Right: Employing the underlying block structure and using the precomputation of sums permits a significant reduction of the computation time CT.
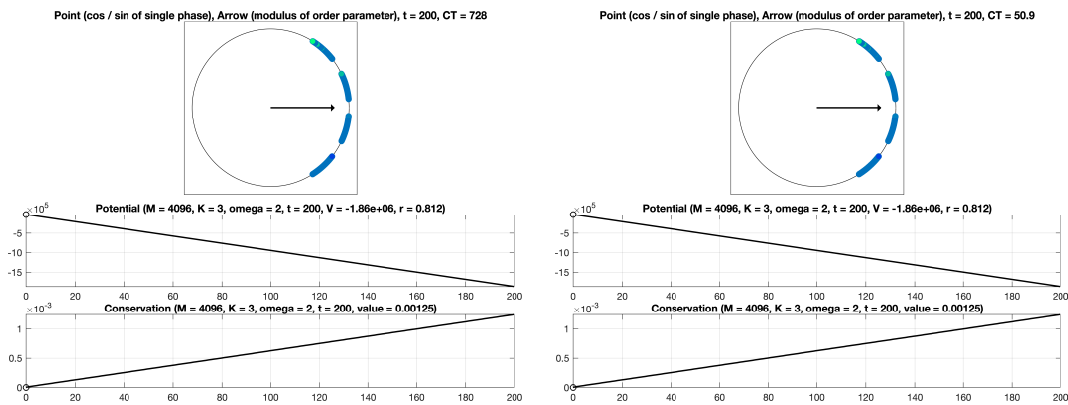

Figure 19. Corresponding results for the non-uniform scaling. Synchronisation within four communities is observed. Due to the lack of symmetry of the system, the conservation property does not hold.