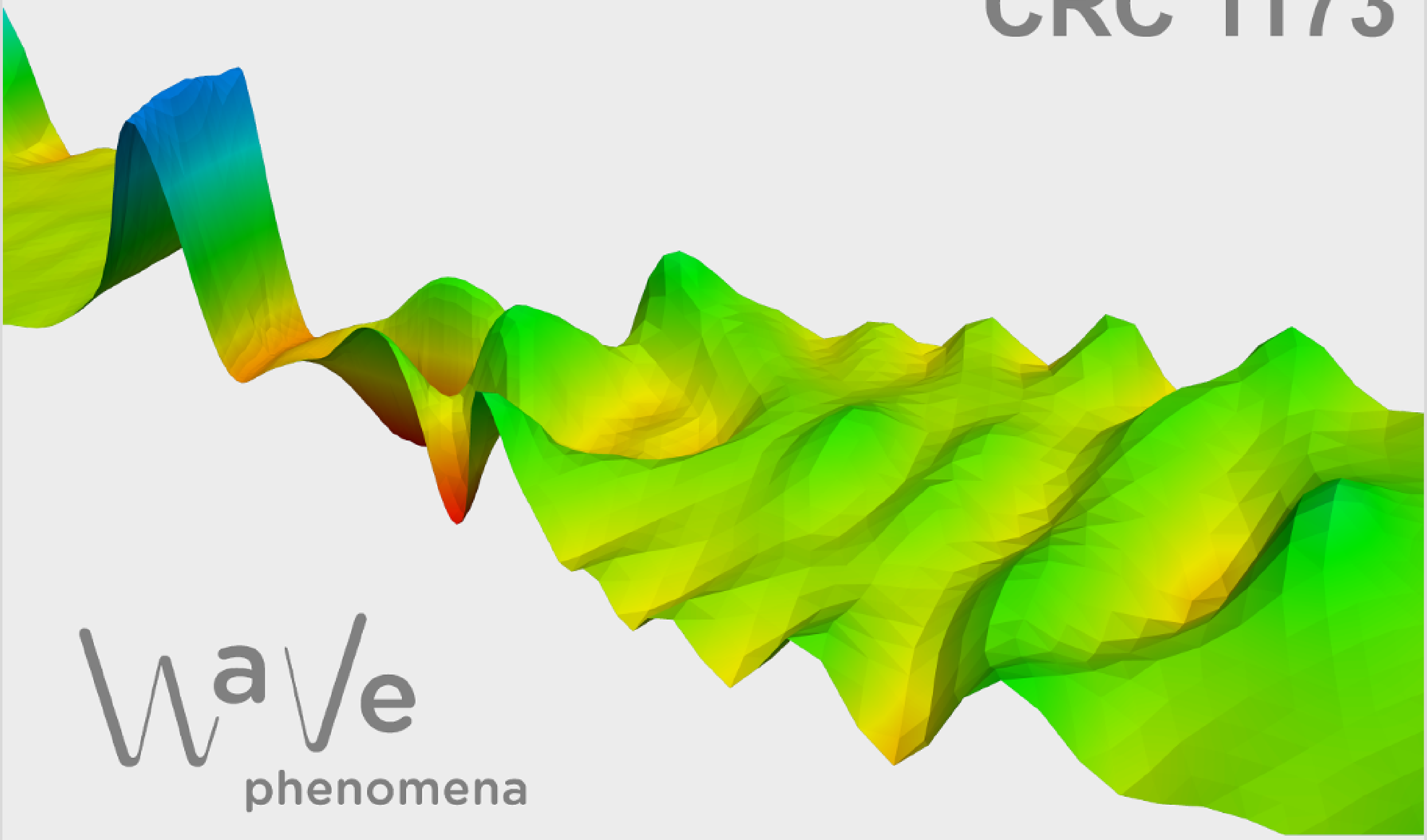
An unconventional robust integrator for dynamical low-rank approximation

Gianluca Ceruti, Christian Lubich

CRC Preprint 2020/41, December 2020

KARLSRUHE INSTITUTE OF TECHNOLOGY

CRC 1173



Participating universities

Universität Stuttgart

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

TU
WIEN

Funded by

DFG

ISSN 2365-662X

Version of 1 October 2020

An unconventional robust integrator for dynamical low-rank approximation

Gianluca Ceruti, Christian Lubich

Abstract We propose and analyse a numerical integrator that computes a low-rank approximation to large time-dependent matrices that are either given explicitly via their increments or are the unknown solution to a matrix differential equation. Furthermore, the integrator is extended to the approximation of time-dependent tensors by Tucker tensors of fixed multilinear rank. The proposed low-rank integrator is different from the known projector-splitting integrator for dynamical low-rank approximation, but it retains the important robustness to small singular values that has so far been known only for the projector-splitting integrator. The new integrator also offers some potential advantages over the projector-splitting integrator: It avoids the backward time integration substep of the projector-splitting integrator, which is a potentially unstable substep for dissipative problems. It offers more parallelism, and it preserves symmetry or anti-symmetry of the matrix or tensor when the differential equation does. Numerical experiments illustrate the behaviour of the proposed integrator.

Keywords dynamical low-rank approximation · structure-preserving integrator · matrix and tensor differential equations · Tucker tensor format

Mathematics Subject Classification (2000) 65L05 · 65L20 · 65L70 · 15A69

1 Introduction

For the approximation of huge time-dependent matrices (or tensors) that are the solution to a matrix differential equation, dynamical low-rank approximation [11, 12] projects the right-hand side function of the differential equation to the tangent space of matrices (or tensors) of a fixed rank at the current approximation. This yields differential equations for the factors of an SVD-like decomposition of the time-dependent low-rank approximation. The direct numerical integration of these differential equations by standard methods such as explicit or implicit Runge–Kutta

G. Ceruti and Ch. Lubich
Mathematisches Institut, Universität Tübingen, Auf der Morgenstelle 10, D-72076 Tübingen, Germany. E-mail: {ceruti, lubich}@na.uni-tuebingen.de

methods is highly problematic because in the typical presence of small singular values in the approximation, it leads to severe step size restrictions proportional to the smallest nonzero singular value. This difficulty does not arise with the projector-splitting integrator proposed in [15], which foregoes a direct time discretization of the differential equations for the factors and instead splits the orthogonal projection onto the tangent space, which is an alternating sum of subprojections. This approach leads to an efficiently implementable integrator that is robust to small singular values [10,15]. It has been extended, together with its robustness properties, from the matrix case to Tucker tensors in [14,17], to tensor trains / matrix product states in [16,7], and to general tree tensor networks in [5].

In the present paper we propose and analyse a different integrator that is shown to have the same robust error behaviour as the projector-splitting integrator. This new integrator can apparently not be interpreted as a splitting integrator or be included in another familiar class of integrators. Its substeps look formally similar to those of the projector-splitting integrator but are arranged in a different, less sequential way. Like in the projector-splitting integrator, the differential equations in the substeps are linear if the original differential equation is linear, even though the projected differential equation becomes nonlinear. The new integrator bears some similarity also to the constant-mean-field integrator of [3] and the splitting integrator of [9].

Beyond the robustness to small singular values, the new integrator has some favourable further properties that are not shared with the projector-splitting integrator. Maybe most importantly, it has no backward time integration substep as in the projector-splitting integrator. This appears advantageous in strongly dissipative problems, where the backward time integration step represents an unstable substep. Moreover, the new integrator has enhanced parallelism in its substeps, and in the Tucker tensor case even a reduced serial computational cost. It preserves symmetry or anti-symmetry of the matrix or tensor when the differential equation does. It reduces to the (anti-)symmetry-preserving low-rank integrator of [4] in this case.

On the other hand, unlike the projector-splitting integrator it cannot be efficiently extended to a time-reversible integrator. When applied to the time-dependent Schrödinger equation (as an integrator for the MCTDH method of quantum molecular dynamics; cf. [2,3,14]), the new integrator preserves the norm, but it has no energy conservation as shown in [14] for the projector-splitting integrator.

In Section 2 we recapitulate dynamical low-rank approximation and the projector-splitting integrator for the matrix case. We restate its exactness property and its robust error bound.

In Section 3 we present the new low-rank matrix integrator and show that it has the same exactness property and robust error bound as the matrix projector-splitting integrator.

In Section 4 we recapitulate dynamical low-rank approximation by Tucker tensors of fixed multilinear rank and the extension of the projector-splitting integrator to the Tucker tensor case.

In Section 5 we present the new low-rank Tucker tensor integrator and show that it has the same exactness property and robust error bound as the Tucker tensor projector-splitting integrator.

In Section 6 we illustrate the behaviour of the new low-rank matrix and Tucker tensor integrators by numerical experiments.

While we describe the integrator for real matrices and tensors, the algorithm and its properties extend in a straightforward way to complex matrices and tensors, requiring only some care in using transposes $\mathbf{U}^\top$ versus adjoints $\mathbf{U}^* = \overline{\mathbf{U}}^\top$.

Throughout the paper, we use the convention to denote matrices by boldface capital letters and tensors by italic capital letters.

2 Recap: the matrix projector-splitting integrator

Dynamical low-rank approximation of time-dependent matrices [11] replaces the exact solution $\mathbf{A}(t) \in \mathbb{R}^{m\times n}$ of a (too large) matrix differential equation

$$\dot{\mathbf{A}}(t) = \mathbf{F}(t, \mathbf{A}(t)), \qquad \mathbf{A}(t_0) = \mathbf{A}_0 \tag{1}$$

by the solution $\mathbf{Y}(t) \in \mathbb{R}^{m\times n}$ of rank r of the differential equation projected to the tangent space of the manifold of rank-r matrices at the current approximation,

$$\dot{\mathbf{Y}}(t) = \mathrm{P}(\mathbf{Y}(t))\mathbf{F}(t, \mathbf{Y}(t)), \qquad \mathbf{Y}(t_0) = \mathbf{Y}_0, \tag{2}$$

where the initial rank-r matrix $\mathbf{Y}_0$ is typically obtained from a truncated singular value decomposition (SVD) of $\mathbf{A}_0$. (We note that $\mathbf{F}(t, \mathbf{Y}) = \dot{\mathbf{A}}(t)$ if $\mathbf{A}(t)$ is given explicitly.) For the actual computation with rank-r matrices, they are represented in a non-unique factorized SVD-like form

$$\mathbf{Y}(t) = \mathbf{U}(t)\mathbf{S}(t)\mathbf{V}(t)^\top, \tag{3}$$

where the slim matrices $\mathbf{U}(t) \in \mathbb{R}^{m\times r}$ and $\mathbf{V}(t) \in \mathbb{R}^{n\times r}$ each have r orthonormal columns, and the small matrix $\mathbf{S}(t) \in \mathbb{R}^{r\times r}$ is invertible.

The orthogonal tangent space projection $\mathrm{P}(\mathbf{Y})$ can be written explicitly as an alternating sum of three subprojections onto the co-range, the intersection of co-range and range, and the range of the rank-r matrix $\mathbf{Y}$ [11]. The projector-splitting integrator of [15] splits the right-hand side of (2) according to the three subprojections in the stated ordering and solves the subproblems consecutively in the usual way of a Lie–Trotter or Strang splitting. This approach yields an efficient time-stepping algorithm that updates the factors in the SVD-like decomposition of the rank-r matrices in every time step, alternating between solving differential equations for matrices of the dimension of the factor matrices and orthogonal decompositions of slim matrices.

One time step from t_0 to $t_1 = t_0 + h$ starting from a factored rank-r matrix $\mathbf{Y}_0 = \mathbf{U}_0\mathbf{S}_0\mathbf{V}_0^\top$ proceeds as follows:

1. **K-step** : Update $\mathbf{U}_0 \to \mathbf{U}_1, \mathbf{S}_0 \to \hat{\mathbf{S}}_1$
   Integrate from $t = t_0$ to t_1 the $m \times r$ matrix differential equation
   $$\dot{\mathbf{K}}(t) = \mathbf{F}(t, \mathbf{K}(t)\mathbf{V}_0^\top)\mathbf{V}_0, \qquad \mathbf{K}(t_0) = \mathbf{U}_0\mathbf{S}_0.$$
   Perform a QR factorization $\mathbf{K}(t_1) = \mathbf{U}_1\hat{\mathbf{S}}_1$.
2. **S-step** : Update $\hat{\mathbf{S}}_1 \to \tilde{\mathbf{S}}_0$
   Integrate from $t = t_0$ to t_1 the $r \times r$ matrix differential equation
   $$\dot{\mathbf{S}}(t) = -\mathbf{U}_1^\top\mathbf{F}(t, \mathbf{U}_1\mathbf{S}(t)\mathbf{V}_0^\top)\mathbf{V}_0, \qquad \mathbf{S}(t_0) = \hat{\mathbf{S}}_1,$$
   and set $\tilde{\mathbf{S}}_0 = \mathbf{S}(t_1)$.

3. **L-step** : Update $\mathbf{V}_0 \to \mathbf{V}_1, \tilde{\mathbf{S}}_0 \to \mathbf{S}_1$
   Integrate from $t = t_0$ to t_1 the $n \times r$ matrix differential equation

$$\dot{\mathbf{L}}(t) = \mathbf{F}(t, \mathbf{U}_1 \mathbf{L}(t)^\top)^\top \mathbf{U}_1, \qquad \mathbf{L}(t_0) = \mathbf{V}_0 \tilde{\mathbf{S}}_0^\top.$$

   Perform a QR factorization $\mathbf{L}(t_1) = \mathbf{V}_1 \mathbf{S}_1^\top$.

   Then, the approximation after one time step is given by

$$\mathbf{Y}_1 = \mathbf{U}_1 \mathbf{S}_1 \mathbf{V}_1^\top.$$

To proceed further, $\mathbf{Y}_1$ is taken as the starting value for the next step, and so on.

The projector-splitting integrator has very favourable properties. First, it reproduces rank-r matrices exactly.

Theorem 1 (Exactness property, [15, Theorem 4.1]) *Let $\mathbf{A}(t) \in \mathbb{R}^{m\times n}$ be of rank r for $t_0 \le t \le t_1$, so that $\mathbf{A}(t)$ has a factorization* (3)*, $\mathbf{A}(t) = \mathbf{U}(t)\mathbf{S}(t)\mathbf{V}(t)^\top$. Moreover, assume that the $r \times r$ matrices $\mathbf{U}(t_1)^\top \mathbf{U}(t_0)$ and $\mathbf{V}(t_1)^\top \mathbf{V}(t_0)$ are invertible. With $\mathbf{Y}_0 = \mathbf{A}(t_0)$, the projector-splitting integrator for $\dot{\mathbf{Y}}(t) = \mathrm{P}(\mathbf{Y}(t))\dot{\mathbf{A}}(t)$ is then exact: $\mathbf{Y}_1 = \mathbf{A}(t_1)$.*

Even more remarkable, the algorithm is robust to the presence of small singular values of the solution or its approximation, as opposed to standard integrators applied to (2) or the equivalent differential equations for the factors $\mathbf{U}(t)$, $\mathbf{S}(t)$, $\mathbf{V}(t)$, which contain a factor $\mathbf{S}(t)^{-1}$ on the right-hand sides [11, Prop. 2.1]. The appearance of small singular values is ubiquitous in low-rank approximation, because the smallest singular value retained in the approximation cannot be expected to be much larger than the largest discarded singular value of the solution, which is required to be small for good accuracy of the low-rank approximation.

Theorem 2 (Robust error bound, [10, Theorem 2.1]) *Let $\mathbf{A}(t)$ denote the solution of the matrix differential equation* (1)*. Assume that the following conditions hold in the Frobenius norm $\|\cdot\| = \|\cdot\|_F$:*

1. $\mathbf{F}$ is Lipschitz-continuous and bounded: for all $\mathbf{Y}, \widetilde{\mathbf{Y}} \in \mathbb{R}^{m\times n}$ and $0 \le t \le T$,

$$\|\mathbf{F}(t,\mathbf{Y}) - \mathbf{F}(t,\widetilde{\mathbf{Y}})\| \le L\|\mathbf{Y} - \widetilde{\mathbf{Y}}\|, \qquad \|\mathbf{F}(t,\mathbf{Y})\| \le B \ .$$

2. The non-tangential part of $\mathbf{F}(t,\mathbf{Y})$ is ε-small:

$$\|(\mathbf{I} - \mathrm{P}(\mathbf{Y}))\mathbf{F}(t,\mathbf{Y})\| \le \varepsilon$$

for all $\mathbf{Y} \in \mathcal{M}$ in a neighbourhood of $\mathbf{A}(t)$ and $0 \le t \le T$.

3. The error in the initial value is δ-small:

$$\|\mathbf{Y}_0 - \mathbf{A}_0\| \le \delta.$$

Let $\mathbf{Y}_n$ denote the rank-r approximation to $\mathbf{A}(t_n)$ at $t_n = nh$ obtained after n steps of the projector-splitting integrator with step-size $h > 0$. Then, the error satisfies for all n with $t_n = nh \le T$

$$\|\mathbf{Y}_n - \mathbf{A}(t_n)\| \le c_0\delta + c_1\varepsilon + c_2 h,$$

where the constants c_i only depend on L, B, and T. In particular, the constants are independent of singular values of the exact or approximate solution.

In [10, Section 2.6.3] it is shown that an inexact solution of the matrix differential equations in the projector-splitting integrator leads to an additional error that is bounded in terms of the local errors in the inexact substeps, again with constants that do not depend on small singular values.

Numerical experiments with the matrix projector-splitting integrator and comparisons with standard numerical integrators are reported in [15, 10].

3 A new robust low-rank matrix integrator

We now present a different integrator that has the same exactness and robustness properties as the projector-splitting integrator but which differs in the following favourable properties:

1. The solution of the differential equations for the $m \times r$ and $n \times r$ matrices can be done in parallel, and also the two QR decompositions can be done in parallel.
2. The differential equation for the small $r \times r$ matrix is solved forward in time, not backwards.
3. The integrator preserves (skew-)symmetry if the differential equation does.

While item 1. can clearly speed up the computation, item 2. is of interest for strongly dissipative problems, for which the S-step in the projector-splitting algorithm with the minus sign in the differential equations is an unstable substep of the algorithm. This does not appear in the new algorithm. We mention that in [1], the problem of the backward substep for parabolic problems has recently been addressed in a different way.

On the other hand, contrary to the projector-splitting integrator, there is apparently no efficient way to construct a time-reversible integrator from this new integrator.

3.1 Formulation of the algorithm

One time step of integration from time t_0 to $t_1 = t_0 + h$ starting from a factored rank-r matrix $\mathbf{Y}_0 = \mathbf{U}_0 \mathbf{S}_0 \mathbf{V}_0^\top$ computes an updated rank-r factorization $\mathbf{Y}_1 = \mathbf{U}_1 \mathbf{S}_1 \mathbf{V}_1^\top$ as follows.

1. Update $\mathbf{U}_0 \to \mathbf{U}_1$ and $\mathbf{V}_0 \to \mathbf{V}_1$ in parallel:

   K-step: Integrate from $t = t_0$ to t_1 the $m \times r$ matrix differential equation

   $$\dot{\mathbf{K}}(t) = \mathbf{F}(t, \mathbf{K}(t)\mathbf{V}_0^\top)\mathbf{V}_0, \qquad \mathbf{K}(t_0) = \mathbf{U}_0 \mathbf{S}_0.$$

   Perform a QR factorization $\mathbf{K}(t_1) = \mathbf{U}_1 \mathbf{R}_1$ and compute the $r \times r$ matrix $\mathbf{M} = \mathbf{U}_1^\top \mathbf{U}_0$.

   L-step : Integrate from $t = t_0$ to t_1 the $n \times r$ matrix differential equation

   $$\dot{\mathbf{L}}(t) = \mathbf{F}(t, \mathbf{U}_0 \mathbf{L}(t)^\top)^\top \mathbf{U}_0, \qquad \mathbf{L}(t_0) = \mathbf{V}_0 \mathbf{S}_0^\top.$$

   Perform a QR factorization $\mathbf{L}(t_1) = \mathbf{V}_1 \widetilde{\mathbf{R}}_1$ and compute the $r \times r$ matrix $\mathbf{N} = \mathbf{V}_1^\top \mathbf{V}_0$.

2. Update $\mathbf{S}_0 \to \mathbf{S}_1$:

   S-step : Integrate from $t = t_0$ to t_1 the $r \times r$ matrix differential equation

   $$\dot{\mathbf{S}}(t) = \mathbf{U}_1^\top \mathbf{F}(t, \mathbf{U}_1 \mathbf{S}(t) \mathbf{V}_1^\top) \mathbf{V}_1, \qquad \mathbf{S}(t_0) = \mathbf{M} \mathbf{S}_0 \mathbf{N}^\top,$$

   and set $\mathbf{S}_1 = \mathbf{S}(t_1)$.

The $m \times r$, $n \times r$ and $r \times r$ matrix differential equations in the substeps are solved approximately using a standard integrator, e.g., an explicit or implicit Runge–Kutta method or an exponential integrator when $\mathbf{F}$ is dominantly linear.

We note that the L-step equals the K-step for the transposed function $\mathbf{G}(t, \mathbf{Y}) = \mathbf{F}(t, \mathbf{Y}^\top)^\top$ and transposed starting values. Unlike the projector-splitting algorithm, the triangular factors of the QR-decompositions are not reused. The S-step can be viewed as a Galerkin method for the differential equation (1) in the space of matrices $\mathbf{U}_1 \mathbf{S} \mathbf{V}_1^\top$ generated by the updated basis matrices. In contrast to the projector-splitting integrator, there is no minus sign on the right-hand side of the differential equation for $\mathbf{S}(t)$. We further note that $\mathbf{U}_1$ of the new integrator is identical to $\mathbf{U}_1$ of the projector-splitting integrator, but $\mathbf{V}_1$ is in general different.

Remark 1 There exists a modification where all three differential equations for $\mathbf{K}$, $\mathbf{L}$ and $\mathbf{S}$ can be solved in parallel. That variant solves the K- and L-steps as above, but in the S-step it solves instead the $r \times r$ matrix differential equation

$$\dot{\mathbf{S}}(t) = \mathbf{U}_0^\top \mathbf{F}(t, \mathbf{U}_0 \mathbf{S}(t) \mathbf{V}_0^\top) \mathbf{V}_0, \qquad \mathbf{S}(t_0) = \mathbf{S}_0$$

and finally sets

$$\mathbf{S}_1 = \mathbf{M}^{-\top}\, \mathbf{S}(t_1)\, \mathbf{N}^{-1}.$$

This modified integrator can be shown to have the same exactness property as proved below for the integrator formulated above, and also a similar robust error bound *under the condition that* the inverses of the matrices $\mathbf{M}$ and $\mathbf{N}$ are bounded by a constant. This condition can, however, be guaranteed only for step sizes that are small in comparison to the smallest nonzero singular value. In our numerical experiments this method did not behave as reliably as the method proposed above, and despite its interesting properties it will therefore not be further discussed in the following.

3.2 Exactness property and robust error bound

We will prove the following remarkable results for the integrator of Section 3.1.

Theorem 3 *The exactness property of Theorem* 1 *holds verbatim also for the new integrator.*

Theorem 4 *The robust error bound of Theorem* 2 *holds verbatim also for the new integrator.*

As in [10, Section 2.6.3], it can be further shown that an inexact solution of the matrix differential equations in the projector-splitting integrator leads to an additional error that is bounded in terms of the local errors in the inexact substeps, again with constants that do not depend on small singular values.

3.3 Proof of Theorem 3

For the proof of Theorem 3 we need the following auxiliary result, which extends an analogous result in [4] for symmetric matrices.

Lemma 1 *Let* $\mathbf{A}(t) \in \mathbb{R}^{m\times n}$ *be of rank* r *for* $t_0 \le t \le t_1$*, so that* $\mathbf{A}(t)$ *has a factorization* (3)*,* $\mathbf{A}(t) = \mathbf{U}(t)\mathbf{S}(t)\mathbf{V}(t)^\top$*. Moreover, assume that the* $r \times r$ *matrix* $\mathbf{V}(t_1)^\top\mathbf{V}(t_0)$ *is invertible. Then,*

$$\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}(t_1) = \mathbf{A}(t_1).$$

Proof The solution of the K-step at time t_1 is

$$\mathbf{K}(t_1) = \mathbf{A}(t_0)\mathbf{V}_0 + \big(\mathbf{A}(t_1) - \mathbf{A}(t_0)\big)\mathbf{V}_0 = \mathbf{A}(t_1)\mathbf{V}_0.$$

Hence,

$$\mathbf{K}(t_1) = \mathbf{U}(t_1)\big[\mathbf{S}(t_1)(\mathbf{V}(t_1)^\top\mathbf{V}(t_0))\big]\ .$$

By assumption, the factor in big square brackets is invertible. Computing a QR-decomposition of this term, we have

$$\mathbf{K}(t_1) = \mathbf{U}(t_1)\mathbf{Q}\mathbf{R}\ ,$$

where $\mathbf{Q} \in \mathbb{R}^{r\times r}$ is an orthogonal matrix and $\mathbf{R} \in \mathbb{R}^{r\times r}$ is invertible and upper triangular. The QR-factorization of $\mathbf{K}(t_1)$ thus yields

$$\mathbf{U}_1 = \mathbf{U}(t_1)\mathbf{Q} \in \mathbb{R}^{m\times r}.$$

To conclude,

$$\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}(t_1) = \mathbf{U}(t_1)\mathbf{Q}\mathbf{Q}^\top\mathbf{U}(t_1)^\top\mathbf{A}(t_1) = \mathbf{U}(t_1)\mathbf{U}(t_1)^\top\mathbf{A}(t_1) = \mathbf{A}(t_1),$$

which is the stated result. □

Proof (of Theorem 3) Since the L-step is the K-step for the transposed matrix $\mathbf{A}(t)^\top$, which has the same rank as $\mathbf{A}(t)$, it follows from Lemma 1 that

$$\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}(t_1) = \mathbf{A}(t_1), \qquad \mathbf{V}_1\mathbf{V}_1^\top\mathbf{A}(t_1)^\top = \mathbf{A}(t_1)^\top. \tag{4}$$

The integrator yields in the S-step

$$\mathbf{S}_1 = \mathbf{U}_1^\top\mathbf{Y}_0\mathbf{V}_1 + \mathbf{U}_1^\top(\mathbf{A}(t_1) - \mathbf{A}(t_0))\mathbf{V}_1 = \mathbf{U}_1^\top\mathbf{A}(t_1)\mathbf{V}_1,$$

since $\mathbf{Y}_0 = \mathbf{A}(t_0)$. The result after a time step of the new integrator is

$$\mathbf{Y}_1 = \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^\top = \mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}(t_1)(\mathbf{V}_1\mathbf{V}_1^\top) = \mathbf{A}(t_1),$$

where the last equality holds because of (4). □

3.4 Proof of Theorem 4

Under the assumptions of Theorem 2, we introduce the quantity

$$\vartheta := (4e^{Lh}BL + 9BL)h^2 + (3e^{Lh} + 4)\varepsilon h + e^{Lh}\delta\,, \tag{5}$$

which is the local error bound of the projector-splitting integrator after one time step, as proved in [10, Theorem 2.1].

Lemma 2 *Let $\mathbf{A}_1$ be the solution at time $t_1 = t_0 + h$ of the full problem (1) with initial condition $\mathbf{A}_0$. Assume that conditions 1.-3. of Theorem 2 are fulfilled. Then,*

$$\|\mathbf{U}_1\mathbf{U}_1^\top \mathbf{A}_1 - \mathbf{A}_1\| \leq \vartheta.$$

Proof The result is proved in the course of the proof of Lemma 1 in [4]. We give the proof here for the convenience of the reader. The local error analysis in [10] shows that the $r \times n$ matrix $\mathbf{Z} = \mathbf{S}_1^{\text{ps}}\mathbf{V}_1^{\text{ps},\top}$, where $\mathbf{S}_1^{\text{ps}}$ and $\mathbf{V}_1^{\text{ps}}$ are the matrices computed in the third substep of the projector-splitting algorithm, satisfies

$$\|\mathbf{U}_1\mathbf{Z} - \mathbf{A}_1\| \leq \vartheta.$$

The square of the left-hand side can be split into two terms:

$$\begin{aligned}\|\mathbf{U}_1\mathbf{Z} - \mathbf{A}_1\|^2 &= \|\mathbf{U}_1\mathbf{Z} - \mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1 + \mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1 - \mathbf{A}_1\|^2 \\ &= \|\mathbf{U}_1\mathbf{U}_1^\top(\mathbf{U}_1\mathbf{Z} - \mathbf{A}_1) + (\mathbf{I} - \mathbf{U}_1\mathbf{U}_1^\top)\mathbf{A}_1\|^2 \\ &= \|\mathbf{U}_1\mathbf{U}_1^\top(\mathbf{U}_1\mathbf{Z} - \mathbf{A}_1)\|^2 + \|(\mathbf{I} - \mathbf{U}_1\mathbf{U}_1^\top)\mathbf{A}_1\|^2.\end{aligned}$$

Hence,

$$\|\mathbf{U}_1\mathbf{U}_1^\top(\mathbf{U}_1\mathbf{Z} - \mathbf{A}_1)\|^2 + \|(\mathbf{I} - \mathbf{U}_1\mathbf{U}_1^\top)\mathbf{A}_1\|^2 \leq \vartheta^2.$$

This yields the stated result for the second term. □

Lemma 3 *Let $\mathbf{A}_1$, $\mathbf{U}_1$ and $\mathbf{V}_1$ be defined as above. The following estimate holds:*

$$\|\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top - \mathbf{A}_1\| \leq 2\vartheta.$$

Proof The L-step is the K-step for the transposed function $\mathbf{G}(t, \mathbf{Y}) = \mathbf{F}(t, \mathbf{Y}^\top)^\top$, which again fulfills conditions 1.-3. of Theorem 2. Conditions 1. and 3. hold because of the invariance of the Frobenius norm under transposition. Condition 2. holds because

$$\|(\mathbf{I} - \mathrm{P}(\mathbf{Y}))\mathbf{G}(t, \mathbf{Y})\| = \|(\mathbf{I} - \mathrm{P}(\mathbf{Y}^\top))\mathbf{F}(t, \mathbf{Y}^\top)\| \leq \varepsilon,$$

where we used the identity $\mathrm{P}(\mathbf{Y})\mathbf{Z}^\top = \big[\mathrm{P}(\mathbf{Y}^\top)\mathbf{Z}\big]^\top$. From Lemma 2 we thus have

$$\begin{aligned}&\|\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1 - \mathbf{A}_1\| \leq \vartheta, \\ &\|\mathbf{V}_1\mathbf{V}_1^\top\mathbf{A}_1^\top - \mathbf{A}_1^\top\| \leq \vartheta.\end{aligned} \tag{6}$$

This implies that

$$\begin{aligned}\|\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top - \mathbf{A}_1\| &\leq \|\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top - \mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top + \mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top - \mathbf{A}_1\| \\ &\leq \|\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top - \mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top\| + \|\mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top - \mathbf{A}_1\| \\ &\leq \|\big(\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1 - \mathbf{A}_1\big)\mathbf{V}_1\mathbf{V}_1^\top\| + \|\mathbf{V}_1\mathbf{V}_1^\top\mathbf{A}_1^\top - \mathbf{A}_1^\top\| \\ &\leq \|\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1 - \mathbf{A}_1\| \cdot \|\mathbf{V}_1\mathbf{V}_1^\top\|_2 + \|\mathbf{V}_1\mathbf{V}_1^\top\mathbf{A}_1^\top - \mathbf{A}_1^\top\|.\end{aligned}$$

Since $\|\mathbf{V}_1\mathbf{V}_1^\top\|_2 = 1$, the result follows from (6). □

In the following lemma, we show that the approximation given after one time step is $O(h(h+\varepsilon))$ close to the solution of system (1) when the starting values coincide.

Lemma 4 (Local Error) *If $\mathbf{A}_0 = \mathbf{Y}_0$, the following local error bound holds:*

$$\|\mathbf{Y}_1 - \mathbf{A}_1\| \le h(\hat{c}_1\varepsilon + \hat{c}_2 h),$$

where the constants only depend on L and B and a bound of the step size. In particular, the constants are independent of singular values of the exact or approximate solution.

Proof With a few crucial modifications, the proof is similar to that of [4, Lemma 2]. We report here the full proof for completeness and convenience of the reader. By the identity $\mathbf{Y}_1 = \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^\top$ and Lemma 3 we have that

$$\begin{aligned}\|\mathbf{Y}_1 - \mathbf{A}_1\| &\le \|\mathbf{Y}_1 - \mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top\| + \|\mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1\mathbf{V}_1^\top - \mathbf{A}_1\|\\ &\le \|\mathbf{U}_1(\mathbf{S}_1 - \mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1)\mathbf{V}_1^\top\| + 2\vartheta\\ &= \|\mathbf{S}_1 - \mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1\| + 2\vartheta.\end{aligned}$$

The analysis of the local error thus reduces to estimating $\|\mathbf{S}_1 - \mathbf{U}_1^\top\mathbf{A}_1\mathbf{V}_1\|$. To this end, we introduce the following quantity: for $t_0 \le t \le t_1$,

$$\widetilde{\mathbf{S}}(t) := \mathbf{U}_1^\top\mathbf{A}(t)\mathbf{V}_1.$$

We write

$$\mathbf{A}(t) = \mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}(t)\mathbf{V}_1\mathbf{V}_1^\top + \Big(\mathbf{A}(t) - \mathbf{U}_1\mathbf{U}_1^\top\mathbf{A}(t)\mathbf{V}_1\mathbf{V}_1^\top\Big) = \mathbf{U}_1\widetilde{\mathbf{S}}(t)\mathbf{V}_1^\top + \mathbf{R}(t),$$

where $\mathbf{R}(t)$ denotes the term in big brackets. Lemma 3 and the bound B of $\mathbf{F}$ yield, for $t_0 \le t \le t_1$,

$$\|\mathbf{A}(t) - \mathbf{A}(t_1)\| \le \int_{t_0}^{t_1} \|\dot{\mathbf{A}}(s)\|\, ds = \int_{t_0}^{t_1} \|\mathbf{F}(s, \mathbf{A}(s))\|\, ds \le Bh.$$

Hence the remainder term is bounded by

$$\|\mathbf{R}(t)\| \le \|\mathbf{R}(t) - \mathbf{R}(t_1)\| + \|\mathbf{R}(t_1)\| \le 2Bh + 2\vartheta.$$

It follows that $\mathbf{F}(t, \mathbf{A}(t))$ can be written as

$$\begin{aligned}\mathbf{F}(t, \mathbf{A}(t)) &= \mathbf{F}(t, \mathbf{U}_1\widetilde{\mathbf{S}}(t)\mathbf{V}_1^\top + \mathbf{R}(t))\\ &= \mathbf{F}(t, \mathbf{U}_1\widetilde{\mathbf{S}}(t)\mathbf{V}_1^\top) + \mathbf{D}(t)\end{aligned}$$

with the defect

$$\mathbf{D}(t) := \mathbf{F}(t, \mathbf{U}_1\widetilde{\mathbf{S}}(t)\mathbf{V}_1^\top + \mathbf{R}(t)) - \mathbf{F}(t, \mathbf{U}_1\widetilde{\mathbf{S}}(t)\mathbf{V}_1^\top).$$

With the Lipschitz constant L of $\mathbf{F}$, the defect is bounded by

$$\|\mathbf{D}(t)\| \le L\|\mathbf{R}(t)\| \le 2L(Bh + \vartheta).$$

We compare the two differential equations

$$\begin{aligned} \dot{\widetilde{\mathbf{S}}}(t) &= \mathbf{U}_1^\top \mathbf{F}(t, \mathbf{U}_1 \widetilde{\mathbf{S}}(t) \mathbf{V}_1^\top) \mathbf{V}_1 + \mathbf{U}_1^\top \mathbf{D}(t) \mathbf{V}_1, \qquad & \widetilde{\mathbf{S}}(t_0) &= \mathbf{U}_1^\top \mathbf{Y}_0 \mathbf{V}_1, \\ \dot{\mathbf{S}}(t) &= \mathbf{U}_1^\top \mathbf{F}(t, \mathbf{U}_1 \mathbf{S}(t) \mathbf{V}_1^\top) \mathbf{V}_1, & \mathbf{S}(t_0) &= \mathbf{U}_1^\top \mathbf{Y}_0 \mathbf{V}_1. \end{aligned}$$

By construction, the solution of the first differential equation at time t_1 is $\widetilde{\mathbf{S}}(t_1) = \mathbf{U}_1^\top \mathbf{A}_1 \mathbf{V}_1$. The solution of the second differential equation is $\mathbf{S}_1$ as given by the S-step of the integrator. With the Gronwall inequality we obtain

$$\|\mathbf{S}_1 - \mathbf{U}_1^\top \mathbf{A}_1 \mathbf{V}_1\| \le \int_{t_0}^{t_1} e^{L(t_1 - s)} \, \|\mathbf{D}(s)\| \, ds \le e^{Lh} \, 2L(Bh + \vartheta)h.$$

The result now follows using the definition of ϑ. □

Using the Lipschitz continuity of the function $\mathbf{F}$, we pass from the local to the global errors by the standard argument of Lady Windermere's fan [8, Section II.3] and thus conclude the proof of Theorem 7.

3.5 Symmetric and skew-symmetric low-rank matrices

We now assume that the right-hand side function in (1) is such that one of the following conditions holds,

$$\mathbf{F}(t, \mathbf{Y}^\top)^\top = \mathbf{F}(t, \mathbf{Y}) \qquad \text{for all} \ \ \mathbf{Y} \in \mathbb{R}^{n \times n} \tag{7}$$

or

$$\mathbf{F}(t, \mathbf{Y}^\top)^\top = -\mathbf{F}(t, -\mathbf{Y}) \qquad \text{for all} \ \ \mathbf{Y} \in \mathbb{R}^{n \times n}. \tag{8}$$

Under these conditions, solutions to (1) with symmetric or skew-symmetric initial data remain symmetric or skew-symmetric, respectively, for all times. We also have preservation of (skew-)symmetry for the new integrator, which does not hold for the projector-splitting integrator.

Theorem 5 *Let* $\mathbf{Y}_0 = \mathbf{U}_0 \mathbf{S}_0 \mathbf{U}_0^\top \in \mathbb{R}^{n \times n}$ *be symmetric or skew-symmetric and assume that the function* $\mathbf{F}$ *satisfies property* (7) *or* (8)*, respectively. Then, the approximation* $\mathbf{Y}_1$ *obtained after one time step of the new integrator is symmetric or skew-symmetric, respectively.*

Proof Let us just consider the skew-symmetric case (8). (The symmetric case is analogous.) The L-step is the K-step for the transposed function $\mathbf{G}(t, \mathbf{Y}) = \mathbf{F}(t, \mathbf{Y}^\top)^\top$, and so the skew-symmetry of $\mathbf{S}_0$ and property (8) imply that $\mathbf{L}(t_1) = -\mathbf{K}(t_1)$, which further yields $\mathbf{V}_1 = \mathbf{U}_1$ and $\mathbf{M} = \mathbf{N}$. These identities show that the initial value and the right-hand side function of the differential equation for $\mathbf{S}(t)$ are skew-symmetric, which implies that $\mathbf{S}(t)$ and hence $\mathbf{S}_1$ are still skew-symmetric. Altogether, the algorithm gives us the skew-symmetric result $\mathbf{Y}_1 = \mathbf{U}_1 \mathbf{S}_1 \mathbf{U}_1^\top$. □

Under condition (7) or (8), the new integrator coincides with the (skew)-symmetry preserving low-rank matrix integrator of [4].

4 Recap: the Tucker tensor projector-splitting integrator

The solution $A(t) \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ of a tensor differential equation

$$\dot{A}(t) = F(t, A(t)), \qquad A(0) = A_0 \tag{9}$$

is approximated by the solution $Y(t) \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ of multilinear rank $\mathbf{r} = (r_1, \dots, r_d)$ of the differential equation projected to the tangent space of the manifold of rank-$\mathbf{r}$ tensors at the current approximation ([12], cf. also [2]),

$$\dot{Y}(t) = \mathrm{P}(Y(t))F(t, Y(t)), \qquad Y(t_0) = Y_0, \tag{10}$$

where Y_0 is a rank-$\mathbf{r}$ approximation to A_0. Tensors $Y(t)$ of multilinear rank $\mathbf{r}$ are represented in the Tucker form [6], written here in a notation following [13]:

$$\begin{aligned} &Y(t) = C(t) \bigtimes_{i=1}^{d} \mathbf{U}_i(t), \\ &i.e., \quad y_{i_1,\dots,i_d}(t) = \sum_{j_1,\dots,j_d} c_{j_1,\dots,j_d}(t)\, u_{i_1,j_1}(t) \dots u_{i_d,j_d}(t), \end{aligned} \tag{11}$$

where the slim basis matrices $\mathbf{U}_i \in \mathbb{R}^{n_i \times r_i}$ have orthonormal columns and the smaller core tensor $C(t) \in \mathbb{R}^{r_1 \times \cdots \times r_d}$ is of full multilinear rank $\mathbf{r}$.

The orthogonal tangent space projection $\mathrm{P}(Y)$ is given as an alternating sum of $2d - 1$ subprojections [14], and like in the matrix case, a projector-splitting integrator with favourable properties can be formulated and efficiently implemented [14,17]. The algorithm runs through the modes $i = 1, \dots, d$ and solves differential equations for matrices of the dimension of the slim basis matrices and for the core tensor, alternating with orthogonalizations of slim matrices. Like the matrix projector-splitting integrator, also the Tucker tensor projector-splitting integrator has the exactness property and a robust error bound independently of small singular values of matricizations of the core tensor [17, Theorems 4.1 and 5.1].

5 A new robust low-rank Tucker tensor integrator

The low-rank numerical integrator defined in Section 3 for the matrix case extends in a natural way to the Tucker tensor format, and this extension still has the exactness property and robust error bounds that are independent of small singular values of matricizations of the core tensor.

In comparison with the Tucker integrator of [14] and [17], the new Tucker tensor integrator has the following favourable properties:

1. The solution of the differential equations for the $n_i \times r_i$ matrices can be done in parallel for $i = 1, \dots, d$, and also the QR decompositions can be done in parallel.
2. No differential equations are solved backward in time. No differential equations for $r_i \times r_i$ matrices need to be solved.
3. The integrator preserves (anti-)symmetry if the differential equation does.

On the other hand, in contrast to the projector-splitting Tucker integrator there is apparently no efficient way to construct a time-reversible integrator from this new Tucker integrator.

5.1 Formulation of the algorithm

One time step of integration from time t_0 to $t_1 = t_0 + h$ starting from a Tucker tensor of multilinear rank $(r_1, \dots, r_d)$ in factorized form, $Y_0 = C_0 \bigtimes_{i=1}^d \mathbf{U}_i^0$, computes an updated Tucker tensor of multilinear rank $(r_1, \dots, r_d)$ in factorized form, $Y_1 = C_1 \bigtimes_{i=1}^d \mathbf{U}_i^1$, in the following way:

1. Update the basis matrices $\mathbf{U}_i^0 \to \mathbf{U}_i^1$ for $i = 1, \dots, d$ in parallel:

   Perform a QR factorization of the transposed i-mode matricization of the core tensor:
$$\mathbf{Mat}_i(C_0)^\top = \mathbf{Q}_i \mathbf{S}_i^{0,\top}.$$
   With $\mathbf{V}_i^{0,\top} = \mathbf{Q}_i^\top \bigotimes_{j \neq i}^d \mathbf{U}_j^{0,\top} \in \mathbb{R}^{r_i \times n_{\neg i}}$ (which yields $\mathbf{Mat}_i(Y_0) = \mathbf{U}_i^0 \mathbf{S}_i^0 \mathbf{V}_i^{0,\top}$)

   and the matrix function $\mathbf{F}_i(t, \cdot) := \mathbf{Mat}_i \circ F(t, \cdot) \circ Ten_i$, integrate from $t = t_0$ to t_1 the $n_i \times r_i$ matrix differential equation
$$\dot{\mathbf{K}}_i(t) = \mathbf{F}_i(t, \mathbf{K}_i(t) \mathbf{V}_i^{0,\top}) \mathbf{V}_i^0, \qquad \mathbf{K}_i(t_0) = \mathbf{U}_i^0 \mathbf{S}_i^0.$$
   Perform a QR factorization $\mathbf{K}_i(t_1) = \mathbf{U}_i^1 \mathbf{R}_i^1$ and compute the $r_i \times r_i$ matrix $\mathbf{M}_i = \mathbf{U}_i^{1,\top} \mathbf{U}_i^0$.
2. Update the core tensor $C_0 \to C_1$:

   Integrate from $t = t_0$ to t_1 the $r_1 \times \cdots \times r_d$ tensor differential equation
$$\dot{C}(t) = F\left(t, C(t) \bigtimes_{i=1}^d \mathbf{U}_i^1\right) \bigtimes_{i=1}^d \mathbf{U}_i^{1,\top}, \quad C(t_0) = C_0 \bigtimes_{i=1}^d \mathbf{M}_i$$
   and set $C_1 = C(t_1)$.

To continue in time, we take Y_1 as starting value for the next step and perform another step of the integrator.

We observe that, in contrast to the Tucker integrators of [17,14], the factors $\mathbf{U}_i \in \mathbb{R}^{n_i \times r_i}$ are updated simultaneously for $i = 1, \dots, d$.

5.2 Exactness property

The following result extends the exactness results of Theorem 3 and [17, Theorem 4.1] to the new Tucker tensor integrator.

Theorem 6 (Exactness property) *Let $A(t) = C(t) \bigtimes_{i=1}^d \mathbf{U}_i(t)$ be of multilinear rank $(r_1, \dots, r_d)$ for $t_0 \leq t \leq t_1$. Moreover, assume that the $r_i \times r_i$ matrix $\mathbf{U}_i(t_1)^\top \mathbf{U}_i(t_0)$ is invertible for each $i = 1, \dots, d$. With $Y_0 = A(t_0)$, the new Tucker integrator with rank $(r_1, \dots, r_d)$ for $\dot{Y}(t) = \mathrm{P}(Y(t)) \dot{A}(t)$ with starting value $Y_0 = A(t_0)$ is then exact: $Y_1 = A(t_1)$.*

Proof For each $i = 1, \dots, d$, we apply Lemma 1 to $\mathbf{Mat}_i(\dot{A}(t))$
$$\mathbf{Mat}_i(A(t_1) \times_i \mathbf{U}_i^1 \mathbf{U}_i^{1,\top}) = \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} \mathbf{Mat}_i(A(t_1)) = \mathbf{Mat}_i(A(t_1)).$$
We tensorize in the i-th mode and obtain
$$A(t_1) \times_i \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} = A(t_1), \qquad i = 1, \dots, d\ .$$

With $Y_0 = A(t_0)$ we obtain from the second substep of the algorithm

$$\begin{aligned} Y_1 &= C_1 \bigtimes_{i=1}^d \mathbf{U}_i^1 \\ &= \Big(Y_0 \bigtimes_{i=1}^d \mathbf{U}_i^{1,\top} + (A(t_1) - A(t_0)) \bigtimes_{i=1}^d \mathbf{U}_i^{1,\top} \Big) \bigtimes_{i=1}^d \mathbf{U}_i^1 \\ &= \Big(A(t_1) \bigtimes_{i=1}^d \mathbf{U}_i^{1,\top} \Big) \bigtimes_{i=1}^d \mathbf{U}_i^1 \\ &= A(t_1) \bigtimes_{i=1}^d \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} = A(t_1)\,, \end{aligned}$$

which proves the exactness. □

5.3 Robust error bound

The robust error bounds from Theorem 4 and [17, Theorem 5.1] extend to the new Tucker tensor integrator as follows. The norm $\|B\|$ of a tensor B used here is the Euclidean norm of the vector of entries of B.

Theorem 7 (Robust error bound) *Let $A(t)$ denote the solution of the tensor differential equation* (9)*. Assume the following:*

1. *F is Lipschitz-continuous and bounded.*
2. *The non-tangential part of $F(t, Y)$ is ε-small:*

$$\|(I - \mathrm{P}(Y))F(t, Y)\| \le \varepsilon$$

for all Y of multilinear rank $(r_1, \ldots, r_d)$ in a neighbourhood of $A(t)$ and $0 \le t \le T$.

3. *The error in the initial value is δ-small:*

$$\|Y_0 - A_0\| \le \delta.$$

Let Y_n denote the approximation of multinear rank $(r_1, \ldots, r_d)$ to $A(t_n)$ at $t_n = nh$ obtained after n steps of the new Tucker integrator with step-size $h > 0$. Then, the error satisfies for all n with $t_n = nh \le T$

$$\|Y_n - A(t_n)\| \le c_0\delta + c_1\varepsilon + c_2 h,$$

where the constants c_i only depend on the Lipschitz constant L and bound B of F, on T, and on the dimension d. In particular, the constants are independent of singular values of matricizations of the exact or approximate solution.

The proof of Theorem 7 proceeds similar to the proof of Theorem 4 for the matrix case. We begin with two key lemmas and are then in a position to analyse the local error produced after one time step. We denote the solution value at t_1 by A_1. The basis matrix computed in the first part of the integrator is denoted by $\mathbf{U}_i^1$ for each $i = 1, \ldots, d$.

Lemma 5 *For each $i = 1, \ldots, d$, the function $\mathbf{F}_i(t, \cdot) := \mathbf{Mat}_i \circ F(t, \cdot) \circ Ten_i$ fulfills Conditions* 1. - 2. *of Theorem* 2*, and the initial matrix $\mathbf{Y}_{(i)}^0 = \mathbf{Mat}_i(Y_0)$ fulfills Condition* 3. *of that theorem.*

Proof For each $i = 1 \dots d$, it holds that for $\mathbf{Y}_{(i)} = \mathbf{Mat}_i(Y)$,

$$\|\mathbf{F}_i(t, \mathbf{Y}_{(i)})\| = \|F(t, Y)\|.$$

The boundedness and Lipschitz condition of the matrix-valued function $\mathbf{F}_i$ follows from the boundedness and Lipschitz condition of the tensor-valued function F.

Condition 2. follows with the help of the correspondingly defined projection

$$\mathrm{P}_i(\mathbf{Y}_{(i)}) := \mathbf{Mat}_i \circ \mathrm{P}(Y) \circ \mathit{Ten}_i \quad \text{ for } \quad \mathbf{Y}_{(i)} = \mathbf{Mat}_i(Y),$$

which is an orthogonal projection onto a subspace of the tangent space at $\mathbf{Y}_{(i)}$ of the manifold of rank-r_i matrices of dimension $n_i \times n_{\neg i}$. Denoting the orthogonal projection onto this tangent space by $\mathrm{P}_{(i)}(\mathbf{Y}_{(i)})$, we thus have

$$\|\big(\mathbf{I} - \mathrm{P}_{(i)}(\mathbf{Y}_{(i)})\big)\mathbf{F}_i(t, \mathbf{Y}_{(i)})\| \le \|\big(\mathbf{I} - \mathrm{P}_i(\mathbf{Y}_{(i)})\big)\mathbf{F}_i(t, \mathbf{Y}_{(i)})\| = \|(\mathbf{I} - \mathrm{P}(Y))F(t, Y)\| \le \varepsilon.$$

Condition 3. holds due to the invariance of the Frobenius norm under matricization,

$$\|\mathbf{Y}^0_{(i)} - \mathbf{Mat}_i(A_0)\| = \|\mathbf{Mat}_i(Y_0 - A_0)\| = \|Y_0 - A_0\| \le \delta,$$

and so we obtain the stated result. □

Lemma 6 *The following estimate holds with ϑ of* (5)*:*

$$\|A_1 \bigtimes_{i=1}^d \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\| \le d\,\vartheta,$$

where c only depends on d and a bound for hL.

Proof From Lemma 5 and Lemma 2,

$$\|\mathbf{U}_i^1 \mathbf{U}_i^{1,\top} \mathbf{Mat}_i(A_1) - \mathbf{Mat}_i(A_1)\| \le \vartheta, \qquad i = 1, \dots, d.$$

The norm is invariant under tensorization and so the bound is equivalent to

$$\|A_1 \times_i \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\| \le \vartheta, \qquad i = 1, \dots, d.$$

To conclude, we observe

$$\begin{aligned}
&\|A_1 \bigtimes_{i=1}^d \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\| \\
&\le \|A_1 \bigtimes_{i=1}^d \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1 \bigtimes_{i=1}^{d-1} \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} + A_1 \bigtimes_{i=1}^{d-1} \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\| \\
&\le \|A_1 \bigtimes_{i=1}^d \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1 \bigtimes_{i=1}^{d-1} \mathbf{U}_i^1 \mathbf{U}_i^{1,\top}\| + \|A_1 \bigtimes_{i=1}^{d-1} \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\| \\
&\le \|(A_1 \times_d \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1) \bigtimes_{i=1}^{d-1} \mathbf{U}_i^1 \mathbf{U}_i^{1,\top}\| + \|A_1 \bigtimes_{i=1}^{d-1} \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\| \\
&\le \|A_1 \times_d \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\| + \|A_1 \bigtimes_{i=1}^{d-1} \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\| \\
&\le \vartheta + \|A_1 \bigtimes_{i=1}^{d-1} \mathbf{U}_i^1 \mathbf{U}_i^{1,\top} - A_1\|,
\end{aligned}$$

and the result follows by an iteration of this argument. □

We are now in a position to analyse the local error produced after one time step of the integrator.

Lemma 7 (Local error) *If $A_0 = Y_0$, the following local error bound holds for the new Tucker tensor integrator:*

$$\|Y_1 - A_1\| \le \hat{c}\, h(BLh + \varepsilon),$$

where $\hat{c}$ only depends on d and a bound of hL. In particular, the constant is independent of singular values of the exact or approximate solution.

We omit the proof because, up to minor modifications analogous to those in the proof of Lemma 4, the result follows as in [4, Section 5.3] on using the two previous lemmas.

Using the Lipschitz continuity of the function F, we pass from the local to the global errors by the standard argument of Lady Windermere's fan [8, Section II.3] and thus conclude the proof of Theorem 7.

5.4 Symmetric and anti-symmetric low-rank Tucker tensors

For permutations $\sigma \in S_d$, we use the notation $\sigma(Y) = \big(y_{i_{\sigma(1)},\dots,i_{\sigma(d)}}\big)$ for tensors $Y = (y_{i_1,\dots,i_d}) \in \mathbb{R}^{n\times\cdots\times n}$ of order d. A tensor Y is called *symmetric* if $\sigma(Y) = Y$ for all $\sigma \in S_d$, and is called *anti-symmetric* if $\sigma(Y) = (-1)^{\mathrm{sign}(\sigma)}\, Y$ for all $\sigma \in S_d$.

We now assume that the right-hand side function in (9) is such that one of the following conditions holds: For all permutations $\sigma \in S_d$ and all tensors $Y \in \mathbb{R}^{n\times\cdots\times n}$ of order d,

$$\sigma\big(F(t, \sigma(Y))\big) = F(t, Y) \tag{12}$$

or

$$\sigma\big(F(t, \sigma(Y))\big) = (-1)^{\mathrm{sign}(\sigma)}\, F(t, Y) \tag{13}$$

Under these conditions, solutions to (1) with symmetric or anti-symmetric initial data remain symmetric or anti-symmetric, respectively, for all times. We also have preservation of (anti-)symmetry for the new integrator, which does not hold for the projector-splitting integrator.

Theorem 8 *Let Y_0 be symmetric or anti-symmetric and assume that the function F satisfies property* (12) *or* (13)*, respectively. Then, the approximation Y_1 obtained after one time step of the new integrator is symmetric or anti-symmetric, respectively.*

The simple proof is similar to the matrix case and is therefore omitted.

Under condition (12) or (13), the new integrator coincides with the (anti)-symmetry preserving low-rank Tucker tensor integrator of [4].

6 Numerical Experiments

In this section, we present results of different numerical experiments. The experiments were done using Matlab R2017a software with TensorLab package v3.0 [18].

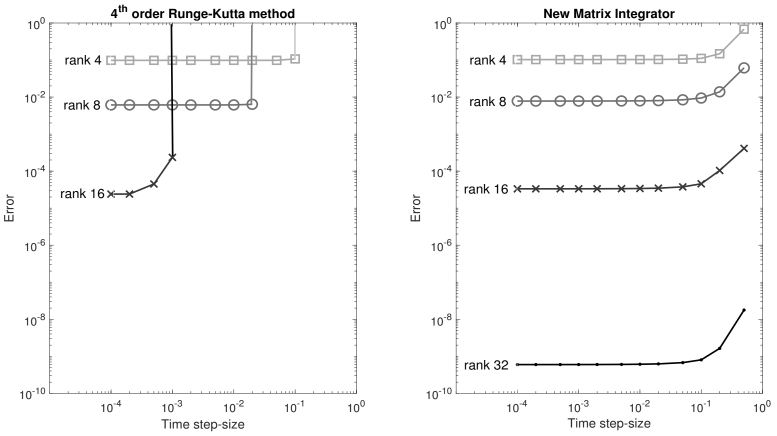


Fig. 1 Comparison of the explicit Runge Kutta method (left) and the proposed new integrator (right) for different approximation ranks and step sizes in the case of a given time-dependent matrix.

6.1 Robustness with respect to small singular values

In the first example, the time-dependent matrix is given explicitly as

$$\mathbf{A}(t) = \left(e^{t\mathbf{W}_1}\right)e^t\mathbf{D}\left(e^{t\mathbf{W}_2}\right)^\top, \quad 0 \le t \le 1 \ .$$

The matrix $\mathbf{D} \in \mathbb{R}^{N\times N}$ is diagonal with entries $d_j = 2^{-j}$. The matrices $\mathbf{W}_1 \in \mathbb{R}^{N\times N}$ and $\mathbf{W}_2 \in \mathbb{R}^{N\times N}$ are skew-symmetric and randomly generated. We note that $e^t 2^{-j}$ are the singular values of $A(t)$. We choose $N = 100$ and final time $T = 1$. We compare the new low-rank integrator presented in Section 3 with a numerical solution obtained with the classical fourth-order explicit Runge-Kutta method applied to the system of differential equations for dynamical low-rank approximation as derived in [11].

The numerical results for different ranks are shown in Figure 1. In contrast to the Runge–Kutta method, the new low-rank integrator does not require a step-size restriction in the presence of small singular values. The same favourable behaviour was shown for the projector-splitting integrator in [10].

6.2 Error behaviour

In the second example, we integrate a (non-stiff) discrete Schrödinger equation in imaginary time,

$$\dot{Y} = -\mathrm{H}[Y], \quad Y(t_0) = C_0 \bigtimes_{i=1}^d \mathbf{U}_i^0 \ .$$

Here,

$$\begin{aligned}
&\mathrm{H}[Y] = -\frac{1}{2}\sum_{j=1}^{d}\left(Y \times_j \mathbf{D}\right) + Y \bigtimes_{i=1}^d \mathbf{V}_{cos} \in \mathbb{R}^{N\times\cdots\times N},\\
&\mathbf{D} = \mathtt{tridiag}(-1,2,-1) \in \mathbb{R}^{N\times N},\\
&\mathbf{V}_{\cos} := \mathrm{diag}\{1-\cos(\frac{2\pi j}{N})\}, \quad j = -N/2,\ldots,N/2-1 \ .
\end{aligned}$$

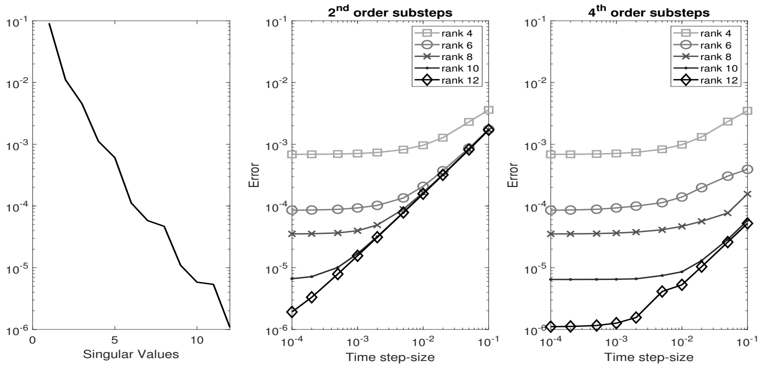


Fig. 2 First twelve singular values of the reference solution at time $T = 0.1$ and approximation errors for different ranks, time-integration methods in the substeps of the new matrix integrator, and step-sizes for the matrix differential equation ($d = 2$).

The function H arises from the Hamiltonian $\mathcal{H} = -\frac{1}{2}\Delta_{\text{discrete}} + V(x)$ on a equidistant space grid with the torsional potential $V(x_1, \ldots, x_d) = \prod_{i=1}^{d}(1 - \cos(x_i))$.

For each $i = 1, \ldots, d$, the orthonormal matrices $\mathbf{U}_i^0 \in \mathbb{R}^{N \times N}$ are randomly generated. The core tensor $C_0 \in \mathbb{R}^{N \times N \times N}$ has only non-zero diagonal elements set equal to $(C_0)_{jjj} = 10^{-j}$ for $j = 1, \ldots N$ in the case $d = 3$, and analogously in the matrix case $d = 2$.

The reference solution was computed with the Matlab solver `ode45` and stringent tolerance parameters {'RelTol', 1e-10, 'AbsTol', 1e-10} . The differential equations appearing in the definition of a step of the new matrix and Tucker integrators have all been solved either with a single step of a second- or fourth-order explicit Runge–Kutta method.

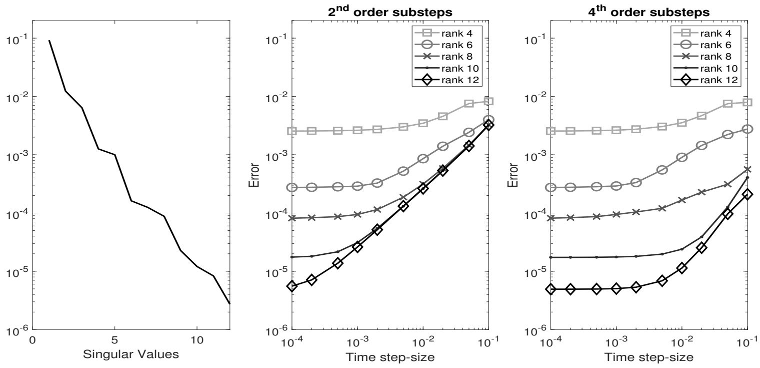


Fig. 3 First twelve singular values of the matricisization in first mode of the reference solution at time $T = 0.1$ and approximation errors for different multi-linear ranks, time-integration methods in the substeps of the new fixed-rank Tucker tensor integrator and step-sizes for the tensor differential equation ($d = 3$).

We choose $N = 100$, final time $T = 0.1$ and $d = 2, 3$. The multi-linear rank is chosen such that $r_1 = r_2 = \cdots = r_d$. The singular values of the matricization in the first mode of the reference solution and the absolute errors $\|Y_n - A(t_n)\|_F$ at time $t_n = T$ of the approximate solutions for different ranks, calculated with different step-sizes and different time integration methods, are shown in Figure 2 for the matrix case($d = 2$), and in Figure 3 for the tensor case($d = 3$). The figures clearly show that solving the substeps with higher accuracy allows us to take larger step-sizes to achieve a prescribed error.

6.3 Comparison with the matrix projector-splitting integrator over different ranks

In the last example, we compare the matrix projector splitting integrator with the new matrix integrator of Section 3. Here, the complex case is considered: in the definition of the sub-problems appearing in the new matrix integrator, it is sufficient to replace the transpose with the conjugate transpose.

We consider a Schrödinger equation as in [10, Section 4.3],

$$\begin{aligned} &i\partial_t u(x,t) = -\frac{1}{2}\Delta u(x,t) + \frac{1}{2}x^\top \mathbf{A} x\, u(x,t), \quad x \in \mathbb{R}^2, t > 0, \\ &u(x,0) = \pi^{-\frac{1}{2}} \exp\Big(\frac{1}{2}x_1^2 + \frac{1}{2}(x_2-1)^2\Big), \\ &\mathbf{A} = \begin{pmatrix} 2 & -1 \\ -1 & 3 \end{pmatrix}. \end{aligned}$$

The problem is discretized with a Fourier collocation method with a grid of $N \times N$ points; the solution is essentially supported within $\Omega = [-7.5, 7.5]^2$. We choose the final time $T = 5$ and $N = 128$, which makes the problem moderately stiff. First, we compute a reference solution with an Arnoldi method and a tiny time-step size $h = 10^{-4}$. Then, for each rank from 1 until 20, we compute a low-rank approximation with the matrix projector splitting integrator and the new matrix integrator. The lower-dimensional sub-problems appearing in the definition of the two integrators are solved with an Arnoldi method and time-step size $h = 0.005$. For each rank, the absolute error in Frobenius norm of the two given approximations at the final time $T = 5$, with respect to the reference solution, are shown in Figure 4.

Acknowledgements The last numerical example is based upon the original source code of [10, Section 4.3]; we would like to thank Hanna Walach for kindly providing it.

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) — Project-ID 258734477 — SFB 1173.

References

1. M. Bachmayr, H. Eisenmann, E. Kieri, and A. Uschmajew. Existence of dynamical low-rank approximations to parabolic problems. *arXiv preprint arXiv:2002.12197*, 2020.
2. M. H. Beck, A. Jäckle, G. A. Worth, and H.-D. Meyer. The multiconfiguration time-dependent Hartree (MCTDH) method: a highly efficient algorithm for propagating wavepackets. *Physics reports*, 324(1):1–105, 2000.
3. M. H. Beck and H.-D. Meyer. An efficient and robust integration scheme for the equations of motion of the multiconfiguration time-dependent Hartree (MCTDH) method. *Z. Physik D*, 42(2):113–129, 1997.

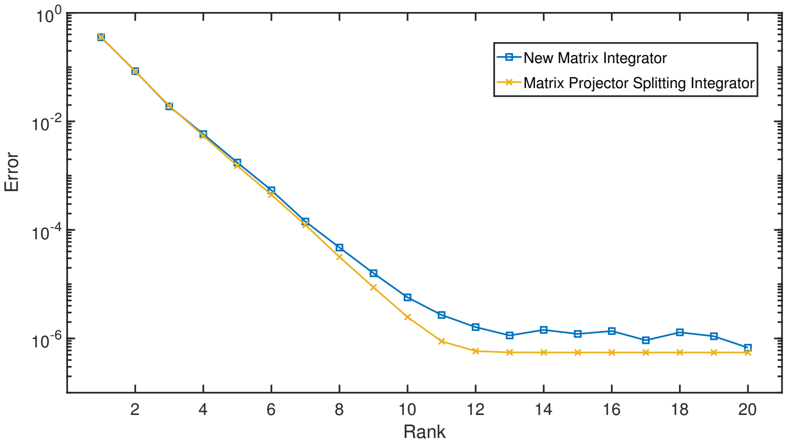


Fig. 4 Approximation errors for different ranks at final time $T = 5$ of the low-rank approximation computed with the matrix projector splitting integrator and the new matrix integrator.

4. G. Ceruti and C. Lubich. Time integration of symmetric and anti-symmetric low-rank matrices and Tucker tensors. *BIT Numer. Math.*, 60:591–614, 2020.
5. G. Ceruti, C. Lubich, and H. Walach. Time integration of tree tensor networks. *arXiv preprint arXiv:2002.11392. To appear in SIAM J. Numer. Anal.*, 2020.
6. L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.
7. J. Haegeman, C. Lubich, I. Oseledets, B. Vandereycken, and F. Verstraete. Unifying time evolution and optimization with matrix product states. *Phys. Rev. B*, 94(16):165116, 2016.
8. E. Hairer, S. P. Nørsett, and G. Wanner. *Solving ordinary differential equations. I. Nonstiff problems*, volume 8 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 1993.
9. B. N. Khoromskij, I. V. Oseledets, and R. Schneider. Efficient time-stepping scheme for dynamics on TT-manifolds. *Preprint 2012-24, MPI Math. Naturwiss. Leipzig*, 2012.
10. E. Kieri, C. Lubich, and H. Walach. Discretized dynamical low-rank approximation in the presence of small singular values. *SIAM J. Numer. Anal.*, 54(2):1020–1038, 2016.
11. O. Koch and C. Lubich. Dynamical low-rank approximation. *SIAM J. Matrix Anal. Appl.*, 29(2):434–454, 2007.
12. O. Koch and C. Lubich. Dynamical tensor approximation. *SIAM J. Matrix Anal. Appl.*, 31(5):2360–2375, 2010.
13. T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Rev.*, 51(3):455–500, 2009.
14. C. Lubich. Time integration in the multiconfiguration time-dependent Hartree method of molecular quantum dynamics. *Appl. Math. Res. Express. AMRX*, 2015(2):311–328, 2015.
15. C. Lubich and I. V. Oseledets. A projector-splitting integrator for dynamical low-rank approximation. *BIT*, 54(1):171–188, 2014.
16. C. Lubich, I. V. Oseledets, and B. Vandereycken. Time integration of tensor trains. *SIAM J. Numer. Anal.*, 53(2):917–941, 2015.
17. C. Lubich, B. Vandereycken, and H. Walach. Time integration of rank-constrained Tucker tensors. *SIAM J. Numer. Anal.*, 56(3):1273–1290, 2018.
18. N. Vervliet, O. Debals, L. Sorber, M. Van Barel, and L. De Lathauwer. Tensorlab 3.0. *available online, URL: www. tensorlab. net*, 2016.