arXiv:2001.08599v1 [math.NA] 23 Jan 2020

A geometry based algorithm for dynamical low-rank approximation

M. Billaud-Friess, A. Falcó & A. Nouy

January 24, 2020

Abstract

In this paper, we propose a geometry based algorithm for dynamical low-rank approximation on the manifold of fixed rank matrices. We first introduce a suitable geometric description of the set of fixed rank matrices which relies on a natural parametrization of matrices. More precisely, it is endowed with the structure of analytic principal bundle, with an explicit description of local charts. For matrix differential equations, we introduce a first order numerical integrator working in local coordinates. The resulting chart based algorithm can be interpreted as a particular splitting of the projection operator onto the tangent space of the low-rank matrix manifold. It is proven to be robust and exact in some particular case. Numerical experiments confirm these results and illustrate the behavior of the proposed algorithm.

Keywords: Dynamical low-rank approximation, matrix manifold, matrix differential equation, splitting integrator

2010 AMS Subject Classifications: 15A23, 65F30, 65L05, 65L20

1 Introduction

High-dimensional dynamical systems arise in variety of applications as quantum chemistry, physics, finance and uncertainty quantification, to name a few. Discretization of such problems with traditional numerical methods often leads to complex numerical problems usually untractable, in particular if they depend on parameters. Model Order Reduction (MOR) methods aim at reducing the complexity of such problems by projecting the solution onto low-dimensional manifolds. In this paper, we particularly focus on dynamical low-rank methods. Such methods have been considered for low-rank approximation of time-dependent matrices [9, 5] with possible symmetry properties [3], and tensors [8, 10, 6, 7]. In the context of parameter-dependent partial differential equations, let us mention also dynamical orthogonal approximation in the lines of [12, 4, 11] and also dynamical reduced basis method [2].

Here, we focus on low-rank approximation of time-dependent matrices $A(t) \in \mathbb{R}^{n\times m}$. Introducing $\dot{A}(t) = \frac{d}{dt}A(t)$ the time derivative, the matrix $A(t)$ is defined as the solution of the following Ordinary Differential Equation (ODE)

$$\dot{A}(t) = F(A(t), t), \qquad A(0) = A^0, \tag{1}$$

given $A^0 \in \mathbb{R}^{n\times m}$ and $F : \mathbb{R}^{n\times m} \times [0, T] \to \mathbb{R}^{n\times m}$. Dynamical low-rank methods aim at approximating at each instant t the matrix $A(t)$ by the matrix $Z(t)$ which belongs to the nonlinear

manifold of fixed rank matrices

$$\mathcal{M}_r(\mathbb{R}^{n\times m}) = \{Z \in \mathbb{R}^{n\times m} : \text{rank}(Z) = r\},$$

where $r \ll \min(n, m)$ stands for the rank. When $A(t)$ is known, $Z(t)$ can be defined as the best rank-r approximation solution of

$$Z(t) = \arg \min_{W\in\mathcal{M}_r(\mathbb{R}^{n\times m})} \|A(t) - W\|, \tag{2}$$

with $\|\cdot\|$ the Frobenius norm. In that case, Z is obtained through a Singular Value Decomposition (SVD) of $A(t)$ for each instant t. Nevertheless, as A is implicitly given by the dynamical system (1), it is more relevant to introduce low-rank approximation using $\dot{A}$. To that goal, the approximation Z is classically obtained through its derivative $\dot{Z}$ which satisfies the Dirac-Frenkel variational principle

$$\dot{Z}(t) = \arg \min_{\delta W\in T_{Z(t)}\mathcal{M}_r(\mathbb{R}^{n\times m})} \|\delta W - F(Z(t), t)\|, \tag{3}$$

given $Z(0) = Z^0 \in \mathcal{M}_r(\mathbb{R}^{n\times m})$ the best rank-r approximation of $A(0)$ and $T_{Z(t)}\mathcal{M}_r(\mathbb{R}^{n\times m})$ the tangent space to $\mathcal{M}_r(\mathbb{R}^{n\times m})$ at $Z(t)$. Equivalently, $\dot{Z}(t)$ corresponds to the orthogonal projection of $F(Z(t), t)$ (see Figure 1) on the solution dependent tangent space, i.e.

$$\dot{Z}(t) = P_{T_Z(t)}F(Z(t), t), \qquad Z(0) = Z^0, \tag{4}$$

where P_{T_Z} denotes the projector onto $T_{Z(t)}\mathcal{M}_r(\mathbb{R}^{n\times m})$.

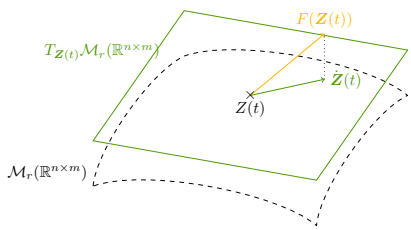


Figure 1: Orthogonal projection on the tangent space $T_{Z(t)}\mathcal{M}_r(\mathbb{R}^{n\times m})$.

In view of MOR, the goal of low-rank methods is to approximate the solution $A(t)$ of Equation (1) with $Z(t)$ solution of Equation (4) which is cheaper to compute. However, in practice additional difficulties appear for the numerical integration of Equation (4).
The first difficulty relies on the proper description of the manifold of fixed rank matrices $\mathcal{M}_r(\mathbb{R}^{n\times m})$. In practice, a way to compute the rank-r matrix $Z(t)$ is done through its parametrization

$$Z(t) = U(t)G(t)V(t)^T, \tag{5}$$

with $U(t) \in \mathbb{R}^{n\times r}, V(t) \in \mathbb{R}^{m\times r}$ and $G(t) \in \mathbb{R}^{r\times r}$. Such a parametrization of the matrix $Z(t)$ is not unique. A way to dodge this undesirable property is to properly define the tangent space $T_{Z(t)}\mathcal{M}_r(\mathbb{R}^{n\times m})$. Given $Z(0) = U(0)G(0)V(0)^T$, the matrix $Z(t)$ admits a unique decomposition of the form (5) when imposing the so-called *gauge conditions* on U, V (see [9, Proposition 2.1]). In addition, the system (4) results in a system ODEs driving the evolution of the parameters U, G, V.

The second difficulty appears when numerical integration is performed for solving the resulting system of ODEs governing the evolution of parameters U, V and G. Indeed in presence of small singular values for $Z(t)$, the matrix $G(t)$ may be ill-conditioned. As consequence, classical integration schemes may be unstable (see e.g. [6, Section 2.1]). Moreover, in case of *overapproximation*, i.e. when the approximation $Z(t)$ has a rank r greater than the rank of the exact solution $A(t)$, these method fails since $G(t)$ becomes singular. A possible way to adress this issue is regularize G such that it remains invertible. Nevertheless, it has the inconvenient to modify the problem (then the solution) and does not prevent ill-conditioning of $G(t)$. In [10] an explicit projector-splitting integrator is proposed to deal with numerical integration of (4). It is based on a Lie-Trotter splitting of the projection operator $P_{T_{Z(t)}}$. In addition to its simplicity, it has the advantage to remains robust in case of small singular values and especially for overapproximation as it avoids the inversion of $G(t)$. The resulting method is fully explicit and first order, extension to second order via Strang splitting scheme could be considered. When interested in higher order approximation, projection based methods [7], in the lines of Riemaniann optimization, combined to explicit Runge-Kutta schemes could be considered. Such methods work as follows. Perform one step of the numerical scheme leaving the manifold, and then project on the manifold by means of retraction. The latter step is usually performed using a r-terms truncated SVD.

In [2], the authors give a different geometric description of the fixed-rank matrix manifold and the associated tangent space. This description combines a natural definition of the neighbourhood of Z together with explicit description of local charts such that the set $\mathcal{M}_r(\mathbb{R}^{n\times m})$ is endowed with the structure of analytic principal bundle [2, Theorem 4.1]. Moreover, it ensures that any matrix in the neighborhood of a matrix Z (including itself) admits a unique representation in the form UGV^T. The main contribution of this paper is twofold. First, we revisit dynamical low-rank approximation by using the geometric description of the matrix manifold given in [2]. The resulting system of ODEs on the parameters is shown to be related to the one obtained in [9] but with no need of gauge conditions. Secondly, based on this geometric description of $\mathcal{M}_r(\mathbb{R}^{n\times m})$, we propose a first order numerical integrator in local coordinates for solving (4) that can be interpreted as a splitting integrator. It is proven to coincide with the so-called *KSL splitting algorithm* introduced in [10] in the particular case where the flux F only depends on t.

The outline of the paper is as follows. We detail in Section 2 the proposed geometric description of the manifold of rank-r matrices. In Section 3 we describe a new splitting algorithm relying on the proposed geometric description Finally, in Section 4, we confront the proposed splitting algorithm to KSL splitting integrator [10] on several numerical test cases.

2 Dynamical low-rank approximation: a geometric approach

Dynamical low-rank approximation consists in approximating at each time t the matrix $A(t)$ by a matrix $Z(t) \in \mathcal{M}_r(\mathbb{R}^{n\times m})$ that can be represented (in non-unique way) by means of the factorization

$$Z = UGV^T,$$

with $U \in \mathcal{M}_r(\mathbb{R}^{n\times r}), V \in \mathcal{M}_r(\mathbb{R}^{m\times r})$ and $G \in \mathrm{GL}_r$. We present in Section 2.1 a geometric description of $\mathcal{M}_r(\mathbb{R}^{n\times m})$. We restrict the presentation to essential elements of geometry required thereafter. The interested reader could consult the original paper [2] for further details. In Section 2.2 we discuss the interest of the proposed description for dynamical low-rank approximation. Finally, we draw the link with the geometric description proposed in [9] in Section 2.3.

2.1 Chart based geometric description of $\mathcal{M}_r(\mathbb{R}^{n\times m})$

Let $Z = UGV^T \in \mathcal{M}_r(\mathbb{R}^{n\times m})$. We consider $U_\perp \in \mathcal{M}_{n-r}(\mathbb{R}^{n\times(n-r)}), V_\perp \in \mathcal{M}_{m-r}(\mathbb{R}^{n\times(m-r)})$, the matrices such that $U_\perp^T U = 0$ and $V_\perp^T V = 0$. The *neighbourhood* $\mathcal{U}_Z$ of Z in $\mathcal{M}_r(\mathbb{R}^{n\times m})$ is defined as the set

$$\mathcal{U}_Z = \{(U + U_\perp X)H(V + V_\perp Y)^T : (X, Y, H) \in \mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathrm{GL}_r\}.$$

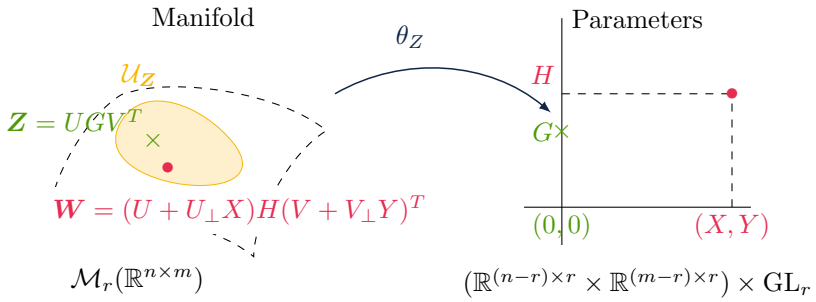


Figure 2: Representation of the local chart θ_Z that associates to $W = (U + U_\perp X)H(V + V_\perp Y)^T$ in $\mathcal{U}_Z \subset \mathcal{M}_r(\mathbb{R}^{n\times m})$ the parameters $(X, Y, H) \in \mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathrm{GL}_r$.

We associate to the neighborhood $\mathcal{U}_Z$ of Z the *local chart* $\theta_Z : \mathcal{U}_Z \to \mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathrm{GL}_r$ (see Figure 2) which is given by

$$\theta_Z(W) = (U_\perp^+ W(V^+)^T(U^+W(V^+)^T)^{-1}, V_\perp^+ W^T(U^+)^T(V^+W^T(U^+)^T)^{-1}, U^+W(V^+)^T)$$

for any $W \in \mathcal{U}_Z$. Here U^+ and V^+ stand for the Moore-Penrose pseudo-inverses [1] of U and V respectively. This means that any matrix W belonging to the neighbourhood $\mathcal{U}_Z$ admits a unique parametrization

$$W = \theta_Z^{-1}(X, Y, H),$$

with parameters $(X, Y, H) \in \mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathrm{GL}_r$ and where the map θ_Z^{-1} is defined by

$$\theta_Z^{-1}(X, Y, H) = (U + U_\perp X)H(V + V_\perp Y)^T.$$

In this description, the parameters are not longer U, V, G but X, Y, H. Note that $\theta_Z^{-1}(0, 0, G) = Z$.

Such geometric description confers the set $\mathcal{M}_r(\mathbb{R}^{n\times m})$ the structure of an analytic manifold and of a principal bundle [2, §4].

Proposition 2.1. *The set of fixed rank matrices $\mathcal{M}_r(\mathbb{R}^{n\times m})$ equipped with the atlas $\mathcal{A}_{n,m,r} = \{(\mathcal{U}_Z, \theta_Z) : Z \in \mathcal{M}_r(\mathbb{R}^{n\times m})\}$ is an analytic $r(n + m - r)$-dimensional manifold modelled on $\mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathbb{R}^{r\times r}$. Moreover $\mathcal{M}_r(\mathbb{R}^{n\times m})$ is an analytic principal bundle with typical fiber GL_r and base[2] $\mathbb{G}_r(\mathbb{R}^n) \times \mathbb{G}_r(\mathbb{R}^m)$.*

We now give a description of the tangent space to the manifold of rank-r matrices at Z denoted $T_Z\mathcal{M}_r(\mathbb{R}^{n\times m})$.

[1] For any $A \in \mathbb{R}^{n\times m}$, the Moore-Penrose pseudo inverse is given by $A = (A^TA)^{-1}A^T$.
[2] Here $\mathbb{G}_r(\mathbb{R}^p) = \{V_r \subset \mathbb{R}^p : \dim(V_r) = r\}$ denotes the Grassmann manifold.

To that goal, we define the tangent map at $Z \in \mathcal{M}_r(\mathbb{R}^{n\times m})$ noted $\mathrm{T}_Z i$ by

$$\begin{aligned}\mathrm{T}_Z i : \mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathbb{R}^{r\times r} &\to \mathbb{R}^{n\times m},\\ (\delta X, \delta Y, \delta H) &\mapsto U_\perp \delta X G V^T + U G (V_\perp \delta Y)^T + U \delta H V^T.\end{aligned}$$

Then, the tangent space to $\mathcal{M}_r(\mathbb{R}^{n\times m})$ at Z is defined as the image through $\mathrm{T}_Z i$ of the tangent space in the local coordinates [3] in $\mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathbb{R}^{r\times r}$

$$T_Z\mathcal{M}_r(\mathbb{R}^{n\times m}) = \{U_\perp \delta X G V^T + U G (V_\perp \delta Y)^T + U \delta G V^T : \delta X \in \mathbb{R}^{(n-r)\times r}, \delta Y \in \mathbb{R}^{(m-r)\times r}, \delta G \in \mathbb{R}^{r\times r}\}.$$

As stated in [2, Proposition 4.3], $\mathrm{T}_Z i$ is an isomorphism between $T_Z\mathcal{M}_r(\mathbb{R}^{n\times m})$ and $\mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathbb{R}^{r\times r}$.

Proposition 2.2. *The tangent map $\mathrm{T}_Z i$ at Z is a linear isomorphism with inverse $(\mathrm{T}_Z i)^{-1}$ given by*

$$(\mathrm{T}_Z i)^{-1}(\delta W) = (U_\perp^+ \delta W (V^+)^T G^{-1}, V_\perp^+ \delta W^T (U^+)^T G^{-T}, U^+ \delta W (V^+)^T)$$

for $\delta W \in \mathbb{R}^{n\times m}$.

By Proposition 2.2, any tangent matrix $\delta Z \in T_Z\mathcal{M}_r(\mathbb{R}^{n\times m})$ admits a unique parametrization of the form

$$\delta Z = \mathrm{T}_Z i(\delta X, \delta Y, \delta H) = U_\perp \delta X G V^T + U G (V_\perp \delta Y)^T + U \delta H V^T, \tag{6}$$

where $(\delta X, \delta Y, \delta H) \in \mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathbb{R}^{r\times r}$ are uniquely given through

$$\begin{array}{rcl}\delta X &=& U_\perp^+ \delta Z (V^+)^T G^{-1},\\ \delta Y &=& V_\perp^+ \delta Z^T (U^+)^T G^{-T},\\ \delta H &=& U^+ \delta Z (V^+)^T.\end{array} \tag{7}$$

Remark 2.3. *The tangent space could be decomposed into distinct pieces which are the vertical tangent space*

$$T_Z^V \mathcal{M}_r(\mathbb{R}^{n\times m}) = \{U \delta G V^T : \delta G \in \mathbb{R}^{r\times r}\},$$

and the horizontal tangent space

$$T_Z^H \mathcal{M}_r(\mathbb{R}^{n\times m}) = \{U_\perp \delta X G V^T + U G (V_\perp \delta Y)^T : \delta X \in \mathbb{R}^{(n-r)\times r}, \delta Y \in \mathbb{R}^{(m-r)\times r}\},$$

where $T_Z^V \mathcal{M}_r(\mathbb{R}^{n\times m})$ is associated to the fiber and $T_Z^H \mathcal{M}_r(\mathbb{R}^{n\times m})$ to the base.

2.2 Dynamical low-rank approximation

In the context of dynamical low-rank approximation, we recall that Z is given through the projected Equation (4). By definition of the tangent space (6), the tangent matrix is given by $\dot{Z} = \mathrm{T}_Z i(\dot{X}, \dot{Y}, \dot{H})$ where the parameters satisfies

$$(\dot{X}, \dot{Y}, \dot{H}) = \mathrm{T}_Z i^{-1}\left(P_{T_Z} F(Z)\right) \tag{8}$$

with

$$P_{T_Z} F(Z) = P_U^\perp F(Z) P_V^T + P_U^T F(Z) (P_V^\perp)^T + P_U F(Z) P_V^T.$$

[3] This means the tangent space to the local parameter space $\mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathbb{R}^{r\times r}$ at $(0, 0, G)$.

Here $P_U = UU^+, P_V = VV^+$ denote the projectors associated to U, V respectively, and their related orthogonal projectors $P_U^\perp = I - P_U$, $P_V^\perp = I - P_V$. Equation (8) yields equivalently to the following system of ODEs on the parameters:

$$\begin{array}{rcl} \dot{X} & = & U_\perp^+ F(Z)(V^+)^T G^{-1}, \\ \dot{Y} & = & V_\perp^+ F(Z)^T (U^+)^T G^{-T}, \\ \dot{H} & = & U^+ F(Z)(V^+)^T. \end{array} \tag{9}$$

The proposed geometrical description ensures that (8) admits a unique maximal solution Z when the original problem is an autonomous dynamical system with *vector field* F [5, Theorem 2.6, Theorem 3.5]

Proposition 2.4. *Let consider the analytical manifold* $(\mathcal{M}_r(\mathbb{R}^{n\times m}), \mathcal{A}_{n,m,r})$ *and assume* $F : \mathbb{R}^{n\times m} \to \mathbb{R}^{n\times m}$ *is a* $\mathcal{C}^p$ *vector field. Then* (1) *admits a unique maximal solution. Moreover,* $\mathrm{T}_Z i^{-1} P_{T_Z} F : \mathbb{R}^{n\times m} \to \mathbb{R}^{(n-r)\times r} \times \mathbb{R}^{(m-r)\times r} \times \mathbb{R}^{r\times r}$ *is a* $\mathcal{C}^p$ *vector field which ensures that* (8) *is a dynamical system which remains well posed.*

2.3 Link with the geometric description introduced in [9].

In this section, we discuss the relation between the proposed geometric approach and the description introduced in [9] . In the latter, the non-uniqueness of the parametrization $Z = UGV^T$ is avoided by computing the tangent matrix δZ in $T_Z\mathcal{M}_r(\mathbb{R}^{n\times m})$. Introducing

$$\delta Z = \delta U G V^T + U \delta G V^T + U G \delta V^T \tag{10}$$

together with the *gauge conditions*

$$U^T \delta U = 0 \text{ and } V^T \delta V = 0, \tag{11}$$

and assuming that U, V are orthogonal, then the parameter derivatives $\delta U, \delta G, \delta V$ are uniquely given [9, Proposition 2.1] by

$$\begin{array}{rcl} \delta U & = & (I - UU^T)\delta Z V G^{-1}, \\ \delta V & = & (I - VV^T)\delta Z^T U G^{-T}, \\ \delta G & = & U^T \delta Z V. \end{array} \tag{12}$$

This can be interpreted as constructing an isomorphism between

$$T_Z\mathcal{M}_r(\mathbb{R}^{n\times m}) \quad \text{and } \{(\delta G, \delta U, \delta V) \in \mathbb{R}^{n\times r} \times \mathbb{R}^{m\times r} \times \mathbb{R}^{r\times r} : U^T\delta U = 0, V^T\delta V = 0\}.$$

The geometric description proposed in Section 2.1 allows to recover (10)-(12). Indeed, by setting $(\delta U, \delta V, \delta G) = (U_\perp \delta X, V_\perp \delta Z, \delta H)$ in the definition (6) of the tangent matrix δZ, we get (10). Moreover, the gauge conditions are naturally satisfied as $U^T\delta U = U^T U_\perp \delta X = 0, V^T \delta V = V^T V_\perp \delta Z = 0$. Finally, for U, V orthogonal we have $UU^+ = UU^T, VV^+ = VV^T$ and by multiplying the first and second equations of (7) by $U_\perp$ and $V_\perp$ respectively we recover (12).

Going back to dynamical low-rank approximation, when Z corresponds to the rank-r approximation of a matrix A through Equation (4), we get from (12) the following system of ODEs governing the evolution of the factors U, G, V,

$$\begin{array}{rcl} \dot{U} & = & (I - UU^T)F(Z)VG^{-1}, \\ \dot{V} & = & (I - VV^T)F(Z)^T UG^{-T}, \\ \dot{G} & = & U^T F(Z) V. \end{array} \tag{13}$$

Again, this system can be deduced from (9) by setting $(\dot{U}, \dot{V}, \dot{G}) = (U_\perp \dot{X}, V_\perp \dot{Z}, \dot{H})$ and assuming that U, V orthogonal.

3 Projector splitting integrator schemes

In this section, we derive suitable schemes for numerical integration of the projected equation (4). Two splitting methods are presented, first in an abstract semi-discretized framework in Section 3.1 and then in their practical form in Section 3.2. The relation between these two algorithms is discussed in Section 3.3 .

3.1 Splitting integrators

3.1.1 Symmetric splitting method

We first consider the setting of the classical description [9] detailed in Section 2.3. To perform time integration, a symmetric Lie-Trotter splitting method [10] is applied to Equation (3). This integration scheme relies on a decomposition of the projector $P_{T_Z(t)}$ in three terms denoted $P_i, 1 \leq i \leq 3$, as follows

$$P_{T_Z(t)} = \underbrace{AP_V^T}_{P_1(A)} - \underbrace{P_U A P_V^T}_{P_2(A)} + \underbrace{P_U A}_{P_3(A)}. \tag{14}$$

Using this splitting, one integration step from t_0 to $t_1 = t_0 + \Delta t$ starting from the factorized rank-r matrix Z_0 under the form $Z_0 = U_0 G_0 V_0^T$ reads as follows.

1. Let $K = UG$. Integrate on $[t_0, t_1]$ the $n \times r$ matrix differential equation

$$\frac{d}{dt}(UG)(t) = F(K(t)V_0^T, t)V_0, \quad (UG)(t^0) = U_0 G_0.$$

   Set $U_1 G_1 = (UG)(t^1)$ by QR factorization and $V_1 = V_0$.

2. Integrate on $[t_0, t_1]$ the $r \times r$ matrix differential equation

$$\dot{G}(t) = -U_1^T F(U_1 G(t) V_1^T, t)V_1, \quad G(t^0) = G_1.$$

   Set $G_2 = G(t_1)$ and $U_2 = U_1, V_2 = V_1$.

3. Let $L = VG^T$. Integrate on $[t_0, t_1]$ the $m \times r$ matrix differential equation

$$\frac{d}{dt}(VG^T)(t) = F(U_2 L(t)^T, t)^T U_2, \quad (VG^T)(t^0) = V_2 G_2^T.$$

   Set $V_3 G_3^T = (VG^T)(t_1)$ by QR factorization and $U_3 = U_2$.

Each Step i corresponds to the integration of the right hand side of (4) associated to the projector part $P_i, 1 \leq i \leq 3$ (for details see [10, Lemma 3.1]).

Remark 3.1. *This splitting algorithm works in the following order. First, it updates UG, G and then VG. As we will discuss in the Section 3.3, this particular choice allow to recover exactness properties of the splitting scheme in the context of matrix approximation [10].*

Remark 3.2. *The inversion of G is avoided. This convenient choice allow to deal with the case of over-approximation, in i.e. when the rank of the approximation Z is smaller then r.*

3.1.2 Chart based splitting method

The second contribution of this paper is to propose a numerical integrator relying on the geometric description of the manifold of fixed rank matrices proposed in Section 2. The guiding idea is to perform some update first the parameter H and then the parameters (X, Y).

Using the same notations as in Section 3.1.1, we propose to perform one integration step from t_0 to $t_1 = t_0 + \Delta t$ starting from $Z_0 = U_0 G_0 V_0^T$ using the following integration scheme with three steps.

1. Integrate on $[t_0, t_1]$ the $r \times r$ matrix differential equation
$$\dot{H}(t) = U_0^+ F(U_0 H(t) V_0^T, t)(V_0^+)^T, \quad H(t^0) = G_0,$$
Set $U_1 = U_0, V_1 = V_0$ and $G_1 = H(t_1)$.
2. Let $K = U_1 G_1 + U_{1,\perp} X H$. Integrate on $[t_0, t_1]$ the $n \times r$ matrix differential equation
$$\frac{d}{dt}(U_{1,\perp} X H)(t) = P_{U_1}^{\perp} F(K(t) V_1^T, t)(V_1^+)^T, \quad (U_{1,\perp} X H)(t^0) = 0.$$
Set $U_2 G_2 = U_1 G_1 + (U_{1,\perp} X G)(t^1)$ by QR factorization, and $V_2 = V_1$.
3. Let $L = V_2 G_2^T + V_{2,\perp} Y H^T$. Integrate on $[t_0, t_1]$ the $m \times r$ matrix differential equation
$$\frac{d}{dt}(V_{2,\perp} Y H^T)(t) = P_{V_2}^{\perp} F(U_3 L(t)^T, t)(U_2^+)^T, \quad (V_{2,\perp} Y H^T)(t^0) = 0.$$
Set $V_3 G_3^T = V_2 G_2^T + (V_{2,\perp} Y G^T)(t^1)$ by QR factorization, and $U_3 = U_2$.

In the lines of the previous method (see Section 3.1.1), the chart based method can be deduced from a Lie-Trotter splitting that relies on the following decomposition of the projector $P_{T_Z(t)}$ as follows
$$P_{T_{Z(t)}} = \underbrace{P_U A P_V^T}_{P_1(A)} + \underbrace{P_U^{\perp} A P_V^T}_{P_2(A)} + \underbrace{P_U A (P_V^{\perp})^T}_{P_3(A)}, \tag{15}$$
where each term P_i of the projector is associated to the EDO solved at Step i. This point is discussed and detailed in the Appendix A.

Remark 3.3. *At each Step i of the chart based method, we work in the neighbourhood of the previous rank-r approximation noted $Z_{i-1} = U_{i-1} G_{i-1} V_{i-1}^T$. In particular, at Step 2 the matrices U_1, V_1 (together with their orthogonal matrices $V_{1,\perp}, U_{1,\perp}$) are fixed implying that $(U_{1,\perp} X G)(t^0) = 0$. Similarly, at Step 3, we work locally in neighborhood of Z_2 leading to $(V_{2,\perp} Y G^T)(t^0) = 0$.*

Remark 3.4. *As for the splitting method of Section 3.1.1, we avoid the inversion of matrix G which allows to deal with overapproximation case where G is singular. However, the two methods differ by the update order. Indeed, the chart based method first updates G and then U, V (or equivalently H and then X, Y).*

Remark 3.5. *One technical difficulty associated to the chart based method is that it requires the use of matrices $U_{i,\perp}, V_{i,\perp}$. Computing these matrices may be tricky as it is costly to compute them at each Step i, and they are not uniquely defined.*
Here, we introduce them for proper description of neighbourhoods but we never need to computed them explicitly. Indeed, we avoid their use by making appear the operators $P_{U_1}^{\perp}, P_{V_2}^{\perp}$. Moreover, they are only present through the update of the product of matrices $U_{1,\perp} X G$ and $V_{2,\perp} Y G^T$ but they are never explicitly computed.

3.2 Practical algorithms

Now, we provide practical formulation of those methods amenable for numerical use. To that goal, let introduce preliminary notations. We consider a uniform discretization of the time interval $[0, T]$, $T > 0$, containing $K + 1$ nodes $0 = t^0 < t^1 < t^2 < \cdots < t^K$ where $t^k = k\Delta t$ and $\Delta t = \frac{T}{K}$. The matrix Z^k is the approximation at each time step t^k of Z computed on $[t_0, t_1] = [t^k, t^{k+1}]$ through the splitting methods given in Section 3.1. Explicit approximation of the flux is performed leading the schemes summarized in the following algorithms.

We first give the *Chart based algorithm* (see Algorithm 3.6).

Algorithm 3.6 (Chart based algorithm)**.** *Given the initial rank-r approximation $Z^0 = U^0 G^0 (V^0)^T$, compute $Z^k \in \mathcal{M}_r(\mathbb{R}^{n\times m})$ for $k \in \{1, ..., K\}$ as follows.*

- *Start with $U_0 = U^{k-1}, G_0 = G^{k-1}$ and $V_0 = V^{k-1}$.*
  1. *Update*
  $$G_1 = G_0 + \Delta t (U_0)^+ F(U_0 G_0 V_0^T, t^{k-1})(V_0^+)^T.$$
  and set $U_1 = U_0, V_1 = V_0$.
  2. *Update*
  $$U_2 G_2 = U_1 G_1 + \Delta t P_{U_1}^{\perp} F(U_1 G_1 V_1^T, t^{k-1})(V_1^+)^T \text{ (using QR)},$$
  and set $V_2 = V_1$.
  3. *Update*
  $$V_3 G_3^T = V_2 G_2^T + \Delta t P_{V_2}^{\perp} F(U_2 G_2 V_2^T, t^{k-1})^T (U_2^+)^T \text{ (using QR)},$$
  and set $U_3 = U_2$.
- *Set $Z^k = U_3 G_3 V_3^T$.*

Now, let introduce the *KSL Algorithm* (see Algorithm 3.7) for the symmetric splitting scheme. Here, we adopt the original name given in [10], with the following correspondence: K stands for UG, S for G and L for VG^T.

Algorithm 3.7 (KSL algorithm)**.** *Given the initial rank-r approximation $Z^0 = U^0 G^0 (V^0)^T$, compute $Z^k \in \mathcal{M}_r(\mathbb{R}^{n\times m})$ for $k \in \{1, ..., K\}$ as follows.*

- *Start with $U_0 = U^{k-1}, G_0 = G^{k-1}$ and $V_0 = V^{k-1}$.*
  1. *Update UG with*
  $$U_1 G_1 = U^{k-1} G^{k-1} + \Delta t F(U_0 G_0 V_0^T, t^{k-1}) V_0 \quad \text{using QR},$$
  and set $V_1 = V_0$.
  2. *Update G with*
  $$G_2 = G_1 + \Delta t U_1^T F(U_1 G_1 V_1^T, t^{k-1}) V_1,$$
  and set $U_2 = U_1$ and $V_2 = V_1$.
  3. *Update VG^T with*
  $$V_3 G_3^T = V_2 G_2^T + \Delta t F(U_2 G_2 V_2^T, t^{k-1})^T U_3^T \quad \text{using QR},$$
  and set $U_3 = U_2$.
- *Set $Z^k = U_3 G_3 V_3^T$.*

3.3 Link between the two algorithms

Here, we discuss the similarity of the two proposed algorithms in the particular case where the flux F is independent of Z which includes the particular case of matrix approximation where $F(Z(t),t) := \dot{A}(t)$ with $A(t) \in \mathbb{R}^{n\times m}$.

Lemma 3.8. *In the case where the flux F is independent of Z meaning $F(Z(t),t) := F(t)$, the sequence of approximations $\{Z^k\}_k \subset \mathcal{M}_r(\mathbb{R}^{n\times m})$ provided by KSL algorithm and chart based algorithm coincide.*

Proof. We assume that the explicit flux evaluation $F(t^{k-1})$, required at each time step k of both KSL and chart based algorithms, is noted F^{k-1} for the sake of presentation. Approximations provided by chart based algorithm are surrounded by the $\tilde{}$ symbol.

At step k, the KSL algorithm provides the following approximations

$$K_1 = U_0G_0 + \Delta t F^{k-1}V_0, \quad (U_1,G_1) = QR(K_1), V_1 = V_0, \tag{16}$$
$$G_2 = G_1 - \Delta t U_1^T F^{k-1}V_1, \qquad U_2 = U_1, V_2 = V_1, \tag{17}$$
$$L_3 = V_2G_2^T + \Delta t (F^{k-1})^T U_2, \quad (V_3,G_3^T) = QR(L_3), U_3 = U_2. \tag{18}$$

Meanwhile, the chart based algorithm gives

$$\tilde{G}_1 = G_0 + \Delta t U_0^T F^{k-1}V_0, \qquad \tilde{U}_1 = U_0, \tilde{V}_1 = V_0, \tag{19}$$
$$\tilde{K}_2 = \tilde{U}_1\tilde{G}_1 + \Delta t(I - P_{\tilde{U}_1})F^{k-1}\tilde{V}_1, \quad (\tilde{U}_2,\tilde{G}_2) = QR(\tilde{K}_2), \tilde{V}_2 = \tilde{V}_1, \tag{20}$$
$$\tilde{L}_3 = \tilde{V}_2\tilde{G}_2^T + \Delta t(I - P_{\tilde{V}_2})(F^{k-1})^T\tilde{U}_2, \quad (\tilde{V}_3,\tilde{G}_3^T) = QR(\tilde{L}_3), \tilde{U}_3 = \tilde{U}_2. \tag{21}$$

As $\tilde{U}_1 = U_0$ and $\tilde{V}_1 = V_0$, expanding $\tilde{G}_1$ in Equation (20) yields

$$\tilde{K}_2 = \tilde{U}_1\tilde{G}_1 + \Delta t(I - P_{\tilde{U}_1})F^{k-1}\tilde{V}_1 = U_0G_0 + \Delta t U_0U_0^T F^{k-1}V_0 + \Delta t(I - U_0U_0^T)F^{k-1}V_0 = K_1.$$

Then $\tilde{G}_2 = G_1$ and $\tilde{U}_2 = U_1$. Moreover $\tilde{V}_2 = V_2 = V_0$. Using these equalities and injecting the expression of L_3, we get

$$L_3 = V_2G_2^T + \Delta t(F^{k-1})^T U_2 = \tilde{V}_2\tilde{G}_2^T - \Delta t\tilde{V}_2\tilde{V}_2^T(F^{k-1})^T\tilde{U}_2 + \Delta t(F^{k-1})^T\tilde{U}_2 = \tilde{L}_3$$

from which we deduce $L_3 = \tilde{L}_3$ then $G_3 = \tilde{G}_3$ and $V_3 = \tilde{V}_3$. This means $Z^3 = \tilde{Z}^3$ which concludes the proof. □

In [10], under the assumption that the set of matrices of bounded rank r is an invariant set for the flow, it follows that the KSL method is first order accurate and exact. In consequence, under the same assumptions from Lemma 3.8 the chart based method satisfies the same properties.

Proposition 3.9. *Assume that $A(t) \in \mathbb{R}^{n\times m}$ is a matrix with bounded rank r for all time t and $F(Z(t),t) := \dot{A}(t)$. Setting $\Delta t F(t^{k-1}) = A(t^k) - A(t^{k-1})$, the chart based splitting algorithm is exact, that is, $Z^k = A(t^k)$.*

Proof. The result follows from Lemma 3.8 and the property of exactness of the KSL algorithm [10, Theorem 4.1] in that context. □

Remark 3.10. *Proposition 3.9 insures that both methods are exact, whereas they rely on two different splitting methods, with different order of parameter updates. This may sound surprisingly at first, as it has been demonstrated by numerical experiment that [10, §5.2] KLS variant* [4] *of the KSL algorithm provides less accurate approximations and loses exactness property.*

[4] For that variant, it means that UG, VG^T and then G are updated.

4 Numerical results

In this section, we confront the two presented splitting algorithms namely **KSL** and **chart based** algorithms for dynamical low-rank approximation in the context of matrix approximation, in Section 4.1, and matrix dynamical systems arising from a parameter-dependent semi-discretized viscous Burgers's equation, in Section 4.2.

4.1 Matrix approximation

As first example, we consider the problem of approximating a matrix $A(t) \in \mathbb{R}^{100\times 100}$ on the time interval $[0, 1]$. This matrix is given explicitly as

$$A(t) = e^{tW_1} D e^t e^{tW_2},$$

where D is a diagonal matrix in $\mathbb{R}^{100\times 100}$ with non zero coefficients $d_{ii} = 2^{-i}, i \leq 10$, and $W_1, W_2 \in$ are two random skew symmetric matrices. Then, $A(t)$ has a rank 10 whose non zero singular values are $\sigma_i(t) = e^t 2^{-i}, i \leq 10$.

The two algorithms are first applied with numerical flux taken equal to $\Delta t F(Z_i, t^{k-1}) = A(t^k) - A(t^{k-1})$, where Z_i stands for the approximation at Step i of the iteration k of the splitting algorithms. In what follows, we study the behaviour of the approximation error $e^k = \|A(t^k) - Z^k\|$ at iteration k. The evolution of e^k for the two methods with respect to t^k is given on Figure 3 for $r \in \{10, 20\}$.

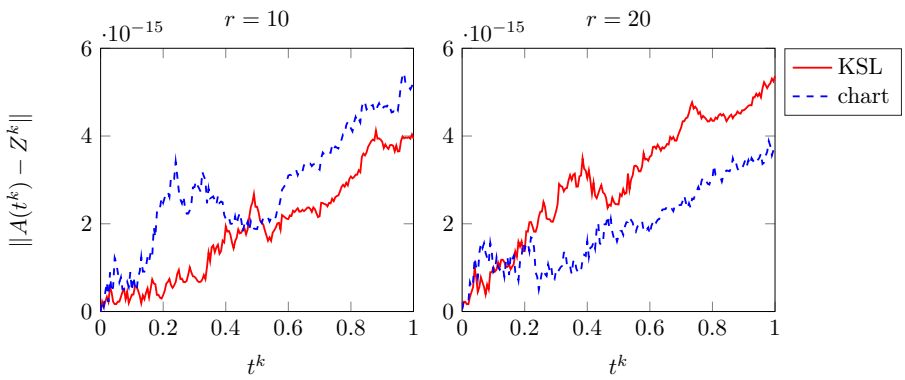


Figure 3: Matrix approximation with $\Delta t F(Z_i, t^{k-1}) = A(t^k) - A(t^{k-1})$: evolution of e^k for $\Delta t = 5\ \ 10^{-3}$, $r \in \{10, 20\}$.

As the matrix $A(t)$ is of rank 10 we recover that both methods are exact for $r = 10$ with approximation error e^k of order 10^{-15}. In case of over-approximation, when $r = 20$, the two methods demonstrate to be robust and again provide exact approximation.

Similar experiments are performed but considering the exact expression of the matrix derivative with $F(Z_i, t^{k-1}) = \dot{A}(t^{k-1})$. Approximation error evolution is depicted on Figure 4.

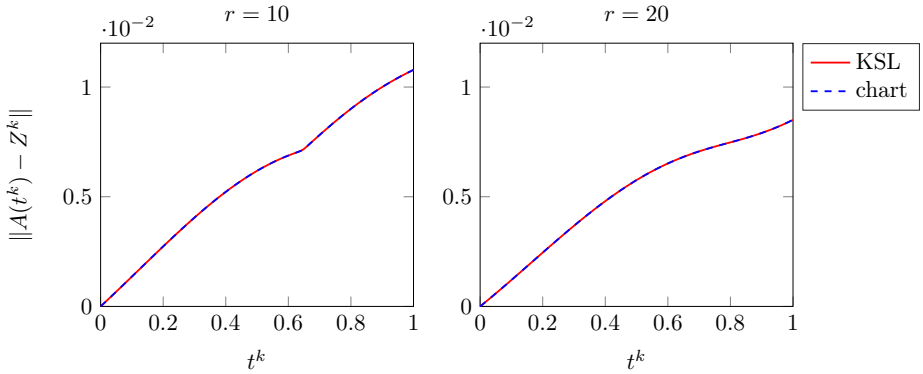


Figure 4: Matrix approximation with $F(Z_i, t^{k-1}) = \dot{A}(t^{k-1})$: evolution of e^k for $\Delta t = 5 \;\; 10^{-3}$, $r \in \{10, 20\}$.

Again, both methods give similar results for $r \in \{10, 20\}$, as shown on Figure 4 where the error plots coincide either in case of overapproximation. However, for this choice of $F(Z_i, t^{k-1})$, both method are no longer exact. Here, the error increases with time up to approximatively 10^{-2} with $r = 10$ and 10^{-3} for $r = 20$. To quantify this error, we perform some convergence study with respect to Δt and r. On Figure 5, we show the behaviour of the final approximation error $e^K = \|A(1) - Z^K\|$ for various rank $r \in \{4, 6, 8, 16, 32\}$ and time step $\Delta t \in \{10^{-5}, ..., 10^{-1}\}$.

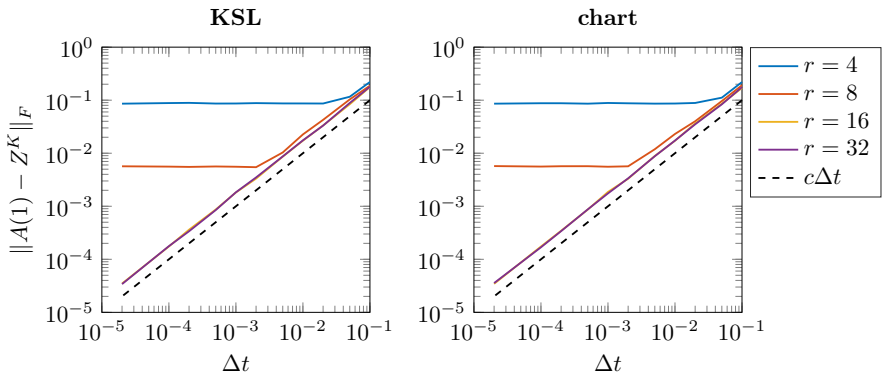


Figure 5: Matrix approximation with $F(Z_i, t^{k-1}) = \dot{A}(t^{k-1})$: final error e^K for both algorithms for different $(r, \Delta t)$.

Figure 5 illustrates that the two methods are first order in time as expected. Indeed we observe a linear decreasing of the error up to some stagnation value for $r \leq 8$. This stagnation is related to the low-rank approximation error as it decreases when r increases.

4.2 Parameter dependent problem

In the lines of [2], we consider the approximation of the parameter-dependent Burger's viscous equation in one dimension. To that goal, let $\Omega \times I = (0, 1) \times [0, 1]$ be a space time domain. We

seek $u(\cdot;\xi)$ the solution of

$$\begin{aligned} \partial_t u(x,t;\xi) - \xi\partial^2_{xx}u(x,t,\xi) + u(t,x;\xi)\partial_x u(x,t;\xi) &= f(x,t), \quad \text{on } \Omega\times I, \\ u^0(x;\xi) &= \sin(x)+\xi, \quad \text{on } \Omega. \end{aligned} \tag{22}$$

supplemented with homogeneous Dirichlet boundary conditions. The solution $u(\cdot;\xi)$ depends on the parameter ξ through the viscosity and and the initial condition u^0. Here, ξ takes its values in $[0.01, 0.06]$. Finally, the source term is defined by means of the function $f : \Omega\times I \to \mathbb{R}$ given by

$$f(x,t) = 4\sin(4\pi t)e^{-(x-0.2)^2/0.03^2}\mathbf{1}_{[0.1,0.3]}(x).$$

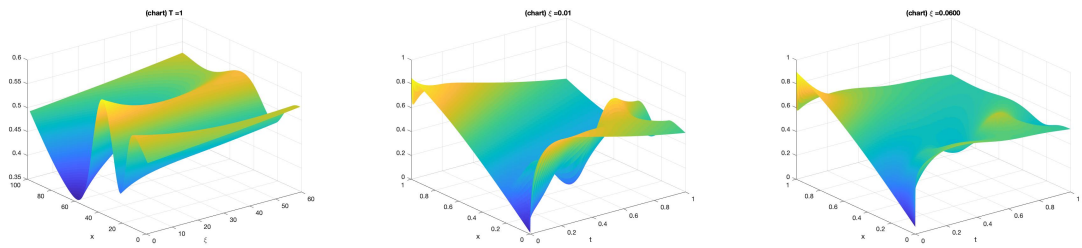
The problem (22) is semi-discretized in space by means of finite difference (FD) schemes with $n = 100$ nodes and $m = 60$ instances of the parameter ξ equally distributed on $[0.01, 0.06]$ such that we get the following dynamical system

$$\dot{X}(t) = LX(t) + h(X,t), \qquad X(0) = 0, \tag{23}$$

where the solution $X(t)$ is a matrix in $\mathbb{R}^{n\times m}$. The tensorized operator $L = D_x \otimes M_\xi$ is defined by means of $D_x \in \mathbb{R}^{n\times n}$ the discrete Laplacian obtained by second order center FD scheme and $M_\xi \in \mathbb{R}^{m\times m}$ a diagonal matrix whose non-zero coefficients are the m instances of ξ. Moreover, we define the matrix valued function h with entries $[h(Z,t)]_{ij} = Z_{ij}[C_x Z]_{ij} + f(x_j,t)$ where $C_x \in \mathbb{R}^{n\times n}$ is the discrete version of the first derivative obtained by 1st order centred FD scheme.

For solving the matrix ODE given by Equation (23), we first confront the chart based and KSL algorithms for various ranks, fixing $\Delta t = 10^{-4}$. The approximations obtained are compared to a reference solution noted $\{X^k_{ref}\}_{k=0}^K$ computed with an explicit Euler scheme with $\Delta t = 10^{-5}$.

Figure 6 illustrates the behaviour of the numerical solution obtained with the chart based algorithm at final time $t = 1$, and $r = 20$ for two different instances of the parameter ξ.



$t = 1$ $\qquad$ $\xi = 0.01$ $\qquad$ $\xi = 0.06$

Figure 6: Burgers's equation: approximation for the chart method with $r = 20$ and $\Delta t = 10^{-4}$, at final time $t = 1$ (left), for $\xi = 0.01$ (center) and $\xi = 0.06$ (right).

As we can observe on Figure (7), the approximations computed with the two methods are in good agreement with the reference solution.

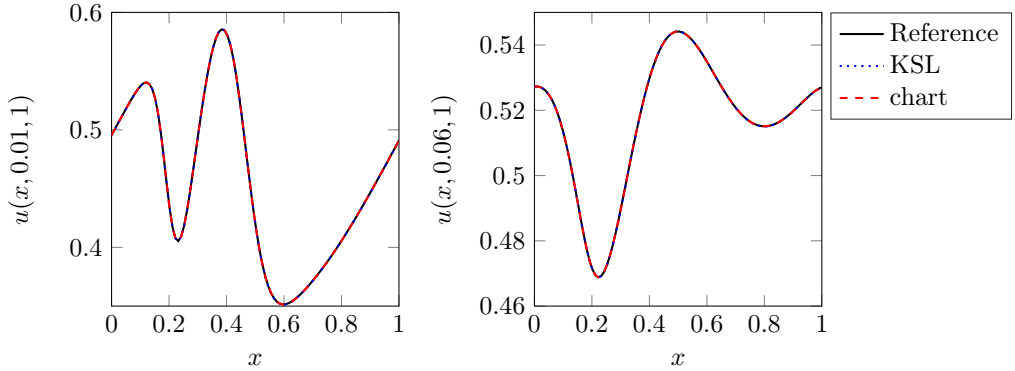


Figure 7: Burgers's equation: approximation for the chart and KSL method compared to the reference solution at final time for $r = 20$ and $\Delta t = 10^{-4}$.

Now, we compare the approximation error to the reference noted $e^k = \|X^k_{ref} - Z^k\|$ for the two splitting algorithms. On Figure 8, the evolution of e^k for both algorithms is studied with different ranks $r \in \{10, 20, 30, 40, 50\}$. As we can observe, the chart based algorithm provides more accurate approximation than the KSL algorithm as the rank increases. In particular for $r = 20$, we observe a gain of almost one order of magnitude. This behaviour is observed for larger ranks but seems constant for $r \geq 20$.

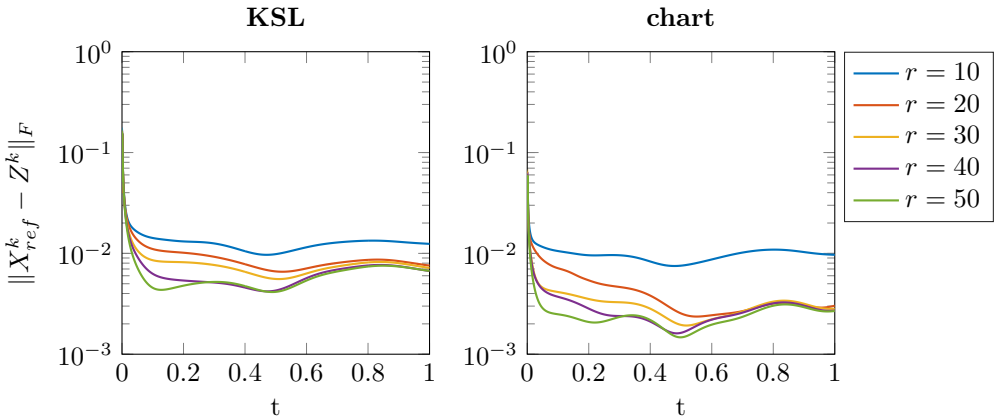


Figure 8: Burgers's equation: evolution of e^k for both algorithms for $r \in \{10, 20, 30, 40, 50\}$.

This observation is confirmed by Figure 9 where a convergence study is performed with respect to rank and time step. As we can see, final error $e^K = \|X^K_{ref} - Z^K\|$ decreases with respect to the rank and time step, and is the smallest for the chart based algorithm.

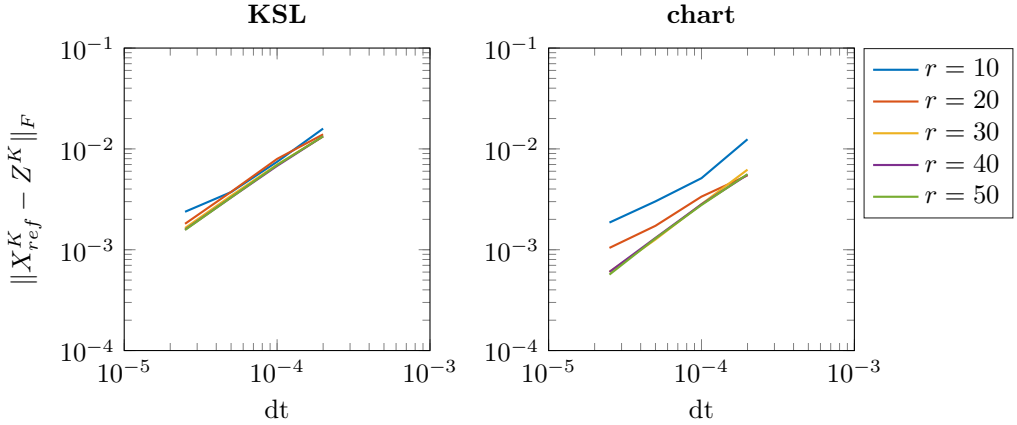


Figure 9: Burgers's equation: final errors e^K for different $(r, \Delta t)$.

5 Conclusion

In this paper, we have introduced and compared some geometry based algorithms for dynamical low-rank approximation. Using a different geometry description of the set of fixed rank matrices relying on charts, we generalized the description of [9]. Then, we proposed a different splitting algorithm which is proved to coincide with the KSL algorithm [10] in the particular case of matrix approximation. Nevertheless, for more general problems arising from the semi-discretization of parameter-dependent non-linear PDEs, the chart based algorithm seems to outperform the KSL algorithm. The properties of the proposed scheme were studied in the particular case of matrix approximation as it inherits from the properties of the KSL algorithm. Further error analysis should be conducted for derivation of rigorous error bounds in more general cases. Moreover, the proposed splitting scheme is a first step towards designing suitable algorithms integrating the geometric structure of the fixed rank matrix manifold, by working in neighbourhoods. Nevertheless, in this form, the algorithm remains blind to the fibre bundle structure of the set of fixed rank matrices. Deriving a numerical scheme working alternatively in the horizontal and vertical tangent spaces (as pointed out in Remark 2.3) is the object of future research.

A Chart based splitting integrator

Following the same lines as in [10], we justify how the chart based method introduced in Section 3.1.2 can be interpreted as a splitting scheme relying on the projector operator decomposition. To that goal, we assume that the flux is not dependent on the matrix Z i.e. $F(Z(t), t) := F(t)$.

By Equation (15), the projector operator is the sum of three contributions $P_{T_Z(t)} = P_1 + P_2 + P_3$. Then, one integration step of the splitting method starting from t_0 to $t_1 = t_0 + \Delta t$ with initial guess $Z_0 \in \mathcal{M}_r(\mathbb{R}^{n\times m})$ proceeds as follows.

(S.1) Find $Z_1 \in \mathcal{U}_{Z_0}$, such that $\dot{Z}_1 = P_{U_0} F P_{V_0}^T$ with initial value $Z_1(t_0) = Z_0$ on $[t_0, t_1]$.

(S.2) Find $Z_2 \in \mathcal{U}_{Z_1}$, such that $\dot{Z}_2 = P_{U_1}^{\perp} F P_{V_1}^T$ with initial value $Z_2(t_0) = Z_1(t_1)$ on $[t_0, t_1]$.

(S.3) Find $Z_2 \in \mathcal{U}_{Z_2}$, such that $\dot{Z}_3 = P_{U_2} F (P_{V_2}^{\perp})^T$ with initial value $Z_3(t_0) = Z_2(t_1)$ on $[t_0, t_1]$.

At each step (S.i), we consider that we work in a fixed neighbourhood associated to the approximation at previous step Z_{i-1}. In particular it implies that the matrices U_i, V_i and their orthogonal $U_{i,\perp}, V_{i,\perp}$ are fixed. For such a construction, the solution of (S.i) is the matrix Z_i provided by the EDO solved at Step i. of the chart based method.

Proposition A.1. *The solution of (S.1) is given by* $Z_1 = (U_0 + U_{0,\perp}X_1)H_1(V_0 + V_{0,\perp}Y_1)^T$ *with*

$$\dot{H}_1 = U_0^+ F (V_0^+)^T, \qquad \dot{X}_1 = 0, \dot{Y}_1 = 0. \tag{24}$$

The solution of (S.2) is given by $Z_2 = (U_1H_2 + U_{1,\perp}X_2H_2)(V_1 + V_{1,\perp}Y_2)^T$ *with*

$$\frac{d}{dt}(U_{1,\perp}X_2H_2) = P_{U_1}^{\perp} F (V_1^+)^T, \qquad \dot{Y}_2 = 0, \frac{d}{dt}(U_1H_2) = 0. \tag{25}$$

The solution of (S.3) is given by $Z_3 = (U_2 + U_{2,\perp}X_3)(V_2H_3^T + V_{2,\perp}Y_3H_3^T)^T$ *with*

$$\frac{d}{dt}(V_{2,\perp}YH^T) = P_{V_2}^{\perp} F (U_2^+)^T, \qquad \dot{X}_3 = 0, \frac{d}{dt}(V_2H_3^T) = 0. \tag{26}$$

Proof. For (S.1), the solution admits the decomposition $Z_1 = (U_0 + U_{0,\perp}X_1)H_1(V_0 + V_{0,\perp}Y_1)^T$ with derivative

$$\dot{Z}_1 = U_{0,\perp}\dot{X}_1H_1(V_0 + V_{0,\perp}Y_1)^T + (U_0 + U_{0,\perp}X_1)\dot{H}_1(V_0 + V_{0,\perp}Y_1)^T + (U_0 + U_{0,\perp}X_1)H_1(V_{0,\perp}\dot{Y}_1)^T.$$

which is equal to $\dot{Z}_1 = P_{U_0}FP_{V_0}^T$. Multiplying on the left by U_0^+ and on the right by $(V_0^+)^T$ the matrix $\dot{Z}_1$ in both expressions leads to $\dot{H}_1 = U_0^+F(V_0^+)^T$ and $\dot{X}_1 = 0, \dot{Y}_1 = 0$.

Now let us turn to (S.2).The solution can be written as $Z_1 = (U_1H_2 + U_{1,\perp}X_2H_2)(V_1 + V_{1,\perp}Y_2)^T$ with derivative

$$\dot{Z}_2 = \frac{d}{dt}(U_1H_2 + U_{1,\perp}X_2H_2)(V_1 + V_{1,\perp}Y_2)^T + (U_1H_2 + U_{1,\perp}X_2H_2)(V_{1,\perp}\dot{Y}_2)^T,$$

which is equal to $\dot{Z}_2 = P_{U_1}^{\perp}FP_{V_1}^T$. By multiplying on the right by $(V_1^+)^T$, equality is satisfied if $\frac{d}{dt}(U_{1,\perp}X_2H_2) = P_{U_1}^{\perp}F(V_1^+)^T$ and $\dot{Y}_2 = 0, \frac{d}{dt}(U_1H_2) = 0$. The third point of the lemma is obtained in the same manner. □

Remark A.2. *As mentioned in Remark 3.4, there is no need to compute the matrices* $U_{i,\perp}, V_{i,\perp}$ *explicitly. They are only introduced for convenient description of* $\mathcal{U}_{Z_1}$ *and* $\mathcal{U}_{Z_2}$ *for* $(S.2)$ *and* $(S.3)$ *respectively.*

References

[1] Billaud-Friess M., A. Nouy.: Dynamical model reduction method for solving parameter-dependent dynamical systems. SIAM Journal on Scientific Computing, 39(4), A1766-A1792 (2017)

[2] Billaud-Friess M. , Falcó A. , Nouy A.: Principal bundle structure of matrix manifolds. arXiv preprint, arXiv:1705.04093 (2017)

[3] Ceruti G., Lubich C.: Time integration of symmetric and anti-symmetric low-rank matrices and Tucker tensors. arXiv preprint, arXiv:1906.01369 (2019)

[4] Cheng M., Hou T.Y., Zhang Z., Sorensen D.-C.:, A dynamically bi-orthogonal method for time-dependent stochastic partial differential equations I: Derivation and algorithms. Journal of Computational Physics, 242(0):843-868 (2013)

[5] Falcó A., Sánchez F.: Model order reduction for dynamical systems: A geometric approach. Comptes Rendus Mécanique, 346(7), 515-523 (2018)

[6] Kieri E., Lubich C., Walach H.: Discretized dynamical low-rank approximation in the presence of small singular values. SIAM Journal on Numerical Analysis, 54(2), 1020-1038 (2016)

[7] Kieri E., Vandereycken B.:. Projection methods for dynamical low-rank approximation of high-dimensional problems. Computational Methods in Applied Mathematics, 19(1), 73-92 (2019)

[8] Khoromskij B. N., Oseledets I., Schneider R.: Efficient time-stepping scheme for dynamics on TT-manifolds. preprint (2012).

[9] Koch O., Lubich C.:. Dynamical low-rank approximation. SIAM Journal on Matrix Analysis and Applications, 29(2), 434-454 (2007)

[10] Lubich C., Oseledets I. V.: A projector-splitting integrator for dynamical low-rank approximation. BIT Numerical Mathematics, 54(1), 171-188 (2014)

[11] Musharbash E., Nobile F. Zhou T.: On the dynamically orthogonal approximation of time-dependent random PDEs, SIAM Journal on Scientific Computing, 2015, 37(2):A776-A810 (2015)

[12] Sapsis T.P., Lermusiaux P.F.J.: Dynamically orthogonal field equations for continuous stochastic dynamical systems. Physica D: Nonlinear Phenomena, 238(23-24):2347–2360 (2009)

[13] Nonnenmacher A. and C.Lubich C.: Dynamical low-rank approximation: applications and numerical experiments. Mathematics and Computers in Simulation, 79(4), 1346-1357 (2008)