An adaptive dynamical low-rank optimizer for solving kinetic parameter identification inverse problems*

Lena Baumann†, Lukas Einkemmer‡, Christian Klingenberg†, and Jonas Kusch §

Abstract. The numerical solution of parameter identification inverse problems for kinetic equations can exhibit high computational and memory costs. In this paper, we propose a dynamical low-rank scheme for the reconstruction of the scattering parameter in the radiative transfer equation from a number of macroscopic time-independent measurements. We first work through the PDE constrained optimization procedure in a continuous setting and derive the adjoint equations using a Lagrangian reformulation. For the scattering coefficient, a periodic B-spline approximation is introduced and a gradient descent step for updating its coefficients is formulated. After the discretization, a dynamical low-rank approximation (DLRA) is applied. We make use of the rank-adaptive basis update & Galerkin integrator and a line search approach for the adaptive refinement of the gradient descent step size and the DLRA tolerance. We show that the proposed scheme significantly reduces both memory and computational cost. Numerical results computed with different initial conditions validate the accuracy and efficiency of the proposed DLRA scheme compared to solutions computed with a full solver.

Key words. parameter identification, inverse problem, dynamical low-rank approximation, radiative transfer equation, PDE constrained optimization, rank adaptivity

MSC codes. 35Q49, 49M41, 65M22, 65M32

1. Introduction.

A classical problem in medical imaging consists in the reconstruction of properties of the examined tissue from measurements without doing harm to the human body. In optical tomography, the propagation of near-infrared light through tissue can be modeled using the *radiative transfer equation (RTE)* [22, 15, 14]. Neglecting boundary effects, the time-dependent form of this kinetic partial differential equation (PDE) can be given in one-dimensional slab geometry as

$$\begin{cases} \partial_t f(t,x,v) + v\partial_x f(t,x,v) &= \sigma(x)\left(\frac{1}{|\Omega_v|}\langle f(t,x,v)\rangle_v - f(t,x,v)\right), \\ f(t=0,x,v) &= f_{\text{in}}(x,v), \end{cases} \tag{1.1}$$

where $f(t,x,v): \mathbb{R}_0^+ \times \Omega_x \times \Omega_v \to \mathbb{R}_0^+$ denotes the distribution function that describes the repartition of photons in phase space. Here, t stands for the time variable, $x \in \Omega_x \subseteq \mathbb{R}$ for the space variable and $v \in \Omega_v = [-1,1]$ for the angular variable. An integration over the corresponding domain is denoted by brackets $\langle\cdot\rangle$ and $|\Omega_v|$ measures the length of the domain

*Submitted to the editors June, 26, 2025.

Funding: The work of the first author was supported by the Würzburg Mathematics Center for Communication and Interaction (WMCCI) as well as the Stiftung der Deutschen Wirtschaft (Foundation of German Business).

†Department of Mathematics, University of Würzburg, Würzburg, DE-97074, Germany (lena.baumann@uni-wuerzburg.de, christian.klingenberg@uni-wuerzburg.de).

‡Numerical Analysis and Scientific Computing, University of Innsbruck, Innsbruck, A-6020, Austria (lukas.einkemmer@uibk.ac.at).

§Department of Data Science, Norwegian University of Life Sciences, Ås, NO-1433, Norway (jonas.kusch@nmbu.no).

This manuscript is for review purposes only.

Ω_v. The function $\sigma(x)$ represents the properties of the background medium, indicating the probability of particles at position x to be scattered into a new direction. We refer to it as the *scattering coefficient*. At the initial time $t = 0$ the function $f_{\text{in}}(x, v)$ shall be prescribed for the distribution function.

The inverse problem associated to the RTE (1.1) considers the reconstruction of the scattering coefficient $\sigma(x)$ from measurements. For the theoretical background on inverse problems in general as well as on the theoretical requirements on $\sigma(x)$ and $f(t, x, v)$ in the inverse transport problem the reader is referred to [13] and to the review articles [1, 24], respectively. For the numerical solution of this parameter identification problem, PDE constrained optimization is deployed. In this setting, we aim for the minimization of the difference between the measurements and the computed solutions under the assumption of the validity of the RTE. Similar to recent papers [18, 7, 11, 9], we pursue a gradient-based approach for which in each iteration the evaluation of both the forward and the adjoint problem is required. Clearly, this can numerically become very costly, especially in higher-dimensional settings.

To reduce the computational cost and memory requirements for the solution of kinetic equations, *dynamical low-rank approximation (DLRA)* [16] can be applied. This approach approximates the kinetic distribution function f up to a certain *rank* r as

$$f(t, x, v) \approx \sum_{i,j=1}^{r} X_i(t, x)\, S_{ij}(t)\, V_j(t, v), \tag{1.2}$$

where $\{X_i : i = 1, .., r\}$ are the orthonormal basis functions in space and $\{V_j : j = 1, .., r\}$ are the orthonormal basis functions in angle. The matrix $\mathbf{S} = (S_{ij}) \in \mathbb{R}^{r\times r}$ contains the coefficients of the approximation and therefore is called the *coefficient* or *coupling matrix*. The idea of DLRA then consists in constraining the evolution dynamics to functions of the form (1.2). There are different integrators that are able to evolve the low-rank factors in time while not suffering from this solution structure. For instance, the *projector-splitting* [19], the *(augmented) basis update & Galerkin* (BUG) [6, 4], and the *parallel BUG integrator* [5] are widely used in various areas of research [2, 17, 10, 8].

For the solution of the inverse transport problem associated to (1.1) the following approach is pursued in this paper: "first optimize, then discretize, then low-rank", i.e. we first perform the optimization in a continuous setting before the resulting equations are discretized and a dynamical low-rank approximation is used. The main features of this paper are:

- *An application of DLRA to a PDE parameter identification inverse problem:* The scattering parameter $\sigma(x)$ is determined by PDE constrained optimization for which after the discretization the dynamical low-rank method is used. To our knowledge, this is the first paper that combines inverse problems and DLRA, leading to a reduction of computational effort from $\mathcal{O}\left(N^{d_x+d_v}\right)$ to $\mathcal{O}\left(rN^{\max(d_x,d_v)}\right)$ in each step, where N denotes the number of grid points in physical as well as angular space and d_x, d_v the dimensions in space and angle, respectively.
- *A setup close to realistic applications:* In most applications measurements are not able to access the full distribution function but at most angle-averaged quantities, i.e. its moments. We will consider such a setup here where it is assumed that only the first moment is accessible by measurements. In addition, optimal tomography commonly

relies on a multitude of measurements from different positions which we incorporate by probing multiple initial values.

- *An adaptive gradient descent step size and an augmented low-rank integrator:* The minimization is performed using a gradient descent method for updating the coefficients of a periodic B-spline approximation of $\sigma(x)$. Similar to [23], the step size is chosen adaptively by a line search approach with Armijo condition. Also the rank of the DLRA algorithm is chosen adaptively by using the augmented BUG integrator from [4]. This allows us to choose the rank in each step such that a given error tolerance is satisfied. In the context of optimization, this enables us to start with a comparatively small rank (when we are still far from the minimum) and then gradually increase the rank as the optimization progresses, thereby enhancing the performance of the low-rank scheme.
- *A series of numerical test examples:* A series of numerical text examples confirms that for the reconstruction of the scattering coefficient the application of DLRA shows good agreement with the full solution while being significantly faster, suggesting that the combination of low-rank methods and inverse problems is a promising field of future research.

The structure of the paper is as follows: After the introduction in Section 1, the PDE constrained optimization procedure for the solution of the inverse parameter identification problem is explained in Section 2. Section 3 is devoted to the discretization of the forward and the adjoint equations as well as of the gradient in angle, space, and time, leading to a fully discrete gradient descent scheme. In Section 4, the concept of DLRA is introduced and subsequently applied to the forward and adjoint equations. An adaptive line search method for refining the gradient descent step size and the DLRA rank tolerance is presented. Numerical results given in Section 5 confirm the accuracy and efficiency of the DLRA scheme compared to the solutions computed with the full solver. Finally, Section 6 gives a brief conclusion and an outlook for possible further research.

2. PDE constrained optimization.

For the reconstruction of the scattering coefficient $\sigma(x)$ a multitude of N_{IC} measurements shall be taken into account. We assume the measurements to be generated by a measurement operator M acting on the angle-averaged solution of the RTE at the final time $t = T$, that has been generated using the corresponding initial condition $f_{\mathrm{in},m}$. For simplicity, the computed data d_m is assumed to be close to the measurements of an angle-averaged solution, i.e.

$$d_m(x) \approx M\left(\langle f_{\sigma,m}(t=T,x,v)\rangle_v\right) \qquad \text{for} \qquad m = 1, ..., N_{\mathrm{IC}},$$

where $f_{\sigma,m}(t,x,v)$ is a solution of

$$\begin{cases} \partial_t f_m(t,x,v) + v\partial_x f_m(t,x,v) & = \sigma(x)\left(\frac{1}{|\Omega_v|}\langle f_m(t,x,v)\rangle_v - f_m(t,x,v)\right), \\ f_m(t=0,x,v) & = f_{\mathrm{in},m}(x,v). \end{cases} \tag{2.1}$$

One then tries to minimize the square loss between the simulated angle-averaged solution and the measured data, i.e. one tries to solve the minimization problem

$$\min_{\sigma} J(\sigma) \quad \text{with} \quad J(\sigma) = \frac{1}{2}\sum_{m=1}^{N_{\mathrm{IC}}} \left\langle \left| \langle f_{\sigma,m}(t=T,x,v)\rangle_v - d_m(x)\right|^2 \right\rangle_x \quad \text{subject to} \quad (2.1). \tag{2.2}$$

Note that this setup is close to realistic applications in the sense as described above. For real-word applications we point out that the considered setting with one spatial and one angular variable may not be sufficient. In addition, it is assumed that there is no noise in the measurements which in practical applications is clearly infeasible. Even though, the results gained from the considered setup can directly be extended to higher-dimensional settings and give valuable insights into the combination of parameter identification and DLRA, which this paper aims for.

In Subsection 2.1 we make use of the method of Lagrange multipliers to derive the adjoint equations associated to the forward problem (2.1). We then derive the explicit gradient descent step in Subsection 2.2.

2.1. Lagrangian formulation.

To reformulate the PDE constrained minimization problem (2.2) into an unconstrained optimization problem the method of Lagrange multipliers is used. Note that from now on, for brevity, we write $f_m(t,x,v)$ instead of $f_{\sigma,m}(t,x,v)$. We aim for a solution of

$$\min \mathcal{L}\left(f_1, ... f_{N_{\mathrm{IC}}}, g_1, ..., g_{N_{\mathrm{IC}}}, \lambda_1, ..., \lambda_{N_{\mathrm{IC}}}, \sigma\right),$$

where

$$\begin{aligned}\mathcal{L} = J\left(f_1, ..., f_{N_{\mathrm{IC}}}\right) &+ \sum_{m=1}^{N_{\mathrm{IC}}} \left\langle g_m, \partial_t f_m + v\partial_x f_m - \sigma(x)\left(\frac{1}{|\Omega_v|}\langle f_m\rangle_v - f_m\right)\right\rangle_{t,x,v} \\ &+ \sum_{m=1}^{N_{\mathrm{IC}}} \langle \lambda_m, f_m(t=0,x,v) - f_{\mathrm{in},m}(x,v)\rangle_{x,v}\end{aligned}$$

and $g_m(t,x,v)$ and $\lambda_m(x,v)$ are Lagrange multipliers with respect to $f_m(t,x,v)$ and the initial distributions $f_{\mathrm{in},m}(x,v)$ for $m=1,...,N_{\mathrm{IC}}$, respectively. Applying integration by parts and assuming periodic boundary conditions, the Lagrangian can be rewritten as

$$\begin{aligned}\mathcal{L} = J\left(f_1, ..., f_{N_{\mathrm{IC}}}\right) &+ \sum_{m=1}^{N_{\mathrm{IC}}} \left\langle f_m, -\partial_t g_m - v\partial_x g_m - \sigma(x)\left(\frac{1}{|\Omega_v|}\langle g_m\rangle_v - g_m\right)\right\rangle_{t,x,v} \\ &+ \sum_{m=1}^{N_{\mathrm{IC}}} \langle g_m(t=T,x,v), f_m(t=T,x,v)\rangle_{x,v} \\ &- \sum_{m=1}^{N_{\mathrm{IC}}} \langle g_m(t=0,x,v), f_m(t=0,x,v)\rangle_{x,v} \\ &+ \sum_{m=1}^{N_{\mathrm{IC}}} \langle \lambda_m, f_m(t=0,x,v) - f_{\mathrm{in},m}(x,v)\rangle_{x,v}.\end{aligned}$$

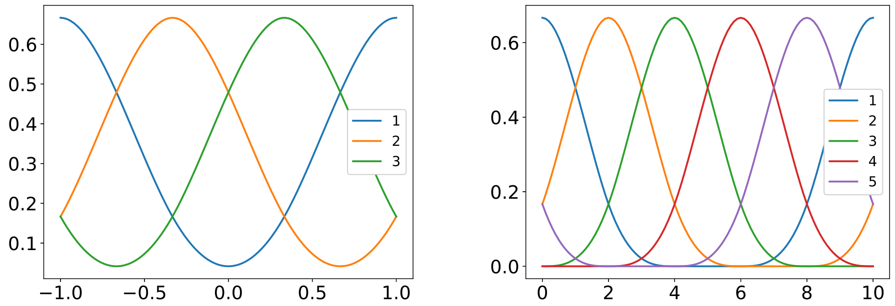

Figure 1. *Cubic periodic B-spline basis functions for $N_c = 3$ (left) and $N_c = 5$ (right) on different spatial domains.*

The corresponding *adjoint* or *dual problems* associated to (2.1) can then be derived by setting $\frac{\partial \mathcal{L}}{\partial f_m} = 0$ for $m = 1, ..., N_{\mathrm{IC}}$. By straightforward calculation one obtains

$$\begin{cases} -\partial_t g_m(t,x,v) - v\partial_x g_m(t,x,v) & = \sigma(x)\left(\frac{1}{|\Omega_v|}\langle g_m(t,x,v)\rangle_v - g_m(t,x,v)\right), \\ g_m(t=T,x,v) & = -\langle f_m(t=T,x,v)\rangle_v + d_m(x). \end{cases} \tag{2.3}$$

The solution of the forward equations (2.1) as well as the adjoint equations (2.3) will then be used for the computation of the gradient in the following gradient descent step.

2.2. Optimization parameters and gradient descent step. When evaluating the scattering coefficient $\sigma(x)$ at each point of the spatial grid and taking these values as the parameters to be optimized, there are several computational disadvantages. For instance, a huge parameter space is obtained and very rough functions are part of the ansatz space. To avoid this, we consider the parametrization of $\sigma(x)$ by splines. In particular, we approximate

$$\sigma(x) \approx \sum_{i=1}^{N_c} c_i B_i(x), \tag{2.4}$$

where N_c denotes the finite number of spline functions, $B_i(x)$ are the periodic B-spline basis functions, and c_i the coefficients of the approximation. In Figure 1 the basis functions for cubic periodic B-splines for $N_c = 3$ and $N_c = 5$ are illustrated.

The gradient descent step for the solution of the minimization problem (2.2) then updates the coefficients c_i^n to c_i^{n+1} for $i = 1, ..., N_c$ in each step by determining

$$c_i^{n+1} = c_i^n - \eta^n \left.\frac{\mathrm{d}J(f_1, ..., f_{N_{\mathrm{IC}}})}{\mathrm{d}c_i}\right|_{c_i = c_i^n}, \tag{2.5}$$

where η^n denotes an adaptively chosen step size.

As f_m satisfies the PDE constraints (2.1) and g_m solves the adjoint equations (2.3) for $m = 1, ..., N_{\mathrm{IC}}$, it holds

$$\mathcal{L}(f_1, ..., f_{N_{\mathrm{IC}}}, g_1, ..., g_{N_{\mathrm{IC}}}, \lambda_1, ..., \lambda_{N_{\mathrm{IC}}}, \sigma) = J(f_1, ..., f_{N_{\mathrm{IC}}}),$$

This manuscript is for review purposes only.

and thus

$$\begin{aligned}\frac{\mathrm{d}J\left(f_1,...,f_{N_{\mathrm{IC}}}\right)}{\mathrm{d}c_i} &= \frac{\mathrm{d}\mathcal{L}\left(f_1,...,f_{N_{\mathrm{IC}}},g_1,...,g_{N_{\mathrm{IC}}},\lambda_1,...,\lambda_{N_{\mathrm{IC}}},\sigma\right)}{\mathrm{d}c_i} \\ &= \sum_{m=1}^{N_{\mathrm{IC}}}\left(\frac{\partial\mathcal{L}}{\partial f_m}\frac{\partial f_m}{\partial\sigma}\frac{\partial\sigma}{\partial c_i}+\frac{\partial\mathcal{L}}{\partial g_m}\frac{\partial g_m}{\partial\sigma}\frac{\partial\sigma}{\partial c_i}+\frac{\partial\mathcal{L}}{\partial\lambda_m}\frac{\partial\lambda_m}{\partial\sigma}\frac{\partial\sigma}{\partial c_i}+\frac{\partial\mathcal{L}}{\partial\sigma}\frac{\partial\sigma}{\partial c_i}\right).\end{aligned}$$

The first three terms again vanish since (2.1) and (2.3) are fulfilled, leading to

$$\frac{\mathrm{d}J\left(f_1,...,f_{N_{\mathrm{IC}}}\right)}{\mathrm{d}c_i} = \sum_{m=1}^{N_{\mathrm{IC}}}\frac{\partial\mathcal{L}}{\partial\sigma}\frac{\partial\sigma}{\partial c_i} = \sum_{m=1}^{N_{\mathrm{IC}}}\left(-\frac{1}{|\Omega_v|}\left\langle\langle f_m\rangle_v,\langle g_m\rangle_v\right\rangle_t+\langle f_m,g_m\rangle_{t,v}\right)B_i. \tag{2.6}$$

Hence, we have derived an explicit formulation depending on the forward and on the adjoint equation as well as on the B-spline basis functions to compute the gradient in the gradient descent step (2.5).

3. Discretization.

For the numerical implementation we discretize the forward problem (2.1), the adjoint problem (2.3) and the gradient (2.6) in angle, space and time, leading to a fully discrete scheme. We begin with the angular discretization in Subsection 3.1, followed by a discretization in space in Subsection 3.2 and in time in Subsection 3.3. Subsection 3.4 then summarizes the fully discrete gradient descent method.

3.1. Angular discretization.

For the discretization in angle we decide on a modal approach making use of normalized Legendre polynomials P_ℓ. This is a standard approach that is commonly used for radiative transfer problems and the derived methods are referred to as *P_N methods* [21, 3, 20]. We use a rescaling of the Legendre polynomials such that $\langle P_k, P_\ell\rangle_v = \delta_{k\ell}$ and $P_0 = \frac{1}{\sqrt{2}}$ holds. They constitute a complete set of orthonormal functions on the interval $[-1,1]$. We expand the distribution functions f_m and g_m for $m = 1, ..., N_{IC}$ in terms of the rescaled Legendre polynomials and obtain the following approximation

$$f_m(t,x,v) \approx \sum_{\ell=0}^{N_v-1} u_{\ell m}(t,x)\,P_\ell(v) \quad \text{and} \quad g_m(t,x,v) \approx \sum_{\ell=0}^{N_v-1} w_{\ell m}(t,x)\,P_\ell(v)\,, \tag{3.1}$$

where the expansion coefficients $u_\ell(t,x)$ and $w_\ell(t,x)$, respectively, are called the *moments* and N_v is called the *order* of the approximation. We insert these representations into the forward problem (2.1) as well as the adjoint problem (2.3), multiply with P_k and integrate over the angular variable v. Using the orthonormality condition from above and the notation $A_{k\ell} = \langle P_k, vP_\ell\rangle_v$ leads to

$$\begin{cases}\partial_t u_{km}(t,x) &= -\sum_{\ell=0}^{N_v-1}\partial_x u_{\ell m}(t,x)\,A_{k\ell}+\sigma(x)\,u_{km}(t,x)\,(\delta_{k0}-1)\,,\\ u_{km}(t=0,x) &= u_{\mathrm{in},km}(x)\,,\end{cases} \tag{3.2}$$

for the forward equations and

$$\begin{cases}-\partial_t w_{km}(t,x) &= \sum_{\ell=0}^{N_v-1}\partial_x w_{\ell m}(t,x)\,A_{k\ell}+\sigma(x)\,w_{km}(t,x)\,(\delta_{k0}-1)\,,\\ w_{km}(t=T,x) &= \left(-2u_{0m}(t=T,x)+\sqrt{2}d_m(x)\right)\delta_{k0},\end{cases} \tag{3.3}$$

for the adjoint equations for $m = 1, ..., N_{IC}$. We collect the entries $A_{k\ell}$ in the symmetric matrix $\mathbf{A} = (A_{k\ell}) \in \mathbb{R}^{N_v \times N_v}$ and note that $\mathbf{A}$ is diagonalizable in the form $\mathbf{A} = \mathbf{Q}\mathbf{M}\mathbf{Q}^\top$ with $\mathbf{Q}$ orthonormal and $\mathbf{M} = \mathrm{diag}(\sigma_1, ..., \sigma_{N_v})$. We then set $|\mathbf{A}| = \mathbf{Q}|\mathbf{M}|\mathbf{Q}^\top$. For the angular discretization of the gradient we insert the representations (3.1) into (2.6) and obtain

(3.4)
$$\frac{\mathrm{d}J\left(f_1, ..., f_{N_{\mathrm{IC}}}\right)}{\mathrm{d}c_i} \approx \sum_{m=1}^{N_{\mathrm{IC}}} \left(-\left\langle u_{0m}\left(t,x\right), w_{0m}\left(t,x\right)\right\rangle_t + \sum_{k=0}^{N_v-1} \left\langle u_{km}\left(t,x\right), w_{km}\left(t,x\right)\right\rangle_t \right) B_i\left(x\right).$$

3.2. Spatial discretization.

The discretization in the spatial variable is performed on a spatial grid with N_x grid cells and equidistant spacing $\Delta x = \frac{1}{N_x}$ such that

$$u_{jkm}\left(t\right) \approx u_{km}\left(t, x_j\right), \; w_{jkm}\left(t\right) \approx w_{km}\left(t, x_j\right), \; \sigma_j \approx \sigma\left(x_j\right), \; d_{jm} \approx d_m\left(x_j\right), \; B_{ji} \approx B_i\left(x_j\right).$$

Spatial derivatives are approximated using a centered finite difference scheme to which a second-order stabilization term is added. We denote $\partial_x \approx \mathbf{D}^x \in \mathbb{R}^{N_x \times N_x}$ and $\partial_{xx} \approx \mathbf{D}^{xx} \in \mathbb{R}^{N_x \times N_x}$ for the tridiagonal stencil matrices with nonzero entries only at

$$D^x_{j,j\pm1} = \frac{\pm 1}{2\Delta x}, \qquad D^{xx}_{j,j} = -\frac{2}{\left(\Delta x\right)^2}, \quad D^{xx}_{j,j\pm1} = \frac{1}{\left(\Delta x\right)^2}.$$

In addition, we assume periodic boundary conditions which results in setting

$$D^x_{1,N_x} = \frac{-1}{2\Delta x}, \qquad D^x_{N_x,1} = \frac{1}{2\Delta x}, \qquad D^{xx}_{1,N_x} = D^{xx}_{N_x,1} = \frac{1}{\left(\Delta x\right)^2}.$$

The spatially discretized forward equations with centered finite differences and an additional second-order stabilization term can then be obtained from (3.2) as

$$(3.5) \quad \begin{cases} \partial_t u_{jkm}\left(t\right) & = -\sum_{i=1}^{N_x} \sum_{\ell=0}^{N_v-1} D^x_{ji} u_{i\ell m}\left(t\right) A_{k\ell} + \frac{\Delta x}{2} \sum_{i=1}^{N_x} \sum_{\ell=0}^{N_v-1} D^{xx}_{ji} u_{i\ell m}\left(t\right) |A|_{k\ell} \\ & \quad + \sigma_j u_{jkm}\left(t\right)\left(\delta_{k0} - 1\right), \\ u_{jkm}\left(t=0\right) & = u_{\mathrm{in},jkm}, \end{cases}$$

and the spatially discretized adjoint equations from (3.3) as

$$(3.6) \quad \begin{cases} -\partial_t w_{jkm}\left(t\right) & = \sum_{i=1}^{N_x} \sum_{\ell=0}^{N_v-1} D^x_{ji} w_{i\ell m}\left(t\right) A_{k\ell} + \frac{\Delta x}{2} \sum_{i=1}^{N_x} \sum_{\ell=0}^{N_v-1} D^{xx}_{ji} w_{i\ell m}\left(t\right) |A|_{k\ell} \\ & \quad + \sigma_j w_{jkm}\left(t\right)\left(\delta_{k0} - 1\right), \\ w_{jkm}\left(t=T\right) & = \left(-2u_{j0m}\left(t=T\right) + \sqrt{2} d_{jm}\right)\delta_{k0}, \end{cases}$$

For the spatial discretization of the gradient we get from (3.4) that

$$(3.7) \quad \frac{\mathrm{d}J\left(f_1, ..., f_{N_{\mathrm{IC}}}\right)}{\mathrm{d}c_i} \approx \sum_{m=1}^{N_{\mathrm{IC}}} \left(-\left\langle u_{j0m}\left(t\right), w_{j0m}\left(t\right)\right\rangle_t + \sum_{k=0}^{N_v-1} \left\langle u_{jkm}\left(t\right), w_{jkm}\left(t\right)\right\rangle_t \right) B_{ji}.$$

3.3. Time discretization. To obtain a fully discrete system, the time interval $[0, T]$ is split equidistantly into a finite number N_t of time cells. An update of the forward equations (3.5) from time t_n to time $t_{n+1} = t_n + \Delta t$ is then computed using an explicit Euler step forward in time such that

$$\begin{cases} u_{jkm}^{n+1} &= u_{jkm}^{n} - \Delta t \sum_{i=1}^{N_x} \sum_{\ell=0}^{N_v-1} D_{ji}^{x} u_{i\ell m}^{n} A_{k\ell} + \Delta t \frac{\Delta x}{2} \sum_{i=1}^{N_x} \sum_{\ell=0}^{N_v-1} D_{ji}^{xx} u_{i\ell m}^{n} \left|A\right|_{k\ell} \\ & \quad + \sigma_j \Delta t u_{jkm}^{n} \left(\delta_{k0} - 1\right), \\ u_{jkm}^{0} &= u_{\text{in},jkm}. \end{cases} \tag{3.8}$$

For the adjoint equations (3.6) we start computations with an end time condition after N_t steps and evolve the solution from time t_n to time $t_{n-1} = t_n - \Delta t$ by an explicit Euler step backwards in time such that

$$\begin{cases} w_{jkm}^{n-1} &= w_{jkm}^{n} + \Delta t \sum_{i=1}^{N_x} \sum_{\ell=0}^{N_v-1} D_{ji}^{x} w_{i\ell m}^{n} A_{k\ell} + \Delta t \frac{\Delta x}{2} \sum_{i=1}^{N_x} \sum_{\ell=0}^{N_v-1} D_{ji}^{xx} w_{i\ell m}^{n} \left|A\right|_{k\ell} \\ & \quad + \sigma_j \Delta t w_{jkm}^{n} \left(\delta_{k0} - 1\right), \\ w_{jkm}^{N_t} &= \left(-2u_{j0m}^{N_t} + \sqrt{2} d_{jm}\right) \delta_{k0}. \end{cases} \tag{3.9}$$

The fully discretized gradient can be obtained from (3.7) by approximating integrals with respect to time by step functions. We get

$$\frac{\mathrm{d}J\left(f_1, \ldots, f_{N_{\mathrm{IC}}}\right)}{\mathrm{d}c_i} \approx \frac{1}{N_t + 1} \sum_{m=1}^{N_{\mathrm{IC}}} \sum_{n=0}^{N_t} \left(-u_{j0m}^{n} w_{j0m}^{N_t - n} + \sum_{k=0}^{N_v - 1} u_{jkm}^{n} w_{jkm}^{N_t - n} \right) B_{ji}. \tag{3.10}$$

3.4. Fully discrete optimization scheme. The strategy for the fully discrete gradient descent method for the solution of the PDE parameter identification problem is summarized in Algorithm 3.1. Note that for the stopping criterion an error estimate `estimated-err` for the deviation of the computed coefficients from the true coefficients is required to run the algorithm.

4. Dynamical low-rank approximation.

For the solution of the PDE parameter identification problem the coefficients c_i of the spline approximation (2.4) of σ are updated several times in the gradient descent step (2.5). For each iteration the solution of the fully discretized forward equations (3.8) as well as of the fully discretized adjoint equations (3.9) have to be computed and stored in order to compute the fully discretized gradient (3.10). A method for the reduction of computational and memory effort for kinetic equations is the concept of dynamical low-rank approximation that shall be applied to the considered inverse transport problem. We begin with some general information on DLRA in Subsection 4.1, before Subsection 4.2 is devoted to a DLRA algorithm for the considered discrete optimization problem.

4.1. Background on dynamical low-rank approximation. In [16], the concept of DLRA has been introduced in a semi-discrete time-dependent matrix setting. We follow the explanations there. Let $\mathbf{f}\left(t\right) \in \mathbb{R}^{N_x \times N_v}$ be the solution of the matrix differential equation

$$\dot{\mathbf{f}}\left(t\right) = \mathbf{F}\left(\mathbf{f}\left(t\right)\right),$$

Algorithm 3.1 Gradient descent method for the PDE parameter reconstruction

Input: measurements $\mathbf{d}_m = (d_{jm}) \in \mathbb{R}^{N_x}$ for $m = 1, ..., N_{\mathrm{IC}}$,
initial data $\mathbf{u}_m^0 = \left(u_{jkm}^0\right) \in \mathbb{R}^{N_x \times N_v}$ for $m = 1, ..., N_{\mathrm{IC}}$,
initial guess for the coefficients $\mathbf{c}^0 = \left(c_i^0\right) \in \mathbb{R}^{N_c}$,
initial step size η^0,
estimated error `estimated-err`,
error tolerance `errtol`,
maximal number of iterations `maxiter`

Output: optimal coefficients $\mathbf{c}_{\mathrm{opt}} = (c_{\mathrm{opt},i}) \in \mathbb{R}^{N_c}$ within the prescribed error tolerance

while `estimated-err` $>$ `errtol` **and** $n \leq$ `maxiter` **do**
Compute $\boldsymbol{\sigma}^n = \left(\sigma_j^n\right) \in \mathbb{R}^{N_x}$ from given coefficients $\mathbf{c}^n$ according to (2.4);
Solve the forward problem according to (3.8) for each $m = 1, ..., N_{\mathrm{IC}}$;
Solve the adjoint problem according to (3.9) for each $m = 1, ..., N_{\mathrm{IC}}$;
Compute the gradient $\frac{\mathrm{d}J}{\mathrm{d}c_i^n}$ using (3.10) and the solutions of (3.8) and (3.9);
Update the coefficients according to (2.5): $c_i^{n+1} = c_i^n - \eta^n \frac{\mathrm{d}J}{\mathrm{d}c_i}\big|_{c_i = c_i^n}$, where η^n is determined adaptively by line search;
end while

for which the right-hand side shall be denoted by $\mathbf{F}\left(\mathbf{f}\left(t\right)\right) : \mathbb{R}^{N_x \times N_v} \to \mathbb{R}^{N_x \times N_v}$. We then seek an approximation of $\mathbf{f}\left(t\right)$ of the form

$$\mathbf{f}_r\left(t\right) = \mathbf{X}\left(t\right)\mathbf{S}\left(t\right)\mathbf{V}\left(t\right)^\top, \tag{4.1}$$

where the matrix $\mathbf{X}\left(t\right) \in \mathbb{R}^{N_x \times r}$ contains the orthonormal basis functions in space and $\mathbf{V}\left(t\right) \in \mathbb{R}^{N_v \times r}$ the orthonormal basis functions in angle. The coefficients of the approximation are stored in the coupling matrix $\mathbf{S}\left(t\right) \in \mathbb{R}^{r \times r}$. The set of all matrices of the form (4.1) then constitutes a low-rank manifold that we denote by $\mathcal{M}_r$. Its tangent space at $\mathbf{f}_r\left(t\right)$ shall be denoted by $\mathcal{T}_{\mathbf{f}_r(t)}\mathcal{M}_r$. For the evolution of the low-rank factors in time we seek a solution of the minimization problem

$$\min_{\dot{\mathbf{f}}_r(t) \in \mathcal{T}_{\mathbf{f}_r(t)}\mathcal{M}_r} \left\|\dot{\mathbf{f}}_r\left(t\right) - \mathbf{F}\left(\mathbf{f}_r\left(t\right)\right)\right\|_F$$

at all times t, where $\left\|\cdot\right\|_F$ denotes the Frobenius norm. In [16] it has been shown that this minimization constraint is equivalent to determining

$$\dot{\mathbf{f}}_r\left(t\right) = \mathbf{P}\left(\mathbf{f}_r\left(t\right)\right)\mathbf{F}\left(\mathbf{f}_r\left(t\right)\right), \tag{4.2}$$

where $\mathbf{P}$ denotes the orthogonal projector onto the tangent space $\mathcal{T}_{\mathbf{f}_r(t)}\mathcal{M}_r$ that can be explicitly given as

$$\mathbf{P}\left(\mathbf{f}_r\left(t\right)\right)\mathbf{F} = \mathbf{X}\mathbf{X}^\top\mathbf{F} - \mathbf{X}\mathbf{X}^\top\mathbf{F}\mathbf{V}\mathbf{V}^\top + \mathbf{F}\mathbf{V}\mathbf{V}^\top.$$

There are different robust time integrators for the solution of (4.2) that are able to evolve the low-rank solution on the manifold $\mathcal{M}_r$ while not suffering from potentially small singular values [12]. The projector-splitting [19], the (augmented) BUG [6, 4], and the parallel BUG integrator [5] are frequently used.

In this work, we use the augmented BUG integrator from [4] that evolves the low-rank factors as follows: In the first two steps, the BUG integrator updates and augments the spatial basis $\mathbf{X}$ and the angular basis $\mathbf{V}$ in parallel, leading to an increase of rank from r to $2r$. Having the augmented bases at hand, a Galerkin step for the coefficient matrix $\mathbf{S}$ is performed. In the last step, all quantities are truncated back to a new rank $r_1 \le 2r$ that is chosen adaptively depending on a prescribed error tolerance. In detail, the augmented BUG integrator evolves the low-rank solution from $\mathbf{f}_r^n = \mathbf{X}^n \mathbf{S}^n \mathbf{V}^{n,\top}$ at time t_n to $\mathbf{f}_r^{n+1} = \mathbf{X}^{n+1} \mathbf{S}^{n+1} \mathbf{V}^{n+1,\top}$ at time $t_{n+1} = t_n + \Delta t$ as follows:

***K*-Step**: We denote $\mathbf{K}(t) = \mathbf{X}(t)\,\mathbf{S}(t)$ and solve the PDE

$$\dot{\mathbf{K}}(t) = \mathbf{F}\left(\mathbf{K}(t)\,\mathbf{V}^{n,\top}\right)\mathbf{V}^n, \quad \mathbf{K}(t_n) = \mathbf{X}^n \mathbf{S}^n.$$

The spatial basis is then updated by determining $\widehat{\mathbf{X}}^{n+1} \in \mathbb{R}^{N_x \times 2r}$ as an orthonormal basis of $[\mathbf{K}(t_{n+1}), \mathbf{X}^n] \in \mathbb{R}^{N_x \times 2r}$, e.g. by QR-decomposition. We store $\widehat{\mathbf{M}} = \widehat{\mathbf{X}}^{n+1,\top} \mathbf{X}^n \in \mathbb{R}^{2r \times r}$. Note that we denote augmented quantities of rank $2r$ with hats.

***L*-Step**: We denote $\mathbf{L}(t) = \mathbf{V}(t)\,\mathbf{S}(t)^\top$ and solve the PDE

$$\dot{\mathbf{L}}(t) = \mathbf{F}\left(\mathbf{X}^n \mathbf{L}(t)^\top\right)^\top \mathbf{X}^n, \quad \mathbf{L}(t_n) = \mathbf{V}^n \mathbf{S}^{n,\top}.$$

The angular basis is then updated by determining $\widehat{\mathbf{V}}^{n+1} \in \mathbb{R}^{N_v \times 2r}$ as an orthonormal basis of $[\mathbf{L}(t_{n+1}), \mathbf{V}^n] \in \mathbb{R}^{N_v \times 2r}$, e.g. by QR-decomposition. We store $\widehat{\mathbf{N}} = \widehat{\mathbf{V}}^{n+1,\top} \mathbf{V}^n \in \mathbb{R}^{2r \times r}$.

***S*-step**: We update the coefficient matrix from $\mathbf{S}^n \in \mathbb{R}^{r \times r}$ to $\widehat{\mathbf{S}}^{n+1} \in \mathbb{R}^{2r \times 2r}$ by solving the ODE

$$\dot{\widehat{\mathbf{S}}}(t) = \widehat{\mathbf{X}}^{n+1,\top} \mathbf{F}\left(\widehat{\mathbf{X}}^{n+1} \widehat{\mathbf{S}}(t)\,\widehat{\mathbf{V}}^{n+1,\top}\right)\widehat{\mathbf{V}}^{n+1}, \quad \widehat{\mathbf{S}}(t_n) = \widehat{\mathbf{M}} \mathbf{S}^n \widehat{\mathbf{N}}^\top.$$

Truncation: We compute the singular value decomposition of $\widehat{\mathbf{S}}^{n+1} = \widehat{\mathbf{P}} \boldsymbol{\Sigma} \widehat{\mathbf{Q}}^\top$, where $\widehat{\mathbf{P}}, \widehat{\mathbf{Q}} \in \mathbb{R}^{2r \times 2r}$ are orthogonal matrices and $\boldsymbol{\Sigma} \in \mathbb{R}^{2r \times 2r}$ is the diagonal matrix containing the singular values $\sigma_1, ..., \sigma_{2r}$. The new rank $r_1 \le 2r$ is determined such that

$$\left(\sum_{j=r_1+1}^{2r} \sigma_j^2\right)^{1/2} \le \vartheta,$$

where ϑ denotes a prescribed tolerance. We set $\mathbf{S}^{n+1} \in \mathbb{R}^{r_1 \times r_1}$ to contain the r_1 largest singular values of $\widehat{\mathbf{S}}^{n+1}$ and $\mathbf{P}^{n+1} \in \mathbb{R}^{2r \times r_1}$ and $\mathbf{Q}^{n+1} \in \mathbb{R}^{2r \times r_1}$ to contain the first r_1 columns of $\widehat{\mathbf{P}}$ and $\widehat{\mathbf{Q}}$, respectively. Finally, we compute $\mathbf{X}^{n+1} = \widehat{\mathbf{X}}^{n+1} \mathbf{P}^{n+1} \in \mathbb{R}^{N_x \times r_1}$ and $\mathbf{V}^{n+1} = \widehat{\mathbf{V}}^{n+1} \mathbf{Q}^{n+1} \in \mathbb{R}^{N_v \times r_1}$.

The update of $\mathbf{f}_r^n$ after one time step is then given by $\mathbf{f}_r^{n+1} = \mathbf{X}^{n+1} \mathbf{S}^{n+1} \mathbf{V}^{n+1,\top}$. Note that in the following, to simplify notation, we will write $\mathbf{f}$ instead of $\mathbf{f}_r$.

4.2. Dynamical low-rank approximation for the discrete optimization problem. The goal of this subsection consists in applying DLRA to the fully discrete gradient descent method proposed in Algorithm 3.1. To this end, we reformulate the forward equations (3.8) as well as the adjoint equations (3.9) using the dynamical low-rank method with augmented BUG integrator.

The initial low-rank factors $\mathbf{X}_m^{0,\text{for}}, \mathbf{S}_m^{0,\text{for}}$, and $\mathbf{V}_m^{0,\text{for}}$ for the forward equations (3.8) are obtained by a singular value decomposition of $\mathbf{u}_m^0$, where $\mathbf{u}_m^0 = \left(u_{jkm}^0\right) \in \mathbb{R}^{N_x \times N_v}$, for which the number of singular values is truncated to the initial rank r. In each time step, the low-rank factors $\mathbf{X}_m^{n,\text{for}}, \mathbf{S}_m^{n,\text{for}}$, and $\mathbf{V}_m^{n,\text{for}}$ are then evolved according to the following scheme.

First, we solve in parallel the equations

(4.3a)
$$\mathbf{K}_m^{n+1,\text{for}} = \mathbf{K}_m^{n,\text{for}} - \Delta t \mathbf{D}^x \mathbf{K}_m^{n,\text{for}} \mathbf{V}_m^{n,\text{for},\top} \mathbf{A}^\top \mathbf{V}_m^{n,\text{for}} + \Delta t \frac{\Delta x}{2} \mathbf{D}^{xx} \mathbf{K}_m^{n,\text{for}} \mathbf{V}_m^{n,\text{for},\top} |\mathbf{A}|^\top \mathbf{V}_m^{n,\text{for}}$$
$$+ \Delta t \operatorname{diag}(\sigma) \mathbf{K}_m^{n,\text{for}} \mathbf{V}_m^{n,\text{for},\top} \mathbf{E} \mathbf{V}_m^{n,\text{for}},$$

(4.3b)
$$\mathbf{L}_m^{n+1,\text{for}} = \mathbf{L}_m^{n,\text{for}} - \Delta t \mathbf{A} \mathbf{L}_m^{n,\text{for}} \mathbf{X}_m^{n,\text{for},\top} \mathbf{D}^{x,\top} \mathbf{X}_m^{n,\text{for}} + \Delta t \frac{\Delta x}{2} |\mathbf{A}| \mathbf{L}_m^{n,\text{for}} \mathbf{X}_m^{n,\text{for},\top} \mathbf{D}^{xx,\top} \mathbf{X}_m^{n,\text{for}}$$
$$+ \Delta t \mathbf{E} \mathbf{L}_m^{n,\text{for}} \mathbf{X}_m^{n,\text{for},\top} \operatorname{diag}(\sigma) \mathbf{X}_m^{n,\text{for}},$$

where $\mathbf{E} = \operatorname{diag}([0, -1, ..., -1])$. In the next step, we perform a QR-decomposition of the augmented quantities $\left[\mathbf{K}_m^{n+1,\text{for}}, \mathbf{X}_m^{n,\text{for}}\right]$ and $\left[\mathbf{L}_m^{n+1,\text{for}}, \mathbf{V}_m^{n,\text{for}}\right]$ to obtain the augmented and time updated spatial bases $\widehat{\mathbf{X}}_m^{n+1,\text{for}}$ and angular bases $\widehat{\mathbf{V}}_m^{n+1,\text{for}}$, respectively. For the S-step we introduce the notation $\widetilde{\mathbf{S}}_m^{n,\text{for}} = \widehat{\mathbf{X}}_m^{n+1,\text{for},\top} \mathbf{X}_m^{n,\text{for}} \mathbf{S}_m^{n,\text{for}} \mathbf{V}_m^{n,\text{for},\top} \widehat{\mathbf{V}}_m^{n+1,\text{for}}$ and compute

$$\begin{aligned} \widehat{\mathbf{S}}_m^{n+1,\text{for}} = \widetilde{\mathbf{S}}_m^{n,\text{for}} &- \Delta t \widehat{\mathbf{X}}_m^{n+1,\text{for},\top} \mathbf{D}^x \widehat{\mathbf{X}}_m^{n+1,\text{for}} \widetilde{\mathbf{S}}_m^{n,\text{for}} \widehat{\mathbf{V}}_m^{n+1,\text{for},\top} \mathbf{A}^\top \widehat{\mathbf{V}}_m^{n+1,\text{for}} \\ &+ \Delta t \frac{\Delta x}{2} \widehat{\mathbf{X}}_m^{n+1,\text{for},\top} \mathbf{D}^{xx} \widehat{\mathbf{X}}_m^{n+1,\text{for}} \widetilde{\mathbf{S}}_m^{n,\text{for}} \widehat{\mathbf{V}}_m^{n+1,\text{for},\top} |\mathbf{A}|^\top \widehat{\mathbf{V}}_m^{n+1,\text{for}} \\ &+ \Delta t \widehat{\mathbf{X}}_m^{n+1,\text{for},\top} \operatorname{diag}(\sigma) \widehat{\mathbf{X}}_m^{n+1,\text{for}} \widetilde{\mathbf{S}}_m^{n,\text{for}} \widehat{\mathbf{V}}_m^{n+1,\text{for},\top} \mathbf{E} \widehat{\mathbf{V}}_m^{n+1,\text{for}}. \end{aligned} \tag{4.3c}$$

Finally, we truncate the time-updated augmented low-rank factors for each $m = 1, ..., N_{IC}$ to a new rank $r_1 \leq 2r$. The time-updated numerical solutions of the forward problem are then given by $\mathbf{u}_m^{n+1} = \mathbf{X}_m^{n+1,\text{for}} \mathbf{S}_m^{n+1,\text{for}} \mathbf{V}_m^{n+1,\text{for},\top} \in \mathbb{R}^{N_x \times N_v}$.

For the adjoint equations (3.9) we perform a singular value decomposition of the end time solutions $\mathbf{w}_m^{N_t} = \left(w_{jkm}^{N_t}\right) \in \mathbb{R}^{N_x \times N_v}$, truncate to the prescribed initial rank r and obtain the low-rank factors $\mathbf{X}_m^{N_t,\text{adj}}, \mathbf{S}_m^{N_t,\text{adj}}$, and $\mathbf{V}_m^{N_t,\text{adj}}$. Then, in each step, the low-rank factors $\mathbf{X}_m^{n,\text{adj}}, \mathbf{S}_m^{n,\text{adj}}$, and $\mathbf{V}_m^{n,\text{adj}}$ are evolved backwards in time as follows.

First, we solve in parallel the equations

$$\mathbf{K}_m^{n-1,\text{adj}} = \mathbf{K}_m^{n,\text{adj}} + \Delta t \mathbf{D}^x \mathbf{K}_m^{n,\text{adj}} \mathbf{V}_m^{n,\text{adj},\top} \mathbf{A}^\top \mathbf{V}_m^{n,\text{adj}} + \Delta t \frac{\Delta x}{2} \mathbf{D}^{xx} \mathbf{K}_m^{n,\text{adj}} \mathbf{V}_m^{n,\text{adj},\top} |\mathbf{A}|^\top \mathbf{V}_m^{n,\text{adj}}$$
$$+ \Delta t \operatorname{diag}(\sigma) \mathbf{K}_m^{n,\text{adj}} \mathbf{V}_m^{n,\text{adj},\top} \mathbf{E} \mathbf{V}_m^{n,\text{adj}},$$
$$\mathbf{L}_m^{n-1,\text{adj}} = \mathbf{L}_m^{n,\text{adj}} + \Delta t \mathbf{A} \mathbf{L}_m^{n,\text{adj}} \mathbf{X}_m^{n,\text{adj},\top} \mathbf{D}^{x,\top} \mathbf{X}_m^{n,\text{adj}} + \Delta t \frac{\Delta x}{2} |\mathbf{A}| \mathbf{L}_m^{n,\text{adj}} \mathbf{X}_m^{n,\text{adj},\top} \mathbf{D}^{xx,\top} \mathbf{X}_m^{n,\text{adj}}$$
$$+ \Delta t \mathbf{E} \mathbf{L}_m^{n,\text{adj}} \mathbf{X}_m^{n,\text{adj},\top} \operatorname{diag}(\sigma) \mathbf{X}_m^{n,\text{adj}}.$$

In the next step, we perform a QR-decomposition of $\left[\mathbf{K}_m^{n-1,\text{adj}}, \mathbf{X}_m^{n,\text{adj}}\right]$ and $\left[\mathbf{L}_m^{n-1,\text{adj}}, \mathbf{V}_m^{n,\text{adj}}\right]$ to obtain the augmented and time updated spatial bases $\widehat{\mathbf{X}}_m^{n-1,\text{adj}}$ and angular bases $\widehat{\mathbf{V}}_m^{n-1,\text{adj}}$, respectively. For the S-step we set $\widetilde{\mathbf{S}}_m^{n,\text{adj}} = \widehat{\mathbf{X}}_m^{n-1,\text{adj},\top}\mathbf{X}_m^{n,\text{adj}}\mathbf{S}_m^{n,\text{adj}}\mathbf{V}_m^{n,\text{adj},\top}\widehat{\mathbf{V}}_m^{n-1,\text{adj}}$ and compute

$$
\begin{aligned}
\widehat{\mathbf{S}}_m^{n-1,\text{adj}} = \widetilde{\mathbf{S}}_m^{n,\text{adj}} &+ \Delta t \widehat{\mathbf{X}}_m^{n-1,\text{adj},\top}\mathbf{D}^x\widehat{\mathbf{X}}_m^{n-1,\text{adj}}\widetilde{\mathbf{S}}_m^{n,\text{adj}}\widehat{\mathbf{V}}_m^{n-1,\text{adj},\top}\mathbf{A}^\top\widehat{\mathbf{V}}_m^{n-1,\text{adj}} \\
&+ \Delta t\frac{\Delta x}{2}\widehat{\mathbf{X}}_m^{n-1,\text{adj},\top}\mathbf{D}^{xx}\widehat{\mathbf{X}}_m^{n-1,\text{adj}}\widetilde{\mathbf{S}}_m^{n,\text{adj}}\widehat{\mathbf{V}}_m^{n-1,\text{adj},\top}|\mathbf{A}|^\top\widehat{\mathbf{V}}_m^{n-1,\text{adj}} \\
&+ \Delta t\widehat{\mathbf{X}}_m^{n-1,\text{adj},\top}\operatorname{diag}(\sigma)\widehat{\mathbf{X}}_m^{n-1,\text{adj}}\widetilde{\mathbf{S}}_m^{n,\text{adj}}\widehat{\mathbf{V}}_m^{n-1,\text{adj},\top}\mathbf{E}\widehat{\mathbf{V}}_m^{n-1,\text{adj}}.
\end{aligned}
$$

Finally, we truncate the time-updated augmented low-rank factors for each $m = 1, ..., N_{IC}$ to a new rank $r_1 \leq 2r$. The time-updated numerical solutions of the adjoint problem are then given by $\mathbf{w}_m^{n-1} = \mathbf{X}_m^{n-1,\text{adj}}\mathbf{S}_m^{n-1,\text{adj}}\mathbf{V}_m^{n-1,\top,\text{adj}} \in \mathbb{R}^{N_x \times N_v}$.

Having determined the low-rank solutions of the forward and the adjoint problems, we can use them to compute the gradient as given in (3.10). For the update of the coefficients according to (2.5) we determine the step size adaptively by a line search approach with Armijo condition similar to [23] and as described in Algorithm 4.1. For a given step size η^n the coefficients and the scattering coefficient are updated to $\mathbf{c}^{n+1}$ and $\boldsymbol{\sigma}^{n+1}$, respectively. Then, the truncation error tolerance ϑ is adjusted using the given step size η^n and the maximal absolute value of $\nabla_{\mathbf{c}^n} J$. We add some safety parameters h_2 and h_3 as well as a lower bound h_1 for the truncation tolerance. In the next step, we compute the value of the goal function J with the low-rank factors of the forward problem at hand. We then solve the forward problem (4.3) with $\boldsymbol{\sigma}^{n+1}$ and the updated ϑ to evaluate the goal function J again with the obtained low-rank factors. While the difference between those values of the goal function J is larger than a prescribed tolerance, the gradient descent step size is reduced by the factor p and the procedure is repeated.

5. Numerical results.

We consider the following test examples in one space and one angular dimension to show the computational accuracy and efficiency of the proposed low-rank scheme.

5.1. Cosine.

For the first numerical experiment the spatial as well as the angular domain shall be set to $\Omega_x = \Omega_v = [-1, 1]$. We consider $N_{\text{IC}} = 3$ initial distributions of Cosine type of the form

$$
u_m(t = 0, x) = 2 + \cos\left(\left(x - \frac{2m}{3}\right)\pi\right) \quad \text{for} \quad m = 1, 2, 3.
$$

The true and the initial spline coefficients for the approximation of the scattering coefficient σ are chosen as

$$
\mathbf{c}_{\text{true}} = (2.1, 2.0, 2.2)^\top \quad \text{and} \quad \mathbf{c}_{\text{init}} = (1.0, 1.5, 3.0)^\top.
$$

The order of the spline basis functions is set to 3, i.e. cubic periodic B-splines are considered. As computational parameters we use $N_x = 100$ cells in the spatial domain and $N_v = 250$ moments for the approximation in the angular variable. The end time is set to $T = 1.0$ and

Algorithm 4.1 Line search method for refining the gradient descent step size and the DLRA rank tolerance

Input: goal function J,
coefficients $\mathbf{c}^n$,
gradient $\nabla_{\mathbf{c}^n} J$ computed using (3.10),
low-rank factors $\mathbf{X}_m^{n,\text{for}}, \mathbf{S}_m^{n,\text{for}}, \mathbf{V}_m^{n,\text{for}}$ of the forward problem (4.3) for $m = 1, ..., N_{\text{IC}}$,
step size η^n,
rank error tolerance ϑ,
step size reduction factor p,
constants h_1, h_2, h_3, h_4

Output: refined step size η^{n+1}, refined rank error tolerance ϑ, updated coefficients $\mathbf{c}^{n+1}$

Update the coefficients: $\mathbf{c}^{n+1} = \mathbf{c}^n - \eta^n \nabla_{\mathbf{c}^n} J$;
Compute $\boldsymbol{\sigma}^{n+1}$ from coefficients $\mathbf{c}^{n+1}$ according to (2.4);
Update $\vartheta = \max\left(h_1, \min\left(h_2, h_3 \left\|\nabla_{\mathbf{c}^n} J\right\|_\infty \eta^n\right)\right)$;

Compute $J^n = J\left(\mathbf{X}_1^{n,\text{for}}\mathbf{S}_1^{n,\text{for}}\mathbf{V}_1^{n,\text{for}}, ..., \mathbf{X}_{N_{\text{IC}}}^{n,\text{for}}\mathbf{S}_{N_{\text{IC}}}^{n,\text{for}}\mathbf{V}_{N_{\text{IC}}}^{n,\text{for}}\right)$;
Compute $\overline{\mathbf{X}}_m^{n,\text{for}}\overline{\mathbf{S}}_m^{n,\text{for}}\overline{\mathbf{V}}_m^{n,\text{for}}$ from (4.3) for $m = 1, ..., N_{\text{IC}}$ with $\boldsymbol{\sigma} = \boldsymbol{\sigma}^{n+1}$ and the updated ϑ;
Compute $\overline{J}^n = J\left(\overline{\mathbf{X}}_1^{n,\text{for}}\overline{\mathbf{S}}_1^{n,\text{for}}\overline{\mathbf{V}}_1^{n,\text{for}}, ..., \overline{\mathbf{X}}_{N_{\text{IC}}}^{n,\text{for}}\overline{\mathbf{S}}_{N_{\text{IC}}}^{n,\text{for}}\overline{\mathbf{V}}_{N_{\text{IC}}}^{n,\text{for}}\right)$;

while $\overline{J}^n > J^n - \eta^n h_4 \left\|\nabla_{\mathbf{c}^n} J\right\|_2^2$ **do**
  Update $\eta^{n+1} = p\eta^n$;

  Update $\mathbf{c}^{n+1} = \mathbf{c}^{n+1} - \eta^{n+1} \nabla_{\mathbf{c}^n} J$;
  Compute $\boldsymbol{\sigma}^{n+1}$ from updated coefficients $\mathbf{c}^{n+1}$;
  Update $\vartheta = \max\left(h_1, \min\left(h_2, h_3 \left\|\nabla_{\mathbf{c}^n} J\right\|_\infty \eta^{n+1}\right)\right)$;

  Compute $\overline{\mathbf{X}}_m^{n,\text{for}}\overline{\mathbf{S}}_m^{n,\text{for}}\overline{\mathbf{V}}_m^{n,\text{for}}$ from (4.3) for $m = 1, ..., N_{\text{IC}}$ with $\boldsymbol{\sigma} = \boldsymbol{\sigma}^{n+1}$ and the updated ϑ;
  Compute $\overline{J}^n = J\left(\overline{\mathbf{X}}_1^{n,\text{for}}\overline{\mathbf{S}}_1^{n,\text{for}}\overline{\mathbf{V}}_1^{n,\text{for}}, ..., \overline{\mathbf{X}}_{N_{\text{IC}}}^{n,\text{for}}\overline{\mathbf{S}}_{N_{\text{IC}}}^{n,\text{for}}\overline{\mathbf{V}}_{N_{\text{IC}}}^{n,\text{for}}\right)$;
end while

Set $\eta^{n+1} = \eta^n$;

the time step size of the algorithm is chosen such that $\Delta t = \text{CFL} \cdot \Delta x$ with a CFL number of $\text{CFL} = 0.99$. For the low-rank computations we start with an initial rank of $r = 5$ in the forward as well as in the adjoint problem. The maximal allowed value of the rank in each step shall be restricted to 20. We begin the gradient descent method with a step size of $\eta^0 = 5 \cdot 10^5$ and a truncation error tolerance of $\vartheta = 10^{-2} \left\|\boldsymbol{\Sigma}\right\|_2$. For the rescaling of the gradient descent step size and the DLRA rank tolerance we use the step size reduction factor $p = 0.5$ as well as the constants $h_1 = 10^{-3} \left\|\boldsymbol{\Sigma}\right\|_2$ for a lower bound of the rank tolerance and $h_2 = 0.1, h_3 = 0.1$ as safety parameters. Also $h_4 = 0.5$ is added as a safety parameter to ensure a reasonable difference between $\overline{J}^n$ and J^n in Algorithm 4.1. The whole gradient descent procedure is then conducted until the prescribed error tolerance `errtol` $= 10^{-4}$ or a maximal number of iterations `maxiter` $= 500$ is reached.

In Figure 2 we compare the solutions of the parameter identification problem computed

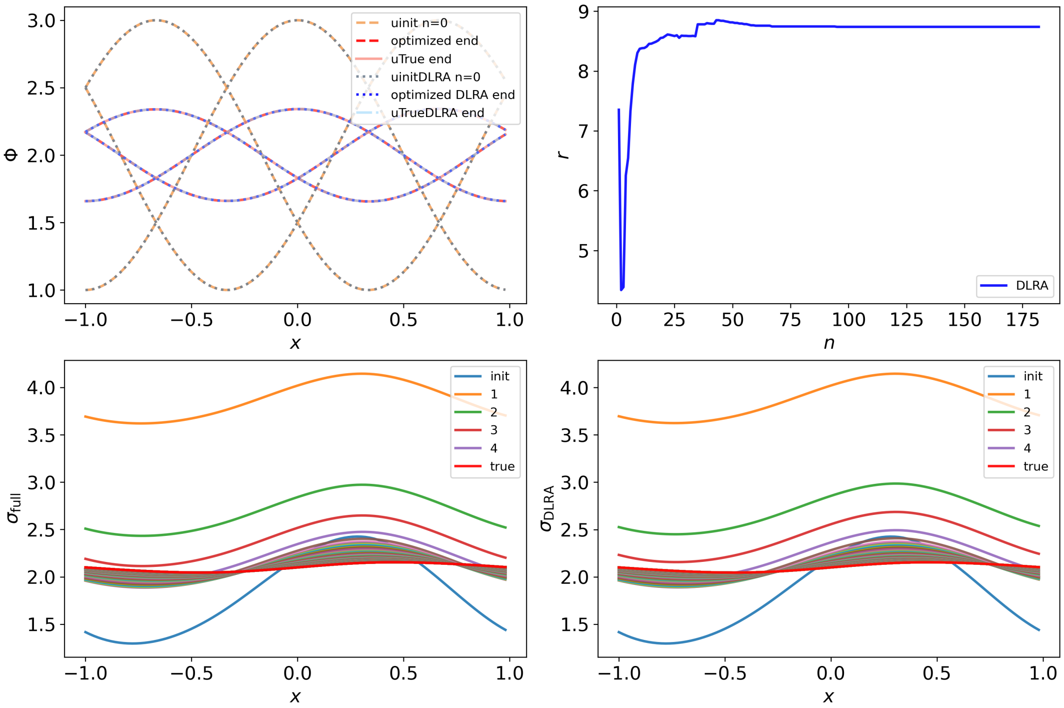

Figure 2. *Top left: Numerical results for the scalar flux Φ of the Cosine problem computed with the full solvers and the DLRA solvers at the initial step $n = 0$, with the true coefficients and with the optimization gradient descent scheme. Top right: Evolution of the averaged rank r for the DLRA method. Bottom row: Iterations for the reconstruction of the scattering coefficient σ computed with both the full solvers (left) and the DLRA solvers (right).*

with the full solvers and the DLRA solvers for both the forward and the adjoint equations. We plot three curves corresponding to the different initial conditions of the scalar flux $\Phi = \frac{1}{\sqrt{2}} \langle f \rangle_v$ at the initial step (uinit n = 0), computed with the true coefficients (u True end) and at the end of the optimization procedure (optimized end), evaluated with both the full and the DLRA solver. We observe that the DLRA solution captures well the behavior of the full solution and that they both approach the solutions computed with the true coefficients. In addition, the parameter reconstruction inverse problem for determining σ is resolved accurately with both solvers. It can be seen that beginning with $\boldsymbol{\sigma}_{\text{init}}$ both the full and the DLRA method converge to the true solution $\boldsymbol{\sigma}_{\text{true}}$. Further, the evolution of the rank r is depicted, where we have averaged the ranks of the forward equations computed with the different initial conditions to obtain r_{for} and the ranks of the adjoint equations computed with the different initial conditions to obtain r_{adj} and finally set $r = \frac{1}{2}\left(r_{\text{for}} + r_{\text{adj}}\right)$. We observe that in the beginning the averaged rank decreases as the initial rank was chosen larger than required. From then on, we observe a relatively monotonous increase until it stays at approximately $r = 9$. This evolution of the rank reflects the fact that in the beginning of the optimization the error tolerance ϑ is chosen quite large as the computed solution is still comparably far away from the true solution. As the optimization algorithm approaches the true coefficients, the DLRA rank tolerance ϑ is

This manuscript is for review purposes only.

decreased, resulting in a higher averaged rank. For the considered setup, the computational benefit of the DLRA method compared to the solution of the full problem is significant. Written in Julia v1.11 and run on a MacBook Pro with M1 chip, the run time decreases by a factor of approximately 2.5 from 139 seconds to 56 seconds while retaining the accuracy of the computed results. Concerning the memory costs, the solutions of the forward problem and of the adjoint problem have to be stored in order to compute the gradient. For each initial condition, the storage of the solution of the forward problem corresponds to a memory cost of $8\,(N_t+1)\,N_xN_v$, which for the DLRA method can be lowered to $8\,(N_t+1)\left(rN_x+rN_v+r^2\right)$, where r is the maximal averaged rank in the simulation.

5.2. Gaussian distribution. In a second test example, we set $\Omega_x=[0,10]$ for the spatial and $\Omega_v=[-1,1]$ for the angular domain. We consider $N_{\mathrm{IC}}=5$ Gaussian initial distributions of the form

$$u_m\,(t=0,x)=\max\left(10^{-8},\frac{1}{\sqrt{2\pi\sigma_{\mathrm{IC}}^2}}\exp\left(-\frac{(x-x_0)^2}{2\sigma_{\mathrm{IC}}^2}\right)\right)\quad\text{for}\quad m=1,2,3,4,5,$$

that are centered around equidistantly distributed x_0 and extended periodically on the domain Ω_x. The standard deviation is set to the constant value $\sigma_{\mathrm{IC}}=0.8$. The true and the initial spline coefficients for the approximation of the scattering coefficient σ are chosen as

$$\mathbf{c}_{\mathrm{true}}=(2.1,2.0,2.2,2.0,1.9)^\top\quad\text{and}\quad\mathbf{c}_{\mathrm{init}}=(2.8,1.5,3.0,2.1,1.2)^\top.$$

All other settings and computational parameters remain unchanged from the previous test example.

In Figure 3 we compare the solutions of the parameter identification problem computed with the full solvers and the DLRA solvers for both the forward and the adjoint equations. We plot five curves corresponding to the different initial conditions of the scalar flux $\Phi=\frac{1}{\sqrt{2}}\langle f\rangle_v$ at the initial step (uinit n = 0), computed with the true coefficients (u True end) and at the end of the optimization procedure (optimized end), evaluated with both the full and the DLRA solver. Again we observe that the DLRA solution captures well the behavior of the full solution and that they both approach the solutions computed with the true coefficients. For the reconstruction of the scattering coefficient σ it can be seen that beginning with $\boldsymbol{\sigma}_{\mathrm{init}}$ both the full and the DLRA method converge to the true solution $\boldsymbol{\sigma}_{\mathrm{true}}$. The averaged rank r first decreases as the initial rank was chosen larger than required. From then on, we observe the expected relatively monotonous increase until it stagnates at a value of approximately $r=11.5$. Written in Julia v1.11 and run on a MacBook Pro with M1 chip, the computational time of the DLRA method compared to the solution of the full problem decreases by a factor of approximately 2 from 11.5 seconds to 6 seconds, showing its computational efficiency.

6. Conclusion and outlook. We have presented a fully discrete DLRA scheme for the reconstruction of the scattering parameter in the radiative transfer equation making use of a PDE constrained optimization procedure. For a further enhancement of its computational advantages compared to a standard full solver, the step size of the gradient descent method

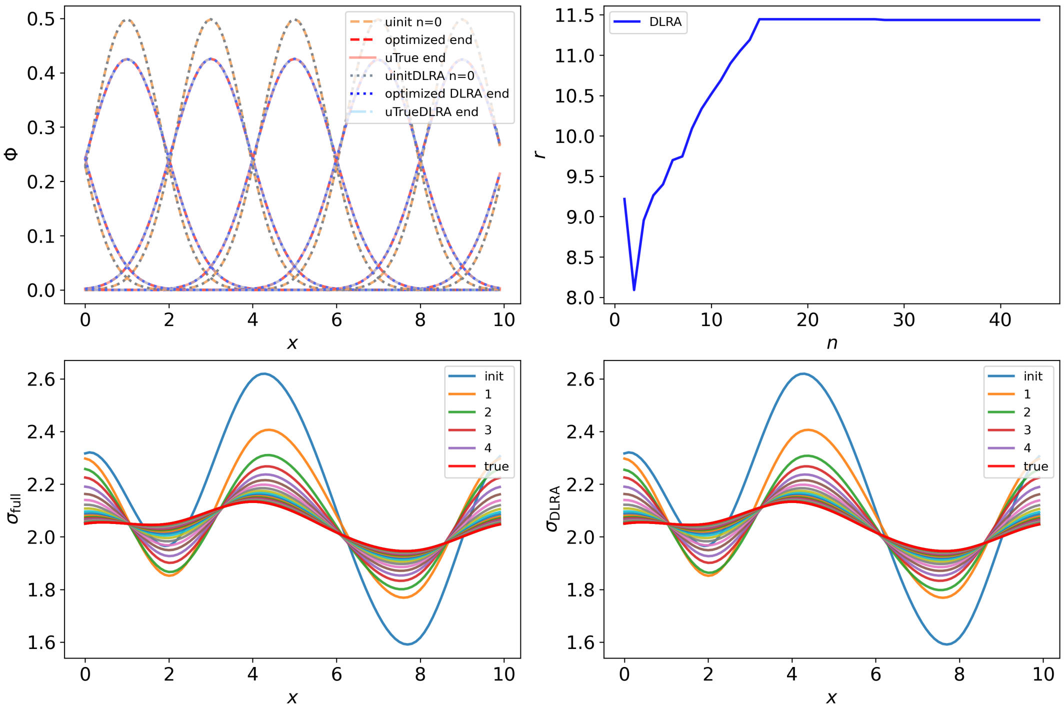

Figure 3. *Top left: Numerical results for the scalar flux Φ of the Gauss problem computed with the full solvers and the DLRA solvers at the initial step $n = 0$, with the true coefficients and with the optimization gradient descent scheme. Top right: Evolution of the averaged rank r for the DLRA method. Bottom row: Iterations for the reconstruction of the scattering coefficient σ computed with both the full solvers (left) and the DLRA solvers (right).*

is determined adaptively in each step and the allowed DLRA rank tolerance is adjusted accordingly. This leads to an efficient and accurate numerical DLRA scheme. For further considerations, numerical examples in more than one spatial and angular variable are of interest as in higher dimensions the savings by the DLRA method are expected to be larger by orders of magnitude. Also, theoretical considerations concerning for instance the stability of DLRA schemes applied to inverse parameter reconstruction problems can provide valuable insights into the structure of such problems. In addition, various questions arise when the structural order of the problem is changed, meaning that for example a "first low-rank, then optimize, then discretize" strategy is pursued. For instance, it is not clear how the adjoint equations can be derived from the low-rank components of the forward problem as the low-rank equations are highly nonlinear. Summarizing, the combination of DLRA methods and parameter identification problems is an interesting field of research with various open problems that are left to future work.

Acknowledgments. The authors thank Kui Ren for helpful and inspiring discussions.

REFERENCES

[1] G. Bal, *Inverse transport theory and applications*, Inverse problems, 25 (2009), p. 053001.

[2] L. Baumann, L. Einkemmer, C. Klingenberg, and J. Kusch, *Energy stable and conservative dynamical low-rank approximation for the Su-Olson problem*, SIAM Journal on Scientific Computing, 46 (2024), pp. B137–B158.

[3] K. M. Case and P. F. Zweifel, *Linear Transport Theory*, Addison-Wesley Series in Nuclear Engineering, Addison-Wesley, Reading, MA, 1967.

[4] G. Ceruti, J. Kusch, and C. Lubich, *A rank-adaptive robust integrator for dynamical low-rank approximation*, BIT Numerical Mathematics, 62 (2022), pp. 1149–1174.

[5] G. Ceruti, J. Kusch, and C. Lubich, *A parallel rank-adaptive integrator for dynamical low-rank approximation*, SIAM Journal on Scientific Computing, 46 (2024), pp. B205–B228.

[6] G. Ceruti and C. Lubich, *An unconventional robust integrator for dynamical low-rank approximation*, BIT Numerical Mathematics, 62 (2022), pp. 23–44.

[7] K. Chen, Q. Li, and J.-G. Liu, *Online learning in optical tomography: a stochastic approach*, Inverse Problems, 34 (2018), p. 075010.

[8] L. Einkemmer, J. Hu, and L. Ying, *An efficient dynamical low-rank algorithm for the Boltzmann-BGK equation close to the compressible viscous flow regime*, SIAM Journal on Scientific Computing, 43 (2021), pp. B1057–B1080.

[9] L. Einkemmer, Q. Li, L. Wang, and Y. Yang, *Suppressing instability in a Vlasov–Poisson system by an external electric field through constrained optimization*, Journal of Computational Physics, 498 (2024), p. 112662.

[10] L. Einkemmer and C. Lubich, *A low-rank projector-splitting integrator for the Vlasov-Poisson equation*, SIAM Journal on Scientific Computing, 40 (2018), pp. B1330–B1360.

[11] K. Hellmuth, C. Klingenberg, Q. Li, and M. Tang, *Reconstructing the kinetic chemotaxis kernel using macroscopic data: Well-posedness and ill-posedness*, SIAM Journal on Applied Mathematics, 85 (2025), pp. 613–635.

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

[13] A. Kirsch, *An Introduction to the Mathematical Theory of Inverse Problems*, vol. 120 of Applied Mathematical Sciences, Springer, Cham, third ed., 2021.

[14] A. D. Klose and A. H. Hielscher, *Optical tomography using the time-independent equation of radiative transfer – Part 2: inverse model*, Journal of Quantitative Spectroscopy & Radiative Transfer, 72 (2002), pp. 715–732.

[15] A. D. Klose, U. Netz, J. Beuthan, and A. H. Hielscher, *Optical tomography using the time-independent equation of radiative transfer – Part 1: forward model*, Journal of Quantitative Spectroscopy & Radiative Transfer, 72 (2002), pp. 691–713.

[16] O. Koch and C. Lubich, *Dynamical low-rank approximation*, SIAM Journal on Matrix Analysis and Applications, 29 (2007), pp. 434–454.

[17] J. Kusch and P. Stammer, *A robust collision source method for rank adaptive dynamical low-rank approximation in radiation therapy*, ESAIM: M2AN, 57 (2023), pp. 865–891.

[18] Q. Li, L. Wang, and Y. Yang, *Monte Carlo gradient in optimization constrained by radiative transport equation*, SIAM Journal on Numerical Analysis, 61 (2023), pp. 2744–2774.

[19] C. Lubich and I. V. Oseledets, *A projector-splitting integrator for dynamical low-rank approximation*, BIT Numerical Mathematics, 54 (2014), pp. 171–188.

[20] R. G. McClarren, J. P. Holloway, and T. A. Brunner, *On solutions to the P_n equations for thermal radiative transfer*, Journal of Computational Physics, 227 (2008), pp. 2864–2885.

[21] G. C. Pomraning, *The Equations of Radiation Hydrodynamics*, vol. 54 of International series of monographs in natural philosophy, Pergamon Press, Oxford, 1973.

[22] K. Ren, G. Bal, and A. H. Hielscher, *Transport- and diffusion-based optical tomography in small domains: a comparative study*, Applied Optics, 46 (2007), pp. 6669–6679.

[23] C. Scalone, L. Einkemmer, J. Kusch, and R. G. McClarren, *A multi-fidelity adaptive dynamical low-rank based optimization algorithm for fission criticality problems*, arXiv preprint arXiv:2409.14938, (2024).

[24] P. Stefanov, *Inverse problems in transport theory*, in Inside Out: Inverse Problems and Applications, G. Uhlmann, ed., vol. 47 of Mathematical Sciences Research Institute Publications, Cambridge University Press, Cambridge, 2003, pp. 111–131.