SPARSE GRID APPROXIMATION OF THE RICCATI OPERATOR FOR CLOSED LOOP PARABOLIC CONTROL PROBLEMS WITH DIRICHLET BOUNDARY CONTROL

HELMUT HARBRECHT* AND ILJA KALMYKOV*

Abstract. We consider the sparse grid approximation of the Riccati operator P arising from closed loop parabolic control problems. In particular, we concentrate on the linear quadratic regulator (LQR) problems, i.e. we are looking for an optimal control u_{opt} in the linear state feedback form $u_{\text{opt}}(t,\cdot) = Px(t,\cdot)$, where $x(t,\cdot)$ is the solution of the controlled partial differential equation (PDE) for a time point t. Under sufficient regularity assumptions, the Riccati operator P fulfills the algebraic Riccati equation (ARE)

$$AP + PA - PBB^{\star}P + Q = 0,$$

where A, B, and Q are linear operators associated to the LQR problem. By expressing P in terms of an integral kernel p, the weak form of the ARE leads to a non-linear partial integro-differential equation for the kernel p – the Riccati-IDE. We represent the kernel function as an element of a sparse grid space, which considerably reduces the cost to solve the Riccati IDE. Numerical results are given to validate the approach.

Key words. Optimal control, Riccati equation, sparse grid approximation

1. Introduction.

Operator Riccati differential equations play an important role in a number of different applications in engineering, physics, and mathematics. To give a few examples, we mention model reduction ([30, 22]), filtering ([31]), scattering theory ([42]), radiative transfer and the solution of two point boundary value problems via the theory of invariant embedding ([2]). A well-known application of the Riccati equation stems from the optimal control theory, in particular from the unconstrained linear quadratic (LQ) optimal control of parabolic partial differential equations, see e.g. [2, 6, 37, 40] and the references therein. In Section 2, we consider unconstrained LQ optimal control for infinite time horizon. In this case, the optimal control can be obtained by solving the algebraic Riccati equation (ARE). We refer to the solution of the ARE as Riccati operator P.

A number of methods is available for the approximation of the ARE (see e.g. [5] or [7] for a survey). One basic approach is to approximate the operator ARE directly (cf. [23, 47, 50]). In this article, we follow the method considered in [10, 29]. Therein, the representation of P in terms of a kernel function $p(x,\xi)$ is considered:

$$(Pu)(x) = \int_{\Omega} p(x,\xi)u(\xi)\,\mathrm{d}\xi.$$

By this means, the solution of the ARE can be characterized via an integro-differential equation of Riccati type (Riccati-IDE) for the kernel $p(x,\xi)$. We present the derivation of the Riccati-IDE for the Dirichlet boundary control of the heat equation in Section 3.

The Riccati-IDE is a non-linear equation with a non-linearity in form of a quadratic term. A number of methods for the solution of non-linear equations, which have been studied for the ARE (see e.g. [3, 4, 5, 34] for a survey), can similarly be implemented for the Riccati-IDE. In this article, we apply Newton's method as suggested in [33]. We describe this approach for the discretization of the Riccati-IDE with a standard finite elements method in Section 4.

*Departement für Mathematik und Informatik, Universität Basel, Spiegelgasse 1, 4051 Basel {helmut.harbrecht,ilja.kalmykov}@unibas.ch

As the Riccati operator P is a linear operator on the state space with domain Ω, the kernel $p(x, \xi)$ is defined on the product domain $\Omega \times \Omega$. Provided we use N degrees of freedom for the discretization of the state space, the discretization of the kernel by a regular tensor product approach $p(x, \xi)$ amounts to N^2 degrees of freedom. This leads in general to a cubic over-all complexity $\mathcal{O}(N^3)$ for the evaluation of the right-hand side and the computation of the gradient in the Newton's method.

The $\mathcal{O}(N^3)$-complexity is a major bottleneck in the numerical treatment of the LQ optimal control problems and large scale AREs. At least for $d = 3$ spatial dimensions, the quadratic growth of the memory requirements makes the discretization in the regular tensor product space prohibitively expensive if not even impossible. This is an example of a more general problem known as curse of dimensionality. At the same time theoretical results on the regularity of the Riccati operator (cf. e.g. [46]) suggest that more efficient numerical schemes for the ARE can be constructed for special cases of the LQ optimal control problem.

Different approaches, like e.g. multigrid methods [20] low-rank techniques or $\mathcal{H}$-matrices [21] have been studied to overcome the $\mathcal{O}(N^3)$-complexity of the numerical approximation of the ARE. In the present article, we discretize the Riccati-IDE in the *sparse* tensor product space – a numerical technique, which allows to overcome the curse of dimensionality to some extend. Thus, the kernel $p(x, \xi)$ is represented by only $\mathcal{O}(N \log N)$ degrees of freedom, which in turn improves the over-all complexity. We will introduce the sparse tensor product space and the corresponding discretization of the Riccati-IDE in Section 5.

As shown in [35], a conforming discretization of the homogeneous Dirichlet boundary condition will not lead to the full rate of convergence. Hence, we briefly outline the Nitsche approximation [45] in Section 6. It is a nonconforming discretization and yields the full rate of convergence of the solution to the Riccati-IDE.

In Section 7, we verify our approach by numerical experiments, in which convergence rates for the approximation of the Riccati kernel $p(x, \xi)$ as well as the computational complexity are considered. Finally, in Section 8, we state concluding remarks.

2. LQR Dirichlet boundary control.

This section briefly describes the main ideas of the linear quadratic (LQ) optimal control of partial differential equations. A detailed discussion of this topic can be found e.g. in [6, 40, 51].

2.1. Parabolic equation with Dirichlet boundary control.

We consider a parabolic equation on the domain $\Omega \subset \mathbb{R}^d$ with Dirichlet boundary control

$$\begin{cases} \dfrac{\partial}{\partial t} z(t,x) + L z(t,x) = 0 & \text{in } \Omega \times (0,T], \\ \qquad\qquad z(0,x) = z_0(x) & \text{for } x \in \Omega, \\ \qquad\qquad z(t,x) = u(t) & (x,t) \in \Sigma = \Gamma \times [0,T]\,, \end{cases} \tag{2.1}$$

where L is a second order uniformly elliptic differential operator with smooth coefficients and symmetric principal part (cf. [16, p. 442] and [37, p. 183]). We assume that $\Omega \subset \mathbb{R}^n$ has a Lipschitz boundary $\Gamma = \partial\Omega$ (cf. [44, p. 5]). Moreover, the initial condition satisfies $z_0 \in L^2(\Omega)$ and $u \in L^2(\Sigma)$ is the given control function. Note that, under these assumptions, the existence and uniqueness of the solution to (2.1) in $L^2\big((0,T);\Omega\big)$ can be shown, e.g., by the method of transposition (cf. [40, Chapter III, Section 9] or [11]). Here, following [6, 13, 38, 36], we will interpret (2.1) as an abstract differential equation. To this end, we first introduce some notation.

Let $\mathcal{H}$, $\mathcal{U}$, $\mathcal{Y}$ be Hilbert spaces of states, controls, and observations, respectively.

In the particular case of Dirichlet control for the equation (2.1), we set

$$\mathcal{H} = L^2(\Omega), \quad \mathcal{U} = L^2(\Gamma), \quad \mathcal{Y} = \mathbb{R}.$$

The abstract differential equation corresponding to the system (2.1) reads

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}t} z(t) = Az(t) + Bu(t), \quad t \in (0,T], \\ z(0) = z_0, \end{cases} \tag{2.2}$$

where

$$u \in L^2\big((0,T);\mathcal{U}\big), \;\; z_0 \in \mathcal{H}.$$

The derivative $\frac{\mathrm{d}}{\mathrm{d}t}$ is interpreted in a vector distributional sense, cf. [6, pp. 87 & 202] and [51, p. 117].

In the following, we shall assume (cf. [6, p. 517] or [37, p. 122]) that:

(A1) A is the infinitesimal generator of a strongly continuous analytic semigroup e^{At} of type ω on $\mathcal{H}$. The semigroup e^{At} is generally unstable, i.e. we may assume $\omega > 0$.

(A2) $B \in \mathcal{L}(\mathcal{U}, [\mathcal{D}(A^\star)]')$. Moreover, for some $0 \leq \gamma < 1$

$$\hat{A}^{-\gamma} B \in \mathcal{L}(\mathcal{U}, \mathcal{H}),$$

where $\hat{A} = \lambda - A$ for $\lambda \in \rho(A)$, $\lambda > \omega$.

With the assumptions (A1) and (A2) at hand, we can use results on existence, uniqueness and regularity of the solution for abstract differential equations ([6, Part II Chapter 3] or [48]), respectively, for the quadratic optimal control over the infinite time horizon ([6, Part V Chapter 2] or [37, Chapter 2]), which is considered in Section 2.

In order to obtain an abstract differential equation corresponding to the problem (2.1), we define the linear operator A as

$$A : \mathcal{D}(A) \subset \mathcal{H} \to \mathcal{H}, \;\; Az = \sum_{i,j=1}^{d} \partial_i (a_{i,j}(x) \partial_j z) + \sum_{i=1}^{d} b_i(x) \partial_i z + c(x) z, \tag{2.3}$$

where $\mathcal{D}(\mathcal{A}) = H_0^1(\Omega) \cap H^2(\Omega)$. We assume coefficients $a_{i,j}(x)$, $b_i(x)$ and $c(x)$ to be smooth and the coefficients $a_{i,j}(x)$ to be symmetric. Moreover, the operator A is a generator of a strongly continuous analytic semigroup over $\mathcal{H}$ (cf. [15, Chapter XIV]), so (A1) is fulfilled.

Following [37, p. 183], we will consider the control operator of the form

$$B = -AD, \quad B : \mathcal{U} \to [\mathcal{D}(A^\star)]'. \tag{2.4}$$

The operator D in (2.4) is the Dirichlet mapping defined as the extension of the Green mapping $G : H^{\frac{1}{2}}(\Gamma) \to H^1(\Omega)$ for the problem

$$\begin{cases} Aw = 0 & \text{in } \Omega, \\ w = g & \text{on } \Gamma, \end{cases}$$

cf. [6, p. 436, 37, p. 181, 44, p. 254]. In other words, we have

$$D : \mathcal{U} \to \mathcal{H}, \quad v \mapsto Dv = w, \;\text{ where } Aw = 0 \text{ in } \Omega, \; w = v \text{ on } \Gamma. \tag{2.5}$$

The operator A in (2.4) is the isomorphic extension of the operator A in (2.3) to $\mathcal{H} \to [\mathcal{D}(A^\star)]'$ (cf. [37, pp. 6 & 181] or [6, p. 181]).

The Dirichlet mapping (2.5) is continuous from $\mathcal{U}$ to $\mathcal{D}(\hat{A}^{\frac{1}{4}-\varepsilon})$, $\varepsilon > 0$ (cf. [36, Remark 6.4] or [12]). For this reason, we obtain $\hat{A}^{-\gamma}B \in \mathcal{L}(\mathcal{U},\mathcal{H})$ with $\gamma = \frac{3}{4} + \varepsilon$, so assumption (A2) is fulfilled.

With these observations regarding the control operator B we can rewrite the problem (2.1) as

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}t} z(t) = Az(t) - ADu(t), \quad t \in (0,T]\,, \\ \quad z(0) = z_0, \end{cases} \tag{2.6}$$

where $u \in L^2\big((0,T);\mathcal{U}\big)$, $z_0 \in \mathcal{H}$, D as in (2.5), and $A : \mathcal{H} \to [\mathcal{D}(A^\star)]'$ being the extension of (2.3). According to [6, Part II, Chapter 3], there exists a unique solution

$$z \in \left\{ v \in L^2\big((0,T);\mathcal{H}\big) : \frac{\mathrm{d}v}{\mathrm{d}t} \in L^2\big((0,T);[\mathcal{D}(A^\star)]'\big) \right\}$$

for abstract differential equations of the type (2.6).

2.2. Optimal control problem. We introduce the following quadratic cost functional for the abstract differential equation (2.6)

$$J_\infty(u) \coloneqq \int_0^\infty \left\{ \|Cz(t)\|_{\mathcal{Y}}^2 + \|u(t)\|_{\mathcal{U}}^2 \right\} \mathrm{d}t,$$

where $C \in \mathcal{L}(\mathcal{H},\mathcal{Y})$ is an observation operator. As we consider the case $T \to \infty$, further assumptions on the existence of a control $u \in L^2\big((0,\infty);\mathcal{U}\big)$ with $J_\infty(u) < \infty$ has to be made. Such a control is called admissible. If there exists an admissible control for each initial state z_0, the system (2.6) is called C-stabilizable, cf. [6, p. 517]. For C-stabilizable systems, we can consider the unconstrained (i.e. with respect to the control space) linear quadratic optimal control problem for the heat equation with Dirichlet boundary control

$$\begin{cases} \displaystyle\min_{u \in L^2((0,\infty);\mathcal{U})} J_\infty(u) \\ \text{subject to system (2.6).} \end{cases} \tag{2.7}$$

The optimal control u_{opt} to the problem (2.7) is given by the feedback formula (cf. [6, Part V, Chapter 2], [18, 38] and [40, Chapter III, Section 4])

$$u_{\mathrm{opt}}(t) = -B^\star P z_{\mathrm{opt}}(t),$$

where $B^\star$ is the adjoint of the control operator B, z_{opt} is the solution of the closed loop system (see e.g. [6, p. 518]) and P is the unique solution of the algebraic Riccati equation (ARE):

$$A^\star P + PA - PBB^\star P + C^\star C = 0. \tag{2.8}$$

It can be shown that P –the Riccati operator– is a positive, self-adjoint, and bounded operator on the state space $\mathcal{H}$. By this result, we can proceed with solving the ARE (2.8) to obtain the solution to the optimization problem (2.7).

3. Riccati partial integro-differential equation. There are different approaches to the solution of equation (2.8) (see, e.g., [10], [29, Chapters 3 & 4], [40, Chapter 3], and [39, 43]). In this article, we concentrate on the representation of the Riccati operator in terms of a kernel function

$$[P\phi](x) = \int_\Omega p(x,\xi)\phi(\xi)\,\mathrm{d}\xi, \tag{3.1}$$

where in general $p(x,\xi)$ is a distribution on $\Omega \times \Omega$ (cf. [40, Chapter III, Section 5]). The existence of such a kernel is guaranteed by the Schwartz kernel theorem.

3.1. Variational formulation. Next, we want to combine (3.1) with the weak form of the ARE (2.8):

$$(A\phi, P\psi) + (P\phi, A\psi) - (B^\star P\phi, B^\star P\psi)_{\mathcal{U}} + (C^\star C\phi, \psi) = 0 \text{ for all } \phi, \psi \in \mathcal{D}(A). \tag{3.2}$$

For the sake of brevity, here and in the following, $(\cdot,\cdot)$ denotes the scalar product in the state space $\mathcal{H}$, while $(\cdot,\cdot)_{\mathcal{U}}$ denotes the scalar product in $\mathcal{U}$. In addition, we shall assume that $p \in H^1(\Omega\times\Omega)$. Then, for all $\phi(x), \psi(\xi) \in C_0^\infty(\Omega)$, we obtain

$$\begin{aligned}(A\phi, P\psi) = (\phi, A^\star P\psi) &= \int_\Omega \phi(x) A^\star \int_\Omega p(x,\xi)\psi(\xi)\,\mathrm{d}x\,\mathrm{d}\xi \\ &= \int_\Omega\int_\Omega \phi(x) A_x^\star p(x,\xi)\psi(\xi)\,\mathrm{d}x\,\mathrm{d}\xi = \int_{\Omega\times\Omega} A_x^\star p(x,\xi)\varphi(x,\xi)\,\mathrm{d}(x,\xi),\end{aligned}$$

where $A^\star$ is the formal adjoint of A (cf. [41, p. 114])

$$A^\star\psi \coloneqq \sum_{i,j=1}^d \partial_i(a_{i,j}(x)\partial_j\psi) - \sum_{i=1}^d \partial_i b_i(x)\psi + \left(c(x) - \sum_{i=1}^d(\partial_i b_i)\right)\psi.$$

Likewise we compute

$$\begin{aligned}(P\phi, A\psi) = (A^\star P\phi, \psi) &= \int_\Omega A^\star \int_\Omega p(x,\xi)\phi(\xi)\,\mathrm{d}\xi\,\psi(x)\,\mathrm{d}x \\ &= \int_\Omega\int_\Omega A_\xi^\star p(x,\xi)\phi(x)\psi(\xi)\,\mathrm{d}x\,\mathrm{d}\xi = \int_{\Omega\times\Omega} A_\xi^\star p(x,\xi)\varphi(x,\xi)\,\mathrm{d}x\,\mathrm{d}\xi,\end{aligned}$$

where we used the relation $p(x,\xi) = p(\xi,x)$ which comes from P being self-adjoint. We deduce

$$(A\phi, P\psi) + (P\phi, A\psi) = \int_{\Omega\times\Omega} (A_x^\star + A_\xi^\star)p(x,\xi)\varphi(x,\xi)\,\mathrm{d}(x,\xi).$$

We proceed with the non-linear term. In the first step we need to compute the operator $B^\star$. To this end, we can use the Green's formula and obtain

$$B^\star = -D^\star A^\star = -\frac{\partial}{\partial\nu_{A^\star}} = \sum_{i,j=1}^d a_{i,j}(x)\frac{\partial}{\partial_j}\nu_i,$$

where $\nu = (\nu_1,\ldots,\nu_d)$ is the outward normal to Γ (see [37, p. 183]). To simplify the notation, we will write $\frac{\partial}{\partial\nu}$ for $\frac{\partial}{\partial\nu_{A^\star}}$.

We can now plug in $B^\star$ into the non-linear term of (3.2)

$$\begin{aligned}(B^\star P\phi, B^\star P\psi)_{\mathcal{U}} &= \int_\Gamma \frac{\partial}{\partial \nu_\zeta} \int_\Omega p(\zeta, x)\phi(x)\,\mathrm{d}x \cdot \frac{\partial}{\partial \nu_\zeta} \int_\Omega p(\zeta,\xi)\psi(\xi)\,\mathrm{d}\xi\,\mathrm{d}\Gamma_\zeta \\ &= \int_\Gamma \frac{\partial}{\partial \nu_\zeta} \int_\Omega p(x,\zeta)\phi(x)\,\mathrm{d}x \cdot \frac{\partial}{\partial \nu_\zeta} \int_\Omega p(\zeta,\xi)\psi(\xi)\,\mathrm{d}\xi\,\mathrm{d}\Gamma_\zeta.\end{aligned}$$

By applying Fubini's theorem, we conclude

$$\begin{aligned}(B^\star P\phi, B^\star P\psi)_{\mathcal{U}} &= \int_\Gamma \int_\Omega \frac{\partial p}{\partial \nu_\zeta}(x,\zeta)\phi(x)\,\mathrm{d}x \int_\Omega \frac{\partial p}{\partial \nu_\zeta}(\zeta,\xi)\psi(\xi)\,\mathrm{d}\xi\,\mathrm{d}\Gamma_\zeta \\ &= \int_{\Omega\times\Omega} \int_\Gamma \frac{\partial p}{\partial \nu_\zeta}(x,\zeta)\frac{\partial p}{\partial \nu_\zeta}(\zeta,\xi)\,\mathrm{d}\Gamma_\zeta\, \varphi(x,\xi)\,\mathrm{d}(x,\xi).\end{aligned}$$

Note that the boundary integral is well-defined if we assume that it holds $\partial p/\partial \nu_x \in L^2(\Gamma\times\Omega)$ and likewise $\partial p/\partial \nu_\xi \in L^2(\Omega\times\Gamma)$.

In order to complete the derivation in terms of kernel functions, we assume in accordance with [10] and [29, Chapter 3] the operator $C : \mathcal{H} \to \mathcal{Y}$ to be of the form

$$C\phi = \int_\Omega s(x)\phi(x)\,\mathrm{d}x$$

with $s \in L^2(\Omega)$. By this means, $C^\star C$ takes the form

$$\begin{aligned}(C^\star C\phi, \psi) = (C\phi, C\psi)_{\mathbb{R}} &= \int_\Omega s(x)\phi(x)\,\mathrm{d}x \int_\Omega s(\xi)\psi(\xi)\,\mathrm{d}\xi \\ &= \int_{\Omega\times\Omega} s(x)s(\xi)\phi(x)\psi(\xi)\,\mathrm{d}(x,\xi).\end{aligned}$$

We thus set

$$Q = C^\star C : \mathcal{H}\to\mathcal{H}, \quad v \mapsto Qv = \int_\Omega s(x)s(\xi)v(\xi)\,\mathrm{d}\xi = \int_\Omega q(x,\xi)v(\xi)\,\mathrm{d}\xi, \tag{3.3}$$

where $q(x,\xi) = s(x)s(\xi)$ is the kernel of Q.

Therefore, since $C_0^\infty(\Omega\times\Omega)$ is dense in $H_0^1(\Omega\times\Omega)$, the kernel p solves the following variational problem

$$\begin{aligned}&\int_{\Omega\times\Omega} (A_x^\star + A_\xi^\star)p(x,\xi)\varphi(x,\xi)\,\mathrm{d}(x,\xi) + \int_{\Omega\times\Omega}\int_\Gamma \frac{\partial p}{\partial \nu_\zeta}(\zeta,x)\frac{\partial p}{\partial \nu_\zeta}(\xi,\zeta)\,\mathrm{d}\Gamma_\zeta \varphi(x,\xi)\,\mathrm{d}(x,\xi) \\ &\qquad = \int_{\Omega\times\Omega} q(x,\xi)\varphi(x,\xi)\,\mathrm{d}(x,\xi) \text{ for all } \varphi \in H_0^1(\Omega\times\Omega).\end{aligned} \tag{3.4}$$

3.2. Boundary conditions. To derive the boundary conditions for $p(x,\xi)$, we use the following regularity result from [6, p. 520]:

$$P \in \mathcal{L}\Big(\mathcal{H}, \mathcal{D}\big((-A)^{1-\alpha}\big)\Big), \tag{3.5}$$

where for A as in (2.3) we can choose $\alpha \in (0, 1/4)$. Furthermore, it holds

$$\mathcal{D}\big((-A)^{1-\alpha}\big) = \begin{cases} H^{2(1-\alpha)}(\Omega), & \text{if } \alpha \in (3/4, 1), \\ \big\{w \in H^{2(1-\alpha)}(\Omega) : w = 0 \text{ on } \partial\Omega\big\}, & \text{if } \alpha \in (0, 3/4). \end{cases} \tag{3.6}$$

Therefore, we deduce from (3.5) and (3.6) that

$$\text{for all } v \in \mathcal{H} : Pv \in \left\{ w \in H^{2(1-\alpha)}(\Omega) : w = 0 \text{ on } \partial\Omega \right\}, \text{ where } \alpha \in (0, 1/4). \tag{3.7}$$

We next assume that there exists a part $\widetilde{\Gamma} \times \widetilde{\Omega} \subset \partial(\Omega \times \Omega)$ of the boundary such that $p(x, \xi) \neq 0$ for almost all $(x, \xi) \in \widetilde{\Gamma} \times \widetilde{\Omega}$. Then, taking some $v \in \mathcal{H}$ with $\widetilde{\Omega} \subset \operatorname{supp} v$, we have

$$w(x) = \int_\Omega p(x, \xi) v(\xi) \, \mathrm{d}\xi \neq 0$$

for almost all $x \in \widetilde{\Gamma}$, which is a contradiction to (3.7). Hence, with the symmetry of $p(x, \xi)$, we conclude

$$\begin{cases} p(x, \xi) = 0, & x \in \Gamma, \quad \xi \in \Omega, \\ p(x, \xi) = 0, & x \in \Omega, \quad \xi \in \Gamma, \end{cases}$$

compare also [40, p. 158].

We shall summarize the results found in Section 3.1 and in this section as follows.

Theorem 3.1. *The kernel $p \in V$ for the Riccati operator associated with the Dirichlet boundary control of the equation (2.2), where*

$$V := \left\{ f \in H_0^1(\Omega \times \Omega) : \frac{\partial f}{\partial \nu_x} \in L^2(\Gamma \times \Omega) \text{ and } \frac{\partial f}{\partial \nu_\xi} \in L^2(\Omega \times \Gamma) \right\},$$

is the weak solution of the following integro-differential equation (IDE) of Riccati type:

$$\begin{cases} (A_x^\star + A_\xi^\star) p(x, \xi) + \displaystyle\int_\Gamma \frac{\partial p}{\partial \nu_\zeta}(x, \zeta) \frac{\partial p}{\partial \nu_\zeta}(\zeta, \xi) \, \mathrm{d}\Gamma_\zeta = q(x, \xi) & \text{in } \Omega \times \Omega, \\ p(x, \xi) = 0 & \text{on } \partial(\Omega \times \Omega). \end{cases} \tag{3.8}$$

Remark 3.2. *In [40, Chapter III, Section 5], several results of this type are derived, in particular for distributed control, i.e. $B \in \mathcal{L}(\mathcal{U}, \mathcal{H})$), and Neumann boundary control. In [29, Chapter 3] autor considers the Riccati-IDE for Robin boundary control. The Riccati-PDE for Dirichlet boundary control in the one-dimensional situation can be found in [10]. There, the results are based on a stronger regularity of the kernel $p(x, \xi)$, i.e. $p \in C(\Omega \times \Omega)$ (cf. [32]).*

4. Finite element discretization.

In this section, we derive a discrete version of the Riccati-IDE (3.8) by means of a Galerkin discretization by finite elements. To this end, we consider the *full* tensor product discretization of functions defined on the product domain $\Omega \times \Omega$.

4.1. Tensor product approximation.

Let Z be a Hilbert space with

$$Z \otimes Z \subset V,$$

where $\otimes$ denotes the algebraic tensor product, cf. [25, p. 52]. The completion can be taken with respect to an appropriate norm. Furthermore, suppose we are given a finite dimensional subspace $Z_J \subset Z$. We define the full tensor product ansatz space V_J via

$$V_J := Z_J \otimes Z_J. \tag{4.1}$$

If $\{\phi_j\}_{j=1}^{N_J}$ is a basis of Z_J, i.e. $N_J = \dim Z_J$, then

$$\{\phi_{j_1} \otimes \phi_{j_2}\}_{j_1,j_2=1}^{N_J}$$

is a basis of V_J, and $\dim V_J = N_J^2$.

The finite dimensional subspace Z_J might be given by the span of globally continuous, piecewise linear ansatz functions defined with respect to a triangulation or tetrahedralization of Ω, respectively. Thus, the tensor product space V_J would be spanned by products of those functions, compare Section 7 for details.

Next, we want to discuss the discretization of Riccati-IDE (3.8) with respect to the full tensor product space V_J. We make the following ansatz

$$p(x,\xi) = \sum_{j_1,j_2=1}^{N_J} p_{j_1,j_2}\phi_{j_1}(x)\phi_{j_2}(\xi) \in V_J \tag{4.2}$$

for the discretization of the kernel function in the space V_J and write

$$p_J := [p_{1,1}, p_{1,2}, \dots, p_{N_J,N_J}]^\intercal$$

for the coefficient vector of the Riccati kernel.

4.2. Linear part and right-hand side. First, let us examine the linear part of (3.4), i.e., the evaluation of

$$\int_{\Omega\times\Omega} (A_x^\star + A_\xi^\star)p(x,\xi)\varphi(x,\xi)\,\mathrm{d}(x,\xi) \text{ for all } \varphi(x,\xi) = \phi_{k_1}(x)\phi_{k_2}(\xi). \tag{4.3}$$

Let $a^\star : H_0^1(\Omega) \times H_0^1(\Omega) \to \mathbb{R}$ be the bilinear form associated to $A^\star$ (cf. [16, p. 320]), i.e.

(4.4)

$$a^\star(\phi,\psi) := \int_\Omega \sum_{i,j=1}^d a_{i,j}(x)\partial_j\phi(x)\partial_i\psi(x) + \sum_{i=1}^d b_i(x)\phi(x)\partial_i\psi(x) + c(x)\phi(x)\psi(x)\,\mathrm{d}x.$$

The discretization of

$$\int_{\Omega\times\Omega} A_x^\star p(x,\xi)\varphi(x,\xi)\,\mathrm{d}(x,\xi) \text{ for all } \varphi(x,\xi) = \phi_{k_1}(x)\phi_{k_2}(\xi).$$

with respect to the space Z_J leads to an equivalent formulation $(A_J^\star \otimes E_J)\,p_J$. Here, $A_J^\star \in \mathbb{R}^{N_J\times N_J}$ is the system matrix of $A_x^\star$ (cf. [24, p. 185]) with entries

$$(A_J^\star)_{k,\ell} = a^\star(\phi_k, \phi_\ell).$$

$E_J = \big[e_{k,\ell}\big]_{k,\ell=1}^{N_J} \in \mathbb{R}^{N_J\times N_J}$ is the mass matrix, i.e. we have

$$e_{k,\ell} = \int_\Omega \phi_k(x)\phi_\ell(x)\,\mathrm{d}x. \tag{4.5}$$

Due to the symmetry of (4.3) with respect to x and ξ, we obtain the following discrete representation for the linear part of the Riccati-IDE (3.4):

$$(A_J^\star \otimes E_J + E_J \otimes A_J^\star)\,p_J.$$

Since the right-hand side is a rank-1 function, compare (3.3), it can simply be computed in accordance with

$$Q_J = q_J \otimes q_J, \text{ where } q_J = [q_k]_{k=1}^{N_J} \text{ and } q_k = \int_\Omega s(x)\phi_k(x)\,\mathrm{d}x.$$

4.3. Nonlinear part. The nonlinear part of the Riccati equation in variational form (3.4) reads

$$\int_{\Omega\times\Omega}\int_\Gamma \frac{\partial p}{\partial\nu_\zeta}(x,\zeta)\frac{\partial p}{\partial\nu_\zeta}(\zeta,\xi)\,\mathrm{d}\Gamma_\zeta\,\phi_{k_1}(x)\phi_{k_2}(\xi)\,\mathrm{d}(x,\xi),\quad k_1,k_2=1,\dots,N_J. \tag{4.6}$$

We use the ansatz (4.2) for the kernel $p(x,\xi)$ and consider first the integral over Γ:

$$\begin{aligned}\int_\Gamma \frac{\partial p}{\partial\nu_\zeta}(x,\zeta)\frac{\partial p}{\partial\nu_\zeta}(\zeta,\xi)\,\mathrm{d}\Gamma_\zeta &\\ = \int_\Gamma \left(\frac{\partial}{\partial\nu_\zeta}\sum_{i_1,i_2=1}^{N_J} p_{i_1,i_2}\phi_{i_1}(x)\phi_{i_2}(\xi)\right)\left(\frac{\partial}{\partial\nu_\zeta}\sum_{j_1,j_2=1}^{N_J} p_{j_1,j_2}\phi_{j_1}(x)\phi_{j_2}(\xi)\right)\mathrm{d}\Gamma_\zeta &\\ = \int_\Gamma \frac{\partial}{\partial\nu_\zeta}\sum_{i_2=1}^{N_J}\phi_{i_2}(\zeta)\frac{\partial}{\partial\nu_\zeta}\sum_{j_1=1}^{N_J}\phi_{j_1}(\zeta)\,\mathrm{d}\Gamma_\zeta\sum_{i_1,j_2=1}^{N_J} p_{i_1,i_2}\phi_{i_1}(x)p_{j_1,j_2}\phi_{j_2}(\xi).&\end{aligned} \tag{4.7}$$

To simplify the notation, we assume in the following that the integrals

$$\int_\Gamma \frac{\partial}{\partial\nu_\zeta}\phi_{i_2}(\zeta)\frac{\partial}{\partial\nu_\zeta}\phi_{j_1}(\zeta)\,\mathrm{d}\Gamma_\zeta,\quad i_2,j_1=1,\dots,N_J, \tag{4.8}$$

can be computed exactly. This assumption holds e.g. for piecewise polynomials as basis functions for Z_J, which are often considered in the finite element method. In a more general case, we may take an appropriate quadrature rule for the evaluation of integrals in (4.8) (cf. Section 5.3). Now, for the integration over Γ in the last line of (4.7), we write

$$\sum_{i_2,j_1=1}^{N_J}\int_\Gamma \frac{\partial}{\partial\nu_\zeta}\phi_{i_2}(\zeta)\frac{\partial}{\partial\nu_\zeta}\phi_{j_1}(\zeta)\,\mathrm{d}\Gamma_\zeta =: \sum_{i_2,j_1=1}^{N_J}\alpha_{i_2,j_1}, \tag{4.9}$$

and obtain for the complete expression

$$\int_\Gamma \frac{\partial p}{\partial\nu_\zeta}(x,\zeta)\frac{\partial p}{\partial\nu_\zeta}(\zeta,\xi)\,\mathrm{d}\Gamma_\zeta = \sum_{i_2,j_1=1}^{N_J}\alpha_{i_2,j_1}\sum_{i_1,j_2=1}^{N_J} p_{i_1,i_2}\phi_{i_1}(x)p_{j_1,j_2}\phi_{j_2}(\xi).$$

With this intermediate result the discretization of (4.6) yields

$$\int_{\Omega\times\Omega}\left(\sum_{i_2,j_1=1}^{N_J}\alpha_{i_2,j_1}\sum_{i_1,j_2=1}^{N_J} p_{i_1,i_2}\phi_{i_1}(x)p_{j_1,j_2}\phi_{j_2}(\xi)\right)\phi_{k_1}(x)\phi_{k_2}(\xi)\,\mathrm{d}(x,\xi). \tag{4.10}$$

Equation (4.10) corresponds to the application of the mass matrix $E_J\otimes E_J$ (cf. (4.5)) to the linear combination

$$\sum_{i_2,j_1=1}^{N_J}\alpha_{i_2,j_1}p_{:,i_2}\otimes p_{j_1,:}, \tag{4.11}$$

where we use the notation

$$p_{:,\ell} := [p_{1,\ell},\dots,p_{N_J,\ell}]^\intercal \text{ and } p_{\ell,:} := [p_{\ell,1},\dots,p_{\ell,N_J}]^\intercal \text{ for } \ell=1,\dots,N_J.$$

Overall, we obtain following expression for the evaluation of the nonlinear part of the Riccati equation (3.8) with ansatz (4.2)

$$E \otimes E \left(\sum_{i_2, j_1 = 1}^{N_J} \alpha_{i_2, j_1} p_{:, i_2} \otimes p_{j_1, :} \right). \tag{4.12}$$

Remark 4.1. *A slightly different representation of quadratic part can be obtained in terms of matrices. Let $X, Y, Z \in \mathbb{R}^{n \times n}$. We can write the product XYZ as*

$$XYZ = \sum_{i=1}^{m} \left(\sum_{j=1}^{m} Y_{j,i} X_{:,j} \right) Z_{i,:}.$$

Observe that this expression is similar to (4.11). In view of definition of α_{i_2, j_1} in (4.9), we can introduce the matrices

$$B_J = \left[\alpha_{k,\ell} \right]_{k,\ell=1}^{N_J} \in \mathbb{R}^{N_J \times N_J}, \quad P_J = \left[p_{k,\ell} \right]_{k,\ell=1}^{N_J} \in \mathbb{R}^{N_J \times N_J},$$

and rewrite the sum in (4.9) as matrix product $P_J B_J P_J$.

Furthermore, by using the identity

$$\operatorname{vec}(XYZ) = \left(Z^T \otimes X \right) \operatorname{vec}(Y),$$

the equation (4.12) may be considered as matrix multiplication

$$E_J \left(\sum_{i_2, j_1 = 1}^{N_J} \alpha_{i_2, j_1} p_{:, i_2} p_{j_1, :}^{\intercal} \right) E_J = E_J P_J B_J P_J E_J.$$

This form corresponds to the usual discretization of the quadratic term of the ARE.

We consider the complexity for the evaluation of (4.12).

Theorem 4.2. *The computational cost of evaluating the Riccati-IDE discretized by the finite element method is of the order $\mathcal{O}(N_J^2 N_J^{\frac{d-1}{d}})$.*

Proof. The computational cost are dominated by the evaluation of the quadratic term. Due to the locality of the finite element basis and of the normal derivative, only $\mathcal{O}(N_J^{\frac{d-1}{d}})$ coefficients α_{i_2, j_1} do not vanish in the sum (4.12). For each summand we need to compute the tensor product $p_{:, i_2} \otimes p_{j_1, :}$, which takes $\mathcal{O}(N_J^2)$ operations. The overall cost amount to $\mathcal{O}(N_J^2 N_J^{\frac{d-1}{d}})$. □

4.4. Newton's method. The Riccati-IDE is a non-linear equation with quadratic non-linearity. To find a solution we have to apply some iterative scheme. To this end, we use Newton's method as suggested in e.g. [33].

We first introduce the following notation to simplify the presentation. The linear part of the Riccati-IDE (3.8) is given by the operator $(A_x^\star + A_\xi^\star)$ on $\Omega \times \Omega$. We set

$$\mathcal{R}_L : p \mapsto \left[\varphi \mapsto a_x^\star(p(x,\xi), \varphi(x,\xi)) + a_\xi^\star(p(x,\xi), \varphi(x,\xi)) \right],$$

where $a_x^\star$, $a_\xi^\star$ are the applications of bilinear form $a^\star$ from (4.4) in x, respectively ξ. The quadratic part is

$$\mathcal{R}_{NL} : p \mapsto \left[\varphi \mapsto \int_{\Omega \times \Omega} \int_{\Gamma} \frac{\partial p}{\partial \nu_\zeta}(x, \zeta) \frac{\partial p}{\partial \nu_\zeta}(\zeta, \xi) \, \mathrm{d}\Gamma_\zeta \, \varphi(x,\xi) \, \mathrm{d}(x,\xi) \right].$$

Finally, the right-hand side can be written as

$$\mathcal{Q} : q \mapsto \left[\varphi \mapsto \int_{\Omega\times\Omega} q(x,\xi)\varphi(x,\xi)\,\mathrm{d}(x,\xi)\right].$$

With these operators at hand, we can write the Riccati-IDE (3.8) as

$$\mathcal{R}_L(p) - \mathcal{R}_{NL}(p) + \mathcal{Q} = 0.$$

Applying the Newton's method to this equation results in

$$D(\mathcal{R}_L - \mathcal{R}_{NL})[p^{(i)}]\big(p^{(i+1)} - p^{(i)}\big) = -\big(\mathcal{R}_L(p^{(i)}) - \mathcal{R}_{NL}(p^{(i)}) + \mathcal{Q}\big),$$

where D denotes the Fréchet derivative and i the iteration index of the Newton's method.

The Fréchet derivative of a linear operator is the operator itself, i.e. we obtain

$$D\mathcal{R}_L[g](h) = \mathcal{R}_L(h),$$

while the Fréchet derivative of the non-linear part is given by

$$D\mathcal{R}_{NL}[g](h) = \\ \left[\varphi \mapsto \int_{\Omega\times\Omega}\int_\Gamma \frac{\partial g}{\partial\nu_\zeta}(x,\zeta)\frac{\partial h}{\partial\nu_\zeta}(\zeta,\xi) + \frac{\partial h}{\partial\nu_\zeta}(x,\zeta)\frac{\partial g}{\partial\nu_\zeta}(\zeta,\xi)\,\mathrm{d}\Gamma_\zeta\,\varphi(x,\xi)\,\mathrm{d}(x,\xi)\right].$$

Therefore, for Newton's method in the i-th iteration, we seek the new iterate $p^{(i+1)}$ such that

$$\big(\mathcal{R}_L - D\mathcal{R}_{NL}[p^{(i)}]\big)(p^{(i+1)}) = -\big(\mathcal{R}_{NL}(p^{(i)}) + \mathcal{Q}\big), \quad i = 1,2,\ldots. \tag{4.13}$$

The discrete version of Newton's method (4.13) is the Sylvester type equation of the form

$$(E_J P_J^{(i)} B_J - A_J^\star)P_J^{(i+1)}E_J + E_J P_J^{(i+1)}(B_J P_J^{(i)} E_J - A_J^\star) = E_J P_J^{(i)} B_J P_J^{(i)} E_J + Q_J.$$

Notice that, in accordance with Theorem 4.2, each iteration of Newton's method can be realized within $\mathcal{O}(N_J^2 N_J^{\frac{d-1}{d}})$ cost if an optimal preconditioner like the multigrid method is used. Therefore, the over-all cost are $\mathcal{O}(N_{\text{iter}} N_J^2 N_J^{\frac{d-1}{d}})$, where N_{iter} denotes the number of iterations used by Newton's method.

5. Sparse grid discretization.

Sparse grids are a numerical discretization approach, which is especially of interest for high dimensional problems. Discretization on a full grid, as described in Section 4, suffers from the curse of dimensionality, i.e. the number of degrees of freedom in the space V_J is N_J^2. However, provided certain regularity, the Riccati kernel can be approximated by sparse grids at essentially same rate with only $N_J \log N_J$ degrees of freedom. In this section, we intend to discretize and evaluate the Riccati-IDE (4.3) in a sparse grid space. A detailed presentation and introduction to sparse grids can be found in [1, 9, 17, 19, 49], see also [8], [24, p. 260], [25, p. 280], and [26, 27, 28]. This section recalls the main ideas, where the representation follows [53].

5.1. Discretization by sparse grids. In the following we denote by small bold letters, e.g. $\boldsymbol{i} \in \mathbb{N}^2$, a 2-dimensional multi-index, $\boldsymbol{i} = (i_1, i_2)$. In contrast, cursive letters, e.g. $i \in \mathbb{N}$, are used as usual indices.

As in Section 4, we consider Hilbert spaces Z and V with $Z \otimes Z \subset V$. Suppose we are given a nested sequence of finite dimensional subspaces Z_j, $j = 0, \dots, J$, of Z, that is

$$Z_0 \subset Z_1 \subset Z_2 \subset \cdots \subset Z_J \subset Z.$$

We are going to construct a finite dimensional subspace of V, which will be our ansatz and test space, upon the spaces Z_j. In accordance with [9, 19, 27], let us introduce hierarchical difference spaces W_j via

$$W_j \coloneqq Z_j \ominus Z_{j-1},\ Z_{-1} \coloneqq \{0\},\ N_j \coloneqq \dim W_j.$$

We refer to j as level. Furthermore, we shall assume that N_j behaves like an increasing geometric sequence. This is, for example, the case if the sequence $\{Z_m\}$ is constructed from dyadic subdivisions of a given coarse grid triangulation or tetrahedralization of the underlying domain. In particular, for a dyadic subdivision we obtain $|N_j| = \mathcal{O}(2^{dj})$.

Let $W_{\boldsymbol{j}} = W_{(j_1,j_2)}$ denote the tensor product of two spaces W_{j_1} and W_{j_2}

$$W_{\boldsymbol{j}} \coloneqq W_{j_1} \otimes W_{j_2} = (Z_{j_1} \ominus Z_{j_1-1}) \otimes (Z_{j_2} \ominus Z_{j_2-1}).$$

The dimension of $W_{\boldsymbol{j}}$ is $N_{\boldsymbol{j}} \coloneqq \dim W_{\boldsymbol{j}} = N_{j_1} N_{j_2}$. With these spaces at hand, we can write the full tensor product space V_J from (4.1) as a direct sum

$$V_J = \bigoplus_{0 \le j_1, j_2 \le J} W_{j_1,j_2} = \bigoplus_{0 \le \|\boldsymbol{j}\|_\infty \le J} W_{\boldsymbol{j}}.$$

In contrast to the full tensor product, the idea of a *sparse grid* is to consider only those basis functions in the space V_J, which have a large contribution to the representation of an interpolated function $f \in V$, cf. [9, 19]. We denote the sparse grid function space with $\widehat{V}_J$ and give the following formal definition

$$\widehat{V}_J \coloneqq \bigoplus_{0 \le j_1 + j_2 \le J} W_{j_1,j_2} = \bigoplus_{0 \le \|\boldsymbol{j}\|_1 \le J} W_{\boldsymbol{j}}. \tag{5.1}$$

From the representation (5.1), we infer that $\widehat{V}_J$ consists only of hierarchical difference spaces with $j_1 + j_2 \le J$. This construction leads to the relation

$$\widehat{N}_J \coloneqq \dim \widehat{V}_J = \mathcal{O}(N_J \log N_J).$$

In general, for sparse grids on m-fold tensor product spaces, there holds $\dim \widehat{V}_J = \mathcal{O}(N_J \log N_J^{m-1})$ while essentially no approximation power is lost provided that the function to be approximated exhibits extra smoothness in terms of bounded mixed derivatives. In other words, the exponential dependency is only in the $\log N_J$ factor, which substantially reduces the dimension of the sparse grid space compared to the full grid.

We proceed analogously to Section 4 and discretize the Riccati kernel in the sparse grid space $\widehat{V}_J$. Let M, with $|M| = N_J$, be an index set. We assume the space Z_J to be spanned by some hierarchical basis $\{\phi_m\}_{m \in M}$, i.e. the spaces Z_j are spanned by subsets of $\{\phi_m\}_{m \in M}$.

To reflect the nested structure of the spaces Z_j, let us introduce a function $\delta : \mathbb{N}^2 \to M$, such that $\delta(j,\cdot) : \mathbb{N} \to M$ is an enumeration of the functions ϕ_m spanning the space W_j. We define the following notation

$$\phi_{\boldsymbol{j}} \in W_k :\Leftrightarrow \phi_{\delta(\boldsymbol{j})} \in W_k, \quad \boldsymbol{j} \in W_k :\Leftrightarrow \phi_{\boldsymbol{j}} \in W_k, \quad \boldsymbol{j} \in Z_J :\Leftrightarrow \exists k \leq J : \boldsymbol{j} \in W_k.$$

The sparse grid ansatz $\widehat{p}$ for the Riccati kernel reads

$$\widehat{p}(x,\xi) = \sum_{k+l\leq J} \sum_{\boldsymbol{i}\in W_k} \sum_{\boldsymbol{j}\in W_l} p_{\boldsymbol{i},\boldsymbol{j}} \varphi_{\boldsymbol{i},\boldsymbol{j}}(x,\xi) \in \widehat{V}_J, \tag{5.2}$$

where we abbreviated $\varphi_{\boldsymbol{i},\boldsymbol{j}} \coloneqq \phi_{\boldsymbol{i}} \otimes \phi_{\boldsymbol{j}}$. The vector $\widehat{p}_J \in \mathbb{R}^{\widehat{N}_J}$ of coefficients takes the form

$$\widehat{p}_J \coloneqq [p_{\boldsymbol{j}}]_{\|\boldsymbol{j}\|_1 \leq J}, \quad p_{\boldsymbol{j}} \coloneqq [p_{\boldsymbol{k},\boldsymbol{l}}]_{\boldsymbol{k}\in W_{j_1}, \boldsymbol{l}\in W_{j_2}},$$

i.e., $p_{\boldsymbol{j}} \in \mathbb{R}^{N_{\boldsymbol{j}}}$ are the coefficient vectors corresponding to the spaces $W_{\boldsymbol{j}}$.

Note that for $n \in \mathbb{N}$ and $x \in \mathbb{R}^{N_J}$ we use an analogous notation

$$x_n = \left[x_{(n,m)}\right]_{(n,m)\in W_n}.$$

5.2. Linear part. First, we investigate the linear part of the Riccati-IDE (3.8). It is possible to consider the application of a linear operator L on V to a function from the sparse grid space $\widehat{V}_J$ in a rather abstract setting. The only additional assumption, beside (5.1), is of the operator L to reflect the tensor product structure of the ansatz space $\widehat{V}_J$. This means that L, restricted to the algebraic tensor space $Z \otimes Z$, is a tensor product of operators T_1 and T_2 acting on Z, i.e. $L|_{Z\otimes Z} = T_1 \otimes T_2$. The resulting discretization matrix of L has block tensor structure, which, in turn, can be utlized for fast matrix–vector multiplication (compare [53]).

Provided the operators T_1, T_2 can be evaluated with linear complexity $\mathcal{O}(\dim Z_m)$ on the spaces Z_m, the product operator L can be applied to an element of $\widehat{V}_J$ with $\mathcal{O}(N_J \log N_J)$ operations. The algorithm for the evaluation of the matrix-vector product in the space $\widehat{V}_J$ is called UniDir, cf. [9, 53]. Algorithms which employ similar techniques have been developed in [26, 27, 28].

The linear part of the Riccati-IDE (3.8) is the operator $\mathcal{R}_L = A_x^\star \otimes \mathrm{Id}_\xi + \mathrm{Id}_x \otimes A_\xi^\star$ on $\Omega \times \Omega$. This operator has tensor product structure for ansatz spaces stemming from a discretization separate with respect to x and ξ, and in particular for $\widehat{V}_J$.

To be more precise we set $Z = H_0^1(\Omega)$ and consider the operators

$$A_x^\star, A_\xi^\star : H_0^1(\Omega) \to H^{-1}(\Omega), \quad v \mapsto [w \mapsto a^\star(v,w)],$$

and

$$\mathrm{Id} : H_0^1(\Omega) \to H^{-1}(\Omega), \quad v \mapsto \left[w \mapsto \int_\Omega vw \,\mathrm{d}x\right]. \tag{5.3}$$

With these operators we can write for the linear part of the Riccati-IDE

$$\mathcal{R}_L|_{H_0^1(\Omega)\otimes H_0^1(\Omega)} = A_x^\star \otimes \mathrm{Id} + \mathrm{Id} \otimes A_\xi^\star : H_0^1(\Omega) \otimes H_0^1(\Omega) \to H^{-1}(\Omega \times \Omega),$$

which is a sum of tensor product operators, as required by the UniDir algorithm.

Besides the tensor product structure, we have to ensure that $A_x^\star$, $A_\xi^\star$, and Id can be evaluated with linear complexity on the discretization spaces Z_m. There are different examples of appropriate sequences $\{Z_m\}$ and corresponding bases, like e.g. hierachical bases ([9]), wavelets ([14]), multilevel frames ([27, 53]), or polynomials of different degrees ([1, 9]). In this article, we will take Z_m spanned by the hierarchical basis of standard hat functions (cf. [9, 27]). We provide an exact definition in Section 7.

Let $m = 0, \dots, J$. We denote by $I^{W_m} : \mathbb{R}^{N_m} \to \mathbb{R}^{N_J}$ the canonical embedding, which maps the coefficient vector $x \in \mathbb{R}^{N_m}$ of a function $\widetilde{x} \in W_m$ with respect to the basis $\delta(m)$ to the coefficient vector $y \in \mathbb{R}^{N_J}$ of a function $\widetilde{x} \in Z_J$. The map $I^{Z_m} : \mathbb{R}^{\dim Z_m} \to \mathbb{R}^{N_J}$ is defined analogously.

Let $I_{W_m} : \mathbb{R}^{N_J} \to \mathbb{R}^{N_m}$ be the operator which projects the coefficients of $\widetilde{x} \in Z_J$ to the coefficients corresponding to the space $W_m \subset Z_J$. Again, $I_{Z_m} : \mathbb{R}^{N_J} \to \mathbb{R}^{\dim Z_m}$ is the analogous map for the space Z_m.

Using the operators I^{W_m} and I_{W_m}, we define for $X \in \mathbb{R}^{N_J \times N_J}$ the matrix $X_{m,n} \coloneqq I_{W_m} X I^{W_n}$ and introduce the following bilinear map

$$\widehat{\otimes} : \mathbb{R}^{N_J \times N_J} \times \mathbb{R}^{N_J \times N_J} \to \mathbb{R}^{\widehat{N}_J \times \widehat{N}_J}, \ (X, Y) \mapsto [X_{i_1,j_1} \otimes Y_{i_2,j_2}]_{\|\boldsymbol{i}\|_1, \|\boldsymbol{j}\|_1 \le J} .$$

With this definition at hand, we can write the application of $\mathcal{R}_L$ to a function $\widehat{p} \in \widehat{V}_J$ as the following matrix-vector product

$$\left(A_J^\star \,\widehat{\otimes}\, E_J + E_J \,\widehat{\otimes}\, A_J^\star\right) \widehat{p}_J.$$

Albeit the matrix $A_J^\star \,\widehat{\otimes}\, E_J + E_J \,\widehat{\otimes}\, A_J^\star$ is not sparse (cf. [27, 53]), the matrix–vector product can be computed with complexity $\mathcal{O}\left(N_J \log N_J\right)$, i.e. linear in the number of degrees of freedom of the sparse grid.

Note that although we can use the UniDir algorithm in the presented abstract setting, we need further assumptions on the ansatz and test spaces $\widehat{V}_J$ to guarantee the convergence of Galerkin method, see e.g. [24, Chapter 8].

5.3. Nonlinear part. In general, for the evaluation of the non-linear part (4.6) of the Riccati-IDE, we have to consider the evaluation of the operator

$$\mathcal{R}_{NL} : p \mapsto \left[\varphi \mapsto \int_{\Omega \times \Omega} \int_\Gamma \frac{\partial p}{\partial \nu_\zeta}(x, \zeta) \frac{\partial p}{\partial \nu_\zeta}(\zeta, \xi) \,\mathrm{d}\Gamma_\zeta \,\varphi(x, \xi) \,\mathrm{d}(x, \xi)\right].$$

We started with discretization of the boundary integral in Section 4.3 and obtained the expression (4.11). We proceed along same lines for the discretization with respect to $\widehat{V}_J$. In the fist step, we apply a quadrature rule with nodes ζ_s and weights w_s, $s \in S$, to calculate the boundary integral, i.e. we approximate the operator

$$p(x, \xi) \mapsto \int_\Gamma \frac{\partial p}{\partial \nu_\zeta}(x, \zeta) \frac{\partial p}{\partial \nu_\zeta}(\zeta, \xi) \,\mathrm{d}\Gamma_\zeta$$

by

$$p(x, \xi) \mapsto \sum_{s=1}^{S} w_s \frac{\partial p}{\partial \nu_\zeta}(x, \zeta_s) \frac{\partial p}{\partial \nu_\zeta}(\zeta_s, \xi). \tag{5.4}$$

Now, similar to (4.9) we can write

$$\begin{aligned} &\int_\Gamma \frac{\partial}{\partial \nu_\zeta} \sum_{\boldsymbol{j} \in Z_J} \phi_{\boldsymbol{j}}(\zeta) \, \frac{\partial}{\partial \nu_\zeta} \sum_{\boldsymbol{k} \in Z_J} \phi_{\boldsymbol{k}}(\zeta) \,\mathrm{d}\Gamma_\zeta \\ &\approx \sum_{s \in S} w_s \frac{\partial}{\partial \nu_\zeta} \sum_{\boldsymbol{j} \in Z_J} \phi_{\boldsymbol{j}}(\zeta_s) \frac{\partial}{\partial \nu_\zeta} \sum_{\boldsymbol{k} \in Z_J} \phi_{\boldsymbol{k}}(\zeta_s) \eqqcolon \sum_{s \in S} \sum_{\boldsymbol{j}, \boldsymbol{k} \in Z_J} \alpha^s_{\boldsymbol{j},\boldsymbol{k}}. \end{aligned}$$

The complete expression (5.4) is of the form

$$\sum_{s\in S}\sum_{\boldsymbol{j},\boldsymbol{k}\in Z_J}\alpha^s_{\boldsymbol{j},\boldsymbol{k}}\sum_{\substack{\boldsymbol{i}\in Z_{J-j_1}\\ \boldsymbol{\ell}\in Z_{J-k_1}}} p_{\boldsymbol{i},\boldsymbol{j}}\phi_{\boldsymbol{i}}(x)\ p_{\boldsymbol{k},\boldsymbol{\ell}}\phi_{\boldsymbol{\ell}}(\xi).$$

Next, let us consider the integration with respect to x and ξ in $\mathcal{R}_{NL}$. We can equivalently write the computation of scalar products with test functions as a matrix multiplication (see also (4.10), (4.11)). To achieve this, we introduce the notation

$$\widehat{p}_{:,\boldsymbol{\ell}} := [p_{\boldsymbol{k},\boldsymbol{\ell}}]_{\boldsymbol{k}\in Z_{J-\ell_1}}\ ,\ \widehat{p}_{\boldsymbol{\ell},:} := [p_{\boldsymbol{\ell},\boldsymbol{k}}]_{\boldsymbol{k}\in Z_{J-\ell_1}}\ ,\ \boldsymbol{\ell}\in Z_J,$$

i.e. $\widehat{p}_{:,\ell},\ \widehat{p}_{\ell,:}\in\mathbb{R}^{\dim Z_{J-\ell_1}}$ are slices of $\widehat{p}\in\mathbb{R}^{\widehat{N}_J}$ which correspond to a particular ansatz function with level ℓ_1 and index ℓ_2. Furthermore, let $x,y\in\mathbb{R}^{N_J}$, $X,Y\in\mathbb{R}^{N_J\times N_J}$ and

$$\begin{aligned}
&\widehat{\otimes}:\mathbb{R}^{N_J}\times\mathbb{R}^{N_J}\to\mathbb{R}^{\widehat{N}_J},\ (x,y)\mapsto\Big[(I_{W_{j_1}}x)\otimes(I_{W_{j_2}}y)\Big]_{\|\boldsymbol{j}\|_1\le J}\,,\\
&\widetilde{\otimes}:\mathbb{R}^{N_J\times N_J}\times\mathbb{R}^{N_J\times N_J}\to\mathbb{R}^{\widehat{N}_J\times N_J^2},\ (X,Y)\mapsto[X_{i_1,j_1}\otimes Y_{i_2,j_2}]_{\|\boldsymbol{i}\|_1,\|\boldsymbol{j}\|_\infty\le J}\,.
\end{aligned}$$

The discretization of the L^2 scalar product in $\mathcal{R}_{NL}$ now yields (cf. (4.12))

$$E\,\widetilde{\otimes}\,E\left(\sum_{s\in S}\sum_{\boldsymbol{j},\boldsymbol{k}\in Z_J}\alpha^s_{\boldsymbol{j},\boldsymbol{k}}\ (I^{Z_{J-j_1}}\widehat{p}_{:,\boldsymbol{j}})\otimes(I^{Z_{J-k_1}}\widehat{p}_{\boldsymbol{k},:})\right)\,. \tag{5.5}$$

Next, we are going to investigate the complexity for the evaluation of the nonlinear part of the Riccati-IDE discretized by sparse grids. The expression in (5.5) can obviously be computed with $\mathcal{O}(N_J^2N_J^{\frac{d-1}{d}})$ operations. In the following, we will reduce this complexity stepwise. We start with an auxiliary lemma.

Lemma 5.1. *For $x,y\in\mathbb{R}^{N_J}$ and $T\in\mathbb{R}^{N_J\times N_J}$ it holds*

$$\big(T\,\widetilde{\otimes}\,T\big)\,x\otimes y=(Tx)\,\widehat{\otimes}\,(Ty)\,.$$

Proof. Straightforward calculation yields

$$\begin{aligned}
(T\,\widetilde{\otimes}\,T)(x\otimes y)&=\left[\sum_{j_1,j_2\le J}(T_{i_1,j_1}\otimes T_{i_2,j_2})\,x_{j_1}\otimes y_{j_2}\right]_{\|\boldsymbol{i}\|_1\le J}\\
&=\left[\left(\sum_{j_1\le J}T_{i_1,j_1}x_{j_1}\right)\otimes\left(\sum_{j_2\le J}T_{i_2,j_2}y_{j_2}\right)\right]_{\|\boldsymbol{i}\|_1\le J}\\
&=\big[\big(I_{W_{i_1}}Tx\big)\otimes\big(I_{W_{i_2}}Ty\big)\big]_{\|\boldsymbol{i}\|_1\le J}=(Tx)\,\widehat{\otimes}\,(Ty)\,.
\end{aligned}$$

□

The main result is the following theorem.

Theorem 5.2. *The computational cost of evaluating the Riccati-IDE discretized by the sparse grid method is of the order $\mathcal{O}(N_JN_J^{\frac{d-1}{d}}\log N_J)$ while the memory requirement is of the order $\mathcal{O}(N_J\log N_J)$.*

Proof. Given a quadrature point ζ_s we note that

$$\alpha^s_{\boldsymbol{j},\boldsymbol{k}} = \frac{\partial}{\partial \nu_\zeta}\phi_{\boldsymbol{j}}(\zeta_s)\frac{\partial}{\partial \nu_\zeta}\phi_{\boldsymbol{k}}(\zeta_s), \quad \boldsymbol{j},\boldsymbol{k} \in Z_J,$$

i.e. $\alpha^s_{\boldsymbol{j},\boldsymbol{k}}$ is a tensor product. We write $\alpha^s_{\boldsymbol{j},\boldsymbol{k}} = \alpha^s_{\boldsymbol{j}}\alpha^s_{\boldsymbol{k}}$. Therefore, each summand in (5.5) is a tensor product of the form

$$\alpha^s_{\boldsymbol{j},\boldsymbol{k}}\ (I^{Z_{J-j_1}}\widehat{p}_{:,\boldsymbol{j}}) \otimes (I^{Z_{J-k_1}}\widehat{p}_{\boldsymbol{k},:}) = (I^{Z_{J-j_1}}\alpha^s_{\boldsymbol{j}}\widehat{p}_{:,\boldsymbol{j}}) \otimes (I^{Z_{J-k_1}}\alpha^s_{\boldsymbol{k}}\widehat{p}_{\boldsymbol{k},:}),$$

and we can take separately the summation over $\boldsymbol{j}$ and $\boldsymbol{k}$ in (5.5)

$$\sum_{\boldsymbol{j},\boldsymbol{k}\in Z_J} \alpha^s_{\boldsymbol{j},\boldsymbol{k}}\ (I^{Z_{J-j_1}}\widehat{p}_{:,\boldsymbol{j}}) \otimes (I^{Z_{J-k_1}}\widehat{p}_{\boldsymbol{k},:}) = \sum_{\boldsymbol{j}\in Z_J} I^{Z_{J-j_1}}\alpha^s_{\boldsymbol{j}}\widehat{p}_{:,\boldsymbol{j}} \otimes \sum_{\boldsymbol{k}\in Z_J} I^{Z_{J-k_1}}\alpha^s_{\boldsymbol{k}}\widehat{p}_{\boldsymbol{k},:}.$$

For each quadrature point ζ_s the sum

$$\sum_{\boldsymbol{j}\in Z_J} I^{Z_{J-j_1}}\alpha^s_{\boldsymbol{j}}\widehat{p}_{:,\boldsymbol{j}} \tag{5.6}$$

has at most $\mathcal{O}(\log N_J) = J$ nonzero coefficients $\alpha^s_{\boldsymbol{j}}$ due to the hierarchical sorting of the basis. Each summand can be added with complexity $\mathcal{O}(\dim Z_{J-j_1}) = \sum_{n=0}^{J-j_1} N_n$. We estimate the complexity for the evaluation of (5.6) by

$$\sum_{j_1=0}^{J}\sum_{n=0}^{J-j_1} N_n = \mathcal{O}\left(\sum_{j_1=0}^{J} 2^{d(J-j_1+1)}\right) = \mathcal{O}\left(N_J\right).$$

By using the above result and Lemma 5.1, we rewrite (5.5) as

$$\begin{aligned}\sum_{s\in S} E \mathbin{\widetilde{\otimes}} E &\left(\sum_{\boldsymbol{j}\in Z_J} I^{Z_{J-j_1}}\alpha^s_{\boldsymbol{j}}\widehat{p}_{:,\boldsymbol{j}} \otimes \sum_{\boldsymbol{k}\in Z_J} I^{Z_{J-k_1}}\alpha^s_{\boldsymbol{k}}\widehat{p}_{\boldsymbol{k},:}\right)\\ = \sum_{s\in S} &\left(E_J\sum_{\boldsymbol{j}\in Z_J} I^{Z_{J-j_1}}\alpha^s_{\boldsymbol{j}}\widehat{p}_{:,\boldsymbol{j}}\right) \mathbin{\widehat{\otimes}} \left(E_J\sum_{\boldsymbol{k}\in Z_J} I^{Z_{J-k_1}}\alpha^s_{\boldsymbol{k}}\widehat{p}_{\boldsymbol{k},:}\right).\end{aligned} \tag{5.7}$$

The evaluation of the map $\widehat{\otimes}$ for each summand is of complexity $\mathcal{O}(N_J \log N_J)$, i.e. the overall complexity for the evaluation of (5.7) is $\mathcal{O}(|S| N_J \log N_J)$. In view of $|S| \sim N_J^{\frac{d-1}{d}}$, since we integrate only over the boundary Γ of Ω, we end up with the computational complexity $\mathcal{O}(N_J N_J^{\frac{d-1}{d}} \log N_J)$ for the nonlinear part.

As described at the beginning of this section, the complexity for the evaluation of the linear part is $\mathcal{O}(N_J \log N_J)$. The overall complexity is thus of the order $\mathcal{O}(N_J N_J^{\frac{d-1}{d}} \log N_J)$.

The requirement for the memory of the order $\mathcal{O}(N_J \log N_J)$ is obvious. ∎

Theorem 5.2 means that we save essentially one order in N_J in both, memory requirement and computation time, compared to the traditional finite element discretization from the previous section.

Remark 5.3. 1. *The realization of Newton's method based on the above algorithms is straightforward, compare Section 4.4. Especially, the over-all cost for computing the optimal kernel function* $\widehat{p}$ *are* $\mathcal{O}(N_{iter} N_J N_J^{\frac{d-1}{d}} \log N)$, *where* N_{iter} *denotes the number of iterations used by Newton's method.*

2. The bottle-neck of the presented sparse grid discretization is the evaluation of the non-linear term $\mathcal{R}_{NL}$, which does not scale linearly. A much more involved algorithm is able to evaluate $\mathcal{R}_{NL}$ in complexity $\mathcal{O}\big(N_J N_J^{\frac{d-1}{2d}}\big)$. This is still not of linear complexity but is essentially the square root of the cost the finite element method has.

3. The discretization of the Riccati-IDE has been performed in an exact way, meaning that we compute the exact Galerkin system. Instead, one could also evaluate $\mathcal{R}_{NL}$ in an approximate way, reducing the over-all complexity further. This would introduce a consistency error which, however, would not matter if it is of the same order as the discretization error.

4. Using similar argumentation as in the proof of Theorem 5.2, we obtain the expression

$$\sum_{s\in S}\left(E_J\sum_{\boldsymbol{j}\in Z_J} I^{Z_{J-j_1}}\alpha^s_{\boldsymbol{j}}p_{:,\boldsymbol{j}}\right)\otimes\left(E_J\sum_{\boldsymbol{k}\in Z_J} I^{Z_{J-k_1}}\alpha^s_{\boldsymbol{k}}p_{\boldsymbol{k},:}\right)$$

for the evaluation of (4.12), i.e. for the discretization of nonlinear term of the Riccati-IDE in a full tensor product space. The vectors $p_{:,\boldsymbol{k}}$ and $p_{\boldsymbol{k},:}$ are defined similar to $\widehat{p}_{:,\boldsymbol{k}}$ and $\widehat{p}_{\boldsymbol{k},:}$. However, the complexity is still of the order $\mathcal{O}(N_J^2N^{\frac{d-1}{d}})$ due to the computation of the tensor product $\otimes$.

Remark 5.4. *Let an approximation of the Riccati kernel in terms of sparse grids ansatz $\widehat{p}(\zeta,x)$ from (5.2), as well as an approximate state $z_h\in Z_J$ be given. In order to compute the approximate optimal control, we have to evaluate the expression*

$$u_h(\zeta) = -\frac{\partial}{\partial\nu_\zeta}\int_\Omega \widehat{p}(\zeta,x)z_h(x)\,\mathrm{d}x. \tag{5.8}$$

The integral transform in (5.8) can be evaluated with complexity $\mathcal{O}(N_J\log N_J)$, we refer to [17, Section 2.3.4] for the details. In the case of boundary control, we can reduce the complexity further by first applying the operator $B^\star$ to the Riccati kernel. This computation corresponds to the evaluation of the feedback gain operator $K = B^\star P$ (cf. [29, Section 4]). To estimate the overall complexity for the evaluation of (5.8), we consider (5.2) and remark that the number of function with non-vanishing normal derivative in the spaces W_k is $\mathcal{O}(d2^{(d-1)k})$. Thus, the overall complexity for the computation of (5.8) is

$$\sum_{k+l\le J} d2^{(d-1)k}2^{dl} \le d\sum_{k\le J}2^{(d-1)k}2^{d(J-k+1)} = d2^{d(J+1)}\sum_{k\le J}2^{-k} = \mathcal{O}\left(dN_J^{-\frac{1}{d}}N_J\right).$$

6. Nitsche Method. In order to approximate the Riccati kernel, Theorem 3.1 suggests the usage of a scheme conforming in the space $H_0^1(\Omega\times\Omega)$, as considered in Sections 4 and 5. Although a convergence result is available for the conforming approximation of (3.8) (cf. [35]), this method is not optimal. It is possible to guarantee a quadratic rate of convergence by using an approximation method with certain additional properties, which are not fulfilled by the conforming scheme. As a particular choice of such a method, the Nitsche approximation from [45] is suggested in [35]. In this section, we discuss hence the approximation scheme for the Riccati-IDE stemming from the Nitsche method. We will use the notation $\langle\cdot,\cdot\rangle = (\cdot,\cdot)_{L^2(\Gamma)}$ for the sake of brevity. The scalar product on the space $L^2(\Omega)$ is denoted by $(\cdot,\cdot)$ as before.

6.1. Linear part. We consider the Nitsche approximation $A^\star_N$ of the operator $A^\star$ with respect to the ansatz space Z_J

$$(A^\star_N\phi,\psi) = (A^\star\phi,\psi) - \left\langle \frac{\partial\phi}{\partial\nu},\psi \right\rangle - \left\langle \phi, \frac{\partial\psi}{\partial\nu} \right\rangle + \beta 2^J \langle\phi,\psi\rangle \text{ for all } \phi,\psi\in Z_J, \tag{6.1}$$

where $\beta > 0$ is sufficiently large (cf. [45]). The discretization of the first term on the right hand side in (6.1) for the full, respectively the sparse grid, is described in Sections 4.2 and 5.2. In addition, the discretization of the boundary terms must be provided for some particular choice of the ansatz space.

Using the notation G^N_J for the discretization matrix of

$$\left\langle \frac{\partial\phi}{\partial\nu},\psi \right\rangle + \left\langle \phi, \frac{\partial\psi}{\partial\nu} \right\rangle$$

and G^D_J for the discretization matrix of $\langle\phi,\psi\rangle$, we obtain similar to Section 4.2 the following expression for the Nitsche approximation of the linear part of the Riccati-IDE

$$\big((A^\star_J - G^N_J + \beta 2^J G^D_J)\otimes E_J + E_J\otimes (A^\star_J - G^N_J + \beta 2^J G^D_J)\big)\, p_J.$$

The corresponding sparse grid expression reads

$$\big((A^\star_J - G^N_J + \beta 2^J G^D_J)\,\widehat{\otimes}\, E_J + E_J\,\widehat{\otimes}\, (A^\star_J - G^N_J + \beta 2^J G^D_J)\big)\, \widehat{p}_J.$$

Assuming the matrix–vector products for G^N_J and G^D_J can be computed in $\mathcal{O}(N_J)$, the Nitsche approximation of $A^\star$ can be evaluated with complexity $\mathcal{O}(N_J\log N_J)$ using the UniDir algorithm.

6.2. Nonlinear part. In accordance with [35], the approximation of the operator $B^\star$ for the Nitsche scheme is

$$\left\langle \left(\frac{\partial}{\partial\nu} - \beta 2^J\right)\phi,\psi \right\rangle \text{ for all } \phi,\psi\in Z_J.$$

This leads, similar to the derivation in Sections 3.1 and 4.3, to the following approximation of the nonlinear term from the Riccati-IDE

$$\int_{\Omega\times\Omega}\int_\Gamma \left(\frac{\partial}{\partial\nu_\zeta} - \beta 2^J\right) p(x,\zeta) \left(\frac{\partial}{\partial\nu_\zeta} - \beta 2^J\right) p(\zeta,\xi)\,\mathrm{d}\Gamma_\zeta\,\phi(x,\xi)\,\mathrm{d}(x,\xi),\ \ \phi\in Z_J.$$

To obtain a discrete expression, we modify the definition of the coefficients α_{i_2,j_1}, $i_2,j_1 = 1,\dots N_J$, (cf. (4.9)) for the discretization with finite elements to

$$\sum_{i_2,j_1=1}^{N_J} \left(\frac{\partial}{\partial\nu_\zeta} - \beta 2^J\right)\phi_{i_2}(\zeta) \left(\frac{\partial}{\partial\nu_\zeta} - \beta 2^J\right)\phi_{j_1}(\zeta)\,\mathrm{d}\Gamma_\zeta =: \sum_{i_2,j_1=1}^{N_J}\alpha_{i_2,j_1}. \tag{6.2}$$

Similarly, in the case of the sparse grid ansatz space, we set $\alpha^s_{\boldsymbol{j},\boldsymbol{k}}$, $\boldsymbol{j},\boldsymbol{k}\in Z_J$, to

$$\sum_{s\in S} w_s \left(\frac{\partial}{\partial\nu_\zeta} - \beta 2^J\right) \sum_{\boldsymbol{j}\in Z_J}\phi_{\boldsymbol{j}}(\zeta_s) \left(\frac{\partial}{\partial\nu_\zeta} - \beta 2^J\right) \sum_{\boldsymbol{k}\in Z_J}\phi_{\boldsymbol{k}}(\zeta_s) =: \sum_{s\in S}\sum_{\boldsymbol{j},\boldsymbol{k}\in Z_J}\alpha^s_{\boldsymbol{j},\boldsymbol{k}}. \tag{6.3}$$

With α_{i_2,j_1}, respectively $\alpha^s_{\boldsymbol{j},\boldsymbol{k}}$, as defined in (6.2) and (6.3), the expressions for the discretization of the nonlinear part are again of the form (4.12) for the finite element ansatz space, and as in (5.5) for the sparse grid.

7. Numerical results.

In this section, we present numerical results for the sparse grid discretization of the Riccati-IDE (3.8) and compare them to the standard full grid discretization.

7.1. Discretization space.

We begin with the construction of a sparse grid ansatz space $\widehat{V}_J$ on $\Omega \times \Omega$, $\Omega = [0,1]^d$. To this end we use piecewise linear hat functions (see e.g. [9]). We start with the standard linear hat function on $[0,1]$

$$\phi(x) \coloneqq \max\{1 - |x|, 0\}, \quad x \in \Omega.$$

Translation and dilatation of $\phi(x)$ is defined as

$$\phi_{(l,k)}(x) \coloneqq \phi\left(\frac{x - k \cdot 2^l}{2^l}\right) = \phi(2^{-l}x - k), \quad l \in \mathbb{N}_0,\ k \leq 2^l,$$

whereby $\phi_{(0,0)}$ and $\phi_{(0,1)}$ are restricted to Ω. The integer l is also referred to as the level and k as the index of the ansatz function.

Next, let $\boldsymbol{l}, \boldsymbol{k} \in \mathbb{N}^d$ be multi-indices and $x \in \mathbb{R}^d$. We define a piecewise d-linear function on Ω as a tensor product

$$\phi_{(\boldsymbol{l},\boldsymbol{k})}(x) \coloneqq \prod_{i=1}^{d} \phi_{(l_i,k_i)}(x_i),$$

and introduce the discrete space Z_j, $j = 0, \ldots, J$, as

$$Z_j \coloneqq \operatorname{span}\left\{\phi_{(\boldsymbol{l},\boldsymbol{k})} : \|\boldsymbol{l}\|_\infty \leq j \text{ and } k_i \leq 2^{l_i} \text{ for } i = 1, \ldots, d\right\}.$$

With spaces Z_j at hand, W_j, $W_{\boldsymbol{j}}$, $\widehat{V}_J$ can be constructed as described in Section 5. To apply the algorithms for the evaluation of the Riccati-IDE on the ansatz space $\widehat{V}_J$ constructed with d-linear hat functions, we note that the mapping δ is given by

$$\delta(j, \cdot) = \begin{cases} \left\{(\boldsymbol{l}, \boldsymbol{k}) : \|\boldsymbol{l}\|_\infty = j,\ k_i \leq 2^{l_i} \text{ odd},\ i = 1, .., d\right\}, & j \geq 1, \\ \left\{(\boldsymbol{l}, \boldsymbol{k}) : \|\boldsymbol{l}\|_\infty = 0,\ k_i \in \{0, 1\},\ i = 1, .., d\right\}, & j = 0. \end{cases}$$

7.2. Parameters of numerical experiments.

For the numerical experiments, we will consider 1D, 2D and 3D control problems, i.e., we will have $d = 1, 2, 3$. The corresponding Riccati kernels are thus defined on 2D, 4D and 6D domains.

The state dynamics operator A from (2.3) will be

$$A : H_0^1(\Omega) \cap H^2(\Omega) \to L^2(\Omega),\ Az = \sum_{i=1}^{d} \partial_{x_i x_i} z - \sum_{i=1}^{d} \partial_{x_i} z + 2 \cdot z$$

and we choose the following right-hand side $q(x, \xi)$, $x, \xi \in \mathbb{R}^d$:

$$q(x, \xi) = \prod_{i=1}^{d} (1 - |2x_i - 1|)(1 - |2\xi_i - 1|).$$

The reference solution is computed by discretizing the weak form of the ARE (3.2) with the ansatz

$$p(x, \xi) = \sum_{\|\boldsymbol{v}\|_\infty, \|\boldsymbol{w}\|_\infty = 1}^{N_{\text{ref}}} p_{\boldsymbol{v},\boldsymbol{w}} \prod_{i=1}^{d} \sqrt{2} \sin(v_i \pi x_i)\, \mathrm{e}^{\frac{1}{2} x_i}\, \sqrt{2} \sin(w_i \pi \xi_i)\, \mathrm{e}^{\frac{1}{2} \xi_i}, \tag{7.1}$$

$\boldsymbol{v}, \boldsymbol{w} \in \mathbb{N}^d$. N_{ref} is specified below for each value of d.

The algorithm for the solution of the Riccati equation is implemented based on a general sparse grids library SG++, see [49, 52].

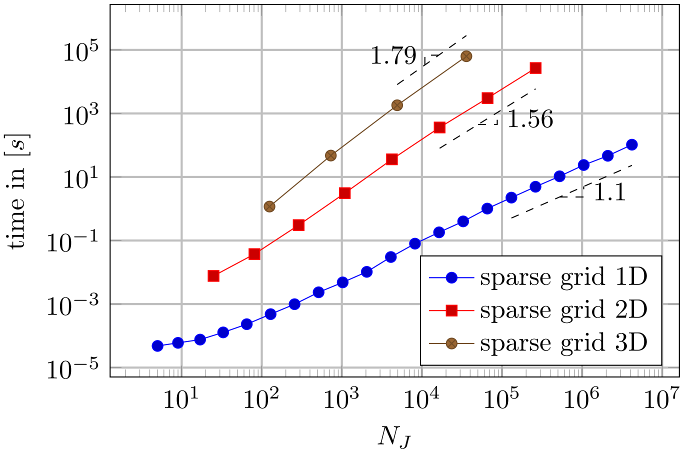

FIGURE 7.1. *Mean values of the computation times for the evaluation of the quadratic term vs. N_J – the dimension of the space Z_J.*

7.3. Computation time. First, we shall confirm the expected complexity of

$$\mathcal{O}(N_J \log N_J N_J^{\frac{d-1}{d}}) = \mathcal{O}(N_J^{\frac{2d-1}{d}} \log N_J)$$

by measuring the computation times. To this end, we use the `boost::timer` library and consider the mean values of the computation time for the evaluation of the quadratic part $\mathcal{R}_{NL}(p)$ with (4.12). The number of measurements for the computation of the mean value is 5000 for $d = 1$ and 10 for $d = 2, 3$. The results are found in Figure 7.1, which shows the logarithm of the measured time against N_J – the dimension of the space Z_J (compare Section 4).

7.4. Convergence. Next, we analyse convergence of the approximation schemes for the full and the sparse grid. Let p_{approx} denote the computed approximation of the Riccati kernel, and p_{ref} – the reference solution. For the one dimensional control problem, we estimate the L^2-error on a mesh $X_{\text{eval}} \subset \Omega \times \Omega$,

$$X \coloneqq \left\{(x, \xi) \in [0, 1]^2 \,:\, (x, \xi) = (i, j) \cdot 1/5000,\ i, j = 0, \ldots, 5000\right\},$$

by computing

$$e^2 = \sqrt{\frac{\sum_{(x,\xi)\in X_{\text{eval}}} (p_{\text{approx}}(x, \xi) - p_{\text{ref}}(x, \xi))^2}{|X_{\text{eval}}|}}. \tag{7.2}$$

For the computation of the reference solution, we set $N_{\text{ref}} = 3500$.

First, we compare the H_0^1-conforming approximation with the Nitsche method. Results for $d = 1$ are presented in Figure 7.2. Herein, 'DoF' is the number of degrees of freedom for the approximation, i.e., the dimension of the ansatz spaces V_J and $\widehat{V}_J$, respectively. One clearly figures out a stagnation of convergence for the H_0^1-conforming approximation, which can be explained by the results proven in [35]. Thus, for the subsequent experiments, we will use the Nitsche approximation scheme.

The results for $d = 1$ are presented in Figure 7.3 and Table 7.1. In the table, we also compute the convergence rates $\rho_i = \operatorname{ld}(e_i/e_{i-1})$, where e_i is the value of error estimator, e^2 or e^∞, for level i. We observe indeed that the convergence of the sparse

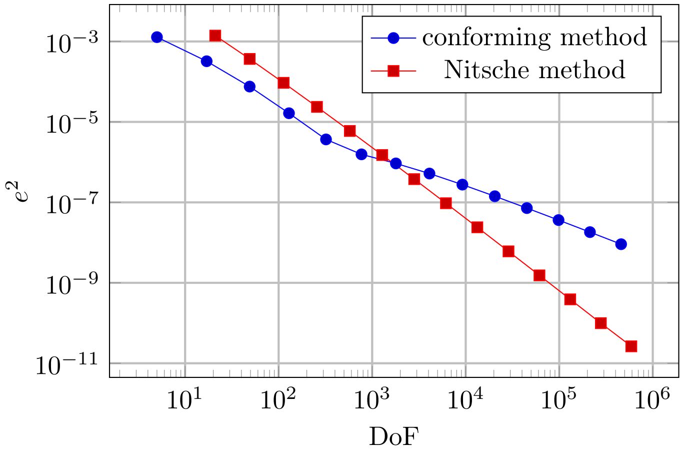

FIGURE 7.2. *H_0^1-conforming approximation versus Nitsche method. Both methods with sparse grids, $d = 1$.*

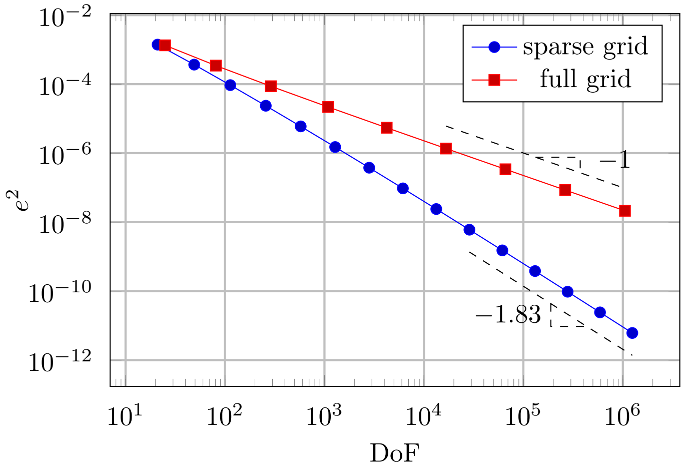

FIGURE 7.3. *Error estimation e^2 of the L^2 error versus the number of degrees of freedom (DoF) of the ansatz space, $d = 1$.*

grid approach is considerably faster compared to the full tensor product approach. We shall in addition mention that also the cost per degrees of freedom is smaller in case of the sparse grid approach, meaning that the speed-up is even higher as seen in Figure 7.3.

A quadrature on a full grid is too expensive to compute the error estimate for four or six dimensional Riccati kernels. Therefore, in these cases, we estimate the L^2 error as

$$
\begin{aligned}
(7.3)\qquad e^2 &= \|p_{\mathrm{ref}}(x,\xi) - p_{\mathrm{approx}}(x,\xi)\|_{L^2} \\
&= \left[\|p_{\mathrm{ref}}(x,\xi)\|_{L^2}^2 - 2\langle p_{\mathrm{ref}}(x,\xi), p_{\mathrm{approx}}(x,\xi)\rangle + \|p_{\mathrm{approx}}(x,\xi)\|_{L^2}^2\right]^{1/2},
\end{aligned}
$$

whereby we use the ansatz (7.1) to evaluate the scalar products directly. Note that

level	e^2	$\rho_i(e^2)$	DoF	level	e^2	$\rho_i(e^2)$	DoF
2	1.40_{-3}	$\star$	21	10	2.39_{-8}	1.99	13,313
3	3.68_{-4}	1.93	49	11	6.02_{-9}	1.99	28,673
4	9.37_{-5}	1.97	113	12	1.52_{-9}	1.99	61,441
5	2.37_{-5}	1.99	257	13	3.81_{-10}	1.99	131,073
6	5.96_{-6}	1.99	577	14	9.60_{-11}	1.99	278,529
7	1.50_{-6}	1.99	1,281	15	2.42_{-11}	1.99	589,825
8	3.78_{-7}	1.99	2,817	16	6.11_{-12}	1.99	1,245,185
9	9.51_{-8}	1.99	6,145	$\star$	$\star$	$\star$	$\star$

Table 7.1

Estimations e^2 of the L^2 errors and the convergence rates $\rho_i(e^2) = \mathrm{ld}(e^2_{i-1}/e^2_i)$ for sparse grid and $d = 1$.

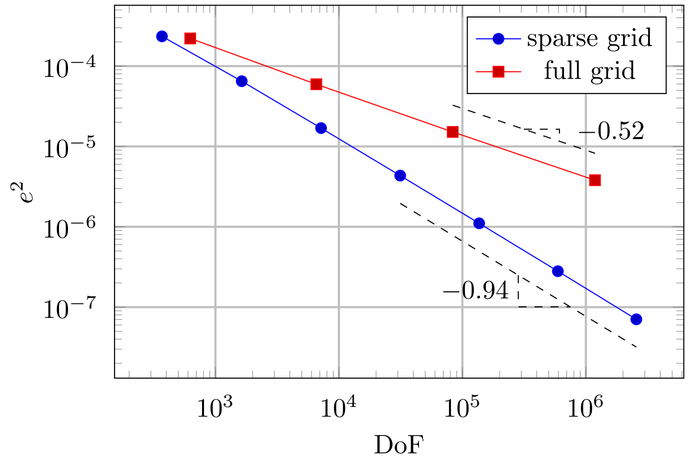

Figure 7.4. *Error estimation e^2 of the L^2 error versus the number of degrees of freedom (DoF) of the ansatz space, $d = 2$.*

the error estimate (7.2) we use for the one dimensional case can be obtained from (7.3) by virtue of a numerical quadrature. Thus, the difference between the formulas (7.2) and (7.3) would be neglectable for sufficiently large N_{ref}.

The approximation results are presented in Figure 7.4 and Table 7.2 for $d = 2$, and in Figure 7.5 and Table 7.3 for $d = 3$. We set $N_{\mathrm{ref}} = 130$ for the two dimensional, and $N_{\mathrm{ref}} = 30$ for the three dimensional control problem. It again turns out that the sparse grid approach is superior over the full tensor product approach. The sparse grid approach realizes a higher rate of convergence with respect to the degrees of freedom at a smaller cost per degree of freedom.

level	e^2	$\rho_i(e^2)$	DoF	level	e^2	$\rho_i(e^2)$	DoF
2	2.35_{-4}	$\star$	369	6	1.10_{-6}	1.97	136,705
3	6.49_{-5}	1.85	1,633	7	2.80_{-7}	1.98	593,409
4	1.69_{-5}	1.94	7,169	8	7.06_{-8}	1.99	2,564,100
5	4.34_{-6}	1.96	31,361	$\star$	$\star$	$\star$	$\star$

Table 7.2

Estimations e^2 of the L^2 and e^∞ of the L^∞ errors and the convergence rates $\rho_i(e) = \mathrm{ld}(e_{i-1}/e_i)$ for sparse grid and $d = 2$.

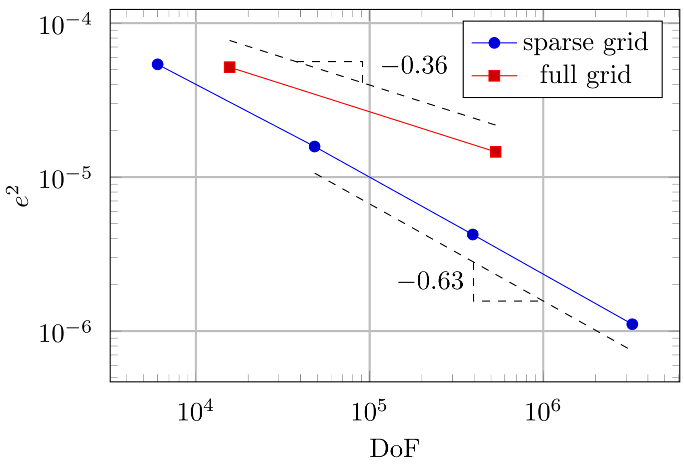

FIGURE 7.5. *Error estimation e^2 of the L^2 error versus the number of degrees of freedom (DoF) of the ansatz space, $d = 3$.*

level	e^2	$\rho_i(e^2)$	DoF	level	e^2	$\rho_i(e^2)$	DoF
2	5.40_{-5}	$\star$	6,021	4	4.24_{-6}	1.90	392,561
3	1.58_{-5}	1.77	48,241	5	1.11_{-6}	1.94	3,252,740

TABLE 7.3

Estimations e^2 of the L^2 and e^∞ of the L^∞ errors and the convergence rates $\rho_i(e) = \mathrm{ld}(e_{i-1}/e_i)$ for sparse grid and $d = 3$.

8. Conclusion. In the present article, we considered the numerical solution of the algebraic Riccati equation by means of sparse grids. To that end, we did not start with the algebraic Riccati equation but with its continuous counterpart – the Riccati-IDE. This partial integro-differential equation has then been discretized by the Galerkin method with sparse grid ansatz spaces. We have shown that both, memory requirements and computation times, are reduced considerably in comparison with a tensor-product finite element discretization. Nonetheless, future research has to be focus on further speeding-up the computational process.

Acknowledgement. This research has been funded in parts by the Swiss National Science Foundation (SNF) under project number 407540_167186 NFP 75 Big Data.

References.

[1] Stefan ACHATZ, *Adaptive finite Dünngitter-Elemente höherer Ordnung für elliptische partielle Differentialgleichungen mit variablen Koeffzienten*, Dissertation, Technische Universität München, München, Germany, 2003.

[2] Richard BELLMAN, *Introduction to matrix analysis*, McGraw-Hill, 1970.

[3] Peter BENNER, *Computational Methods for Linear-Quadratic Optimization*, Research report 98-04, Zentrum für Technomathematik, Universität Bremen, 1998.

[4] Peter BENNER and Jens SAAK, *Numerical solution of large and sparse continuous time algebraic matrix Riccati and Lyapunov equations: a state of the art survey*, GAMM-Mitt. 36.1 (2013), pp. 32–52.

[5] Peter Benner et al., *A numerical comparison of different solvers for large-scale, continuous-time algebraic Riccati equations.* SIAM Journal on Scientific Computing 42.2 (2020), A957–A996.

[6] Alain Bensoussan et al., *Representation and Control of Infinite Dimensional Systems*, Birkhäuser, Boston, 2007.

[7] Dario A. Bini, Bruno Iannazzo, and Beatrice Meini, *Numerical Solution of Algebraic Riccati Equations*, vol. 9, SIAM, 2012.

[8] Hans-Joachim Bungartz, *A Multigrid algorithm for higher order finite elements on spase grids*, Electronic Transactions on Numerical Analysis 6 (1997), pp. 63–77.

[9] Hans-Joachim Bungartz and Michael Griebel, *Sparse grids*, Acta Numerica 13 (2004), pp. 1–123.

[10] John A. Burns and Kevin P. Hulsing, *Numerical methods for approximating functional gains in LQR boundary control problems*, Mathematical and Computer Modelling 33.33 (2001), pp. 89–100.

[11] Martin Costabel, *Boundary integral operators for the heat equation*, Integral equations and Operator Theory 13 (1990), pp. 498–552.

[12] Martin Costabel, *Boundary Integral operators on Lipschitz Domains: Elementary Results*, SIAM Journal on Mathematical Analysis 19 (1988), pp. 613–626.

[13] Ruth F. Curtain and Hans Zwart, *An Introduction to Infinite-Dimensional Linear Systems Theory*, Texts in Applied Mathematics, Springer, New York, 1995.

[14] Wolfgang Dahmen, *Wavelet and multiscale methods for operator equations*, Acta Numerica 6 (1997), pp. 55–228.

[15] Nelson Dunford and Jacob T. Schwartz, *Linear Operators*, vol. II: Spectral Theory, John Wiley & Sons, 1988.

[16] L. C. Evans, *Partial Differential Equations*, American Mathematical Society, 1998.

[17] Christian Feuersänger, *Sparse Grids Methods for Higher Dimensional Approximation*, Dissertation, Rheinische Friedrich–Wilhelms–Universität Bonn, Bonn, Germany, 2010.

[18] Franco Flandoli, *Algebraic Riccati equation arising in boundary control problems*, SIAM Journal on Control and Optimization 25.3 (1987), pp. 612–636.

[19] Jochen Garcke, *Sparse grids in a nutshell*, in: *Sparse grids and applications*, ed. by Jochen Garcke and Michael Griebel, vol. 88, Lecture Notes in Computational Science and Engineering, Springer, Berlin-Heidelberg, 2013, pp. 57–80.

[20] Lars Grasedyck and Wolfgang Hackbusch, *A multigrid method to solve large scale Sylvester equations*, SIAM Journal on Matrix Analysis and Applications 29.3 (2007), pp. 870–894, issn: 1095-7162.

[21] Lars Grasedyck, Wolfgang Hackbusch, and Boris N. Khoromskij, *Solution of large scale algebraic matrix Riccati equations by use of hierarchical matrices*, Computing 70.2 (2003), pp. 121–165.

[22] Serkan Gugercin and Athanasios C. Antoulas, *A survey of model reduction by balanced truncation and some new results*, International Journal of Control 77.8 (2004), pp. 748–766.

[23] Bernard Haasdonk and Andreas Schmidt, *Reduced basis approximation of large scale parametric algebraic Riccati equations.* ESAIM: COCV 24.1 (2018), pp. 129–151.

[24] Wolfgang HACKBUSCH, *Elliptic Differential Equations, Theory and Numerical Treatment*, Springer, Germany, 2017.

[25] Wolfgang HACKBUSCH, *Tensor Spaces and Numerical Tensor Calculus*, Springer, Berlin-Heidelberg, 2012.

[26] Helmut HARBRECHT, *A finite element method for elliptic problems with stochastic input data*, Applied Numerical Mathematics 60.227–244 (2010).

[27] Helmut HARBRECHT, Reinhold SCHNEIDER, and Christoph SCHWAB, *Multilevel frames for sparse tensor product spaces*, Numerische Mathematik 110.199–220 (2008).

[28] Helmut HARBRECHT and Christoph SCHWAB, *Sparse tensor finite elements for elliptic multiple scale problems*, Computer Methods in Applied Mechanics and Engineering 200.45–46 (2011), pp. 3100–3110.

[29] Kevin P. HULSING, *Methods for Computing Functional Gains for LQR Control of Partial Differential Equations*, Dissertation, Virginia Polytechnic Institute and State University, Virginia, USA, 1999.

[30] Edmond A. JONCKHEERE and Leonard M. SILVERMAN, *A new set of invariants for linear systems – Application to reduced order compensator design*, IEEE Transactions on Automatic Control 28.10 (1983), pp. 953–964.

[31] Rudolf E. KÁLMÁN and Richard S. BUCY, *New results in linear filtering and prediction theory*, Journal of Basic Engineering 83.1 (1961), pp. 95–108.

[32] Belinda B. KING, *Representation of feedback operators for parabolic control problems*, Proceedings of the American Mathematical Society 128.5 (2000), pp. 89–100.

[33] David L. KLEINMAN, *On an iterative technique for Riccati equation computations*, IEEE Transactions on Automatic Control 13.1 (1968), pp. 114–115.

[34] Vladimiír KUČERA, *A review of the matrix Riccati equation*, Kybernetika 9.2 (1973), pp. 42–61.

[35] Irena LASIECKA, *Convergence rates for the approximations of the solutions to algebraic Riccati Equations with unbounded coefficients: case of analytic semigroups*, Numerische Mathematik 63 (1992), pp. 357–390.

[36] Irena LASIECKA, *Unified Theory for Abstract Parabolic Boundary Probblems – A Semigroup Approach*, Applied Mathematics and Optimization 6 (1980), pp. 287–333.

[37] Irena LASIECKA and Roberto TRIGGIANI, *Control Theory for Partial Differential Equations: Continuous and Approximation Theories*, vol. I: Abstract Parabolic Systems, Encyclopedia of Mathematics and Its Applications, Cambridge University Press, 1999.

[38] Irena LASIECKA and Roberto TRIGGIANI, *The regulator problem for parabolic equations with Dirichlet boundary control; Part I: Riccati's feedback synthesis and regularity of optimal solutions*, Applied Mathematics and Optimization 16 (1987), pp. 147–168.

[39] Irena LASIECKA and Roberto TRIGGIANI, *The regulator problem for parabolic equations with Dirichlet boundary control; Part II: Galerkin approximation*, Applied Mathematics and Optimization 16 (1987), pp. 198–216.

[40] Jacques-Louis LIONS, *Optimal Control of Systems Governed by Partial Differential Equations*, Springer, Berlin-Heidelberg, 1971.

[41] Jacques-Louis LIONS and Enrico MAGENES, *Non-Homogeneous Boundary Value Problems and Applications*, vol. Volume I, Springer, Berlin-Heidelberg, 1972.

[42] Lennart Ljung, Thomas Kailath, and Benjamin Friedlander, *Scattering theory and linear least squares estimation – Part I: Continuous-time problems*, Proceedings of the IEEE 64.1 (1976), pp. 131–139.

[43] Hermann Mena, *Numerical Solution of Differential Riccati Equations Arising in Optimal Control Problems for Parabolic Partial Differential Equations*, Dissertation, Escuela Politécnica Nacional, Quito, Ecuador, 2007.

[44] Jindřich Nečas, *Direct Methods in the Theory of Elliptic Equations*, Springer, Berlin-Heidelberg, 2012.

[45] Joachim Nitsche, *Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind*, Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg 36 (1971), pp. 9–15.

[46] Mark R. Opmeer, *Decay of singular values of the Gramians of infinite-dimensional systems*, in: *ECC15 – European Control Conference*, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 1183–1188.

[47] Mark R. Opmeer, Timo Reis, and Winnifried Wollner, *Finite-Rank ADI Iteration for Operator Lyapunov Equations*, SIAM Journal on Control and Optimization 51.5 (2013), pp. 4084–4117.

[48] Amnon Pazy, *Introduction to Algorithms.* Vol. 44, Springer, New York, 1983.

[49] Dirk Pflüger, *Spatially Adaptive Sparse Grids for High-Dimensional Problems*, Dissertation, Institut für Informatik, Technische Universität München, München, Germany, 2010.

[50] I. G. Rosen, *Convergence of Galerkin Approximations for Operator Riccati Equations – A Nonlinear Evolution Equations Approach.* Journal of Mathematical Analysis and Applications 155 (1991), pp. 226–248.

[51] Fredi Tröltzsch, *Optimale Steuerung partieller Differentialgleichungen, Theorie, Verfahren und Anwendungen*, Vieweg+Teubner Verlag, Wiesbaden, 2009.

[52] Julian Valentin and Dirk Pflüger, *Hierarchical gradient-based optimization with B-splines on sparse grids*, in: *Sparse Grids and Applications – Stuttgart 2014*, ed. by Jochen Garcke and Dirk Pflüger, vol. 109, Lecture Notes in Computational Science and Engineering, Springer International Publishing, Switzerland, 2016, pp. 315–336.

[53] Andreas Zeiser, *Fast matrix-vector multiplication in the sparse-grid Galerkin method*, Journal of Scientific Computing 47.3 (2011), pp. 328–346.