IEEE Control Systems Letters paper presented at
2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Towards Grassmannian Dimensionality Reduction in MPC

Roland Schurig*, Andreas Himmel* and Rolf Findeisen*

***Abstract*— Model predictive control presents remarkable potential for the optimal control of dynamic systems. However, the necessity for an online solution to an optimal control problem often renders it impractical for control systems with limited computational capabilities. To address this issue, specialized dimensionality reduction techniques designed for optimal control problems have been proposed. In this paper, we introduce a methodology for designing a low-dimensional subspace that provides an ideal representation for a predefined finite set of high-dimensional optimizers. By characterizing the subspace as an element of a specific Riemannian manifold, we leverage the unique geometric structure of the subspace. Subsequently, the optimal subspace is identified through optimization on the Riemannian manifold. The dimensionality reduction for the model predictive control scheme is achieved by confining the search space to the optimized low-dimensional subspace, enhancing both efficiency and applicability.**

I. Introduction

At the core of a model predictive control scheme, a constrained optimization problem must be solved online at every sampling instance over a finite horizon [1], [2]. The first element of the optimal sequence is applied to the plant, and the procedure is repeated with a shifted horizon at the subsequent time instance. However, the repeated online solution of the optimal control problem might be challenging, depending on the hardware that solves the problem, the size of the plant under consideration, the prediction horizon of the MPC scheme, and the sampling time of the control system. In this paper, the key idea to alleviate the computational burden is to reduce the dimension of the decision variables.

One approach to reduce the number of decision variables in MPC is *move blocking* [3], [4]. Other methods to reduce the dimension rely on either *singular value decomposition* (SVD) [5], [6], [7] or *Laguerre functions* [8], [9], see [10] for further details. Recently, an approach to ensure stability and recursive feasibility for an MPC scheme with a lower-dimensional input space $\mathcal{U}$ was presented [10]. Therein *recursive* feasibility is established by constructing an admissible solution for the current time step from the admissible (sub-)optimal solution from the last time step, and handing the admissible guess to the solver as a "fall-back" option. Considering the shifted sequence from the last time step is also the classical procedure to establish recursive feasibility in move blocking [3], [4]. At the very first time step, however, the *original* (i.e., not reduced in dimension) MPC scheme is solved in [10], which establishes *initial* feasibility in their case. This requirement to solve the original problem once is exactly what we seek to avoid.

* TU Darmstadt, Control and Cyber-Physical Systems Laboratory, {roland.schurig, andreas.himmel,rolf.findeisen}@iat.tu-darmstadt.de

The authors acknowledge support within the frame of the KI-Embedded project funded by the BMWK and the LOEWE initiative emergenCITY.

Our contribution improves on that by explicitly considering *initial* feasibility in the lower-dimensional input space $\mathcal{U}$. More precisely, given a data set $\mathcal{C}$ containing a finite number of feasible states x_r and the corresponding optimal input sequences z_r, we aim to construct – in a systematic way – an $\mathcal{U}$ which admits at least one admissible input sequence for all states x_r. That is, we do not loose feasibility for the states x_r in $\mathcal{C}$. On the other hand, we still want to be as close to the optimal solution as possible.

The approach is related to [5], [11], with the key difference that we translate the problem to the Grassmann manifold, which is a specific Riemannian manifold. We obtain the lower-dimensional input space $\mathcal{U}$ via an optimization on the Grassmann manifold. This allows for a rather "natural" problem formulation since the resulting algorithm exploits the geometric structure of the problem [12], [13], [14].

Thus, this paper considers a similar problem as [10], but takes a different approach in shifting the focus on feasibility during the construction of $\mathcal{U}$. The advantage is that the reduced-order MPC scheme can then be used without ever solving the original high-order problem online.

II. Preliminaries

We use differential and Riemannian geometry. Thus we require a series of mathematical objects which we briefly outline. For further details see [15], [16].We consider Riemannian manifolds $\left(\mathcal{M}, g\right)$ consisting of a differentiable manifold $\mathcal{M}$ and a metric g. The tangent space to $\mathcal{M}$ at a point $p \in \mathcal{M}$ is denoted by $T_p\mathcal{M}$. The tangent bundle $T\mathcal{M}$ is the disjoint union of all tangent spaces, i.e., $T\mathcal{M} = \dot{\bigcup}_{p\in\mathcal{M}} T_p\mathcal{M}$. The metric g defines at each point $p \in \mathcal{M}$ an inner product on the tangent space via $\langle\cdot,\cdot\rangle_p \colon T_p\mathcal{M} \times T_p\mathcal{M} \to \mathbb{R}$, $(\xi,\eta) \mapsto \langle\xi,\eta\rangle_p \coloneqq g_p(\xi,\eta)$, where g_p denotes the evaluation of g at p. Let $\phi \colon \mathcal{M} \to \mathbb{R}$ be a real-valued function on $\mathcal{M}$. The differential (or derivative) of ϕ at p is a linear map $\mathrm{d}\,\phi(p) \colon T_p\mathcal{M} \to \mathbb{R}$. Using the metric g, we can assign to the differential $\mathrm{d}\,\phi(p)$ an unique tangent vector $\operatorname{grad}\phi(p) \in T_p\mathcal{M}$ called Riemannian gradient of ϕ at $p \in \mathcal{M}$ defined by $\langle \operatorname{grad}\phi(p), \cdot\rangle_p = \mathrm{d}\,\phi(p)[\cdot]$.
A vector field assigns to each point $p \in \mathcal{M}$ an element of the tangent space $T_p\mathcal{M}$. The set of all smooth vector fields on $\mathcal{M}$ is denoted by $\mathfrak{X}(\mathcal{M})$. The Riemannian connection on $\mathcal{M}$ is a map $\nabla \colon \mathfrak{X}(\mathcal{M}) \times \mathfrak{X}(\mathcal{M}) \to \mathfrak{X}(\mathcal{M})$, $(\eta,\xi) \mapsto \nabla_\eta\,\xi$ measuring how a vector field ξ changes when moving in the direction of another field η. Based on the concept of connection, one can define the Riemannian Hessian of ϕ at $p \in \mathcal{M}$ as $\operatorname{Hess}\phi(p) \colon T_p\mathcal{M} \to T_p\mathcal{M}$, $\eta \mapsto \operatorname{Hess}\phi(p)[\eta] \coloneqq \nabla_\eta \operatorname{grad}\phi(p)$. Furthermore, a concept related to optimization on manifolds is that of retraction, which is a map $R \colon T\mathcal{M} \to \mathcal{M}$. Its restriction to $T_p\mathcal{M}$

is denoted by R_p. If the manifold has the structure of a vector space, the directional derivative of a function defined on it can be given more directly. Specifically, we consider the vector space $\mathbb{R}^{d\times p}$ of all $d \times p$ matrices. This vector space is called *Euclidean* if the metric defined on it admits an inner product of the form $\langle\cdot,\cdot\rangle\colon \mathbb{R}^{d\times p}\times\mathbb{R}^{d\times p}\to\mathbb{R}$, $(A,B)\mapsto\langle A,B\rangle\coloneqq\operatorname{tr}\left(A^\top B\right)$. Here, $\operatorname{tr}(\cdot)$ denotes the trace of a matrix. Furthermore, let $f\colon\mathbb{R}^{d\times p}\to\mathbb{R}$ be a function on an Euclidean space, we define the Euclidean gradient component wise by $\left(\operatorname{grad}^{\mathrm{eucl}} f\right)_{ij}\coloneqq\partial_{ij}f$, where $\partial_{ij}f\colon\mathbb{R}^{d\times p}\to\mathbb{R}$ denotes the partial derivative of f with respect to the input component (i,j). The gradient $\operatorname{grad}^{\mathrm{eucl}} f(p)\in\mathbb{R}^{d\times p}$ at a point $p\in\mathbb{R}^{d\times p}$ is again a $d\times p$ matrix. To denote positive real numbers, we use $\mathbb{R}_{>0}\coloneqq\{x\in\mathbb{R}\mid x>0\}$ and analogously $\mathbb{R}_{\geq 0}$. The identity in $\mathbb{R}^d$ is denoted by I_d and we use $x^j\in\mathbb{R}$ for the j^{th} component of $x\in\mathbb{R}^d$.

III. Problem Formulation

For simplicity, we focus on linear systems with states $x\in\mathbb{R}^n$ and inputs $u\in\mathbb{R}^m$, subject to state constraints $\mathbb{X}$ and input constraints $\mathbb{U}$. We define the MPC problem over a finite horizon $N\in\mathbb{N}$ with a positive definite quadratic stage cost, a positive semidefinite quadratic terminal cost, and a terminal constraint set, for further details on MPC see [1].

The *condensed* MPC optimization problem is given by:

$$\mathscr{P}(x)\colon \underset{z\in\mathbb{R}^d}{\text{minimize}}\ J_N(x,z),\ \text{subject to}\ c_N(x,z)\leq 0. \quad (1)$$

Here, $z\in\mathbb{U}^N\subseteq\mathbb{R}^d$ represents the input sequence of length N and $d\coloneqq Nm$. The function $c_N\colon\mathbb{R}^d\times\mathbb{R}^d\to\mathbb{R}^q$ collects all constraints, i.e., all state and input constraints over the horizon N. We call $\mathscr{P}(x)$ a *multiparametric quadratic program* (mp-QP) with parameter $x\in\mathbb{X}$ (the initial state) and decision variable $z\in\mathbb{U}^N$ [17].

Succeeding, we use the following objects: The *feasible set* $\mathcal{X}_N$, which is given by $\mathcal{X}_N=\{x\in\mathbb{R}^n\mid\exists z\in\mathbb{R}^d\colon c_N(x,z)\leq 0\}$. The *set of admissible inputs* of length N for $x_0\in\mathcal{X}_N$ defined by $\mathbb{U}^N(x_0)\coloneqq\{z\in\mathbb{R}^d\mid c_N(x_0,z)\leq 0\}$. The *optimal value function* $V_N\colon\mathcal{X}_N\to\mathbb{R}$, which maps any $x_0\in\mathcal{X}_N$ to the optimal value of (1), i.e., $V_N(x_0)\leq J_N(x_0,z)$ holds for all $z\in\mathbb{U}^N(x_0)$. The *optimizer mapping* $\mu_N\colon\mathcal{X}_N\to\mathbb{U}^N$, which maps any $x_0\in\mathcal{X}_N$ to the corresponding optimizer of (1), i.e., $V_N(x_0)=J_N(x_0,\mu_N(x_0))$.

A classical approach to diminish the computational burden in $\mathscr{P}(x)$ is to reduce the number of (scalar) decision variables from d to p, with $p\ll d$, see, e.g., [10], [5], [3].

Our main idea is to restrict the search for the optimizer $\mu_N(x)$ of $\mathscr{P}(x)$ to a p-dimensional subspace $\mathcal{U}$. The challenge is to design a subspace which leads to a good approximation, while providing admissible input sequences for a range of relevant initial states $x_0\in\mathcal{X}_N$.

Subspace approximation: For the subspace approximation we introduce the *Grassmann manifold* $\mathcal{GR}(p,d)$, which is defined as the set of all p-dimensional linear subspaces of $\mathbb{R}^d$ [15], [18], [19], i.e.,

$$\mathcal{GR}(p,d)\coloneqq\{\mathcal{U}\subset\mathbb{R}^d\mid\mathcal{U}\text{ is a subspace, }\dim(\mathcal{U})=p\}. \quad (2)$$

In the following, we omit the term "linear" for readability.

TABLE I: Different perspectives of the Grassmann manifold.

Subspace perspective	Projector perspective	Orthonormal Basis (ONB) perspective
$\mathcal{U}\in\mathcal{GR}(p,d)$	$P\in\mathcal{GR}(p,d)$	$[U]\in\mathcal{GR}(p,d)$
subspace $\mathcal{U}$ of $\mathbb{R}^d$	$\mathcal{U}=\operatorname{range}(P)$	$\mathcal{U}\cong P=\pi(U)$
refers to the subspace as a subset of $\mathbb{R}^d$	emphasize that any point in $\mathbb{R}^d$ can be projected onto the subspace & to define a function on $\mathcal{GR}(p,d)$	emphasize that any point in the subspace has a representative in $\mathbb{R}^p$ via a matrix in $[U]$
$z\in\mathcal{U}$	$z_{\mathrm{p}}=Pz\in\mathcal{U},\ z\in\mathbb{R}^d$	$z_{\mathrm{p}}=Uv\in\mathcal{U},\ v\in\mathbb{R}^p$
defined in (2)	defined in (3)	defined in (4)

One can describe a subspace via the "projector perspective" [19]. There, we identify a subspace $\mathcal{U}$ with the unique orthogonal projector P onto $\mathcal{U}$. Thus, with slight abuse of notation, we identify

$$\mathcal{GR}(p,d)=\{P\in\mathbb{R}^{d\times d}\mid P^\top=P,\ P^2=P,\operatorname{rank}(P)=p\}. \quad (3)$$

This way, we associate a subspace $\mathcal{U}$ with the range of the orthogonal projector P. Any projector $P\in\mathcal{GR}(p,d)$ can be identified with a set of orthogonal matrices whose columns span the same subspace as P [19]. To this end, we introduce the compact *Stiefel manifold* $\mathcal{ST}(p,d)\coloneqq\{U\in\mathbb{R}^{d\times p}\mid U^\top U=I_p\}$. The projection map $\pi\colon\mathcal{ST}(p,d)\to\mathcal{GR}(p,d),\ U\mapsto\pi(U)\coloneqq UU^\top$ establishes a link between the two spaces [19], i.e., $\pi(U)$ is a projector.

Let the subspace $\mathcal{U}$ be associated with the projector $P=\pi(U)$. Then $\mathcal{U}=\operatorname{span}(U)$. The *equivalence class* $[U]\coloneqq\left\{\widetilde{U}\in\mathcal{ST}(p,d)\mid\widetilde{U}=UR,\ R\in\mathrm{O}(p)\right\}$ for $U\in\mathcal{ST}(p,d)$ collects all elements of $\mathcal{ST}(p,d)$ whose columns span the same subspace, i.e., $\operatorname{span}(U)=\operatorname{span}(\widetilde{U})$ holds for all $\widetilde{U}\in[U]$.[1]Here, $\mathrm{O}(p)\coloneqq\{R\in\mathbb{R}^{p\times p}\mid R^\top R=I_p=RR^\top\}$ denotes the orthogonal group of dimension p.

This equivalence class, gives rise to yet another identification of a subspace $\mathcal{U}$, namely $\mathcal{U}\cong[U]$, where $\mathcal{U}=\operatorname{span}(U)$. Thus, for this identification, we may write – with another slight abuse of notation – the Grassmann manifold as the quotient space [19]

$$\mathcal{GR}(p,d)=\mathcal{ST}(p,d)/\mathrm{O}(p). \quad (4)$$

Table I provides an overview of the notations.

Reduced-Order MPC Problem: The previous section outlined several ways to represent a subspace $\mathcal{U}\in\mathcal{GR}(p,d)$. For our purposes, the ONB perspective has numerical advantages, as outlined next. Given the assumption that the optimizers $\mu_N(x)$ of $\mathscr{P}(x)$ mainly evolve in a subspace $\mathcal{U}\cong[U]\in\mathcal{GR}(p,d)$, we introduce the approximation

$$z\approx Uv, \quad (5)$$

which can be handed to the condensed MPC problem $\mathscr{P}(x)$ (1), turning it into a minimization with respect to v, thereby reducing the dimension of the decision variable from d to p.

First observe that both admissibility and suboptimality of the approximation solely depend on the subspace $\mathcal{U}\cong[U]\in\mathcal{GR}(p,d)$, and not on the chosen Stiefel representative in (5). We state this observation in Lemma 1.

Lemma 1. *Consider $U\in\mathcal{ST}(p,d)$ and $\widetilde{U}=UR\in[U]$. Assume that $x\in\mathcal{X}_N$, $z=Uv\in\mathbb{U}^N(x)$ and let $\tilde{v}=R^\top v$. Then: (i) $\widetilde{U}\tilde{v}\in\mathbb{U}^N(x)$, and (ii) $J_N(x,\widetilde{U}\tilde{v})=J_N(x,Uv)$.*

[1]Since $\widetilde{U}^\top\widetilde{U}=I_p$ for all $\widetilde{U}\in[U]$, such a basis of $\mathcal{U}$ is *orthonormal* [13]

Proof. Note that $\tilde{z} \coloneqq \widetilde{U}\tilde{v} = \widetilde{U}R^\top v = Uv = z$. It follows that $\tilde{z} \in \mathbb{U}^N(x)$ and $J_N(x,\tilde{z}) = J_N(x,z)$. □

What remains open is how exactly the subspace $\mathcal{U} \cong P = \pi(U)$ should be designed so that the suboptimality introduced by the approximation in (5) is purposeful. In order to discuss the approximation quality, we consider a given data set

$$\mathcal{C} = \{(x_r, z_r)\}_{r=1}^L \subset \mathcal{X}_N \times \mathbb{U}^N, \tag{6}$$

where $z_r = \mu_N(x_r)$. Clearly, a good representation of all optimal input sequences z_r in $\mathcal{U} \cong [U]$ is desirable. On the other hand, being close to all z_r is futile if the respective reduced-order admissible sets $\mathbb{U}^N(x_r) \cap \mathcal{U}$ are empty, i.e., if no admissible input sequence for x_r is contained in $\mathcal{U}$. Accordingly, admissibility for $\mathcal{C}$ becomes our first priority.

Our idea is to ensure that the *projection of z_r onto $\mathcal{U}$* is contained in $\mathbb{U}^N(x_r)$. To this end, we introduce the mapping

$$g_r \colon \mathcal{GR}(p,d) \to \mathbb{R}^q,\ P \mapsto g_r(P) \coloneqq c_N(x_r, Pz_r), \tag{7}$$

such that $g_r(P) \leq 0 \Leftrightarrow Pz_r \in \mathbb{U}^N(x_r)$ holds. Furthermore, $\mathcal{A}_{p,d} \coloneqq \{P \in \mathcal{GR}(p,d) \mid \forall r \in \mathcal{I}_L \colon g_r(P) \leq 0\} \subset \mathcal{GR}(p,d)$ describes the set of projectors that render the projection of *all* z_r onto $\mathcal{U} \cong P$ admissible, where $\mathcal{I}_L \coloneqq \{1, \dots, L\}$ for $1 \leq L \in \mathbb{N}$. We can now state the following design problem:

Problem 1 (Subspace design)**.** Consider the data set $\mathcal{C}$ in (6). Design a subspace $\mathcal{U} \cong P \in \mathcal{GR}(p,d)$ such that

(i) The subspace $\mathcal{U}$ is *admissible* for $\mathcal{C}$: $P \in \mathcal{A}_{p,d}$.
(ii) The subspace $\mathcal{U}$ minimizes $J_N(x_r, Pz_r) - V_N(x_r)$ for all $r \in \mathcal{I}_L$. This we will call *good performance*.

Remark 1. Our main contribution is the inclusion of the subspace admissibility (i) directly into the subspace *design*. Since $Pz_r \in \mathbb{U}^N(x_r)$ for all $r \in \mathcal{I}_r$, i.e., (1) has a solution for x_r in $\mathcal{U}$, stability and recursive feasibility of the reduced MPC scheme are established by the arguments in [10], without ever needing to explicitly solve the original problem (1).

In order to solve Problem 1, we choose an optimization based approach, to take care of condition (ii). To this end, the following section discusses suitable objective functions.

Objective functions on the Grassmann manifold: Consider a subspace $\mathcal{U} \cong [U]$, identified with the projector $P = \pi(U) \in \mathcal{GR}(p,d)$, and the data set $\mathcal{C}$ in (6). In this section, we introduce two objective functions, each tailored to one of the conditions in Problem 1.

First, we consider the condition (i). As described in the next section, the optimization on the Grassmann manifold does not allow to explicitly impose constraints of the form $g_r(P) \leq 0$ on $\mathcal{U} \cong P \in \mathcal{GR}(p,d)$. Therefore, we include the constraints in a *penalty function* $\psi \colon \mathcal{GR}(p,d) \to \mathbb{R}_{\geq 0}$ which takes values close to zero if $g_r(P) \leq 0$ holds for all $r \in \mathcal{I}_L$, and values much larger than one if $g_r(P) > 0$ holds for at least one r. A candidate is

$$\psi(P) \coloneqq \frac{1}{Lq} \sum_{r=1}^{L} \sum_{i=1}^{q} \exp\bigl(\lambda_r^i g_r^i(P)\bigr), \tag{8}$$

where the $\lambda_r^i \in \mathbb{R}_{>0}$ are weighting factors and $g_r^i \colon \mathcal{GR}(p,d) \to \mathbb{R}$ is the i^{th} component function of g_r.

The hope is that the minimization of ψ leads to a local minimizer $P_0 \in \mathcal{A}_{p,d}$. Given this admissible subspace $\mathcal{U}_0 \cong P_0$, we further aim at improving the performance of the subspace, c.f. (ii) in Problem 1. To this end, we set $\varepsilon_r(P) \coloneqq (J_N(x_r, Pz_r) - V_N(x_r)) \cdot (V_N(x_r))^{-1}$ for $r \in \mathcal{I}_L$. Then, we introduce $\varepsilon \colon \mathcal{GR}(p,d) \to \mathbb{R}_{\geq 0},\ P \mapsto \varepsilon(P) \coloneqq \frac{1}{L}\sum_{r=1}^{L} \varepsilon_r(P)$, as a performance measure on $\mathcal{GR}(p,d)$, i.e., the mean of the relative errors in the cost function of (1) that are introduced by the projection onto $\mathcal{U} \cong P$ is minimized.

We want to ensure that $P \in \mathcal{A}_{p,d}$ holds while minimizing ε. Thus, we introduce $\beta \colon \mathcal{A}_{p,d} \to \mathbb{R},\ P \mapsto \beta(P) \coloneqq -\frac{1}{Lq}\sum_{r=1}^{L}\sum_{i=1}^{q} \log\bigl(-g_r^i(P)\bigr)$ as a *barrier function*. Finally, to minimize the performance measure while still guaranteeing admissibility, we set

$$\phi_\mu \coloneqq \varepsilon + \mu\,\beta : \mathcal{A}_{p,d} \to \mathbb{R} \tag{9}$$

as the *performance function*, with a tunable parameter $\mu > 0$. Note that, since the domain of ϕ_μ is $\mathcal{A}_{p,d}$, we need an initial guess $P_0 \in \mathcal{A}_{p,d}$ to start the minimization, c.f. the discussion for the local minimizer of ψ. We present the detailed algorithm to tackle Problem 1 in Section IV.

Minimizing an objective function $\phi \colon \mathcal{GR}(p,d) \to \mathbb{R}$ – in our case either ψ or ϕ_μ – requires special care. Consequently, we discuss next how Problem 1 can be casted as an optimization problem on the Grassmann manifold.

IV. Optimization on the Grassmann manifold

We follow the ideas outlined in [12], which discusses optimization algorithms on Riemannian manifolds. For conciseness, we introduce and comment on the iterative trust-region algorithm from [12] that we use.

Trust-region algorithm on a Riemannian manifold: Let $\mathcal{M}$ be a Riemannian manifold and $\phi \colon \mathcal{M} \to \mathbb{R}$ an objective function. The considered algorithm produces a sequence of iterates $\{p_k \in \mathcal{M}\}$ that converges to a local minimizer of ϕ. The idea is to consider for the current iterate $p_k \in \mathcal{M}$ a *model* m_{p_k} of ϕ to locally approximate ϕ around p_k. The model should be (i) a "good" approximation of ϕ and (ii) simpler to use in an optimization algorithm than ϕ [12].

First observe that – given a retraction $R \colon T\mathcal{M} \to \mathcal{M}$ and an iterate $p_k \in \mathcal{M}$ – we can always locally express the minimization problem for ϕ on $\mathcal{M}$ as one on $T_{p_k}\mathcal{M}$ by considering the pullback $\hat{\phi}_{p_k} \coloneqq \phi \circ R_{p_k} \colon T_{p_k}\mathcal{M} \to \mathbb{R}$. Thus, instead of locally approximating ϕ around p_k, we may also consider a model $\hat{m}_{p_k} \colon T_{p_k}\mathcal{M} \to \mathbb{R}$ to locally approximate $\hat{\phi}_{p_k}$. Let us make that precise:

Definition 1 (Quadratic models on Riemannian manifolds)**.** Given $\phi \colon \mathcal{M} \to \mathbb{R}$ and $p \in \mathcal{M}$, a *quadratic model* is a map

$$\hat{m}_p \colon T_p\mathcal{M} \to \mathbb{R},\ \eta \mapsto \hat{m}_p(\eta), \tag{10}$$

with $\hat{m}_p(\eta) \coloneqq \phi(p) + \langle \operatorname{grad}\phi(p), \eta\rangle_p + \frac{1}{2}\langle \operatorname{Hess}\phi(p)[\eta], \eta\rangle_p$.

Remark 2. The map $\hat{m}_p$ approximates the pullback map $\hat{\phi}_p$ of ϕ in a neighborhood of the zero element of $T_p\mathcal{M}$. That is, we propose $\hat{m}_p(\eta)$ as an approximation for $\phi\,(R_p(\eta))$.

Algorithm 1 Trust-Region Algorithm on a Riemannian Mfd.

Input: Initial iterate $p_0 \in \mathcal{M}$, threshold $\rho' \in [0, 1/4)$.

Set $k = 0$
repeat
  Calculate $\eta_k \in T_{p_k}\mathcal{M}$ by solving (11)
  using the tCG strategy in [12, Algorithm 11]
  Evaluate ρ_k using (12)
  $\Delta_{k+1} = \text{update\_radius}(\eta_k, \rho_k)$ [12, Section 7.2.2]
  if $\rho_k > \rho'$ **then** ▷ model was accurate enough
    $p_{k+1} = R_{p_k}(\eta_k)$
  else
    $p_{k+1} = p_k$
  end if
  $k = k + 1$
until convergence;

Definition 1 allows the introduction of

$$\underset{\eta \in T_{p_k}\mathcal{M}}{\text{minimize}} \ \hat{m}_{p_k}(\eta), \quad \text{subject to } \langle \eta, \eta \rangle_{p_k} \leq \Delta_k^2 \tag{11}$$

as the *trust-region subproblem*. Here, Δ_k denotes the *trust-region radius* of the current iteration.

Subproblem (11) is to be solved at every iteration $k \in \{0, 1, 2, \dots\}$. Note that the trust-region problem is solved in the linear space $T_{p_k}\mathcal{M}$, which can be turned into an Euclidean space by virtue of the Riemannian metric $\langle \cdot, \cdot \rangle_{p_k}$ on $T_{p_k}\mathcal{M}$. Given the trust-region subproblem (11), which uses the quadratic model (10), the general iterative procedure to minimize ϕ is intuitive:

(i) Provide an initial guess $p_0 \in \mathcal{M}$.
(ii) At iteration $k \in \{0, 1, 2, \dots\}$, compute an update vector $\eta_k \in T_{p_k}\mathcal{M}$ by (approximately) solving (11).
(iii) A candidate for the next iterate is $p_{k+1} = R_{p_k}(\eta_k)$.

Remark 3. Note that, by virtue of the retraction, the algorithm inherently ensures $p_k \in \mathcal{M}$ for all $k \in \{0, 1, 2, \dots\}$.

As stated in step (ii), subproblem (11) has to be solved in each iteration. For this purpose, we apply an adapted truncated conjugate-gradient (tCG) strategy, cf. [12, Algorithm 11]. Another important step of the trust-region algorithm is to update the radius Δ_k. To implement this update, we use the quality factor

$$\rho_k \coloneqq \frac{\phi(p_k) - \phi\left(R_{p_k}(\eta_k)\right)}{\hat{m}_{p_k}\left(0_{p_k}\right) - \hat{m}_{p_k}\left(\eta_k\right)}, \tag{12}$$

where 0_{p_k} is the zero element of $T_{p_k}\mathcal{M}$. This factor assesses how well the model approximates the function over the step η_k [12]. Intuitively, ρ_k relates the *actual* cost improvement to the *predicted* cost improvement, i.e., $\rho_k = 1$ implies a perfect fit between objective function and model over the step. Therefore, the step is accepted if ρ_k is larger than a user-defined threshold ρ', and rejected otherwise. Algorithm 1 summarizes the trust-region optimization minimizing an objective ϕ on a Riemannian manifold $\mathcal{M}$.

It can be shown that Algorithm 1 converges to a stationary point of ϕ – which is very likely a minimizer – under mild assumptions [12]. In the next section, we discuss what the optimization algorithm looks like specifically for the Grassmann manifold $\mathcal{M} = \mathcal{GR}(p, d)$.

Trust-region algorithm on the Grassmann Manifold: This section introduces the necessary objects on the Grassmann manifold that are required for the implementation of Algorithm 1, c.f. Table II. For this, we rely on the description of [19]. To support our discussion presented in this section, please refer to Fig. 1. It illustrates the two manifolds and their tangent spaces, and connects the concepts in form of a commutative diagram. The key idea is to express the objective $\phi\colon \mathcal{GR}(p, d) \to \mathbb{R}$ and its Riemannian gradient and Hessian, as well as the retraction in terms of Stiefel representatives. In other words, we calculate the iterations $P_0, P_1, \dots$ not by using the tangent spaces to the Grassmann manifold, but by using the ones to the Stiefel manifold.

We start with a subspace $\mathcal{U} \cong [U]$ identified with the projector $P = \pi(U) \in \mathcal{GR}(p, d)$. The respective tangent spaces are given by $T_P\mathcal{GR}(p, d) = \{\xi \in \mathbb{R}^{d \times d} \mid \xi^\top = \xi,\ \xi = P\xi + \xi P\}$, $T_U\mathcal{ST}(p, d) = \{D \in \mathbb{R}^{d \times p} \mid U^\top D = -D^\top U\}$. The tangent space $T_U\mathcal{ST}(p, d)$ can be decomposed (with respect to the projection map π and the Riemannian metric $\langle \cdot, \cdot \rangle_U$ on $\mathcal{ST}(p, d)$)[2] into a horizontal and a vertical component, i.e., $T_U\mathcal{ST}(p, d) = \operatorname{Ver}_U\mathcal{ST}(p, d) \oplus \operatorname{Hor}_U\mathcal{ST}(p, d)$, where $\operatorname{Hor}_U\mathcal{ST}(p, d) = \{D \in \mathbb{R}^{d \times p} \mid U^\top D = 0\}$. Any tangent vector $T_P\mathcal{GR}(p, d)$ can be uniquely identified with an element of the horizontal space $\operatorname{Hor}_U\mathcal{ST}(p, d)$. This allows to work on the level of Stiefel representatives when considering tangent vectors in $T_P\mathcal{GR}(p, d)$.

We introduce the concept of horizontal lifts formally:

Definition 2 (Horizontal lift [19])**.** Let $P = \pi(U) \in \mathcal{GR}(p, d)$ and $\zeta \in T_P\mathcal{GR}(p, d)$. We call the *unique* element $\zeta_U^{\text{hor}} = \zeta U \in \operatorname{Hor}_U\mathcal{ST}(p, d)$ the *horizontal lift of* ζ *to* U.

Remark 4. Any $W \in \mathbb{R}^{d \times p}$ can be projected onto an element of the horizontal space $\operatorname{Hor}_U\mathcal{ST}(p, d)$ by virtue of

$$\Pi_U^{\perp}\colon \mathbb{R}^{d \times p} \to \operatorname{Hor}_U\mathcal{ST}(p, d),\ W \mapsto \left(I_d - UU^\top\right) W. \tag{13}$$

We are now fully prepared to present all required objects, c.f. Table II. On Riemannian manifolds, there is a specific candidate for a retraction R_P, namely the *exponential map*. On the Grassmann manifold, we can state the exponential map in terms of the horizontal lifts. It is given by $\operatorname{Exp}_U\colon \operatorname{Hor}_U\mathcal{ST}(p, d) \to \mathcal{ST}(p, d), \xi_U^{\text{hor}} \mapsto UV\cos(\Sigma)V^\top + Q\sin(\Sigma)V^\top$, where $\xi_U^{\text{hor}} = Q\Sigma V^\top$ is a compact SVD decomposition and $\cos(\cdot)$, $\sin(\cdot)$ is evaluated element-wise on the main diagonal. Thus, for $\eta \in T_P\mathcal{GR}(p, d)$, we use the exponential map as the retraction on $\mathcal{GR}(p, d)$, i.e., $R_P(\eta) = \pi\left(\operatorname{Exp}_U\left(\eta_U^{\text{hor}}\right)\right)$.

Besides, the Riemannian gradient of ϕ at $P = \pi(U)$, horizontally lifted to U, is given by the projection (13) of the Euclidean gradient of the lifted objective function $\bar{\phi} = \phi \circ \pi$ onto the horizontal space to the Stiefel manifold. We obtain $\left(\operatorname{grad}\phi(P)\right)_U^{\text{hor}} = \Pi_U^{\perp}\left(\operatorname{grad}^{\text{eucl}}\bar{\phi}(U)\right)$ for the horizontal lift of $\operatorname{grad}\phi(P)$ to U. Furthermore, we introduce $h\colon \mathbb{R}^{d \times p} \to \mathbb{R}^{d \times p},\ W \mapsto (I_d - WW^\top)\operatorname{grad}^{\text{eucl}}\bar{\phi}(W)$ as

[2] We skip the details for brevity, see [19, Section 2]

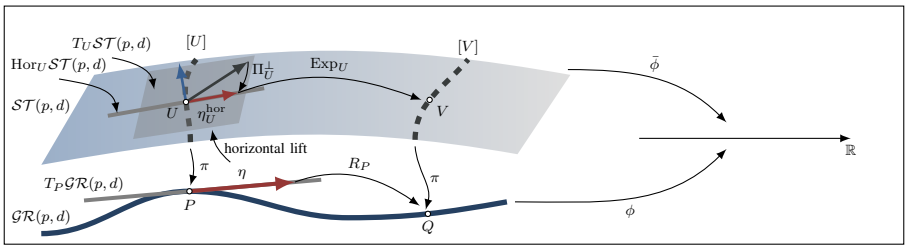


Fig. 1: Commutative diagram for the bundle of Grassmann and Stiefel manifold and their tangent spaces.

TABLE II: Overview of the objects on $\mathcal{GR}(p,d)$ used in Algorithm 1.

Name	Object required by Algorithm 1	Calculation via
Retraction	$R_P\colon T_P\mathcal{GR}(p,d) \to \mathcal{GR}(p,d)$	$R_P(\eta) \coloneqq \pi\left(\mathrm{Exp}_U\left(\eta_U^{\mathrm{hor}}\right)\right)$
Riemannian gradient	$\mathrm{grad}\,\phi\colon \mathcal{GR}(p,d) \to T_P\mathcal{GR}(p,d)$	$(\mathrm{grad}\,\phi(P))_U^{\mathrm{hor}} \coloneqq \Pi_U^{\perp}\left(\mathrm{grad}^{\mathrm{eucl}}\,\bar{\phi}(U)\right)$
Riemannian Hessian	$\mathrm{Hess}\,\phi(P)\colon T_P\mathcal{GR}(p,d) \to T_P\mathcal{GR}(p,d)$	$(\mathrm{Hess}\,\phi(P)[\eta])_U^{\mathrm{hor}} \coloneqq H_U^{\mathrm{hor}}[\eta_U^{\mathrm{hor}}]$
Riemannian metric	$\langle\cdot,\cdot\rangle_P\colon T_P\mathcal{GR}(p,d) \times T_P\mathcal{GR}(p,d) \to \mathbb{R}$	$\langle\xi,\eta\rangle_P \coloneqq \mathrm{tr}\left(\left(\xi_U^{\mathrm{hor}}\right)^{\top}\eta_U^{\mathrm{hor}}\right)$

an extension to $\mathbb{R}^{d\times p}$ of the horizontally lifted gradient. This enables the introduction of

$$H_U^{\mathrm{hor}}\colon \mathrm{Hor}_U\mathcal{ST}(p,d) \to \mathrm{Hor}_U\mathcal{ST}(p,d),$$
$$\eta_U^{\mathrm{hor}} \mapsto H_U^{\mathrm{hor}}[\eta_U^{\mathrm{hor}}] \coloneqq \Pi_U^{\perp}\left(\frac{\mathrm{d}}{\mathrm{d}\,t}\bigg|_{t=0} h\left(U + t\,\eta_U^{\mathrm{hor}}\right)\right),$$

where $U + t\,\eta_U^{\mathrm{hor}}$ is to be understood as an operation on the vector space $\mathbb{R}^{d\times p}$. This map is used to calculate the horizontal lift $(\mathrm{Hess}\,\phi(P)[\eta])_U^{\mathrm{hor}} = H_U^{\mathrm{hor}}[\eta_U^{\mathrm{hor}}]$ of the tangent vector $\mathrm{Hess}\,\phi(P)[\eta]$ to U, for $\eta \in T_P\mathcal{GR}(p,d)$.

The Riemannian metric can also be expressed in terms of the horizontal lifts

$$\langle\xi,\eta\rangle_P = \mathrm{tr}\left(\left(\xi_U^{\mathrm{hor}}\right)^{\top}\eta_U^{\mathrm{hor}}\right). \tag{14}$$

Since all required objects are expressed in terms of the horizontal lifts, we must also state the quadratic model (10) on this level. Thus, given an objective function $\phi\colon \mathcal{GR}(p,d) \to \mathbb{R}$ and $P = \pi(U) \in \mathcal{GR}(p,d)$, we state the quadratic model $\hat{m}_U\colon \mathrm{Hor}_U\mathcal{ST}(p,d) \to \mathbb{R}$ on $\mathrm{Hor}_U\mathcal{ST}(p,d)$ as $\hat{m}_U\left(\eta_U^{\mathrm{hor}}\right) \coloneqq \bar{\phi}(U) + \langle\xi_U^{\mathrm{hor}},\eta_U^{\mathrm{hor}}\rangle_U + \frac{1}{2}\langle H_U^{\mathrm{hor}}[\eta_U^{\mathrm{hor}}],\eta_U^{\mathrm{hor}}\rangle_U$, with $\bar{\phi} = \phi\circ\pi$, $\xi = \mathrm{grad}\,\phi(P) \in T_P\mathcal{GR}(p,d)$ and where $\langle\cdot,\cdot\rangle_U\colon \mathrm{Hor}_U\mathcal{ST}(p,d) \times \mathrm{Hor}_U\mathcal{ST}(p,d) \to \mathbb{R}, (\xi,\eta) \mapsto \mathrm{tr}\left(\xi^{\top}\eta\right)$ derives from the Riemannian metric (14). Note that with this model, we hand an $U_0 \in \mathcal{ST}(p,d)$ as an initial guess to Algorithm 1, and obtain an $U^\star \in \mathcal{ST}(p,d)$ as the optimizer, which represents the subspace $\mathcal{U}^\star \cong [U^\star] \in \mathcal{GR}(p,d)$. Next, we present a specific algorithm to tackle Problem 1.

Algorithm to identify an admissible optimal subspace: In order to calculate an admissible and optimal subspace $\mathcal{U}^\star$, we first minimize the penalty function ψ to establish an initial subspace $\mathcal{U}_0 \cong [U_0]$ with $\pi(U_0) \in \mathcal{A}_{p,d}$.[3] Afterwards, we focus on ϕ_μ, aiming to improve the performance by minimizing ϕ_μ. As usual in optimization with logarithmic barrier functions, we solve this minimization problem repeatedly with decreasing value μ (i.e., $\mu \to 0$). The procedure is described in Algorithm 2, while the following Corollary summarizes the achieved result.

[3]If that fails, a potential remedy is to adapt the weights λ_r^i (c.f. (8)) of the constraints that were violated and call Algorithm 2 again.

Corollary 1. *Assume that $\mathcal{U}^\star \cong [U^\star] \in \mathcal{GR}(p,d)$ is returned by Algorithm 2, i.e., it is able to find $U_0 \in \mathcal{A}_{p,d}$ in step 2. Then, $\mathcal{U}^\star$ satisfies the conditions in Problem 1.*

Proof. The statement follows from ψ and ϕ_μ definitions. □

Remark 5. The states x_r contained in $\mathcal{C}$ affect the convergence of Algorithm 2. Additionally, the outcome of both Algorithm 1 and Algorithm 2 depend on the initial values p_0 and U_{init}, respectively. These choices are subject of future work.

Algorithm 2 Admissible optimal subspace (AOS)

1: Set admisisble = false
2: Call Algorithm 1 for ψ with initial guess U_{init} and obtain $\mathcal{U}_0 \cong [U_0]$
3: **if** $\pi(U_0) \in \mathcal{A}_{p,d}$ **then**
4: Set admissible = true; $k = 1$; $\varepsilon^\star = \infty$
5: **repeat**
6: Call Algorithm 1 for ϕ_{μ_k} with initial guess U_{k-1} and obtain $\mathcal{U}_k \cong [U_k]$
7: **if** $\varepsilon\left(\pi(U_k)\right) < \varepsilon^\star$ **then** ▷ at least true for $k = 1$
8: $\varepsilon^\star = \varepsilon\left(\pi(U_k)\right)$; $U^\star = U_k$
9: **end if**
10: $\mu_{k+1} = \mathrm{decrease}(\mu_k)$; $k = k + 1$
11: **until** convergence; ▷ No significant changes in $\varepsilon^\star$ over the last iterations
12: **end if**
13: **if** admissible **then return** $\mathcal{U}^\star \cong [U^\star]$
14: **elsereturn** ERROR: Establishing admissibility failed!
15: **end if**

V. Example

We consider the simplified model $(\dot{x}^1,\dot{x}^2) = (x^2,\sin(x^1) + u)$ of an inverted pendulum, where x^1 is the angular position, x^2 the angular velocity and u the applied torque. This model is linearized around the upright position $x_{\mathrm{s}} = (0,0)$ and discretized with a sampling time $T_{\mathrm{s}} = 0.1\,\mathrm{s}$ to obtain the linear, discrete-time model. The states are constrained by $\mathbb{X} = \{x \in \mathbb{R}^2 \mid |x^1| \le 1, |x^2| \le 2\}$ and the control input by $\mathbb{U} = \{u \in \mathbb{R} \mid |u| \le 1\}$.

TABLE III: Results of Algorithm 2.

N	$\varepsilon(P^\star)$	$(\varepsilon(P_0)-\varepsilon(P^\star))/\varepsilon(P_0)$
20	1.8%	86.75%
25	2.64%	87.49%
30	3.22%	83.03%
35	3.91%	77.63%
40	5.02%	73.01%

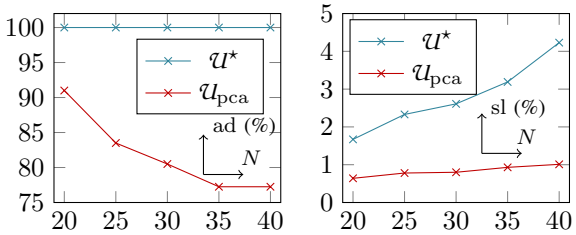
The MPC controller is designed with a stage cost $\ell(x,u) = x^\top \operatorname{diag}(10,1)x + 0.1u^2$. The terminal cost and terminal set are based on an unconstrained LQR controller. YALMIP and the MPT3 toolbox are used to set up (1).

The control task is to stabilize the pendulum in the upright position after it has been disturbed from it by an external force. We model that by considering initial values from the set $\mathbb{X}_0 := \{x \in \mathbb{R}^2 \mid 0.3 \leq x^{1,2} \leq 0.4, \}$, which is close to the boundary of $\mathcal{X}_{20}$. This underlines the necessity to guarantee admissibility during the subspace *design*, since we do not know the initial state exactly, but only its distribution. We draw $L = 100$ uniformly distributed samples x_r from $\mathbb{X}_0$, and build the data set $\mathcal{C}$ for different prediction horizons N by adding the corresponding optimal input sequences $\mu_N(x_r)$. The original dimension is $d = N$ and we always reduce to $p = 2$, i.e., we reduce by a factor of 10 for $N = 20$ and by a factor of 20 for $N = 40$. We call Algorithm 2 to obtain the optimized subspace $P^\star \cong \mathcal{U}^\star$. As the initial guess, we select $\mathcal{U}_{\text{pca}}$ stemming from a principal component analysis, i.e., the subspace spanned by the p top left singular vectors of the matrix containing $\mu_N(x_r)$ as columns. P_0 denotes the initially admissible subspace returned by the penalty optimization, c.f. line 2 in Algorithm 2.

We report the results of Algorithm 2 in Table III. It can be seen in the second column that the projection of the optimal sequences onto $\mathcal{U}^\star$ leads to small relative errors in the cost on average. Also, as visible in the last column, the performance optimization in Algorithm 2 in lines 3 – 11 is able to greatly improve on P_0. Additionally, we investigate how well $\mathcal{U}^\star$ and $\mathcal{U}_{\text{pca}}$ generalize in $\mathbb{X}_0$, after being constructed by only considering $L = 100$ states. To this end, we sample states in a uniform grid from $\mathbb{X}_0$ and study the outcomes for the considered prediction horizons in Fig. 2. We report the admissibility ratio ad as the percentage of states over the grid for which (1) *can* be solved in the respective subspace. We characterize the level of suboptimality sl as the mean of the relative error in the optimal value that we make when solving (1) in the subspace. While $\mathcal{U}_{\text{pca}}$ slightly outperforms $\mathcal{U}^\star$ in terms of suboptimality, $\mathcal{U}^\star$ has the desired advantage that it provides admissible input sequences for *all* states in $\mathbb{X}_0$, while (1) can only be solved in $\mathcal{U}_{\text{pca}}$ for 77% of the states in $\mathbb{X}_0$ for $N = 40$.

VI. Conclusion

We introduce an approach on the Grassmann manifold $\mathcal{GR}(p,d)$ to design a p-dimensional linear subspace $\mathcal{U}$ of $\mathbb{R}^d$. This subspace ensures that the projections of optimal control sequences in a data set $\mathcal{C}$ onto $\mathcal{U}$ are admissible. Additionally, the relative error in the cost function caused by the projection is minimized for the considered data set. While the presented results are only preliminary, we do believe that they are promising. Next steps are the application of the approach to a higher-dimension to study its scalability and computational complexity, the addition of formal guarantees and a refinement of the data collection process. Future work should discuss how the presented approach compares to similar ideas as in [10], [5], [11].



(a) Admissibility (b) Suboptimality

Fig. 2: Results of the reduced MPC problem over $\mathbb{X}_0$.

References

[1] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: theory, computation, and design*, vol. 2. Nob Hill Publishing, 2017.

[2] R. Findeisen and F. Allgöwer, "An Introduction to Nonlinear Model Predictive Control," in *21st Benelux Meet. Syst. Cont.*, pp. 119–141, 2002.

[3] R. Cagienard, P. Grieder, E. Kerrigan, and M. Morari, "Move blocking strategies in receding horizon control," *J. Proc. Cont.*, vol. 17, no. 6, pp. 563–570, 2007.

[4] R. C. Shekhar and C. Manzie, "Optimal move blocking strategies for model predictive control," *Automatica*, vol. 61, pp. 27–34, 2015.

[5] A. Bemporad and G. Cimini, "Reduction of the number of variables in parametric constrained least-squares problems," *arXiv:2012.10423*, 2020.

[6] J. Unger, M. Kozek, and S. Jakubek, "Reduced order optimization for model predictive control using principal control moves," *J. Proc. Cont.*, vol. 22, no. 1, pp. 272–279, 2012.

[7] O. J. Rojas, G. C. Goodwin, M. M. Serón, and A. Feuer, "An SVD based strategy for receding horizon control of input constrained linear systems," *Int. J. Rob. Cont.*, vol. 14, no. 13-14, pp. 1207–1226, 2004.

[8] L. Wang, "Discrete model predictive controller design using laguerre functions," *J. Proc. Cont.*, vol. 14, no. 2, pp. 131–142, 2004.

[9] M. Ławryńczuk, "Nonlinear model predictive control for processes with complex dynamics: A parameterisation approach using laguerre functions," *Int. J. Appl. Math. Comp. Scie.*, vol. 30, no. 1, pp. 35–46, 2020.

[10] G. Pan and T. Faulwasser, "NMPC in active subspaces: Dimensionality reduction with recursive feasibility guarantees," *Automatica*, vol. 147, p. 110708, 2023.

[11] G. Goebel and F. Allgöwer, "State dependent parametrizations for nonlinear MPC," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 1005–1010, 2014. 19th IFAC World Congress.

[12] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ: Princeton University Press, 2008.

[13] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal.& Appl.*, vol. 20, no. 2, pp. 303–353, 1998.

[14] N. Boumal, *Optimization and estimation on manifolds*. PhD thesis, Université catholique de Louvain, 2014.

[15] W. Boothby, *An Introduction to Differentiable Manifolds and Riemannian Geometry*. Academic Press, 1975.

[16] C. J. Isham, *Modern Differential Geometry for Physicists*. World Scientific, 2nd ed., 1999.

[17] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[18] P.-A. Absil, R. E. Mahony, and R. Sepulchre, "Riemannian geometry of Grassmann manifolds with a view on algorithmic computation," *Acta Applicandae Mathematica*, vol. 80, pp. 199–220, 2004.

[19] T. Bendokat, R. Zimmermann, and P.-A. Absil, "A Grassmann manifold handbook: Basic geometry and computational aspects," *arXiv:2011.13699*, 2020.