Appendix

A Justification for Sparsity Assumption for $\hat{\mathcal{B}}$

Physically, structural assumption (11) is not as mysterious as it appears. Indeed, consider the standard dynamics form for mechanical systems:

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = B(q)u,$$

where $q \in \mathbb{R}^{n_q}$ is the configuration vector, $H \in \mathbb{S}_{n_q}^{\geq 0}$ is the inertia matrix, $C(q, \dot{q})\dot{q}$ contains the centrifugal and Coriolis terms, $g$ are the gravitational terms, $B \in \mathbb{R}^{n_u \times m}$ is the (full rank) input matrix mapping and $u \in \mathbb{R}^m$ is the input. For fully actuated systems, $\text{rank}(B) = m = n_q$. For underactuated systems, $m < n_q$. By rearranging the configuration vector [24, 28, 34], one can partition $q$ as $(q_u, q_a)$ where $q_u \in \mathbb{R}^{n_q-m}$ represents the unactuated degrees of freedom and $q_a \in \mathbb{R}^m$ represents the actuated degrees of freedom. Applying this partitioning to the dynamics equation above yields

$$\begin{bmatrix}
H_{uu}(q) & H_{ua}(q) \\
H_{au}(q) & H_{aa}(q)
\end{bmatrix}
\begin{bmatrix}
\dot{q}_u \\
\dot{q}_a
\end{bmatrix}
+ \begin{bmatrix}
\tau_u(q, \dot{q}) \\
\tau_a(q, \dot{q})
\end{bmatrix}
= \begin{bmatrix}
O_{(n_q-m) \times m} \\
b(q)
\end{bmatrix}
u,$$

where $b \in \mathbb{R}^{m \times m}$ is an invertible square matrix. As observed in [28], a substantial class of underactuated systems can be represented in this manner. Of course, fully actuated systems also take this form $(m = n_q)$. Thus, by taking as state $x = (q, p) \in \mathbb{R}^n$ where $p = H(q)\dot{q}$ is momentum (so that $n = 2n_q$), the dynamics can be written as (5):

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} H^{-1}(q)p \\ \dot{H}(q)\dot{q} - \tau(q, \dot{q}) \end{bmatrix} + \begin{bmatrix} O_{(n-m) \times m} \\ b(q) \end{bmatrix}u. \tag{20}$$

Notice that the input matrix takes the desired normal form in (11). To address the apparent difficult of working with the state representation of $(q, p)$ (when usually only measurements of $(q, \dot{q}, \ddot{q})$ are typically available and the inertia matrix $H(q)$ is unknown), we make use of the following result from [17]:

**Theorem 1 (CCM Invariance to Diffeomorphisms).** Suppose there exists a valid CCM $M_x(x)$ with respect to the state $x$. Then, if $z = \psi(x)$ is a diffeomorphism, then, the CCM conditions also hold with respect to state $z$ with metric $M_z(z) = \Psi(z)^{-1}M_x(x)\Psi(z)^{-1}$, where $\Psi(z) = \partial\psi(x)/\partial x$ evaluated at $x = \psi^{-1}(z)$.

Thus, for the (substantial) class of underactuated systems of the form (20), one would solve problem (10) by alternating between fitting $H, \tau, b$ (using the demonstrations $(q, \dot{q}, \ddot{q})$) and leveraging Theorem 1 and the previous estimate of $H$ to enforce the matrix inequality constraints using the state representation $(q, p)$. This allows us to borrow several existing results from adaptive control on estimating mechanical system by leveraging the known linearity of $H(q)$ in terms of unknown mass property parameters multiplying known physical basis functions. We leave this extension however, to future work.

B Derivation of Problem (16)

To go from the general problem definition in (10) to the finite dimensional problem in (16), we first must define appropriate function classes for $f, b_j$, and $W$. We will do this using the framework of RKHS.
B.1 Reproducing Kernel Hilbert Spaces

Scalar-valued RKHS: Kernel methods \cite{30} constitute a broad family of non-parametric modeling techniques for solving a range of problems in machine learning. A scalar-valued positive definite kernel function \( k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R} \) generates a Reproducing Kernel Hilbert Space (RKHS) of functions, with the nice property that if two functions are close in the distance derived from the norm (associated with the Hilbert space), then their pointwise evaluations are close at all points. This continuity of evaluation functionals has the far reaching consequence that norm-regularized learning problems over RKHSs admit finite dimensional solutions via Representer Theorems. The kernel \( k \) may be (non-uniquely) associated with a higher-dimensional embedding of the input space via a feature map, \( \phi : \mathbb{R}^n \mapsto \mathbb{R}^D \), such that \( k(x,z) = \langle \phi(x), \phi(z) \rangle \), where \( D \) is infinite for universal kernels associated with RKHSs that are dense in the space of square integrable functions. Standard regularized linear statistical models in the embedding space implicitly provide non-linear inference with respect to the original input representation. In a nutshell, kernel methods provide a rigorous algorithmic framework for deriving non-linear counterparts of a whole array of linear statistical techniques, e.g. for classification, regression, dimensionality reduction and unsupervised learning. For details, we point the reader to \cite{7}.

Vector-valued RKHS: Dynamics estimation is a vector-valued learning problem. Such problems can be naturally formulated in terms of vector-valued generalizations of RKHS concepts. The theory and formalism of vector-valued RKHS can be traced as far back as the work of Laurent Schwarz in 1964, with applications ranging from solving partial differential equations to machine learning. Informally, we say that that \( \mathcal{H} \) is an RKHS of \( \mathbb{R}^n \)-valued maps if for any \( v \in \mathbb{R}^n \), the linear functional that maps \( f \in \mathcal{H} \) to \( v^T f(x) \) is continuous.

More formally, denote the standard inner product on \( \mathbb{R}^n \) as \( \langle \cdot, \cdot \rangle \) and let \( \mathbb{R}^n(\mathcal{X}) \) be the vector space of all functions \( f : \mathcal{X} \to \mathbb{R}^n \) and let \( \mathcal{L}(\mathbb{R}^n) \) be the space of all bounded linear operators on \( \mathbb{R}^n \), i.e., \( n \times n \) matrices. A function \( K : \mathcal{X} \times \mathcal{X} \to \mathcal{L}(\mathbb{R}^n) \) is an operator-valued positive definite kernel if for all \( (x,z) \in \mathcal{X} \times \mathcal{X} \), \( K(x,z)^T = K(z,x) \) and for all finite set of points \( \{x_i\}_{i=1}^N \in \mathcal{X} \) and \( \{y_i\}_{i=1}^N \in \mathbb{R}^n \),

\[
\sum_{i,j=1}^N (y_i, K(x_i, x_j) y_j) \geq 0.
\]

Given such a \( K \), we define the unique \( \mathbb{R}^n \)-valued RKHS \( \mathcal{H}_K \subset \mathbb{R}^n(\mathcal{X}) \) with reproducing kernel \( K \) as follows. For each \( x \in \mathcal{X} \) and \( y \in \mathbb{R}^n \), define the function \( K_x y = K(\cdot, x) y \in \mathbb{R}^n(\mathcal{X}) \). That is, for all \( z \in \mathcal{X} \), \( K_x y(z) = K(z,x) y \). Then, the Hilbert space \( \mathcal{H}_K \) is defined to be the completion of the linear span of functions \( \{K_x y \mid x \in \mathcal{X}, y \in \mathbb{R}^n\} \) with inner product between functions \( f = \sum_{i=1}^N K_{x_i} y_i \), \( g = \sum_{j=1}^M K_{z_j} w_j \in \mathcal{H}_K \), defined as:

\[
\langle f,g \rangle_{\mathcal{H}_K} = \sum_{i=1}^N \sum_{j=1}^M \langle y_i, K(x_i, z_j) w_j \rangle.
\]

Notably, the kernel \( K \) satisfies the following reproducing property:

\[
\langle f(x), y \rangle = \langle f, K_x y \rangle_{\mathcal{H}_K} \quad \forall (x,y) \in \mathcal{X} \times \mathbb{R}^n \land f \in \mathcal{H}_K,
\]

(21)
and is thereby referred to as the *reproducing kernel* for $\mathcal{H}_K$. As our learning problems involves the Jacobian of $f$, we will also require a representation for the derivative of the kernel matrix. Accordingly, suppose the kernel matrix $K$ lies in $C^2(\mathcal{X} \times \mathcal{X})$. For any $j \in \{1, \ldots, n\}$, define the matrix functions $\frac{\partial}{\partial s^j} K : \mathcal{X} \times \mathcal{X} \to \mathcal{L}(\mathbb{R}^n)$ and $\frac{\partial}{\partial r^j} K : \mathcal{X} \times \mathcal{X} \to \mathcal{L}(\mathbb{R}^n)$ as

$$\left. \frac{\partial}{\partial s^j} K(z,x) := \frac{\partial}{\partial s^j} K(r,s) \right|_{r=z, s=x}, \quad \left. \frac{\partial}{\partial r^j} K(z,x) := \frac{\partial}{\partial r^j} K(r,s) \right|_{r=z, s=x},$$

where the derivative of the kernel matrix is understood to be element-wise. Define the $C^1$ function $\partial_j K x y$ on $\mathbb{R}^n(\mathcal{X})$ as

$$\partial_j K x y(z) := \frac{\partial}{\partial s^j} K(z,x) y \quad \forall z \in \mathcal{X}.$$ 

The following result provides a useful reproducing property for the derivatives of functions in $\mathcal{H}_K$ that will be instrumental in deriving the solution algorithm.

**Theorem 2 (Derivative Properties on $\mathcal{H}_K$ [21])**. Let $\mathcal{H}_K$ be a RKHS in $\mathbb{R}^n(\mathcal{X})$ with reproducing kernel $K \in C^2(\mathcal{X} \times \mathcal{X})$. Then, for all $(x, y) \in \mathcal{X} \times \mathbb{R}^n$:

(a) $\partial_j K x y \in \mathcal{H}_K$ for all $j = 1, \ldots, n$.

(b) The following derivative reproducing property holds for all $j = 1, \ldots, n$:

$$\langle \frac{\partial f(x)}{\partial x^j}, y \rangle = \langle f, \partial_j K x y \rangle_{\mathcal{H}_K}, \quad \forall f \in \mathcal{H}_K.$$ 

As mentioned in Section 5, the bi-linearity of constraint (9) forces us to adopt an alternating solution strategy whereby in the “dynamics” sub-problem, $\{W, w, \overline{w}\}$ are held fixed and we minimize $J_d$ with respect to $\{f, B, \lambda\}$. In the “metric” sub-problem, $\{f, B, \lambda\}$ are held fixed and we minimize $J_m$ with respect to $\{W, w, \overline{w}\}$.

In the following we derive several useful representer theorems to characterize the solution of the two sub-problems, under the two simplifying assumptions introduced in Section 4.2.

**B.2 Dynamics Sub-Problem**

Let $K^f$ be the reproducing $C^2$ kernel for an $\mathbb{R}^n$-valued RKHS $\mathcal{H}^f_K$ and let $K^B$ be another reproducing kernel for an $\mathbb{R}^n$-valued RKHS $\mathcal{H}^B_K$. Define the finite-dimensional subspaces:

$$\mathcal{V}_f := \left\{ \sum_{i=1}^{N_u} K^f_{x_i} a_i + \sum_{i=1}^{N_x} \sum_{p=1}^n \partial_p K^f_{x_i} a'_{ip}, \quad a_i, a'_{ip} \in \mathbb{R}^n \right\} \subset \mathcal{H}^f_K. \quad (22)$$

$$\mathcal{V}_B := \left\{ \sum_{i=1}^{N_u} K^B_{x_i} c_i + \sum_{i=1}^{N_x} \sum_{p=1}^n \partial_p K^B_{x_i} c'_{ip}, \quad c_i, c'_{ip} \in \mathbb{R}^n \right\} \subset \mathcal{H}^B_K. \quad (23)$$

Note that all $x_i$ taken from the training dataset of $(x_i, u_i, \dot{x}_i)$ tuples are a subset of $X_c$. 

Theorem 3 (Representer Theorem for $f, B$). Suppose the reproducing kernel $K_B$ is chosen such that all functions $g \in H_B^K$ satisfy the sparsity structure $g_j(x) = 0$ for $j = 1, \ldots, n - m$. Consider then the pointwise-relaxed dynamics sub-problem for problem (10):

$$\begin{align*}
\min_{f \in H_f^K, \ b_j \in H_B^K, j = 1, \ldots, m} \ J_d(\hat{f}, \hat{B}) \\
\text{s.t.} \quad F(x_i; \hat{f}, W, \lambda) \preceq 0, \quad \forall x_i \in X_c. \quad (24)
\end{align*}$$

Suppose the feasible set for the LMI constraint is non-empty. Denote $f^*$ and $b_j^*$ as the optimizers for this sub-problem. Then, $f^* \in V_f$ and $b_j^* \in V_B$ for all $j = 1, \ldots, m$.

Proof. For a fixed $W$, the constraint is convex in $\hat{f}$ by linearity of the matrix $F$ in $\hat{f}$ and $\partial \hat{f} / \partial x$. By assumption (1), and the simplifying form for $B_\perp$, the matrix $F$ is additionally independent of $B$. Then, by strict convexity of the objective functional, there exist unique minimizers $f^*, b_j^* \in H_K$, provided the feasible region is non-empty [12].

Since $V_f$ and $V_B$ are closed, by the Projection theorem, $H_f^K = V_f \oplus V_f'$ and $H_B^K = V_B \oplus V_B'$. Thus, any $g \in H_K^K$ may be written as $g^f + g^\perp$ where $g^f \in V_f$, $g^\perp \in V_f'$, and $\langle g^f, g^\perp \rangle_{H_f^K} = 0$. Similarly, any $g \in H_B^K$ may be written as $g^B + g^\perp$ where $g^B \in V_B$, $g^\perp \in V_B'$, and $\langle g^B, g^\perp \rangle_{H_B^K} = 0$.

Let $f = f^f + f^\perp$ and $b_j = b_j^B + b_j^\perp$, and define

$$\begin{align*}
H^f(x, u) &= f^f(x) + \sum_{j=1}^{m} u^j b_j^B(x) \\
H^\perp(x, u) &= f^\perp(x) + \sum_{j=1}^{m} u^j b_j^\perp(x).
\end{align*}$$

We can re-write $J_d(f, B)$ as:

$$\begin{align*}
J_d(f, B) = \sum_{i=1}^{N} \langle H^f(x_i, u_i) - \dot{x}_i, H^f(x_i, u_i) - \dot{x}_i \rangle + 2 \langle H^f(x_i, u_i) - \dot{x}_i, H^\perp(x_i, u_i) \rangle \\
+ \langle H^\perp(x_i, u_i), H^\perp(x_i, u_i) \rangle + \mu_f \left( \|f^f\|^2_{H^K_f} + \|f^\perp\|^2_{H^K_\perp} \right) \\
+ \mu_b \left( \sum_{j=1}^{n} \|b_j^B\|^2_{H^K_B} + \|b_j^\perp\|^2_{H^K_\perp} \right).
\end{align*}$$
Now,
\[
\langle H^V(x_i, u_i) - \dot{x}_i, H^\perp(x_i, u_i) \rangle
\]
\[
= \langle f^\perp(x_i) + \sum_{j=1}^{m} u^j_i b^j_i(x_i), H^V(x_i, u_i) - \dot{x}_i \rangle
\]
\[
= \langle f^\perp, K^f_{x_i} (H^V(x_i, u_i) - \dot{x}_i) \rangle_{H^K} + \left\langle \sum_{j=1}^{m} u^j_i b^j_i, K^B_{x_i} (H^V(x_i, u_i) - \dot{x}_i) \right\rangle_{H^K}
\]

since \(K^f_{x_i} (H^V(x_i, u_i) - \dot{x}_i) \in V_f, \forall K^B_{x_i} \) is closed under addition, and \(K^B_{x_i} (H^V(x_i, u_i) - \dot{x}_i) \in V_B \). Thus, \(J_{d}(f, B) \) simplifies to
\[
\sum_{i=1}^{N} \left| \|H^V(x_i, u_i) - \dot{x}_i\|_2^2 + \mu_f \|f^V_i\|_{H^K}^2 + \mu_b \sum_{j=1}^{n} \|b_i^j\|_{H^K}^2 \right| + \left[ \sum_{i=1}^{N} \|H^\perp(x_i, u_i)\|_2^2 + \mu_f \|f^\perp\|_{H^K}^2 + \mu_b \sum_{j=1}^{n} \|b_i^j\|_{H^K}^2 \right].
\]

(25)

Now, the \((p, q)\) element of \(\partial f_W(x_i)\) takes the form
\[
\left\langle \frac{\partial w_{pq}(x_i)}{\partial x}, f(x_i) \right\rangle = \left\langle f, K^f_{x_i} \frac{\partial w_{pq}(x_i)}{\partial x} \right\rangle_{H^K} = \left\langle f^V_i, K^f_{x_i} \frac{\partial w_{pq}(x_i)}{\partial x} \right\rangle_{H^K}.
\]

Column p of \(\frac{\partial f(x_i)}{\partial x} W(x_i)\) takes the form
\[
\sum_{j=1}^{n} \lambda_j p(x_i) \frac{\partial f(x_i)}{\partial x} = \left[ \sum_{j=1}^{n} \lambda_j p(x_i) \left\langle \frac{\partial f(x_i)}{\partial x}, e_1 \right\rangle \right] = \left[ \sum_{j=1}^{n} \lambda_j p(x_i) \left\langle \frac{\partial f(x_i)}{\partial x}, e_n \right\rangle \right] = \left[ \sum_{j=1}^{n} \lambda_j p(x_i) \left\langle f^V_i, \partial j \right\rangle_{H^K} \right],
\]

where \(e_i\) is the \(i^{th}\) standard basis vector in \(\mathbb{R}^n\). Thus, \(f^\perp\) plays no role in pointwise relaxation of constraint (9) and thus does not affect problem feasibility. Given assumption 1, \(b^\perp\) also has no effect on problem feasibility. Thus, by non-negativity of the term in the square brackets in (25), we have that the optimal \(f\) lies in \(V_f\) and optimal \(b_j\) lies in \(V_B\).

The key consequence of this theorem is the reduction of the infinite-dimensional search problem for the functions \(f\) and \(b_j, j = 1, \ldots, m\) to a finite-dimensional convex optimization problem for the constant vectors \(a_i, a_i', \{c_j^{(j)}\}, \forall c_j^{(j)} \in \mathbb{R}, \) by choosing the function classes \(H^f = H^K\) and \(H^B = H^K\). Next, we characterize the optimal solution to the metric sub-problem.
B.3 Metric Sub-Problem

By the simplifying assumption in Section 4.1, constraint (8) requires that \( W \) be only a function of the first \((n - m)\) components of \( x \). Thus, define \( \kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \) as a scalar reproducing kernel with associated real-valued scalar RKHS \( \mathcal{H}_\kappa \). Additionally, define \( \hat{\kappa} : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \) as another scalar reproducing kernel with associated real-valued scalar RKHS \( \mathcal{H}_{\hat{\kappa}} \). In particular, \( \hat{\kappa} \) is only a function of the first \((n - m)\) components of \( x \) in both arguments.

For all \((p, q) \in \{1, \ldots, (n - m)\} \), we will take \( w_{pq} \in \mathcal{H}_\kappa \) (corresponding to \( W_1 \)) and all other entries of \( W \) as functions in \( \mathcal{H}_{\hat{\kappa}} \). Define the kernel derivative functions:

\[
\partial_j \kappa_x(z) := \frac{\partial \kappa}{\partial x_j}(r, s) \bigg|_{(r=x, s=z)} \quad \partial_j \hat{\kappa}_x(z) := \frac{\partial \hat{\kappa}}{\partial x_j}(r, s) \bigg|_{(r=x, s=z)} \quad \forall z \in \mathcal{X}.
\]

From [38], it follows that the kernel derivative functions satisfy the following two properties, similar to Theorem 2:

\[
\partial_j \kappa_x \in \mathcal{H}_\kappa \quad \forall j = 1, \ldots, n, \ x \in \mathcal{X}
\]

\[
\frac{\partial f}{\partial x_j}(x) = \langle f, \partial_j \kappa_x \rangle_{\mathcal{H}_\kappa}, \quad \forall f \in \mathcal{H}_\kappa.
\]

A similar property holds for \( \hat{\kappa} \) and \( \mathcal{H}_{\hat{\kappa}} \). Consider then the following finite-dimensional spaces:

\[
\mathcal{V}_\kappa := \left\{ \sum_{i=1}^{N_c} a_i \kappa_{x_i} + \sum_{i=1}^{N_c} \sum_{p=1}^{n} a_{ip}' \partial_p \kappa_{x_i}, \ a_i, a_{ip}' \in \mathbb{R} \right\} \subset \mathcal{H}_\kappa \quad (26)
\]

\[
\mathcal{V}_{\hat{\kappa}} := \left\{ \sum_{i=1}^{N_c} c_i \hat{\kappa}_{x_i} + \sum_{i=1}^{N_c} \sum_{p=1}^{n} c_{ip}' \partial_p \hat{\kappa}_{x_i}, \ c_i, c_{ip}' \in \mathbb{R} \right\} \subset \mathcal{H}_{\hat{\kappa}} \quad (27)
\]

and the proposed representation for \( W(x) \):

\[
W(x) = \sum_{i=1}^{N_c} \hat{\Theta}_i \hat{\kappa}_{x_i}(x) + \sum_{i=1}^{N_c} \sum_{j=1}^{n} \partial'_{ij} \partial_j \hat{\kappa}_{x_i}(x)
\]

\[
+ \sum_{i=1}^{N_c} \Theta_i \kappa_{x_i}(x) + \sum_{i=1}^{N_c} \sum_{j=1}^{n} \Theta'_{ij} \partial_j \kappa_{x_i}(x), \quad (28)
\]

where \( \Theta, \Theta' \in \mathbb{S}_n \) are constant symmetric matrices with non-zero entries only in the top-left \((n - m) \times (n - m)\) block, and \( \Theta, \Theta' \in \mathbb{S}_n \) are constant symmetric matrices with zero entries in the top-left \((n - m) \times (n - m)\) block.

**Theorem 4 (Representer Theorem for \( W \)).** Consider the pointwise-relaxed metric sub-problem for problem (10):

\[
\begin{align*}
\min_{w_{pq} \in \mathcal{H}_\kappa, (p, q) \in \{1, \ldots, (n - m)\}} & \quad J_m(W, \overline{w}, \overline{w}) \\
\text{subject to} & \quad F(x_i; \hat{f}, W, \lambda) \leq 0, \quad \forall x_i \in X_c, \\
& \quad \overline{w} I_n \preceq W(x_i) \preceq \overline{w} I_n, \quad \forall x_i \in X_c,
\end{align*}
\]

where \( \overline{w} \) is a vector of weights with non-zero entries only in the top-left \((n - m) \times (n - m)\) block.
Suppose the feasible set of the above LMI constrains is non-empty. Denote $W^*$ as the optimizer for this sub-problem. Then, $W^*$ takes the form given in (28).

Proof. Notice that while the regularizer term is strictly convex, the surrogate loss function for the condition number is affine. However, provided the feasible set is non-empty, there still exists a minimizer (possible non-unique) for the above sub-problem.

Now, notice that the $(p,q)$ element of $\partial f W_{\perp} (x_i)$ takes the form

$$\partial f w_{\perp pq} (x_i) = \sum_{j=1}^n \frac{\partial w_{\perp pq} (x_i)}{\partial x^j} f^j (x_i)$$

$$= \sum_{j=1}^n \langle w_{\perp pq}, \partial_j \hat{\kappa}_{x_i} \rangle_{\mathcal{H}_{\hat{\kappa}}} f^j (x_i)$$

$$= \langle w_{\perp pq}, \sum_{j=1}^n f^j (x_i) \partial_j \hat{\kappa}_{x_i} \rangle_{\mathcal{H}_{\hat{\kappa}}}$$

$$= \langle w\hat{\kappa}, \sum_{j=1}^n f^j (x_i) \partial_j \hat{\kappa}_{x_i} \rangle_{\mathcal{H}_{\hat{\kappa}}}$$

Additionally, the $(p,q)$ element of $W(x_i)$ takes the form

$$w_{pq} (x_i) = \begin{cases} 
\langle w\hat{\kappa}, \hat{\kappa}_{x_i} \rangle_{\mathcal{H}_{\hat{\kappa}}} & \text{if } (p,q) \in \{1, \ldots, (n-m)\} \\
\langle w\kappa, \kappa_{x_i} \rangle_{\mathcal{H}_{\kappa}} & \text{else.}
\end{cases}$$

That is, constraints (9) and uniform definiteness at the constraint points in $X_c$ can be written in terms of functions in $\mathcal{V}_{\kappa}$ and $\mathcal{V}_{\hat{\kappa}}$ alone. By strict convexity of the regularizer, and recognizing that $W(x)$ is symmetric, the result follows.

Similar to the previous section, the key property here is the reduction of the infinite-dimensional search over $W(x)$ to the finite-dimensional convex optimization problem over the constant symmetric matrices $\Theta_i, \Theta'_i, \hat{\Theta}_i, \hat{\Theta}'_i$ by choosing the function class for the entries of $W$ using the scalar-valued RKHS.

At this point, both sub-problems are finite-dimensional convex optimization problems. Crucially, the only simplifications made are those given in Section 4.2. However, a final computational challenge here is that the number of parameters scales with the number of training $N$ and constraint $N_c$ points. This is a fundamental challenge in all non-parametric methods. In the next section we present the dimensionality reduction techniques used to alleviate these issues.

B.4 Approximation via Random Matrix Features

The size of the problem using full matrix-valued kernel expansions in grows rapidly in $N_c n$, the number of constraint points times the state dimensionality. This makes training slow for even moderately long demonstrations even in low-dimensional settings. The induced dynamical system is slow to evaluate and integrate at inference time. Random feature approximations to kernel functions have been extensively used to scale up training complexity and inference speed of kernel methods [9, 25] in a number of applications. The quality of approximation can be explicitly controlled by the number of
random features. In particular, it has been shown [26] that any function in the RKHS associated with the exact kernel can be approximated to arbitrary accuracy by a linear combination of sufficiently large number of random features.

These approximations have only recently been extended to matrix-valued kernels [2, 22]. Given a matrix-valued kernel \( K \), one defines an appropriate matrix-valued feature map \( \Phi : \mathcal{X} \to \mathbb{R}^{d \times n} \) with the property

\[
K(x, z) \approx \Phi(x)^T \Phi(z),
\]

where \( d \) controls the quality of this approximation. With such an approximation, notice that any function \( g \) in the associated RKHS \( \mathcal{H}_K \) can be re-parameterized as:

\[
g(x) = \sum_{i=1}^{N_c} K(x, x_i) a_i \approx \sum_{i=1}^{N_c} \Phi(x)^T \Phi(x_i) a_i = \Phi(x)^T \alpha,
\]

where \( \alpha = \sum_{i=1}^{N_c} \Phi(x_i) a_i \in \mathbb{R}^d \). Thus, the function \( g \) is now summarized by the \( d \)-dimensional vector \( \alpha \) instead of \( N_c \) vectors in \( \mathbb{R}^n \) (\( N_c n \) parameters). Applying a similar trick to:

\[
\frac{\partial}{\partial s j} K(x, z) \approx \Phi(x)^T \frac{\partial}{\partial s j} \Phi(s) \bigg|_{s=z},
\]

one can approximate terms of the form

\[
\sum_{i=1}^{N_c} \sum_{j=1}^{n} \partial_j K_{x, i} c_{ij}(x)
\]

by \( \Phi(x)^T v \) for some \( v \in \mathbb{R}^d \). Applying this approximation to the subspaces in (22), (23), and (28), we finally arrive at the function parameterizations\(^4\) in eqs. (13)–(15), and the constraints reformulation in problem (16) in terms of the constant vectors. The regularization terms in the objective follow from the definition of the inner product in \( \mathcal{H}_K \):

\[
\|g\|_{\mathcal{H}_K}^2 = \sum_{i, j=1}^{N_c} \langle a_i, K(x_i, x_j) a_j \rangle \approx \sum_{i, j=1}^{N_c} \langle \Phi(x_i) a_i, \Phi(x_j) a_j \rangle = \left\langle \sum_{i=1}^{N_c} \Phi(x_i) a_i, \sum_{j=1}^{N_c} \Phi(x_j) a_j \right\rangle = \|\alpha\|_2.
\]

A canonical example is the the Gaussian separable kernel \( K_\sigma(x, z) := e^{-\frac{\|x-z\|^2}{2\sigma^2}} I_n \) with feature map:

\[
\frac{1}{\sqrt{s}} \begin{bmatrix}
\cos(\omega_1^T x) \\
\sin(\omega_1^T x) \\
\vdots \\
\cos(\omega_s^T x) \\
\sin(\omega_s^T x)
\end{bmatrix} \otimes I_n
\]

where \( \omega_1, \ldots, \omega_s \) are i.i.d. draws from \( \mathcal{N}(0, \sigma^{-2} I_n) \).

\(^4\) To apply this decomposition to \( W(x) \), we simply leverage vector-valued feature maps for each entry of \( W(x) \).
C Solution Parameters for PVTOL

Notice that the true input matrix for the PVTOL system satisfies Assumption 1. Furthermore, it is a constant matrix. Thus, the feature mapping $\Phi_b$ is therefore just a constant matrix with the necessary sparsity structure.

The feature matrix for $f$ was generated using the Random Fourier approximation to the Gaussian separable matrix-valued kernel (see Appendix B.4) with $\sigma = 6$ and $s = 8n = 48$ sampled Gaussian directions, yielding a feature mapping matrix $\Phi_f$ with $d_f = 576$ (96 features for each component of $f$). The scalar-valued reproducing kernels for the entries of $W$ were taken to be the Gaussian kernel with $\sigma = 15$. To satisfy condition (8), the kernel for $w_{ij}$, $(i,j) \in \{1, \ldots, (n - m)\}$ was only a function of the first $n - m$ components of $x$. A total of $s = 36$ Gaussian samples were taken to yield feature vectors $\phi_w$ and $\hat{\delta}_w$ of dimension $d_w = 72$. Furthermore, by symmetry of $W(x)$ only $n(n + 1)/2$ functions were actually parameterized. Thus, the learning problem in (16) comprised of $d_f + d_w n(n + 1)/2 + 4 = 508$ parameters for the functions, plus the extra scalar constants $\lambda, w, \bar{w}$.

The learning parameters used were: model N-R (all $N$): $\mu_f = 0, \mu_b = 10^{-6}$, R-R (all $N$): $\mu_f = 10^{-6}, \mu_b = 10^{-6}$, CCM-R (all $N$): $\mu_f = 10^{-3}, \mu_b = 10^{-6}$; $N \in \{100, 250, 500\}$ : $\mu_w = 10^{-3}, N = 1000 : \mu_w = 10^{-4}$. Tolerance parameters: constraints: $\{\delta_\lambda, \epsilon_\lambda, \delta_w, \epsilon_w\} = \{0.01, 0.01, 0.01, 0.01\}$; discard tolerance $\delta = 0.05$. Note that a small penalization on $\mu_b$ was necessary for all models due to the fact that the feature matrix $\Phi_b$ is rank-deficient.

D CCM Controller Synthesis

Let $\Gamma(p, q)$ be the set of smooth curves $c : [0, 1] \to \mathcal{X}$ satisfying $c(0) = p, c(1) = q$, and define $\delta_c(s) := \partial c(s)/\partial s$. At each time $t$, given the nominal state/control pair $(x^*, u^*)$ and current actual state $x$:

Step 1: Compute a curve $\gamma \in \Gamma(x^*, x)$ defined by:

$$\gamma \in \arg\min_{c \in \Gamma(x^*, x)} \int_0^1 \delta_c(s)^T M(c(s)) \delta_c(s) ds,$$

and let $E$ denote the minimal value.

Step 2: Define:

$$E_d(k) := 2\delta_c(1)^T M(x)(f(x) + B(x)(u^* + k))$$

$$- 2\delta_c(0)^T M(x^*)(f(x^*) + B(x^*)u^*).$$

Step 3: Choose $k(x^*, x)$ to be any element of the set:

$$K := \{k : E_d(k) \leq -2\lambda E\}.$$

By existence of the metric/differential controller pair $(M, \delta_u)$, the set $K$ is always non-empty. The resulting $k(x^*, x)$ then ensures that the solution $x(t)$ indeed converges towards $x^*(t)$ exponentially [32].
From an implementation perspective, note that having obtained the curve $\gamma$, constraint (33) is simply a linear inequality in $k$. Thus, one can analytically compute a feasible feedback control value, e.g., by searching for the smallest (in any norm) element of $K$; for additional details, we refer the reader to [17, 32].