mathematics

MDPI

Article

Error Amplification in Koopman Predictive Control: Computable Bounds, a Conditional Lyapunov Certificate, and Horizon-Consistent Estimation

Jia'en Zhang [1,*] and Di Tian [2]

[1] Department of Data and Systems Engineering, The University of Hong Kong, Pokfulam, Hong Kong SAR, China
[2] Mechanical Engineering College, Xi'an Shiyou University, Xi'an 710065, China; tiandi@xsyu.edu.cn
* Correspondence: u3625911@connect.hku.hk

Abstract

Koopman model predictive control (MPC) iterates a lifted linear predictor over a horizon of tens of samples. Extended dynamic mode decomposition with control (EDMDc), however, fits that predictor by minimising only the one-step residual in the lifted space. This paper bounds the h-step output error of any lifted linear predictor by its one-step residual times an amplification factor $\Gamma_h = \sum_{j<h} \|CA^j\|_2$ that is computable from the model alone. The bound is attained for single-output systems, takes a horizon-uniform form through a scaled Lyapunov certificate, and extends to the suboptimality of the input sequence that MPC selects. The proposed HC-EDMDc minimises the rolled-out output error under a spectral-radius constraint, starts from the closed-form EDMDc solution as a warm start, and is selected by a held-out rolled-out criterion. The evaluation uses four measured benchmarks: a six-axis KUKA KR300 industrial robot, an electro-mechanical positioning system, a cascaded-tank process and the Silverbox oscillator. The one-step estimator returns a non-Schur transition matrix on 8 of 20 fits across 3 records, and its free run leaves the data range; the constraint removes this failure for at most 8.4 % of twenty-step accuracy. HC-EDMDc improves long-horizon accuracy on 2 records, reducing the twenty-step NRMSE on the cascaded tanks from 0.3431 for EDMDc + SR to 0.2827, and returns its warm start on the other 2; the size of the one-step residual does not predict which case a record falls into. Minimising $\varepsilon\,\Gamma_H$ collapses the predictor towards a mean-reverting model on 3 of the 4 records, and the certified bound moves opposite to the realised error on 2 of them. A valid bound is therefore not a model-selection criterion.

Keywords: Koopman operator; extended dynamic mode decomposition; model predictive control; system identification; prediction horizon; transient amplification; industrial robot; data-driven control

MSC: 93B30; 93B45; 47B33

Check for updates

Academic Editor: Xi Wang

Received: 20 August 2026
Revised: 14 October 2026
Accepted: 22 October 2026
Published: 4 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Model predictive control (MPC) is the standard method for driving constrained multivariable machinery, from industrial manipulators to process units, because it recasts a control specification as an optimisation problem solved online [1,2]. On embedded hardware, its appeal rests on one structural property. When the prediction model is linear, the optimisation is a convex quadratic program (QP), and modern operator-splitting solvers

dispatch such problems in microseconds [3]. Nonlinear plants therefore set fidelity against tractability, and much of the data-driven control literature is devoted to reconciling the two.

The Koopman operator offers one reconciliation. Rather than linearising the dynamics around an operating point, it lifts the state into a space of observables in which the evolution is linear but infinite dimensional [4–7]. Dynamic mode decomposition made this operator accessible from data by extracting modes directly from snapshot sequences [8], and its connection to the Koopman spectrum was established shortly afterwards [9]. Extended dynamic mode decomposition (EDMD) approximates the operator on a finite dictionary by least squares [10,11]. Its convergence to the Koopman operator as the dictionary and the sample count grow is understood [12–14], spectral and stability-analytic uses of the operator are well developed [15,16], and finite-data error bounds are now available [17–19]. How to choose or learn the dictionary remains an active question [20–23], and Koopman autoencoders learn the observables jointly with the linear dynamics, often under multi-step or consistency losses [24–26].

Embedding the resulting linear predictor in MPC yields Koopman MPC [27]. The scheme has been applied to fluid systems [28], soft manipulators [29] and robotic platforms [30,31], extended to robust and tube-based formulations [32–34], and equipped with closed-loop stability guarantees under explicit error bounds [35–37]. Bilinear lifted forms retain more of the input dependence at the price of a non-convex control problem and constitute a parallel line of work [38–40].

Data-driven and machine-learning methods now extend well beyond control, spanning structural health monitoring and civil infrastructure [41–46], graph representation learning and network topology identification [47–51], antenna and electromagnetic design [52–54], energy and battery management systems [55–58], image reconstruction and remote sensing [59,60], fire-risk prediction [61,62], biomedical therapeutics [63], enterprise and social analytics [64–66], financial risk management [67], and natural-language processing [68]. In the control domain that the present work occupies, a different route to the same goal dispenses with an identified model altogether. Behavioural methods predict directly from raw trajectory libraries [69–71]. Adaptive dynamic programming and reinforcement-learning methods learn the control law or value function from data. Event-triggered adaptive dynamic programming with a dynamic penalty factor handles input saturation with uniformly ultimately bounded closed loops [72], and event-triggering conditions derived from the Hamilton–Jacobi–Bellman solution come with a performance upper bound and a minimum inter-event time [73]. Extended disturbance and state observers support event-triggered data-driven control of networked and disturbed nonlinear systems [74,75], a pseudo-partial-derivative description of the input–output map yields a Q-learning control law with an event-triggering rule [76], and off-policy interleaved Q-learning obtains optimal controllers for unknown affine nonlinear systems from measured trajectories alone [77]. These methods do not iterate a predictive model, which suits online adaptation and communication-limited settings. The present paper concerns the other, model-based branch, in which a lifted linear predictor is identified first and then embedded in MPC. Only this branch provides the QP structure, constraint handling and a state-space model that can be simulated, and it is in this branch that the mismatch between one-step fitting and multi-step use arises.

However, the predictor is *fitted* in one way and *used* in another. EDMD with control (EDMDc) minimises a one-step residual in the lifted space, whereas MPC propagates the model over a horizon of tens of samples, so the relevant quantity is the accuracy of the h-step map for h up to the horizon. Composing an approximate operator with itself amplifies its residual. The amplification is governed by the transient behaviour of the lifted transition matrix rather than by its spectral radius, a distinction that is classical for

non-normal matrices [78] and that, as shown below, dominates the practical behaviour of Koopman predictors on real records. Recent work reaches the same diagnosis from the estimation side and learns the condensed multi-step map directly, which restores convexity at the price of horizon-specific matrices [79].

To address these gaps, this paper approaches the problem from the opposite direction. It keeps the single recursive linear predictor, which is what makes free-run simulation, spectral analysis and state-space stability certificates available, and asks what has to change in the estimator. The study combines an analysis of how the one-step residual propagates over the horizon, an estimator built on that analysis, a bound that carries model error into the closed loop, and an evaluation on measured data. The contributions are as follows.

(i) An error-amplification analysis (Section 3) bounds the h-step output error of any lifted linear predictor by the product of its one-step residual and the amplification factor $\Gamma_h = \sum_{j<h} \|CA^j\|$, which is computed directly from (A, C). A Lyapunov certificate gives a horizon-uniform version of the bound, and a construction shows that the bound is attained in the single-output case. Whereas existing finite-data error bounds [17,18] express the compounded error through spectral or Lipschitz-type constants, Γ_h is a diagnostic of a single identified model and explains quantitatively why a predictor with an almost vanishing one-step residual can be useless over an MPC horizon.

(ii) A stability-constrained horizon-consistent estimator, HC-EDMDc (Section 4), minimises the rolled-out output error over the horizon subject to a Schur constraint enforced by a spectral-radius retraction. It is warm-started at the closed-form EDMDc solution and selected on a held-out rolled-out criterion, so by construction, it is never worse than its warm start on that criterion (Proposition 3). At $H = 1$, it reduces to a convex one-step *output*-error problem that differs from EDMDc by the readout projection (Proposition 4). Unlike the horizon-specific condensed maps of [79], HC-EDMDc retains a single recursive predictor that can be simulated indefinitely, carries a spectral-radius certificate, and has a parameter count independent of H. Compared with the multi-step and consistency losses of Koopman autoencoders [24–26], it adds the spectral-radius constraint and the held-out selection rule.

(iii) A negative result shows that a valid bound is not automatically a model-selection criterion. The bound of Theorem 1 is valid and worst-case tight, yet along a regularisation path its value is *negatively* correlated with the realised horizon error on two of the four benchmarks, uncorrelated on a third and positively correlated on the fourth. Minimising the product $\varepsilon\,\Gamma_H$, the surrogate an optimiser is naturally handed, collapses the predictor towards a mean-reverting model on the three records on which that optimisation makes any progress at all.

(iv) The methods are evaluated on four *measured* benchmarks rather than on simulated examples, namely a six-degree-of-freedom KUKA KR300 R2500 industrial robot, an electro-mechanical positioning system, a cascaded-tank process and the Silverbox electronic oscillator. A closed-loop study is carried out on a first-principles positioning-system model identified from the same measurements. The paper reports where the horizon-consistent fit helps, where it is neutral and why, and it also reports the computational cost (Section 8.2) and when the estimator is worth using (Section 8.3). Section 8.4 compares the approach with the related methods discussed above.

No single estimator dominates. The stability constraint prevents the free runs that leave the data range, which the plain one-step estimator produces on two of the four benchmarks, at a cost of up to 8.4 % in twenty-step accuracy and considerably more at one step. The horizon-consistent refinement helps materially on the two records where it improves on its closed-form warm start, and is neutral on the other two, where it returns that warm start unchanged. Which regime a record falls into is not predicted by the size of

its one-step residual, and Section 7 states this explicitly rather than leaving the reader to infer a rule that the data do not support.

The remainder of this paper is organised as follows. As Figure 1 shows, measured records pass through lifting and closed-form estimation to the amplification analysis, the horizon-consistent estimator, and finally open-loop and closed-loop evaluation. Section 2 fixes notation and recalls EDMDc and Koopman MPC. Section 3 develops the amplification analysis, and Section 4 defines the estimator and states its properties. Section 5 carries the model error into the closed loop. Section 6 describes the benchmarks and the protocol, and Section 7 reports the results. Section 8 discusses what the results show, the computational cost and scalability, when to use the estimator, its relation to the literature, and the limitations of the study (Section 8.5). Section 9 concludes.

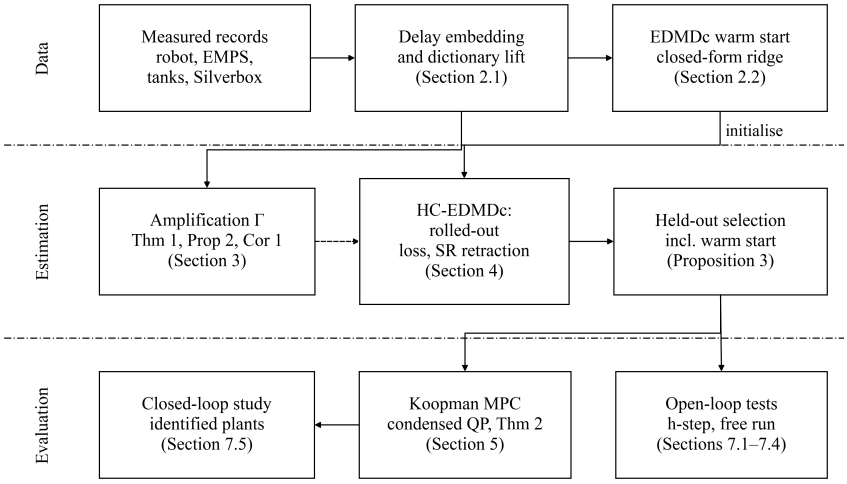


Figure 1. Overview of the proposed framework. Measured records are delay-embedded and lifted by a dictionary. The closed-form EDMDc solution serves as the warm start, and the amplification analysis yields Γ_h. HC-EDMDc minimises the rolled-out output loss under a spectral-radius retraction and selects, by held-out loss, among candidates that include the warm start. The resulting predictor is used for open-loop evaluation and in Koopman MPC, the latter in a closed-loop study on an identified first-principles plant. Dashed arrows indicate where the analysis informs the design of the estimator.

2. Problem Setting and Preliminaries

2.1. Plant, Data, and Lifted Predictor

We consider a discrete-time plant with input $u_k \in \mathbb{R}^m$ and measured output $y_k \in \mathbb{R}^p$ generated by an unknown dynamical system

$$\xi_{k+1} = f(\xi_k, u_k), \qquad y_k = g(\xi_k), \tag{1}$$

whose internal state ξ_k is not available. The data are one or more finite input–output records $\{(u_k, y_k)\}_{k=0}^{T-1}$ collected from the physical device.

Since the state is unmeasured, we work with a delay-embedded surrogate state. Fix delay orders $n_a, n_b \in \mathbb{N}$ and set

$$z_k = \begin{bmatrix} y_k^\top & \cdots & y_{k-n_a+1}^\top & u_{k-1}^\top & \cdots & u_{k-n_b}^\top \end{bmatrix}^\top \in \mathbb{R}^{d_z}, \qquad d_z = n_a p + n_b m. \tag{2}$$

Under the usual observability and genericity conditions, the delay vector (2) is a valid state for the input–output behaviour of (1) [80], and delay coordinates are a standard choice of Koopman observables [81,82].

The Koopman operator $\mathcal{K}$ associated with (1) acts on observables $\varphi : \mathbb{R}^{d_z} \to \mathbb{R}$ by composition,

$$(\mathcal{K}^u \varphi)(z) = \varphi\big(F(z,u)\big), \tag{3}$$

where F denotes the induced map on delay coordinates. It is linear but infinite dimensional, and a finite-dimensional approximation is obtained by restricting it to the span of a dictionary $\Psi = (\psi_1, \ldots, \psi_N)^\top$. We use the standard control-oriented form in which the lifted state carries the delay state itself,

$$\psi_k = \Psi(z_k) = \begin{bmatrix} z_k \\ \phi(z_k) \end{bmatrix} \in \mathbb{R}^N, \tag{4}$$

with ϕ a fixed nonlinear feature map. Two dictionaries are used below: random Fourier features approximating a Gaussian kernel [83], and a random subset of quadratic monomials. The predictor is the linear system

$$\hat{\psi}_{k+1} = A\hat{\psi}_k + Bu_k, \qquad \hat{y}_k = C\hat{\psi}_k, \tag{5}$$

with $A \in \mathbb{R}^{N\times N}$, $B \in \mathbb{R}^{N\times m}$ and a readout $C \in \mathbb{R}^{p\times N}$. Because the lifted state contains z_k, whose leading block is y_k, the readout is exactly representable; we obtain it by least squares on the fitting data,

$$C = \arg\min_{\tilde{C}} \sum_{k\in\mathcal{T}} \|\tilde{C}\Psi(z_k) - y_k\|_2^2. \tag{6}$$

Crucially, once (5) is rolled forward, the lifted state is *not* re-lifted: the predictor is a genuine linear system, which is exactly the property that makes the downstream control problem a QP.

2.2. EDMD with Control

Given consecutive pairs $\{(\psi_k, u_k, \psi_{k+1})\}_{k\in\mathcal{T}}$ on a fitting index set $\mathcal{T}$ with $|\mathcal{T}| = T_f$, EDMDc [10,27,84] determines (A, B) from the regularised one-step problem

$$(A,B) = \arg\min_{\tilde{A},\tilde{B}} \frac{1}{T_f} \sum_{k\in\mathcal{T}} \big\|\psi_{k+1} - \tilde{A}\psi_k - \tilde{B}u_k\big\|_2^2 + \lambda\big(\|\tilde{A}\|_F^2 + \|\tilde{B}\|_F^2\big), \tag{7}$$

whose solution is available in closed form. Writing $G_k = [\psi_k^\top \; u_k^\top]^\top$,

$$\begin{bmatrix} A & B \end{bmatrix} = \Big(\tfrac{1}{T_f}\sum_{k\in\mathcal{T}} \psi_{k+1} G_k^\top\Big)\Big(\tfrac{1}{T_f}\sum_{k\in\mathcal{T}} G_k G_k^\top + \lambda I\Big)^{-1}. \tag{8}$$

The residual that (7) controls is the root-mean-square one-step lifted residual

$$\varepsilon(A,B) = \Big(\frac{1}{T_f}\sum_{k\in\mathcal{T}} \big\|\psi_{k+1} - A\psi_k - Bu_k\big\|_2^2\Big)^{1/2}. \tag{9}$$

2.3. Koopman MPC

For a prediction horizon N_p, weights $Q \succeq 0$, $R \succ 0$, and $S \succeq 0$, and input box constraints $u \in [\underline{u}, \overline{u}]$, the tracking problem solved at each sampling instant is

$$\begin{aligned} \min_{u_{0:N_p-1}} \quad & \sum_{i=1}^{N_p} \|\hat{y}_i - r_i\|_Q^2 + \sum_{i=0}^{N_p-1} \Big(\|u_i\|_R^2 + \|u_i - u_{i-1}\|_S^2\Big) \\ \text{s.t.} \quad & \hat{\psi}_{i+1} = A\hat{\psi}_i + Bu_i, \quad \hat{y}_i = C\hat{\psi}_i, \quad \hat{\psi}_0 = \Psi(z_k), \\ & \underline{u} \le u_i \le \overline{u}, \qquad i = 0, \ldots, N_p - 1. \end{aligned} \tag{10}$$

Eliminating the lifted states gives the condensed prediction

$$\underbrace{\begin{bmatrix}\hat{y}_1\\ \vdots\\ \hat{y}_{N_p}\end{bmatrix}}_{\hat{Y}} = \underbrace{\begin{bmatrix}CA\\ \vdots\\ CA^{N_p}\end{bmatrix}}_{E}\hat{\psi}_0 + \underbrace{\begin{bmatrix}CB & & \\ \vdots & \ddots & \\ CA^{N_p-1}B & \cdots & CB\end{bmatrix}}_{F}\underbrace{\begin{bmatrix}u_0\\ \vdots\\ u_{N_p-1}\end{bmatrix}}_{U}, \tag{11}$$

so that (10) becomes the box-constrained QP

$$\min_{U}\ \tfrac{1}{2}U^\top PU + q(\hat{\psi}_0, r, u_{-1})^\top U \quad \text{s.t.} \quad \underline{U} \le U \le \overline{U}, \tag{12}$$

with $P = 2\big(F^\top \bar{Q}F + \bar{R} + D^\top \bar{S}D\big)$, $\bar{Q} = I_{N_p} \otimes Q$, $\bar{R} = I_{N_p} \otimes R$, $\bar{S} = I_{N_p} \otimes S$ and D the first-difference matrix on U; the implementation uses a scalar S. Equation (11) makes the object of interest explicit: the controller consumes the powers CA^j and the Markov parameters CA^jB for $j < N_p$, not the one-step map.

3. Error Amplification over the Prediction Horizon

This section makes precise what the estimator (7) does and does not control. Let $\psi_k = \Psi(z_k)$ denote the lifted states actually generated by the plant data and define the one-step lifted residuals

$$r_k = \psi_{k+1} - A\psi_k - Bu_k, \qquad k = 0, 1, \dots \tag{13}$$

so that ε in (9) is the root-mean-square norm of r_k over the fitting set. We also allow for a readout mismatch $e_k = y_k - C\psi_k$, which is zero up to numerical precision for the lift (4) because y_k is a linear function of z_k.

Definition 1 (Amplification factor)**.** *For a predictor (A, B, C) and a horizon $h \ge 1$,*

$$\Gamma_h(A, C) = \sum_{i=0}^{h-1} \big\|CA^i\big\|_2. \tag{14}$$

Theorem 1 (Horizon error bound)**.** *Let $\hat{y}_{k+h}$ be the h-step prediction obtained by rolling (5) forward from $\hat{\psi}_k = \psi_k$ with the realised inputs $u_k, \dots, u_{k+h-1}$. Then,*

$$\psi_{k+h} = A^h\psi_k + \sum_{j=0}^{h-1} A^{h-1-j}\big(Bu_{k+j} + r_{k+j}\big), \tag{15}$$

and, consequently,

$$\big\|y_{k+h} - \hat{y}_{k+h}\big\|_2 \ \le\ \Gamma_h(A, C)\ \max_{0\le j<h}\|r_{k+j}\|_2 + \|e_{k+h}\|_2. \tag{16}$$

Moreover, let $\mathcal{S}_h$ be a set of window starts such that $r_t, \dots, r_{t+h-1}$ and e_{t+h} are defined for every $t \in \mathcal{S}_h$, and let

$$\varepsilon_h = \Big(\frac{1}{h|\mathcal{S}_h|}\sum_{t\in\mathcal{S}_h}\sum_{j=0}^{h-1}\|r_{t+j}\|_2^2\Big)^{1/2}, \qquad \bar{e}_h = \Big(\frac{1}{|\mathcal{S}_h|}\sum_{t\in\mathcal{S}_h}\|e_{t+h}\|_2^2\Big)^{1/2} \tag{17}$$

be the root-mean-square residual over the windows and the root-mean-square readout mismatch at their endpoints. Then, by the Cauchy–Schwarz inequality followed by the triangle inequality in $\ell^2(\mathcal{S}_h)$,

$$\Big(\tfrac{1}{|\mathcal{S}_h|}\textstyle\sum_{t\in\mathcal{S}_h}\|y_{t+h}-\hat{y}_{t+h}\|_2^2\Big)^{1/2} \;\le\; \sqrt{h\,\Gamma_h^{(2)}(A,C)}\;\varepsilon_h+\bar{e}_h, \qquad \Gamma_h^{(2)}=\textstyle\sum_{i=0}^{h-1}\|CA^i\|_2^2. \quad (18)$$

Proof. Equation (15) follows by induction on h: it holds trivially for $h=1$ by (13), and if it holds for h, then applying (13) once more and collecting terms gives the statement for $h+1$. The rolled-out prediction is $\hat{\psi}_{k+h}=A^h\psi_k+\sum_{j<h}A^{h-1-j}Bu_{k+j}$, so subtracting and multiplying by C,

$$y_{k+h}-\hat{y}_{k+h}=\sum_{j=0}^{h-1}CA^{h-1-j}r_{k+j}+e_{k+h}. \quad (19)$$

The triangle inequality and submultiplicativity give (16). For (18), apply the Cauchy–Schwarz inequality to (19) in the form $\|\sum_j CA^{h-1-j}r_{t+j}\|_2^2\le\big(\sum_j\|CA^{h-1-j}\|_2^2\big)\big(\sum_j\|r_{t+j}\|_2^2\big)=\Gamma_h^{(2)}\sum_j\|r_{t+j}\|_2^2$, average over $t\in\mathcal{S}_h$ to obtain $\Gamma_h^{(2)}\,h\,\varepsilon_h^2$, take square roots, and add the endpoint mismatch with the triangle inequality in $\ell^2(\mathcal{S}_h)$. □

The averaged form is stated over the windows $\mathcal{S}_h$ rather than over the fitting index set, because the residuals entering an h-step window need not all lie in the set on which ε of (9) is computed. The two statistics differ even when they do. Consecutive windows overlap, so a residual in the interior of $\mathcal{S}_h$ is counted once for each of the h windows that contain it while one near either end is counted fewer times, which makes ε_h an overlap-weighted root-mean-square rather than the uniform one of (9). On the records used here, the ratio $\varepsilon_h/\varepsilon$ stays within two percent of unity, but ε_h is the statistic the argument requires and Section 7.2 evaluates the certificate with it.

Theorem 1 separates two independent quantities. The estimator (7) minimises ε; nothing in it constrains Γ_h. The next two results show that the second factor is neither a technicality nor conservative.

Proposition 1 (Worst case over residual sequences)**.** *Fix A, C and h, and consider residual sequences with $\|r_{k+j}\|_2\le\varepsilon$. The worst-case residual contribution to the h-step output error equals*

$$\sup_{\|r_{k+j}\|_2\le\varepsilon}\Big\|\sum_{j=0}^{h-1}CA^{h-1-j}r_{k+j}\Big\|_2=\varepsilon\,\tilde{\Gamma}_h, \qquad \tilde{\Gamma}_h=\max_{\|w\|_2=1}\sum_{i=0}^{h-1}\big\|(CA^i)^\top w\big\|_2\;\le\;\Gamma_h. \quad (20)$$

For a single-output system ($p=1$), one has $\tilde{\Gamma}_h=\Gamma_h$, so in that case, the residual term of (16) is attained; for $p>1$, the worst case is $\tilde{\Gamma}_h$, which may be strictly smaller than Γ_h. The readout term $\|e_{k+h}\|_2$ of (16) is attained only if it is aligned with the residual contribution, so (16) as a whole is attained when $e\equiv 0$, which is the case for the lift (4) up to numerical precision.

Proof. Because the residuals are unconstrained relative to one another, the supremum decouples once a direction w of the output error is fixed: for fixed w, the largest achievable value of $w^\top\sum_j CA^{h-1-j}r_{k+j}$ is $\varepsilon\sum_j\|(CA^{h-1-j})^\top w\|_2$, attained at $r_{k+j}=\varepsilon(CA^{h-1-j})^\top w/\|(CA^{h-1-j})^\top w\|_2$ when that vector is non-zero and at an arbitrary admissible r_{k+j} otherwise, since such a term contributes nothing. Maximising over $\|w\|_2=1$ gives (20); the inequality $\tilde{\Gamma}_h\le\Gamma_h$ follows from $\|(CA^i)^\top w\|_2\le\|CA^i\|_2$. If $p=1$, then CA^i is a row vector, $w=\pm1$, and $\|(CA^i)^\top w\|_2=\|CA^i\|_2$ for every i. □

Corollary 1 (Horizon-uniform certificate)**.** *Let $\rho(A)<\bar{\rho}<1$. The scaled discrete Lyapunov equation*

$$(A/\bar{\rho})^\top P(A/\bar{\rho})-P=-I \quad (21)$$

has a unique solution $P\succ 0$, and with $M=\sqrt{\kappa(P)}$, $\kappa(P)=\lambda_{\max}(P)/\lambda_{\min}(P)$,

$$\|A^h\|_2 \le M\bar{\rho}^{\,h} \quad \textit{for all } h \ge 0, \qquad \textit{hence} \qquad \Gamma_h \le \|C\|_2\, M\, \frac{1-\bar{\rho}^{\,h}}{1-\bar{\rho}} \le \frac{\|C\|_2 M}{1-\bar{\rho}}. \tag{22}$$

Proof. Existence and uniqueness of $P \succ 0$ follow from $\rho(A/\bar{\rho}) < 1$. Writing $\|x\|_P^2 = x^\top P x$, (21) gives $\|(A/\bar{\rho})x\|_P^2 = \|x\|_P^2 - \|x\|_2^2 \le \|x\|_P^2$, so $\|(A/\bar{\rho})^h x\|_P \le \|x\|_P$. Converting between the P- and Euclidean norms with $\lambda_{\min}(P)\|x\|_2^2 \le \|x\|_P^2 \le \lambda_{\max}(P)\|x\|_2^2$ yields $\|A^h\|_2 \le \sqrt{\kappa(P)}\,\bar{\rho}^{\,h}$. Summing the geometric series and using $\|CA^i\|_2 \le \|C\|_2\|A^i\|_2$ gives (22). □

Two consequences are worth stating explicitly, because both are visible in the experiments of Section 7. First, $\rho(A) < 1$ alone does *not* keep Γ_h small: the constant M in (22) is a norm-equivalence factor between the Euclidean and the P-weighted norms. It equals one exactly when $A/\bar{\rho}$ is normal in that inner product and grows with the departure from normality, but it also grows when the modal decay rates are widely spread or when $\bar{\rho}$ is taken close to $\rho(A)$, so it is not a pure measure of non-normality. Either way, it can be arbitrarily large, which is the discrete-time face of transient growth in non-normal systems [78]. Second, $\rho(A) \ge 1$ does not by itself make Γ_h diverge—if the unstable modes are unobservable through C, as for $A = \mathrm{diag}(2, \frac{1}{2})$ with $C = [0\ \ 1]$, the sum (14) stays bounded—but it removes any horizon-uniform certificate, and whenever an unstable mode is observable, the sum grows geometrically in h. The retraction of Section 4 constrains $\rho(A)$, which is sufficient for a horizon-uniform bound and not necessary; we use it because it is the cheap and structure-agnostic choice, and because a free-run simulation of an unconstrained fit does leave the data range on two of our benchmarks.

Setting $\bar{\rho} = 1$ in the experiments is compatible with the hypothesis $\rho(A) < \bar{\rho} < 1$ of Corollary 1 in the following sense. Theorem 1 holds for any A and any finite h, so with $\bar{\rho} = 1$, it still provides a finite-horizon bound on every horizon used. The spectral-radius retraction returns a matrix with $\rho(A) \le 1$. Where the spectral radius is strictly below one, as for most of the fitted models in Section 7, any $\bar{\rho} \in (\rho(A), 1)$ satisfies the hypothesis of Corollary 1 and the horizon-uniform certificate applies. A fit that the retraction places exactly on the unit circle keeps the finite-horizon bound but has no horizon-uniform certificate. Section 7.4 reports the results for $\bar{\rho} \in \{0.99, 0.999, 0.9999\}$, values at which the corollary applies directly, and the free-run simulation errors for $\bar{\rho} = 0.9999$ and $\bar{\rho} = 1$ differ by at most 1 %. Taking $\bar{\rho} = 1$ in the main experiments therefore leaves the comparison between estimators unchanged.

The last result of this section explains *why* minimising a one-step criterion is not merely uninformative about Γ_h but actively biased, and it does so in a form that the noise experiment of Section 7 can test.

Proposition 2 (Attenuation bias of the one-step criterion)**.** *Let y be the stationary first-order autoregression $y_{k+1} = a\,y_k + e_{k+1}$ with $0 < a < 1$ and e zero-mean, i.i.d., of variance σ_e^2, so that $s^2 = \mathrm{Var}(y_k) = \sigma_e^2/(1-a^2)$. Let the measurements be $\tilde{y}_k = y_k + v_k$ with v_k zero-mean, i.i.d., independent of y and of e, of variance σ^2, and write $\nu = \sigma^2/s^2$. Let β_H be the population minimiser of the H-step criterion $\mathbb{E}\big[(\tilde{y}_{k+H} - \beta\,\tilde{y}_k)^2\big]$ and let $\alpha_H = \beta_H^{1/H}$ be the corresponding per-step rate. Then,*

$$\beta_H = \frac{a^H}{1+\nu}, \qquad \alpha_H = a\,(1+\nu)^{-1/H}, \qquad \alpha_1 = \frac{a}{1+\nu}, \qquad \alpha_H \uparrow a \ \textit{ as } H \to \infty. \tag{23}$$

Proof. The criterion is quadratic in β with minimiser $\beta_H = \mathbb{E}[\tilde{y}_{k+H}\tilde{y}_k]/\mathbb{E}[\tilde{y}_k^2]$. Independence of v from y gives $\mathbb{E}[\tilde{y}_{k+H}\tilde{y}_k] = \mathbb{E}[y_{k+H}y_k] = a^H s^2$ and $\mathbb{E}[\tilde{y}_k^2] = s^2 + \sigma^2$, so

$\beta_H = a^H/(1+\nu)$. Since $0 < a < 1$ and $\nu > 0$, $\beta_H > 0$ and $\alpha_H = \beta_H^{1/H} = a(1+\nu)^{-1/H}$ is the unique positive root. Monotonicity in H follows because $(1+\nu)^{-1/H}$ increases to 1. □

The restriction to $0 < a < 1$ keeps the per-step rate α_H well defined: for $a < 0$ and even H, the H-step criterion determines only β_H, and α_H is defined up to sign.

Proposition 2 is the familiar equation-error versus output-error distinction of system identification [85,86] in a form tied to the horizon: the one-step criterion shrinks the identified dynamics towards zero by a factor that a horizon-H criterion reduces to its H-th root. Read as a prediction about identification from noisy records, it suggests that a horizon-consistent fit should matter more as the output noise grows and less as the fit becomes exact. Section 7 tests that reading on the four benchmarks and does not confirm it; the proposition itself is a population statement about a scalar autoregression and is verified as such. Every statement of this section was checked against a brute-force computation before being written down, using the script `code/verify_theory.py` in the Supplementary Materials. The two sides of (15), computed once with explicit matrix powers and once by stepwise roll-out, differ by at most 3.6×10^{-14} in absolute value, and the inequalities were checked instance by instance on randomly generated examples, where (16) held on all 400 instances with a largest ratio of 0.64 between the actual error and $\Gamma_h \max_j \|r_{k+j}\|_2$, (18) held with a largest ratio of 0.41 between its two sides, and the ratio $\|A^h\|_2/(M\bar{\rho}^h)$ of Corollary 1 never exceeded 0.977. For Proposition 1, the analytic worst-case sequence attains Γ_h to a relative difference of 4×10^{-16}, whereas a direct search over 24,000 random residual sequences reaches at most 99.8 % of Γ_h and never exceeds it (with several outputs, $\tilde{\Gamma}_h \le \Gamma_h$ and the analytic sequence attains $\tilde{\Gamma}_h$), and the rate α_H of Proposition 2 agrees with Monte Carlo simulation to within 3.1×10^{-4} for $\nu \in \{0.05, 0.2, 0.5\}$ and $H \in \{1, 2, 5, 10\}$.

4. Stability-Constrained Horizon-Consistent Estimation

4.1. The Estimator

Theorem 1 suggests two modifications of (7): replace the one-step residual by the quantity the controller actually incurs, and keep Γ_h under control uniformly in h, which by Corollary 1 follows from a Schur constraint. Let $\mathcal{S}$ collect the indices t for which the whole window $t, \dots, t+H$ lies inside one contiguous record, and let $w_1, \dots, w_H \ge 0$ with $\sum_h w_h = 1$ be horizon weights. Define the rolled-out output loss

$$J_H(A,B) = \frac{1}{|\mathcal{S}|} \sum_{t \in \mathcal{S}} \sum_{h=1}^{H} w_h \left\| C\left(A^h \psi_t + \sum_{j=0}^{h-1} A^{h-1-j} B u_{t+j}\right) - y_{t+h} \right\|_2^2 + \lambda\left(\|A\|_F^2 + \|B\|_F^2\right), \tag{24}$$

and the (non-convex) stability set

$$\mathcal{A}_{\bar{\rho}} = \left\{ A \in \mathbb{R}^{N \times N} : \rho(A) \le \bar{\rho} \right\}. \tag{25}$$

The proposed estimator, HC-EDMDc, is

$$(A^\star, B^\star) \in \arg\min_{A \in \mathcal{A}_{\bar{\rho}},\, B} J_H(A,B), \tag{26}$$

approached numerically from the closed-form warm start (8). Three implementation elements matter and are stated explicitly so that the reported results are reproducible.

Rolled-out gradient. The map $A \mapsto A^h$ makes (24) a polynomial of degree H in A, so (26) is non-convex for $H > 1$; we evaluate J_H by propagating the predictor forward inside each window—never re-lifting—and differentiate through that recursion. Writing $\hat{\psi}_{t,h}$ for the h-step iterate and $\delta_{t,h} = C^\top\left(C\hat{\psi}_{t,h} - y_{t+h}\right)$, back-substitution gives

$$\nabla_A J_H = \frac{2}{|\mathcal{S}|} \sum_{t \in \mathcal{S}} \sum_{h=1}^{H} w_h \sum_{i=0}^{h-1} (A^\top)^{h-1-i} \, \delta_{t,h} \, \hat{\psi}_{t,i}^\top + 2\lambda A,$$
$$\nabla_B J_H = \frac{2}{|\mathcal{S}|} \sum_{t \in \mathcal{S}} \sum_{h=1}^{H} w_h \sum_{i=0}^{h-1} (A^\top)^{h-1-i} \, \delta_{t,h} \, u_{t+i}^\top + 2\lambda B. \tag{27}$$

Feasibility restoration. $\mathcal{A}_{\bar{\rho}}$ is not convex and the Euclidean projection onto it has no closed form. We use the scaling retraction

$$\mathcal{R}_{\bar{\rho}}(A) = \begin{cases} A, & \rho(A) \le \bar{\rho}, \\ \dfrac{\bar{\rho}}{\rho(A)} A, & \rho(A) > \bar{\rho}, \end{cases} \tag{28}$$

which is exact on the boundary, leaves feasible iterates untouched, and preserves the eigenvector structure of A while scaling its spectrum. It is a feasibility restoration, not a projection: $\mathcal{R}_{\bar{\rho}}(A)$ is in general not the closest feasible matrix to A. Constraining $\|A\|_2$ instead would be convex but is far too strong here, since a delay-embedded predictor is strongly non-normal and routinely has $\|A\|_2 \gg \rho(A)$.

Model selection. The ridge parameter λ and the stopping iterate are selected on a held-out slice of the training record, using the *data term* of (24) evaluated on that slice—the penalty is a fitting device and is not part of the validation criterion. The warm start is included among the candidates, and the retraction (28) is applied to every candidate before it is scored, so the model that is returned is the model whose validation loss was measured.

4.2. Properties

Proposition 3 (No degradation relative to the warm start)**.** *Let $\mathcal{C}$ be the set of iterates produced by the optimiser together with the retracted warm start $\mathcal{R}_{\bar{\rho}}(A_0), B_0$ of (8), every element of which is feasible for (25), and let the returned model minimise the held-out loss J_H^{val} over $\mathcal{C}$. Then, the returned model is feasible and $J_H^{\mathrm{val}}(A^\star, B^\star) \le J_H^{\mathrm{val}}(\mathcal{R}_{\bar{\rho}}(A_0), B_0)$.*

Proof. Immediate from $(\mathcal{R}_{\bar{\rho}}(A_0), B_0) \in \mathcal{C}$ and the definition of the selection rule; feasibility holds because (28) is applied to every candidate before it is scored. □

Proposition 3 is deliberately modest and close to a tautology. It guarantees only that a selection rule which counts the warm start among its candidates cannot return a model that is worse *on the criterion it selects by*. Nothing follows for the training loss or the test error, and the experiments report both; on one of the four benchmarks, the two do come apart. The statement is still worth one line, because it rules out a real failure mode of non-convex refinement, namely an optimiser that replaces a good closed-form solution by a worse iterate.

Proposition 4 (Behaviour at $H = 1$; non-convexity for $H \ge 2$)**.** *For $H = 1$, (24) reduces to the one-step* output-*error criterion $\frac{1}{|\mathcal{S}|} \sum_t \|C(A\psi_t + Bu_t) - y_{t+1}\|_2^2 + \lambda(\|A\|_F^2 + \|B\|_F^2)$, which is convex in (A, B); it differs from (7), which measures the residual in the lifted space, by the projection C. The same holds whenever $w_h = 0$ for all $h \ge 2$. If instead $w_h > 0$ for some $h \ge 2$, (24) is in general non-convex in A: for $N = p = 1$, $C = 1$, $B = 0$, $H = 2$, $w_2 = 1$, a single window with $\psi_t = 1$ and $y_{t+2} = 1$ gives $J_2(A) = (A^2 - 1)^2 + \lambda A^2$, whose second derivative at $A = 0$ is $-4 + 2\lambda$, negative for $\lambda < 2$.*

Proof. The first claim follows by substituting $H = 1$, $w_1 = 1$ into (24); the resulting map is a squared Euclidean norm of an affine function of (A, B). The second follows from the displayed instance, which is an admissible special case of (24). □

Four guarantees hold for HC-EDMDc. First, every candidate passes through the spectral-radius retraction (28) before it is scored, so the returned model satisfies $\rho(A) \le \bar{\rho}$. Second, by Proposition 3, it is no worse than the warm start on the held-out loss J_H^{val}. Third, (27) is the exact gradient of J_H; checked against automatic differentiation, the largest absolute discrepancy is 5.7×10^{-14} for $\nabla_A J_H$ and 2.8×10^{-14} for $\nabla_B J_H$. Fourth, for $H = 1$, the problem is convex (Proposition 4). For $H \ge 2$, J_H is a polynomial of degree H in A and the problem is non-convex, so we claim neither a global minimiser nor convergence to a first-order stationary point or a local minimiser. The retraction is a feasibility restoration rather than a Euclidean projection, and the convergence theory of projected gradient methods therefore does not cover these iterates. Whether the first-order condition at the warm start and a short-horizon refinement actually produce descent is checked on data in Section 8.3. Taken together, these guarantees mean that the returned predictor is always feasible and at least as good as the closed-form solution on the criterion used to select it.

4.3. Why Minimising the Bound Surrogate Is Not the Right Objective

Theorem 1 suggests a cheap proxy for the horizon error: the product of its two ingredients, $\varepsilon(A,B)\,\Gamma_H(A,C)$. We call that product a surrogate rather than a bound throughout, because it is neither of the two certified quantities. The pointwise form (16) pairs Γ_H with the largest residual in each window; the root-mean-square form (18) pairs the larger constant $\sqrt{H\Gamma_H^{(2)}} \ge \Gamma_H$ with the overlap-weighted residual ε_H and adds the readout term $\bar{e}_H$. The product substitutes the fitting-set root-mean-square ε for both residual statistics and drops $\bar{e}_H$, so it is not an upper bound on the horizon error; the only inequality that holds universally between the two constants is $\Gamma_H \le \sqrt{H\Gamma_H^{(2)}}$. It is nonetheless the quantity an optimiser is naturally handed, since it is the cheapest expression built from the two factors, and it is tempting to minimise it directly and thereby avoid the rolled-out recursion altogether. We implemented this surrogate-minimising estimator, which we label BR-EDMDc, as

$$(A,B) \in \arg\min_{A \in \mathcal{A}_{\bar{\rho}},\, B} \log \varepsilon(A,B) + \log \Gamma_H(A,C) + \lambda\left(\|A\|_F^2 + \|B\|_F^2\right), \tag{29}$$

where the two factors need no relative weight—that is the appeal of the criterion—and the small ridge term is retained only for conditioning, with the same λ as the warm start. The following observation explains what we then measured.

Proposition 5 (The surrogate prefers the trivial predictor)**.** *Let $B_0 \in \arg\min_B \frac{1}{T_f}\sum_k \|\psi_{k+1} - Bu_k\|_2^2$ be the best input-only model, $\varepsilon_0 > 0$ its residual, and let (A_1, B_1) be any feasible pair with residual $\varepsilon_1 > 0$; assume $C \neq 0$. The pair $(0, B_0)$ is feasible with $\Gamma_H(0,C) = \|C\|_2$, and the unregularised criterion of (29) assigns it a strictly smaller value than (A_1, B_1) whenever*

$$\Gamma_H(A_1, C) \;>\; \|C\|_2 \,\frac{\varepsilon_0}{\varepsilon_1}. \tag{30}$$

Proof. Compare the two values $\|C\|_2\varepsilon_0$ and $\Gamma_H(A_1,C)\,\varepsilon_1$ of the product $\varepsilon\Gamma_H$ and rearrange; the logarithm is increasing. That $\Gamma_H(0,C) = \|C\|_2$ follows from (14) with $A = 0$, only the $i = 0$ term surviving, and $\rho(0) = 0 \le \bar{\rho}$. □

Proposition 5 is a pairwise comparison, not a statement that the minimiser of (29) equals $A = 0$: it says that whenever (30) holds—taking for (A_1, B_1) the one-step solution, the estimator one would otherwise have used—the criterion ranks the predictor that ignores the state above it. That is already enough to make the criterion useless for selection, and Section 7 reports what the numerical minimiser actually returns.

Condition (30) is satisfied by a wide margin on every benchmark considered here, because $\varepsilon_0/\varepsilon_1$ is a modest ratio while Γ_H is one to two orders of magnitude larger than $\|C\|_2$. The measured consequence is reported in Section 7: on the three records where the optimiser moves away from its warm start, BR-EDMDc drives Γ into single digits and returns predictors whose accuracy is that of a mean-reverting model; on the fourth, it makes no progress at all. We record the result because the temptation to optimise a certificate is a natural one. The bound of Theorem 1 takes the worst case over residual *directions*—exactly $\tilde{\Gamma}_h$ of Proposition 1, which equals Γ_h for a single output—while the realised error involves the specific residual sequence produced by the data. The gap between the two is what makes the surrogate unusable as a training objective, and Section 7.2 measures it along a regularisation path.

5. From Horizon Error to Control Performance

The estimator of Section 4 targets the horizon prediction error. This section connects that quantity to the quality of the decision the controller makes with it. We keep the statement at the level at which it can be proved without extra assumptions—the suboptimality of the input sequence chosen at one sampling instant—and refer to the literature for the closed-loop stability results that build on error bounds of this type [2,35,37,87].

Fix a sampling instant, a reference $R = (r_1, \dots, r_{N_p})$ and the measured lifted state ψ_0. For an admissible input sequence $U \in \mathcal{U} = [\underline{U}, \overline{U}]$, let $\hat{Y}(U)$ be the prediction (11) and $Y(U)$ the response the plant would actually produce. With $\bar{Q} = I_{N_p} \otimes Q$, $\bar{R} = I_{N_p} \otimes R$ and $\bar{S} = I_{N_p} \otimes S$ as in (12), and $s_0 = [u_{-1}^\top \; 0 \; \cdots \; 0]^\top$ carrying the previously applied input, define the predicted and realised costs

$$\begin{aligned} &\hat{J}(U) = \|\hat{Y}(U) - R\|_{\bar{Q}}^2 + \ell(U), \qquad J(U) = \|Y(U) - R\|_{\bar{Q}}^2 + \ell(U), \\ &\ell(U) = \|U\|_{\bar{R}}^2 + \|DU - s_0\|_{\bar{S}}^2, \end{aligned} \tag{31}$$

and the prediction gap over the admissible set,

$$\Delta = \sup_{U \in \mathcal{U}} \big\|\hat{Y}(U) - Y(U)\big\|_{\bar{Q}}. \tag{32}$$

If, in addition, the one-step residuals and the readout mismatch generated by *every* admissible input sequence obey $\|r\|_2 \le \bar{r}$ and $\|e\|_2 \le \bar{e}$ uniformly along the window, then Theorem 1 applied horizon by horizon gives the computable bound

$$\Delta \;\le\; \sqrt{\lambda_{\max}(Q) \sum_{h=1}^{N_p} \big(\Gamma_h \bar{r} + \bar{e}\big)^2}. \tag{33}$$

That uniformity assumption is strong: the residuals are a property of the plant's response to counterfactual inputs, and the data only reveal them along the recorded trajectories. The theorem below therefore uses Δ as defined in (32), and (33) should be read as what the analysis yields when such a uniform bound is available.

Theorem 2 (Open-loop suboptimality of the certainty-equivalent input)**.** *Let* $U^\star \in \arg\min_{U \in \mathcal{U}} \hat{J}(U)$ *be the sequence returned by the QP* (12) *and* $U^{\mathrm{opt}} \in \arg\min_{U \in \mathcal{U}} J(U)$ *the sequence a controller with perfect predictions would choose. Then,*

$$\sqrt{J(U^\star)} \;\le\; \sqrt{J(U^{\mathrm{opt}})} + 2\Delta. \tag{34}$$

Proof. Collect the three cost contributions of (31) into a single vector,

$$\hat{v}(U) = \begin{bmatrix} \bar{Q}^{1/2}(\hat{Y}(U) - R) \\ \bar{R}^{1/2}U \\ \bar{S}^{1/2}(DU - s_0) \end{bmatrix}, \qquad v(U) = \begin{bmatrix} \bar{Q}^{1/2}(Y(U) - R) \\ \bar{R}^{1/2}U \\ \bar{S}^{1/2}(DU - s_0) \end{bmatrix}, \tag{35}$$

so that $\hat{J}(U) = \|\hat{v}(U)\|_2^2$ and $J(U) = \|v(U)\|_2^2$. The two vectors differ only in their first block; hence, $\|\hat{v}(U) - v(U)\|_2 = \|\hat{Y}(U) - Y(U)\|_{\bar{Q}} \leq \Delta$ for every $U \in \mathcal{U}$, and the reverse triangle inequality gives $\left|\|\hat{v}(U)\|_2 - \|v(U)\|_2\right| \leq \Delta$. Therefore,

$$\sqrt{J(U^\star)} \leq \sqrt{\hat{J}(U^\star)} + \Delta \leq \sqrt{\hat{J}(U^{\mathrm{opt}})} + \Delta \leq \sqrt{J(U^{\mathrm{opt}})} + 2\Delta, \tag{36}$$

where the middle inequality is the optimality of $U^\star$ for $\hat{J}$ over $\mathcal{U}$, which contains U^{opt}. □

Theorem 2 is what ties the estimator to the controller: the loss incurred by using an approximate predictor is controlled by Δ, which is a weighted horizon prediction error of the same kind that (24) minimises. The two are not the same functional—Δ is a worst case over admissible input sequences with the weights Q, whereas (24) is an empirical average over recorded windows with the weights w_h—but both are controlled by the product $\Gamma_h \bar{r}$ of (16), and reducing $\bar{r}$ without controlling Γ_h, which is what the one-step criterion does, need not reduce either.

Two caveats should be stated plainly. First, (34) is an open-loop, per-instant statement; the recursive-feasibility and stability questions for Koopman MPC require additional structure (terminal ingredients or cost controllability) and are treated in [35–37], building on the input-to-state-stability view of inherently robust MPC [88]. Second, Δ as defined in (32) is a supremum over admissible input sequences, so the bound is conservative whenever the realised inputs stay in a small part of $\mathcal{U}$; the experiments in Section 7 therefore report measured closed-loop cost rather than the bound.

6. Experimental Setup

6.1. Benchmarks

All identification experiments use physically measured records released for public benchmarking; no synthetic data enter the identification or the reported prediction results. Table 1 summarises them.

Table 1. Measured benchmarks used in this study. Every record is a physical measurement released for public benchmarking; no synthetic data enter the identification experiments. Sample counts are after the fixed per-plant decimation described in Section 6.

Benchmark	Inputs	Outputs	Train Samples	Test Samples	1/Δ*t* (Hz)	Input/Output Units
Industrial robot	6	6	39,988	3636	10	Nm/deg
EMPS	1	1	2485	2485	100	N/m
Cascaded tanks	1	1	1024	1024	0.25	V/V
Silverbox	1	1	65,062	21,688	610.4	V/V

The *industrial robot* record is the forward-identification task of the KUKA KR300 R2500 ultra SE benchmark [89]: six measured motor torques drive six measured joint positions, sampled at 10 Hz after the filtering and decimation prescribed by the benchmark. The plant exhibits backlash in every joint, pose-dependent inertia, gravitational and hydraulic loading, and temperature-dependent friction, so it is a genuinely hard multiple-input multiple-output identification problem; the benchmark's own linear state-space baseline attains a test NRMSE of 1.1173 in simulation mode. The *EMPS* record is an electro-mechanical positioning system—the prismatic joint of a robot—driven by a measured motor force [90].

The *cascaded tanks* record is a two-tank liquid-level process with a soft saturation and a hard overflow nonlinearity [91], and the *Silverbox* record is an electronic realisation of a Duffing oscillator [92,93].

Decimation factors are fixed once per plant from its bandwidth and are identical for every estimator: the robot and the tanks are used as supplied, the EMPS is decimated from 1 kHz to a 100 Hz motion-control rate, and the Silverbox is kept at its full rate because its resonance lies close to the Nyquist frequency of any decimated rate.

6.2. Leakage Control in the Robot Record

The robot benchmark concatenates 33 trajectories, each executed twice. Recovering the segment boundaries from the homing pose shows 66 segments organised in blocks of six, with the second execution of segment i appearing as segment $i + 3$; the median correlation between an execution and its partner is 1.0000. A naive contiguous or random split would therefore place near duplicates on both sides of the fit/validation boundary. All robot splits below cut only at multiples of six segments, so both executions of every trajectory stay on the same side.

6.3. Protocol

The lifted state is the delay embedding (2) followed by a dictionary; the delay order and dictionary are selected once per benchmark on the rolled-out validation criterion, from a grid of nine candidates spanning delay orders $\{3, 5, 10, 20\}$, no dictionary, random Fourier features with 50–200 components, and 200 random quadratic monomials. Inputs and outputs are standardised with training-set statistics; all reported errors are converted back to physical units.

Model selection uses a rotating validation fold: with $K = 5$ folds, seed s holds out the s-th contiguous block of the training record (snapped to trajectory-group boundaries for the robot) and fits on the remainder. This makes the reported spread a genuine measure of sensitivity to the fit/validation split rather than an artefact of a random dictionary draw. The ridge parameter λ is chosen from a nine-point logarithmic grid on the *same* rolled-out criterion for every estimator, including the one-step baselines, so the baselines are tuned for exactly the figure of merit on which they are scored.

Figures of merit follow the industrial-robot benchmark [89]. For channel n of a record of length K,

$$\mathrm{NRMSE}_n = \frac{1}{\sigma_n}\sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(y_{n,k} - \hat{y}_{n,k}\right)^2}, \qquad R_n^2 = 100\left(1 - \frac{\sum_k (y_{n,k} - \hat{y}_{n,k})^2}{\sum_k (y_{n,k} - \bar{y}_n)^2}\right), \tag{37}$$

with σ_n and $\bar{y}_n$ the standard deviation and mean of the measured channel; both are averaged over channels. We report two variants. NRMSE_h is the h-step-ahead error obtained by re-initialising the predictor at every measured sample and rolling forward h steps—the quantity MPC consumes. $\mathrm{NRMSE}_{\mathrm{sim}}$ is the free-run error over the entire test record from a single initial delay state, which is the benchmark's own simulation-mode figure of merit. A free run that returns a non-finite value is recorded as diverged; no run reported here did so, and the free-run failures discussed below are finite values that leave the data range by many orders of magnitude.

6.4. Estimators Compared

- **EDMDc**: The closed-form one-step estimator (8).
- **EDMDc + SR**: The same estimator followed by the spectral-radius retraction (28) with $\bar{\rho} = 1$. The returned matrix satisfies $\rho(A) \leq 1$; how this relates to the condition of

Corollary 1 is explained in the remark following that corollary in Section 3, and results for $\bar{\rho} < 1$ are reported in Section 7.4.

- **BR-EDMDc**: Minimisation of the bound surrogate (29).
- **HC-EDMDc (no SR)**: The rolled-out objective (24) without the stability constraint, included to separate the two ingredients.
- **HC-EDMDc**: The proposed estimator (26), with the same stability radius $\bar{\rho} = 1$.
- **NARX**: A feed-forward network on the same delay state, trained either on the one-step or on the rolled-out criterion, as a black-box reference outside the Koopman class. Training a black-box model on a rolled-out criterion is the neural counterpart of output-error identification and has been studied in its own right [94–96].

Both constrained estimators therefore use the same stability radius $\bar{\rho} = 1$, so any difference between EDMDc + SR and HC-EDMDc comes from the objective rather than from the constraint level.

Setting the dictionary to the identity recovers a linear model in delay coordinates, which is the regression form of classical subspace identification [97]; it is included in the dictionary grid and is selected on two of the four benchmarks.

The rolled-out objective is minimised in double precision with Adam on mini-batches of 2048 windows, started from the closed-form solution (8) and run for at most 150 epochs at a base learning rate of 10^{-3}. Three details of the schedule matter for reproducing the numbers. The horizon follows a doubling curriculum, training at $H = 1, 2, 4, \ldots$ before the full horizon, which keeps the early iterates from being dominated by the deepest compositions. Feasibility is restored by the retraction (28) every ten optimiser steps rather than at every step, and once more at the end, so the returned iterate is always feasible. Gradients are norm-clipped at one, the learning rate is halved after twelve epochs without improvement, and training stops early on the held-out rolled-out loss with a patience of forty epochs; the selected epoch is the best one seen on that held-out loss, which is what makes Proposition 3 hold by construction. Experiments run on an NVIDIA RTX 5080 GPU; wall-clock fitting times are reported in Table 2.

6.5. Closed-Loop Study

No physical hardware was available for this study, so the controllers are evaluated against first-principles plant models whose parameters were identified from the same measured records and whose open-loop fits on the held-out measurements are reported. The EMPS positioning system uses the standard rigid-joint model

$$M\ddot{q} + F_v\dot{q} + F_c\,\mathrm{sign}(\dot{q}) + \tau_{\mathrm{off}} = \tau, \tag{38}$$

whose four parameters are obtained by least squares on the measured torque and the finite-difference velocity and acceleration; on the held-out measured record, it reproduces the position with $R^2 = 98.7\,\%$. The cascaded tanks use the two-state square-root outflow model

$$\dot{x}_1 = -k_1\sqrt{x_1} + k_4 u, \qquad \dot{x}_2 = k_2\sqrt{x_1} - k_3\sqrt{x_2}, \tag{39}$$

with an overflow saturation and the four constants fitted by minimising the open-loop simulation error; it reaches $R^2 = 91.0\,\%$ on the held-out record.

The predictors in the loop are fitted only on the measured records and never see data from the simulated plants. Even the zero-mismatch level therefore contains a structural plant–model mismatch between the measured apparatus and its first-principles surrogate, of the size indicated by the open-loop fits above. The 10 and 20 % levels move the plant further from the apparatus on which the predictors were based by perturbing its dominant physical constants. On the EMPS, the inertia M and the viscous friction F_v are inflated

together. On the tanks, the upper-tank outflow coefficient is raised and the pump gain lowered by the same fraction, $k_1(1+\delta)$ and $k_4(1-\delta)$ with $\delta \in \{0, 0.1, 0.2\}$. Results split by mismatch level are given in Section 7.5.

Table 2. Prediction accuracy on the measured test records, mean $\pm$ standard deviation over five validation folds. NRMSE_h is the h-step-ahead error from every test sample; $\mathrm{NRMSE}_{\mathrm{sim}}$ is the free-run error over the whole test record. A superscript counts folds whose free run diverged, and "div." marks a row in which every fold diverged. SR denotes the spectral-radius retraction.

Estimator	$\mathrm{NRMSE}_{h=5}$	$\mathrm{NRMSE}_{h=20}$	$\mathrm{NRMSE}_{h=50}$	$\mathrm{NRMSE}_{\mathrm{sim}}$	$\rho(A)$	Γ_{20}	t_{fit} (s)
Industrial robot							
EDMDc	0.0008	0.0423 ± 0.0009	0.7835 ± 0.0284	2.045 ± 0.192	1.0002	21.6	0.6
EDMDc + SR	0.0018 ± 0.0013	0.0433 ± 0.0013	0.7782 ± 0.0308	1.914 ± 0.099	0.9999	21.6	0.6
BR-EDMDc	0.0018 ± 0.0013	0.0433 ± 0.0013	0.7782 ± 0.0308	1.914 ± 0.099	0.9999	21.6	3.4
HC-EDMDc (no SR)	0.0049 ± 0.0042	0.0442 ± 0.0047	0.8932 ± 0.1336	2.1×10^{17}	1.0010	21.6	25.5
HC-EDMDc	0.0018 ± 0.0013	0.0433 ± 0.0013	0.7782 ± 0.0308	1.914 ± 0.099	0.9999	21.6	28.4
EMPS							
EDMDc	0.0013 ± 0.0002	0.0166 ± 0.0023	0.0666 ± 0.0060	0.090 ± 0.020	0.9980	20	0.4
EDMDc + SR	0.0016 ± 0.0006	0.0168 ± 0.0022	0.0669 ± 0.0058	0.090 ± 0.020	0.9979	20	0.5
BR-EDMDc	0.9904 ± 0.0037	0.9895 ± 0.0020	0.9919 ± 0.0022	0.988 ± 0.002	0.9177	2.04	5.0
HC-EDMDc (no SR)	0.0013 ± 0.0002	0.0166 ± 0.0023	0.0666 ± 0.0060	0.090 ± 0.020	0.9980	20	2.5
HC-EDMDc	0.0016 ± 0.0006	0.0168 ± 0.0022	0.0669 ± 0.0058	0.090 ± 0.020	0.9979	20	4.4
Cascaded tanks							
EDMDc	0.1332 ± 0.0266	0.3164 ± 0.0529	0.4430 ± 0.0550	1.5×10^{8}	1.0084	19.7	0.0
EDMDc + SR	0.1471 ± 0.0293	0.3431 ± 0.0730	0.4971 ± 0.1043	0.443 ± 0.159	1.0000	18.5	0.0
BR-EDMDc	0.8150 ± 0.3691	0.8449 ± 0.3111	0.8711 ± 0.2585	0.851 ± 0.292	0.9835	7.08	2.6
HC-EDMDc (no SR)	0.1222 ± 0.0257	0.2762 ± 0.0186	0.4094 ± 0.0451	0.359 ± 0.029	0.9834	20	1.2
HC-EDMDc	0.1146 ± 0.0200	0.2827 ± 0.0424	0.3952 ± 0.0527	0.352 ± 0.027	0.9839	20	1.2
Silverbox							
EDMDc	0.0314 ± 0.0001	0.1027 ± 0.0003	0.1344 ± 0.0004	0.137 ± 0.001	0.9674	14.4	0.1
EDMDc + SR	0.0314 ± 0.0001	0.1027 ± 0.0003	0.1344 ± 0.0004	0.137 ± 0.001	0.9674	14.4	0.1
BR-EDMDc	0.8623 ± 0.0413	0.8808 ± 0.0159	0.8809 ± 0.0159	0.881 ± 0.016	0.9013	1.97	4.3
HC-EDMDc (no SR)	0.0310 ± 0.0004	0.0976 ± 0.0002	0.1268 ± 0.0004	1.2×10^{21}	0.9755	14.5	27.3
HC-EDMDc	0.0338 ± 0.0002	0.0985 ± 0.0001	0.1263 ± 0.0003	0.129	0.9662	14.4	27.3

The two plants play different roles. On the EMPS record, all estimators return the same predictor, so that loop is a consistency check, whereas the tank record is the one on which the predictors genuinely differ. The coarser tank surrogate is a limitation of the second experiment and is discussed in Section 8.5.

The controller is the QP (12) solved with OSQP [3]. Its weights are fixed once on the one-step baseline and held identical for every predictor, so that differences in tracking reflect the predictor rather than retuning. A textbook constant output-disturbance estimator provides offset-free tracking, and the input box constraints are taken from the measured input range. Three reference profiles (square wave, sinusoid, smoothed multi-level set-point sequence), three prediction horizons ($N_p \in \{25, 50, 100\}$ on the EMPS, $N_p \in \{5, 10, 20\}$ on the tanks) and the three mismatch levels are combined with three validation folds, giving 81 runs per estimator and plant, 27 at each mismatch level.

7. Results

7.1. Prediction Accuracy

Table 2 and Figure 2 report the test accuracy of the five estimators on the four measured records. Three observations organise the rest of this section.

The one-step estimator is not Schur stable. The closed-form estimator (8) returns $\rho(A) \geq 1$ on 8 of the 20 fits, spread over 3 of the 4 benchmarks, even though its ridge parameter was selected on a rolled-out criterion. We count individual fits rather than fold-

averaged spectral radii because averaging first conceals a record on which only one fold leaves the unit disc. On the two records whose fold-mean radius also exceeds one, the free-run simulation leaves the data range: on the cascaded tanks the free-run NRMSE of EDMDc is 1.5×10^8 against 0.443 once the retraction (28) is applied, and the same comparison on the robot record reads 2.045 against 1.914. The retraction costs one eigendecomposition each time it is applied—once per ridge candidate during selection, and periodically during the rolled-out fit—and it is not free in accuracy. At $h = 20$, it raises the test error by 8.4 % on the tanks and 2.4 % on the robot; at $h = 1$, where the unconstrained fit comes closest to interpolating the record, it costs 706.2 % on the robot and 197.9 % on EMPS. On Silverbox, where $\rho(A) < 1$ already holds, it changes nothing. The trade is therefore up to 8.4 % of twenty-step accuracy, and much more at $h = 1$, against a free run that otherwise leaves the data range by eight orders of magnitude, and we take it as the default for a predictor that will be rolled out over an MPC horizon—not for one that will only ever be used a step at a time. The ablation *HC-EDMDc (no SR)* shows that this is not an artefact of the one-step criterion: removing the constraint from the horizon-consistent estimator also produces free runs that leave the numerical range on the robot and Silverbox records.

The horizon-consistent fit helps on two of the four records, and the separator is not the residual. On the cascaded tanks, HC-EDMDc improves the twenty-step error from 0.3431 to 0.2827 and the free-run error from 0.443 to 0.352. On Silverbox, the twenty-step error moves from 0.1027 to 0.0985. On the robot and EMPS records, the refinement returns its warm start: on every fold of those two records, no iterate generated by the optimiser improved on the warm start's held-out rolled-out loss, so the selection rule of Proposition 3 returned it. That is a statement about the finite set of candidates the optimiser produced under the schedule of Section 6, not a proof that the warm start is a stationary point or a minimiser of (24); we did not test for either.

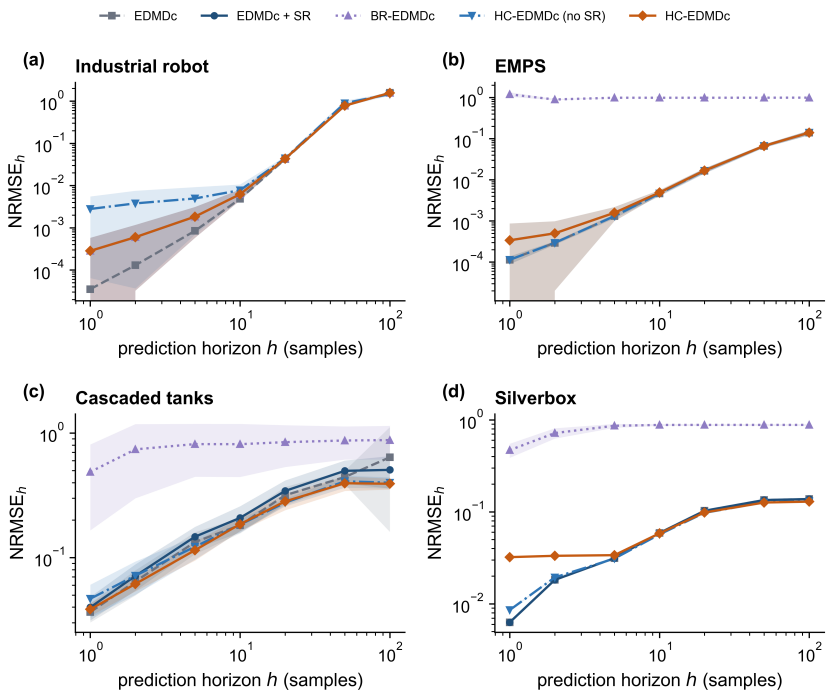


Figure 2. Test error as a function of the prediction horizon h, mean over five validation folds with the shaded band showing one standard deviation. Both axes are logarithmic. The curves for BR-EDMDc sit at the level of a mean-reverting predictor on three of the four records.

It is tempting to summarise this as "the refinement helps where the one-step residual is not negligible", and we did so in an earlier draft, but the data do not support it. The

root-mean-square lifted residual ε of the one-step fit is 1.90 on the robot, 6.29 on EMPS, 2.88 on the tanks and 0.99 on Silverbox: the largest of the four belongs to EMPS, which is one of the two neutral records. ε is in any case not comparable across these records, since it is measured in the lifted coordinates and those differ in dimension and in output scaling. What separates the regimes here is the observable stated above—whether the refinement generates any candidate that improves on its warm start—and we have not established what determines that in advance. Across the paired fold-by-fold comparison of HC-EDMDc against EDMDc + SR, the proposed estimator is no worse on 19 of 20 pairs at $h = 20$: 9 improvements, 10 exact ties contributed by the two records on which the refinement returns its warm start, and 1 regression. The ties pin the median relative change over all pairs to 0.0 %; over the 10 pairs that actually differ, it is 4.4 %. At $h = 50$, the same comparison gives 10 improvements, 10 ties and 0 regressions, with a median of 6.1 % over the differing pairs. These counts are descriptive and we attach no significance test to them: the folds of a record share one held-out test trajectory and largely overlapping training data, so they are not independent replicates, and pooling absolute NRMSE differences across four records of different difficulty would not be testing a single effect.

Where short horizons pay for long ones. With uniform horizon weights, the rolled-out criterion spreads its attention over $h = 1, \ldots, H$, so nothing obliges it to match the one-step estimator at $h = 1$. That trade-off materialises on one of the four records. On Silverbox, the one-step error of HC-EDMDc is 5.1 times that of EDMDc + SR, the two curves cross between $h = 5$ and $h = 20$, and beyond the crossing, the rolled-out fit is the more accurate of the two. On the cascaded tanks, no such price is paid: the rolled-out fit is at least as accurate at every horizon tested, $h = 1$ included. On the robot and EMPS records, the two estimators coincide. Figure 2 shows all four profiles. For an MPC application, the relevant question is therefore not which estimator is better in general but which horizon the controller uses; Section 7.5 takes that question up directly.

7.2. The Certified Bound Does Not Rank Models

Theorem 1 is a valid bound—we verified it numerically on random instances and it is attained in the worst case for single-output systems—so it is natural to ask whether it can be used to rank models. It cannot, and the reason is instructive.

Which quantity is being ranked matters here, and we distinguish two. The certified root-mean-square bound of (18) is $\sqrt{H\Gamma_H^{(2)}}\,\varepsilon_H + \bar{e}_H$, and we evaluate it as stated: with the window-shifted residual ε_H of (17) over the admissible windows, not with the fitting-set residual ε. The two differ by at most a factor 1.017 on these records, and $\bar{e}_H$ is at the level of 10^{-6} because the readout is exact for the lift (4), but the certificate is stated in terms of ε_H and reporting it with ε would understate it. The bare product $\Gamma_H\varepsilon$ that BR-EDMDc minimises in (29) is not a certificate at all: it pairs the smaller constant $\Gamma_H \le \sqrt{H\Gamma_H^{(2)}}$ with a different residual statistic and omits $\bar{e}_H$. We therefore report the certified bound in Table 3 and treat $\Gamma_H\varepsilon$ as the surrogate it is.

Figure 3 traces the ridge path: for each benchmark the regularisation parameter λ is swept over nine decades while everything else is held fixed. Panel (a) shows the trade-off the surrogate is made of. Increasing λ damps the transition matrix and drives Γ_{20} down—by a factor of 7 to 10 depending on the record, to values near $\|C\|_2$—while the one-step residual ε moves comparatively little over most of the path and then rises at the damped end. The certified bound is consequently not monotone in λ: it decreases along the path on EMPS and the tanks and turns back upwards on the robot and Silverbox, where a rising ε overtakes the falling amplification. Panel (b) shows the realised error over the same path, which rises throughout.

The log–log correlation between the certified bound and the twenty-step test error is strongly negative on 2 of the 4 records (EMPS -0.90, cascaded tanks -0.93), essentially absent on Silverbox (-0.04) and positive on the robot ($+0.46$). For the surrogate $\Gamma_H\varepsilon$, the picture is more uniformly negative—3 of 4 records, Silverbox reaching -0.92—which is worth stating because it is the surrogate, not the certificate, that an optimiser would be handed. On no record does either quantity rank models the way a selection criterion needs to, and on two of them, ranking by the bound would prefer the model that predicts worst. Table 3 gives the path in full.

Table 3. Ridge path with everything else held fixed, mean over three validation folds. The fourth column is the certified root-mean-square bound of Theorem 1, evaluated with the overlap-weighted residual ε_H of (17); each entry is the mean of the per-fold bound, not a bound formed from averaged factors. The column $\rho(A)$ before SR is the spectral radius of the closed-form estimate before the retraction is applied.

λ	ε	Γ_{20}	$\sqrt{H\Gamma_{20}^{(2)}}\,\varepsilon_H + \bar{e}_H$	$\rho(A)$ Before SR	$\mathrm{NRMSE}_{h=20}$
Industrial robot					
1×10^{-10}	1.9	21.6	41	1.0002	0.0427
1×10^{-8}	1.9	21.6	41	1.0002	0.0427
1×10^{-6}	81.2	5.34	640	1.2761	0.9999
1×10^{-5}	92.8	3.54	587	1.5938	1.0026
0.0001	21.6	3.44	140	1.4912	0.9899
0.001	3.05	20.3	62.1	0.9956	0.2865
0.01	3.1	18.6	57.8	0.9869	0.5525
0.1	3.32	9.64	36.6	0.9062	1.1122
1	7.32	2.12	40.6	0.4984	1.0246
EMPS					
1×10^{-10}	6.32	20.1	127	0.9973	0.0164
1×10^{-8}	6.32	20.1	127	0.9973	0.0164
1×10^{-6}	6.32	20.1	127	0.9973	0.0164
1×10^{-5}	6.32	20.1	127	0.9973	0.0163
0.0001	6.32	20.1	127	0.9972	0.0163
0.001	6.32	19.9	126	0.9971	0.0311
0.01	6.34	18.1	115	0.9894	0.1992
0.1	6.48	9.24	68.1	0.9086	0.8415
1	9.18	1.99	47.4	0.4995	0.9893
Cascaded tanks					
1×10^{-10}	3.93	19.4	78.6	1.0090	0.3623
1×10^{-8}	3.92	19.4	78.1	1.0090	0.3623
1×10^{-6}	3.36	18.5	62.7	1.0090	0.3619
1×10^{-5}	2.93	18	53.6	1.0090	0.3587
0.0001	2.9	17.9	52.7	1.0086	0.3894
0.001	2.9	17.9	52.6	1.0066	0.5304
0.01	2.9	17.1	50.2	0.9963	0.5316
0.1	2.97	9.19	31.4	0.9082	0.8409
1	4.49	2	23.5	0.4991	0.9927
Silverbox					
1×10^{-10}	0.989	14.4	14.5	0.9674	0.1029
1×10^{-8}	0.989	14.4	14.5	0.9674	0.1029
1×10^{-6}	0.989	14.4	14.5	0.9674	0.1029
1×10^{-5}	0.989	14.4	14.5	0.9674	0.1029
0.0001	1.16	8.89	11.8	1.0680	0.5087
0.001	2.28	3.9	15.1	1.2980	0.8082
0.01	2.15	3.41	13.5	1.3451	0.8319
0.1	1.42	7.5	13.1	0.8901	0.6619
1	3.35	1.93	17.1	0.4939	0.9368

This is Proposition 5 seen along a continuum rather than at a single optimum. The bound is worst case over residual *directions*, and its amplification factor is minimised exactly by the damped models that have no dynamics left to amplify. Panel (c) makes the mechanism visible on the tank record: at $\lambda = 10^{-10}$, the gain profile $\|CA^j\|_2$ stays near unity out to $j = 40$, whereas at $\lambda = 10^{-1}$, it decays geometrically—small Γ, and a model that has forgotten its input by the time the horizon ends.

One use of Γ_h does survive. It is the quantity that grows without bound in h once an unstable mode is observable through C, and the raw spectral radius along the ridge path reaches 1.5938 on the robot, 1.0090 on the tanks and 1.3451 on Silverbox at some λ, so the constraint of Section 4 is doing real work at exactly those settings. It does *not* explain which records the refinement helps on: the two neutral records carry the two largest warm-start values, $\Gamma_{20} = 21.6$ on the robot and 20.0 on EMPS, against 18.5 and 14.4 on the two active ones.

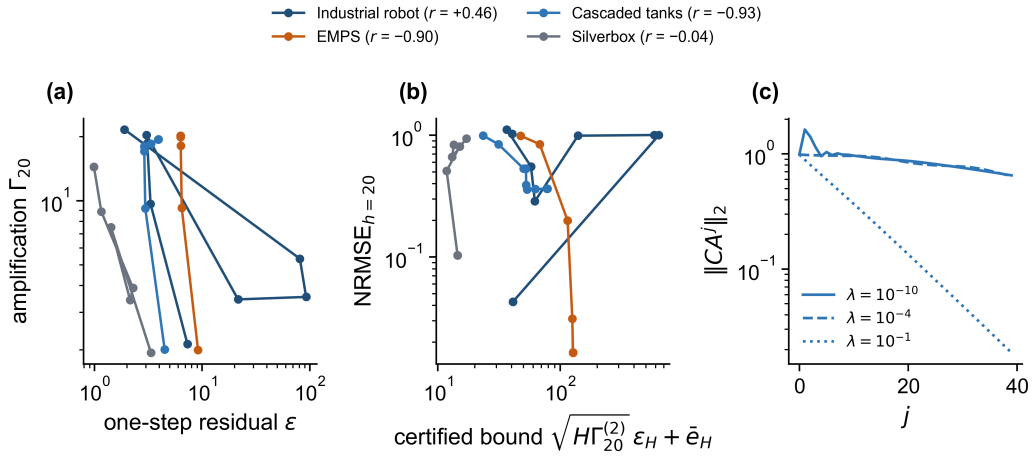


Figure 3. Behaviour along the ridge path, mean over three validation folds; the parameter λ is swept over nine decades with everything else fixed. (**a**) One-step residual against amplification factor—the two factors of the surrogate. (**b**) Certified root-mean-square bound $\sqrt{H\Gamma_{20}^{(2)}}\,\varepsilon_H + \bar{e}_H$ against the realised twenty-step test error, with the log–log correlation along each path. (**c**) Transient gain profiles $\|CA^j\|_2$ on the cascaded-tank record at three values of λ.

Figure 4 makes the consequence concrete on the robot record: the free-run simulation of all six joints over the whole six-minute test trajectory, from a single initial delay state.

7.3. Minimising the Surrogate Does Not Work

Section 7.2 showed that neither the certified bound nor its surrogate ranks models along a regularisation path. Optimising the surrogate directly is the same phenomenon taken to its conclusion. On 3 of the 4 records, BR-EDMDc moves away from its warm start and reduces Γ_{20} from 14.4–20.0 to 2.0–7.1, exactly as its objective asks, and in doing so destroys the predictor: its twenty-step errors in Table 2 land within a factor of two of a model that predicts the channel mean, whose NRMSE is one by the normalisation used here. This is Proposition 5 in action, and it is worth reporting because minimising a certificate—or, as here, the cheap surrogate standing in for one—is an attractive shortcut that avoids the non-convex rolled-out recursion entirely.

The robot record is the exception, and we report it as one rather than averaging it away. On that record, no iterate of the BR-EDMDc fit improved on its own objective, the best-iterate rule returned the closed-form warm start, and the corresponding row of Table 2 therefore coincides with EDMDc + SR at $\Gamma_{20} = 21.6$. That record carries the largest lifted state in the study, $N = 240$. We read the outcome as a failure of that optimisation

to make progress on the largest instance, not as evidence that minimising the surrogate is harmless: the ridge path of Section 7.2, which lowers the same quantity by a different route, is anti-correlated with the realised error on two of the other three records.

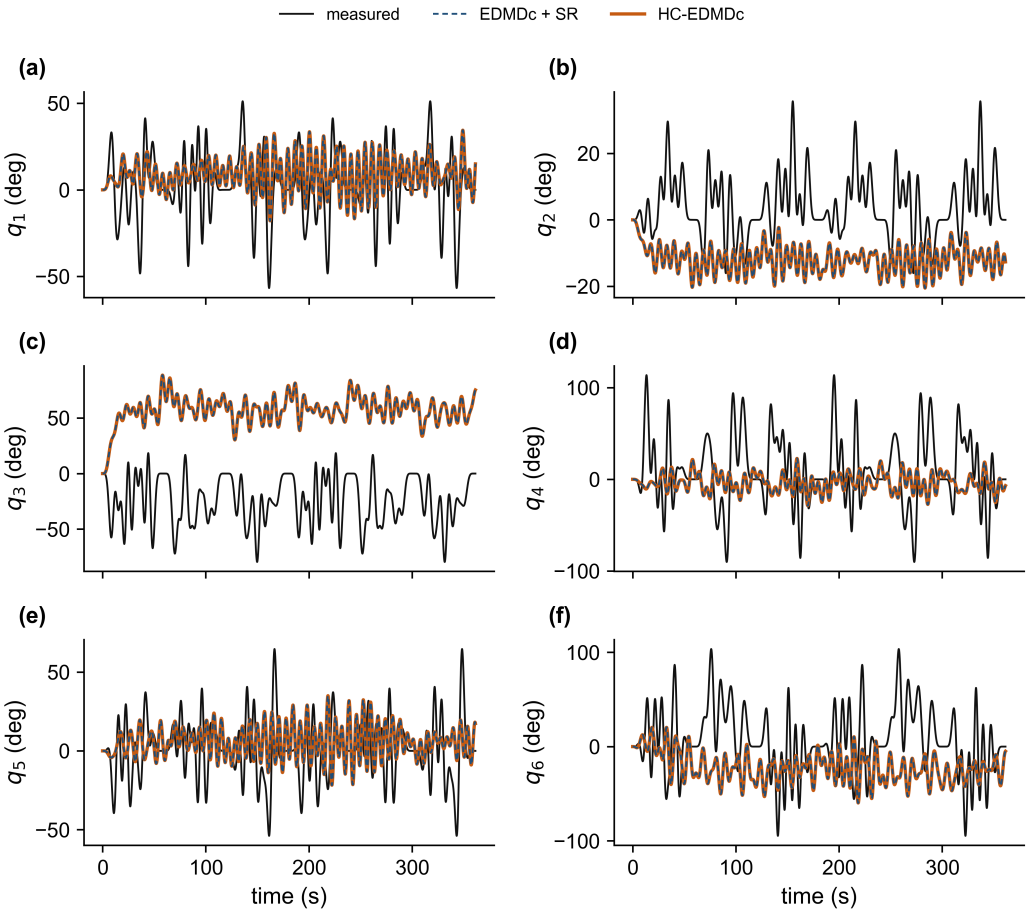


Figure 4. Free-run simulation of the six measured joint positions of the KUKA KR300 test record from a single initial delay state, for the one-step estimator with the stability retraction and for the proposed estimator. Panels (**a**–**f**) are joints q_1 to q_6.

7.4. Ablations

Table 4 varies the training horizon H. The pattern follows Proposition 2: on the benchmarks where the refinement is active, accuracy at $h = 20$ improves as H approaches the evaluation horizon and saturates beyond it, while on the benchmarks where the one-step fit is already near-exact, the choice of H is immaterial.

Table 4. Effect of the training horizon H on the 20-step test error (mean ± standard deviation over three folds). The last column repeats the free-run error of the $H = 20$ models for reference.

Benchmark	Estimator	$H = 5$	$H = 10$	$H = 20$	$H = 50$	$\mathrm{NRMSE}_{\mathrm{sim}}$ ($H = 20$)
Industrial robot	EDMDc + SR	0.0427 ± 0.0008	0.0427 ± 0.0008	0.0427 ± 0.0008	0.1199 ± 0.1329	1.942 ± 0.128
Industrial robot	HC-EDMDc	0.0427 ± 0.0008	0.0427 ± 0.0008	0.0427 ± 0.0008	0.1199 ± 0.1329	1.942 ± 0.128
EMPS	EDMDc + SR	0.0164 ± 0.0027	0.0164 ± 0.0027	0.0164 ± 0.0027	0.0164 ± 0.0027	0.088 ± 0.027
EMPS	HC-EDMDc	0.0164 ± 0.0027	0.0164 ± 0.0027	0.0164 ± 0.0027	0.0164 ± 0.0027	0.088 ± 0.027
Cascaded tanks	EDMDc + SR	0.4024 ± 0.1382	0.3869 ± 0.1361	0.3595 ± 0.0938	0.3614 ± 0.0997	0.446 ± 0.197
Cascaded tanks	HC-EDMDc	0.3582 ± 0.0662	0.3203 ± 0.0728	0.3085 ± 0.0309	0.3436 ± 0.0915	0.350 ± 0.037
Silverbox	EDMDc + SR	0.1029 ± 0.0003	0.1029 ± 0.0003	0.1029 ± 0.0003	0.1029 ± 0.0003	0.138
Silverbox	HC-EDMDc	0.1029 ± 0.0003	0.1029 ± 0.0003	0.0984 ± 0.0001	0.0981 ± 0.0002	0.129

Table 5 varies the stability radius $\bar{\rho}$, and the pattern is more heterogeneous than the constraint's role would suggest. Tightening from $\bar{\rho} = 1$ to 0.99 is not free: on EMPS, the free-run error rises from 0.088 to 0.664 and on the tanks from 0.446 to 0.553, because a strict radius damps modes the record actually needs. Relaxing to $\bar{\rho} = 1.001$ does not produce a matching collapse either—the tank free run is unchanged to three digits and Silverbox and EMPS are unaffected, since their unconstrained radii already lie below the threshold—and only the robot record degrades, from 1.942 to 2.014. The reading we take from this is narrow: $\bar{\rho}$ is a feasibility switch rather than a performance knob, and values just below one are the ones that leave the fit alone while still bounding the free run. Note that $\bar{\rho} = 1$, used throughout the main experiments, is enough for Γ_h to be finite at every finite h but not for the horizon-*uniform* certificate of Corollary 1, which requires $\rho(A) < \bar{\rho} < 1$; the $\bar{\rho} \in \{0.99, 0.999, 0.9999\}$ rows of this table are the ones for which that certificate applies.

Table 5. Free-run test error as a function of the stability radius $\bar{\rho}$ imposed by the retraction (mean ± standard deviation over three folds).

Benchmark	Estimator	0.99	0.999	0.9999	1	1.001
Industrial robot	EDMDc + SR	1.636 ± 0.073	1.798 ± 0.103	1.923 ± 0.125	1.942 ± 0.128	2.014 ± 0.175
Industrial robot	HC-EDMDc	1.636 ± 0.073	1.798 ± 0.103	1.923 ± 0.125	1.942 ± 0.128	2.014 ± 0.175
EMPS	EDMDc + SR	0.664 ± 0.066	0.095 ± 0.017	0.088 ± 0.027	0.088 ± 0.027	0.088 ± 0.027
EMPS	HC-EDMDc	0.664 ± 0.066	0.095 ± 0.017	0.088 ± 0.027	0.088 ± 0.027	0.088 ± 0.027
Cascaded tanks	EDMDc + SR	0.553 ± 0.173	0.451 ± 0.201	0.446 ± 0.197	0.446 ± 0.197	0.443 ± 0.191
Cascaded tanks	HC-EDMDc	0.485 ± 0.232	0.351 ± 0.039	0.350 ± 0.034	0.350 ± 0.037	0.364 ± 0.030
Silverbox	EDMDc + SR	0.138	0.138	0.138	0.138	0.138
Silverbox	HC-EDMDc	0.129	0.129	0.129	0.129	0.129

Tables 6 and 7 were run to test the regime dependence that Proposition 2 suggests, and they do not confirm it. Adding output noise to the training record does not widen the gap in favour of the horizon-consistent fit. On the cascaded tanks—the one record where the refinement is active at every noise level—the twenty-step advantage is largest on the clean record at 14.2 % and falls to 1.0–7.8 % once noise is added. On Silverbox, an advantage of 4.3 % on the clean record disappears as soon as noise is present. On EMPS at the largest noise level tested, a relative standard deviation of 0.1, the gap turns negative at −6.4 %: the refined model is worse than its warm start on the test record, which Proposition 3 permits, since the guarantee is on the held-out criterion the model is selected on and not on test error at $h = 20$. The training-size sweep is likewise non-monotone; on Silverbox the advantage *grows* with record length, from 0.6 % at a tenth of the record to 4.3 % at the full record, which is the opposite of the predicted direction.

Table 6. Twenty-step test error when zero-mean Gaussian noise of the stated relative standard deviation is added to the training outputs (mean ± standard deviation over three folds).

Benchmark	Estimator	0	0.01	0.02	0.05	0.1
Industrial robot	EDMDc + SR	0.0427 ± 0.0008	0.4652 ± 0.0047	0.5558 ± 0.0080	0.6590 ± 0.0086	0.8170 ± 0.0132
Industrial robot	HC-EDMDc	0.0427 ± 0.0008	0.4652 ± 0.0047	0.5558 ± 0.0080	0.6590 ± 0.0086	0.8170 ± 0.0132
EMPS	EDMDc + SR	0.0164 ± 0.0027	0.0402 ± 0.0035	0.0573 ± 0.0018	0.0678 ± 0.0033	0.0763 ± 0.0051
EMPS	HC-EDMDc	0.0164 ± 0.0027	0.0402 ± 0.0035	0.0573 ± 0.0018	0.0678 ± 0.0033	0.0812 ± 0.0092
Cascaded tanks	EDMDc + SR	0.3595 ± 0.0938	0.3378 ± 0.0581	0.3143 ± 0.0081	0.3845 ± 0.0938	0.3814 ± 0.0944
Cascaded tanks	HC-EDMDc	0.3085 ± 0.0309	0.3127 ± 0.0679	0.3111 ± 0.0190	0.3546 ± 0.0681	0.3552 ± 0.0687
Silverbox	EDMDc + SR	0.1029 ± 0.0003	0.1009 ± 0.0001	0.1023 ± 0.0001	0.1043 ± 0.0002	0.1083 ± 0.0002
Silverbox	HC-EDMDc	0.0984 ± 0.0001	0.1009 ± 0.0001	0.1022 ± 0.0002	0.1043 ± 0.0002	0.1083 ± 0.0002

We report this as a failure of the prediction to transfer, not as evidence against Proposition 2 itself: that proposition is a population statement about a scalar autoregres-

sion, and it is verified as such against a Monte Carlo simulation in Section 3. Two features of the experiment separate it from that setting. The rolled-out criterion is evaluated on the same noisy outputs that corrupt the one-step criterion, so both objectives degrade together rather than one relative to the other; and the model is selected on a held-out loss computed on that same noisy record, which on Silverbox returns the warm start at every noise level above zero. Establishing the regime boundary that Proposition 2 points to would need a design that separates estimation noise from selection noise, which we did not run. Figure 5 summarises both sweeps together with the per-fold spread of the improvement.

Table 7. Twenty-step test error as a function of the fraction of the training record used (mean ± standard deviation over three folds).

Benchmark	Estimator	0.1	0.25	0.5	0.75	1
Industrial robot	EDMDc + SR	0.1151 ± 0.0213	0.6926 ± 0.5413	0.0454 ± 0.0005	0.0458 ± 0.0033	0.0427 ± 0.0008
Industrial robot	HC-EDMDc	0.1151 ± 0.0213	0.6926 ± 0.5413	0.0454 ± 0.0005	0.0458 ± 0.0033	0.0427 ± 0.0008
EMPS	EDMDc + SR	1.1438 ± 0.0250	0.9080 ± 0.0800	0.0163 ± 0.0032	0.0161 ± 0.0024	0.0164 ± 0.0027
EMPS	HC-EDMDc	1.1540 ± 0.0140	0.8444 ± 0.1535	0.0163 ± 0.0032	0.0161 ± 0.0024	0.0164 ± 0.0027
Cascaded tanks	EDMDc + SR	1.0018 ± 0.0010	0.8996 ± 0.1985	0.7867 ± 0.1948	0.6279 ± 0.2927	0.3595 ± 0.0938
Cascaded tanks	HC-EDMDc	2.6410 ± 2.8394	0.8294 ± 0.1780	0.9503 ± 0.1875	0.4877 ± 0.2549	0.3085 ± 0.0309
Silverbox	EDMDc + SR	0.1041 ± 0.0012	0.1026 ± 0.0001	0.1024 ± 0.0003	0.1027 ± 0.0004	0.1029 ± 0.0003
Silverbox	HC-EDMDc	0.1035 ± 0.0001	0.1026 ± 0.0001	0.0991 ± 0.0007	0.0988 ± 0.0004	0.0984 ± 0.0001

Table 8 places the comparison outside the Koopman model class. Two NARX networks with identical architecture, differing only in whether training minimises the one-step or the rolled-out output error, reproduce the same ordering as the two linear estimators, which supports the reading that the effect is a property of the training criterion rather than of the lifted representation. The networks are not uniformly more accurate than the lifted linear predictor at the horizons tested, and they forfeit the QP structure that motivates the Koopman construction in the first place.

Table 8. Neural black-box reference. The two NARX networks share an architecture and differ only in the training horizon, which isolates the effect of the rolled-out criterion outside the Koopman model class; the proposed estimator is repeated from Table 2 for orientation.

Benchmark	Model	$\mathrm{NRMSE}_{h=20}$	$\mathrm{NRMSE}_{\mathrm{sim}}$	Parameters	t_{fit} (s)
Industrial robot	NARX ($H = 1$)	1.4578 ± 0.0134	1.514 ± 0.095	65,414	11
Industrial robot	NARX ($H = 20$)	1.1696 ± 0.0594	1.210 ± 0.034	65,414	67
Industrial robot	HC-EDMDc	0.0433 ± 0.0013	1.914 ± 0.099	–	28
EMPS	NARX ($H = 1$)	0.1329 ± 0.0121	0.227 ± 0.077	35,969	1
EMPS	NARX ($H = 20$)	0.0826 ± 0.0095	0.193 ± 0.008	35,969	7
EMPS	HC-EDMDc	0.0168 ± 0.0022	0.090 ± 0.020	–	4
Cascaded tanks	NARX ($H = 1$)	0.6991 ± 0.2814	0.990 ± 0.138	34,177	0
Cascaded tanks	NARX ($H = 20$)	0.4008 ± 0.0856	0.639 ± 0.038	34,177	2
Cascaded tanks	HC-EDMDc	0.2827 ± 0.0424	0.352 ± 0.027	–	1
Silverbox	NARX ($H = 1$)	0.0190 ± 0.0053	0.991 ± 0.866	38,529	13
Silverbox	NARX ($H = 20$)	0.0103 ± 0.0002	0.011	38,529	186
Silverbox	HC-EDMDc	0.0985 ± 0.0001	0.129	–	27

7.5. Closed-Loop Tracking

Table 9 and Figure 6 report the closed-loop study described in Section 6.5. The result is essentially null, and we report it as such.

Splitting the runs by plant–model mismatch level, as in Table 10, leaves the picture unchanged. On the EMPS plant, the three predictors coincide at every level, and the tracking error grows with the mismatch, from 0.4304 at 0 % to 0.4595 at 20 %. On the

cascaded tanks, HC-EDMDc gives the lowest mean tracking NRMSE at all three levels, below EDMDc + SR by 1.4 %, 0.8 % and 0.6 % at 0 %, 10 % and 20 % mismatch respectively. Each of these differences is smaller than the standard deviation of the corresponding cell, which covers its 27 runs over three reference profiles, three prediction horizons and three validation folds. The ordering of the predictors therefore does not change with the mismatch level, so the near-zero differences in Table 9 are not produced by averaging over levels on which the predictors would separate. No run diverged at any level.

Table 9. Closed-loop tracking with the same MPC law and weights, driven by predictors from different estimators. Each row aggregates 81 runs: three reference profiles × three prediction horizons × three plant–model mismatch levels × three validation folds. The plants are first-principles models whose parameters were identified from the same measured records (Section 6.5); no hardware experiment was performed.

Plant	Predictor	Tracking NRMSE	IAE	$\overline{u^2}$	QP Time (ms)	Diverged
EMPS	EDMDc	0.4450 ± 0.0523	0.0081 ± 0.0039	2.89	0.18	0%
EMPS	EDMDc + SR	0.4450 ± 0.0523	0.0081 ± 0.0039	2.89	0.18	0%
EMPS	HC-EDMDc	0.4450 ± 0.0523	0.0081 ± 0.0039	2.89	0.18	0%
Cascaded tanks	EDMDc	1.4519 ± 0.3655	2.0110 ± 0.4910	20.8	0.08	0%
Cascaded tanks	EDMDc + SR	1.4527 ± 0.3654	2.0123 ± 0.4911	20.8	0.08	0%
Cascaded tanks	HC-EDMDc	1.4387 ± 0.3984	1.9918 ± 0.5474	20.8	0.08	0%

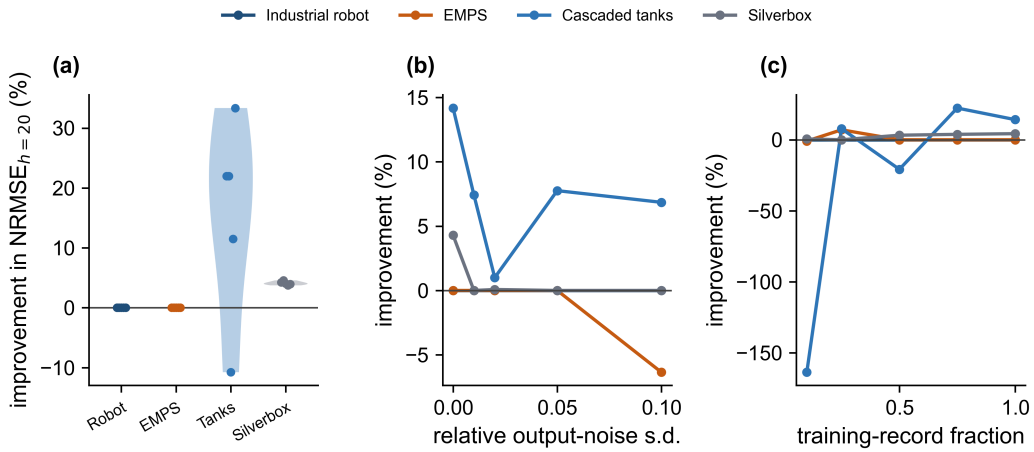


Figure 5. (**a**) Per-fold relative improvement of HC-EDMDc over EDMDc + SR at $h = 20$; violins show the distribution over validation folds and dots the individual folds. (**b**) Improvement as a function of the relative output-noise standard deviation added to the training record. (**c**) Improvement as a function of the fraction of the training record used. Values above zero favour the proposed estimator.

Table 10. Closed-loop tracking NRMSE split by plant–model mismatch level, mean ± standard deviation over the 27 runs in each cell (three reference profiles × three prediction horizons × three validation folds). The mismatch levels are defined in Section 6.5; the runs are those of Table 9, regrouped.

Plant	Predictor	0%	10%	20%
EMPS	EDMDc	0.4304 ± 0.0484	0.4451 ± 0.0505	0.4595 ± 0.0528
EMPS	EDMDc + SR	0.4304 ± 0.0484	0.4451 ± 0.0505	0.4595 ± 0.0528
EMPS	HC-EDMDc	0.4304 ± 0.0484	0.4451 ± 0.0505	0.4595 ± 0.0528
Cascaded tanks	EDMDc	1.7042 ± 0.3311	1.1093 ± 0.1286	1.5422 ± 0.2837
Cascaded tanks	EDMDc + SR	1.7043 ± 0.3310	1.1106 ± 0.1290	1.5432 ± 0.2841
Cascaded tanks	HC-EDMDc	1.6808 ± 0.3919	1.1012 ± 0.1787	1.5341 ± 0.3213

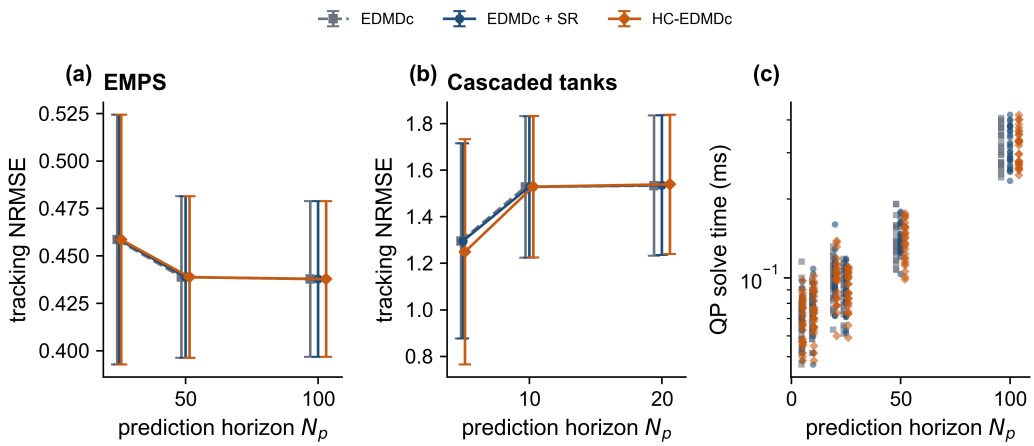


Figure 6. Closed-loop tracking, aggregated over three reference profiles, three plant–model mismatch levels and three validation folds. (**a**) Identified EMPS plant, on which the three estimators return the same predictor and the three curves coincide; (**b**) identified tank plant, on which the predictors differ; and (**c**) QP solve time per sampling instant. The three predictors are offset slightly along the horizontal axis for visibility.

On the EMPS plant the three estimators return numerically identical predictors—this is the interpolating regime of Section 7—so the three loops coincide to four decimal places. That is the expected consistency check rather than a finding. On the tank plant, where the predictors differ by roughly a fifth in twenty-step open-loop error, the mean closed-loop tracking cost differs between them by 1.0 %, and the ordering is not consistent across the three horizons: the proposed estimator is better by 3.6 % at $N_p = 5$ and worse by 0.2 % at $N_p = 20$. The mean tracking NRMSE exceeds unity on the tank plant for every controller (1.453 and 1.439), which says plainly that at a twenty-step open-loop error of 0.2827–0.3431, no predictor in this comparison is accurate enough for the horizon the controller uses.

The mechanism behind the compression is worth naming, because it qualifies Theorem 2. The bound there is per-instant and open-loop; in closed loop, the state is re-measured and the disturbance estimate re-anchored at every sample, so only the first few steps of each prediction are ever acted upon, and errors at $h \approx N_p$ are largely discarded before they matter. Feedback therefore attenuates exactly the quantity the horizon-consistent objective improves. The practical reading is that open-loop horizon accuracy is a necessary but far from sufficient proxy for closed-loop benefit, and that a closed-loop study should not be presented as a corollary of an open-loop one.

The QP itself is never the bottleneck: over every closed-loop run, the mean solve time stays below 0.42 ms and the 95th percentile reaches 1.00 ms at the longest horizon tested, $N_p = 100$, on a lifted state of a few hundred dimensions, so the entire cost of the method sits in the offline fit reported in Table 2.

8. Discussion

8.1. What the Results Say

The experiments support a narrower claim than "horizon-consistent training improves Koopman predictors". They support the statement that the horizon error of a lifted linear predictor factorises, in the sense of Theorem 1, into a residual and an amplification, and that the two factors have to be managed separately.

The amplification factor is the part that is cheap to control and expensive to ignore. Requiring $\rho(A) \leq 1$ costs one eigendecomposition per retraction and, on the records where the constraint binds, up to 8.4 % of twenty-step accuracy—considerably more at $h = 1$—and it is the difference between a predictor that can be simulated over a six-

minute record and one that cannot. It is not needed for Theorem 1 itself, which is a finite-horizon statement and holds for every A; a strict radius buys the horizon-*uniform* certificate of Corollary 1, which requires $\rho(A) < \bar{\rho} < 1$ and therefore does not apply at the $\bar{\rho} = 1$ used in the main experiments. That said, $\rho(A) \leq 1$ certifies very little on its own: the measured transient gains in Figure 3c grow by orders of magnitude before they decay, and Corollary 1 shows why—the constant M is a condition number of the Lyapunov solution, not a spectral quantity.

The residual is the part we understand less well. Proposition 2 offers a candidate mechanism. The one-step criterion shrinks the identified dynamics by a factor $(1 + \nu)^{-1}$, where ν is the noise-to-signal ratio in the regressors, and a horizon-H criterion reduces that to its H-th root, so a rolled-out fit should matter more where the one-step fit is less exact. On these four records, that reading does not survive contact with the data. The record with the largest lifted residual is EMPS, and it is one of the two on which the refinement changes nothing; and deliberately degrading the records with output noise, which by the same argument should enlarge the effect, shrank it instead (Section 7). A narrower and more mechanical observable separates the records. On the two neutral records, the optimiser generated no candidate that improved on its closed-form warm start, so the selection returned it. The warm start is not a first-order stationary point there, yet no epoch of the fit lowers either the training objective or the held-out loss, and Section 8.3 shows that this is visible within the first twenty epochs. Reporting both regimes rather than the favourable one is, we think, the useful contribution. It tells a practitioner to read the opening epochs of the fit before paying for the whole non-convex optimisation, and it marks where our own explanation stops being predictive, since the residual does not say in advance which regime a record falls into.

The negative result on the bound deserves emphasis, because it cuts against the way such bounds are often used. Theorem 1 is valid and, by Proposition 1, attained in the worst case; nevertheless, along a regularisation path, its value is strongly negatively correlated with the realised horizon error on 2 of the four records, uncorrelated on a third and positively correlated only on the fourth, and minimising the surrogate that an optimiser would actually be handed drives the estimator to a nearly trivial model on 3 of them. The reason is structural rather than a matter of constants: the amplification factor is minimised by damped models, and damping is precisely how a predictor stops being useful. A valid bound is not automatically a valid model-selection criterion, and reporting a small certified error can mean the model has been flattened rather than improved. A related point has been made from a different direction for uncontrolled EDMD, namely that the one-step residual is not the quantity that measures predictive quality and that consistency-based diagnostics track it better [98].

The size of the certificate on the measured records makes the same point in absolute terms. On the warm-start models (EDMDc + SR with the ridge parameter selected by the original protocol, $\bar{\rho} = 1$, $H = 20$), evaluated on the fitting windows $\mathcal{S}_H$ in standardised units and averaged over five folds, the certified root-mean-square bound of (18) is 41.1 on the industrial robot against a realised twenty-step RMS error of 0.083, 126 against 0.0050 on EMPS, 54.5 against 0.23 on the cascaded tanks and 14.5 against 0.093 on Silverbox. The ratio of realised error to bound lies between 2.0×10^{-3} and 2.2×10^{-3}, 3.1×10^{-5} and 6.6×10^{-5}, 2.5×10^{-3} and 9.0×10^{-3}, and 6.2×10^{-3} and 6.6×10^{-3} on the four records, so the certificate exceeds the error by between two and four and a half orders of magnitude. On random instances, the same ratio reaches 0.41 for (18) and 0.64 for (16), and Proposition 1 attains equality for a single output. The bound is therefore tight in the worst-case sense and numerically large on measured data, because the realised residual sequences are far from aligned with the worst-case directions. The factorisation of Theorem 1 does provide an

explanation of *where* the horizon error comes from, in particular of why an unstable mode that is observable through C is fatal while $\rho(A) < 1$ is not by itself reassuring, together with the guarantee that the error stays finite over every finite horizon. It is not a numerical forecast of that error. The constant 2Δ of Theorem 2 is to be read in the same way, which is why the closed-loop part of this study reports measured cost (Section 7.5) rather than the bound.

8.2. Computational Cost and Scalability

The cost of the horizon-consistent fit is set by the rolled-out recursion. One epoch propagates every window through H steps, and each step multiplies the lifted state by an $N \times N$ matrix and the input by an $N \times m$ matrix and applies one $p \times N$ readout, so the forward and backward passes together cost $O(|\mathcal{S}|\, H\, N(N + m + p))$ per epoch. The automatic-differentiation graph holds $O(b\, H\, N)$ entries for a batch of b windows, and the spectral-radius retraction adds one $O(N^3)$ eigendecomposition every ten steps. The closed-form EDMDc estimate costs $O(T_f(N + m)^2 + (N + m)^3)$ and is computed once. On the four records, this difference appears in the t_{fit} column of Table 2, where the fit time rises from 0.6 s for EDMDc to 28.4 s for HC-EDMDc on the robot, from 0.4 s to 4.4 s on EMPS, from below 0.1 s to 1.2 s on the tanks and from 0.1 s to 27.3 s on Silverbox.

Table 11 separates the three factors of the per-epoch cost on the Silverbox record, varying the lifted dimension N, the training horizon H and the number of training windows one at a time.

Table 11. Training time per epoch of the horizon-consistent fit on the Silverbox record, measured on an RTX 5080 GPU in double precision with a batch size of 2048; each entry is the mean over ten timed epochs, without the horizon curriculum. The last column is the time of one closed-form EDMDc fit at the same lifted dimension. The three blocks vary the lifted dimension N, the training horizon H and the number of training windows (10, 25, 50 and 100 % of the training record) in turn.

N	H	Windows	Time per Epoch (s)	Closed-Form Fit (s)
Lifted dimension N				
40	20	52,030	1.73	0.009
140	20	52,030	1.68	0.21
240	20	52,030	2.02	0.28
440	20	52,030	4.34	1.16
840	20	52,030	13.59	1.53
Training horizon H				
40	5	52,045	0.55	0.009
40	10	52,040	0.93	0.012
40	20	52,030	1.63	0.007
40	50	52,000	4.05	0.009
40	100	51,950	8.30	0.009
Number of training windows				
40	20	5,185	0.23	0.001
40	20	12,993	0.47	0.003
40	20	26,005	0.86	0.004
40	20	52,030	1.81	0.008

The time per epoch is close to linear in the training horizon and in the number of windows, rising from 0.55 s at $H = 5$ to 8.30 s at $H = 100$ and from 0.23 s at 5185 windows to 1.81 s at 52,030. The dependence on N is flatter than the complexity count suggests at small sizes. Up to $N = 240$, the GPU is not saturated and the time stays between 1.68 s and 2.02 s, whereas from $N = 440$ to $N = 840$, it grows from 4.34 s to 13.59 s, about 3.1 times, in line with the N^2 term. The closed-form fit stays at or below 1.53 s over the whole range,

and the setting $N = 40$, $H = 20$ that recurs in all three blocks (1.63–1.81 s) indicates the resolution of these timings.

The online cost of Koopman MPC does not depend on how the predictor was identified. The dimension of the condensed QP is fixed by the prediction horizon N_p and the input dimension m, so every estimator in this study yields a QP of the same size. In the closed-loop runs, the mean solve time stayed below 0.42 ms and the 95th percentile reached 1.00 ms at $N_p = 100$ (Section 7.5), which places the whole additional cost of the method in the offline fit.

The horizon-consistent fit is therefore suited to applications in which the model is identified once offline and then used over many control periods. There, an offline cost of a few seconds to a few tens of seconds is an acceptable price for long-horizon accuracy or a stable free-run simulation. Adaptive schemes that re-identify the model at every sampling period are better served by the closed-form EDMDc estimate or its recursive form.

8.3. When to Use the Horizon-Consistent Fit

The results of Section 7, together with a short probe from the warm start, give a practitioner an ordered sequence of checks. The probe was run on every record and fold. It evaluates the gradient of J_H at the warm start and then runs twenty epochs from it at the original learning rate over the full horizon $H = 20$, recording after each epoch the training rolled-out objective and the held-out loss. The four steps are as follows.

1. Fit the closed-form EDMDc estimate first and check $\rho(A)$. The one-step estimator returned $\rho(A) \geq 1$ on 8 of the 20 fits in this study, and whenever the radius is not below one, the spectral-radius retraction should be applied. Its price is at most 8.4 % of twenty-step accuracy, against a free run that otherwise leaves the data range.
2. Establish which part of the horizon the controller actually uses. A controller that relies on one-step or very short predictions is better served by EDMDc + SR, since on Silverbox the one-step error of HC-EDMDc is 5.1 times that of EDMDc + SR and the two curves cross only between $h = 5$ and $h = 20$. For long horizons, HC-EDMDc is the appropriate choice, with the training horizon matched to the control horizon, because Table 4 shows the accuracy at $h = 20$ improving as H approaches the evaluation horizon. On the cascaded tanks the rolled-out fit is no worse at any horizon tested.
3. Read the first twenty epochs of the fit. The gradient at the warm start is non-zero on every record and fold, with the scale-free first-order measure $\|\nabla J_H\|_F \|\theta\|_F / J_H$ between 5×10^1 and 6×10^4, so the warm start is never a first-order stationary point. Even so, on the robot and EMPS records, no epoch-end iterate lowered the training objective or the held-out loss on any of the five folds, and the held-out criterion returns the warm start. Both warm starts nearly interpolate their records, with $\mathrm{NRMSE}_{h=5}$ of 0.0008 and 0.0013 against 0.1332 on the tanks and 0.0314 on Silverbox, and a first-order method finds no descent step from them. On the cascaded tanks the training objective fell on four of five folds, by 10.5–41.7 %, while the held-out loss fell on two folds (17.3 % and 45.3 %) and rose on two (57.3 % and 112.8 %). That record is short, and selection on the held-out set decides the outcome. On Silverbox, the training objective fell on all five folds by 5.5–8.6 % and the held-out loss by 5.2–8.4 %, which is the regime in which the refinement pays. The probe took 2.5 s on EMPS, 2.0 s on the tanks, 17.1 s on the robot and 39.0 s on Silverbox.
4. Do not decide on the size of the one-step residual or of the certified bound. The residual ε is 1.90 on the robot, 6.29 on EMPS, 2.88 on the tanks and 0.99 on Silverbox, and the largest of the four belongs to EMPS, on which the refinement changes nothing. The bound does not rank models either (Section 7.2).

These checks distinguish three regimes, namely a warm start from which no descent step is found, a short record on which held-out selection dominates, and a record on which training and held-out losses fall together. They turn a boundary that could previously be observed only after the full fit into a quantity that can be read once the fit has started.

8.4. Relation to the Literature

Earlier work has already observed that composing an EDMD approximant amplifies its error. This observation motivates the finite-data error bounds of [17,18] and the closed-loop guarantees built on them [35,37], and it is the explicit starting point of the multi-step formulation of [79].

Our treatment differs from these works in two respects. First, we express the amplification through the computable quantity Γ_h rather than through a spectral or Lipschitz surrogate, which makes it available as a per-model diagnostic and shows, via Proposition 1, that it is attained for single-output systems. Second, and in deliberate contrast to [79], we keep a single recursive lifted linear predictor rather than learning horizon-dependent condensed maps. This choice gives up convexity, the strength of their formulation, and in return provides a model that can be simulated beyond its training horizon, a spectral-radius certificate, and a parameter count independent of H. Which trade is preferable depends on whether the model is wanted as a one-shot horizon predictor or as a state-space object.

Neither ingredient of the proposed estimator is new on its own. Training on a rolled-out criterion is what separates output-error identification from equation-error methods [85,86], and it appears in the learning literature as multi-step or consistency losses for Koopman autoencoders [24–26]. Stability-constrained learning of linear dynamics has a similar history [99–102]. Recent work parametrises Koopman models with built-in stability and stabilisability [103] or learns a noise-robust stable Koopman operator from Hankel-DMD observables for model predictive control [104]. Our contribution is to tie both ingredients to the same computable bound and to measure what each is worth on measured industrial records.

Table 12 places the proposed estimator among five classes of existing approaches. HC-EDMDc is the only row that combines a single recursive predictor, an explicit stability constraint and a computable multi-step error bound. The price is a training problem that is non-convex for every $H \geq 2$.

The analysis also applies beyond the estimator and plant studied here. Theorem 1, Proposition 1 and Corollary 1 use only the fact that the predictor (A, B, C) is applied recursively, so they hold for any lifted linear predictor, including the local predictor of each subsystem in distributed MPC. In Koopman-DMPC under bounded communication delay [105], neighbour information arrives up to d samples late, each local prediction extrapolates $N_p + d$ steps, and the relevant amplification factor is Γ_{N_p+d}. HC-EDMDc can then be trained with $H = N_p + d$, with the neighbours' coupling signals entering B as additional input channels. Bilinear lifted forms [38,39] have a state-dependent input matrix, so the bound has to be re-derived for them. The experiments in this paper cover only centralised MPC of a single plant.

8.5. Limitations

Several limitations remain. First, the closed-loop plant is not hardware but a first-principles plant model identified from the same measured record as the predictor. With an open-loop fit of $R^2 = 98.7\,\%$, the EMPS model is a reasonable stand-in. The cascaded-tanks model is considerably coarser ($R^2 = 91.0\,\%$), so the absolute tracking levels obtained on it do not indicate attainable performance. On that record, only the comparison between predictors run against the identical plant carries meaning, and that comparison differed

by less than one percent. A hardware loop, or a plant identified from records held out from predictor training, is needed before the closed-loop numbers can be read as more than relative.

Second, the closed-loop study does not separate the estimators, and no setting has been constructed in which it would. An open-loop gain of roughly one fifth at $h = 20$ changed closed-loop cost by 1.0 %. The reason is structural and is given in Section 7.5. Receding-horizon feedback discards most of each prediction. A closed-loop benefit from horizon-consistent identification can only be shown in a regime that forces the loop to rely on the far end of its prediction, such as long sampling intervals, input-rate limits or explicitly economic objectives, and that experiment remains to be done.

Table 12. Training criterion, predictor structure, stability handling and guarantees of related identification approaches and of the proposed estimator.

Approach	Training Criterion	Predictor Structure	Stability Handling	Multi-Step Error Bound	Convex Fit
EDMDc/Koopman MPC [10,27,84]	One-step lifted residual	Single recursive lifted linear model	None	Finite-data, via spectral or Lipschitz-type constants [17,18]	Yes (closed form)
Output-error and simulation-error identification [85,86,94–96]	Multi-step or simulation output error	Linear or nonlinear state-space model	Model-structure specific	Not stated in this form	No
Koopman autoencoders with multi-step losses [24–26]	Multi-step or consistency loss in learned observables	Learned encoder, linear latent dynamics	None or soft penalty	No	No
Stability-constrained linear and Koopman learning [99–104]	One-step or simulation-error fit	Linear or lifted linear model	Constraint generation, Lyapunov-type certificates, nearest stable matrix or stable parametrisation	No horizon-wise bound	Depends on the formulation
Condensed multi-step maps [79]	Horizon-wise output error	Horizon-specific matrices (not recursive)	Not required within the horizon	n/a	Yes
This paper (HC-EDMDc)	Rolled-out output error over H, held-out selection	Single recursive lifted linear model	Spectral-radius retraction, $\rho(A) \leq \bar{\rho}$	Computable Γ_h bound, attained for $p = 1$, horizon-uniform Lyapunov version	Convex at $H = 1$ only

Third, the rolled-out objective (24) is non-convex. It is solved by a first-order method under a horizon curriculum, starting from a closed-form warm start. Proposition 3 guarantees only that the returned model does not worsen the criterion on which it is selected. No guarantee of reaching a global minimiser is available, and on the two benchmarks where no generated iterate improved on the warm start, the optimiser made no progress.

Fourth, the retraction (28) restores feasibility without being the Euclidean projection onto $\mathcal{A}_{\bar{\rho}}$. The iterates therefore fall outside the convergence theory of projected gradient methods.

Fifth, the spectral-radius constraint governs asymptotic behaviour only. By Corollary 1, the constant M in $\|A^h\|_2 \leq M\bar{\rho}^h$ is the condition number of the Lyapunov solution and is not bounded by $\rho(A)$. Imposing $\rho(A) \leq \bar{\rho}$ consequently neither limits transient amplification nor guarantees better prediction accuracy. In Figure 3c, the gain curves of models that satisfy the constraint grow by several orders of magnitude before decaying, and in Table 5, tightening the radius to $\bar{\rho} = 0.99$ raises the EMPS free-run simulation error from 0.088 to 0.664. Criteria that act on transient amplification directly do exist. A spectral-norm or contraction constraint $\|A\|_2 \leq 1$ is convex and would restore the projected-gradient theory, but it is far too strong for delay-embedded predictors. A Lyapunov inequality $A^\top PA - \bar{\rho}^2 P \preceq 0$

for a given P, or an LMI constraint with an upper bound on $\kappa(P)$, constrains M as well. Stable parametrisations that are stable by construction [102–104] and constraint-generation or nearest-stable-matrix methods [99–101] offer further routes. Combining these criteria with the rolled-out objective and comparing their accuracy cost has not been done here and is the next step.

Sixth, the evidence comes from four records, and however different they are, four records remain a small sample. Three are single-output. Only the robot record is genuinely multivariable, and the refinement is neutral on it. The regime boundary proposed in Section 7 thus rests on the two records where the refinement is active, and the noise and training-size sweeps designed to probe it did not reproduce the predicted direction. The boundary is a conjecture, and a weaker one than expected. The distributed and delayed setting of Koopman MPC discussed in the preceding subsection was not tested experimentally.

Finally, Theorem 2 is an open-loop suboptimality statement that holds at each instant. A closed-loop guarantee would require terminal ingredients or a cost-controllability assumption of the kind used in [2,35]. The constant Δ in (32) is a supremum over the whole admissible input set, so it is conservative whenever the realised inputs cover only part of that set.

9. Conclusions

This paper quantified the gap between how extended dynamic mode decomposition fits a lifted linear predictor and how model predictive control uses it. The fit minimises a one-step residual, whereas the controller applies the predictor over a horizon of tens of samples. Theorem 1 bounds the h-step output error by the product of the one-step residual and the amplification factor $\Gamma_h = \sum_{j<h} \|CA^j\|_2$, and Proposition 1 shows that the bound is attained for single-output systems. Corollary 1 gives a horizon-uniform version through a scaled Lyapunov certificate, whose constant is a norm-equivalence factor rather than a spectral one. Theorem 2 carries the same quantity into the control loop and bounds the suboptimality of the certainty-equivalent input sequence by twice the weighted horizon prediction error.

The two ingredients of the resulting estimator, HC-EDMDc, behave differently on four measured industrial records, and both outcomes are worth stating. HC-EDMDc minimises the rolled-out output error under a spectral-radius constraint, starts from the closed-form solution as its warm start and is selected on a held-out rolled-out criterion. The stability constraint prevents the free runs that leave the data range, which the unconstrained one-step estimator produces on two of the four benchmarks, at a cost of at most 8.4 % in twenty-step accuracy and considerably more at one step. The horizon-consistent refinement improves horizon accuracy on the two records where it finds an improvement over its warm start—by roughly a fifth on the cascaded-tank record—and returns that warm start unchanged on the other two. The size of the one-step residual does not predict which case a record falls into, since the largest residual of the four belongs to a neutral record. The first twenty epochs of the fit already show the case (Section 8.3); on the robot and EMPS records, no iteration lowers the training objective, on the cascaded tanks, the training objective falls while the held-out loss rises and falls, and on Silverbox, both fall together. Proposition 2 offers a candidate explanation for the boundary through the attenuation bias of the one-step criterion, but the noise and data-size sweeps do not confirm the direction it predicts, and we report them as they came out; the experiments do not identify a quantity that predicts the regime in advance. We also report a negative result. Although the bound of Theorem 1 is valid and worst-case tight, its value moves *opposite* to the realised horizon error along a regularisation path on 2 of the four records, is absent on a third and runs the other way on the fourth. Minimising the product $\varepsilon\,\Gamma_H$ directly collapses the predictor

towards a mean-reverting model on the three records on which that optimisation improves its own objective at all. On the four recorded warm-start models, the certified RMS bound exceeds the realised twenty-step error by two to four and a half orders of magnitude (Section 8). A valid certificate is therefore not automatically a model-selection criterion.

Five directions follow from these results.

1. The amplification factor Γ_h is differentiable and cheap to evaluate, so it can enter the choice of dictionary and delay order rather than serve only as a post hoc diagnostic.
2. The boundary between the regimes identified here is empirical. A finite-sample statement relating dictionary richness, sample count and the horizon at which the two criteria separate would allow for the three cases to be predicted before fitting.
3. The spectral-radius constraint can be replaced by stability conditions that limit transient amplification directly, such as Lyapunov or LMI constraints with an upper bound on $\kappa(P)$ and parametrisations that are stable by construction. Combining these conditions with the rolled-out criterion would show how their accuracy cost compares with that of the spectral-radius constraint.
4. The analysis and the estimator can be extended to distributed Koopman MPC and to Koopman MPC with communication delay, where the amplification factor becomes Γ_{N_p+d}, and to event-triggered data-driven control. In each setting, closed-loop stability and recursive feasibility have to be established under a terminal ingredient or a cost-controllability assumption.
5. The closed-loop evaluation here is against an identified first-principles plant. A hardware study on an industrial manipulator, with scenarios that force the controller to rely on the far end of the prediction (long sampling intervals, input-rate limits, economic objectives), would test whether the open-loop gains turn into closed-loop gains.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/math14214047/s1, Analysis code (data staging, estimators, baselines, Koopman MPC controller, figure generation and numerical checks of the theoretical results) and the per-run result files behind every number and table reported in the article.

Author Contributions: Conceptualization, D.T.; methodology, J.Z. and D.T.; software, J.Z.; validation, J.Z.; formal analysis, J.Z.; investigation, J.Z.; data curation, J.Z.; writing—original draft preparation, J.Z.; writing—review and editing, D.T.; visualization, J.Z.; supervision, D.T.; project administration, D.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All four records analysed in this study are publicly released measurement data and no new data were created. The industrial-robot identification benchmark is available from the RPTU Kaiserslautern research-data repository at https://doi.org/10.26204/DATA/5 under a dual MIT/CC BY-SA 4.0 licence (accessed on 11 August 2026). The electro-mechanical positioning system, cascaded-tank and Silverbox records are available from the nonlinear-system-identification benchmark collection at https://www.nonlinearbenchmark.org/benchmarks/emps (accessed on 11 August 2026), https://www.nonlinearbenchmark.org/benchmarks/cascaded-tanks (accessed on 11 August 2026) and https://www.nonlinearbenchmark.org/benchmarks/silverbox (accessed on 11 August 2026). The Supplementary Materials contain the complete analysis code—data staging, estimators, baselines, controller, figures of merit and the numerical checks of the theorems, propositions and corollary—together with the per-run result files behind every number and table reported here. The measured records themselves are not redistributed in them: `prepare_data.py` stages them from the sources above into the form the remaining scripts consume, after which those scripts reproduce the reported values.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Rawlings, J.B.; Mayne, D.Q.; Diehl, M.M. *Model Predictive Control: Theory, Computation, and Design*, 2nd ed.; Nob Hill Publishing: Madison, WI, USA, 2017.
2. Grüne, L.; Pannek, J. *Nonlinear Model Predictive Control: Theory and Algorithms*, 2nd ed.; Springer: Cham, Switzerland, 2017.
3. Stellato, B.; Banjac, G.; Goulart, P.; Bemporad, A.; Boyd, S. OSQP: An operator splitting solver for quadratic programs. *Math. Program. Comput.* **2020**, *12*, 637–672. [CrossRef]
4. Koopman, B.O. Hamiltonian Systems and Transformation in Hilbert Space. *Proc. Natl. Acad. Sci. USA* **1931**, *17*, 315–318. [CrossRef]
5. Mezić, I. Spectral Properties of Dynamical Systems, Model Reduction and Decompositions. *Nonlinear Dyn.* **2005**, *41*, 309–325. [CrossRef]
6. Budišić, M.; Mohr, R.; Mezić, I. Applied Koopmanism. *Chaos Interdiscip. J. Nonlinear Sci.* **2012**, *22*, 047510. [CrossRef]
7. Brunton, S.L.; Budišić, M.; Kaiser, E.; Kutz, J.N. Modern Koopman Theory for Dynamical Systems. *SIAM Rev.* **2022**, *64*, 229–340. [CrossRef]
8. Schmid, P.J. Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.* **2010**, *656*, 5–28. [CrossRef]
9. Rowley, C.W.; Mezić, I.; Bagheri, S.; Schlatter, P.; Henningson, D.S. Spectral analysis of nonlinear flows. *J. Fluid Mech.* **2009**, *641*, 115–127.. [CrossRef]
10. Williams, M.O.; Kevrekidis, I.G.; Rowley, C.W. A Data–Driven Approximation of the Koopman Operator: Extending Dynamic Mode Decomposition. *J. Nonlinear Sci.* **2015**, *25*, 1307–1346. [CrossRef]
11. Klus, S.; Nüske, F.; Koltai, P.; Wu, H.; Kevrekidis, I.; Schütte, C.; Noé, F. Data-Driven Model Reduction and Transfer Operator Approximation. *J. Nonlinear Sci.* **2018**, *28*, 985–1010. [CrossRef]
12. Korda, M.; Mezić, I. On Convergence of Extended Dynamic Mode Decomposition to the Koopman Operator. *J. Nonlinear Sci.* **2018**, *28*, 687–710. [CrossRef]
13. Mezić, I. On Numerical Approximations of the Koopman Operator. *Mathematics* **2022**, *10*, 1180. [CrossRef]
14. Arbabi, H.; Mezić, I. Ergodic Theory, Dynamic Mode Decomposition, and Computation of Spectral Properties of the Koopman Operator. *SIAM J. Appl. Dyn. Syst.* **2017**, *16*, 2096–2126. [CrossRef]
15. Mauroy, A.; Mezic, I. Global Stability Analysis Using the Eigenfunctions of the Koopman Operator. *IEEE Trans. Autom. Control* **2016**, *61*, 3356–3369. [CrossRef]
16. Mauroy, A. Koopman Operator Framework for Spectral Analysis and Identification of Infinite-Dimensional Systems. *Mathematics* **2021**, *9*, 2495. [CrossRef]
17. Nüske, F.; Peitz, S.; Philipp, F.; Schaller, M.; Worthmann, K. Finite-Data Error Bounds for Koopman-Based Prediction and Control. *J. Nonlinear Sci.* **2023**, *33*, 14. [CrossRef]
18. Philipp, F.M.; Schaller, M.; Worthmann, K.; Peitz, S.; Nüske, F. Error bounds for kernel-based approximations of the Koopman operator. *Appl. Comput. Harmon. Anal.* **2024**, *71*, 101657. [CrossRef]
19. Kostic, V.; Novelli, P.; Maurer, A.; Ciliberto, C.; Rosasco, L.; Pontil, M. Learning Dynamical Systems Via Koopman Operator Regression in Reproducing Kernel Hilbert Spaces. In Proceedings of the Advances in Neural Information Processing Systems 35, New Orleans, LA, USA, 28 November–9 December 2022; pp. 4017–4031. [CrossRef]
20. Li, Q.; Dietrich, F.; Bollt, E.M.; Kevrekidis, I.G. Extended dynamic mode decomposition with dictionary learning: A data-driven adaptive spectral decomposition of the Koopman operator. *Chaos Interdiscip. J. Nonlinear Sci.* **2017**, *27*, 103111. [CrossRef]
21. Folkestad, C.; Pastor, D.; Mezic, I.; Mohr, R.; Fonoberova, M.; Burdick, J. Extended Dynamic Mode Decomposition with Learned Koopman Eigenfunctions for Prediction and Control. In Proceedings of the 2020 American Control Conference (ACC), Denver, CO, USA, 1–3 July 2020; pp. 3906–3913. [CrossRef]
22. Korda, M.; Mezic, I. Optimal Construction of Koopman Eigenfunctions for Prediction and Control. *IEEE Trans. Autom. Control* **2020**, *65*, 5114–5129. [CrossRef]
23. Cibulka, V.; Korda, M.; Haniš, T. Dictionary-Free Koopman Model Predictive Control with Nonlinear Input Transformation. *SIAM J. Appl. Dyn. Syst.* **2025**, *24*, 94–130. [CrossRef]
24. Lusch, B.; Kutz, J.N.; Brunton, S.L. Deep learning for universal linear embeddings of nonlinear dynamics. *Nat. Commun.* **2018**, *9*, 4950. [CrossRef]
25. Otto, S.E.; Rowley, C.W. Linearly Recurrent Autoencoder Networks for Learning Dynamics. *SIAM J. Appl. Dyn. Syst.* **2019**, *18*, 558–593. [CrossRef]
26. Azencot, O.; Erichson, N.B.; Lin, V.; Mahoney, M.W. Forecasting Sequential Data Using Consistent Koopman Autoencoders. In *Proceedings of the 37th International Conference on Machine Learning*; Proceedings of Machine Learning Research: Cambridge, MA, USA, 2020; Volume 119, pp. 475–485.

27. Korda, M.; Mezić, I. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica* **2018**, *93*, 149–160. [CrossRef]
28. Arbabi, H.; Korda, M.; Mezic, I. A Data-Driven Koopman Model Predictive Control Framework for Nonlinear Partial Differential Equations. In *Proceedings of the 2018 IEEE Conference on Decision and Control (CDC)*; IEEE: New York, NY, USA, 2018; pp. 6409–6414. [CrossRef]
29. Bruder, D.; Fu, X.; Gillespie, R.B.; Remy, C.D.; Vasudevan, R. Koopman-Based Control of a Soft Continuum Manipulator Under Variable Loading Conditions. *IEEE Robot. Autom. Lett.* **2021**, *6*, 6852–6859. [CrossRef]
30. Mamakoukas, G.; Castano, M.; Tan, X.; Murphey, T. Local Koopman Operators for Data-Driven Control of Robotic Systems. In Proceedings of the Robotics: Science and Systems XV, Freiburg, Germany, 22–26 June 2019. [CrossRef]
31. Bevanda, P.; Sosnowski, S.; Hirche, S. Koopman operator dynamical models: Learning, analysis and control. *Annu. Rev. Control* **2021**, *52*, 197–212. [CrossRef]
32. Zhang, X.; Pan, W.; Scattolini, R.; Yu, S.; Xu, X. Robust tube-based model predictive control with Koopman operators. *Automatica* **2022**, *137*, 110114. [CrossRef]
33. Mamakoukas, G.; Di Cairano, S.; Vinod, A.P. Robust Model Predictive Control with Data-Driven Koopman Operators. In Proceedings of the 2022 American Control Conference (ACC), Atlanta, GA, USA, 8–10 June 2022; pp. 3885–3892. [CrossRef]
34. Caldarelli, E.; Chatalic, A.; Colomé, A.; Molinari, C.; Ocampo-Martinez, C.; Torras, C.; Rosasco, L. Linear quadratic control of nonlinear systems with Koopman operator learning and the Nyström method. *Automatica* **2025**, *177*, 112302. [CrossRef]
35. Bold, L.; Grüne, L.; Schaller, M.; Worthmann, K. Data-Driven MPC with Stability Guarantees Using Extended Dynamic Mode Decomposition. *IEEE Trans. Autom. Control* **2025**, *70*, 534–541. [CrossRef]
36. Narasingam, A.; Son, S.H.; Kwon, J.S.I. Data-driven feedback stabilisation of nonlinear systems: Koopman-based model predictive control. *Int. J. Control* **2023**, *96*, 770–781. [CrossRef]
37. Strässer, R.; Worthmann, K.; Mezić, I.; Berberich, J.; Schaller, M.; Allgöwer, F. An overview of Koopman-based control: From error bounds to closed-loop guarantees. *Annu. Rev. Control* **2026**, *61*, 101035. [CrossRef]
38. Goswami, D.; Paley, D.A. Bilinearization, Reachability, and Optimal Control of Control-Affine Nonlinear Systems: A Koopman Spectral Approach. *IEEE Trans. Autom. Control* **2022**, *67*, 2715–2728. [CrossRef]
39. Iacob, L.C.; Tóth, R.; Schoukens, M. Koopman form of nonlinear systems with inputs. *Automatica* **2024**, *162*, 111525. [CrossRef]
40. Haseli, M.; Cortés, J. Generalizing dynamic mode decomposition: Balancing accuracy and expressiveness in Koopman approximations. *Automatica* **2023**, *153*, 111001. [CrossRef]
41. Huang, J.; Huang, Z.; Xie, C.; Zhang, Y.; Ostachowicz, W.; Cao, M. Unsupervised deep learning framework for damage identification under ambient excitations: Trait of damage localization and demonstrative applications. *Mech. Syst. Signal Process.* **2026**, *250*, 114123. [CrossRef]
42. Zhou, X.; Li, Z.; Liu, X.; Wang, Y.; Du, L. Calculation model for the thermal conductivity of concrete considering microstructural characteristics. *Case Stud. Therm. Eng.* **2026**, *79*, 107789. [CrossRef]
43. Liu, X.; Fang, L.; Zhou, X.; Xu, Q.; Cheng, K. Mechanical response characteristics and influencing factors analysis of non-parallel complex subway station-tunnel groups close proximity construction. *Tunn. Undergr. Space Technol.* **2026**, *171*, 107472. [CrossRef]
44. Zhou, X.; Li, Z.; Liu, X.; Wang, Y.; Liang, N.; Zhong, Z. Physics-informed neural network (PINN) for predicting the temperature field in high geothermal tunnels. *Tunn. Undergr. Space Technol.* **2026**, *169*, 107265. [CrossRef]
45. Cui, B.; Wang, H. Optimization of Risk-Based and Cost-Minimized Maintenance Strategy for Corroded Pipeline Using Reinforcement Learning. *Reliab. Eng. Syst. Saf.* **2026**, *276*, 112929. [CrossRef]
46. Liu, Z.; Gu, X.; Kwan, M.P.; Cui, B. Integrated Ultrasonic Testing and Numerical Simulation for Damage Detection in Steel Bridge Deck Pavements. *Eng. Struct.* **2026**, *355*, 122421. [CrossRef]
47. Nguyen Thi My, L.; Vo, T. An Integration of Fuzzy-Enhanced Heterophilic Graph Representation Learning for Improving Performance of Link Prediction. *Cybern. Syst.* **2025**, *57*, 2084–2112. [CrossRef]
48. Cui, Q.; Ding, Z.; Chen, F. Hybrid Directed Hypergraph Learning and Forecasting of Skeleton-Based Human Poses. *Cyborg Bionic Syst.* **2024**, *5*, 0093. [CrossRef]
49. Wang, H.; Wang, Y.; Yan, S.; Du, X.; Gao, Y.; Liu, H. Merge-and-Split Graph Convolutional Network for Skeleton-Based Interaction Recognition. *Cyborg Bionic Syst.* **2024**, *5*, 0102. [CrossRef]
50. Ye, W.; Wang, J.; Chen, L.; Dai, L.; Sun, Z.; Liang, Z. Adaptive Spatial–Temporal Aware Graph Learning for EEG-Based Emotion Recognition. *Cyborg Bionic Syst.* **2024**, *5*, 0088. [CrossRef]
51. Chen, Y.; Liu, Z.W.; Qin, Y. Finite-Time Topology Identification of Delayed Complex Dynamical Networks and Its Application. *Cyborg Bionic Syst.* **2024**, *5*, 0092. [CrossRef]
52. Ji, Z.; Sun, G.; Wang, K.; Wong, H.; Yu, Z.; Li, Z.; Wei, C.; Liu, P. A Circularly Polarized Complementary Antenna with Substrate Integrated Coaxial Line Feed for X-Band Applications. *Electronics* **2024**, *13*, 785. [CrossRef]

53. Wei, C.; Wu, K.L. Array-antenna decoupling surfaces for quasi-yagi antenna arrays. In *Proceedings of the 2017 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting*; IEEE: New York, NY, USA, 2017; pp. 2103–2104.. [CrossRef]
54. Wu, W.J.; Cheng, Y.F.; Chen, S.; Wei, C.; Wang, G. Decoupling and bandwidth equalization for MIMO antennas by using novel network design. *Microw. Opt. Technol. Lett.* **2022**, *64*, 1462–1467. [CrossRef]
55. Mao, F.; Lin, J.; Hou, Y. Dynamic evolution and frontiers of energy decentralization research: A bibliometric-based review. *Energy Strategy Rev.* **2026**, *64*, 102097. [CrossRef]
56. Xiao, B.; Guo, X.; Si, F. Reassessing determinants of urban energy intensity in China: Insights from controllable and uncontrollable factors. *Environ. Sci. Pollut. Res.* **2024**, *31*, 38367–38384. [CrossRef]
57. Zhu, B.; Yan, J.; Hu, M.; Yang, Y.; Ji, B.; She, X. A Highly Simplified Active Battery Equalizer Based on a High-Accuracy Series-Port Converter with Voltage Self-Equalization. *IEEE Trans. Ind. Electron.* **2026**, 1–11. [CrossRef]
58. Zhu, B.; Liu, C.; She, X.; Wu, H.; Yang, Y. A Series-Port Voltage High-Precision Self-Balancing Based on Flyback Converter Circuit for Active Balancing of Battery Packs. *IEEE Trans. Power Electron.* **2026**, 1–12. [CrossRef]
59. Yang, S.; Ren, X.; Zha, B.; Bao, Q.; Feng, B.; Xu, Z. EFSRNet: Multi-scale exposure normalization and dual-branch aggregation for overexposed face super-resolution. *Knowl.-Based Syst.* **2026**, *339*, 115578. [CrossRef]
60. Liu, Z.; Kwan, M.P.; Jiang, W.; Liu, Y.; Cui, B. A Lightweight Multiscale Fusion Framework for Traffic Vehicle Detection from Satellite Remote Sensing, UAV, and CCTV Imagery. *IEEE Trans. Geosci. Remote Sens.* **2026**, *64*, 5632717. [CrossRef]
61. Yan, W.; Ren, J.; Feng, J.; Duan, Y.; Wei, C. A New Forest Fire Risk Rating Forecast Model Based on XGBoost. In Proceedings of the 2022 International Seminar on Computer Science and Engineering Technology (SCSET), Indianapolis, IN, USA, 8–9 January 2022; pp. 227–230. [CrossRef]
62. Wei, C.; Guo, C.; Yan, W. Forest Fire Risk Forecast Method with Pseudo Label Based on Semi-supervised Learning. In Proceedings of the 2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), Taiyuan, China, 3–5 December 2021; pp. 36–39. [CrossRef]
63. Xia, Y.; Shi, B.; Wang, K.; Hu, L.; Wang, Q.; Xu, S.; Wang, X.; Xu, P.; She, Y.; Xie, H.; et al. A trinity STING-activating nanoparticle harnesses cancer cell STING machinery for enhanced immunotherapy. *J. Control. Release* **2025**, *377*, 256–266. [CrossRef]
64. Chen, Y.; Si, F.; Lu, X.; Li, X. Research on the Influence Mechanism of the Across-Industrial-Chain Investment Speed on Innovation Performance of AI Enterprises: Improvement Path of Artificial Intelligence Technology Application. *Mob. Inf. Syst.* **2021**, *2021*, 6149746. [CrossRef]
65. Liu, Y.; Chen, Y.; Peng, K.; Si, F. Tourist perception of city marketing quality: Scale development and validation. *Asia Pac. J. Tour. Res.* **2025**, *30*, 1610–1640. [CrossRef]
66. Qi, Z.; Chen, Z.; Si, F. Multi-Algorithm Optimization of Explanation Stability in Ensemble Learning for Human Capital Attrition Risk Prediction. *Mathematics* **2026**, *14*, 3283. [CrossRef]
67. Zhang, Y.; Zhang, Y.; Liang, Y.; Mu, S.; Zhang, Z.; Chen, X. AD-VGF: An Improved Generative Adversarial Network for Credit Card Risk Identification and Management in Digital Economy. *Cybern. Syst.* **2025**, *57*, 2058–2083. [CrossRef]
68. Li, S.; Si, F.; Mo, D.; He, C. Stochastic Mirror-Prox Optimisation of Group Distributionally Robust Multimodal Logistic Regression for Multilingual Fact-Checking Verdict Prediction. *Mathematics* **2026**, *14*, 3574. [CrossRef]
69. Coulson, J.; Lygeros, J.; Dorfler, F. Data-Enabled Predictive Control: In the Shallows of the DeePC. In Proceedings of the 2019 18th European Control Conference (ECC), Naples, Italy, 25–28 June 2019; pp. 307–312. [CrossRef]
70. Markovsky, I.; Dörfler, F. Behavioral systems theory in data-driven analysis, signal processing, and control. *Annu. Rev. Control* **2021**, *52*, 42–64. [CrossRef]
71. Berberich, J.; Allgöwer, F. An Overview of Systems-Theoretic Guarantees in Data-Driven Model Predictive Control. *Annu. Rev. Control Robot. Auton. Syst.* **2025**, *8*, 77–100. [CrossRef]
72. Shen, M.; Wang, X.; Zhu, S.; Wu, Z.; Huang, T. Data-Driven Event-Triggered Adaptive Dynamic Programming Control for Nonlinear Systems With Input Saturation. *IEEE Trans. Cybern.* **2024**, *54*, 1178–1188. [CrossRef]
73. Luo, B.; Yang, Y.; Liu, D.; Wu, H.N. Event-Triggered Optimal Control With Performance Guarantees Using Adaptive Dynamic Programming. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 76–88. [CrossRef]
74. Shen, M.; Wang, X.; Park, J.H.; Yi, Y.; Che, W.W. Extended Disturbance-Observer-Based Data-Driven Control of Networked Nonlinear Systems With Event-Triggered Output. *IEEE Trans. Syst. Man. Cybern. Syst.* **2023**, *53*, 3129–3140. [CrossRef]
75. Wang, X.; Qin, W.; Park, J.H.; Shen, M. Event-triggered data-driven control of discrete-time nonlinear systems with unknown disturbance. *ISA Trans.* **2022**, *128*, 256–264. [CrossRef]
76. Shen, M.; Wang, X.; Zhu, S.; Huang, T.; Wang, Q.G. Event-Triggered Data-Driven Control of Nonlinear Systems via *Q*-Learning. *IEEE Trans. Syst. Man. Cybern. Syst.* **2025**, *55*, 1069–1077. [CrossRef]
77. Li, J.; Chai, T.; Lewis, F.L.; Ding, Z.; Jiang, Y. Off-Policy Interleaved *Q*-Learning: Optimal Control for Affine Nonlinear Discrete-Time Systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 1308–1320. [CrossRef]

78. Trefethen, L.N.; Embree, M. *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*; Princeton University Press: Princeton, NJ, USA, 2005.
79. Wu, L.; Tan, W.G.Y.; Zhou, L.; Braatz, R.D.; Drgoňa, J. Least-Squares Multi-Step Koopman Operator Learning for Model Predictive Control. *arXiv* **2026**, arXiv:2601.11901. [CrossRef]
80. Takens, F. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980*; Lecture Notes in Mathematics; Springer: Berlin/Heidelberg, Germany, 1981; Volume 898, pp. 366–381. [CrossRef]
81. Kamb, M.; Kaiser, E.; Brunton, S.L.; Kutz, J.N. Time-Delay Observables for Koopman: Theory and Applications. *SIAM J. Appl. Dyn. Syst.* **2020**, *19*, 886–917. [CrossRef]
82. Brunton, S.L.; Brunton, B.W.; Proctor, J.L.; Kaiser, E.; Kutz, J.N. Chaos as an intermittently forced linear system. *Nat. Commun.* **2017**, *8*, 19. [CrossRef]
83. Rahimi, A.; Recht, B. Random Features for Large-Scale Kernel Machines. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 3–6 December 2007; Volume 20, pp. 1177–1184.
84. Proctor, J.L.; Brunton, S.L.; Kutz, J.N. Dynamic Mode Decomposition with Control. *SIAM J. Appl. Dyn. Syst.* **2016**, *15*, 142–161. [CrossRef]
85. Ljung, L. *System Identification: Theory for the User*, 2nd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 1999.
86. Schoukens, J.; Ljung, L. Nonlinear System Identification: A User-Oriented Road Map. *IEEE Control Syst.* **2019**, *39*, 28–99. [CrossRef]
87. Berberich, J.; Kohler, J.; Muller, M.A.; Allgower, F. Data-Driven Model Predictive Control With Stability and Robustness Guarantees. *IEEE Trans. Autom. Control* **2021**, *66*, 1702–1717. [CrossRef]
88. Limon, D.; Alamo, T.; Raimondo, D.M.; de la Peña, D.M.; Bravo, J.M.; Ferramosca, A.; Camacho, E.F. Input-to-State Stability: A Unifying Framework for Robust Model Predictive Control. In *Nonlinear Model Predictive Control;* Lecture Notes in Control and Information Sciences; Springer: Berlin/Heidelberg, Germany, 2009; pp. 1–26. [CrossRef]
89. Weigand, J.; Götz, J.; Ulmen, J.; Ruskowski, M. Dataset and Baseline for an Industrial Robot Identification Benchmark. In Proceedings of the 6th Workshop on Nonlinear System Identification Benchmarks, Leuven, Belgium, 25–27 April 2022. [CrossRef]
90. Janot, A.; Gautier, M.; Brunot, M. Data Set and Reference Models of an Electro-Mechanical Positioning System. In Proceedings of the Workshop on Nonlinear System Identification Benchmarks, Eindhoven, The Netherlands, 10–12 April 2019.
91. Schoukens, M.; Mattson, P.; Wigren, T.; Noël, J.P. Cascaded Tanks Benchmark Combining Soft and Hard Nonlinearities. In Proceedings of the Workshop on Nonlinear System Identification Benchmarks, Brussels, Belgium, 25–27 April 2016.
92. Wigren, T.; Schoukens, J. Three free data sets for development and benchmarking in nonlinear system identification. In Proceedings of the 2013 European Control Conference (ECC), Zürich, Switzerland, 17–19 July 2013; pp. 2933–2938. [CrossRef]
93. Schoukens, M.; Noël, J. Three Benchmarks Addressing Open Challenges in Nonlinear System Identification. *IFAC-PapersOnLine* **2017**, *50*, 446–451. [CrossRef]
94. Forgione, M.; Piga, D. *dynoNet*: A neural network architecture for learning dynamical systems. *Int. J. Adapt. Control Signal Process.* **2021**, *35*, 612–626. [CrossRef]
95. Beintema, G.I.; Schoukens, M.; Tóth, R. Deep subspace encoders for nonlinear system identification. *Automatica* **2023**, *156*, 111210. [CrossRef]
96. Ribeiro, A.H.; Tiels, K.; Umenberger, J.; Schön, T.B.; Aguirre, L.A. On the smoothness of nonlinear system identification. *Automatica* **2020**, *121*, 109158. [CrossRef]
97. Van Overschee, P.; De Moor, B. N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica* **1994**, *30*, 75–93. [CrossRef]
98. Haseli, M.; Cortes, J. Temporal Forward–Backward Consistency, Not Residual Error, Measures the Prediction Accuracy of Extended Dynamic Mode Decomposition. *IEEE Control Syst. Lett.* **2023**, *7*, 649–654. [CrossRef]
99. Boots, B.; Gordon, G.J.; Siddiqi, S.M. A Constraint Generation Approach to Learning Stable Linear Dynamical Systems. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 3–6 December 2007; Volume 20, pp. 1329–1336.
100. Umenberger, J.; Manchester, I.R. Specialized Interior-Point Algorithm for Stable Nonlinear System Identification. *IEEE Trans. Autom. Control* **2019**, *64*, 2442–2456. [CrossRef]
101. Gillis, N.; Mehrmann, V.; Sharma, P. Computing the nearest stable matrix pairs. *Numer. Linear Algebra Appl.* **2018**, *25*, e2153. [CrossRef]
102. Bevanda, P.; Beier, M.; Kerz, S.; Lederer, A.; Sosnowski, S.; Hirche, S. Diffeomorphically Learning Stable Koopman Operators. *IEEE Control Syst. Lett.* **2022**, *6*, 3427–3432. [CrossRef]
103. Fan, F.; Yi, B.; Rye, D.; Shi, G.; Manchester, I.R. Learning stable Koopman embeddings for identification and control. *Automatica* **2026**, *189*, 113003. [CrossRef]

104. Sakib, S.A.; Pan, S. Learning Noise-Robust Stable Koopman Operator for Control With Hankel DMD. *IEEE Trans. Control Syst. Technol.* **2026**, *34*, 139–153. [CrossRef]
105. Zhang, C.; Ma, H.; Cheng, Z.; Ren, H. A Koopman-DMPC Strategy for Nonlinear Cyberphysical Systems Under Bounded Communication Delays. *IEEE Trans. Syst. Man. Cybern. Syst.* **2026**, *56*, 5595–5605. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.