Data-Driven Koopman-Based Model Predictive Control for a Three-Tank Hydraulic System

Wilder Hernandez Manosalva, Harrynson Ramirez-Murillo, and Duvan Tellez-Castro

Abstract— **This paper presents an applied study of data-driven Model Predictive Control (MPC) based on the Koopman operator framework for a three-tank hydraulic benchmark. The central contribution is a compact, physics-informed lifting strategy: observable functions are chosen directly from Torricelli's law governing turbulent orifice flow, yielding a seven-dimensional Extended Dynamic Mode Decomposition (EDMD) model that captures the dominant nonlinearities with fewer basis functions than generic dictionaries. The resulting Koopman-MPC replaces the nonconvex optimization of nonlinear MPC (NMPC) with a convex quadratic program, achieving comparable tracking accuracy (0.65 cm vs. 0.64 cm mean error) while reducing the average per-step computation time by $36.7\times$ (50 ms vs. 1807 ms in Python/SciPy). A Moving Horizon Estimator (MHE) operating on the full nonlinear model reconstructs the unmeasured tank level with accuracy comparable to the 2 mm measurement noise floor. These results provide a quantitative benchmark for Koopman-based predictive control on a nonlinear hydraulic system with bidirectional inter-tank coupling and regime-dependent flow transitions.**

Index Terms— **Data-driven control, hydraulic system, Koopman operator, model predictive control, nonlinear systems.**

I. Introduction

Model-based control strategies built on physical principles offer high fidelity but incur significant computational costs when applied to nonlinear processes. In each sampling period, NMPC solves a nonconvex constrained optimization problem whose cost grows as the prediction horizon becomes large and the model becomes more complex [1], [2]. This makes real-time NMPC difficult to deploy on embedded platforms with limited computational resources, a persistent challenge in modern industrial automation.

Hydraulic systems with coupled tanks are a canonical example of this difficulty. Their dynamics are governed by Torricelli's law [3], in which turbulent outflow through an orifice is proportional to the square root of the pressure head. Valve interactions and laminar-to-turbulent flow transitions introduce additional nonlinearities, so that a single linear model cannot accurately represent the system across a wide operating range [4], [5].

Data-driven modeling offers a practical path forward [6]–[8]. Among these approaches, the Koopman operator framework is particularly suited to control design [9], [10]: it lifts nonlinear dynamics into a higher-dimensional space of observable functions where the evolution is linear, converting the nonconvex NMPC problem into a convex quadratic program solvable in real time [11]–[13]. Its practical value depends critically on the choice of observables [9], [11]; *physics-informed* dictionaries that encode known system structure yield more compact and interpretable models [14], [15]. Koopman-MPC has been validated on hydraulic benchmarks including quadruple-tank and interconnected configurations [16]–[18].

The present paper applies this framework to the three-tank benchmark, making the following contributions. A seven-dimensional physics-informed lifting map is derived directly from Torricelli's law, using square-root terms $\sqrt{h_i}$ to linearize the dominant outlet-flow nonlinearity and product terms $h_i h_j$ to capture inter-tank coupling. This dictionary is more compact than the twelve-dimensional generic alternative of [17] and carries a clear physical rationale for each basis function; the design principle extends to any Torricelli-type hydraulic network. A quantitative per-step computational comparison with NMPC under identical conditions is provided, yielding a measured $36.7\times$ speedup. An MHE is included to reconstruct the unmeasured level h_2 at the noise floor ($\sigma = 2$ mm), and the three-tank topology — with bidirectional inter-tank flow and regime-dependent transitions — constitutes a more demanding test than the quadruple-tank configuration of prior work. Although the EDMD procedure, Koopman-MPC formulation, and physics-informed observable concept are established methods, the contribution lies in their careful co-design and systematic evaluation on this benchmark.

The remainder of the paper is organized as follows. Section II develops the three-tank model; Section III reviews the Koopman operator and MPC; Section IV presents the control design; Section V reports simulation results; Section VI provides physical insight and deployment analysis; and Section VII concludes with future directions.

II. Three-Tank Hydraulic System Model

The three-tank hydraulic benchmark consists of three identical upright cylindrical tanks (Fig. 1). Tank 1 receives a pump-controlled inflow q_{i1} and discharges through a bottom orifice while exchanging flow with Tank 2 through a coupling valve. Tank 2 is an intermediate node that receives flow from both adjacent tanks and also discharges through its own orifice. Tank 3 receives a second pump-controlled inflow q_{i3} and is coupled back to Tank 2 through a second valve.

The system state vector is $\mathbf{x} = [h_1, h_2, h_3]^T$, where $h_i \in [0, 0.55]$ m denotes the water level in tank i. The control inputs are the normalized pump commands $\mathbf{u} =$

All authors are with Faculty of Engineering, Universidad Distrital Francisco José de Caldas, Bogotá D.C., Colombia {wshernandezm,hramirezm,duatellezc}@udistrital.edu.co

$[u_1, u_3]^T \in [0,1]^2$, producing inlet flows $q_{i1} = q_{\max} u_1$ and $q_{i3} = q_{\max} u_3$.

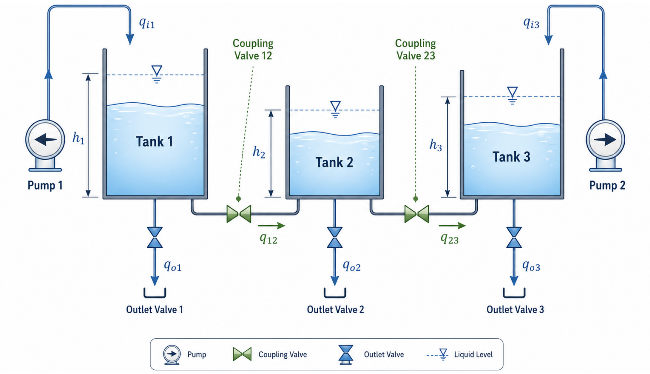

Fig. 1. Schematic of the three-tank hydraulic benchmark. Pump 1 and Pump 2 supply inflows q_{i1} and q_{i3} to Tank 1 and Tank 3, respectively. Inter-tank coupling valves govern bidirectional flows q_{12} and q_{23}, while outlet valves discharge q_{o1}, q_{o2}, and q_{o3}.

Applying conservation of mass to each tank yields:

$$\dot{h}_1 = \frac{1}{A_{\text{tank}}}\big(q_{i1} - q_{12} - q_{o1}\big), \tag{1}$$

$$\dot{h}_2 = \frac{1}{A_{\text{tank}}}\big(q_{12} - q_{23} - q_{o2}\big), \tag{2}$$

$$\dot{h}_3 = \frac{1}{A_{\text{tank}}}\big(q_{i3} + q_{23} - q_{o3}\big). \tag{3}$$

The outlet discharge of tank i follows Torricelli's law [3]:

$$q_{oi}(h_i) = \begin{cases} \alpha_{oi} A_{oi} \sqrt{2 g h_i}, & h_i > h_{\min}, \\ 0, & h_i \le h_{\min}, \end{cases} \tag{4}$$

where α_{oi} is the vena-contracta discharge coefficient, A_{oi} is the orifice area, and $h_{\min} = 0.01$ m prevents numerical singularities. The square-root dependence on pressure head, which implies $\dot{h}_i \approx -(\alpha_{oi} A_{oi} \sqrt{2g}/A_{\text{tank}})\sqrt{h_i}$, is the dominant nonlinearity of the system and directly motivates the observable selection in Section IV.

The inter-tank valves exhibit smooth laminar-to-turbulent transitions. The local Reynolds number characterizing the flow regime is

$$\lambda_{ij} = D_{ij}\,\frac{\rho}{\eta}\,\sqrt{2g|h_i - h_j|}, \tag{5}$$

where D_{ij} is the hydraulic diameter of the connecting valve and $v_{ij} = \sqrt{2g|h_i - h_j|}$ is the characteristic flow velocity estimated by Torricelli's law, so that $\lambda_{ij} = D_{ij}\rho v_{ij}/\eta$ is the standard local Reynolds number. The flow contraction coefficient transitions smoothly as

$$\alpha_{ij}(\lambda_{ij}) = \frac{\alpha_{ij,0}}{e}\tanh\!\Big(\frac{2\lambda_{ij}}{\lambda_{c,ij}}\Big), \tag{6}$$

and the resulting bidirectional coupling flow is

$$q_{ij} = \alpha_{ij}(\lambda_{ij})\,A_{ij}\sqrt{2g|h_i - h_j|}\cdot \operatorname{sgn}(h_i - h_j). \tag{7}$$

In compact notation the model reads $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$, $\mathbf{y} = C\mathbf{x}$, with

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{8}$$

reflecting the partial measurement scenario in which only h_1 and h_3 are directly measured while h_2 must be estimated.

III. Koopman-Based Predictive Control Framework

MPC determines the control input by solving a finite-horizon constrained optimization problem at each sampling instant [1], [19]. When the prediction model is linear, the resulting optimization is a convex quadratic program. When the model is nonlinear, the program is generally nonconvex and computationally expensive.

The Koopman operator provides an alternative representation in which nonlinear dynamics evolve linearly in a higher-dimensional space of observable functions [9]. For a discrete-time system $x_{k+1} = T(x_k)$, the Koopman operator $\mathcal{K}$ acts on observable functions $\psi : \mathcal{X} \to \mathbb{R}$ as

$$(\mathcal{K}\psi)(x) = \psi(T(x)). \tag{9}$$

Defining a finite-dimensional lifting map $\Psi(x) = [\psi_1(x), \ldots, \psi_{n_z}(x)]^T$, the dynamics of a controlled system are approximated in lifted coordinates as

$$\Psi(x_{k+1}) \approx \mathbf{A}\Psi(x_k) + \mathbf{B}u_k, \tag{10}$$

where $\mathbf{A} \in \mathbb{R}^{n_z \times n_z}$ and $\mathbf{B} \in \mathbb{R}^{n_z \times n_u}$ are constant matrices [10]. Note that the control input u_k enters exclusively through $\mathbf{B}$; it does not belong to the lifting map Ψ, which is defined solely as a function of the state x.

The matrices $\mathbf{A}$ and $\mathbf{B}$ are identified from data using EDMD [11]. Given a dataset of M state-input-successor triplets $\{(x_k, u_k, x_{k+1})\}_{k=1}^{M}$, the lifted linear model is obtained by solving a regularized least-squares problem [12]:

$$[\mathbf{A}\ \mathbf{B}] = \arg\min_{\mathbf{K}} \sum_{k=1}^{M} \big\|\Psi(x_{k+1}) - \mathbf{K}[\Psi(x_k); u_k]\big\|_2^2 + \gamma\|\mathbf{K}\|_F^2, \tag{11}$$

where $\mathbf{K} = [\mathbf{A}\ \mathbf{B}] \in \mathbb{R}^{n_z \times (n_z + n_u)}$ and $\gamma > 0$ is a Tikhonov regularization parameter.

IV. Koopman-MPC Design

The quality of a Koopman lifting depends fundamentally on the choice of observable functions. Rather than relying on generic polynomial or radial-basis-function dictionaries, the proposed approach exploits prior knowledge of the hydraulic system dynamics. Torricelli's law (Eq. (4)) shows that outlet flows depend on the square root of the water levels: $q_{oi} \propto \sqrt{h_i}$. Consequently, the dominant nonlinearity in the state equations (1)–(3) follows $\dot{h}_i \approx -c_i\sqrt{h_i}$, where $c_i = \alpha_{oi} A_{oi}\sqrt{2g}/A_{\text{tank}}$ is a constant. Introducing the lifted variable $g_i = \sqrt{h_i}$ and differentiating gives $\dot{g}_i = \dot{h}_i/(2\sqrt{h_i}) \approx -c_i/2$, which is approximately constant under typical operating conditions, so the dynamics of g_i are nearly linear in the lifted coordinate and $\sqrt{h_i}$ is a natural observable.

To capture the principal inter-tank coupling nonlinearities, product terms $h_i h_j$ are included. These terms arise from the interaction between tank levels through the coupling valves and are physically interpretable as second-order mixing effects. A constant bias term ($= 1$) is also included to accommodate steady-state offsets and affine components in the dynamics. No other basis functions are needed: the resulting lifting map is defined solely as a function of the state $\mathbf{h}$:

$$\Psi(\mathbf{h}) = \left[\sqrt{h_1},\ \sqrt{h_2},\ \sqrt{h_3},\ h_1h_2,\ h_2h_3,\ h_1h_3,\ 1\right]^\top \in \mathbb{R}^7. \tag{12}$$

This physics-informed dictionary is more compact than generic alternatives reported in the literature (7 versus 12 basis functions in [17]), while maintaining direct correspondence with the dominant terms in the governing equations. Importantly, this design principle generalizes beyond the three-tank system: any hydraulic or fluid network whose flow equations follow Torricelli-type laws can be lifted using the same square-root and product-term strategy.

The lifted linear model is then identified from data using EDMD. The training dataset consists of 50 closed-loop trajectories at 10 reference levels ($h_2^{\text{ref}} \in [0.12, 0.35]$ m), producing 8,300 state-input-successor samples. Solving the regularized least-squares problem (11) yields matrices $\mathbf{A} \in \mathbb{R}^{7\times7}$ and $\mathbf{B} \in \mathbb{R}^{7\times2}$ with RMSE $= 1.7 \times 10^{-5}$ m, confirming excellent one-step prediction accuracy across the full operating envelope.

The EDMD procedure yields the following matrices (6 significant figures), where rows correspond to the seven observables $[\sqrt{h_1}, \sqrt{h_2}, \sqrt{h_3}, h_1h_2, h_2h_3, h_1h_3, 1]^\top$ and columns of $\mathbf{B}$ correspond to pump inputs $[u_1, u_3]^\top$:

$$\mathbf{A} = \begin{bmatrix} +0.9918 & +0.0046 & +0.0021 & +0.0068 & -0.0157 & +0.0060 & +0.0005 \\ +0.0065 & +0.9660 & +0.0274 & +0.0089 & +0.0117 & -0.0182 & -0.0005 \\ +0.0022 & +0.0256 & +0.9706 & -0.0151 & +0.0078 & +0.0100 & +0.0001 \\ +0.0073 & +0.0047 & -0.0123 & +0.9365 & +0.0351 & +0.0352 & -0.0004 \\ -0.0154 & +0.0055 & +0.0097 & +0.0357 & +0.9226 & +0.0252 & +0.0009 \\ +0.0044 & -0.0090 & +0.0046 & +0.0344 & +0.0227 & +0.9468 & -0.0005 \\ +0.0002 & -0.0001 & -0.0001 & -0.0004 & +0.0010 & -0.0004 & +1.0000 \end{bmatrix}, \tag{13}$$

$$\mathbf{B} = \begin{bmatrix} +1.383 \times 10^{-3} & -4.9 \times 10^{-5} \\ -8.0 \times 10^{-6} & +3.3 \times 10^{-5} \\ -1.6 \times 10^{-5} & +1.622 \times 10^{-3} \\ +3.24 \times 10^{-4} & +1.04 \times 10^{-4} \\ +4.3 \times 10^{-5} & +1.40 \times 10^{-4} \\ +3.12 \times 10^{-4} & +4.63 \times 10^{-4} \\ -2.0 \times 10^{-6} & +6.0 \times 10^{-6} \end{bmatrix}. \tag{14}$$

The structure of $\mathbf{A}$ is physically interpretable: the dominant diagonal entries (0.922–0.999) reflect the slow, stable drainage dynamics of each tank, while the off-diagonal terms capture inter-tank coupling. The last diagonal element (≈ 1) corresponds to the constant bias observable, which is nearly invariant by construction. All eigenvalues of $\mathbf{A}$ lie strictly inside the unit disk ($|\lambda|_{\min} = 0.9998$, $|\lambda|_{\min} = 0.887$), confirming that the identified lifted model is asymptotically stable in open loop. The columns of $\mathbf{B}$ show that Pump 1 (u_1) primarily excites the $\sqrt{h_1}$ observable (first row), while Pump 3 (u_3) primarily excites the $\sqrt{h_3}$ observable (third row), consistent with the physical topology of the system.

Using the identified model (10), the Koopman-based MPC solves a convex quadratic program at each sampling instant:

$$\min_{\{\tilde{u}_n\}} \sum_{n=1}^{N} \|\tilde{z}_n - z^{\text{ref}}\|_{\mathbf{Q}}^2 + \sum_{n=0}^{N-1} \|\tilde{u}_n\|_{\mathbf{R}}^2, \tag{15}$$

subject to the lifted dynamics $\tilde{z}_{n+1} = \mathbf{A}\tilde{z}_n + \mathbf{B}\tilde{u}_n$, the initial condition $\tilde{z}_0 = \Psi(x_k)$, and box constraints on inputs and levels. The reference in lifted coordinates is $z^{\text{ref}} = \Psi(h^{\text{ref}})$, where $h^{\text{ref}} = [h_{1,ss},\ h_2^{\text{ref}},\ h_{3,ss}]^T$ is the steady-state level vector. Both $\tilde{z}_0$ and z^{ref} depend only on state measurements; the control input u does not appear in the lifting map.

V. Simulation Results and Validation

The proposed approach is validated through three representative scenarios covering the main operational requirements of hydraulic control systems, plus independent baseline implementations in MATLAB/Simulink and CasADi[1]. All physical parameters match the certified specifications of the three-tank benchmark (Table I).

TABLE I
Three-Tank System Physical Parameters.

Parameter	Symbol	Value
Tank cross-sectional area	A_{tank}	$153.9 \times 10^{\ 4}$ m^2
Maximum pump flow	$q_{\max}$	$75 \times 10^{\ 6}$ m^3/s
Gravitational acceleration	g	9.81 m/s^2
Water density	ρ	997 kg/m^3
Dynamic viscosity	η	$8.9 \times 10^{\ 4}$ Pa·s
Outlet 1 discharge coeff.	α_{o1}	0.0583
Outlet 1 orifice area	A_{o1}	$1.767 \times 10^{\ 4}$ m^2
Outlet 2 discharge coeff.	α_{o2}	0.1039
Outlet 2 orifice area	A_{o2}	$1.043 \times 10^{\ 4}$ m^2
Outlet 3 discharge coeff.	α_{o3}	0.0600
Outlet 3 orifice area	A_{o3}	$1.767 \times 10^{\ 4}$ m^2
Valve 12 area	A_{12}	$0.555 \times 10^{\ 4}$ m^2
Valve 12 diameter	D_{12}	$7.7 \times 10^{\ 3}$ m
Valve 12 nominal coeff.	$\alpha_{12,0}$	0.3038
Valve 12 critical number	$\lambda_{c,12}$	24000
Valve 23 area	A_{23}	$1.767 \times 10^{\ 4}$ m^2
Valve 23 diameter	D_{23}	$15 \times 10^{\ 3}$ m
Valve 23 nominal coeff.	$\alpha_{23,0}$	0.1344
Valve 23 critical number	$\lambda_{c,23}$	29600

Two Python implementations are compared in the three-scenario study: an NMPC solved by L-BFGS-B with $T_s = 0.3$ s, and the Koopman-MPC (hereafter *Koopman*) with the same sampling period and horizon $N = 10$. A separate CasADi/IPOPT NMPC baseline, implemented at $T_s = 2$ s, serves to verify the nonlinear model and establish MHE performance references; it is not included in the per-step timing comparison. All Python simulations were executed on an Intel Core i7 laptop.

The NMPC baseline solves a finite-horizon optimization at each step using fourth-order Runge-Kutta at $T_s = 0.3$ s. The cost function penalizes tracking error on Tank 2 ($w_{h2} = 20000$), secondary levels ($w_{h1h3} = 5.0$), input magnitude ($w_u = 0.1$), and input rate of change ($w_{\Delta u} = 1.0$), with terminal weight $w_P = 50$ and prediction horizon $N = 10$. The L-BFGS-B solver requires on average 1807 ms per control step in Python/SciPy.

[1]Simulation code available at: `https://github.com/StevenHernandez88/Control-Sistema-de-3-tanques`

A. Scenario 1: Set-Point Regulation

The system starts 1.21 cm below the Tank 2 setpoint $h_2^{\text{ref}} = 0.25$ m. Both NMPC and Koopman-B converge within 20 s with steady-state errors of 0.51 cm and 0.54 cm respectively (Fig. 2). Mean tracking errors are virtually identical (0.24 cm vs. 0.25 cm). The Koopman-B average solve time is 57.4 ms per step versus 1955.5 ms for NMPC, yielding a $\mathbf{34.1}\times$ speedup.

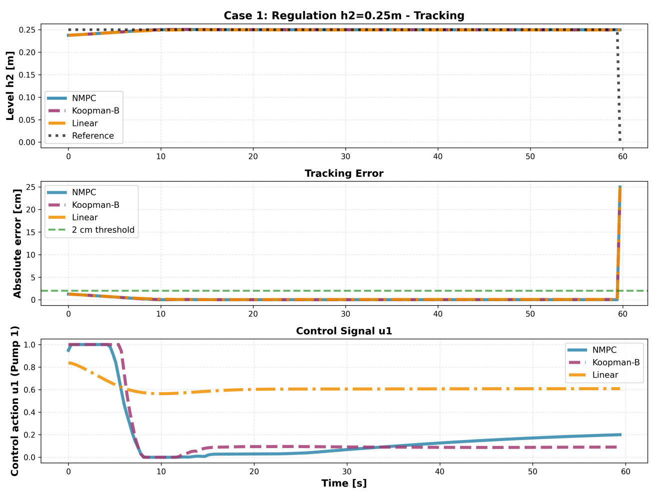

Fig. 2. Set-point regulation at $h_2 = 0.25$ m. Both controllers achieve sub-centimeter steady-state error; Koopman-B executes 34.1× faster than NMPC.

B. Scenario 2: Multi-Step Reference Tracking

The reference ascends monotonically through four 5 cm steps from 0.15 m to 0.30 m over 240 s, covering the complete training envelope. Both controllers exhibit near-identical performance: mean tracking errors of 1.54 cm (NMPC) and 1.55 cm (Koopman-B), with 5 cm peak transients at each step (Fig. 3). Steady-state errors are 0.61 cm and 0.62 cm, respectively. The computational speedup reaches $\mathbf{40.0}\times$ (36.3 ms vs. 1450.5 ms per step).

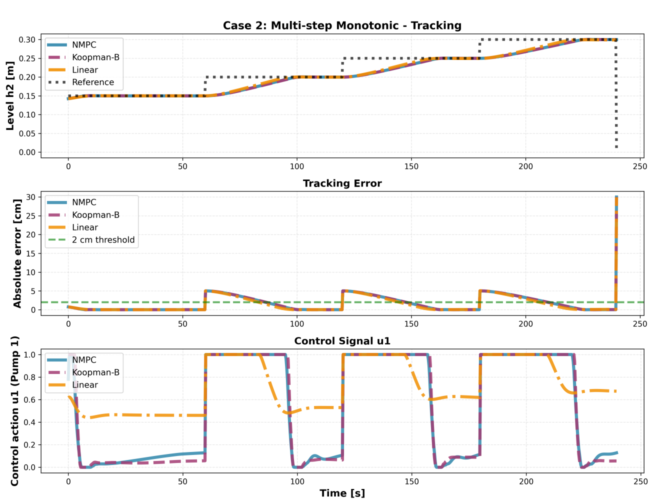

Fig. 3. Multi-step reference tracking from 0.15 m to 0.30 m. Both controllers achieve 1.54–1.55 cm mean tracking error across the full operating envelope.

C. Scenario 3: Disturbance Rejection

An 8 cm step disturbance is injected into Tank 1 at $t = 40$ s. Three strategies are compared: linear gain-scheduling (GS), NMPC, and Koopman-B. The linear GS controller achieves the fastest rejection (0.8 s settling time) at the cost of 27.2% overshoot. NMPC and Koopman-B exhibit larger transient excursions (77.5% and 69.0% peak overshoot, respectively) but achieve lower steady-state errors (0.52 cm and 0.58 cm) and similar settling times (11.0 s and 10.7 s) (Fig. 4). These complementary behaviors reflect the fundamental trade-off between disturbance rejection speed and steady-state precision: gain-scheduling, designed at multiple fixed operating points, can react aggressively to local perturbations, while Koopman-MPC and NMPC optimize a global cost that prioritizes steady-state accuracy. In practice, Koopman-MPC is preferred when precise regulation across a wide operating range is required without manual gain-scheduling design at each operating point. The Koopman-B speedup in this scenario is $\mathbf{36.2}\times$ (55.6 ms vs. 2013.5 ms per step).

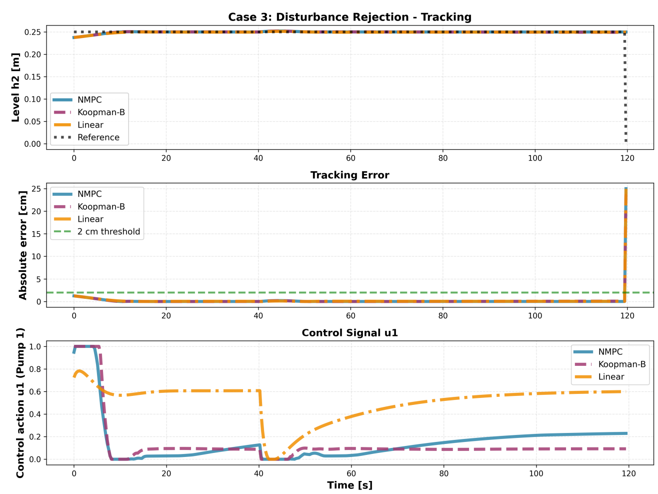

Fig. 4. Disturbance rejection following an 8 cm step in Tank 1 at $t = 40$ s. Linear GS achieves the fastest settling (0.8 s, 27.2% overshoot), while NMPC and Koopman-B achieve lower steady-state errors (0.52 cm and 0.58 cm) with similar settling times (11.0 s and 10.7 s).

TABLE II
QUANTITATIVE COMPARISON OF CONTROL STRATEGIES ACROSS ALL SCENARIOS.

Scenario	Method	Mean Error (cm)	SS Error (cm)	ms/step	Speedup (×)
1: Regulation	NMPC	0.24	0.51	1955.5	1.0 (baseline)
	Koopman-B	0.25	0.54	57.4	34.1×
	Linear GS	0.27	0.55	<1	—
2: Multi-step	NMPC	1.54	0.61	1450.5	1.0 (baseline)
	Koopman-B	1.55	0.62	36.3	40.0×
	Linear GS	1.40	0.68	<1	—
3: Disturbance	NMPC	0.14	0.52	2013.5	1.0 (baseline)
	Koopman-B	0.15	0.58	55.6	36.2×
	Linear GS	0.16	0.55	<1	—
Average	**NMPC**	**0.64**	—	**1807**	1.0
	Koopman-B	**0.65**	—	**50**	**36.7**×

Table II aggregates accuracy and computational cost across all scenarios. Koopman-B matches NMPC accuracy while reducing the average solve time from 1807 ms to 50 ms per step, yielding an average **36.7**× speedup. A compiled implementation of Koopman-MPC would further reduce solve times to the 1–5 ms range, enabling closed-loop operation at 30 Hz or higher on embedded hardware.

D. Reference Control and Estimation Baselines

Three additional implementations in MATLAB/Simulink and CasADi serve to verify the nonlinear model and establish quantitative performance references; all use $T_s = 2$ s and are not directly comparable in timing to the Python/SciPy results above. A linear MPC was first implemented using discrete-time matrices from zero-order-hold discretization at the nominal operating point, with $N = 10$, output weight $q = 10\ \mathrm{m}^{-2}$, and control weight $R_1 = \mathrm{diag}(1, 10)$. The controller satisfies box constraints but exhibits overshoot at the highest reference level (Fig. 6), confirming the limited validity of linear approximations far from the linearization point.

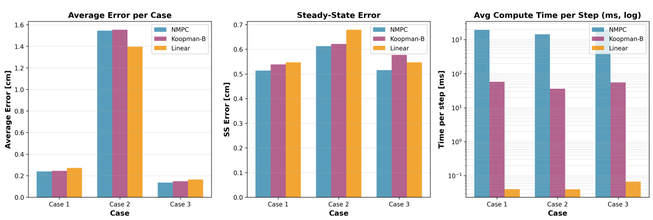

Fig. 5. Global performance summary across all scenarios. Koopman-B achieves NMPC-equivalent accuracy (0.65 cm mean error) with a 36.7× average computational speedup per control step.

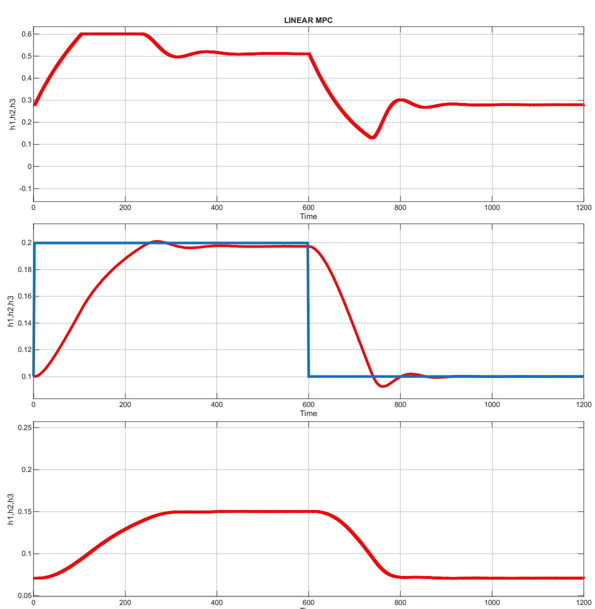

Fig. 6. Linear MPC response in Simulink. Performance degrades noticeably far from the linearization point.

A second baseline employs the full nonlinear model via complete-discretization optimal control in CasADi with IPOPT, using fourth-order Runge-Kutta at $T_s = 2$ s. $T_s = 2$ s. The optimal control problem is

$$(u_n^*) = \arg\min_{(\tilde{u}_n)} \sum_{n=1}^{N} \|\tilde{y}_n - y^{\mathrm{ref}}\|_q^2 + \sum_{n=0}^{N-1} \Big(\|\tilde{u}_n\|_{R_1}^2 + \|\tilde{u}_n - \tilde{u}_{n-1}\|_{R_2}^2 \Big), \tag{16}$$

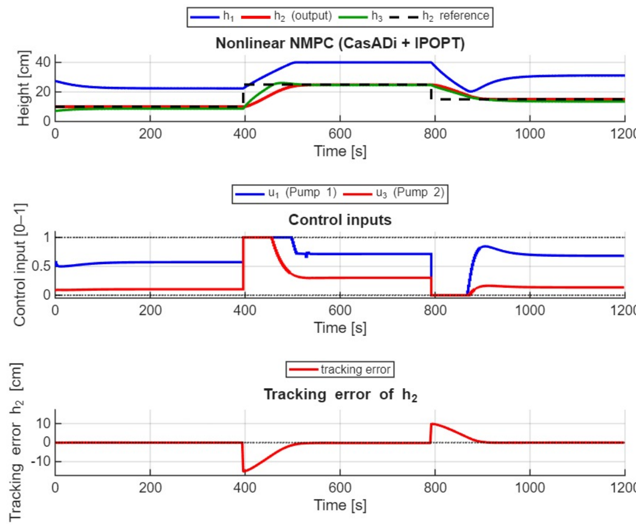

Fig. 7. NMPC response (CasADi + IPOPT, $T_s = 2$ s). Top: levels h_1, h_2, h_3 and reference. Middle: control signals. Bottom: tracking error.

To handle the partial-measurement scenario, an MHE was formulated over $N = 10$ steps, minimizing

$$J_N = \|\tilde{x}_0 - \bar{x}_{k-N}\|_S^2 + \sum_{j=0}^{N-1} \|\tilde{x}_{j+1} - F(\tilde{x}_j, u_{k-N+j})\|_Q^2 + \sum_{j=0}^{N-1} \|y_{k-N+j} - C\tilde{x}_j\|_R^2, \tag{17}$$

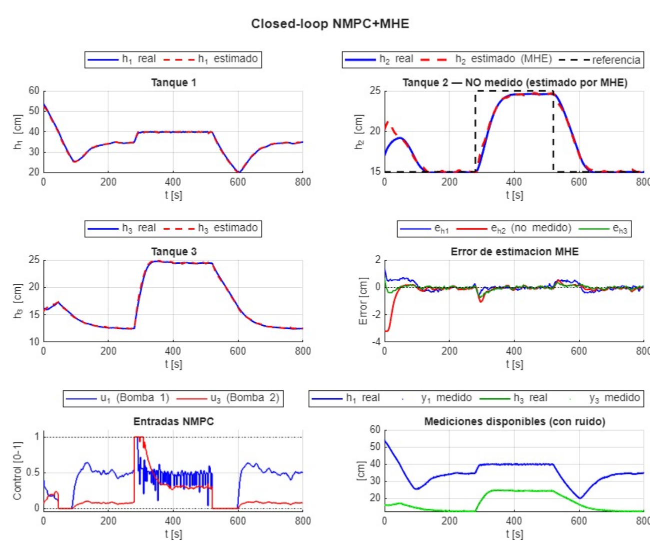

Fig. 8. Closed-loop NMPC+MHE results. Left panels: true states (blue) vs. MHE estimates (red dashed). Top center: unmeasured h_2 and reference. Right panels: estimation error and noisy measurements.

VI. Analysis and Discussion

The effectiveness of the proposed lifting map can be understood from its direct correspondence with the system's dominant nonlinearities. Setting $g_i = \sqrt{h_i}$ and differentiating gives $\dot{g}_i = \dot{h}_i/(2\sqrt{h_i}) \approx$ const, so the dynamics of g_i are nearly linear in the lifted space, directly absorbing the Torricelli nonlinearity. Product terms $h_i h_j$ encode inter-tank coupling in a physically interpretable form, and the constant bias accommodates steady-state offsets. The structure of $\mathbf{A}$ (Eq. (13)) corroborates this reasoning: diagonal entries between 0.922 and 0.999 reflect the slow drainage dynamics of each tank, and all eigenvalues lie strictly inside the unit disk, confirming open-loop asymptotic stability of the lifted model. The computational benefit follows directly from this linear structure. Lifted-space prediction reduces to matrix multiplications, and the MPC becomes a convex quadratic program that converges in far fewer iterations than the nonconvex L-BFGS-B problem of NMPC, yielding a $36.7\times$ average speedup (50 ms vs. 1807 ms per step in Python/SciPy).

A compiled C++ implementation would reduce solve times further to 1–5 ms, enabling deployment on ARM Cortex-M class microcontrollers that cannot run NMPC in real time. Within the training envelope ($h_2 \in [0.12, 0.35]$ m), Koopman-MPC matches NMPC accuracy. Outside this range, performance may degrade; extensions using multiple local Koopman models with gain scheduling [16], neural-network-augmented observables [14], or online re-identification can address this. In disturbance rejection (Scenario 3), gain-scheduling achieves a faster transient response (0.8 s settling time) because it is locally optimized at fixed operating points, whereas Koopman-MPC minimizes a global cost that prioritizes steady-state accuracy — a trade-off the practitioner must weigh according to the application requirements. Formal closed-loop stability guarantees for Koopman-based MPC remain an open and important research direction [10]. Finally, the square-root lifting strategy is not limited to the three-tank system: any hydraulic network governed by Torricelli-type laws ($q \propto \sqrt{\Delta h}$) can adopt the same observable map, providing a systematic physics-guided alternative to generic dictionaries.

VII. Conclusions

This paper presented a Koopman-MPC engineering study for the three-tank hydraulic benchmark. A seven-dimensional physics-informed lifting map derived from Torricelli's law achieves NMPC-level tracking accuracy (0.65 cm vs. 0.64 cm mean error) with a $36.7\times$ reduction in per-step computation (50 ms vs. 1807 ms, Python/SciPy). The identified model is open-loop asymptotically stable (eigenvalues of $\mathbf{A}$ inside the unit disk) and reaches one-step RMSE of 1.7×10^{-5} m. The MHE reconstructs the unmeasured level h_2 at the 2 mm noise floor. The three-tank topology — with bidirectional coupling and regime-dependent flow transitions — provides a more demanding test than the quadruple-tank benchmark of prior work.

Although the EDMD procedure, Koopman-MPC formulation, and physics-informed observable concept are established methods, their co-design yields a compact, interpretable model with a clear physical rationale extendable to any Torricelli-type hydraulic network. Future work will address hardware-in-the-loop validation, wider operating envelopes, robustness under larger noise and model mismatch, and formal closed-loop stability guarantees.

References

[1] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Nob Hill Publishing, 2017.

[2] L. T. Biegler, *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes*. SIAM, 2010.

[3] H. E. Merritt, *Hydraulic Control Systems*. New York: Wiley, 1967.

[4] Q. H. Ngo, T. N. Dang, T. D. Pham, and S. Lee, "Gain scheduling control of a three-tank hydraulic system," in *Proceedings of the 10th Asian Control Conference*, 2015, pp. 1–6.

[5] C. B. Duarte, W. Costa, and D. S. Oliveira, "Adaptive model predictive control of a hydraulic three-tank benchmark system," *Journal of Control, Automation and Electrical Systems*, vol. 31, pp. 1102–1115, 2020.

[6] S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the National Academy of Sciences*, vol. 113, no. 15, pp. 3932–3937, 2016.

[7] R. Strässer, K. Worthmann, I. Mezić, J. Berberich, M. Schaller, and F. Allgöwer, "An overview of Koopman-based control: From error bounds to closed-loop guarantees," *Annual Reviews in Control*, vol. 61, p. 101035, 2026.

[8] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the deepc," in *2019 18th European control conference (ECC)*. IEEE, 2019, pp. 307–312.

[9] A. Mauroy, Y. Susuki, and I. Mezić, Eds., *The Koopman Operator in Systems and Control: Concepts, Methodologies, and Applications*, ser. Lecture Notes in Control and Information Sciences. Springer, 2020, vol. 484.

[10] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.

[11] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, vol. 25, no. 6, pp. 1307–1346, 2015.

[12] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Generalizing Koopman theory to allow for inputs and control," *SIAM Journal on Applied Dynamical Systems*, vol. 17, no. 1, pp. 909–930, 2018.

[13] H. Arbabi and I. Mezić, "Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator," *SIAM Journal on Applied Dynamical Systems*, vol. 16, no. 4, pp. 2096–2126, 2017.

[14] E. Kaiser, J. N. Kutz, and S. L. Brunton, "Data-driven discovery of koopman eigenfunctions for control," *Machine Learning: Science and Technology*, vol. 2, no. 3, p. 035007, 2021.

[15] S. S. K. S. Narayanan, D. Tellez-Castro, S. Sutavani, and U. Vaidya, "Se(3) Koopman-mpc: Data-driven learning and control of quadrotor uavs," in *IFAC-PapersOnLine*, vol. 56, no. 3, 2023, pp. 607–612.

[16] D. Tellez-Castro, C. Garcia-Tenorio, E. Mojica-Nava, J. Sofrony, and A. V. Wouwer, "Data-driven predictive control of interconnected systems using the Koopman operator," *Actuators*, vol. 11, no. 6, p. 151, 2022.

[17] D. S. Aponte and D. Tellez-Castro, "Koopman-based MPC for the quadruple tank system: An experimental comparison," in *2025 IEEE 7th Colombian Conference on Automatic Control (CCAC)*. IEEE, 2025, pp. 1–6.

[18] A. Narasingam and J. S.-I. Kwon, "Application of Koopman operator for model-based control of fracture propagation and proppant transport in hydraulic fracturing operation," *Journal of Process Control*, vol. 91, pp. 25–36, 2020.

[19] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.