Symbolic Kolmogorov-Arnold Networks as a Constant-Time Alternative to LTV-MPC for Embedded Control of Non-Minimum Phase Systems

Adilkhan Salkimbayev
Student, undergraduate
Kazakh-British Technical University
Almaty, Kazakhstan
a_salkimbayev@kbtu.kz

Timur Samigulin
PhD, Scientific Advisor
Kazakh-British Technical University
Almaty, Kazakhstan
t.samigulin@kbtu.kz

***Abstract*—Model Predictive Control (MPC) is the gold standard for constrained multi-variable control, yet its computational burden often precludes deployment on low-cost embedded hardware. In this work, we propose a Symbolic Kolmogorov-Arnold Network (KAN) as a computationally efficient explicit controller. We benchmark both approaches on a Johansson Quad-Tank system exhibiting Non-Minimum Phase (NMP) dynamics. Experiments conducted via Hardware-in-the-Loop (HIL) on an STM32H7 microcontroller demonstrate that the Symbolic KAN approximates the MPC policy with high fidelity while reducing inference time by four orders of magnitude (19ms to 3µs), enabling kilohertz-frequency optimal control on edge devices.**

***Index Terms*—Model Predictive Control, Kolmogorov-Arnold Networks, Symbolic Regression, Embedded AI, Non-Minimum Phase Systems, Explainable AI.**

I. Introduction

Control of non-minimum phase (NMP) systems remains a significant challenge due to the inherent inverse response dynamics [1] characterized by the presence of a non-minimum phase zero in the right half-plane (RHP). Model Predictive Control (MPC) has become the industrial standard for such systems, offering a framework to handle constraints while providing theoretical guarantees on closed-loop stability and optimality [2], [3]. However, the requirement for online quadratic programming (QP) solving creates a bottleneck for embedded deployment.

While Bemporad [4] proposed Explicit MPC to move computation offline, the exponential growth of the solution complexity with respect to the prediction horizon (N) creates a prohibitive memory footprint for complex dynamics. For instance, Alvarado et al. [5] successfully implemented MPC on a quadruple-tank system, but were constrained to short prediction horizons and low sampling frequencies due to the hardware limitations of the era and the complexity of the parametric solution.

While modern primal-dual solvers like OSQP [6] mitigate the solution time compared to earlier methods, they remain iterative and non-deterministic, creating the latency spikes observed in this study. Thus, a gap remains for a controller that offers the optimality of MPC with the constant-time inference of a PID law.

Recent advances in Scientific Machine Learning (SciML) propose replacing heavy solvers with Neural Networks [7]. Some of the more promising attempts at such control have been introduced by Raissi [8], Brunton [9] and Chen [10]. However, standard Multi-Layer Perceptrons (MLPs) suffer from a lack of interpretability ("Black Box" problem) [11], making them risky for safety-critical applications [12].

This paper introduces a methodology using Kolmogorov-Arnold Networks (KANs) [13] to bridge this gap. By exploiting the KAN's ability to learn spline-based activation functions, we extract a symbolic control law that is both computationally efficient and mathematically auditable.

II. Mathematical Modeling

The system under test is the Quadruple-Tank process described by Johansson [14]. The nonlinear dynamics are governed by Bernoulli's principle and Torricelli's equations:

$$\begin{aligned}
\frac{dh_1}{dt} &= -\frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_3}{A_1}\sqrt{2gh_3} + \frac{\gamma_1 k_1}{A_1}v_1 \\
\frac{dh_2}{dt} &= -\frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_4}{A_2}\sqrt{2gh_4} + \frac{\gamma_2 k_2}{A_2}v_2 \\
\frac{dh_3}{dt} &= -\frac{a_3}{A_3}\sqrt{2gh_3} + \frac{(1-\gamma_2)k_2}{A_3}v_2 \\
\frac{dh_4}{dt} &= -\frac{a_4}{A_4}\sqrt{2gh_4} + \frac{(1-\gamma_1)k_1}{A_4}v_1
\end{aligned} \tag{1}$$

To tackle the nonlinear dynamics within a convex optimization framework, we employ Successive Linearization. At each time step k, the nonlinear model (1) is approximated by an Affine Linear Time-Varying (LTV) structure:

$$x_{k+1} = A_k x_k + B_k u_k + d_k \tag{2}$$

where d_k represents the linearization offset. The Jacobian matrix A is defined as:

$$A = \begin{pmatrix} -\frac{a_1}{A_1}\sqrt{\frac{g}{2h_1}} & 0 & \frac{a_3}{A_1}\sqrt{\frac{g}{2h_3}} & 0 \\ 0 & -\frac{a_2}{A_2}\sqrt{\frac{g}{2h_2}} & 0 & \frac{a_4}{A_2}\sqrt{\frac{g}{2h_4}} \\ 0 & 0 & -\frac{a_3}{A_3}\sqrt{\frac{g}{2h_3}} & 0 \\ 0 & 0 & 0 & -\frac{a_4}{A_4}\sqrt{\frac{g}{2h_4}} \end{pmatrix} \tag{3}$$

Matrix B is calculated by Johansson as:

$$B = \begin{pmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2)k_2}{A_3} \\ \frac{(1-\gamma_1)k_1}{A_4} & 0 \end{pmatrix} \tag{4}$$

Matrix C is defined as:

$$C = \begin{pmatrix} k_c & 0 & 0 & 0 \\ 0 & k_c & 0 & 0 \end{pmatrix} \tag{5}$$

Where k_c had been set to 1 $\frac{V}{cm}$ for direct conversion.

III. Control Architecture

The control loop consists of an Extended Kalman Filter (EKF) [15] for state estimation and a Gain-Scheduled Controller (LTV-MPC/Symbolic KAN).

A. Stochastic Estimation (EKF)

Since only the lower tank levels are measured ($y = [h_1, h_2]^T$), an EKF is implemented to reconstruct the full state vector $x = [h_1, h_2, h_3, h_4]^T$.

B. The Teacher: LTV-MPC

The optimal control problem is solved using the OSQP solver [6], a successor of the CVXGEN framework [16]. The cost function minimizes tracking error and control effort:

$$J = \sum_{k=0}^{N-1} (\|x_k - x_{ref}\|_Q^2 + \|u_k\|_R^2) \tag{6}$$

In matrix form, this equation becomes:

$$J = \sum_{k=0}^{N-1} (\|x_k - x_{ref}\|^T Q \|x_k - x_{ref}\| + \|u_k\|^T R \|u_k\|) \tag{7}$$

C. The Student: Symbolic Kolmogorov-Arnold Network

Two Symbolic KANs are structured to approximate the optimal control law $u_{opt} = \pi(x_{est}, x_{ref})$. To accommodate the distinct dynamics of the Non-Minimum Phase (NMP) and Minimum Phase (MP) regimes, we employ a Gain-Scheduled architecture where the active network is switched based on the operating region.

TABLE I
Experimental Parameters for MP and NMP Configurations

Parameter	Symbol	MP	NMP
Valve Ratios	γ_1, γ_2	$0.70, 0.60$	$0.43, 0.34$
Pump Gains	k_1, k_2	$3.33, 3.35$	$3.14, 3.29$
Initial State (cm)	x_0	$[12.4, 12.7, 1.8, 1.4]^T$	$[12.6, 13.0, 4.8, 4.9]^T$
Target State (cm)	x_{ref}	$[10.0, 10.0, 2.0, 2.0]^T$	

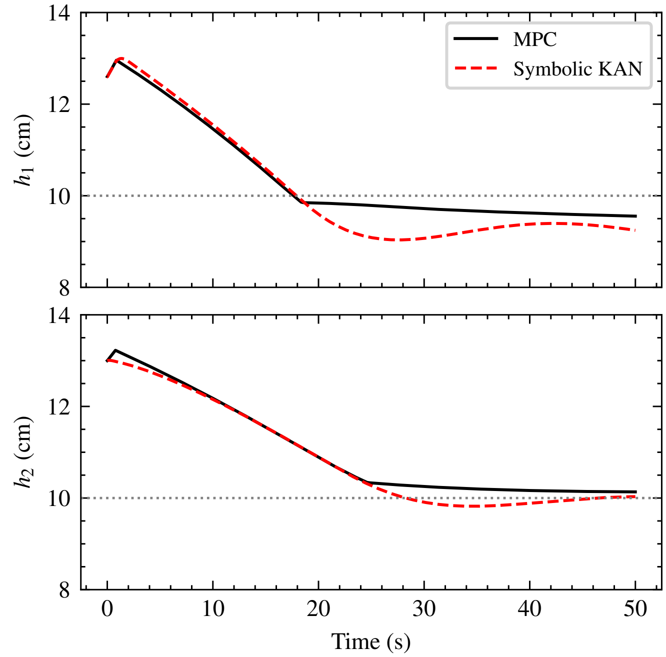

Fig. 1. **Control Performance.** Levels of tanks 1 and 2 throughout the duration of the control loop are shown. MPC exhibits a physics-based, predictive behavior. KAN displays awareness of non-linearity, having approximated the control law.

IV. Experimental Results

A. Simulation Setup

Simulation was performed using the HIL methodology. The Nucleo-H753ZI (Cortex-M7, 480MHz, DP-FPU) communicates with the PC via UART at 921,600 baud.

The KANs were trained offline using the PyKAN framework [13] on a dataset generated by the LTV-MPC. To overcome the computational bottlenecks of standard B-spline grids, we adopted initialization and grid-adaptation strategies derived from FastKAN [17] and AF-KAN [18]. These optimizations, combined with the L-BFGS optimizer [19] being chosen over standard SGD [20], accelerated convergence and facilitated the extraction of compact symbolic formulas. The procedure was followed by pruning to isolate the dominant control interactions, achieving a 4.84% Mean Absolute Error (MAE) on the test set. The resulting tracking performance is illustrated in Fig. 1

B. Experimental Procedure

The simulation employed constants for both MP and NMP systems provided by Johansson, which are shown in Table 1.

The simulation's length had been set to 50 seconds and discretized into intervals of 3 milliseconds each, yielding a total of 16666 data points. The simulation ran in Synchronized Time, with runtime directly calculated in the main program loop and being transmitted over to the Python loop initiating transfer.

The plot of the executed control loop was yielded with NMP system configuration. (Fig. 1)

As evidenced by the control loop simulation,

C. Safety Audit: The "Sign Flip"

During symbolic extraction, the KAN initially identified a control law with a negative gain on the error term for Pump 2 (Fig. 2), creating a positive feedback loop. Due to the interpretable nature of KANs, this was manually corrected to negative feedback prior to C deployment.

Raw Symbolic Output

`... + 53.32934*x_5` $\mathbf{-50.5462} * x\_6$ `- 0.32*x_7 + ...`

Detected Anomaly: $u_{pump2} \propto \mathbf{-50.55} \cdot x_{error}$
(Positive Feedback Loop Identified via Symbolic Regression)

Fig. 2. **The "White Box" Safety Audit.** Symbolic extraction (top) revealed a critical sign inversion. The highlighted term (-50.55) was flagged and manually corrected to ensure negative feedback stability.

V. Conclusion

Finally, these results challenge the prevailing trend of solving control complexity by upgrading hardware. We demonstrate that the bottleneck for deploying advanced optimal control on legacy 40nm lithography (e.g., Cortex-M7) is not the hardware's clock speed, but the algorithmic reliance on iterative online solvers. By shifting the computational burden offline via Symbolic KANs, we validate a hypothesis of 'Industrial Sufficiency': that low-cost, mature microcontroller architectures possess ample compute power for high-frequency nonlinear control, provided the control law is appropriately compressed.

References

[1] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*. John Wiley & Sons, 2005.

[2] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[3] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

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

[5] I. Alvarado, D. Limon, D. M. De La Peña, J. M. Maestre, M. Ridao, H. Scheu, W. Marquardt, R. Negenborn, B. De Schutter, F. Valencia *et al.*, "A comparative analysis of distributed mpc techniques applied to the hd-mpc four-tank benchmark," *Journal of Process Control*, vol. 21, no. 5, pp. 800–815, 2011.

[6] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "Osqp: An operator splitting solver for quadratic programs," *Mathematical Programming Computation*, vol. 12, no. 4, pp. 637–672, 2020.

[7] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger, "Learning-based model predictive control: Toward safe learning in control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, pp. 269–296, 2020.

[8] M. Raissi, "Deep hidden physics models: Deep learning of nonlinear partial differential equations," *Journal of Machine Learning Research*, vol. 19, no. 25, pp. 1–24, 2018.

[9] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Sparse identification of nonlinear dynamics with control (sindyc)," *IFAC-PapersOnLine*, vol. 49, no. 18, pp. 710–715, 2016.

[10] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "Neural ordinary differential equations," *Advances in neural information processing systems*, vol. 31, 2018.

[11] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[12] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, "Concrete problems in ai safety," *arXiv preprint arXiv:1606.06565*, 2016.

[13] Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljačić, T. Y. Hou, and M. Tegmark, "Kan: Kolmogorov-arnold networks," *arXiv preprint arXiv:2404.19756*, 2024.

[14] K. H. Johansson, "The quadruple-tank process: A multivariable laboratory process with an adjustable zero," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 3, pp. 456–465, 2000.

[15] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[16] J. Mattingley and S. Boyd, "Cvxgen: A code generator for embedded convex optimization," in *Optimization and Engineering*, vol. 13, no. 1. Springer, 2012, pp. 1–27.

[17] Z. Li, "Kolmogorov-arnold networks are radial basis function networks," *arXiv preprint arXiv:2405.06721*, 2024.

[18] H.-T. Ta and A. Tran, "Af-kan: Activation function-based kolmogorov-arnold networks for efficient representation learning," *arXiv preprint arXiv:2503.06112*, 2025.

[19] S. Wright, J. Nocedal *et al.*, "Numerical optimization," *Springer Science*, vol. 35, no. 67-68, p. 7, 1999.

[20] H. Robbins and S. Monro, "A stochastic approximation method," *The Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400–407, 1951.