MLSS 2026 — Max Planck Institute, Tübingen, September 2026

SDO

Accelerating Full-Scale NMPC via **Structured Learning Methods**

Perceval Beja-Battais[1,2] • Alain Grossetête[2] • Nicolas Vayatis[1]
[1] Université Paris-Saclay, ENS Paris-Saclay, CNRS, Centre Borelli [2] Framatome, Reactor Process Department
perceval.beja-battais@ens-paris-saclay.fr arxiv.org/abs/2604.05566

Glossary

NMPC Nonlinear Model Predictive Control **OCP** Optimal Control Problem **SDO** Surrogate Dynamics Optimization
BC Behavior Cloning **PWR** Pressurized Water Reactor **DAE** Differential-Algebraic Equations
SRLBC Structured Representation Learning for Behavior Cloning

Motivation

NMPC solves an **OCP** at every control step — computationally prohibitive for high-dimensional, stiff, or multi-time-scale industrial systems.

Problem: How can ML accelerate NMPC while preserving safety guarantees?

Solution: Providing a high-quality initial guess for the optimization solver.

This can be done by: [1, 2]

- **Learning to control:** methods from Reinforcement Learning (RL) to Imitation Learning (IL).
- **Learning the dynamics** to solve a lighter problem, e.g., Surrogate Dynamics Optimization (SDO).

Key bottleneck: the quality of the initialization directly limits convergence speed. Classical strategies (e.g., *shift-initialization*) fail for complex systems, or under abrupt, last-minute changes in operating conditions, cost function, or exogenous inputs. RL strategies fail because they require massive data.

Key Contributions

Idea: We generate high-quality initial guesses by leveraging representation learning **(SRLBC)** for short-term control, and dynamics learning **(SDO)** for long-term control.

- **Structured Representation Learning for Behavior Cloning (SRLBC).**
  SRLBC is an encoder-controller architecture that leverages the physical knowledge of a system in order to enhance performance and interpretability of the network.
  → We demonstrate its efficiency for short-horizon, multi-scale NMPC.
- **Surrogate Dynamics Optimization (SDO).**
  SDO is a methodology combining surrogate dynamics learning with auxiliary optimisation for structured warm-start generation.
  → We show its effectiveness for long-horizon, high-dimensional NMPC.

In these two contributions, we emphasize on **safety**, **data efficiency**, **online budget**, and **industrial validation**.

Problem Statement

NMPC using Single Shooting Transcription: [3]

$$\min_{\mathbf{u}=[u_0,\dots,u_N]^\top} J(u) := \sum_{k=0}^{N} \ell(x(t_k), z(t_k), u_k) \quad (1)$$

such that $0 \geq h(x(t_k), z(t_k), w(t_k), u_k)$,

where $x(t), z(t), w(t), u(t)$ are respectively the differential, algebraic, exogenous, and control variables.

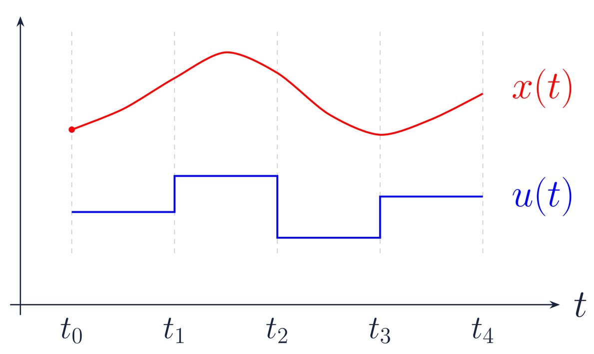


Figure: Single Shooting transcription.

SRLBC for Short-Horizon, Multi-Scale NMPC

Traditional BC aims to learn the solution of (1) using data from the NMPC expert. SRLBC proposes to encode the variables from each timescale into separate embeddings. The embedding then exhibits block-diagonal Jacobian matrix with respect to the state and exogenous variables. Then, SRLBC learns to solve (1) through the decomposition described in the following figure.

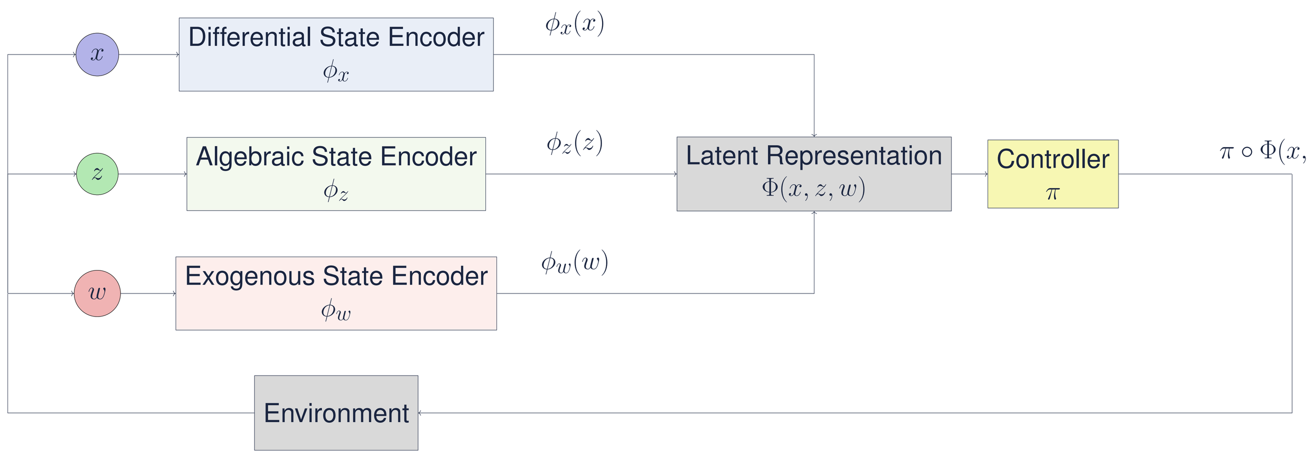
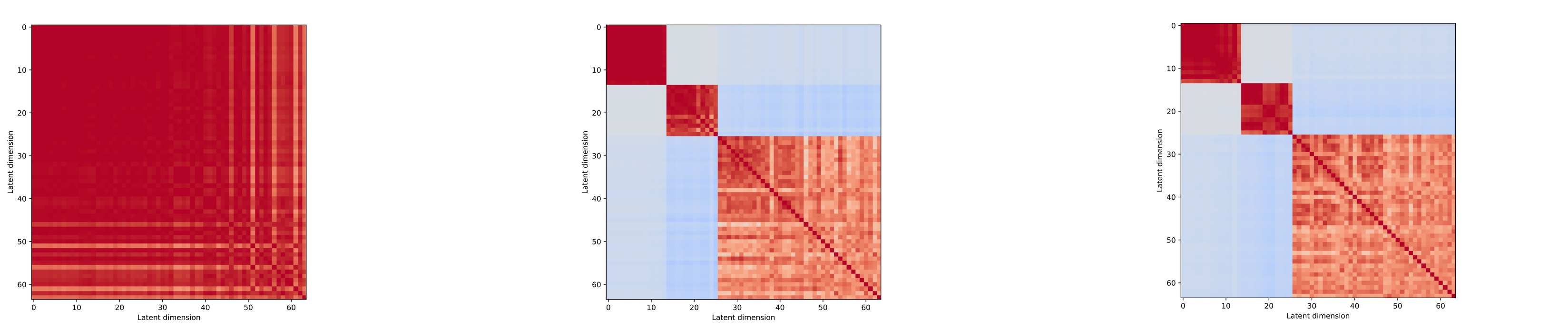


Figure: Proposed multi-stream architecture: we encode the input data in separated latent spaces and the controller π acts on the product latent space. We can further impose independant representations by minimizing a decorrelation loss on the latent space.



(a) Naive embedding $\Phi(x, z, w)$. (b) $\Phi(x, z, w) = (\phi_x(x), \phi_z(z), \phi_w(w))$. (c) $\Phi(x, z, w) = (\phi_x^\perp(x), \phi_z^\perp(z), \phi_w^\perp(w))$.

Figure: Correlation matrix between the representations in each embedding. Naive embedding result in a much more redundant representation. Enforcing decorrelation during training results in more independent representations.

Selected References

[1] Hewing, L. et al., *Learning-based model predictive control: Toward safe learning in control*, Annual Review of Control, Robotics, and Autonomous Systems, 2020.
[2] D. J. Foster et al., *Is behavior cloning all you need? understanding horizon in imitation learning*, Advances in Neural Information Processing Systems, 2024.
[3] M. Diehl et al., *Efficient numerical methods for nonlinear MPC and moving horizon estimation*, Nonlinear model predictive control, 2009.
[4] G. Dupré et al., *Enhanced flexibility of PWRs (Mode A) using an efficient NMPC-based boration/dilution system*, European Control Conference, 2021.

SDO for Long-Horizon, High-Dimensional NMPC

Let $X = [x, z]^\top$. We introduce SDO: a combination of dynamics learning and surrogate optimization to provide an initial guess for complex, high-dimensional NMPC.

Phase 1 — Offline Training of Surrogate Dynamics Model
Train $\hat{X} \approx X$ on sampled trajectories

↓

Phase 2 — Online Optimisation using Surrogate Dynamics
Solve lightweight NLP (2) with m_θ [$\leq 5\%$ budget]

↓

Phase 3 — Full-Scale Refinement **Safety filter**
Re-optimise in original NLP (1) with $\hat{u}$ as warm-start

Lightweight optimization problem with soft constraints:

$$\hat{\mathbf{u}} = \arg\min_{\mathbf{u}} \sum_{k=0}^{N-1} \left[\ell(\hat{X}(t_k), u_k) + \nu \,\|h^+(\hat{X}(t_k), u_k)\|\right] \quad (2)$$

Proposition (simplified). Let $M = \max_{u\in\mathcal{U}} \|X^{[1:N]} - \hat{X}^{[1:N]}\|$.

If $X \mapsto \sum_{k=0}^{N} \ell(X_k, u_k)$ is K_J-Lipschitz, then:

$$J(\hat{u}) \leq J(u^*) + 2K_J M,$$

and if J is μ-strongly convex,

$$\|u^* - \hat{u}\| \;\leq\; 2\sqrt{\frac{K_J\, M}{\mu}}.$$

$\Longrightarrow$ Increasing surrogate precision *directly* controls warm-start quality. Monotone relationship between validation MSE and acceleration metrics are observed empirically.

Industrial Benchmark: Nuclear Reactor

Variable	Symbol	Short-horizon	Long-horizon
Differential state	$x(t)$	$\mathbb{R}^{13}$	$\mathbb{R}^{14}$
Algebraic state	$z(t)$	$\mathbb{R}^{9}$	$\mathbb{R}^{8}$
Control input	$u(t)$	$\mathbb{R}^{3}$	$\mathbb{R}^{100+}$
Exogenous input	$w(t)$	$\mathbb{R}^{30}$	$\mathbb{R}^{1000+}$
Horizon length	T	$30\,\mathrm{min}$	$24\,\mathrm{h}$

Nonlinear Constraint: $|\Delta\mathrm{AO}(P)| \leq 5\%$ (safety limit at all times)

A **Pressurised Water Reactor (PWR)** under 24-hour load-following control — a high-dimensional, stiff DAE system with safety constraints [4].

DAE model ($n_z = 6$ axial nodes):

$$\frac{dI}{dt} = \gamma_I P - \lambda_I I, \qquad \frac{dX}{dt} = \gamma_X P + \lambda_I I - (\lambda_X + \sigma_X P) \odot X,$$

$$0 = \rho(t) \odot P(t) + DM_{\mathrm{exch}} P(t), \qquad 0 = \sum_{i=1}^{n_z} P_i(t) - w(t).$$

For short-horizon, we further have $\frac{dC_{\mathrm{b}}}{dt} = f(u(t))$ and $\frac{dh_{\mathrm{cr}}}{dt} = g(w(t))$, while for long-horizon, we have $\frac{dh_{\mathrm{cr}}}{dt} = u(t)$ and $0 = T_{\mathrm{in}}(t) - T_{\mathrm{ref}}(w(t))$.

Quantitative Results

Short-horizon Benchmark: Integral cost, BC loss, and constraint violation metrics on roll-outs. When warmstarting a convex problem, cost ratio and gap to control are expected to converge respectively towards 1 and 0.

Embed. (Size)	Without Warmstart				With Warmstart		
	Cost Ratio	Relative Gap to Control	Feasibility	Comput. Time (s)	Cost Ratio	Feasibility	Comput. Time (s)
BC Baseline (64)	1.12 ± 0.12	$(8.0 \pm 2.8) \cdot 10^{-3}$	94.1%	20.4 ± 0.6	$\mathbf{1.00 \pm 0.01}$	$\mathbf{100.0\%}$	292.3 ± 17.9
BC Baseline (128)	1.15 ± 0.15	$(9.8 \pm 3.9) \cdot 10^{-3}$	92.2%	20.5 ± 0.6	$\mathbf{1.00 \pm 0.01}$	99.6%	293.8 ± 17.6
SRLBC (64)	1.08 ± 0.11	$(7.8 \pm 3.5) \cdot 10^{-3}$	$\mathbf{97.8\%}$	20.5 ± 0.6	$\mathbf{1.00 \pm 0.01}$	$\mathbf{100.0\%}$	$\mathbf{285.7 \pm 16.1}$
SRLBC (64) + Decor.	$\mathbf{1.07 \pm 0.08}$	$\mathbf{(6.8 \pm 2.5) \cdot 10^{-3}}$	95.3%	20.5 ± 0.6	$\mathbf{1.00 \pm 0.01}$	$\mathbf{100.0\%}$	293.0 ± 17.3
Optimal Policy – π_{MPC}	*1.00*	*0*	*100.0%*	*348.3 ± 14.6*	*1.00*	*100%*	*348.3 ± 14.5*

Long-Horizon Benchmark: The optimal policy π_{MPC} is not available: we compare the initial guesses quality and after a fixed-budget optimization process. Computation time is similar for each method (duration of the optimization using surrogate model is marginal): total budget of 10 gradient descent iterations, 1 iteration $\approx$ 1 min. Full-scale optimization is more robust using SDO.

Method	Initial Guess			After 5 iters.			Total budget		
	J/J_{best}	Avg Rank	Feasability	J/J_{best}	Avg Rank	Feasability	J/J_{best}	Avg Rank	Feasability
Cold-start	53.7 ± 64.5	3.9	0%	13.3 ± 8.4	3.7	80%	8.9 ± 5.7	3.7	$\mathbf{100\%}$
Shift-Init	16.5 ± 10.5	3.1	40%	4.3 ± 4.4	2.8	90%	3.8 ± 3.3	2.8	$\mathbf{100\%}$
BC	$\mathbf{1.0 \pm 1.6}$	$\mathbf{1.4}$	$\mathbf{50\%}$	2.0 ± 1.5	1.9	$\mathbf{95\%}$	3.1 ± 1.5	2.1	$\mathbf{100\%}$
SDO	4.1 ± 4.1	1.6	45%	$\mathbf{1.0 \pm 1.2}$	$\mathbf{1.5}$	$\mathbf{95\%}$	$\mathbf{1.0 \pm 0.5}$	$\mathbf{1.4}$	100%

Conclusion & Perspectives

NMPC warmstart using SRLBC and SDO position **ML-based controllers as NMPC accelerators**, not substitutes, supporting the complementarity of data-driven and model-based control in safety-critical applications.

Future directions:

- Designing a trust-region surrogate dynamics optimization solver.
- Extension to stochastic settings and multi-day planning.
- Theoretical convergence under weak assumptions.
- Broader comparison of warm-start methods across industrial benchmarks.