CAUSAL GP-MPC: WHERE STRUCTURE, SAFETY, AND ONLINE LEARNING MEET FOR ROBUST ACCELERATOR CONTROL

S. Hirlaender*, O. Mironova†, S. Trausner, University of Salzburg, Salzburg, Austria
L. Fischl, MedAustron GmbH, Wiener Neustadt, Austria
T. Gallien, JOANNEUM RESEARCH, Graz, Austria
L. Grech, University of Malta, Msida, Malta

Abstract

Robust accelerator control increasingly relies on data-driven optimisation, yet balancing adaptability with operational safety remains challenging. Simulation-driven physics-informed reinforcement learning (RL) relies on soft constraints without firm safety guarantees, and classical matrix inversion becomes suboptimal under noise and hard actuator limits. Using the AWAKE electron beam steering task at CERN as a high-fidelity benchmark, we formulate beam steering as a stochastic control problem in a linear Markov Decision Process with continuous state and action spaces and realistic constraints, and compare classical inversion, Model Predictive Control (MPC), data-driven Gaussian-Process MPC (GP-MPC) and RL. Our main contribution is a Causal GP-MPC scheme that embeds the beamline's causal layout directly into the GP prior and kernel design. This structural inductive bias reduces model complexity, improves conditioning, and enables accurate multi-step prediction from limited data. In simulation studies based on the measured response matrix, Causal GP-MPC achieves performance comparable to MPC with the perfect model while requiring only observational data. It outperforms unstructured GP-MPC and RL baselines in sample efficiency, noise robustness, and online optimisation time. Taken together, these results demonstrate that causally structured learning offers a promising pathway toward data-efficient, interpretable, and deployable control strategies for complex accelerator systems.

INTRODUCTION

Particle accelerators are complex, high-dimensional systems where optimal performance often depends on precise alignment and trajectory control. While classical model-based control relies on accurate physics models (e.g., response matrices), these models often drift or suffer from measurement noise, leading to suboptimal performance. Conversely, Reinforcement Learning [1] promotes adaptability but typically requires prohibitive amounts of interaction data and often lacks safety guarantees during training.

To address this gap, we propose **Causal GP-MPC**, a method that hybridises the sample efficiency of Bayesian inference with the constrained planning of Model Predictive Control. By explicitly encoding the physical causal structure of the accelerator beamline into the learning kernel, we significantly reduce the search space, enabling rapid online learning while hard actuator limits and multi-step planning improve robustness.

The AWAKE Experiment

The Advanced WAKEfield Experiment (AWAKE) at CERN is the first facility to demonstrate proton-driven plasma wakefield acceleration [2]. High-intensity 400 GeV proton bunches from the Super Proton Synchrotron drive wakefields in a 10 m plasma cell, accelerating injected electrons to GeV energies in a single stage. Precise control of the electron beam trajectory entering the plasma is critical: transverse misalignments of even a few hundred microns can disrupt the trapping process and degrade beam quality. The electron beamline comprises ten dipole correctors interleaved with ten Beam Position Monitors (BPMs), whose sequential arrangement imposes a natural causal ordering exploited by our method.

MATHEMATICAL FRAMEWORK

Beam Steering as a Constrained MDP with Noisy Observations

We model beam steering as an MDP [3] $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, R, \rho_0)$ with continuous 10-dimensional state and action spaces and linear transition dynamics.

State and action spaces. States $s_k \in \mathcal{S} \subset \mathbb{R}^{10}$ are normalised transverse beam positions at ten BPMs; actions $a_k \in \mathcal{A} = [-1, 1]^{10}$ are incremental changes to ten dipole corrector strengths, box-constrained by the physical actuator range.

System dynamics and transition kernel P. The response matrix $B \in \mathbb{R}^{10\times 10}$, derived from lattice-optics simulations (MAD-X [4]), maps corrector kicks to BPM displacements. Because elements are arranged sequentially along the beamline, B is *lower-triangular*: corrector j affects only downstream BPMs $i \geq j$. The state evolves deterministically as

$$s_{k+1} = s_k + B\, a_k, \tag{1}$$

so the transition kernel is $P(s' \mid s, a) = \delta(s' - s - Ba)$. The controller observes $o_k = s_k + v_k$ with $v_k \sim \mathcal{N}(0, \sigma^2 I_{10})$ (I_{10}: 10×10 identity matrix). Strictly this is a POMDP, but since each controller acts on o_k directly (KalmanQP on its filtered estimate), we retain the MDP formulation, consistent with common practice [1].

Episodic structure. Each episode starts from a random s_0 within the aperture ($\|s_0\|_\infty \leq 1$). The safety constraint $\|s_k\|_\infty \leq 1$ must hold for all k; violation triggers beam

* Equal contribution. simon.hirlaender@plus.ac.at
† Equal contribution

loss and termination. Episodes also end upon success ($\mathrm{RMS}(s_k - s_{\text{target}}) < 0.1$, i.e. $r > -0.1$) or truncation at $T_{\max}$ steps.

Control objective. The goal is to find a policy π that maximises the expected undiscounted return of the negative RMS orbit error subject to dynamics, actuator bounds, and safety constraints:

$$\begin{aligned} \max_{\pi} \quad & \mathbb{E}_{\pi}\left[\sum_{k=0}^{T_{\max}-1} r_k\right], \quad r_k = -\sqrt{\tfrac{1}{10}\textstyle\sum_{i=0}^{9}(s_{i,k} - s_{\text{target},i})^2} \\ \text{s.t.} \quad & s_{k+1} = s_k + B\,a_k, \quad a_k = \pi(o_k), \\ & a_k \in [-1,1]^{10}, \quad \|s_k\|_\infty \le 1, \quad \forall\, k. \end{aligned} \tag{2}$$

Baseline Controllers

We compare against RMI, KalmanQP, PPO, and unstructured GP-MPC (structural ablation).

Response Matrix Inversion (RMI). The simplest baseline applies $a_k = -B^{-1}(o_k - s_{\text{target}})$. RMI is memoryless (< 1 ms) but maps every noisy observation directly to a corrector update. Actions exceeding $\|a_k\|_\infty > 1$ are rescaled.

KalmanQP (MPC). KalmanQP pairs a Kalman filter [5] (dynamics (1), $R = \sigma^2 I$, $Q = (0.1\sigma)^2 I$) with a one-step QP ($H = 1$ [6, 7]) minimising $\|\hat{s}_k^+ + B\,a - s_{\text{target}}\|_2^2$ s.t. $a \in [-1,1]^{10}$ [8]. The single-step horizon suffices because the exact linear model enables deadbeat correction; noise degradation arises from state-estimation error growing with σ, not from insufficient planning depth.

Proximal Policy Optimisation (PPO). PPO [9] is the model-free baseline: an MLP actor-critic trained for 10^6 steps via the clipped surrogate objective (Stable-Baselines3 [10]).

GP-MPC. As a structural ablation, plain GP-MPC [11, 12] uses the same online learning and planning framework but each of the $n = 10$ GPs receives the full $D = 20$-dimensional input $z_k = (s_k, a_k) \in \mathbb{R}^{20}$, including physically impossible acausal couplings, requiring more data to converge.

Causal GP-MPC

GP dynamics learning. Building on prior work applying GP-MPC to accelerator steering [11], the GP-MPC controller learns the one-step state increment $\Delta s_k \triangleq s_{k+1} - s_k$ from observed transitions. Each of $n = 10$ independent GPs (one per BPM) uses a zero-mean prior with an Automatic Relevance Determination (ARD) scaled radial-basis-function (RBF) kernel, where lengthscales $\ell_{i,j}$ and output scale $\sigma_{f,i}^2$ are learned online via marginal likelihood optimisation [13]. Both GP-MPC variants use a zero-mean prior and learn Δs_k entirely from observed transitions without incorporating the nominal response matrix B. This keeps the method fully model-free; using B as a physics-informed mean function could further improve sample efficiency but would reintroduce a model dependency. The **Causal GP-MPC** variant restricts the i-th GP to its upstream **Causal Cone**

$$\mathcal{I}_i = \{s^{(0)}, \ldots, s^{(i)}\} \cup \{a^{(0)}, \ldots, a^{(i)}\}, \tag{3}$$

reducing the input dimensionality from $D = 20$ to $d_i = 2(i+1)$, which significantly shrinks the hyperparameter space and improves covariance matrix conditioning. Prediction uncertainty is propagated via moment matching [12, 14] for multi-step planning.

Receding-horizon planning. At each step, an L-BFGS-B optimiser [15] minimises the accumulated multi-step tracking cost $\sum_{h=0}^{H-1} \|\hat{s}_{k+h} - s_{\text{target}}\|^2$ over $H = 4$ predicted steps, subject to hard actuator limits $a_h \in [-1,1]^{10}$ at every horizon step.

RESULTS

Sample Efficiency. All experiments use $T_{\max} = 50$. The training budget N denotes the number of transitions accumulated online during closed-loop operation under the controller's own exploration policy; GP hyperparameters are updated after every new transition via incremental marginal-likelihood optimisation. Figure 1 compares prediction accuracy between causal and unstructured GP-MPC. Causal GP-MPC converges ≈2.2× faster in one-step prediction root-mean-square error (RMSE), with the largest gains for upstream BPMs whose causal cones are sparsest. Figure 2 displays reward dynamics at $N = 50$; Fig. 3 sweeps over N, showing that at $N = 40$ Causal GP-MPC already reaches ≈−0.25 cumulative reward, while unstructured GP-MPC requires $N = 100$ for comparable performance. PPO requires $O(10^6)$ transitions to converge (Table 1), confirming that model-free RL is ∼25,000× less sample-efficient than Causal GP-MPC. This reduction in sample complexity is essential for online commissioning where beam time is limited.

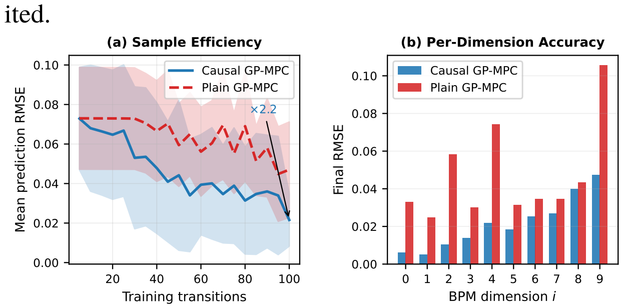

Figure 1: Causal GP-MPC vs. Plain GP-MPC. *(a)* Causal GP-MPC converges faster in one-step prediction RMSE. *(b)* Per-BPM RMSE confirms causal structure benefits all 10 dimensions, with the largest gains for upstream BPMs. Some BPMs are more sensitive than others, so in these cases the causal structure is even more beneficial.

Noise Robustness. We evaluated robustness by sweeping the observation noise magnitude σ (Fig. 4). For each noise level, the GP-MPC models are retrained from scratch with $N = 50$ online transitions; PPO uses a single policy trained at $\sigma = 0.025$. While KalmanQP performs optimally at near-zero noise, it exhibits a brittle failure mode: although the steady-state Kalman gain is noise-invariant, the absolute estimation error grows linearly with σ, causing the deadbeat QP ($H = 1$) to produce saturating, sign-changing corrector kicks. Causal GP-MPC maintains stability in high-noise

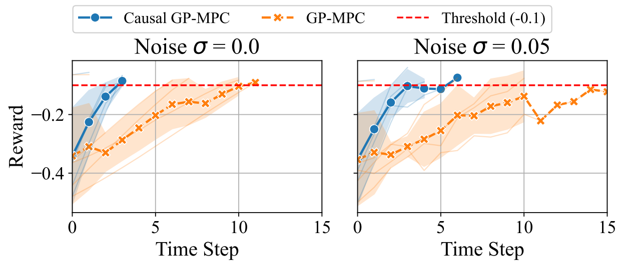

Figure 2: Per-step reward trajectories at $N = 50$ for noise levels $\sigma \in \{0, 0.05\}$. Causal GP-MPC (blue) converges faster and with lower variance than plain GP-MPC (orange). Red dashed line: success threshold $r = -0.1$. Shaded uncertainty bands are suppressed beyond the step where fewer than three active trajectories remain due to variable episode termination.

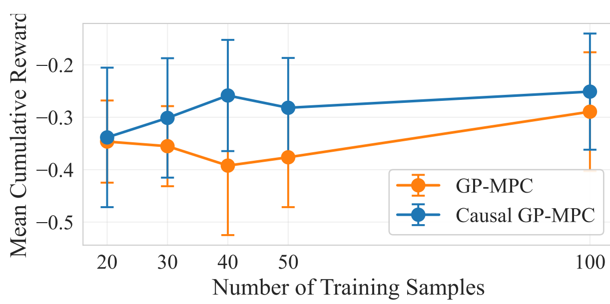

Figure 3: Sample efficiency at $\sigma = 0.05$, averaged over 10 seeds. Causal GP-MPC (blue) consistently outperforms plain GP-MPC (orange) across all training budgets N.

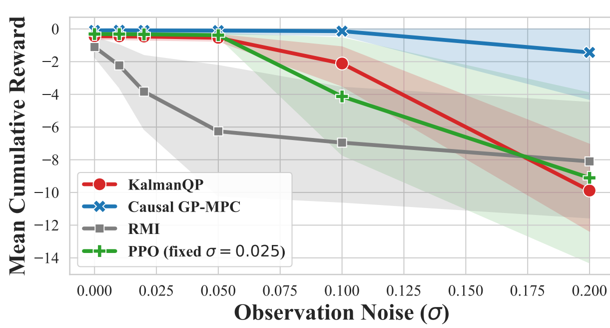

Figure 4: Cumulative reward robustness benchmark across observation noise σ. Causal GP-MPC maintains high reward throughout; KalmanQP collapses beyond $\sigma \geq 0.1$; PPO's reward also degrades substantially at high noise (cf. survival rate in Table 2).

regimes, as the multi-step GP-based planner ($H = 4$) smooths over measurement noise through its probabilistic posterior and the averaging effect of horizon-length optimisation.

Computational Efficiency. Table 1 compares per-step wall-clock latencies. GP-MPC's sub-Hz update rate is viable for accelerator control loops on the seconds timescale; the causal cone restriction reduces optimisation time by ~33% by shrinking each GP's input space.

Operational Safety. Table 2 evaluates survival across 50 random initial conditions at five noise levels (beam loss at $\|s_k\|_\infty \geq 1$). PPO ranks first but requires 10^6 samples. Causal GP-MPC ranks second, outperforming both model-based controllers at $\sigma \geq 0.1$ and unstructured GP-MPC by 9× at $\sigma = 0.1$ (54% vs. 6%), using only $N = 50$ transitions.

Table 1: Method Comparison: Training Cost, Latency, and Noise Robustness

Method	**Model req.**	**Train samples**	**Latency**	**Noise rob.**
RMI	Yes	0	< 1 ms	Low
KalmanQP	Yes	0	~ 5 ms	Med.*
PPO	No†	~10^6	~ 1 ms	Med.
GP-MPC	No	~100	~ 6 s	Med.
Causal GP-MPC	No	**~40**	~ 4 s‡	**High**

*Degrades at $\sigma \gtrsim 0.1$. †Requires simulator. ‡33% less than GP-MPC.

Table 2: Safety Evaluation: Survival (%) and Mean Episode Length Across 50 Episodes per Noise Level

Method	**Metric**	**Noise level σ**				
		0.01	**0.025**	**0.05**	**0.1**	**0.2**
RMI	Surv. (%)	92	76	70	32	10
	Len. (steps)	8.2	11.2	15.2	16.0	10.4
KalmanQP	Surv. (%)	92	74	54	20	2
	Len. (steps)	6.9	7.3	8.6	10.6	7.6
PPO	Surv. (%)	100	100	100	98	70
	Len. (steps)	2.4	2.5	3.5	24.2	40.4
GP-MPC	Surv. (%)	44	44	32	6	2
	Len. (steps)	8.7	7.4	8.2	12.0	7.1
Causal GP-MPC	**Surv. (%)**	**72**	**78**	**74**	**54**	**12**
	Len. (steps)	**15.9**	**16.1**	**15.3**	**22.3**	**17.2**

CONCLUSION

Causal GP-MPC embeds the beamline's causal layout into GP learning, reducing input dimensionality, improving RMSE, and cutting computation by ~33%. With only $O(50)$ transitions it achieves the highest cumulative reward across five noise levels and ranks second in survival, outperforming both model-based controllers at $\sigma \geq 0.1$ and unstructured GP-MPC by 9× at $\sigma = 0.1$ (54% vs. 6%). While KalmanQP remains superior at near-zero noise, Causal GP-MPC provides the best overall performance across the full noise spectrum. These results establish causal structure as a practical pathway toward data-efficient, empirically safer, and interpretable accelerator control.

ACKNOWLEDGEMENTS

Supported by the WISS 2025 project 'IDA Lab Salzburg' (20102/F2300464-KZP, 20204-WISS/225/348/3-2023).

REFERENCES

[1] V. Kain *et al.*, “Sample-Efficient Reinforcement Learning for CERN Accelerator Control”, *Phys. Rev. Accel. Beams*, vol. 23, p. 124801, 2020. doi:10.1103/PhysRevAccelBeams.23.124801

[2] E. Adli *et al.*, “Acceleration of electrons in the plasma wakefield of a proton bunch”, *Nature*, vol. 561, pp. 363–367, 2018. doi:10.1038/s41586-018-0485-4

[3] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Wiley, New York, NY, USA, 1994.

[4] CERN, “MAD-X (Methodical Accelerator Design)”, https://madx.web.cern.ch/.

[5] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems”, *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, 1960. doi:10.1115/1.3662552

[6] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*, Cambridge University Press, Cambridge, UK, 2017.

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

[8] O. Mironova, “Bridging Classical Control and Reinforcement Learning via Structured Priors for Robustness in Constrained Physical Systems: An Application to Beam Steering at CERN AWAKE”, MSc thesis, Dept. of Artificial Intelligence and Human Interfaces, University of Salzburg, Salzburg, Austria, 2026. urn:nbn:at:at-ubs:1-61544

[9] J. Schulman *et al.*, “Proximal Policy Optimization Algorithms”, 2017. doi:10.48550/arXiv.1707.06347

[10] A. Raffin *et al.*, “Stable-Baselines3: Reliable reinforcement learning implementations”, *J. Mach. Learn. Res.*, vol. 22, no. 268, pp. 1–8, 2021.

[11] S. Hirlaender, L. Lamminger, Z. Della Porta, V. Kain, “Ultra fast reinforcement learning demonstrated at CERN AWAKE”, in *Proc. IPAC'23*, Venice, Italy, May 2023, pp. 4510–4513. doi:10.18429/JACoW-IPAC2023-THPL038

[12] S. Kamthe and M. P. Deisenroth, “Data-Efficient Reinforcement Learning with Probabilistic Model Predictive Control”, in *Proc. 21st Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, PMLR 84, 2018, pp. 1701–1710.

[13] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, USA, 2006.

[14] M. P. Deisenroth and C. E. Rasmussen, “PILCO: A Model-Based and Data-Efficient Approach to Policy Search”, in *Proc. 28th Int. Conf. on Machine Learning (ICML'11)*, Bellevue, WA, USA, 2011, pp. 465–472.

[15] P. Virtanen *et al.*, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”, *Nat. Methods*, vol. 17, pp. 261–272, 2020. doi:10.1038/s41592-019-0686-2