Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Physics-Informed Machine Learning for Predicting the Robustness of Rydberg-Atom Controlled-Z Gate Pulses

Abhinavu Babu
Indian Institute of Technology Roorkee
Mayank Goswami
mayankfph@iitr.ac.in

Indian Institute of Technology Roorkee

Research Article

Keywords: Rydberg atoms, controlled-Z gate, quantum optimal control, robust control, surrogate model, gradient boosting, conformal prediction

Posted Date: November 11th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11679320/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Physics-Informed Machine Learning for Predicting the Robustness of Rydberg-Atom Controlled-Z Gate Pulses

Abhinavu Babu[1] and Mayank Goswami,[1,2]

[1]Department of Physics, IIT Roorkee
[2]MF School of AI and DS, IIT Roorkee
mayank.goswami@ph.iitr.ac.in

***Abstract*—Neutral-atom quantum computers entangle qubits with laser pulses that must stay accurate despite shot-to-shot laser and motional errors. Choosing among candidate pulses therefore requires a *robustness score*: the expected gate infidelity under a distribution of errors. Computing this score exactly takes many simulations per pulse. This paper asks a deliberately narrow question: *for a single-pulse Rydberg-blockade controlled-Z (CZ) gate, when does machine learning (ML) predict this score better than a cheap physics baseline, and when does it not?* Using a simulator validated against a published time-optimal pulse (infidelity 1.6×10^{-5} without decay and 1.17×10^{-3} with decay, against 1.13×10^{-3} reported), we label 3 000 training and 2 000 test pulses with an exact quadrature of the expected infidelity, and compare four predictors: a second-order perturbative baseline (M0), gradient-boosted trees on raw pulse parameters (M1), trees on raw plus physics features (M2), and the perturbative baseline plus a learned residual (M3). All ML predictors are given split-conformal 90% intervals. The results are mostly negative for ML at realistic parameters and positive only in a regime map: at the measured error strengths the perturbative baseline already predicts $\log_{10} R$ with an RMSE of about 10^{-4}, whereas raw ML is about 2500 times worse; ML helps once quasi-static errors reach several percent (RMSE of M0 grows to 0.117 at $\sigma = 0.15$, while M2/M3 stay near 0.013), or when noise is correlated in time. Physics features are essential, since raw ML only overtakes the baseline at the largest error strength we tested. Designs optimised at infinite blockade lose rank correlation (Spearman $0.99 \to 0.50$) as the blockade strength drops to $8\,\Omega$. A shift test shows that the conformal intervals lose their coverage on pulses from other generators, and fair search baselines show that ML screening offers no advantage over ranking by nominal infidelity at realistic noise. We state limitations plainly: simulation only, one gate and one five-parameter pulse family, no experimental data, and no pulse that beats the published one. The code is short and fully reproducible.**

***Index Terms*—Rydberg atoms, controlled-Z gate, quantum optimal control, robust control, surrogate model, gradient boosting, conformal prediction**

Corresponding author: mayank.goswami@ph.iitr.ac.in

I. Introduction

Neutral atoms held in optical tweezers are a leading platform for quantum computation and simulation [1], [2]. Two-qubit gates use the Rydberg blockade: a laser briefly lifts atoms into a high-lying Rydberg state, where the strong interaction prevents simultaneous excitation of two close atoms [3], [4]. Recent experiments report controlled-Z (CZ) fidelities near 99.5% [5]–[8]. At that level, the gate error is dominated not by the idealised pulse but by imperfections: Rydberg-state decay, laser frequency and intensity noise, and thermal motion [5], [8], [10].

A *pulse* (the laser amplitude, detuning and phase versus time) can be designed by numerical optimal control [13], [14], and for the Rydberg CZ gate by dedicated time-optimal and robust constructions [9]–[11]. Designing for robustness requires many evaluations of the expected infidelity under noise. If each evaluation is expensive, a fast and trustworthy *surrogate* is valuable. Machine learning has been applied to quantum control in many forms, including reinforcement learning [17]–[19], neural pulse generators [20]–[22] and supervised or surrogate models [23], [24]. However, claims of ML benefit are rarely tested against a strong physics baseline, and rarely mapped against the strength and structure of the noise.

This work. We study one narrow, fully specified problem and report what we find, including the negative results. Our contributions are:

1) A compact, validated simulator of the global-drive blockade CZ gate (Sec. IV) that reproduces a published time-optimal pulse.
2) A controlled comparison of four predictors of the robustness score, with calibrated uncertainty and an extrapolation test (Sec. V).
3) A *regime map* showing where physics alone suffices and where ML is needed, using a noise-strength sweep, five-seed repetition, and a learning curve (Sec. VI).
4) Two transfers: time-correlated detuning noise and finite blockade strength.

5) An honest account of what this does *not* show (Sec. VIII).

We do *not* propose a new ML algorithm. The novelty claim is limited to the systematic, noise-resolved comparison against a perturbative baseline for this gate family.

II. Background and Related Work

A. Rydberg blockade CZ gates

In the Rydberg blockade protocol, each qubit state $|1\rangle$ is coupled to a Rydberg state $|r\rangle$ with Rabi frequency Ω. A global drive acts on both atoms. If the interaction B is much larger than Ω, the doubly excited state is shifted out of resonance. The dynamics then split into independent blocks, with the $|11\rangle$ block driven at the enhanced frequency $\sqrt{2}\,\Omega$ [1], [2]. A pulse that returns all blocks to the computational subspace with the correct relative phases implements a CZ gate. Jandura and Pupillo [9] found time-optimal single-pulse solutions, and Evered *et al.* demonstrated a CZ gate with measured fidelity 99.5% using a phase-modulated pulse of the form $\phi(t) = A\cos(\omega t - \phi_0) + \delta_0 t$ [5]. We use that pulse family and parameters as validation. Optimising pulses against decay and quasi-static errors is studied in [10], [11], and Tsai *et al.* benchmarked such gates on ^{88}Sr with an error budget and a fidelity-response theory [8].

B. Robust control

Composite pulses cancel systematic errors by design, for example BB1 [15]. GRAPE-type gradient methods optimise pulses against a model [13], and an average over sampled errors turns a pulse into a robust one [14]. Time-correlated noise is handled with filter functions [16].

C. Machine learning for quantum control

Closest to the present work are supervised or surrogate approaches. Shi *et al.* train composite-pulse phases against sampled errors with a GRAPE-style supervised loop (no neural network) [23]. Surrey *et al.* train a recurrent network on a simulated singlet-triplet qubit and then optimise pulses through the learned model [24]. Yang *et al.* designed robust pulse sequences with neural networks [20]. Deep reinforcement learning has been used for Rydberg gate design [19] and for robust site-selective single-qubit pulses on tweezer atoms [21]; a multilayer perceptron has been proposed to emulate optimal-control outputs on low-latency hardware [22]. To our knowledge these works do not compare against a perturbative baseline resolved by error strength for the Rydberg CZ gate, which is the gap this report addresses. Our literature search was limited in time, so this statement is a pointer, not a proof of novelty.

III. Problem Statement

A. Setting

Let $\boldsymbol{\theta} = (A, \omega, \phi_0, \delta_0, T) \in \Theta \subset \mathbb{R}^5$ parametrise a pulse (Sec. IV). Let $\boldsymbol{\eta} = (\epsilon, \delta)$ be two error variables: a relative Rabi-frequency error ϵ and a laser detuning error δ (in units of Ω). Both are quasi-static: constant during one gate, random from shot to shot, with a Gaussian distribution $p_\sigma(\boldsymbol{\eta}) = \mathcal{N}(0, \mathrm{diag}(\sigma_\epsilon^2, \sigma_\delta^2))$. Let $I(\boldsymbol{\theta}, \boldsymbol{\eta}) = 1 - F$ be the gate infidelity (Sec. IV). The *robustness score* is

$$R(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\eta}\sim p_\sigma}\big[I(\boldsymbol{\theta}, \boldsymbol{\eta})\big]. \tag{1}$$

Because infidelities span orders of magnitude, we work with $y(\boldsymbol{\theta}) = \log_{10} R(\boldsymbol{\theta})$.

B. The learning task

Given a training set $\{(\boldsymbol{\theta}_i, y_i)\}_{i=1}^N$ of pulses labelled by simulation, learn $\hat{y}(\boldsymbol{\theta}) \approx y(\boldsymbol{\theta})$ together with an interval $[\hat{y} - q\,\hat{s},\ \hat{y} + q\,\hat{s}]$ that contains the true y with probability $\geq 1 - \alpha$ ($\alpha = 0.1$) on new pulses from the same distribution.

C. Why this problem, and what is the need

Searching for a robust pulse needs R for thousands of candidates. In our toy model, one R costs 49 simulations, about 0.1–0.15 ms, which is cheap. The motivation applies when the label is expensive: Monte Carlo over correlated noise (about 3–5 ms per pulse here), master-equation simulation with many Rydberg levels, or hardware-in-the-loop evaluation. A surrogate is only useful if (i) it is more accurate than what physics already gives for free, (ii) it knows when it is unreliable, and (iii) it generalises beyond its training pulses. The problem statement is therefore a *test* of these three requirements rather than an assumption that ML helps.

D. Research questions and objectives

- RQ1 At realistic measured error strengths, does ML add anything over perturbation theory?
- RQ2 As the error strength grows, or the noise becomes time-correlated, when does ML become necessary, and do physics features matter?
- RQ3 Do robustness rankings made at infinite blockade survive at finite blockade strength?

Objectives. (a) validate the simulator against a published pulse; (b) quantify predictor accuracy, ranking quality, and interval coverage; (c) test extrapolation to pulse durations not seen in training; (d) compare pulse-search strategies at equal exact-evaluation budgets; (e) report limitations.

E. Evaluation metrics

We report the root-mean-square error (RMSE) and mean absolute error of $\log_{10} R$; the Spearman rank correlation ρ (ranking is what a search needs); top-1% recall, the fraction of the true best 1% of test pulses that lie in the predicted best 5%; and the empirical coverage of the 90% intervals.

IV. Simulation Model and Data

A. Gate model

We consider two identical atoms with a global drive in the infinite-blockade limit. With $\Omega = 1$ as the unit

TABLE I
SIMULATION PARAMETERS (UNITS: $\Omega = 1$).

Quantity	Realistic	Stress
Decay γ/Ω	3.9×10^{-4}	3.9×10^{-4}
σ_ϵ	0.008	0.08
σ_δ/Ω	0.0163	0.08
Pulse box A	$[0, 2\pi]$	
ω, ϕ_0	$[0.3, 2.5]$, $[-\pi, \pi]$	
δ_0, T	$[-1.5, 1.5]$, $[6, 14]\,\Omega^{-1}$	
Time steps N_t	48	

of frequency, each non-trivial block is a two-level system $\{|g\rangle, |r\rangle\}$ with Hamiltonian

$$H_c = \begin{pmatrix} 0 & \frac{c(1+\epsilon)}{2} e^{-i\xi(t)} \\ \frac{c(1+\epsilon)}{2} e^{+i\xi(t)} & -\delta - \frac{i\gamma}{2} \end{pmatrix}, \tag{2}$$

with $c = 1$ for the $\{|01\rangle, |10\rangle\}$ block, $c = \sqrt{2}$ for the $\{|11\rangle, |W\rangle\}$ block ($|W\rangle = (|0r\rangle + |r0\rangle)/\sqrt{2}$), and $|00\rangle$ undriven. The phase is $\xi(t) = A\cos(\omega t - \phi_0) + \delta_0 t$ and γ is the Rydberg decay rate, entering as a non-Hermitian loss. Each pulse is integrated with $N_t = 48$ piecewise-constant steps using the closed-form 2×2 step propagator. Let $a = \langle g|U_1|g\rangle$ and $b = \langle g|U_{\sqrt{2}}|g\rangle$ be the surviving amplitudes. Define $M = \mathrm{diag}(1, a, a, b)$. The average gate fidelity to a CZ gate up to single-qubit Z phases, including leakage, is [12]

$$F = \frac{\mathrm{Tr}(M^\dagger M) + \max_\theta |1 + 2a\, e^{-i\theta} - b\, e^{-2i\theta}|^2}{20}. \tag{3}$$

The maximisation over θ is done by a grid plus Newton refinement.

B. Parameters and noise

Table I lists the parameters. The "realistic" preset takes $\Omega/2\pi = 4.6\,\mathrm{MHz}$ and $T_1 = 88\,\mu\mathrm{s}$ from the Harvard/QuEra experiment [5], giving $\gamma/\Omega = 3.9 \times 10^{-4}$. The detuning spread uses the measured $T_2^* = 3\,\mu\mathrm{s}$ of that experiment; because the source does not state how T_2^* maps to a Gaussian σ_δ, we take $\sigma_\delta/\Omega = 0.0163$ (assuming $T_2^* = \sqrt{2}/\sigma_\delta$; 0.0115 if $T_2^* = 1/\sigma_\delta$), so this number is uncertain by a factor ~ 1.4. The amplitude spread $\sigma_\epsilon = 0.8\%$ is the shot-to-shot Rabi-frequency noise used in the ^{88}Sr experiment of Tsai *et al.* [8], a different atomic species. A "stress" preset with $\sigma_\epsilon = \sigma_\delta = 0.08$ is deliberately large to expose non-perturbative behaviour and is *not* a platform value.

C. Labels

R in (1) is computed by a deterministic 7×7 Gauss–Hermite quadrature over (ϵ, δ): 49 simulations per pulse, no sampling noise. The label is $y = \log_{10} R$.

D. Validation against a published pulse

The published time-optimal pulse of Evered *et al.* ($A = 2\pi \cdot 0.1122$, $\omega = 1.0431\,\Omega$, $\phi_0 = -0.7318$, $\delta_0 = 0$, $\Omega T/2\pi = 1.215$) [5] gives, in our simulator, an infidelity of 1.6×10^{-5} with no decay (about 6×10^{-6} with 800 steps) and 1.17×10^{-3} with $T_1 = 88\,\mu\mathrm{s}$ decay, compared with 1.13×10^{-3} in the published error budget. Reversing the sign of ϕ_0 raises the infidelity to about 0.7, so the convention is fixed by this check. The code asserts the first value before running. This validates the physics of (2)–(3); it does not validate the noise parameters.

E. Pulse data

Random pulses are poor training data because almost all have large nominal infidelity. We therefore generate pulses with a batched hill-climb on the *nominal* infidelity (60% of coordinates perturbed per step, annealed step size, 200 iterations) from uniform starts, keeping a mid-way snapshot for diversity. Training (3 000), calibration (1 000), test (2 000) and search-pool (30 000) sets use different random seeds (101, 202, 303, 999). Training and test pulses therefore share one generator; this limits what "in-distribution" means (Sec. VIII).

V. Machine-Learning Methodology

A. Features

Raw features: $A, \omega, \cos\phi_0, \sin\phi_0, \delta_0, T$ (the phase is encoded by sine and cosine to remove the 2π discontinuity). **Physics features:** for small errors, $I(\boldsymbol{\theta}, \boldsymbol{\eta}) \approx I_0 + \kappa_\epsilon\epsilon^2 + \kappa_\delta\delta^2$ with curvatures $\kappa_\epsilon = \frac{I(h,0)+I(-h,0)-2I_0}{2h^2}$, and similarly κ_δ, estimated by central differences with $h = 0.03$ (five simulations: nominal and four displaced). Averaging over symmetric independent Gaussians kills linear and cross terms at this order, giving

$$R_q = I_0 + \kappa_\epsilon\sigma_\epsilon^2 + \kappa_\delta\sigma_\delta^2. \tag{4}$$

The physics features are $\log_{10} I_0$, $\log_{10}\kappa_\epsilon$, $\log_{10}\kappa_\delta$ and $\log_{10} R_q$. They cost 5 simulations instead of 49.

B. Four predictors

- **M0** (physics only): $\hat{y} = \log_{10} R_q$. No training.
- **M1** (raw ML): gradient-boosted trees on raw features.
- **M2** (hybrid features): trees on raw plus physics features.
- **M3** (physics + residual): trees predict $y - \log_{10} R_q$ from raw plus physics features, and $\hat{y} = \log_{10} R_q + \hat{r}$.

The trees are histogram-based gradient boosting [28], [30] (400 iterations, learning rate 0.05, at most 31 leaves, $\ell_2 = 10^{-3}$). Each model is a bootstrap ensemble of five [29]; the ensemble mean is the prediction and the ensemble standard deviation $\hat{s}$ feeds the interval.

C. Conformal intervals

Split-conformal prediction with normalised scores [25]–[27]: on the calibration set compute $s_j = |y_j - \hat{y}_j|/(\hat{s}_j + 10^{-3})$ and let q be the $\lceil (n+1)(1-\alpha)\rceil/n$ empirical quantile. The interval for a new pulse is $\hat{y} \pm q(\hat{s} + 10^{-3})$. This guarantees marginal coverage $\geq 1 - \alpha$ if calibration and test pulses are exchangeable. It does not guarantee coverage conditional on a pulse, or under distribution shift.

TABLE II
Realistic preset: test accuracy on 2 000 held-out pulses (seed 303).

Model	RMSE	Spearman	top-1%	cov.
M0 physics	1×10^{-4}	1.000	1.00	–
M1 raw ML	0.2524	0.905	0.85	0.907
M2 raw+phys.	0.0034	1.000	1.00	0.905
M3 phys.+resid.	$< 5\times10^{-5}$	1.000	1.00	0.891

TABLE III
Stress preset (8% errors): RMSE of $\log_{10} R$ in-distribution and extrapolating to $T \geq 9.5\,\Omega^{-1}$.

Model	In-dist.	Extrap.	Spearman
M0 physics	0.0325	0.0389	0.995
M1 raw ML	0.1211	0.2614	0.937
M2 raw+phys.	0.0065	0.0129	1.000
M3 phys.+resid.	0.0048	0.0328	1.000

D. Experiments

(E1) In-distribution accuracy at the realistic and stress presets. **(E2) Noise sweep:** $\sigma_\epsilon = \sigma_\delta = \sigma \in \{0.02, 0.05, 0.10, 0.15\}$. **(E3) Extrapolation:** train only on $T < 9.5\,\Omega^{-1}$, test on $T \geq 9.5\,\Omega^{-1}$. **(E4) Repeated seeds:** five independent train/test draws. **(E5) Learning curve:** training size 250 to 10^4. **(E6) Search:** from a pool of 30 000 pulses, compare random sampling, a $(1+\lambda)$ evolution strategy ($\lambda = 20$) run directly on exact R, and ML screening (rank the pool by M3 and verify the top B exactly) at equal exact-evaluation budgets B. **(E7) Correlated noise:** detuning follows an Ornstein–Uhlenbeck process [31] with correlation time τ_c; labels use 256 Monte Carlo trajectories with common random numbers; predictors still use the quasi-static features. **(E8) Finite blockade:** add the doubly excited state $|rr\rangle$ with shift B in the $|11\rangle$ block (three levels) and check how infinite-blockade rankings transfer.

VI. Results

All numbers are from the project's Colab runs (seeds 101/202/303/999) and were independently reproduced by a second run of the same code, agreeing to about four digits. Seed-repeat statistics in Sec. VI-D come from the repeat runs.

A. Realistic parameters (RQ1)

Table II and Fig. 1 show that M0 is nearly exact (RMSE $\approx 10^{-4}$) and M3 adds only a vanishing correction. Raw ML (M1) has an RMSE of 0.252 in $\log_{10} R$, i.e. a factor ~ 1.8 in R, and M2 reaches 0.0034. The reason is structural: at $\sigma \sim 1\%$ the infidelity surface is almost exactly quadratic in (ϵ, δ), and Eq. (4) captures precisely that. A black-box learner must rediscover the curvatures from 3 000 examples of a five-dimensional function that varies over orders of magnitude. **Answer to RQ1: at realistic quasi-static error strengths, ML adds nothing over perturbation theory.**

B. Large errors (RQ2)

At 8% errors (Table III) the quadratic baseline begins to fail (RMSE 0.033), raw ML is better in relative terms than at realistic noise (0.121), and the hybrid models M2 and M3 are best (0.0065 and 0.0048). For extrapolation to longer pulses, raw ML degrades the most (RMSE 0.261), M2 stays accurate (0.013), and M3 is only marginally better than the baseline (0.033 vs. 0.039): the learned residual does not extrapolate. In the realistic setting, our re-run of the extrapolation test gives RMSE 0.486 for M1, 0.012 for M2 and 10^{-4} for M0 and M3.

TABLE IV
Noise sweep: test RMSE of $\log_{10} R$ ($\sigma_\epsilon = \sigma_\delta = \sigma$). Bold: best.

σ	M0	M1	M2	M3
0.02	0.0004	0.2318	0.0032	**0.0002**
0.05	0.0098	0.1661	0.0040	**0.0017**
0.10	0.0538	0.0994	0.0091	**0.0073**
0.15	0.1168	0.0662	0.0138	**0.0127**

C. Noise sweep: the regime map

Table IV and Fig. 2 give RMSE against σ. The baseline error grows by more than two orders of magnitude from $\sigma = 0.02$ to 0.15; raw ML *improves* with σ (the target surface becomes smoother and easier to learn), and **overtakes M0 between $\sigma = 0.10$ and 0.15**. The hybrids beat both baselines from $\sigma = 0.05$ upward. The practical regime map is: perturbation theory below $\sigma \approx 0.05$, physics-informed ML above, and raw ML only when errors are very large.

D. Repeated seeds and data size

Table V repeats the train/test generation with five independent seed sets (training 3 000, test 2 000 pulses each; mean ± standard deviation across seeds). The conclusions of the single-seed tables are stable: M1's realistic error is 0.254 ± 0.010, M3 is again essentially exact, and under stress M3 is no better than M0 when extrapolating (0.033 vs. 0.039) while M2 remains the most reliable.

Learning curve. Table VI shows test RMSE versus training-set size (single seed). The baseline M0 has RMSE 1.1×10^{-4} (realistic) and 0.0316 (stress) and needs no training data; raw ML does not reach it even with 10^4 pulses.

E. Shift of the pulse distribution

The first version tested only on pulses from the training generator. We added four test sets from different generators (2 000 pulses each; models and conformal calibration unchanged): uniform random pulses in the parameter box, a short hill-climb (40 iterations instead of 200), Gaussian jitter around the published pulse, and long pulses ($T \geq 12\,\Omega^{-1}$). Tables VII and VIII give RMSE and coverage.



Fig. 1. Realistic preset. Left: predicted vs. exact $\log_{10} R$ on held-out pulses (M3). Middle: test RMSE by model. Right: best $\log_{10} R$ found vs. number of exact evaluations (random, evolution strategy, ML screening; dotted: pool optimum).

TABLE V
FIVE-SEED REPEAT: RMSE OF $\log_{10} R$, MEAN ± STD OVER SEEDS. "EXTRAP." TRAINS ON $T < 9.5\,\Omega^{-1}$ AND TESTS ON $T \geq 9.5\,\Omega^{-1}$.

	Realistic preset		Stress preset	
Model	In-distribution	Extrap.	In-distribution	Extrap.
M0 physics	$(1.1 \pm 0.1)\times10^{-4}$	$(1.3 \pm 0.2)\times10^{-4}$	0.0327 ± 0.0009	0.0390 ± 0.0012
M1 raw ML	0.2539 ± 0.0099	0.4470 ± 0.0198	0.1205 ± 0.0026	0.2459 ± 0.0144
M2 raw+phys.	0.0039 ± 0.0003	0.0141 ± 0.0026	0.0065 ± 0.0005	0.0160 ± 0.0013
M3 phys.+resid.	$(0.4 \pm 0.1)\times10^{-4}$	$(1.3 \pm 0.1)\times10^{-4}$	0.0045 ± 0.0004	0.0333 ± 0.0008

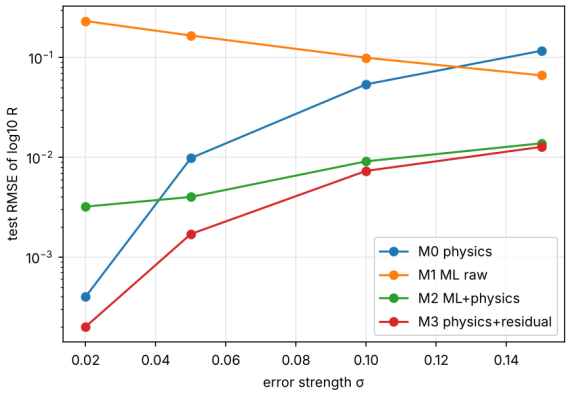


Fig. 2. Test RMSE of $\log_{10} R$ vs. error strength σ for the four predictors (log scale).

TABLE VI
LEARNING CURVE: TEST RMSE OF $\log_{10} R$ VS. NUMBER OF TRAINING PULSES N.

	Realistic			Stress		
N	M1	M2	M3	M1	M2	M3
250	0.3925	0.0465	0.7×10^{-4}	0.2038	0.0220	0.0083
500	0.3448	0.0214	0.7×10^{-4}	0.1795	0.0156	0.0073
1000	0.3105	0.0152	0.6×10^{-4}	0.1523	0.0120	0.0069
3000	0.2570	0.0055	0.4×10^{-4}	0.1239	0.0073	0.0051
10000	0.2284	0.0029	0.4×10^{-4}	0.1060	0.0049	0.0041

TABLE VII
REALISTIC PRESET: RMSE OF $\log_{10} R$ AND 90% COVERAGE OF M1/M2/M3 ON SHIFTED PULSE SETS.

Test set	M0	M1	M2	M3	c1	c2	c3
Same generator	1.0×10^{-4}	0.2521	0.0034	0.4×10^{-4}	0.91	0.92	0.89
Uniform random	0.0006	0.6118	0.1336	0.0005	0.39	0.22	0.40
Short hill-climb	0.7×10^{-4}	0.3155	0.0053	0.5×10^{-4}	0.82	0.90	0.82
Near published	0.2×10^{-4}	0.9458	0.0200	0.2×10^{-4}	0.30	0.85	0.94
Long ($T \geq 12$)	1.6×10^{-4}	0.2391	0.0031	0.5×10^{-4}	0.91	0.91	0.82

The physics baseline M0 is accurate on all sets (realistic: below 6×10^{-4}), and M3 inherits this at realistic noise. The pure ML models are not safe under shift. On uniform random pulses the error of M2 grows from 0.0034 to 0.13 (about 40×) and that of M1 from 0.25 to 0.61; near the published pulse M1 grows to 0.95 (about 4×) and M2 to 0.020 (about 6×); the milder shifts (short hill-climb, long pulses) change little. In addition, *conformal coverage collapses*: for the "near published" set the nominal 90% interval of M1 covers only 30% (realistic) and of M2 85%; for uniform pulses, coverage is 22–40% for all ML models. This is expected, since exchangeability is violated, but it shows that the calibrated intervals must not be trusted outside the training distribution. The practical remedy within this study is to compare the ML prediction with the physics baseline: a large $|\hat{y}_{M3} - \hat{y}_{M0}|$ flags pulses where the surrogate has moved far from physics.

F. Sensitivity to the detuning-noise convention

Because the conversion from T_2^* to σ_δ is uncertain, we repeated the realistic experiment (two seeds) with $\sigma_\delta = 0.0115$ and 0.025 in place of 0.0163. The picture does not change: M0 stays at about 10^{-4}, M1 at 0.268 ± 0.010 and

TABLE VIII
Stress preset: same as Table VII.

Test set	M0	M1	M2	M3	c1	c2	c3
Same generator	0.0325	0.1206	0.0065	0.0048	0.89	0.90	0.89
Uniform random	0.0133	0.4215	0.1172	0.0107	0.28	0.25	0.63
Short hill-climb	0.0209	0.1870	0.0076	0.0047	0.77	0.88	0.89
Near published	0.0059	0.4338	0.0186	0.0022	0.46	0.90	0.96
Long ($T \geq 12$)	0.0481	0.1035	0.0078	0.0059	0.90	0.87	0.87

0.251 ± 0.010, and M2 at 0.0043 and 0.0033. The qualitative conclusions therefore do not depend on this convention within a factor 1.5 either way.

G. Uncertainty calibration and speed

Over all runs the empirical coverage of the nominal 90% conformal intervals for M1–M3 lies between 0.88 and 0.92 (Tables II, III and IV and the repeat runs), as the marginal guarantee predicts. Coverage under distribution shift is examined in Sec. VI-E and fails. On speed: in the basic model an exact label costs about 0.1–0.15 ms per pulse and the ML path (five feature simulations plus the ensemble prediction) 0.12–0.3 ms, depending on the machine and the number of CPU threads, so there is *no speed-up* in this cheap model.

H. Pulse search against stronger baselines

The first version of this study compared ML screening with a weak, untuned evolution strategy and reported the best of 20 random draws as "random". Both choices flattered ML, and the second was a mistake in our code (a minimum over repeats instead of a mean). We therefore repeated the comparison with fairer baselines on the same pool of 30 000 pulses: random sampling (mean ± std over 30 repeats), a tuned $(1+\lambda)$ evolution strategy with step-size adaptation (10 seeds, started either from a random pool pulse, "cold", or from the pulse with the lowest *nominal* infidelity, "warm"), and three screening rules that rank the whole pool for free and verify the top B exactly: by nominal infidelity I_0, by the perturbative baseline M0, and by M3. Tables IX and X give the best $\log_{10} R$ found.

At realistic noise, *every* screening rule, including the trivial ranking by nominal infidelity, finds the pool optimum ($\log_{10} R = -2.753$) with only 10 exact evaluations. Random sampling does not reach it even at $B = 500$ (-2.688 ± 0.037), and the cold-started evolution strategy stays far above. ML therefore gives no advantage in search here: the pool was built by hill-climbing the nominal infidelity, and at realistic noise nominal infidelity is already an excellent proxy for R. Under the stress preset the physics-based rules (M0 and M3) reach the optimum (-1.751) at $B = 10$, while the nominal ranking needs $B = 20$; the gap is small. The published pulse ($\log_{10} R = -2.762$ realistic, -1.523 stress) is better than every pulse in the realistic pool, so *we do not find a better pulse*; under stress about 3% of the pool beats it. A warm-started evolution strategy reaches -2.760 at $B = 500$, essentially the published value.

TABLE IX
Realistic preset: best $\log_{10} R$ found vs. exact evaluations B. ES: evolution strategy.

B	Random	ES cold	ES warm	I_0 scr.	M0 scr.	M3 scr.
10	-2.150 ± 0.301	-1.287	-2.753	-2.753	-2.753	-2.753
20	-2.405 ± 0.205	-1.130	-2.753	-2.753	-2.753	-2.753
50	-2.453 ± 0.164	-1.148	-2.753	-2.753	-2.753	-2.753
100	-2.549 ± 0.111	-1.299	-2.753	-2.753	-2.753	-2.753
200	-2.645 ± 0.068	-1.390	-2.755	-2.753	-2.753	-2.753
500	-2.688 ± 0.037	-1.836	-2.760	-2.753	-2.753	-2.753

TABLE X
Stress preset (pool optimum -1.751).

B	Random	ES warm	I_0 scr.	M0 scr.	M3 scr.
10	-1.447 ± 0.132	-1.562	-1.567	-1.751	-1.751
20	-1.538 ± 0.097	-1.562	-1.746	-1.751	-1.751
50	-1.586 ± 0.091	-1.562	-1.746	-1.751	-1.751
100	-1.638 ± 0.062	-1.562	-1.746	-1.751	-1.751
200	-1.663 ± 0.056	-1.568	-1.746	-1.751	-1.751
500	-1.711 ± 0.029	-1.570	-1.751	-1.751	-1.751

I. Extension A: time-correlated detuning noise

For realistic strengths, the correlation time changes R by only about 0.4% relative to the quasi-static value (median ratio 0.996 at $\tau_c = 0.1/\Omega$). Nevertheless the quasi-static baseline M0 has an error of 0.015 in $\log_{10} R$ there, against 0.0036 for M3. At $5\times$ noise and $\tau_c = 0.1/\Omega$, R falls by about 7% relative to the quasi-static value, M0's error is 0.118 and M3's is 0.017, equal to within the Monte Carlo sampling floor (0.018) (Fig. 3). Measured in one process at a time, a Monte Carlo label costs 3–5 ms per pulse and the ML path (five feature simulations plus ensemble prediction) 0.12–0.3 ms, a speed-up of roughly 17–25×. The ranges cover two cloud machines and one or two CPU threads (first machine, one thread: 5.2 and 0.31 ms; second machine: 3.0 and 0.16 ms with one thread, 0.12 ms with two). An exact quasi-static label costs 0.09–0.16 ms, so the ML path is no faster than it.

J. Extension B: finite blockade (RQ3)

Table XI shows that rankings made at infinite blockade transfer well for $B \gtrsim 100\,\Omega$ (Spearman 0.99) and degrade for $B \leq 30\,\Omega$. At $B = 16\,\Omega$ the best infinite-blockade pulse is worse than the best at that B by 0.18 in $\log_{10} R$, and the published pulse worsens from -2.76 to -2.59. Designs made at infinite blockade therefore need re-checking at finite B (Fig. 4).

VII. Discussion

What the evidence supports. The perturbative baseline is a strong, nearly free predictor at the error strengths measured in current experiments. A learned model earns its place only when (i) errors are so large that the quadratic expansion fails, (ii) noise is correlated in time so that the quasi-static features no longer represent the labels, or (iii) labels are expensive enough (Monte Carlo, 3–5 ms) that a 0.1–0.3 ms ML evaluation beats

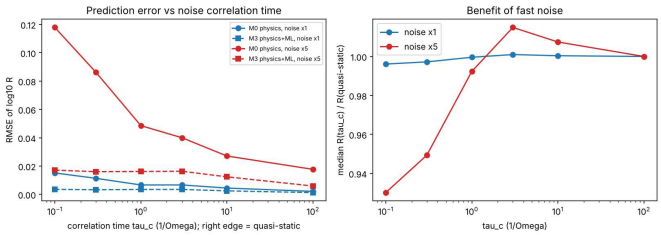


Fig. 3. Correlated detuning noise (1× and 5× the realistic strength). Left: prediction error of the quasi-static baseline (M0, solid) and physics+ML (M3, dashed) vs. correlation time τ_c; the right edge is the quasi-static limit. Right: median ratio of $R(\tau_c)$ to the quasi-static value.

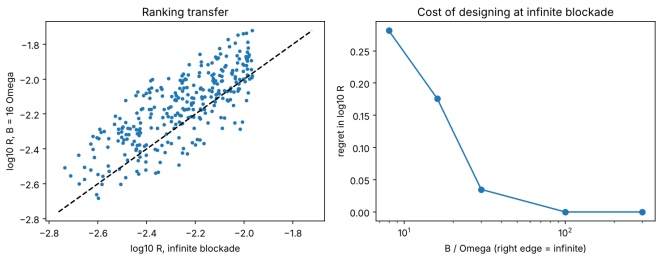


Fig. 4. Finite blockade. Left: $\log_{10} R$ of test pulses at $B = 16\,\Omega$ against infinite blockade (dashed: equality). Right: regret in $\log_{10} R$ from designing at infinite blockade, vs. B/Ω.

TABLE XI
FINITE BLOCKADE: TRANSFER OF INFINITE-BLOCKADE RESULTS.

B/Ω	Spearman ρ	Published $\log_{10} R$	Regret
∞	1.000	−2.762	0
100	0.990	−2.760	0
30	0.887	−2.713	0.035
16	0.772	−2.592	0.176
8	0.504	−2.266	0.281

exact computation. In every setting we tested, the best predictors were *hybrids* that build on the physics baseline (M2, M3). Raw ML sees the pulse parameters only through a highly non-linear map and needs far more data, an effect that the learning curve (Sec. VI-D) quantifies: even with 10^4 training pulses raw ML (M1) still has an RMSE of 0.228 at realistic noise and 0.106 under stress, while M3 is already accurate with 250 pulses.

What it does not support. We have not shown a faster route to a better gate: the surrogate does not beat the published pulse, the speed-up is absent in the cheap model, and the speed-up of roughly 17–25× exists only against Monte Carlo labels. We have not shown improvement for any experimental hardware.

Practical recommendation (within the model). Use Eq. (4) (five simulations) for robustness ranking at realistic quasi-static noise; switch to a hybrid physics+residual model with conformal intervals when $\sigma \gtrsim 0.05$ or when noise is correlated; and always re-check designs at the true blockade strength.

VIII. Limitations and Threats to Validity

1) **Simulation only.** All labels come from our own simulator. It reproduces one published pulse and its decay budget but we have no hardware data; noise parameters come from published setups and mix species (^{87}Rb and ^{88}Sr). The $T_2^* \to \sigma_\delta$ conversion depends on an unstated convention (±40%).
2) **Narrow scope.** One gate, global drive, a five-parameter phase-modulated pulse family with constant amplitude, mainly infinite blockade and quasi-static Gaussian noise. We omit laser phase and intensity noise spectra, atomic motion, finite rise times, and further Rydberg or intermediate levels, all of which matter in experiments [8].
3) **Distribution shift.** Training and standard test pulses share one hill-climb generator. Sec. VI-E tests four other generators: the physics baseline and M3 (realistic) remain accurate, but M1/M2 degrade and conformal coverage collapses to 22–85%. Other pulse families and gates are untested.
4) **Cheap simulator.** The central "fast surrogate" motivation is weak in the basic model; only the Monte Carlo extension has a meaningful cost gap.
5) **Search comparison.** Our first comparison used weak baselines and an erroneous best-of-20 random baseline; it was redone in Sec. VI-H. With fair baselines, ML screening has no advantage at realistic noise, because nominal infidelity already ranks the (pre-optimised) pool well. We did not test pools that are not hill-climbed, or other optimisers (e.g. CMA-ES or gradient-based GRAPE).
6) **Conformal guarantees** are marginal and assume exchangeability; we verified empirically that coverage fails under shift (Sec. VI-E).
7) **Statistics.** The headline tables use one seed set; Sec. VI-D adds five-seed repeats for accuracy, and the search comparison uses 30 random and 10 evolution-strategy repeats, but the extension experiments and the OOD tests have a single seed.
8) **Novelty.** Using ML surrogates for pulse robustness is not new [19], [23], [24]. Our literature search was limited in time and is not a systematic review.

IX. Conclusion and Future Work

We built and validated a simulator for the Rydberg blockade CZ gate, defined the robustness-prediction problem precisely, and compared a perturbative baseline with three ML variants. The result is a regime map: at realistic quasi-static errors physics suffices; physics-informed ML helps from roughly 5% errors or with correlated noise; raw ML is the weakest until very large errors. Finite-blockade effects invalidate infinite-blockade rankings below $B \sim 30\,\Omega$. Future work should use measured noise spectra with a filter-function surrogate [16], re-optimise

at finite blockade with more levels, tune stronger search baselines, add repeated seeds for every experiment, and use the conformal uncertainty for active learning so that exact simulation is spent only where the surrogate is unsure.

Reproducibility

The code is a small set of Python files (NumPy, SciPy, scikit-learn [30], [32], [33]): the simulator (`rydberg_core.py`), the main pipeline, the two extensions, and three robustness-check scripts (repeated seeds, learning curve, and shift/search/timing checks). Presets and test parts are selected by environment variables, and every script fixes its random seeds. The simulator asserts agreement with the published pulse before any experiment runs. Timings in this report were measured on a 2-core cloud machine and will differ on other hardware.

X. Video

https://youtu.be/CWAnzmU10Uc

References

[1] M. Saffman, T. G. Walker, and K. Mølmer, "Quantum information with Rydberg atoms," *Rev. Mod. Phys.*, vol. 82, pp. 2313–2363, 2010.

[2] M. Morgado and S. Whitlock, "Quantum simulation and computing with Rydberg-interacting qubits," *AVS Quantum Sci.*, vol. 3, 023501, 2021.

[3] D. Jaksch *et al.*, "Fast quantum gates for neutral atoms," *Phys. Rev. Lett.*, vol. 85, pp. 2208–2211, 2000.

[4] M. D. Lukin *et al.*, "Dipole blockade and quantum information processing in mesoscopic atomic ensembles," *Phys. Rev. Lett.*, vol. 87, 037901, 2001.

[5] S. J. Evered *et al.*, "High-fidelity parallel entangling gates on a neutral-atom quantum computer," *Nature*, vol. 622, pp. 268–272, 2023.

[6] P. Scholl *et al.*, "Erasure conversion in a high-fidelity Rydberg quantum simulator," *Nature*, vol. 622, pp. 273–278, 2023.

[7] S. Ma *et al.*, "High-fidelity gates and mid-circuit erasure conversion in an atomic qubit," *Nature*, vol. 622, pp. 279–284, 2023.

[8] R. B.-S. Tsai, X. Sun, A. L. Shaw, R. Finkelstein, and M. Endres, "Benchmarking and fidelity response theory of high-fidelity Rydberg entangling gates," arXiv:2407.20184, 2024.

[9] S. Jandura and G. Pupillo, "Time-optimal two- and three-qubit gates for Rydberg atoms," *Quantum*, vol. 6, p. 712, 2022.

[10] S. Jandura, J. D. Thompson, and G. Pupillo, "Optimizing Rydberg gates for logical-qubit performance," *PRX Quantum*, vol. 4, 020336, 2023.

[11] C. Fromonteil, D. Bluvstein, and H. Pichler, "Protocols for Rydberg entangling gates featuring robustness against quasi-static errors," *PRX Quantum*, vol. 4, 020335, 2023.

[12] L. H. Pedersen, N. M. Møller, and K. Mølmer, "Fidelity of quantum operations," *Phys. Lett. A*, vol. 367, pp. 47–51, 2007.

[13] N. Khaneja *et al.*, "Optimal control of coupled spin dynamics: Design of NMR pulse sequences by gradient ascent algorithms," *J. Magn. Reson.*, vol. 172, pp. 296–305, 2005.

[14] S. J. Glaser *et al.*, "Training Schrödinger's cat: quantum optimal control," *Eur. Phys. J. D*, vol. 69, 279, 2015.

[15] S. Wimperis, "Broadband, narrowband, and passband composite pulses for use in advanced NMR experiments," *J. Magn. Reson. A*, vol. 109, pp. 221–231, 1994.

[16] T. J. Green *et al.*, "Arbitrary quantum control of qubits in the presence of universal noise," *New J. Phys.*, vol. 15, 095004, 2013.

[17] M. Bukov *et al.*, "Reinforcement learning in different phases of quantum control," *Phys. Rev. X*, vol. 8, 031086, 2018.

[18] M. Y. Niu *et al.*, "Universal quantum control through deep reinforcement learning," *npj Quantum Inf.*, vol. 5, 33, 2019.

[19] Y. Cai, H. Zhang, K. Zhang, and J. Qian, "Intelligent optimal control of Rydberg gates with incremental-update deep reinforcement learning," arXiv:2605.04628, 2026.

[20] X.-C. Yang, M.-H. Yung, and X. Wang, "Neural-network-designed pulse sequences for robust control of singlet-triplet qubits," arXiv:1708.00238, 2017.

[21] S. Park and D. Kim, "Autonomously designed pulses for precise, site-selective control of atomic qubits," arXiv:2511.12524, 2025.

[22] D. Xu *et al.*, "Neural network accelerator for quantum control," arXiv:2208.02645, 2022.

[23] Z.-C. Shi *et al.*, "Supervised learning for robust quantum control in composite-pulse systems," *Phys. Rev. Applied*, vol. 21, 044012, 2024.

[24] P. Surrey, J. D. Teske, T. Hangleiter, H. Bluhm, and P. Cerfontaine, "Data-driven qubit characterization and optimal control using deep learning," arXiv:2601.18704, 2026.

[25] V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic Learning in a Random World*. New York: Springer, 2005.

[26] J. Lei *et al.*, "Distribution-free predictive inference for regression," *J. Amer. Statist. Assoc.*, vol. 113, pp. 1094–1111, 2018.

[27] A. N. Angelopoulos and S. Bates, "Conformal prediction: A gentle introduction," *Found. Trends Mach. Learn.*, vol. 16, no. 4, pp. 494–591, 2023.

[28] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, pp. 1189–1232, 2001.

[29] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, pp. 123–140, 1996.

[30] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[31] G. E. Uhlenbeck and L. S. Ornstein, "On the theory of the Brownian motion," *Phys. Rev.*, vol. 36, pp. 823–841, 1930.

[32] C. R. Harris *et al.*, "Array programming with NumPy," *Nature*, vol. 585, pp. 357–362, 2020.

[33] P. Virtanen *et al.*, "SciPy 1.0: fundamental algorithms for scientific computing in Python," *Nat. Methods*, vol. 17, pp. 261–272, 2020.