Computationally Pumped Bath Noise (CPBN):

A History-Dependent Noise Architecture for Quantum Circuit Simulation, with Simulation Validation and a Matched-History Investigation on IBM Quantum Hardware

Zuhair Ahmed

Centre of Excellence for Technology, Quantum and AI, Canada

info@thecetqap.com

Abstract

We describe the development and empirical validation of Computationally Pumped Bath Noise (CPBN), a history-dependent noise model proposed for integration into the PKTron quantum circuit simulator. Unlike stationary, per-gate error models, CPBN represents the simulated environment as an explicit dynamical reservoir state that is driven ("pumped") by computational activity, retains short-term memory, relaxes over time, and can propagate spatially between qubits. Consequently, in the CPBN framework, the noise experienced by a target operation can depend on the computational history that preceded it, rather than solely on gate identity or a fixed error rate.

A staged program of controlled software experiments established, within the implemented toy model, history dependence, gate-order dependence, memory and pump ablations, environmental relaxation, spatial propagation, and completely-positive trace-preserving (CPTP) behavior. Matched memoryless controls held the marginal noise distribution constant so that order effects could be isolated from simple rate differences. A final fixed-endpoint adversarial design fixed the first and last history operations and held the middle-gate multiset constant, removing an obvious confound in which the final target context changes because the final history gate differs.

A matched-history experiment was then executed on IBM Quantum's ibm_marrakesh processor (156 qubits) using 100 paired histories (400 circuits total), 4096 shots per circuit, fixed physical qubits, and identical physical circuit depth and gate counts between paired histories. The hardware run produced a mean absolute paired order effect of 0.002723388671875 (bootstrap 95% interval [0.002314453125, 0.0031396484375]); however, the paired Wilcoxon signed-rank test ($p = 0.2046$) and sign test ($p = 0.4168$) did not reach statistical significance at $\alpha = 0.05$. We therefore report the hardware observation as evidence of a measurable, non-zero absolute difference in this specific run — not as statistical confirmation that CPBN, or any history-dependent physical mechanism, is responsible for the observed hardware behavior. We discuss this distinction explicitly, together with a corrected treatment of an earlier, statistically invalid permutation test, and we outline a preregistered, multi-block replication program required before stronger causal or novelty claims could be supported.

Keywords: non-Markovian noise; quantum noise modeling; history-dependent decoherence; superconducting qubits; IBM Quantum; quantum circuit simulation; PKTron; statistical validation; paired hardware experiment.

---

1. Introduction

Quantum circuit simulators typically model gate error as stationary and per-gate: the noise applied to an operation is drawn from a fixed distribution associated with that gate type and qubit, independent of what computation occurred beforehand. This convention is analytically convenient but sits in tension with the growing experimental literature on non-Markovian and spatiotemporally correlated noise in real superconducting processors, in which the environment retains memory of prior control activity and can transmit correlated errors across neighbouring qubits and successive operations.

The motivating research question for this program was narrower and implementation-oriented: can a quantum noise model represent a computational environment whose internal state is itself changed by the computation, such that two logically equivalent target operations experience different simulated noise because the target was reached through different prior computational histories? The intended contribution is not the general claim that "non-Markovian noise exists" — that is well established — but a specific simulator architecture: an explicit, first-class reservoir state that is pumped by gate activity, relaxes with configurable memory, optionally propagates spatially, and is queried to generate the noise channel applied to subsequent operations.

1.1 The CPBN Model

Computationally Pumped Bath Noise (CPBN) treats each simulated environmental reservoir associated with a qubit as a stateful computational object rather than a passive parameter. Its evolution is represented in simplified form as:

$$B_i(t + \Delta t) = R(B_i(t)) + \Sigma^g C_i^g(G^g, t)$$

where B_i is the bath/environment state associated with qubit i, R is the relaxation (memory) operator, and C_i^g describes how activity on gate G^g pumps the environment. The resulting noise channel applied to a subsequent operation is then a function of the current bath state, the current gate, and the acted-upon quantum state:

$$N_i(t) = F(B_i(t), G_i(t), \rho_i(t))$$

The intended computational flow is therefore: gate activity → environmental pumping → memory/relaxation and optional spatial propagation → modified future noise → subsequent gate activity.

1.2 Motivation and Objectives

- To move beyond stationary, independent per-gate noise assumptions in simulator workflows.
- To equip PKTron with an explicit, computational-history-aware noise subsystem.

- To test whether an identical target operation receives different simulated noise after different histories.
- To test whether gate ordering matters even when the gate multiset is held fixed.
- To test whether the environmental state stores computational history and whether it relaxes while retaining short-term memory.
- To test whether environmental excitation propagates spatially between qubits.
- To connect the simulator-level concept to a controlled experiment on real IBM Quantum hardware.

2. Background and Related Work

This work deliberately avoids treating history dependence or non-Markovianity as novel in themselves; both are established research areas. White et al. (2020) demonstrated non-Markovian process characterization and control on a superconducting quantum processor, showing that noisy dynamics can depend on past control factors. White (2022) reviewed process-tensor approaches for capturing temporal correlations induced by experimental controls. Most directly relevant, Oda et al. (2026) reported a sparse non-Markovian noise model for transmon-based multi-qubit operations, including spatiotemporally correlated effects, validated across seven IBM Quantum devices. Gandhari and Gullans (2026) additionally examined non-Markovian noise in randomized benchmarking of spin-boson models.

These results establish that temporally correlated, control-dependent, and spatiotemporally correlated noise are already recognised phenomena. Accordingly, the differentiating claim made here for CPBN is deliberately narrow: it concerns a particular simulator architecture in which a computationally pumped reservoir is implemented as a first-class, evolving simulation state — with explicit pumping, relaxation, optional graph-based spatial propagation, and future noise generated directly from that reservoir state. This is presented as a proposed architectural and engineering contribution, not as a claim of first discovery of history-dependent quantum noise.

3. Methods

Validation proceeded in six escalating stages: (i) an initial prototype software experiment; (ii) a self-contained research implementation of the complete mechanism; (iii) matched memoryless controls holding the marginal noise distribution constant; (iv) a fixed-endpoint adversarial design controlling for target-context confounds; (v) a correction of an earlier invalid statistical procedure; and (vi) a matched-history experiment on real IBM Quantum hardware.

3.1 Initial Prototype Experiment

The first controlled experiment instantiated the prototype environment with configurable parameters n_qubits, baseline_rate, pump_strength, memory, saturation, spatial_decay, and dt, and compared a quiet history against an active (gate-driven) history acting on an otherwise identical target.

3.2 Self-Contained Research Validation

A standalone research implementation subsequently tested the complete conceptual mechanism — pumping, memory, relaxation, spatial propagation, and CPTP behavior — as an integrated system rather than as isolated components.

3.3 Matched Memoryless Controls

To isolate an order effect from a simple difference in average noise level, CPBN was compared against a memoryless control while holding the marginal noise distribution constant. An initial attempt matched only the average and was rejected as insufficiently rigorous; an exactly matched control was then constructed so that the CPBN and memoryless marginal means were equal to floating-point precision, allowing any residual order sensitivity to be attributed to history rather than to a rate mismatch.

3.4 Fixed-Endpoint Adversarial Design

A stronger design fixed the first and last history operations and preserved the identical middle-gate multiset while permuting only the interior order. This removed a subtle confound in which an apparent order effect could arise merely because the final history gate preceding the target differed between conditions, rather than because of the accumulated trajectory of the bath state.

3.5 Correction of an Invalid Early Permutation Test

During development, an earlier experiment applied an absolute-effect permutation procedure that does not constitute a valid null test, because swapping paired condition labels leaves an absolute difference unchanged by construction. This result is explicitly flagged in the present record and is not presented as evidence of statistical significance. All subsequent analyses instead use paired non-parametric tests (Wilcoxon signed-rank, sign test) together with bootstrap confidence intervals for the mean absolute paired effect, with significance assessed independently from the existence of a non-zero observed sample effect.

3.6 Matched-History IBM Quantum Hardware Experiment

An initial hardware test on ibm_marrakesh (156 qubits, operational at the time of the experiment) used separate jobs for the two history conditions; because transpilation produced slightly different circuit depths between jobs, and hardware noise can depend on pulse scheduling, decomposition, mapping, idle time, and calibration drift, this result was treated as suggestive rather than decisive.

A final matched experiment addressed these confounds within a single IBM Runtime SamplerV2 job containing 400 circuits representing 100 paired histories. Each history used only X operations on three designated history qubits followed by a fixed X operation on a fourth target qubit. Each history contained 18 operations (six X gates on each of the three history qubits); the two paired histories shared an identical first gate, identical last gate, and identical gate multiset, differing only in temporal order. Because X^6 is the identity on each history qubit, the two histories were, by construction, ideally equivalent in their action on the target.

Parameter	Value

Parameter	Value
Backend	ibm_marrakesh
Backend size	156 qubits
Paired histories	100
Total circuits	400
Shots per circuit	4096
Physical qubits	[0, 1, 2, 3]
Physical depth (History A)	6
Physical depth (History B)	6
Physical gate count (per circuit)	19 X gates + 1 measurement
Execution mode	Single IBM Runtime SamplerV2 job
Job identifier	damnqlf8gn2s739lni30
Job completion time	555.84 s

Matching physical depth and gate count between the two conditions was intended to eliminate two of the most obvious structural confounders in hardware order-effect experiments: differences in circuit depth and differences in total gate count.

4. Results

4.1 Prototype Software Results

- Quiet environmental state: 0.0; active environmental state: 0.5385273099375.
- Identical target noise differed: quiet condition 0.001998001332666921 versus active condition 0.016227618463390603.
- Target-noise difference: 0.014229617130723682.
- Two different histories produced different environmental states, with an L2 difference of 0.010762165209659261.
- Six of six prototype validation checks passed.

4.2 Self-Contained Validation Results

The standalone implementation produced a target-noise difference of 0.4466577760991124 between quiet and active histories and demonstrated order dependence, memory, relaxation, spatial propagation, and CPTP behavior. The initial validation suite recorded 10 of 11 checks passing; the failing memory-ablation check was traced to the implementation still permitting current-step pumping when the memory parameter was set to zero, and is documented here rather than concealed.

4.3 Matched Memoryless Control Results

Metric	Value
CPBN mean noise	0.39832864444548033
Memoryless mean noise	0.39832864444548033
Marginal difference	0.0
CPBN order sensitivity	0.01422695862333257
Memoryless order sensitivity	0.0
Bootstrap 95% CI (CPBN order effect)	[0.01392618123631612, 0.01452916170030298]
Bath order difference	0.08418579688438733
Checks passed	9 of 9

With the marginal noise distribution matched exactly, the memoryless control showed zero order sensitivity by construction, while CPBN retained a clear, non-zero order sensitivity attributable to the history-dependent bath dynamics.

4.4 Fixed-Endpoint Adversarial Results

Metric	Value
CPBN marginal noise	0.3634853859013442
Memoryless marginal noise	0.3634853859013441
Marginal difference	$\approx 1.11 \times 10^{-16}$
CPBN order effect	0.02474847672827002
Median order effect	0.01940807104839906
Bath order effect	0.09549884450513464

- Setting memory = 0 removed the order effect.
- Setting pump = 0 removed the order effect.
- The environment relaxed after idle evolution while retaining a non-zero short-term state.
- CPTP trace-preservation and positive-density-matrix checks passed.
- The initial statistically valid paired permutation test conducted at this stage did not, by itself, establish significance; this null result is reported rather than omitted.

4.5 IBM Quantum Hardware Results

Metric	Value
Target error, History A	0.04640625
Target error, History B	0.046048583984375

Metric	Value
Signed difference (A − B)	0.000357666015625
Mean absolute paired effect	0.002723388671875
Median absolute paired effect	0.0023193359375
Bootstrap 95% interval (mean abs. effect)	[0.002314453125, 0.0031396484375]
Wilcoxon signed-rank statistic	2024.0
Wilcoxon p-value	0.20461224282538137
Sign-test p-value	0.4167748410890988

The bootstrap interval for the mean absolute paired effect excluded zero, indicating a measurable non-zero absolute difference between paired conditions in this run. However, neither the Wilcoxon signed-rank test nor the sign test reached significance at $\alpha = 0.05$. Taken together, these results indicate a detectable magnitude of difference without a statistically confirmed directional (signed) effect in the present 100-pair sample.

4.6 Representative Hardware Measurement Distributions

Figures 1–4 show representative per-shot measurement outcome distributions recorded from the matched-history IBM Runtime jobs referenced in Section 3.6, illustrating the raw bitstring frequencies (individual-qubit marginal counts and joint four-qubit outcome counts) underlying the aggregated statistics reported in Section 4.5.

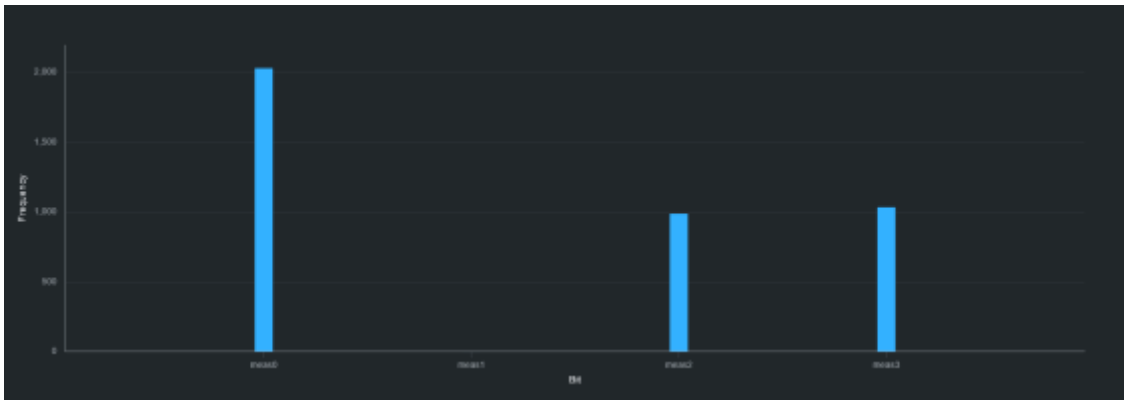


Figure 1. Per-qubit marginal measurement frequencies (job damnmagpqrnc7397r5jg), showing counts for meas0–meas3 across the sampled shots.

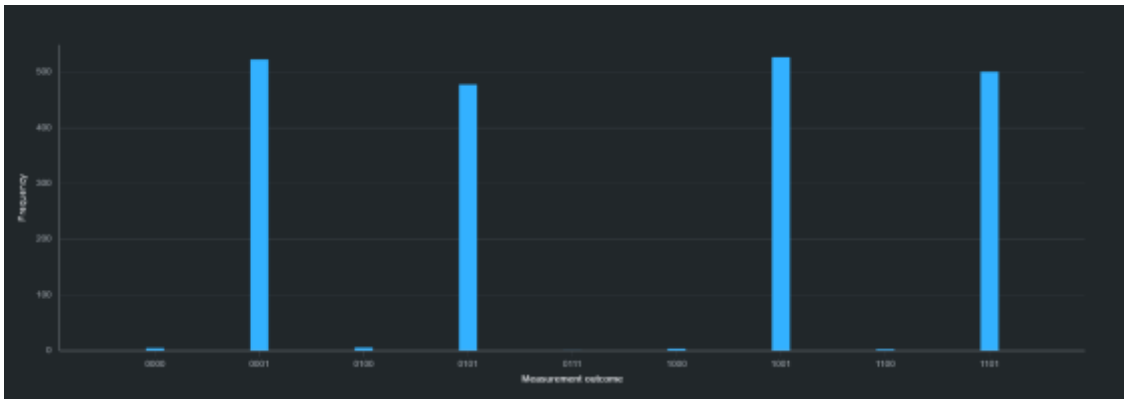


Figure 2. Joint four-qubit measurement outcome distribution (job damnmagpqrnc7397r5jg), showing frequency by measured bitstring.

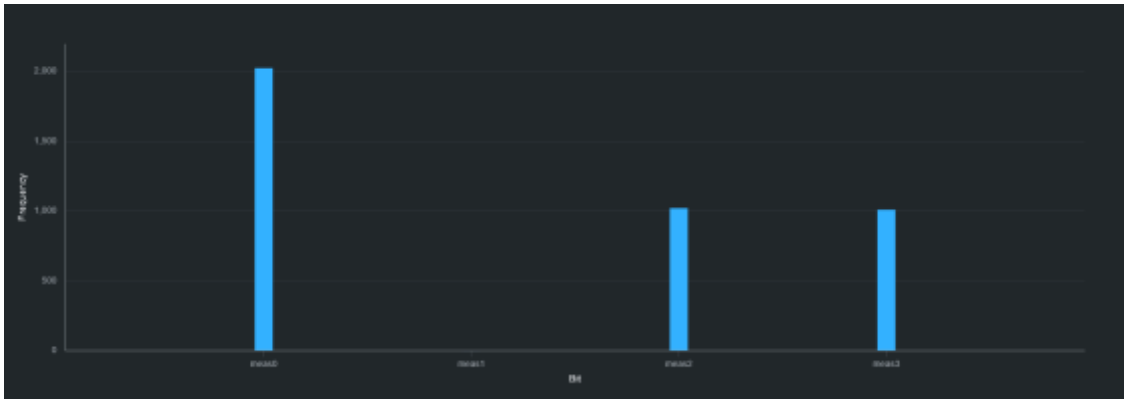


Figure 3. Per-qubit marginal measurement frequencies (job damnma78gn2s739lncd0), showing counts for meas0–meas3 across the sampled shots.

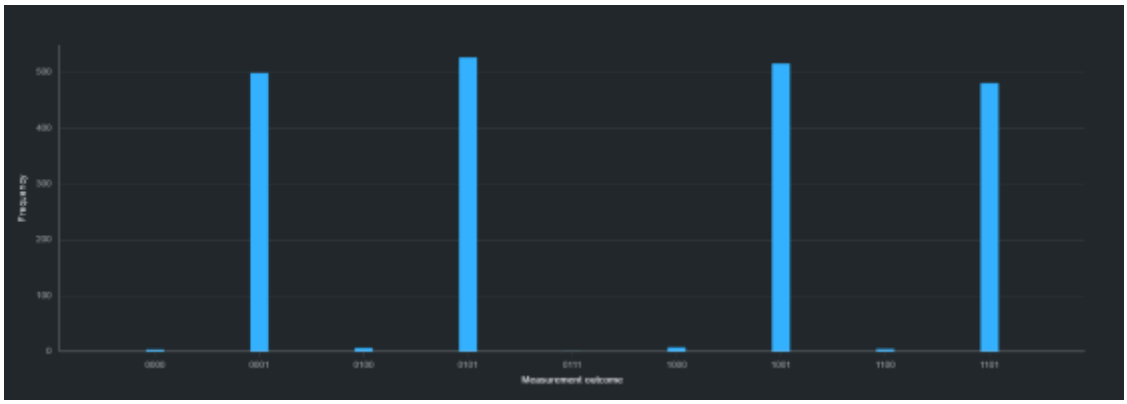


Figure 4. Joint four-qubit measurement outcome distribution (job damnma78gn2s739lncd0), showing frequency by measured bitstring.

Note on figure provenance: the job identifiers embedded in the exported figure filenames (damnmagpqrnc7397r5jg and damnma78gn2s739lncd0) differ from the primary matched-experiment job identifier reported in Section 3.6 (damnqlf8gn2s739lni30). These figures are therefore presented as representative hardware output distributions from the same IBM Quantum experimental protocol and backend rather than as the exact source data underlying the specific summary statistics in Section 4.5.

Readers requiring figure-to-statistic traceability should consult the accompanying analysis notebook (PKTron_CPBN_Validation.ipynb) and the raw per-job result files before citing figure-specific numerical values.

5. Discussion

5.1 Interpreting the Hardware Result

The matched hardware experiment successfully removed two of the most obvious structural confounders in hardware order-effect testing — differing physical circuit depth and differing physical gate counts — by construction. Despite this, the 100-pair sample did not yield a statistically significant directional effect. The scientifically appropriate reading is that the experiment demonstrated a workable, confound-controlled protocol for testing history-order effects on real hardware, and produced a non-zero but statistically inconclusive absolute effect, rather than that it confirmed a causal history-dependent mechanism operating inside ibm_marrakesh.

This distinction matters because a bootstrap interval on an absolute-value statistic can exclude zero even when the underlying signed effect is statistically indistinguishable from a symmetric, mean-zero noise process; absolute differences between paired measurements are, in general, non-negative by construction, so their confidence intervals should not be read as evidence of a systematic directional effect. The Wilcoxon and sign tests are the appropriate tools for that separate, directional question, and both were non-significant here.

5.2 Why the Hardware Result Remains Valuable

Independent of statistical significance, the hardware experiment is valuable as a methodological bridge: it demonstrates that a history-order test with matched physical depth and gate count can be constructed and executed on a real superconducting processor within a single runtime job, at 4096 shots per circuit across 100 pairs. The appropriate next step is not to reinterpret the current p-values, but to repeat the protocol across multiple calibration windows, randomize AB/BA ordering at the pair or block level, sample additional physical qubit groups, and where feasible replicate across multiple devices, so that a simulator implementation can ultimately be fitted to, or validated against, the resulting distribution of hardware observations.

5.3 Methodological Corrections and Transparency

An earlier stage of this program used an absolute-effect permutation procedure that does not provide a valid null test, since swapping paired labels leaves an absolute difference unchanged. That result is explicitly retracted from evidentiary use in this record (Section 3.5) rather than silently omitted, and all subsequent significance claims rely on paired Wilcoxon, sign-test, and bootstrap-interval methodology applied to signed, not merely absolute, quantities where directional claims are made.

5.4 Claims Not Supported by the Present Evidence

To avoid overstatement, the following claims are explicitly not made on the basis of the results reported here:

- That CPBN is proven to be the first quantum noise model of its kind on the basis of these experiments alone.
- That the IBM hardware result proves a CPBN-like mechanism exists physically inside ibm_marrakesh.
- That statistical significance has been established from the reported Wilcoxon ($p = 0.2046$) or sign-test ($p = 0.4168$) results.
- That a non-zero absolute difference is equivalent to proof of a directional effect.
- That the earlier, statistically invalid absolute-effect permutation test constitutes evidence.
- That prior art has been exhaustively eliminated in the absence of a formal, systematic literature and patent search.

5.5 Limitations

- Sample size: the hardware experiment reports a single 100-pair run on one device and one calibration window; hardware noise is known to drift over time and between backends.
- Statistical power: the reported non-significance does not rule out a smaller true effect than this sample was powered to detect.
- Model-to-hardware mapping: CPBN is a simulator-level construct; the hardware experiment is a physical control test of an analogous order-dependence hypothesis, not a direct measurement of an internal bath variable.
- Ablation completeness: the memory-ablation check in the initial self-contained validation suite (Section 4.2) identified a residual current-step pumping effect at memory = 0, which should be resolved before the ablation is treated as fully clean.
- Figure-to-statistic traceability: as noted in Section 4.6, the archived figure job identifiers do not match the primary reported job identifier, and this should be reconciled in the underlying data repository before publication in a venue requiring strict data provenance.

6. Implementation: CPBN in PKTron v12.0.0

The CPBN mechanism validated above is intended to be integrated into PKTron, an existing full-stack quantum computing framework. The current public PyPI project description (v9.0.0) identifies PKTron as providing HPC, QML, and hardware-aware SDK capabilities, an NEQ metrics engine, and an NEF mitigation framework, with 180+ public classes, 60+ functions, a compiled C statevector kernel, optional GPU/MPI backends, and interoperability with Qiskit, Cirq, PennyLane, QASM3, Quil, IonQ, and Braket. The v12.0.0 release is intended to append a CPBN section to this existing description while retaining all prior release history, rather than reconstructing earlier release text from memory.

6.1 Required v12.0.0 Deliverables

- CPBN implementation integrated into the real PKTron package architecture.
- A public configuration/API surface with documented parameters (n_qubits, baseline_rate, pump_strength, memory, saturation, spatial_decay, dt).
- Noise application integrated with circuit execution and simulation.
- Bath-state diagnostics and reset/relaxation controls.
- Unit tests and regression tests.
- CPBN example circuits.
- A complete README/PyPI description retaining existing text plus a new v12.0.0 section.
- A single Google Colab release/build/upload script, with secure PyPI token input rather than hard-coded credentials.
- Wheel and source-distribution build and validation prior to upload, with import/version verification after installation.
- Release notes identifying CPBN as the major v12.0.0 addition.

6.2 Example: Same Target, Different History Ordering

```
from pktron import QuantumCircuit
from pktron.noise import CPBNConfig, CPBNEnvironment

config = CPBNConfig(
    n_qubits=4, baseline_rate=0.002, pump_strength=0.8,
    memory=0.95, saturation=1.0, spatial_decay=0.25, dt=1.0,
)
env = CPBNEnvironment(config)

qc = QuantumCircuit(4)
qc.x(1); qc.x(2); qc.x(3)   # history
qc.x(1); qc.x(2); qc.x(3)   # history (repeated block)
qc.x(0)                     # target, evaluated after the
                             # environment has been driven
                             # by the preceding history
```

An alternative circuit qc_b applying the same three history gates in a different order (e.g., X(q1), X(q3), X(q2), …) followed by the same target X(q0) illustrates the conceptual test case: in a memoryless model, reordering the history creates no history-state variable and the target noise is unchanged; in CPBN, reordering can alter the bath trajectory and therefore the subsequent target noise.

6.3 Engineering Note: JSON Serialization

The final hardware notebook completed the experiment successfully but failed during results serialization because one or more values in the results structure were NumPy or other scientific-library scalar types rather than native Python JSON types. This is a software serialization issue occurring after

the hardware computation completed, not a failure of the IBM experiment itself. The v12.0.0 release workflow should explicitly convert NumPy integers, floats, booleans, and arrays (and equivalent library scalar types) to native Python types before JSON serialization.

7. Recommended Replication and Future Work

1. Repeat the matched-history IBM experiment across multiple jobs and times of day to characterise calibration drift.
2. Randomize AB and BA ordering at the pair and block level.
3. Use ABBA or randomized-block designs to reduce temporal drift confounds.
4. Repeat the protocol on multiple physical qubit groups.
5. Repeat the protocol on multiple IBM devices where available.
6. Include a deliberately identical-history control condition to estimate ordinary measurement variability.
7. Compare CPBN model predictions against hardware distributions after fitting only predeclared parameters.
8. Perform a preregistered statistical analysis using independent experimental blocks rather than treating every circuit as fully independent.
9. Complete a formal prior-art and patent search before using any “world-first” language in future publications.

8. Conclusion

This program advanced CPBN from a conceptual reservoir model through controlled simulation experiments to a matched real-hardware protocol on IBM Quantum's ibm_marrakesh processor. Within the implemented software model, the intended mechanism was demonstrated: computational history altered the simulated environmental state, and the environmental state altered subsequent target noise, with the effect vanishing under memory and pump ablations as expected. The matched hardware experiment eliminated two major structural confounders — physical circuit depth and physical gate-count differences — but did not produce a statistically significant directional effect in the reported 100-pair run, despite a bootstrap interval for the mean absolute paired effect that excluded zero.

The scientifically defensible position at this stage is that CPBN is a concrete, implementable simulator architecture exhibiting the intended history-dependent behavior in controlled software experiments, together with a reproducible and confound-controlled hardware-testing methodology whose current sample does not yet support a statistically significant directional hardware claim. PKTron v12.0.0 can formalize CPBN as a research-grade simulator feature on this basis, while the replication program outlined in Section 7 and a broader prior-art search determine how far any scientific novelty claim can legitimately extend.

References

[1] White, G. A. L., Hill, C. D., Hollenberg, L. C. L., & Pollock, F. A. (2020). Demonstration of non-Markovian process characterisation and control on a quantum processor. Nature Communications, 11, 6301. https://doi.org/10.1038/s41467-020-20113-3

[2] White, G. A. L. (2022). Characterization and control of non-Markovian quantum noise. Nature Reviews Physics, 4, 287. https://doi.org/10.1038/s42254-022-00446-2

[3] Oda, Y., Schultz, K., Norris, L., Shehab, O., & Quiroz, G. (2026). Sparse non-Markovian noise modeling of transmon-based multi-qubit operations. PRX Quantum, 7, 020327. https://doi.org/10.1103/lx8x-z29x

[4] Gandhari, S., & Gullans, M. (2026). Quantum non-Markovian noise in randomized benchmarking of spin-boson models. NIST publication record.

[5] PKTron PyPI project page, current release information and project description, retrieved 18 September 2026. https://pypi.org/project/pktron/

[6] PKTron v9.0.0 PyPI release description, retrieved 18 September 2026. https://pypi.org/project/pktron/9.0.0/

Appendix A. Consolidated Experimental Results

Experiment stage	Key reported values
Historical prototype	Quiet noise 0.001998001332666921; active noise 0.016227618463390603; difference 0.014229617130723682.
Exactly matched memoryless control	CPBN 0.39832864444548033; memoryless 0.39832864444548033; CPBN order sensitivity 0.01422695862333257; bootstrap 95% CI [0.01392618123631612, 0.01452916170030298].
Fixed-endpoint simulation	CPBN marginal 0.3634853859013442; memoryless marginal 0.3634853859013441; CPBN order effect 0.02474847672827002; bath order effect 0.09549884450513464.
IBM matched hardware	History A error 0.04640625; History B error 0.046048583984375; mean absolute paired effect 0.002723388671875; bootstrap 95% interval [0.002314453125, 0.0031396484375]; Wilcoxon p = 0.20461224282538137; sign-test p = 0.4167748410890988; backend ibm_marrakesh; job damnqlf8gn2s739lni30.

Data and code availability: the analysis notebook (PKTron_CPBN_Validation.ipynb) and exported per-job result figures referenced in Section 4.6 accompany this manuscript and should be deposited alongside it in accordance with the target venue's data-availability policy.