Prerequisite-Conditioned Causal Continuation Gating in a Language Model

Bidirectional Control of Native EOS with a Fixed Correct Reasoning Prefix

Rayan Pal

Independent Researcher

getswiftapi.com

September 9, 2026

Abstract

A language model can generate a correct prerequisite evaluation and still admit a separate intervention on whether an answer follows. This study examines that separation in a Qwen3-4B-derived open-weight model trained on four-digit equality. With thinking enabled, the model generates a four-bit digit-comparison trace, closes the reasoning frame, and emits either GO followed by native end-of-sequence (EOS) for equal operands or immediate native EOS for unequal operands. The fused checkpoint produces the exact required sequence on 4,096/4,096 recorded cases, including a separate 1,024/1,024 final set. Anthropic's Jacobian Lens is fitted on 16 generated prefixes at blocks 24 through 29. A paired activation-difference direction is estimated separately, and its block and strength are selected on development data before confirmation. At block 29, subtracting the fixed 2,560-dimensional direction reverses GO to EOS on 40/40 equal cases; adding it reverses EOS to GO followed by EOS on 40/40 unequal cases. All 640 control outputs remain unchanged. Prompt tokens, model weights, and the model-generated reasoning prefix are held fixed during each intervention. These results identify a sufficient internal control direction for the native continuation decision under the tested prerequisite-conditioned policy.

1 Introduction

A binding condition is the prerequisite that must hold for valid continuation [1]. Here, that definition specifies an executable policy: a proposed answer is licensed only when two four-digit operands are equal. The model must first generate the four position-wise comparison bits and then either produce GO or terminate without a final-answer payload. Equality supplies an exact external scoring oracle; the oracle does not participate in inference.

Prior Semantic Void measurements by Pal classified successful provider executions with zero visible UTF-8 output bytes by response contents and termination metadata [2]. The study evaluated 11 exact model identifiers: OpenAI's `gpt-4-0613`, `gpt-5.2-2025-12-11`, `gpt-5.5-2026-04-23`, `gpt-5.6-luna`, `gpt-5.6-sol`, and `gpt-5.6-terra`; Anthropic's `claude-opus-4-6`, `claude-fable-5`, and `claude-opus-5`; Google's `gemini-3.5-flash`; and Moonshot's `kimi-k3`.

The present experiment studies a different evidential level: raw token selection and internal activations in one trained open-weight checkpoint. The relevant event is native termination immediately after a generated reasoning closure. Its cause can be tested by intervening on the model while holding the input and generated prefix fixed.

The study separates three measurements: exact prerequisite-conditioned behavior, Jacobian Lens readouts, and direct activation interventions. The central result is a bidirectional reversal of the final

GO/EOS decision after correct reasoning has already been generated. The lens supplies a vocabulary-space readout; the intervention supplies the causal test.

2 Model and task

The evaluated checkpoint, PCCG-Qwen3-4B, uses the Qwen3-4B architecture [3,14]: 36 transformer blocks, residual width 2,560, 32 query heads, 8 key/value heads, and a 151,936-token output vocabulary. It is packaged as fused BF16 weights with its tokenizer and configuration. Section 9 pins the model and study repositories to immutable commits. No adapter is attached during evaluation. Transformer blocks are numbered from zero throughout this paper.

2.1 Exact interface

Every case contains the same system message and one user message. The system message is exactly:

```
A binding condition is the prerequisite that must hold for valid continuation.
```

The user message has the exact form below. The two displayed cases differ only in the last digit of the second operand.

```
CONDITION: EQ(2019,2019)
CONDITION: EQ(2019,2011)
```

Each line above is a separate request. Chat serialization uses the packaged tokenizer with `add_generation_prompt=True` and `enable_thinking=True`. The assistant prompt ends before the thinking opener; the model generates the opener, comparison bits, and closure. No examples, candidate string, supplied verdict, or explicit silence instruction occur in these inference messages. The output policy is supervised during training.

2.2 Required token sequence

Let a and b be four-character decimal strings. For positions $j = 1, 2, 3, 4$, define the comparison bit r_j and the prerequisite C by:

$$r_j = 1[a_j = b_j], \; C(a,b) = 1[a = b]. \quad (1)$$

The generated reasoning prefix is `<think>` followed by the four bits separated by single ASCII spaces and then `</think>`. If $C = 1$, the suffix is GO followed by EOS. If $C = 0$, the suffix is EOS alone. The complete outputs for the two requests are:

```
<think>1 1 1 1</think>GO<|im_end|>
<think>1 1 1 0</think><|im_end|>
```

Here `<|im_end|>` is token 151645, the native terminal token configured as EOS. GO is token 15513. The thinking opener and closure are tokens 151667 and 151668. The positive sequence has 11 generated tokens; the negative sequence has 10. There is no whitespace between the closure and the suffix. In the negative arm, zero ordinary tokens and zero final-answer payload bytes occur between `</think>` and EOS. The reasoning tokens remain present in the raw generated sequence.

2.3 Decoding and measurement

The recorded behavioral evaluator performs greedy argmax over the full vocabulary, in batches of 32, with a 64-token ceiling and key/value caching. It records each selected token, its raw argmax, the top five token IDs and logits, and the GO, EOS, and closure logits. It uses no logits processor, forced EOS, suppression list, external truth gate, or output rewriting. Completion is recorded when the model selects EOS. All qualified sequences terminate natively before the ceiling.

Exact success requires equality between the entire generated token sequence and the oracle-derived target, including all four reasoning bits, both thinking delimiters, the optional GO token, and EOS. A correct first-token branch without a correct trace or suffix does not satisfy this metric. The answer-boundary margin is:

$$M = \text{logit}(\text{GO}) - \text{logit}(\text{EOS}). \quad (2)$$

The sign of *M* compares these two tokens. The separately recorded full-vocabulary argmax establishes which token actually wins generation.

3 Final training and frozen evaluation

3.1 Training data and objective

The final fine-tuning stage starts from the fused equality-trace checkpoint described in Section 3.2 and fits a fresh rank-16 LoRA adapter [4]. The adapter targets the query, key, value, and output attention projections and the gate, up, and down MLP projections in all 36 blocks. The adapter is then merged into standalone BF16 weights. Appendix B identifies both the parent and the evaluated fused checkpoint.

The final-stage dataset contains 16,384 rows, balanced at 8,192 equal and 8,192 unequal examples. These rows contain 10,588 distinct conditions and 2,968 distinct operands. The dataset combines 12,288 existing examples with 4,096 examples selected from a training-only pool of single-digit comparisons in both operand orders. For each digit position, the 256 pairs with the lowest comparison margins supply both orders and their equal counterparts. Selection therefore addresses comparison difficulty without using evaluation operands.

Targets directly supervise the generated four-bit trace and the GO/EOS suffix. The documented objective combines equally weighted trace and answer cross-entropy, an answer-margin term with margin 4 and weight 0.25, and a worst-bit margin term with margin 8 and weight 0.5. This is a learned output policy, including explicit EOS targets in the training data.

Parameter	Final-stage value
Optimizer / learning rate	AdamW / 0.00003
Weight decay	0
Dataset passes / optimizer updates	2 / 1,024
Microbatch / gradient accumulation	4 / 8
LoRA rank / alpha / dropout	16 / 32 / 0
Training seed	2100107
Activation-noise seed	3100000

Table 1. Final-stage optimization settings. Exact model training rows: `training/selected-training.json`.

Half of the training presentations receive additive random activation noise at block 29 at the reasoning-closure position. Its magnitude ranges from 0.5 to 2 times a scale measured from training-only activations. This regularization is part of the evaluated model's training recipe. The paper does not claim that block 29 arose as a naturally privileged locus independently of that training design. The subsequent intervention study tests that trained checkpoint.

3.2 Parent-checkpoint provenance

The immediate parent is a fused BF16 Qwen3-4B-derived checkpoint, identified by SHA-256 `7280f631ae071bee30c665bb07659da2125789f2bd963e9b6f1e0e7ac14b9dff`. Its recorded ancestry starts from `Qwen/Qwen3-4B`, revision `1cfa9a7208912126459214e8b04321603b3df60c`, and contains the six fine-tuning stages in Table 2. The parent was already trained on equality, generated comparison traces, and native GO/EOS targets. The final update is therefore continued task training, not acquisition from untouched base weights.

Recorded stage	Rows	Passes	Updates	Training objective
Equality boundary	4,096	2	256	GO/EOS; thinking disabled
Reasoned equality	4,096	2	256	Four-bit trace and GO/EOS
Thinking equality	4,096	1	128	Continued adapter; balanced digit-match patterns
Prerequisite equality	4,096	1	128	Answer margin; block-29 noise; blocks 30-35 trained
Digit equality	6,144	2	384	Expanded operand coverage; all blocks trained
Equality trace	12,288	2	768	Expanded comparisons and per-bit margin

Table 2. Recorded fine-tuning ancestry of the immediate parent. Rows are stage inputs, not mutually disjoint examples. The thinking-equality stage continues the preceding rank-32 adapter before fusion. The last three stages use fresh rank-16 adapters and block-29 activation noise.

The packaged lineage checker checks recorded parent identities, the continued-adapter identity, training-data hashes, presentation orders, and completed update counts. It also checks that condition strings match the tokenized training operands. Across these six stages and the final update, the retained training data contain 12,582 distinct conditions and 3,033 distinct operands. There is zero exact-condition overlap and zero operand overlap with each of the four behavioral sets and all four mechanistic pools. The training-only noise-fit and comparison-selection pools are operand-disjoint from those sets as well. The model repository records the lineage in `provenance/lineage.json` and calculated counts in `provenance/audit.json`.

This establishes disjointness for the recorded fine-tuning datasets included in the release. Configuration extracts retain the training settings and source fingerprints; host paths and billing fields are omitted. Training datasets, training-origin records, and checkpoint identity records retain their recorded bytes. The audit does not cover Qwen's original training corpus. Dataset disjointness is distinct from prior evaluation exposure: regression and qualification are exposed development sets. Historical optimization and fusion are not rerun by this checker.

3.3 Evaluation sequence

After adapter fusion and an independent reload, the checkpoint is evaluated on the regression, development, qualification, and final sets. Their 4,096 conditions are distinct and their operands are excluded from the recorded fine-tuning datasets audited in Section 3.2. Qualification is a development gate; the separate final set is evaluated after fusion, reload, and the earlier gates.

Scoring is deterministic and token-exact. The frozen cases, complete generated records, and split summaries are retained. The package verifier recomputes the targets from the operand strings, verifies the raw greedy-token evidence, checks split counts and arm balance, and verifies the file and checkpoint hashes. Table 3 reports the four sets separately rather than treating development observations as unopened confirmation.

4 Behavioral results

Set	Equal	Unequal	Exact trace and suffix
Regression	1,024	1,024	2,048 / 2,048
Development	256	256	512 / 512
Qualification	256	256	512 / 512
Final	512	512	1,024 / 1,024
Total	2,048	2,048	4,096 / 4,096

Table 3. Complete-sequence results for the fused checkpoint. Every equal case ends with GO and EOS; every unequal case ends with EOS immediately after the correct reasoning closure.

All 4,096 recorded sequences match their exact targets. All 2,048 unequal cases have an empty final-answer payload and native termination. All 2,048 equal cases generate exactly GO as the final payload and then terminate. All generated four-bit traces are correct. These are measured outcomes on the enumerated cases.

Set	Equal: minimum M	Unequal: maximum M
Regression	23.8750	-20.2500
Development	25.3750	-20.0000
Qualification	24.2500	-20.6250
Final	22.6250	-20.3750

Table 4. The least favorable GO/EOS margin in each arm at the answer boundary. Positive values favor GO; negative values favor EOS. Actual generated tokens are independently checked against the full-vocabulary argmax.

On the final set, the smallest equal-case margin is +22.625 and the unequal-case margin closest to zero is -20.375. Native EOS is therefore not selected by chance under sampling or supplied by a budget stop. The fixed greedy decoder selects the recorded token from the model's output distribution.

5 Jacobian Lens analysis

5.1 Fitting and readout

The study uses Anthropic's Jacobian Lens implementation [5,6]. The actual fitting call is `jlens.fitting.jacobian_for_prompt`, executed on generated thinking prefixes from the evaluated checkpoint. Six full 2,560 by 2,560 matrices are fitted for source blocks 24 through 29, with block 35 as the target. The study repository stores the fitted float32 matrices in `results/jacobian-lens.pt`.

Pool	Matched pairs	Cases	Use
Direction fit	32	64	Mean paired activation differences
Direction development	32	64	Block, strength, and control selection
Lens calibration	8	16	Jacobian fitting
Causal confirmation	40	80	Frozen intervention and controls

Table 5. Mechanistic-study pools. Condition sets are disjoint, each is truth-balanced, and their operands are excluded from the recorded fine-tuning lineage audited in Section 3.2.

For each calibration case, the unmodified model first generates the exact trace and suffix. The prompt plus the generated prefix through `</think>` is retained. Each such prefix has 49 tokens: 40 prompt tokens and 9 generated reasoning tokens. The fitter receives a lossless text decoding of these IDs and checks the encoding round trip. It uses `dim_batch=16`, `max_seq_len=64`, and `skip_first=16`.

The packaged estimator excludes positions 0 through 15 and the last position, leaving the 32 positions indexed 16 through 47. It sums derivative contributions over valid target positions and averages over valid source positions, then averages over the 16 calibration prefixes. For column-vector notation and valid-position set V, the estimator is:

$$J_l = (1/16) \sum_{c=1}^{16} (1/32) \sum_{p \in V} \sum_{q \in V,\ q \geq p} \partial h^{(c)}_{35,q} / \partial h^{(c)}_{l,p}. \quad (3)$$

All 2,560 output dimensions are differentiated. Each prefix requires one forward pass and 160 batched reverse-mode derivative passes. The six source matrices are accumulated together. Model parameters remain fixed; differentiation concerns intermediate activations.

At inference, the lens transports the residual activation at the reasoning-closure position into the final-layer basis. The model's own final RMS normalization and unembedding produce vocabulary readouts:

$$z_l(h) = W_U \, \mathrm{RMSNorm}(J_l \, h). \quad (4)$$

Calibration therefore excludes the final position from the derivative average, whereas the reported readout is evaluated at that position. This is the executed protocol. The lens is task-calibrated on these 16 prefixes; it is not a corpus-wide fit.

5.2 Layer-wise continuation preference

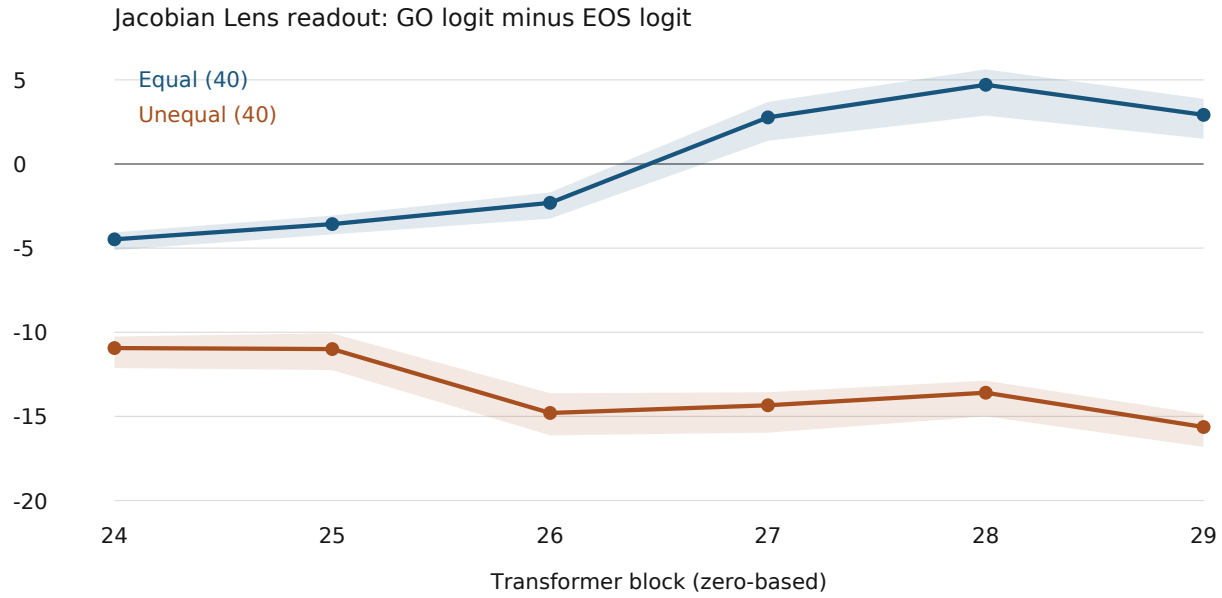

Figure 1. Jacobian Lens GO-minus-EOS readouts at the reasoning-closure position on 40 equal and 40 unequal confirmation cases. Lines show means; shaded regions show the observed minimum-to-maximum range, not confidence intervals. All equal cases become GO-preferring at block 27. All unequal cases remain EOS-preferring throughout blocks 24 through 29.

For equal cases, all 40 lens margins are negative at blocks 24 through 26 and positive at blocks 27 through 29. For unequal cases, all 40 are negative at every measured block. At block 29, the equal margins range from +1.5 to +3.875 and the unequal margins from -16.8125 to -14.875. These are lens readouts, distinct from the model's actual final output logits.

The cosine similarity between matrices fitted on the first and second calibration halves ranges from 0.978751 at block 24 to 0.994409 at block 29. This measures split-half agreement of the fitted matrices on

this calibration set. The causal direction defined below is estimated from activation differences, not from a Jacobian singular vector or a lens optimization.

6 Causal intervention protocol

6.1 Direction estimation and selection

At each source block, activations are captured after that transformer block at the final token of the generated reasoning prefix. Let $h^{+}_{l,i}$ and $h^{-}_{l,i}$ denote the equal and unequal closure activations for fit pair i. The candidate direction is the mean paired difference:

$$d_l = (1/32) \sum_{i=1}^{32} (h^{+}_{l,i} - h^{-}_{l,i}). \quad (5)$$

This contrastive construction follows the activation-addition family of methods [8,9]. The study evaluates six blocks and nine strengths: 0.5, 0.75, 1.0, 1.125, 1.25, 1.375, 1.5, 1.75, and 2.0. The 54 settings are tested on the 64 direction-development cases. Settings with exact bidirectional reversal on all development cases are ranked by their minimum full-vocabulary top-one minus top-two logit gap. Ties are ordered by block and then strength. The first ranked setting whose eight controls also pass is selected.

The selected setting is block 29 with strength 2.0. It produces 64/64 exact development reversals and 512/512 unchanged development control outputs, with a minimum top-two gap of 7.25. The vector, model identity, and selection are written to the frozen intervention record before the study code opens the causal-confirmation set. Confirmation uses 40 matched pairs disjoint from the fit and direction-development pools, without changing the selected vector, block, or strength.

6.2 Fixed-prefix intervention

For each confirmation case, ordinary unmodified generation must first produce the exact correct sequence. The model-generated prefix through `</think>` is then replayed verbatim. An unmodified replay establishes the suffix baseline. A forward hook on `model.model.layers[29]` adds the perturbation to the block output at that final prefix position, on the first forward call only:

$$h'_{29,close} = h_{29,close} - 2d_{29} \text{ if } C = 1, \quad (6a)$$

$$h'_{29,close} = h_{29,close} + 2d_{29} \text{ if } C = 0. \quad (6b)$$

The experimental arm determines the intervention sign. No sign or truth label is provided to the unmodified model. Prompt tokens, weights, tokenizer, greedy decoding rule, and generated reasoning-prefix IDs are identical between a case's replay and intervention. Only the selected activation is changed.

Suffix generation uses full-vocabulary argmax without a key/value cache, with a ceiling of four tokens. The hook acts only during the first call that chooses the answer-boundary token. If GO is selected, the next pass recomputes the prefix and GO without that activation edit and selects the following token. Thus an EOS-to-GO success requires the exact suffix `[GO, EOS]`, not merely a positive GO margin.

6.3 Controls

Eight control interventions are applied to the same 80 confirmation contexts. Each must retain the complete baseline suffix. The zero perturbation is a sham. Same-truth replacement substitutes the closure activation from the next case of the same truth class, implemented as index (i + 2) modulo the pool length. The remaining controls use three random directions and three directions orthogonal to the fitted direction.

Each random vector is sampled independently for its case and normalized to the primary perturbation norm, $2\|d_{29}\|$. Orthogonal vectors first have their component along d_{29} removed, then receive the same

norm. These are norm-matched controls; same-truth replacement is a separate activation-substitution control. The recorded seeds and exact construction are given in Appendix A.

7 Causal results

Measurement	Exact result
Unmodified trace and suffix	80 / 80
Unmodified fixed-prefix replay	80 / 80
Equal: GO to immediate EOS	40 / 40
Unequal: EOS to GO followed by EOS	40 / 40
Zero-perturbation control	80 / 80 unchanged
Same-truth replacement	80 / 80 unchanged
Three random-direction controls	240 / 240 unchanged
Three orthogonal-direction controls	240 / 240 unchanged
All controls	640 / 640 unchanged

Table 6. Frozen causal-confirmation results. The 640 control observations are eight interventions on 80 contexts, not 640 independent prompts. Every scheduled confirmation case is included.

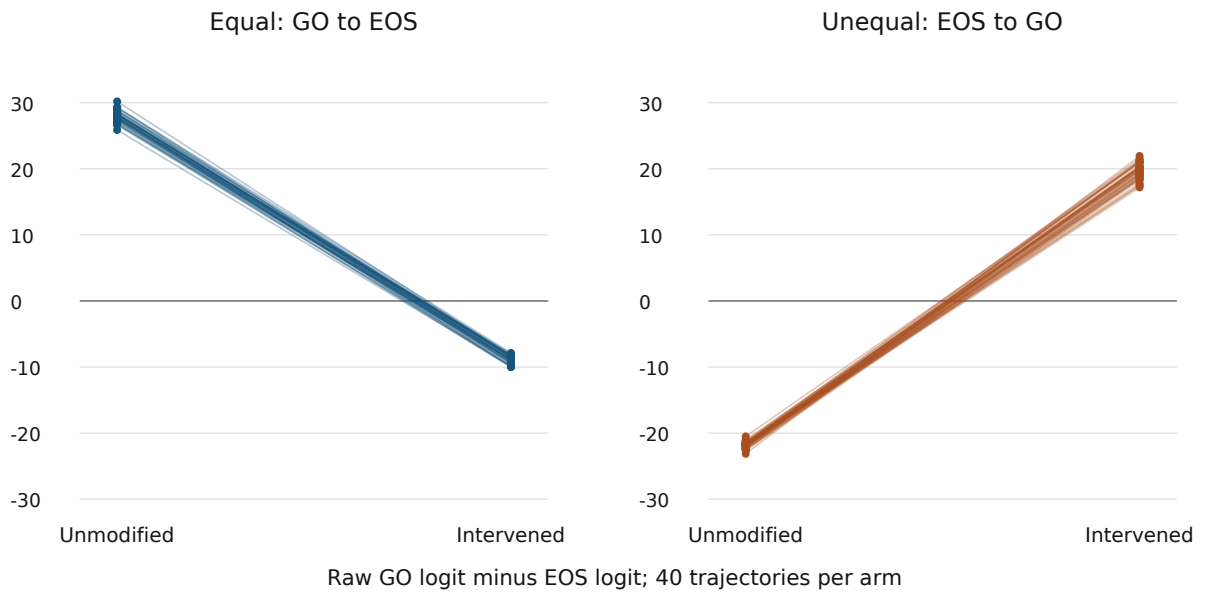

Figure 2. Raw answer-boundary margin before and after the block-29 intervention. Each line is one fixed-prefix context. Every equal-case margin changes from positive to negative; every unequal-case margin changes from negative to positive. Exact generation, not margin sign alone, determines success.

Equal-case margins range from +25.875 to +30.1875 before intervention and from -10.0 to -7.875 after subtraction. Unequal-case margins range from -23.125 to -20.5 before intervention and from +17.1875 to +21.9375 after addition. The smallest intervened full-vocabulary top-two gap is 7.875 in the equal arm and 6.875 in the unequal arm. All decisions are actual greedy winners over the complete vocabulary.

7.1 Exact matched example

The first confirmation pair is `EQ(6953,6953)` and `EQ(6953,4376)`. Its raw baseline generations are:

```
<think>1 1 1 1</think>GO<|im_end|>
<think>0 0 0 0</think><|im_end|>
```

With the same respective prefixes replayed and the block-29 perturbation applied, the complete prefix-plus-suffix sequences are:

```
<think>1 1 1 1</think><|im_end|>
<think>0 0 0 0</think>GO<|im_end|>
```

Condition	Perturbation	M before	M after
EQ(6953,6953)	-2d	+28.1875	-8.2500
EQ(6953,4376)	+2d	-22.4375	+19.1250

Table 7. Exact raw margin reversals for the first confirmation pair. The comparison bits remain correct and byte-identical. The final action reverses.

7.2 Direction readout

The selected direction has 2,560 components and Euclidean norm 131.801727. Recomputing the mean paired block-29 activation difference from the stored fit activations reproduces the saved vector exactly. Transporting this direction through the fitted block-29 lens and applying the unembedding with the final normalization weights gives an unnormalized GO projection of +66.315041 and EOS projection of -98.314674. GO is the highest entry in this linear projection. These values describe a direction in vocabulary space; they are not generation probabilities or the intervened logits reported above.

7.3 Intervention magnitude

For each confirmation case *i*, the nominal perturbation magnitude is compared with the native residual norm at block 29 and the reasoning-closure position:

$$s_i = 2||d_{29}|| \,/\, ||h_{29,\mathrm{close},i}||. \quad (7)$$

Norms are recalculated in float64 from the saved activation and direction tensors. The nominal perturbation norm is 263.603442; the small difference from twice the six-decimal norm in Section 7.2 reflects float32 versus float64 norm evaluation. Table 8 reports per-case ratios before the inference hook casts the perturbation to BF16 and adds it to the residual.

Cases	n	Median ‖h‖	Range ‖h‖	Median s	Range s
Equal	40	219.699	218.034 to 221.524	1.199837	1.189957 to 1.208999
Unequal	40	234.701	230.117 to 239.375	1.123148	1.101217 to 1.145520
All	80	225.820	218.034 to 239.375	1.167738	1.101217 to 1.208999

Table 8. Nominal perturbation-to-residual scale over all 80 confirmation contexts. Medians are calculated across per-case ratios, not by dividing the perturbation norm by a group median residual norm.

The perturbation is 110.12% to 120.90% of the native residual norm, with a median of 116.77%. It is a large activation edit, not a microscopic perturbation. The random and orthogonal controls match this nominal norm and retain their baseline outputs. The result therefore demonstrates directional specificity at the tested scale, not control by an arbitrarily small edit.

8 Interpretation and related work

8.1 What the intervention establishes

A fixed internal activation direction is causally sufficient to reverse native continuation versus termination in both directions under the tested policy. This establishes a causal continuation-control direction, not a unique or necessary native gate. The equal-to-EOS intervention suppresses a licensed final answer. The unequal-to-GO intervention produces an unlicensed final answer despite a correct generated inequality trace. The latter is a controlled violation of the learned policy, not an improvement in its accuracy.

A fixed correct reasoning prefix means that the four-bit comparison trace is generated normally and then held token-for-token fixed during the continuation experiment. It does not mean that every internal state remains unchanged: the intervention deliberately changes a residual activation and its downstream computation. The result separates a fixed correct reasoning transcript from the subsequent action. It does not perturb generation from the beginning or demonstrate an identical internal reasoning trajectory under perturbation.

8.2 Relation to existing methods

Newman et al. [7] study how learning EOS affects hidden-state dynamics and length extrapolation. Their analysis identifies length-dependent representations and states in which EOS becomes the highest-probability prediction. The present task instead fixes a short reasoning format and conditions the presence of a final payload on equality.

Activation Addition [8] and Contrastive Activation Addition [9] establish that differences between contrasting residual activations can steer model outputs. Arditi et al. [10] identify a direction that mediates textual refusal. This study uses that broader intervention methodology to target a different exact outcome: native EOS versus one licensed continuation token, after the reasoning prefix is fixed. It does not introduce activation steering as a new method.

Wu et al. [11] combine state-conditioned activation steering with EOS-weighted supervision for long-form report generation. That work directly connects internal interventions and learned termination. Here, the controlled variable is a fixed equality prerequisite, the negative final payload is exactly empty, and one fixed vector is tested for bidirectional reversal at a single post-reasoning position.

The tuned lens [12] learns layer-wise translators for intermediate predictions. Jacobian Lens [5,6] instead transports activations using averaged derivatives into the final-layer basis. This study uses the latter implementation for readout and an independently estimated activation-difference vector for intervention. The observed block-wise GO/EOS separation is a readout result; the 80 exact reversals are the causal result.

Work on chain-of-thought faithfulness shows that generated explanations need not fully describe the computations determining an answer [13]. The present measurement is narrower and directly controlled: the four comparison bits are objectively correct, and the recorded prefix is unchanged while the final token decision reverses. The experiment measures this separation rather than inferring internal computation from prose.

8.3 Experimental scope

The tested domain is fixed-width decimal equality with one fixed final payload, GO, in one Qwen3-4B-derived checkpoint. The reasoning is a four-bit positional comparison trace. The behavioral result concerns the enumerated frozen sets; the intervention result concerns 40 confirmation pairs under the specified layer, strength, position, and decoding procedure.

The measured direction is sufficient for control under intervention. The experiment does not test its uniqueness, necessity, or completeness as a circuit. Equal and unequal fit examples differ in condition,

correct trace, and baseline action, so the fitted vector does not by itself separate a representation of equality from a representation of the output decision. The six-layer search also follows a training recipe that includes activation noise at block 29.

The unmodified counterfactual is the same fused checkpoint. This study does not attribute the effect to a particular training component or establish the mechanism behind the provider-level observations in [2]. The 16-prefix Jacobian calibration supports the task-specific readouts reported here. Corpus-wide interpretability, other predicates, free-form reasoning, and cross-model transfer are separate experimental questions.

9 Reproducibility

The source releases are pinned to the full commits below. Artifact hashes in Section 9.1 and Appendix B link to their exact files at these revisions.

Artifact	Repository and immutable commit
PCCG-Qwen3-4B	https://huggingface.co/sharthokrayanpal/PCCG-Qwen3-4B `5f3b70d51f068a7a8a108b99f66b2fdfd15e25ce`
Causal study	https://github.com/theonlypal/PCCG-Qwen3-4B-continuation-control `d08b157ac9273974b75d7ada2a894a1afe86e3f2`

Source releases. Repository links open the recorded commit.

All artifact paths in this paper are relative to the root of the identified repository. Download the complete model snapshot at its pinned revision, including `weights/` and `provenance/`. Clone the study at its pinned commit with Git LFS enabled; `git lfs pull` materializes the fitted lens. Appendix B links the evidence files.

From the model repository root, run `reproduce.py` for the behavioral demonstration and `verify.py` for saved-record verification:

```
python reproduce.py
python verify.py
```

The demonstration loads the fused checkpoint and runs the equal/unequal example in Section 2 using explicit greedy generation settings. It requires Linux, CUDA, and sufficient memory for native BF16 inference; its preflight requires at least 12 GiB of free GPU memory. It installs the pinned Python inference dependencies in an external cache when needed. The separate verification command hashes the files and recalculates all 4,096 saved results without model execution.

For mechanistic replay, run from the study repository root using `requirements.txt` and `reproduce.py`. Set `PCCG_MODEL` to the downloaded model snapshot and `PCCG_OUTPUT` to a new output location outside both checkouts. These variables specify execution locations, not artifact identities.

```
python -m pip install -r requirements.txt
python reproduce.py --model "$PCCG_MODEL" \
  --output "$PCCG_OUTPUT"
```

The output directory must not already exist and must be outside both packages. The command verifies the inputs, stages the recorded source and cases, fits the Jacobian matrices, repeats development selection, and executes confirmation and controls. It performs activation differentiation and inference with the fixed checkpoint. It does not train model weights. The saved study can be checked separately:

```
python verify.py --model "$PCCG_MODEL"
```

Component	Recorded or pinned setting
Study GPU	RTX PRO 4500 Blackwell, 32 GB
Model arithmetic / attention	BF16 / PyTorch SDPA
PyTorch	2.8.0
Transformers	4.56.2
PEFT / Accelerate / Safetensors	0.17.1 / 1.10.1 / 0.6.2
Study seed / CPU threads	942889 / 4
Python reproduction requirement	3.11 or later

Table 9. Execution settings from the companion packages. The pinned source and input artifacts define the reproduction target.

The reported behavioral and causal scores were recalculated from the packaged raw records. Both package verifiers passed, checking 37 model-package files and 115 study-package files. The study inventory includes its Git LFS attributes; the fitted lens is verified as materialized bytes, not as a pointer file. The figures are computed from the saved per-case readouts and raw logits.

9.1 Portable training-provenance audit

The model repository contains `provenance/stages/`, `provenance/lineage.json`, `provenance/audit.json`, and `provenance/verify.py`. From its root, set `PCCG_STUDY` to the pinned study checkout and run:

```
python provenance/verify.py --study "$PCCG_STUDY"
```

This standard-library command checks the provenance manifest and recalculates the recorded lineage and overlap results using only the two release directories. It does not require ancestor weight files, private host paths, a GPU, or historical training execution. The supplement has its own manifest; existing model and study records retain their original hashes.

```
Model: provenance/SHA256SUMS
ec298a4406cbeb5a9e2c3fa3b9fe4feac32a59402401127617b2d25f49e7d82d
```

Table 8 is calculated directly from the study repository's `results/activations.pt` and `results/confirmation-prefixes.json`, without model inference.

10 Conclusion

The fused model implements the tested prerequisite-conditioned policy with exact reasoning and native termination on 4,096 recorded conditions. On 40 confirmation pairs, one fixed block-29 activation direction reverses both continuation outcomes while the correct generated reasoning prefix is held fixed. Jacobian Lens characterizes the corresponding vocabulary readout; direct activation intervention establishes control. The resulting artifact is a trained continuation policy with an independently inspectable internal direction sufficient to reverse its final GO/EOS decision.

Appendix A Exact interface and intervention details

A.1 Serialized request

The exact serialized equal-case request is shown below. Line breaks are LF bytes. The serializer adds one final LF after the assistant header. There is no space around the comma in the condition and no appended `<think>` in the prompt.

```
<|im_start|>system
A binding condition is the prerequisite that must hold for valid continuation.<|im_end|>
<|im_start|>user
CONDITION: EQ(2019,2019)<|im_end|>
<|im_start|>assistant
```

The unequal request substitutes `EQ(2019,2011)`. The system sentence, role delimiters, and all other bytes are unchanged. Tokenization uses the checkpoint's own chat template. The generated token sequences for this pair are exactly:

```
Equal:
[151667, 16, 220, 16, 220, 16, 220, 16, 151668, 15513, 151645]
Unequal:
[151667, 16, 220, 16, 220, 16, 220, 15, 151668, 151645]
```

Token	ID	Role
GO	15513	Licensed final payload
<|im_end|>	151645	Native EOS
<think>	151667	Generated reasoning opener
</think>	151668	Generated reasoning closure
0 / 1	15 / 16	Comparison bits
ASCII space	220	Between comparison bits

Table A1. Tokens used in the exact output contract. No token follows the final EOS in a recorded sequence.

A.2 Position and control construction

The 49-token intervention input is the 40-token serialized prompt concatenated with the 9 generated tokens through `</think>`. The edited position is therefore token index 48, at the output of transformer block 29. The full generation budget is 64 tokens; the fixed-prefix suffix budget is 4 tokens. These are ceilings, not forced stopping points.

For a control case index i and control replicate k in {0,1,2}, the CPU random generator seed is $942991 + i + 10000k$. Orthogonal controls add 50000 to that seed. A sampled standard-normal vector z is normalized to $2||d||$. For orthogonal controls, let $u = d/||d||$ and replace z by $z - (z \cdot u)u$ before normalization. Same-truth replacement uses the captured activation at index $(i + 2)$ modulo 80, which preserves truth because cases alternate equal and unequal.

The hook is registered before the suffix loop and removed in a finally block. Its call counter permits one edit only. The recorded suffix loop recomputes the full prefix without caching on subsequent steps. This bounds the intervention to the first continuation decision.

Appendix B Artifact identities and evidence map

Each SHA-256 below links to its exact file at the commit pinned in Section 9. The model's `weights/identity.json` binds the weight and tokenizer hashes. The parent identity identifies the checkpoint before the final adapter update.

```
Fused model identity
7a7c7943f465259528de409eb8d02b5bf600378f738229b9cb482e889cbc8d1c
```

```
Parent checkpoint identity
7280f631ae071bee30c665bb07659da2125789f2bd963e9b6f1e0e7ac14b9dff
```

```
Fixed direction.json
b04d186311bcd921eae7b1941ec21545f049990a4f3bc4b09e0105ba39c2a344
```

```
Fitted jacobian-lens.pt
5d55e51351b2e661fdbd73d6781b76e4351ef90827d44db8fc55bb9f8485809e
```

```
Calibration prefixes
679de806fe62ac3532445b7ecc7bf3fc6cec4b042eb5fb370105d6bde6b4b740
```

```
Model package SHA256SUMS
fa4d6cbb6282b27f96a283f949ecdabf0254c02d15829fdea9f8b29324244e91
```

```
Study package SHA256SUMS
36e35bbc93bce643728d5bde86e1e83b566dafa0156c06b66e867890955409a1
```

B.1 Repository-relative evidence paths

Evidence	Exact path at the pinned repository commit
Model identity	`weights/identity.json`
Final-stage data	`training/selected-training.json`

Behavioral cases	`evaluation/cases/regression.json`
	`evaluation/cases/dev.json`
	`evaluation/cases/qualification.json`
	`evaluation/cases/final.json`

Behavioral raw records	`evaluation/fused-regression.json`
	`evaluation/fused-dev.json`
	`evaluation/fused-qualification.json`
	`evaluation/fused-final.json`

Study implementation	`source/study.py`
Jacobian fitting implementation	`source/vendor/jlens/fitting.py`
Selection grid	`results/direction-development.json`
Frozen selected intervention	`results/intervention-frozen.json`
Fitted activations and direction	`results/activations.pt` `results/direction.json`
Baseline and replay	`results/confirmation-baseline.json` `results/confirmation-replay.json`
Causal suffixes and logits	`results/confirmation-primary.json`

Control suffixes and logits	`results/confirmation-control-self.json`
	`results/confirmation-control-same_truth.json`
	`results/confirmation-control-random_0.json`
	`results/confirmation-control-random_1.json`
	`results/confirmation-control-random_2.json`
	`results/confirmation-control-orthogonal_0.json`
	`results/confirmation-control-orthogonal_1.json`
	`results/confirmation-control-orthogonal_2.json`

Layer readouts	`results/confirmation-readouts.json`

Table B1. Every path is an exact, clickable repository-relative path. The first four rows refer to the Hugging Face model repository; subsequent rows refer to the GitHub study repository.

References

[1] Rayan Pal. 2026. *The Binding Condition for Artificial General Intelligence.* March 24. https://doi.org/10.5281/zenodo.19211116

[2] Rayan Pal. 2026. *Cross-Vendor Semantic Void Matrix.* July 29. Zenodo. https://doi.org/10.5281/zenodo.21696066

[3] An Yang, Anfeng Li, Baosong Yang, et al. 2025. *Qwen3 Technical Report.* arXiv:2505.09388. https://arxiv.org/abs/2505.09388

[4] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. *LoRA: Low-Rank Adaptation of Large Language Models.* arXiv:2106.09685. https://arxiv.org/abs/2106.09685

[5] Wes Gurnee, Nicholas Sofroniew, Adam Pearce, et al. 2026. *Verbalizable Representations Form a Global Workspace in Language Models.* Transformer Circuits, July 6. https://www.transformer-circuits.pub/2026/workspace/index.html

[6] Anthropic. 2026. *Jacobian Lens.* Reference implementation accompanying the global workspace study. The executed implementation is included in the causal-study package. https://github.com/anthropics/jacobian-lens

[7] Benjamin Newman, John Hewitt, Percy Liang, and Christopher D. Manning. 2020. *The EOS Decision and Length Extrapolation.* Proceedings of BlackboxNLP, pp. 276-291. https://aclanthology.org/2020.blackboxnlp-1.26/

[8] Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. *Steering Language Models With Activation Engineering.* arXiv:2308.10248. https://arxiv.org/abs/2308.10248

[9] Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. *Steering Llama 2 via Contrastive Activation Addition.* Proceedings of ACL, pp. 15504-15522. https://aclanthology.org/2024.acl-long.828/

[10] Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. 2024. *Refusal in Language Models Is Mediated by a Single Direction.* arXiv:2406.11717. https://arxiv.org/abs/2406.11717

[11] Ning Wu, Rui Liu, Xinkun Lin, Weixing Chen, Jinxi Xiang, Tao Wei, Lina Yao, and Mingjie Li. 2026. *Not All Tokens Matter Equally: Dynamic In-context Vector Distillation with Decisive-Token Supervision for Long-form Medical Report Generation.* arXiv:2605.27194. https://arxiv.org/abs/2605.27194

[12] Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. 2023. *Eliciting Latent Predictions from Transformers with the Tuned Lens.* arXiv:2303.08112. https://arxiv.org/abs/2303.08112

[13] Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. *Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting.* arXiv:2305.04388. https://arxiv.org/abs/2305.04388

[14] Qwen Team. 2025. *Qwen3-4B model card and tokenizer interface.* Hugging Face. https://huggingface.co/Qwen/Qwen3-4B