

Minimal Witness Reinforcement Learning

T. Y. Tsui¹, Zihao Ye², Pengxiang Cai², Yanchao Li², Yuqiang Li^{2,†}, Zhehong Ai^{2,†}

¹University of Pennsylvania

²Shanghai AI Laboratory

[†]Corresponding authors.

tuenyue.tsui@gmail.com

Abstract

“What are the irreducible conditions that are sufficient to produce an outcome?” is one of the most common questions that recur across computation and science. Its answers, the minimal sufficient witnesses, are what we mean by explanations, mechanisms and reasons. These problems usually ask for multiple minimal witnesses, yet standard RL methods may reveal only one solution or redundant ones. We formalize this problem as minimal-witness identification and introduce Minimal-Witness Reinforcement Learning (MWRL). MWRL takes the union of the sets certified by successful proposals sampled from the policy and credits each proposal for the coverage the group union would lose without that proposal. This credit assignment, derived directly from the problem definition, unifies the demands for minimality and recovery of alternatives from a single black-box verifier bit. Under this principle, we derive a value iteration planner that recovers the entire family of witnesses and a policy gradient method that can scale to large language models. Across different experimental settings, MWRL recovers most minimal witnesses, while other methods return redundant supersets or a single witness. By making witness families learnable from verifier feedback, MWRL expands the scope of reinforcement learning beyond single-solution optimization. Our code is available at <https://github.com/TSUITUENYUE/MWRL>.

1 Introduction

Sufficiency questions recur across computation and science. In Boolean logic, we ask which literals force a formula to be true (Quine 1952). In chemical engineering, we ask which reaction conditions raise yields above a threshold. The same question arises when identifying the components that implement a network behavior (Conmy et al. 2023; Wang et al. 2023) or the input features sufficient for a prediction (Ribeiro, Singh, and Guestrin 2018; Carter et al. 2019). In each case, we ask which set of factors is sufficient for an outcome. For those tasks, the answer is an **irreducible** set, one from which no element can be removed without losing sufficiency. Such a set is defined as a minimal sufficient witness. Usually, a task may have several minimal sufficient witnesses.

Minimality here is easy to misunderstand as a preference for minimum size. A sufficient set is minimal when none of its proper subsets is sufficient. Different minimal witnesses are incomparable and together form an antichain of alternatives (Méloux et al. 2025). Minimality therefore defines a

valid answer and cannot be replaced by a size penalty in the reward. Another misunderstanding is treating the alternatives in the antichain as a preference for diversity. Diversity methods rely on a metric, behavior descriptor or kernel (Pathak et al. 2017; Haarnoja et al. 2018; Lehman and Stanley 2011; Mouret and Clune 2015). In minimal-witness identification, the subset relation determines which sufficient sets belong in the answer. If the ground-truth answer contains only one witness, the best answer is to only recover that witness. If it contains three, the answer should return all three. Diversity methods, however, keep rewarding more varied solutions without deciding when the answer is complete.

Take a server diagnosis as an example. Assume it goes down when its power supply fails, when both of its mirrored disks fail, and when one mirrored disk plus a CPU burns out. There are several concepts we can explain here. Sufficiency: the faults $\{\text{disk}_1, \text{power}\}$ together will make the server fail, but the disk failure is redundant in this set because $\{\text{power}\}$ is already sufficient. Once $\{\text{power}\}$ alone is verified, every set of faults containing it is also known to be sufficient. Minimality: the set $\{\text{power}, \text{disk}_1\}$ has the same size as the pair $\{\text{disk}_1, \text{disk}_2\}$, but only the pair is a minimal witness, while $\{\text{power}\}$ alone also counts as a minimal witness. Both $\{\text{disk}_1, \text{CPU}\}$ and $\{\text{disk}_1, \text{disk}_2\}$ are minimal witnesses because each pair causes an outage and neither component alone does. The pairs share a disk, but neither contains the other. Antichain: all irreducible sets matter. Imagine you are the maintainer of the server. To prevent the outage from happening again, you need to know all the minimal causes, and the antichain is the set of all of them.

When the predicate is available as a formula or constraint system, we have algorithms that enumerate its minimal witnesses. These witnesses appear as prime implicants of Boolean functions (McCluskey 1956), as minimal diagnoses and unsatisfiable subsets (Reiter 1987; Marques-Silva, Janota, and Belov 2013; Liffiton et al. 2016), and as minimal hypergraph transversals (Eiter and Gottlob 1995; Fredman and Khachiyan 1996; Murakami and Uno 2014). Other algorithms find them by querying a monotone predicate (Bioch and Ibaraki 1995; Junker 2004; Janota and Marques-Silva 2016). These algorithms need a deterministic predicate and run a new search for every context. However, many important settings do not expose such structure. They provide only a verifier that accepts or rejects a proposal, and we have to

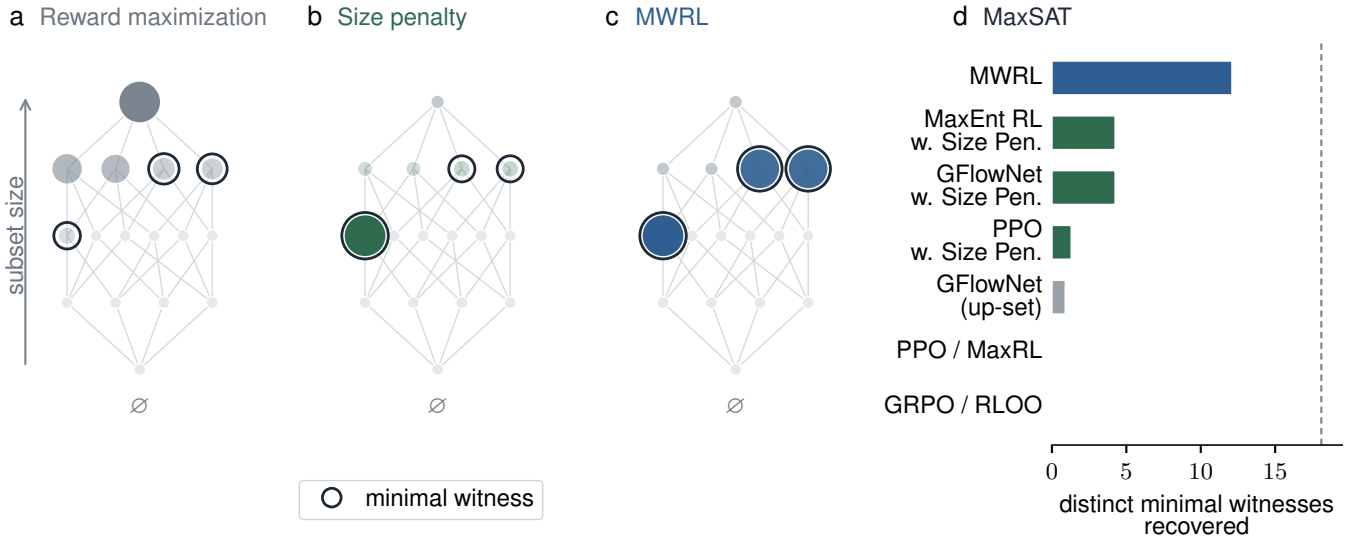


Figure 1: From proposal-local reward to the group-certified region. **(a) to (c)** Terminal probability on $2^{\mathcal{E}}$ for $\mathcal{E} = \{a, b, c, d\}$; the target antichain is ringed. A success-only reward spreads probability over all sufficient sets (a) and a size penalty collapses onto the smallest witness (b); MWRL credits each success for the part of the group’s certified region that only it certifies and puts its probability on the antichain (c). **(d)** Finite-budget recovery on enumerable MaxSAT instances: MWRL recovers most of each instance’s antichain, and methods that value each proposal in isolation recover at most a quarter of it. The size-penalized MaxEnt RL and GFlowNet bars use the best penalty of a sweep, and the dashed vertical line is the mean antichain size $|\mathcal{M}|$.

identify minimal witnesses from this verifier feedback alone.

Since each verifier call returns only success or failure, as in reinforcement learning with verifiable rewards (RLVR), we can use this verifier bit as a reward and train a policy to propose subsets. A success-only reward gives a minimal witness and each of its sufficient supersets the same reward. A size penalty gives smaller sets more reward, and the policy then concentrates on the smallest witness even when larger witnesses are also minimal. Both rewards **value each proposal in isolation** and therefore cannot tell a new minimal witness from an accepted set that contains another accepted proposal of the group. A better optimizer does not solve this problem, because the reward itself does not contain this information.

Whether an accepted proposal adds new information depends on the other proposals in its group. Under monotone sufficiency, every superset of a sufficient set is also sufficient. A single verifier call on a set therefore certifies all of its supersets. When one accepted set contains another, the region certified by the smaller set already includes the region certified by the larger set. When neither set contains the other, each of the two sets certifies supersets outside the region of the other. The reward therefore has to compare the proposals of a group.

We formalize this problem as minimal-witness identification and introduce Minimal-Witness Reinforcement Learning (MWRL). The supersets certified by an accepted proposal form an up-set of the subset lattice. MWRL measures the union of these up-sets over a group and gives each proposal the part of this value that would be lost without it. A proposal that repeats or contains another accepted proposal receives zero credit. When the union is measured by

a probability measure that gives every set positive mass, a proposal receives positive credit whenever it certifies a set outside the region certified by the other proposals. The objective therefore rewards removing unnecessary elements from an accepted proposal and finding minimal witnesses that the rest of the group has not found.

Contributions.

- We formalize *minimal-witness identification*: for a context, recover the antichain of subset-minimal sufficient witnesses from a black-box verifier alone (Section 2).
- We propose Minimal-Witness Reinforcement Learning (MWRL). Its objective leads us to a planner and a scalable group policy gradient for minimal-witness identification (Section 3).
- We evaluate MWRL across different settings: on prime-implicant enumeration against the ground-truth antichain to verify our theory; on scientific variable identification in Suzuki–Miyaura coupling to test amortization; and on sparse-circuit recovery in frozen language models under a raw verifier that can violate monotonicity to test the robustness of our method (Section 4).

We discuss related work in Appendix H and other settings that fit this formulation in Appendix I.

2 Problem Formulation

Let $\mathcal{E} = \{1, \dots, n\}$ be the ground set of elements (notation in Table 1) and let $s : 2^{\mathcal{E}} \rightarrow \{0, 1\}$ be a *sufficiency predicate*: $s(S) = 1$ if and only if the subset S is sufficient for the target outcome. We assume monotonicity: adding more elements cannot turn a success into a failure, so $S \subseteq T$ implies $s(S) \leq s(T)$.

$s(T)$. A *witness* is any S with $s(S) = 1$. A *minimal witness* is a witness none of whose proper subsets is a witness. The minimal witnesses form an antichain (no member contains another),

$$\mathcal{M} = \min_{\subseteq} \{ S \subseteq \mathcal{E} : s(S) = 1 \}. \quad (1)$$

These sets are the prime implicants of s (Quine 1952; Crama and Hammer 2011).

Minimal-witness identification fixes a context c with predicate s_c and antichain $\mathcal{M}(c)$, and seeks a policy $\pi_\theta(\cdot | c)$ whose sampled proposals recover $\mathcal{M}(c)$. We report two metrics: the born-minimal rate is the fraction of successful proposals that are minimal, and antichain recall is the fraction of the target family recovered as distinct minimal witnesses.

A policy that always proposes the same witness cannot recover a family. We therefore train a policy whose terminal distribution is $q_\theta(\cdot | c)$, using groups $S_{1:K} \stackrel{\text{iid}}{\sim} q_\theta$ of K proposals per update. To recover $\mathcal{M}(c)$, the return of a group has to depend on two properties of each successful proposal. The first is minimality, which holds when no proper subset of the proposal is sufficient. The second is the relation of the proposal to the other proposals in its group, since a proposal that repeats or contains another successful proposal is not a new witness. A group return is *proposal-separable* when it sums per-proposal rewards $\sum_i r(S_i, s_c(S_i))$. Such a return evaluates each proposal without comparing it with the others in the group. A reward is *local* when it is the same function of $(S, s_c(S))$ for every monotone predicate. A local reward cannot depend on minimality either, because the verifier bit of a proposal is the same whether or not a proper subset of the proposal is sufficient.

Theorem 1 (Relational necessity). *Let a method target the maximizers of the separable objective $J_{\text{sep}}(q) = \mathbb{E}_{S_{1:K} \sim q^{\otimes K}} [\sum_i r(S_i, s_c(S_i))]$, the law $q \propto r(\cdot, s_c(\cdot))$ for a local reward $r \geq 0$, or the maximizers of an objective that depends on q only through the success probability $\Pr_{S \sim q}[s_c(S) = 1]$. Then, for $|\mathcal{E}| \geq 3$, some monotone predicate with $|\mathcal{M}(c)| \geq 2$ has a target law whose support differs from $\mathcal{M}(c)$.*

Each condition of Theorem 1 describes a category of existing methods (Appendix A.3). Separable objectives, including PPO, GRPO, and RLOO (Schulman et al. 2017; Shao et al. 2024; Ahmadian et al. 2024), sum a reward over the proposals of a group. A law that always proposes one highest-reward set maximizes a separable objective, which is linear in q . When the predicate has two minimal witnesses, this law misses one of them. Proportional samplers, including Max-Ent RL and GFlowNet (Haarnoja et al. 2018; Bengio et al. 2021), sample from a law proportional to a local reward. A local reward gives a set the same weight under every predicate that accepts it, including a predicate under which the set is not minimal. Success-probability objectives, including maximum likelihood RL (Tajwar et al. 2026) and pass@ k objectives, depend on q only through the success probability. Under a monotone predicate, if any set is sufficient, the full set $\mathcal{E}$ is also sufficient because it contains that set. A law that moves all the probability of sufficient sets onto $\mathcal{E}$ keeps the

same success probability and gives zero probability to every minimal witness. A size penalty added to a separable objective or a proportional sampler keeps its reward local, and the method stays in its category. To recover the whole antichain under every monotone predicate, a method therefore needs a return outside these three categories, one that depends on which sets appear together in a group.

3 Minimal-Witness Reinforcement Learning

3.1 Monotonicity and the certified region

A rollout ends with a proposed subset, which the verifier accepts or rejects. MWRL uses the proposed subsets and their verifier bits to compute a group return. The construction applies to any episodic policy whose terminal subset is scored by $s_c(S_T) \in \{0, 1\}$ (Appendix C.1). In our experiments, the policy walks the subset lattice $2^{\mathcal{E}}$ with state $x_t = (c, S_t, t)$ and single-element moves, inserting from $\emptyset$ or pruning from the full set.

Under monotonicity, a successful terminal subset S certifies all its supersets, and a failure rules out all its subsets. We write the certified set as the up-set $\uparrow S = \{T \subseteq \mathcal{E} : T \supseteq S\}$, and the sufficient subsets form $F(c) = \{S : s_c(S) = 1\} = \uparrow \mathcal{M}(c)$. For a group with verifier bits $y_i = s_c(S_i)$, we define the certified region of the successes $W = \{S_i : y_i = 1\}$ as the finitely generated up-set

$$U(W) = \bigcup_{S \in W} \uparrow S. \quad (2)$$

If a set in W contains another set in W , the up-set of the smaller set already includes the up-set of the larger set. We define such a larger set as *redundant*. Removing redundant sets and repeated sets from W therefore leaves $U(W)$ unchanged. Conversely, the minimal elements of $U(W)$ are the minimal sets of W (Davey and Priestley 2002), which we define as the *group-minimal successes*.

Some raw predicates can reject a superset of an accepted set. For such a predicate, we can restore monotonicity by testing whether any subset of S is accepted. This gives the existential closure

$$s_\exists(S) = \mathbf{1}[\exists Q \subseteq S : s(Q) = 1].$$

The closure is monotone and has the same minimal accepted sets as the raw predicate. Without the closure, we can still check a weaker property. An accepted set is *1-minimal* when the verifier rejects it after the removal of any single element, and under a monotone predicate a 1-minimal set is minimal. The language-model experiment uses the raw circuit verifier even though it can violate monotonicity. We measure these violations and report born-minimal rates as 1-minimality (Appendix A.4).

3.2 The group return

MWRL values the successes of a group together through their certified region, with the group return

$$G(W) = \mathcal{R}(U(W)), \quad (3)$$

where $\mathcal{R}$ assigns a value to each up-set. We train the policy to maximize $J_K(\theta) = \mathbb{E}[\mathcal{R}(U_K)]$ for groups of K proposals

Symbol	Meaning	Symbol	Meaning
General notation		Problem formulation	
$\mathcal{E}, n$	Ground set and its size	$s_c(S)$	Sufficiency predicate for context c
$S, T, Q \subseteq \mathcal{E}$	Proposed or auxiliary subsets	$\mathcal{M}(c), L$	Minimal-witness antichain and its size
$2^{\mathcal{E}}, \subseteq$	Subset lattice and inclusion order	$\min_{\subseteq}$	Inclusion-minimal members of a family
$ S $	Number of elements in S	$F(c)$	Set of all sufficient subsets
$\uparrow S, \downarrow S$	Supersets and subsets of S	$s_{\exists}$	Monotone closure of a raw predicate
c	Context for the task	r, J_{sep}	Local reward and separable objective
Certified regions and coverage		Value iteration	
W, W_K	Successful proposals; distinct group successes	F, b	Certified region and commitments left
$U(W), U_K$	Region certified by a set or group	$\Delta_{\mathcal{R}}(S \mid F)$	Value of adding $\uparrow S$ beyond F
$\mathcal{R}, \mathcal{R}_{\#}$	Region valuation and counting measure	$V_b^*(F)$	Optimal value with b commitments
$\mu, \mathcal{R}_{\mu}$	Base measure and coverage measure $\mu(U)$	F_k, S_k	Region and choice at planner step k
$\nu(S), p$	Witness value $\mu(\uparrow S)$ and the parameter of μ	λ	Size penalty in scalar planning reference
Policy and sampling		Group value and deletion credit	
θ, π_{θ}	Policy parameters and proposal policy	$G(W), J_K$	Group return and expected objective
x_t, a_t, τ_i	State, action, and trajectory i	$g, \mathcal{A}(W)$	Symmetric value and minimal group successes
$q_{\theta}(S \mid c)$	Terminal proposal law	D_i	Minimal successes excluding proposal i
$\psi(\tau), P_{\theta}(\tau \mid c)$	Terminal map and trajectory law	$U_{-i}, U_{-i, -j}$	Region without indexed proposals
$\rho(\tau)$	Policy-independent trajectory factors	$A_i, \bar{A}_i$	Deletion credit and baseline-adjusted credit
K, B	Training group size and deployment budget	b_i	Baseline independent of trajectory i
y_i	Verifier bit $s_c(S_i)$	$\hat{A}_i, N$	Monte Carlo credit and sample count
$\bar{g}_N$	Mean gradient over N rollout groups	$T^{(m)}$	Base-measure draw for Monte Carlo credit
Derivations and diagnostics		LLM circuit experiment	
ϵ	Lower bound on each witness’s proposal mass	$P_i, P_{S,i}, P_{\emptyset,i}$	Clean, retained, and fully ablated token laws
$\rho_{\theta}(T), p_m$	Certification and witness probabilities	$\text{KL}(S), \text{KL}_{\emptyset}$	Mean clean-to-masked and clean-to-empty KL
$\mathcal{A}_q$	Successful subsets with positive proposal mass	α	Required fraction of the clean-to-empty KL gap
$p_t, \bar{p}$	Elementwise inclusion probabilities and bound	KL_{held}	Normalized held-out divergence retained
$\Phi(S)$	Canonical witness returned by a canonicalizer	$s_0(S)$	Raw circuit verifier
σ	Standard deviation in experiment figures	P	Number of probe prompts

Table 1: Notation grouped by its role in the paper. Symbols used only in derivations or diagnostics are included for reference. We omit c when it is fixed; i, j index proposals, t policy steps, and k planner steps.

(Figure 1c). Submodular and global RL also optimize non-additive utilities, but the task supplies them (Prajapat et al. 2024; De Santi, Prajapat, and Krause 2024). MWRL builds its group return from the verifier bits and the subset order.

Theorem 2 (Representation). *Let g be a symmetric group value on $(S_i, y_i)_{i \leq K}$ that vanishes on the empty group, depends only on the success multiset, and is unchanged by deleting a duplicate or redundant success. Then $g = \mathcal{R}(U(W))$ for some set function $\mathcal{R}$ on up-sets with $\mathcal{R}(\emptyset) = 0$, and g is non-decreasing in added successes iff $\mathcal{R}$ is monotone. The proof is in Appendix A.*

Two groups with the same minimal successes certify the same $U(W)$ and, under the theorem’s assumptions, have the same value. The certified region uniquely determines those minimal successes. A group value that ignores repeats and redundant successes can therefore be written as a function of the certified region. MWRL chooses how to value that region through $\mathcal{R}$.

Throughout, we fix a full-support probability measure μ on $2^{\mathcal{E}}$, the base measure, and value a region by the coverage measure $\mathcal{R}_{\mu}(U) = \mu(U)$. Computing this return requires only the proposed subsets of a group, their verifier bits, and μ . A canonicalizer is a routine that shrinks an accepted set to

a minimal witness. When a cheap and reliable canonicalizer is available, the group return can count the distinct witnesses it returns (Appendix B).

3.3 Minimal-witness value iteration

With this group return, we can define a Bellman equation for minimal-witness identification. The planner commits successful proposals one at a time. Its state $F \subseteq F(c)$ is the union of up-sets certified by earlier commitments. The value of adding a successful S is $\Delta_{\mathcal{R}}(S \mid F) = \mathcal{R}(F \cup \uparrow S) - \mathcal{R}(F)$. With b commitments remaining, the Bellman value is

$$V_b^*(F) = \max_{S: s_c(S)=1} \{ \Delta_{\mathcal{R}}(S \mid F) + V_{b-1}^*(F \cup \uparrow S) \}, \quad (4)$$

$$V_0^*(F) = 0.$$

Backward iteration computes the optimal cover for a fixed budget on a finite lattice (Bellman 1957). To enumerate the complete antichain, the planner can continue choosing successes without fixing a budget. The update is

$$F_{k+1} = F_k \cup \uparrow S_k, \quad S_k \in \arg \max_{S: s_c(S)=1} \Delta_{\mathcal{R}}(S \mid F_k), \quad (5)$$

and it stops when the largest marginal is zero.

Proposition 1 (Antichain recovery by value iteration). *Let $\mathcal{R} = \mathcal{R}_\mu$ for a full-support measure μ , and assume that the maximization in (5) is exact. Starting from $F_0 = \emptyset$, the planner selects a new member of $\mathcal{M}(c)$ at every step and halts after $|\mathcal{M}(c)|$ steps at $F^* = \uparrow \mathcal{M}(c) = F(c)$.*

An undiscovered minimal witness has positive marginal because its own lattice point is not covered by any other minimal witness. Every non-minimal successful set with positive marginal contains an undiscovered minimal witness, and its marginal is smaller than the marginal of that witness. In contrast, value iteration on a scalar reward with a size penalty selects one highest-valued witness and stops at that single element of $\mathcal{M}(c)$.

3.4 The MWRL policy gradient

The value iteration planner requires enumeration of the up-set states and maximization over every successful subset, so we use it on small ground sets. For larger spaces, we optimize the same MWRL group return with an on-policy gradient. Given K proposals $S_1, \dots, S_K$ with verifier bits $y_i = s_c(S_i)$, let $U_K = \bigcup_{j: y_j=1} \uparrow S_j$ and $U_{-i} = \bigcup_{j \neq i, y_j=1} \uparrow S_j$. MWRL credits proposal i by its deletion credit, the change in the group return when proposal i is removed,

$$A_i = \mathcal{R}(U_K) - \mathcal{R}(U_{-i}). \quad (6)$$

The second term depends only on the other trajectories, so subtracting it leaves the expected score-function gradient for trajectory i unchanged. Under the coverage measure $\mathcal{R} = \mathcal{R}_\mu$, the resulting credit is the probability mass of the sets certified only by proposal i : $A_i = y_i \mu(\uparrow S_i \setminus U_{-i})$. The certified regions of different successes can overlap, since $\uparrow S \cap \uparrow T = \uparrow(S \cup T)$, and $\mathcal{R}_\mu$ counts each overlap once. A new success cannot reduce coverage, and it adds less when earlier successes already cover much of its up-set, which makes $\mathcal{R}_\mu(U(W))$ monotone and submodular in the selected successes (Nemhauser, Wolsey, and Fisher 1978). The deletion credit is therefore the submodular marginal gain of the coverage measure (Figure 2). The planner credits a witness by $\Delta_{\mathcal{R}}(S \mid F)$ against the committed up-set, while (6) evaluates the same marginal against the up-set certified by the rest of the sampled group.

How strongly the credit favors a smaller successful set depends on μ . We define the witness value of S as $\nu(S) := \mu(\uparrow S) = \Pr_{T \sim \mu}[T \supseteq S]$, equal to $2^{-|S|}$ under the independent- $\frac{1}{2}$ product measure. A strict subset has greater witness value ν , while incomparable witnesses contribute distinct parts of the union. For a single success, μ determines how much its witness value $\nu(S)$ rises when an element is removed. We require the same multiplicative increase whether the current subset is large or small.

Proposition 2 (Rank-uniform refinement). *Let ν depend on S only through $|S|$. The refinement ratio $\nu(|S|)/\nu(|S|+1)$ is independent of $|S|$ if and only if $\nu(S) = p^{|S|}$ for some $p \in (0, 1)$, realized by the product measure that includes each element independently with probability p . The proof is in Appendix D.*

Removing one element multiplies the witness value by $1/p$, regardless of the current set size. The only free constant

Algorithm 1 On-policy MWRL.

Require: policy π_θ , group size K , valuation $\mathcal{R}$, verifier access s_c

- 1: **for** each update **do**
- 2: roll out $\tau_1, \dots, \tau_K$ under π_θ for context c , with terminal subsets $S_1, \dots, S_K$
- 3: call the verifier once per terminal: $y_i = s_c(S_i)$
- 4: $U_K \leftarrow \bigcup_{j: y_j=1} \uparrow S_j$
- 5: $U_{-i} \leftarrow \bigcup_{j \neq i, y_j=1} \uparrow S_j$ for each i
- 6: $A_i \leftarrow \mathcal{R}(U_K) - \mathcal{R}(U_{-i})$
- 7: $\tilde{A}_i \leftarrow A_i$, optionally adjusted by the l2o baseline
- 8: ascend $\sum_{i,t} \tilde{A}_i \nabla_\theta \log \pi_\theta(a_{i,t} \mid x_{i,t})$
- 9: **end for**

is p . We define the geometric measure as this product measure and use $p = 0.7$ throughout (Appendix D).

The policy gradient (Williams 1992; Sutton et al. 1999) is

$$\nabla_\theta J = \mathbb{E} \left[\sum_i A_i \sum_t \nabla_\theta \log \pi_\theta(a_{i,t} \mid x_{i,t}) \right].$$

We derive it from the grouped objective $J_K = \mathbb{E}[\mathcal{R}(U_K)]$ in Appendix C and give the training loop in Algorithm 1.

For monotone $\mathcal{R}$, the deletion credit is non-negative. Using it directly as $\tilde{A}_i = A_i$ gives the leave-one-out estimator *l1o*. To let the credit take negative values without changing the expected gradient, we compute the baseline for trajectory i from the other trajectories (Mnih and Rezende 2016; Ahmadian et al. 2024). The leave-two-out estimator *l2o* subtracts such a baseline,

$$\tilde{A}_i = A_i - \left[\mathcal{R}(U_{-i}) - \frac{1}{K-1} \sum_{j \neq i} \mathcal{R}(U_{-i,-j}) \right],$$

where $U_{-i,-j}$ is the up-set certified without proposals i and j (Appendix C, Figure 5).

Proposition 3 (Credit support). *If $\mathcal{R} = \mathcal{R}_\mu$ for a full-support measure μ , then*

$$A_i > 0 \iff y_i = 1 \text{ and } \nexists j \neq i : y_j = 1, S_j \subseteq S_i.$$

The condition compares each successful proposal with the others by subset inclusion, the same order used to define $\mathcal{M}$ in (1). A globally non-minimal proposal may still receive positive credit when the group does not contain a successful subset of it. Consequently, the same proposal can receive positive credit in one group and zero credit in another whose proposals refine it. If each minimal witness can be sampled, larger groups are more likely to contain all of them.

Proposition 4 (Group minimality becomes minimality). *Let W_K be the distinct successful proposals of a group of size K , and suppose $q(m) \geq \epsilon > 0$ for every $m \in \mathcal{M}(c)$, with $L = |\mathcal{M}(c)|$. Then $\min_{\subseteq} W_K = \mathcal{M}(c)$ with probability at least $1 - Le^{-K\epsilon}$. The proof is in Appendix B.*

The probability that a particular minimal witness is absent from a group of size K is at most $(1 - \epsilon)^K \leq e^{-K\epsilon}$. A union bound gives the stated probability for the whole antichain.

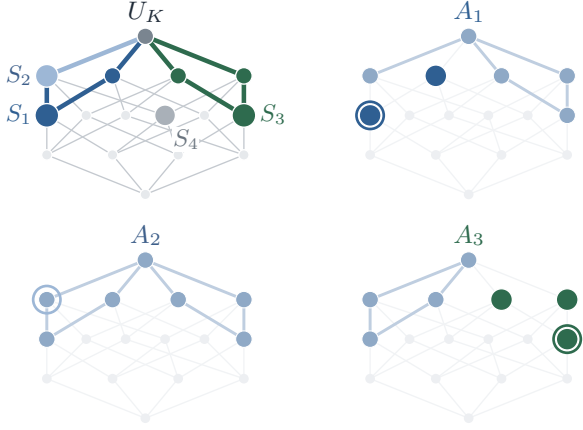


Figure 2: **Credit as a deletion marginal on the subset lattice** ($\mathcal{E} = \{a, b, c, d\}$; height is subset size). A success certifies its up-set, the set of its supersets, and the up-set of $S_2 = \{a, b, c\}$ is smaller than that of $S_1 = \{a, b\}$. U_K is the region the group certifies (S_1, S_3 minimal, S_2 redundant, S_4 rejected); in each A_i , grey denotes U_{-i} , the sets certified without S_i . The coloured region contains the sets whose certification is lost when S_i is removed; its measure gives the credit A_i .

Once all minimal witnesses are present, the group-minimal successes are the full antichain in (1), and every other success contains one of them and receives zero deletion credit (Proposition 3). After training, we sample the frozen policy under a deployment budget of B proposals, separate from K . The same bound with B in place of K gives the probability that the B proposals contain the complete antichain (Appendix B).

Computing the deletion credit has to account for overlap among the up-sets of distinct group-minimal successes. One way to do this is inclusion–exclusion, but it has a term for each combination of those successes and grows exponentially with their number. We avoid this expansion with a Monte Carlo estimator that computes every credit from one shared batch of sets drawn from μ . For each proposal, its work scales with the Monte Carlo sample count and the number of other group-minimal successes, without additional verifier calls (Appendix D). We test the accuracy of its credits and policy gradients in the ablations.

4 Experiments

Unless stated otherwise, we evaluate raw policies at fixed proposal budgets using the metrics of Section 2.

4.1 Prime implicant enumeration

We begin with monotone formulas in propositional logic, whose minimal witnesses can all be enumerated. Like a fixed maze used to test an RL method, each formula gives us a controlled instance with a known target: we can tell whether a policy found the whole family or stopped after one valid solution, and we can compare it with the value iteration planner. A monotone k -CNF instance induces the sufficiency

predicate $s(S) = [S \text{ satisfies every clause}]$, whose minimal witnesses are the prime implicants of the monotone function, equivalently the minimal hitting sets of the clauses (Berge 1989; Eiter and Gottlob 1995; Eiter, Gottlob, and Makino 2003). We take $n = 14$ with 16 to 19 minimal witnesses per instance (mean $|\mathcal{M}| = 18.1$) of sizes 2 to 4, and average over instances. The policy adds variables one at a time and chooses when to stop.

Under a success-only reward, policies are rewarded only for their probability of success. The policies trained with it produce redundant supersets. With a size penalty, the policy favors the smallest witness and recovers about one minimal witness per instance. GFlowNet (Bengio et al. 2021; Malkin et al. 2022) targets a law proportional to its reward and samples many sufficient sets, but many are non-minimal supersets, leaving most of the antichain uncovered. We add a size penalty to GFlowNet and to maximum-entropy (MaxEnt) RL (Haarnoja et al. 2018), and both targets stay proportional to local rewards; at the best penalty strength both recover about a quarter of the antichain, and most of their successful proposals remain non-minimal supersets (Appendix E.12). MWRL refines proposals along containment chains and recovers incomparable witnesses, covering most of the antichain (Table 2). We also vary antichain size, witness size, and overlap on planted instances (Appendix E.9).

Method	born-min. rate $\uparrow$	antichain recall $\uparrow$	#found $\uparrow$	raw $ S $
PPO / MaxRL	0.00 $\pm$ 0.00	0.00 $\pm$ 0.00	0.0 $\pm$ 0.0	12.0 $\pm$ 0.0
GRPO / RLOO	0.00 $\pm$ 0.00	0.00 $\pm$ 0.00	0.0 $\pm$ 0.0	12.0 $\pm$ 0.0
GFlowNet (success)	0.01 $\pm$ 0.00	0.02 $\pm$ 0.02	0.4 $\pm$ 0.3	7.8 $\pm$ 0.1
GFlowNet (up-set)	0.01 $\pm$ 0.00	0.05 $\pm$ 0.02	0.9 $\pm$ 0.3	7.0 $\pm$ 0.4
<i>with a size penalty</i>				
PPO	1.00 $\pm$ 0.00	0.07 $\pm$ 0.01	1.3 $\pm$ 0.1	2.8 $\pm$ 0.0
MaxEnt RL	0.12 $\pm$ 0.01	0.24 $\pm$ 0.02	4.3 $\pm$ 0.3	4.8 $\pm$ 0.1
GFlowNet	0.07 $\pm$ 0.00	0.24 $\pm$ 0.01	4.3 $\pm$ 0.2	5.0 $\pm$ 0.1
MWRL	0.76$\pm$0.01	0.67$\pm$0.01	12.1$\pm$0.3	4.0$\pm$0.1
<i>planning references (full information)</i>				
Scalar VI	1.00	0.06	1.0	2.8
Minimal-witness VI	1.00	1.00	18.1	4.1

Table 2: Prime-implicant recovery with ground truth. Columns report born-minimal rate, antichain recall, the number of distinct witnesses found, and raw proposal size. Scalar VI is value iteration on a size-penalized scalar reward and returns one witness. Minimal-witness VI plans on (F, b) and recovers the full antichain at $b = |\mathcal{M}|$ (Appendix E.4). Learned rows report mean $\pm$ standard deviation over three policy seeds after averaging the eight instances (training configuration in Appendix E.3). The size-penalized MaxEnt RL and GFlowNet rows report the best setting of a sweep over the penalty strength, trained with 57,600 verifier calls per formula.

4.2 Scientific variable identification

We train one policy across many Suzuki–Miyaura substrate pairs and test it on pairs withheld during training. Each pair is

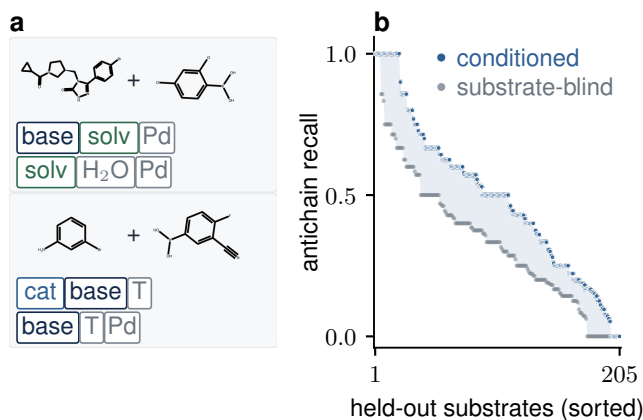


Figure 3: Amortized discovery of minimal reaction-condition sets on the Suzuki coupling. **(a)** Two held-out substrate pairs with their ground-truth antichains. Each row of chips names one minimal set of dimensions allowed to vary from the baseline (cat catalyst system, solv solvent, Pd loading, T temperature). **(b)** Recall on each held-out substrate within the window of 32 distinct condition sets, with the values of each policy sorted in decreasing order. The shaded gap between the conditioned and the substrate-blind policy is the recall that the substrate fingerprint adds.

a context, and the policy receives a fingerprint of the substrate pair as input.

The variables are 14 reaction-condition dimensions defined relative to a shared baseline protocol. For a substrate pair c , a subset S is sufficient when the benchmark contains a reaction that changes only dimensions in S and reaches the target yield. Conditions outside S remain at baseline. A minimal witness is therefore an inclusion-minimal set of dimensions that can be varied to achieve the target yield. One substrate pair can admit several incomparable minimal witnesses.

The verifier evaluates this predicate by closed-world lookup in the benchmark reaction table. It returns $s_c(S) = 1$ if and only if a tabulated reaction with deviation set contained in S reaches the yield threshold; unlisted assignments count as failures. MWRL receives only this verifier bit as the reward. In a wet-lab deployment, the same verifier can be implemented by Bayesian optimization restricted to S , with each query executing a reaction and measuring its yield.

We build the dataset by augmenting real literature reactions from a curated catalog (Cai et al. 2026) that supplies the substrate pairs, products, and condition vocabulary. However, the catalog does not contain complete condition panels. We construct the panels using mechanistic chemical rules calibrated on measured flow-chemistry screens (Perera et al. 2018) (Appendix F).

We show that the conditioned policy recovers minimal condition sets for substrate pairs it never saw during training. Its held-out recall is close to its seen recall and reaches a large fraction of the recall obtained by training a separate policy for each substrate (Table 3, Figure 3). Its born-minimal rate also exceeds that of these separately trained policies. Without

the substrate fingerprint, recall falls to about half their recall; a scalar reward does not recover any minimal witness.

Method	born-min. rate $\uparrow$	recall $\uparrow$		#found $\uparrow$
		seen	held-out	
Per-substrate	0.20	0.67	—	4.9
<i>Amortized MWRL</i>				
Conditioned	0.26 $\pm$ 0.02	0.50 $\pm$ 0.02	0.48$\pm$0.03	3.6$\pm$0.2
Substrate-blind	0.20 $\pm$ 0.01	0.37 $\pm$ 0.01	0.34 $\pm$ 0.02	2.6 $\pm$ 0.1
Scalar RL	0.00 $\pm$ 0.00	0.00 $\pm$ 0.00	0.00 $\pm$ 0.00	0.0 $\pm$ 0.0

Table 3: Amortized recovery of minimal reaction-condition sets within a window of 32 distinct condition sets per substrate. We report antichain recall on substrates seen in training and on held-out substrates, and the held-out column measures amortization. The column #found counts distinct minimal witnesses per held-out context. The two amortized rows train the same MWRL objective and differ only in access to the substrate fingerprint. Amortized rows give the mean $\pm$ standard deviation over three seeds that control both the substrate split and the policy. Per-substrate search retrains for each substrate to mark the ceiling, and nothing is held out from it.

4.3 LLM sparse circuit discovery

We search subsets of attention heads and MLP blocks in a frozen Qwen3 (Yang et al. 2025). To test a subset, we give every component outside it the activation it has on another prompt of the same task. We measure the KL divergence between the resulting next-token distribution and that of the unmodified model. The subset is accepted when this divergence stays below $1 - \alpha = 0.2$ times the corresponding divergence with every component replaced (Conmy et al. 2023). We use the raw verifier even though it can reject a superset of an accepted circuit (Appendix A.4).

We compare recovered circuits with circuit-discovery and pruning methods on MMLU under matched sparsity (Appendix G.1).

On Qwen3-1.7B (476 components), we evaluate the dense model on all 57 MMLU subjects (Hendrycks et al. 2021) and retain the ten on which the correct answer receives higher probability than a foil answer on at least 0.92 of the probes. We compare against top- k selection by EAP attribution (Syed, Rager, and Conmy 2024), weight magnitude, and the activation-aware Wanda score (Sun et al. 2024). We also include a per-subject HardConcrete mask trained through differentiable ablation (Louizos, Welling, and Kingma 2018), greedy ACDC-style faithfulness pruning (Conmy et al. 2023), and random- k . Every method uses the same subject-specific active-component budget, except ACDC, which uses its converged size. We discover circuits on one prompt split, then freeze them and evaluate them on questions drawn from disjoint dataset rows.

On the discovery prompts, only the verifier-guided searches find sufficient circuits at the matched budget. ACDC returns one circuit per subject, and MWRL returns a family

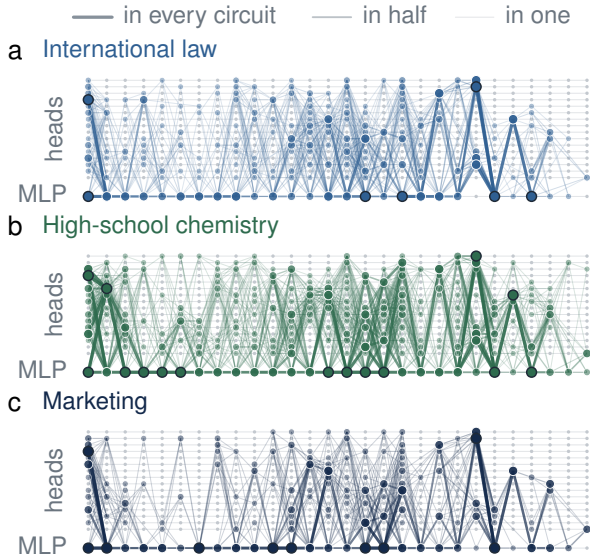


Figure 4: Complete recovered circuit families for three MMLU subjects on Qwen3-1.7B, shown on the module graph (28 layers $\times$ 16 heads, MLP rail below). The recovered families contain 14, 10 and 8 circuits, with median sizes 55, 67 and 49, respectively. Dark rings mark the modules present in every recovered circuit: 7 of the 189 modules used in (a), 15 of 230 in (b), and 10 of 136 in (c). The remaining modules vary across circuits.

of circuits with similar discovery retention (Figure 4). On held-out questions, we measure how much of the original behavior each circuit keeps with KL_{held} (Table 4). MWRL circuits keep substantially more of the original behavior than ranking-based masks.

4.4 Ablations

Credit estimator. The raw deletion credit is the change in the region a group certifies when one trajectory is removed. Using this non-negative credit directly (l1o) gives an unbiased update. The l2o estimator subtracts a baseline computed from the other trajectories, which gives signed credits without changing the expected gradient. Group-mean centering also gives signed credits, but its baseline depends on the trajectory being updated and leaves a finite-group bias (Figure 5).

Training group size and deployment budget. A larger training group is more likely to contain both a witness and a successful superset of it, allowing deletion credit to distinguish them. We hold total verifier calls fixed, which gives larger groups fewer updates. Recovery still rises with K (Figure 6a). After training, we freeze the policies and vary the deployment budget B . Small budgets favor concentrated policies, while larger budgets favor broader ones (Figure 6b).

Shared-sample Monte Carlo credit. Exact inclusion-exclusion is used when a group has at most twelve group-minimal successes. Beyond that point, all deletion credits and l2o baselines are computed from one shared batch of

Method	$KL_{\text{held}} \uparrow$	retention $\uparrow$	circuits $\uparrow$
Dense model	1.00	1.00	—
Random- k	-0.01 ± 0.01	0.18 ± 0.00	—
Magnitude	0.02 ± 0.00	0.21 ± 0.00	—
Wanda	-0.05 ± 0.00	0.20 ± 0.01	—
EAP	0.15 ± 0.01	0.20 ± 0.02	—
ACDC (greedy)	0.42 ± 0.01	0.74 ± 0.01	1
HardConcrete	0.02 ± 0.00	0.18 ± 0.01	1
MWRL	0.38 ± 0.02	0.74 ± 0.04	8.9 ± 1.3

Table 4: Circuit recovery on ten MMLU subjects with Qwen3-1.7B, every mask at the subject’s matched active-component budget (ACDC at its converged size). All columns evaluate the frozen circuit with every other component given its activation on another prompt. KL_{held} is one minus the circuit’s KL divergence from the unmodified model, divided by the divergence when every component is replaced, on questions disjoint from discovery (1 = the circuit behaves like the full model, 0 = no better than replacing every component); retention is discovery-prompt accuracy relative to the dense model; circuits is the number of distinct proposals per subject that are 1-minimal under the raw verifier. Means over ten subjects and three discovery seeds, $\pm$ the across-seed standard deviation.

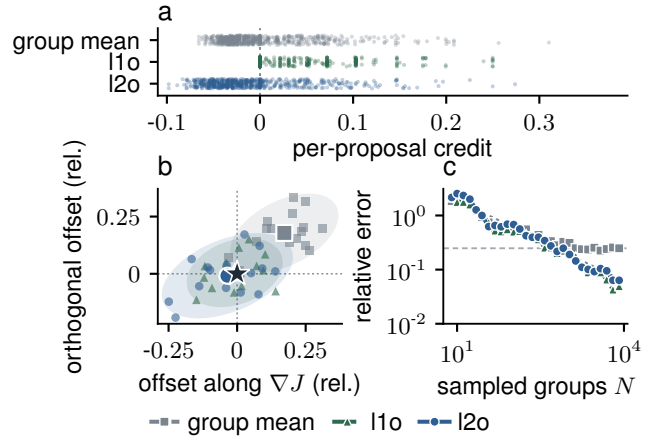


Figure 5: Credit estimators against the exact policy gradient on an enumerable MaxSAT instance with a frozen mid-training policy and $K = 6$; the reference is the exact gradient from full enumeration of the subset lattice. (a) Per-proposal credit over shared rollout groups: the l1o deletion credit is non-negative, while l2o and group-mean centering are signed. (b) Small points are 512-group estimates, large points show their mean across sixteen independent estimates, ellipses show 2σ dispersion, and the star is the exact gradient. (c) Running relative error $\|\bar{g}_N - \nabla J\| / \|\nabla J\|$; the dotted guide is $1/\sqrt{N}$ and the dashed line the group mean’s bias.

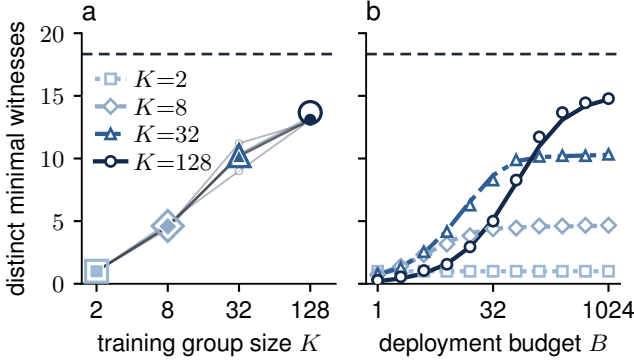


Figure 6: Training group size and deployment budget at 36,864 training verifier calls per instance; the dashed line is the antichain size $|\mathcal{M}|$. (a) Recovery at $B = 256$ proposals: grey paths are the three seeds after averaging six instances, filled markers their means, open rings empirical estimates. (b) Deployment sweep of the frozen policies: lines show the expected number of distinct minimal witnesses in B proposals, computed in closed form (Appendix B.5), and markers show the number found in the first B sampled proposals.

N sets sampled from the product measure. Computing the credits costs $O(NK)$ and does not require additional verifier calls.

We compare $N \in \{256, 1024, 5000, 20000\}$ with exact enumeration of all 2^{14} states. The constructed $K = 48$ group has 32 successful proposals, eight with positive raw credit, and includes duplicates, redundant successes, and failures. We average the credits over twenty independent batches. For the policy gradient, we freeze an on-policy model after 40 updates and compare estimated and exact directions across twelve sampled groups. From $N = 5,000$ on, the Monte Carlo estimate gives almost the same credits and policy gradient as exact enumeration (Table 5).

N	$\cos(\hat{A}, \tilde{A}) \uparrow$	relative ℓ_1 error $\downarrow$	support recall $\uparrow$	policy-grad. cosine $\uparrow$
256	0.866 $\pm$ 0.138	0.516 $\pm$ 0.176	0.594 $\pm$ 0.140	0.888 $\pm$ 0.163
1,024	0.970 $\pm$ 0.027	0.232 $\pm$ 0.069	0.850 $\pm$ 0.150	0.972 $\pm$ 0.020
5,000	0.995 $\pm$ 0.004	0.094 $\pm$ 0.033	1.000 $\pm$ 0.000	0.995 $\pm$ 0.004
20,000	0.998 $\pm$ 0.001	0.059 $\pm$ 0.017	1.000 $\pm$ 0.000	0.999 $\pm$ 0.001

Table 5: Shared-sample Monte Carlo fidelity against exact enumeration under the geometric measure with $p = 0.7$. Credit-vector metrics average 20 sample batches for a fixed $K = 48$ group; the policy-gradient cosine averages 12 on-policy groups. The first two columns use the baseline-adjusted $\hat{A}$, support recall uses raw A , and N counts auxiliary samples from μ , which do not spend verifier calls.

Base-measure choice. The product measure determines how strongly credit favors a smaller witness over a larger one. Increasing p weakens that size preference. We vary only p in the MaxSAT protocol (Figure 7). The born-minimal rate changes little across the tested range, while antichain recall

risks, because a larger p brings the witness values of large and small witnesses closer together, and the policy samples larger minimal witnesses more often. We fixed $p = 0.7$ before this sweep, near the value $1/\sqrt{2}$, which has equal logarithmic distances to strong refinement ($p = \frac{1}{2}$) and to no refinement preference ($p = 1$) (Appendix D).

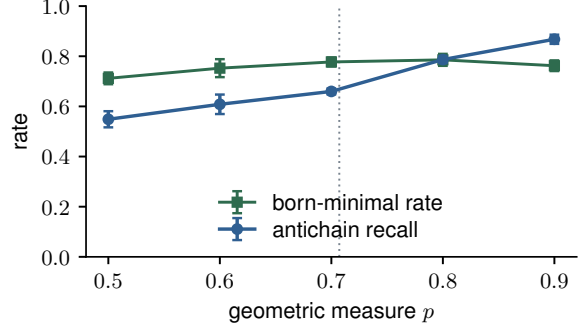


Figure 7: Base-measure sweep on the MaxSAT benchmark. Only the parameter p of the geometric measure varies; the dotted line marks $p = 1/\sqrt{2}$. We report means and across-seed standard deviations over three seeds and eight instances.

5 Conclusion

We formalized minimal-witness identification, the recovery of every minimal sufficient set of a task from binary verifier feedback, and introduced MWRL to solve it. MWRL values the region certified by a group of proposals and credits each proposal for the part of this region that would be lost without it. From this objective, we derived a planner that recovers every minimal witness on enumerable instances and a group policy gradient that scales to language models. MWRL recovers most minimal witnesses of each task, while methods that value each proposal in isolation return redundant supersets or a single witness. One trained policy also recovers minimal witnesses on tasks unseen in training. Wherever a verifier certifies an outcome, the family of minimal witnesses of that outcome becomes a learnable object.

References

- Ahmadian, A.; Cremer, C.; Gallé, M.; Fadaee, M.; Kreutzer, J.; Pietquin, O.; Üstün, A.; and Hooker, S. 2024. Back to Basics: Revisiting REINFORCE-Style Optimization for Learning from Human Feedback in LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 12248–12267. Association for Computational Linguistics.
- Barlow, R. E.; and Proschan, F. 1975. *Statistical Theory of Reliability and Life Testing: Probability Models*. Holt, Rinehart and Winston.
- Bellman, R. 1957. A Markovian Decision Process. *Indiana University Mathematics Journal*, 6(4): 679–684.
- Bengio, E.; Jain, M.; Korablyov, M.; Precup, D.; and Bengio, Y. 2021. Flow Network Based Generative Models for Non-Iterative Diverse Candidate Generation. In *Advances in*

- Neural Information Processing Systems*, volume 34, 27381–27394.
- Berge, C. 1989. *Hypergraphs: Combinatorics of Finite Sets*, volume 45 of *North-Holland Mathematical Library*. North-Holland.
- Beume, N.; Naujoks, B.; and Emmerich, M. 2007. SMS-EMOA: Multiobjective Selection Based on Dominated Hypervolume. *European Journal of Operational Research*, 181(3): 1653–1669.
- Bioch, J. C.; and Ibaraki, T. 1995. Complexity of Identification and Dualization of Positive Boolean Functions. *Information and Computation*, 123(1): 50–63.
- Buneman, P.; Khanna, S.; and Tan, W.-C. 2001. Why and Where: A Characterization of Data Provenance. In *Proceedings of the 8th International Conference on Database Theory*, 316–330. Springer.
- Cai, P.; Ye, Z.; Sun, S.; Zhang, L.; Ai, Z.; and Li, Y. 2026. ChemRANG: Building AI-ready datasets via mechanism-informed curation to learn reaction feasibility boundaries. *Chinese Chemical Letters*, 113222.
- Carter, B.; Mueller, J.; Jain, S.; and Gifford, D. 2019. What Made You Do This? Understanding Black-Box Decisions with Sufficient Input Subsets. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, 567–576. PMLR.
- Chen, Y.; and Mauch, L. 2024. Order-Preserving GFlowNets. In *International Conference on Learning Representations*.
- Conmy, A.; Mavor-Parker, A. N.; Lynch, A.; Heimersheim, S.; and Garriga-Alonso, A. 2023. Towards Automated Circuit Discovery for Mechanistic Interpretability. In *Advances in Neural Information Processing Systems*, volume 36.
- Crama, Y.; and Hammer, P. L. 2011. *Boolean Functions: Theory, Algorithms, and Applications*. Cambridge University Press.
- Davey, B. A.; and Priestley, H. A. 2002. *Introduction to Lattices and Order*. Cambridge University Press, 2 edition.
- De Santi, R.; Prajapat, M.; and Krause, A. 2024. Global Reinforcement Learning: Beyond Linear and Convex Rewards via Submodular Semi-Gradient Methods. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 10235–10266. PMLR.
- Eiter, T.; and Gottlob, G. 1995. Identifying the Minimal Transversals of a Hypergraph and Related Problems. *SIAM Journal on Computing*, 24(6): 1278–1304.
- Eiter, T.; Gottlob, G.; and Makino, K. 2003. New Results on Monotone Dualization and Generating Hypergraph Transversals. *SIAM Journal on Computing*, 32(2): 514–537.
- Fredman, M. L.; and Khachiyan, L. 1996. On the Complexity of Dualization of Monotone Disjunctive Normal Forms. *Journal of Algorithms*, 21(3): 618–628.
- Guerreiro, A. P.; Fonseca, C. M.; and Paquete, L. 2021. The Hypervolume Indicator: Computational Problems and Algorithms. *ACM Computing Surveys*, 54(6): 119:1–119:42.
- Haarnoja, T.; Zhou, A.; Abbeel, P.; and Levine, S. 2018. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 1861–1870.
- Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2021. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*.
- Ignatiev, A.; Narodytska, N.; and Marques-Silva, J. 2019. Abduction-Based Explanations for Machine Learning Models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(1): 1511–1519.
- Ijima, H.; and Yawata, K. 2026. Hypergraph Neural Networks Accelerate MUS Enumeration. arXiv:2604.09001.
- Janota, M.; and Marques-Silva, J. 2016. On the Query Complexity of Selecting Minimal Sets for Monotone Predicates. *Artificial Intelligence*, 233: 73–83.
- Junker, U. 2004. QuickXPlain: Preferred Explanations and Relaxations for Over-Constrained Problems. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence*, 167–172. AAAI Press.
- Koh, P. W.; and Liang, P. 2017. Understanding Black-Box Predictions via Influence Functions. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, 1885–1894. PMLR.
- Lehman, J.; and Stanley, K. O. 2011. Abandoning Objectives: Evolution Through the Search for Novelty Alone. *Evolutionary Computation*, 19(2): 189–223.
- Liffiton, M. H.; Previti, A.; Malik, A.; and Marques-Silva, J. 2016. Fast, Flexible MUS Enumeration. *Constraints*, 21(2): 223–250.
- Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; and Liang, P. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12: 157–173.
- Lochab, A.; Li, B.; and Zhang, R. 2026. Uniform-Correct Policy Optimization: Breaking RLVR’s Indifference to Diversity. arXiv:2605.00365.
- Louizos, C.; Welling, M.; and Kingma, D. P. 2018. Learning Sparse Neural Networks through L_0 Regularization. In *International Conference on Learning Representations*.
- Malkin, N.; Jain, M.; Bengio, E.; Sun, C.; and Bengio, Y. 2022. Trajectory Balance: Improved Credit Assignment in GFlowNets. In *Advances in Neural Information Processing Systems*, volume 35.
- Marques-Silva, J.; Janota, M.; and Belov, A. 2013. Minimal Sets over Monotone Predicates in Boolean Formulae. In *Computer Aided Verification*, 592–607. Springer.
- McCluskey, E. J. 1956. Minimization of Boolean Functions. *Bell System Technical Journal*, 35(6): 1417–1444.
- Méloux, M.; Portet, F.; Maniu, S.; and Peyrard, M. 2025. Everything, Everywhere, All at Once: Is Mechanistic Interpretability Identifiable? In *International Conference on Learning Representations*.

- Mnih, A.; and Rezende, D. J. 2016. Variational Inference for Monte Carlo Objectives. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, 2188–2196. PMLR.
- Mouret, J.-B.; and Clune, J. 2015. Illuminating Search Spaces by Mapping Elites. arXiv:1504.04909.
- Murakami, K.; and Uno, T. 2014. Efficient Algorithms for Dualizing Large-Scale Hypergraphs. *Discrete Applied Mathematics*, 170: 83–94.
- Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L. 1978. An Analysis of Approximations for Maximizing Submodular Set Functions—I. *Mathematical Programming*, 14(1): 265–294.
- Orth, J. D.; Thiele, I.; and Palsson, B. Ø. 2010. What Is Flux Balance Analysis? *Nature Biotechnology*, 28(3): 245–248.
- Pathak, D.; Agrawal, P.; Efros, A. A.; and Darrell, T. 2017. Curiosity-Driven Exploration by Self-Supervised Prediction. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, 2778–2787.
- Perera, D.; Tucker, J. W.; Brahmabhatt, S.; Helal, C. J.; Chong, A.; Farrell, W.; Richardson, P.; and Sach, N. W. 2018. A Platform for Automated Nanomole-Scale Reaction Screening and Micromole-Scale Synthesis in Flow. *Science*, 359(6374): 429–434.
- Prajapat, M.; Mutný, M.; Zeilinger, M. N.; and Krause, A. 2024. Submodular Reinforcement Learning. In *International Conference on Learning Representations*.
- Quine, W. V. 1952. The Problem of Simplifying Truth Functions. *The American Mathematical Monthly*, 59(8): 521–531.
- Reiter, R. 1987. A Theory of Diagnosis from First Principles. *Artificial Intelligence*, 32(1): 57–95.
- Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2018. Anchors: High-Precision Model-Agnostic Explanations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).
- Rothman, K. J. 1976. Causes. *American Journal of Epidemiology*, 104(6): 587–592.
- Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347.
- Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y. K.; Wu, Y.; and Guo, D. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300.
- Sheyner, O.; Haines, J.; Jha, S.; Lippmann, R.; and Wing, J. M. 2002. Automated Generation and Analysis of Attack Graphs. In *Proceedings of the 2002 IEEE Symposium on Security and Privacy*, 273–284. IEEE.
- Shih, A.; Choi, A.; and Darwiche, A. 2018. A Symbolic Approach to Explaining Bayesian Network Classifiers. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 5103–5111.
- Sun, M.; Liu, Z.; Bair, A.; and Kolter, J. Z. 2024. A Simple and Effective Pruning Approach for Large Language Models. In *International Conference on Learning Representations*.
- Sutton, R. S.; McAllester, D. A.; Singh, S. P.; and Mansour, Y. 1999. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In *Advances in Neural Information Processing Systems*, volume 12, 1057–1063.
- Syed, A.; Rager, C.; and Conmy, A. 2024. Attribution Patching Outperforms Automated Circuit Discovery. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, 407–416. Association for Computational Linguistics.
- Tajwar, F.; Zeng, G.; Zhou, Y.; Song, Y.; Arora, D.; Jiang, Y.; Schneider, J.; Salakhutdinov, R.; Feng, H.; and Zanette, A. 2026. Maximum Likelihood Reinforcement Learning. In *Proceedings of the 43rd International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR.
- Takahashi, K.; and Yamanaka, S. 2006. Induction of Pluripotent Stem Cells from Mouse Embryonic and Adult Fibroblast Cultures by Defined Factors. *Cell*, 126(4): 663–676.
- Van Moffaert, K.; Drugan, M. M.; and Nowé, A. 2013. Hypervolume-Based Multi-Objective Reinforcement Learning. In *Evolutionary Multi-Criterion Optimization*, 352–366.
- Walder, C.; and Karkhanis, D. 2025. Pass@K Policy Optimization: Solving Harder Reinforcement Learning Problems. arXiv:2505.15201.
- Wang, K. R.; Variengien, A.; Conmy, A.; Shlegeris, B.; and Steinhardt, J. 2023. Interpretability in the Wild: A Circuit for Indirect Object Identification in GPT-2 Small. In *International Conference on Learning Representations*.
- Williams, R. J. 1992. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning*, 8(3–4): 229–256.
- Yang, A.; et al. 2025. Qwen3 Technical Report. arXiv:2505.09388.
- Zeller, A.; and Hildebrandt, R. 2002. Simplifying and Isolating Failure-Inducing Input. *IEEE Transactions on Software Engineering*, 28(2): 183–200.
- Zhang, X.; Lin, X.; Xue, B.; Chen, Y.; and Zhang, Q. 2023. Hypervolume Maximization: A Geometric View of Pareto Set Learning. In *Advances in Neural Information Processing Systems*, volume 36.

A Foundations and verifier scope

A.1 Up-set representation

Under monotonicity, a successful proposal certifies all its supersets and a failed proposal rules out all its subsets: $s_c(S) = 1 \Rightarrow \uparrow S \subseteq F(c)$ and $s_c(S) = 0 \Rightarrow \downarrow S \cap F(c) = \emptyset$, where $\downarrow S = \{T : T \subseteq S\}$. A group’s successes thus certify $U_K \subseteq F(c)$.

Theorem 2 (Representation). Let g be a symmetric group value on $(S_i, y_i)_{i \leq K}$ that vanishes on the empty group, depends only on the success multiset, and is unchanged by deleting a duplicate or redundant success. Then $g = \mathcal{R}(U(W))$ for some set function $\mathcal{R}$ on up-sets with $\mathcal{R}(\emptyset) = 0$, and g is non-decreasing in added successes iff $\mathcal{R}$ is monotone.

Proof. Dependence on the success multiset gives $g = g(W)$. Deleting repeats and redundant elements leaves $g(W) = g(\mathcal{A}(W))$, where $\mathcal{A}(W) = \min_{\subseteq}(\text{supp } W)$ and $\text{supp } W$ is the set of distinct elements of W . Antichains and up-sets are in bijection: $\mathcal{A} \mapsto \uparrow \mathcal{A}$ and $U \mapsto \min_{\subseteq} U$ are mutually inverse. Hence $\mathcal{R}(U) := g(\min_{\subseteq} U)$ is well defined and $g = \mathcal{R}(U_K)$. Adding successes enlarges U_K , so monotonicity of the group value is equivalent to monotonicity of $\mathcal{R}$. $\square$

Thus U_K determines the group return, and $J_K = \mathbb{E}[\mathcal{R}(U_K)]$.

A.2 Group returns and verifier access

The coverage measure $\mathcal{R}_\mu(U) = \mu(U)$ values the certified region directly and is the measure used throughout the paper. The counting measure first reduces each success to a minimal witness with a canonicalizer and then assigns unit mass to every distinct canonical witness. Its canonicalizer uses additional verifier calls (Appendix B).

A.3 Proposal-separable objectives

A symmetric group return is proposal-separable if, for each K , there are a constant $b_K(c)$ and a per-proposal reward $r_c : 2^\mathcal{E} \times \{0, 1\} \rightarrow \mathbb{R}$ such that

$$G(S_{1:K}, y_{1:K}) = b_K(c) + \sum_{i=1}^K r_c(S_i, y_i).$$

The reward depends on the complete proposal and its own verifier bit, but not on the other proposals. The constant shifts every comparison below equally, so we set $b_K(c) = 0$.

A sampler can also target the law $q \propto r(\cdot, s_c(\cdot))$ for a local reward $r \geq 0$, where a reward is local when it is the same function of $(S, s_c(S))$ for every monotone predicate. An objective can also depend on q only through the success probability $q(F(c)) = \Pr_{S \sim q}[s_c(S) = 1]$. The theorem covers both rules together with the separable objective.

Theorem 1 (Relational necessity). Let a method target the maximizers of the separable objective $J_{\text{sep}}(q) = \mathbb{E}_{S_{1:K} \sim q^{\otimes K}}[\sum_i r(S_i, s_c(S_i))]$, the law $q \propto r(\cdot, s_c(\cdot))$ for a local reward $r \geq 0$, or the maximizers of an objective that depends on q only through the success probability $\Pr_{S \sim q}[s_c(S) = 1]$. Then, for $|\mathcal{E}| \geq 3$, some monotone predicate with $|\mathcal{M}(c)| \geq 2$ has a target law whose support differs from $\mathcal{M}(c)$.

Proof. Each rule is presumed to define a target for every monotone predicate: the objective in the third rule attains its maximum, and $r(\cdot, s_c(\cdot))$ in the second has a positive total under every monotone predicate s .

Separable objective. Write $\rho(S) = r(S, s_c(S))$. Independence gives

$$J_{\text{sep}}(q) = \sum_{i=1}^K \mathbb{E}_{S_i \sim q} \rho(S_i) = K \sum_{S \subseteq \mathcal{E}} q(S) \rho(S),$$

linear in q . Set $\rho^* = \max_S \rho(S)$ and $Q^* = \arg \max_S \rho(S)$. Then

$$J_{\text{sep}}(q) = K\rho^* - K \sum_S q(S) (\rho^* - \rho(S)) \leq K\rho^*,$$

with equality iff $q(Q^*) = 1$. Hence the maximizers are exactly the laws supported on Q^* , and every point mass on Q^* is optimal. For $|\mathcal{M}(c)| \geq 2$ such a point mass misses some member of $\mathcal{M}(c)$. If $\rho(A) > \rho(B)$ for $A, B \in \mathcal{M}(c)$, then $B \notin Q^*$ and every maximizer gives B mass zero. This case holds for any reward, local or not.

Proportional law. Take distinct $a, b, d \in \mathcal{E}$ and the monotone predicates s_1 and s_2 with antichains $\{\{a, b\}, \{d\}\}$ and $\{\{a\}, \{d\}\}$. The set $X = \{a, b\}$ satisfies $s_1(X) = s_2(X) = 1$, and locality gives it the weight $r(X, 1)$ under both predicates. If the target law under s_1 has a support different from its antichain, s_1 is the required predicate. Otherwise X lies in that support and $r(X, 1) > 0$. The target law under s_2 then gives positive mass to X , which contains the witness $\{a\}$ and is not minimal under s_2 .

Success probability. Let q^* maximize the objective and set $t = q^*(F(c))$. When $|\mathcal{M}(c)| \geq 2$, the empty set lies outside $F(c)$, since $\emptyset \in F(c)$ would make $\{\emptyset\}$ the whole antichain. The ground set $\mathcal{E}$ lies in the nonempty up-set $F(c)$ and strictly contains every minimal witness, and it is therefore not minimal. The law $t \delta_{\mathcal{E}} + (1-t) \delta_{\emptyset}$ gives $F(c)$ probability t , attains the objective value of q^* , and is supported inside $\{\emptyset, \mathcal{E}\}$, which contains no minimal witness. $\square$

The statement transfers to the estimators built on separable objectives. An admissible baseline leaves the expected policy gradient unchanged (Appendix C.2), and group normalization rescales each group's contribution by a positive statistic; both are used for variance control in the gradient estimator of J_{sep} . The theorem concerns the objective being estimated. Best-of- K aggregation replaces the sum by $\max_i \rho(S_i)$. Since $\mathbb{E}_q[\max_i \rho(S_i)] \leq \rho^*$, with equality at a point mass on Q^* , that point mass remains optimal, and with a binary success reward this objective equals $1 - (1 - q(F(c)))^K$, indifferent among all laws supported inside $F(c)$. This objective depends on q only through $q(F(c))$ and falls under the third rule of the theorem, with maximum likelihood RL, which maximizes $\log q(F(c)) = -\sum_{k \geq 1} (1 - q(F(c)))^k / k$. Maximum-entropy RL with reward r_0 and temperature α targets $q(S) \propto n(S) e^{r_0(S, s_c(S))/\alpha}$, where $n(S)$ counts the construction paths that end at S , and a GFlowNet with reward R targets $q \propto R$. Both weights are local rewards, and both samplers fall under the second rule.

The preceding theorem concerns the optima of separable objectives. The next theorem shows that the coverage return itself cannot be written as a sum of per-proposal rewards.

Theorem 3 (Proposal-separable inexpressibility). Let $K \geq 2$, let μ have full support on $2^\mathcal{E}$, and suppose that the accepting set contains a strict chain $A \subsetneq B$. Then the coverage return

$$G_\mu(S_{1:K}, y_{1:K}) = \mu \left(\bigcup_{i: y_i=1} \uparrow S_i \right)$$

is not proposal-separable.

Proof. Write $w_A = \mu(\uparrow A)$ and $w_B = \mu(\uparrow B)$. Full support and $A \subsetneq B$ give $w_A > w_B$. A group of K copies of A and a group of $K - 1$ copies of A plus one copy of B both have coverage w_A . Separability would force $r_c(A, 1) = r_c(B, 1)$.

A group of K copies of B would then have the same return as a group of K copies of A , contradicting $w_B < w_A$. $\square$

The inexpressibility theorem rules out a proposal-separable form of the coverage return. The coverage return is a scalar, but its value depends on overlap and containment among the proposals. Its marginal credit therefore depends on the relations among the proposals of a group.

A.4 Nonmonotone verifiers and existential closure

Without monotonicity, acceptance shows that a proposal itself is sufficient, but it does not certify the supersets of the proposal. The coverage return still compares containment and overlap among accepted proposals, although the resulting union need not lie within $F(c)$. A weaker property can still be checked. An accepted set is *1-minimal* when the verifier rejects it after the removal of any single element. Under a monotone predicate, a 1-minimal set is minimal, and without monotonicity the two properties can differ.

The existential closure would restore certification of all supersets and the support characterization of Proposition 3. The coverage credit depends only on the raw verifier bit and the raw terminal subset. The language-model experiment therefore trains on the raw predicate and does not apply a canonicalizer during training. We approximate the existential closure only in the audit below. We define the *finite closure* of a rejected set as the result of the circuit canonicalizer Φ (Appendix G) applied to it, which searches for an accepted subset. The reported born-minimal rates are 1-minimality under the raw predicate.

Measured violation and repair. For each behavior, we obtain verified circuits by applying Φ to the full component set and to random dense subsets. We add random components to each circuit and query the resulting supersets; a monotone predicate would accept them all. We report raw rejection, finite-closure repair, and residual rates in Table 6 and plot rejection and faithfulness against the number of components added (Figure 8). On 1.7B, the rejected supersets fall further below the faithfulness threshold as components are added. On 8B, rejections become less common and the rejected supersets remain close to the 0.8 threshold. Since the raw predicate can reject a superset of an accepted circuit, pointwise acceptance does not certify an entire up-set. The reported residual rate bounds the violations that remain after this finite closure search.

Quality of the returned circuits. Every circuit returned by the finite closure was 1-minimal under the raw predicate. The unrepaired violations are systematic misses of the attribution-guided search, and searching the same violating supersets again does not recover them. The canonicalizer is also not constant on an up-set. Among repaired supersets, Φ returns the circuit used to construct the superset in about half of the cases and a different 1-minimal circuit in the others. The counting measure deduplicates proposals by $\Phi(S)$, and a change in the returned circuit changes which proposals are counted as distinct. The coverage measure uses only the verifier bit and the raw terminal, and this instability of Φ does not affect its credit.

	Bench10	8B family	8B leaf
Probes	630	546	2,688
Raw violation $\downarrow$	0.83	0.56	0.56
Closure repair $\uparrow$	0.70	0.73	0.68
Residual $\downarrow$	0.25	0.15	0.18
1-minimal	15/15	13/13	64/64

Table 6: Non-monotonicity at $\alpha = 0.8$. Bench10 uses Qwen3-1.7B; the other two verifiers use Qwen3-8B. Raw violation is the fraction of sampled supersets rejected by the raw verifier; closure repair is the fraction of these violations in which the finite closure finds an accepted subset. Residual bounds the violations left after the finite closure. 1-minimal counts the returned circuits that are 1-minimal.

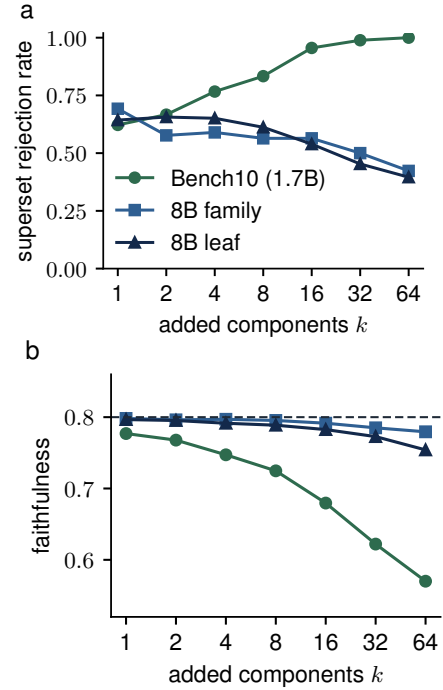


Figure 8: Violation of monotonicity in the circuit verifier. (a) Fraction of random supersets of a verified circuit rejected by the raw predicate, against the number of added components. A monotone predicate would reject none. (b) Mean faithfulness of the rejected supersets; the dashed line is the acceptance threshold $\alpha = 0.8$.

B Planning and finite-budget recovery

B.1 Minimal-witness value iteration

Throughout this section, set $\mathcal{R} = \mathcal{R}_\mu$ for a full-support measure μ . Fix a context c and write $F(c) = \uparrow \mathcal{M}(c)$. A state is a certified region $F \subseteq F(c)$ together with a remaining proposal budget b . A completed trajectory commits a successful subset S , moves the state to $F \cup \uparrow S$, and receives $\Delta_{\mathcal{R}_\mu}(S | F) = \mu(F \cup \uparrow S) - \mu(F)$. This recursion is ordinary finite-horizon value iteration on the augmented state (F, b) , and for $b < |\mathcal{M}(c)|$ it computes the optimal cover

under that budget.

Theorem 4 (Recovery by minimal-witness value iteration). *Let s_c be monotone and let μ have full support on $2^{\mathcal{E}}$. Initialize $F_0 = \emptyset$ and apply the minimal-witness update (5) until its maximum marginal is zero. Every selected proposal is a previously unselected member of $\mathcal{M}(c)$. The planner stops after $|\mathcal{M}(c)|$ steps and satisfies*

$$\{S_1, \dots, S_{|\mathcal{M}(c)|}\} = \mathcal{M}(c), \quad F_{|\mathcal{M}(c)|} = F(c).$$

Proof. Assume the selections so far are distinct members of $\mathcal{M}(c)$. For unselected $m \in \mathcal{M}(c)$, the point m lies in no other minimal witness’s up-set, so $\Delta(m \mid F) \geq \mu(\{m\}) > 0$. Let S be a non-minimal success, so $m \subsetneq S$ for some $m \in \mathcal{M}(c)$. If m is selected, $\uparrow S \subseteq \uparrow m \subseteq F$ and $\Delta(S \mid F) = 0$; if not, $\uparrow S \setminus F \subsetneq \uparrow m \setminus F$ with m in the difference, so $\Delta(S \mid F) < \Delta(m \mid F)$. The maximizer therefore selects a new member of $\mathcal{M}(c)$, and by induction it selects one at every step. After $|\mathcal{M}(c)|$ steps the selected up-sets have union $F(c)$ and every marginal is zero. $\square$

On the MaxSAT lattices, the executed recursion on (F, b) supplies the planning reference in Table 2; Appendix E.4 documents the recursion and its validation.

B.2 Finite-group objective identities

Let q_θ be the terminal proposal distribution. Define $\rho_\theta(T) = \Pr_{S \sim q_\theta}[s_c(S) = 1, S \subseteq T]$ and $p_m = \Pr_{S \sim q_\theta}[s_c(S) = 1, \Phi(S) = m]$. Independence of the K proposals gives

$$J_K^{\text{cov}} = \mathbb{E}_{T \sim \mu}[1 - (1 - \rho_\theta(T))^K], \quad (7)$$

$$J_K^{\text{cnt}} = \sum_{m \in \mathcal{M}(c)} [1 - (1 - p_m)^K]. \quad (8)$$

The coverage objective averages the probability that T is certified over $T \sim \mu$. The counting objective sums the probabilities of observing each canonical witness. For groups of more than one proposal, the derivative with respect to a certification or witness probability is larger when that probability is smaller. A minimal witness with zero probability under the policy is never sampled and receives zero credit in a score-function update. Exploration therefore controls which parts of the antichain become reachable during learning.

B.3 The counting measure

The counting measure uses a canonicalizer Φ to reduce each successful proposal to a minimal witness. Calls made by Φ count toward the verifier budget, so the canonicalizer must be cheap and reliable for this variant to be practical. With successes $W = \{S_j : y_j = 1\}$, write $\Phi(W) = \{\Phi(S) : S \in W\}$ for the distinct canonical witnesses of the group. Every canonical witness is a minimal witness, so these witnesses form an antichain and are the minimal elements of the up-set $U(\Phi(W))$ they certify. The counting measure assigns unit mass to each of them,

$$\mathcal{R}_\#(U(\Phi(W))) = |\Phi(W)|. \quad (9)$$

This return is a function of the region certified by the canonical witnesses. The raw region $U(W)$ does not determine it,

Measure	Interface	minimal-output rate $\uparrow$	antichain recall $\uparrow$
Coverage $\mathcal{R}_\mu$	$S, s_c(S), \mu$	0.758 $\pm$ 0.010	0.667 $\pm$ 0.016
Counting	$S, s_c(S), \Phi(S)$	1.000 $\pm$ 0.000	0.993 $\pm$ 0.007

Table 7: Coverage and counting measures. The interface column lists each measure’s inputs. Minimal-output rate is the fraction of delivered witnesses that are minimal: the born-minimal rate for coverage, one by construction for counting. Mean $\pm$ standard deviation over three seeds and eight MaxSAT instances.

because Φ need not be constant on an up-set (Appendix A.4). With $W_{-i} = \{S_j : j \neq i, y_j = 1\}$, the deletion credit is the 0/1 signal

$$\begin{aligned} A_i &= \mathcal{R}_\#(U(\Phi(W))) - \mathcal{R}_\#(U(\Phi(W_{-i}))) \\ &= y_i \mathbf{1}[\Phi(S_i) \neq \Phi(S_j) \ \forall j \neq i : y_j = 1], \end{aligned} \quad (10)$$

awarding unit credit exactly when no other successful proposal is mapped to the same canonical witness. If another proposal has the same canonical witness, removing either proposal leaves the count unchanged and its deletion credit is zero. This is the leave-one-out marginal of the counting measure; the leave-two-out baseline of Appendix C applies to the coverage measure only.

We compare the two measures on the eight MaxSAT formulas (Table 7). Coverage uses only the verifier bit and must learn minimality, reaching a 0.76 born-minimal rate and two thirds of the antichain; the counting canonicalizer forces every output minimal and nearly saturates recall, at the cost of the extra verifier calls made by Φ .

B.4 Finite-budget optimum of the counting measure

Theorem 5 (Finite-budget optimum of the counting measure). *Fix $L = |\mathcal{M}(c)| \geq 1$. Assume that the terminal policy class can realize every categorical distribution on $\mathcal{M}(c)$, with any remaining probability assigned to failure. For $K \geq 2$, (8) has the unique maximizer*

$$p_m^* = \frac{1}{L} \quad (m \in \mathcal{M}(c)), \quad p_\perp^* = 0,$$

and

$$\max_p J_K^{\text{cnt}} = L \left[1 - \left(1 - \frac{1}{L} \right)^K \right].$$

For $K = 1$, every failure-free distribution is optimal.

Proof. The function $f_K(p) = 1 - (1 - p)^K$ is increasing, so failure mass cannot occur at an optimum. For $K \geq 2$ it is strictly concave on the relevant simplex when $L > 1$. Jensen’s inequality under $\sum_m p_m = 1$ gives $L^{-1} \sum_m f_K(p_m) \leq f_K(1/L)$, with equality only when all probabilities are equal. The case $L = 1$ is immediate. For $K = 1$ the objective is $\sum_m p_m = 1$ for every failure-free distribution. $\square$

The uniform optimum has expected antichain recall $1 - (1 - 1/L)^K$, and a group of $K < L$ proposals cannot contain the full antichain.

B.5 Deployment-budget recovery

Evaluating Equation (8) at deployment budget B gives the expected number of distinct minimal witnesses in B independent proposals, $\sum_m (1 - (1 - p_m)^B)$, where p_m is the probability of terminating at witness m . On the enumerable MaxSAT lattice, a forward pass over the subset states computes every p_m in closed form. We compare the resulting deployment curves with the number of distinct minimal witnesses found in the first B sampled proposals (Figure 6). As $B \rightarrow \infty$, each curve saturates at the number of minimal witnesses to which the policy gives positive probability, at most $|\mathcal{M}|$.

B.6 Asymptotic recovery under full support

Theorem 6 (Asymptotic recovery under full support). *For a fixed terminal law q , let $\mathcal{A}_q = \{S \in F(c) : q(S) > 0\}$ and $L = |\mathcal{M}(c)|$. Then*

$$\lim_{K \rightarrow \infty} J_K^{\text{cov}}(q) = \mu(\uparrow \mathcal{A}_q).$$

If μ has full support, this limit equals $\mu(F(c))$ if and only if $q(m) > 0$ for every $m \in \mathcal{M}(c)$. If $\epsilon = \min_{m \in \mathcal{M}(c)} q(m) > 0$, then

$$\begin{aligned} \mu(F(c)) - J_K^{\text{cov}}(q) &\leq \mu(F(c))(1 - \epsilon)^K, \\ \Pr[U_K = F(c)] &\geq 1 - Le^{-K\epsilon}. \end{aligned}$$

Proof. The integrand in (7) tends to one iff $\rho_q(T) > 0$, i.e. on $\uparrow \mathcal{A}_q$; the lattice is finite, so termwise convergence gives the limit. Under full support the limit equals $\mu(F(c))$ iff every $m \in \mathcal{M}(c)$ is covered, and a successful subset of m equals m by minimality, so this holds iff every minimal witness has positive mass. With mass at least ϵ , each $T \in F(c)$ is missed with probability at most $(1 - \epsilon)^K$; summing over T gives the value bound, and a union bound over $\mathcal{M}(c)$ gives the recovery bound. $\square$

The same sampling bound shows when the group-minimal successes recover the full antichain.

Proposition 4 (Group minimality becomes minimality).

Let W_K be the distinct successful proposals of a group of size K , and suppose $q(m) \geq \epsilon > 0$ for every $m \in \mathcal{M}(c)$, with $L = |\mathcal{M}(c)|$. Then $\min_{\subseteq} W_K = \mathcal{M}(c)$ with probability at least $1 - Le^{-K\epsilon}$.

Proof. Each $m \in \mathcal{M}(c)$ is missing from the group with probability at most $(1 - \epsilon)^K \leq e^{-K\epsilon}$, and a union bound over $\mathcal{M}(c)$ puts all of $\mathcal{M}(c)$ inside W_K with probability at least $1 - Le^{-K\epsilon}$. Work on that event. Every $S \in W_K$ satisfies $s_c(S) = 1$ and therefore contains some $m \in \mathcal{M}(c)$. A member of $\mathcal{M}(c)$ has no proper successful subset, which makes it minimal in W_K . Any other $S \in W_K$ strictly contains such an m , which is present, and is excluded from $\min_{\subseteq} W_K$. The two directions give $\min_{\subseteq} W_K = \mathcal{M}(c)$. $\square$

When the group contains every minimal witness, its group-minimal successes are $\mathcal{M}(c)$; every other success contains one of them and receives zero deletion credit (Proposition 3). Increasing group size raises the probability of this event.

C Group policy gradients and control variates

We derive the l1o credit and the l2o baseline from the grouped objective.

C.1 Grouped objective and REINFORCE

Admissible proposal processes. The derivations of this section hold for any episodic process that proposes subsets, under the following conditions. (i) *Termination*: every episode ends within a finite horizon, so the trajectory space at a context is finite and gradients pass through the sums below. (ii) *Terminal interface*: a map ψ sends each completed trajectory τ to a proposal $S = \psi(\tau) \subseteq \mathcal{E}$, the verifier is evaluated once per episode on $\psi(\tau)$ alone, and the group return depends on trajectory i only through $S_i = \psi(\tau_i)$ and $y_i = s_c(S_i)$. (iii) *Differentiable factorization*: $P_\theta(\tau | c) = \rho(\tau) \prod_t \pi_\theta(a_t | x_t)$, where ρ gathers the θ -independent initial and transition factors and every decision factor $\pi_\theta(a_t | x_t)$ is positive and differentiable at the realized actions. (iv) *Independent groups*: the K episodes of a group are drawn i.i.d. from $P_\theta(\cdot | c)$. Under (i) and (ii) the objective is a functional of the terminal law $q_\theta(S) = \Pr_\theta[\psi(\tau) = S]$ alone; the internal states, the move set, and the path of an episode never enter the return. For a group of K trajectories drawn i.i.d. at context c , write

$$U_K = \bigcup_{i: y_i=1} \uparrow S_i, \quad \mathcal{R}(\emptyset) = 0, \quad \mathcal{R} \text{ monotone.}$$

This is the group state of (2).

Reachability and expressiveness. Two further conditions tie a proposal process to the recovery results of the preceding appendix, and they constrain the move set only through the terminal laws it induces. First, every minimal witness must be a possible proposal, $\mathcal{M}(c) \subseteq \bigcup_\theta \text{supp } q_\theta$: by identity (8), a witness outside every support has $p_m = 0$ and caps expected recovery at the reachable count. Second, the closure of the realizable laws $\{q_\theta\}_\theta$ must contain every categorical distribution on $\mathcal{M}(c)$, which is the realizability hypothesis of the finite-budget optimum theorem and makes the full-support regime of asymptotic recovery attainable. MaxSAT episodes add elements starting from $\emptyset$; circuit episodes start from the full set and remove components, with add actions for backtracking. Both satisfy the conditions with a horizon long enough to reach every witness. A policy that emits the elements of its proposal in a fixed linear order generates each subset along exactly one path, so the chain rule over the prefix tree realizes every law on the reachable subsets, in particular every law on $\mathcal{M}(c)$; redundant moves add paths without shrinking this family. Factorized mask emission and autoregressive emission by a sequence model satisfy conditions (i) to (iv) whenever the output parses to a subset. The two reachability conditions above are separate requirements: every minimal witness must receive positive probability, and the realizable laws must include every categorical distribution on $\mathcal{M}(c)$.

Score-function gradient. Fix c . The trajectories are i.i.d., so

$$J_K(\theta) = \sum_{\tau_{1:K}} \left(\prod_{i=1}^K P_\theta(\tau_i) \right) \mathcal{R}(U_K).$$

Apply $\nabla_\theta p = p \nabla_\theta \log p$ to the product:

$$\begin{aligned} \nabla_\theta J_K &= \sum_{\tau_{1:K}} \left(\prod_i P_\theta(\tau_i) \right) \mathcal{R}(U_K) \sum_{i=1}^K \nabla_\theta \log P_\theta(\tau_i) \\ &= \mathbb{E} \left[\mathcal{R}(U_K) \sum_{i=1}^K \nabla_\theta \log P_\theta(\tau_i) \right]. \end{aligned} \quad (11)$$

Since ρ is independent of θ , $\nabla_\theta \log P_\theta(\tau) = \sum_t \nabla_\theta \log \pi_\theta(a_t | x_t)$; the gradient does not pass through the verifier. Equation (11) is group REINFORCE: unbiased, but every sample is weighted by the same whole-group return.

C.2 Admissible baselines and leave-one-out credit

Admissible baseline. Subtract from sample i a control variate b_i that does not depend on its own trajectory τ_i (it may depend on c and on the others τ_{-i}). Conditioning on τ_{-i} and summing over τ_i ,

$$\begin{aligned} \mathbb{E}[b_i \nabla_\theta \log P_\theta(\tau_i)] &= \mathbb{E}_{\tau_{-i}} \left[b_i \sum_{\tau_i} P_\theta(\tau_i) \nabla_\theta \log P_\theta(\tau_i) \right] \\ &= \mathbb{E}_{\tau_{-i}} \left[b_i \nabla_\theta \sum_{\tau_i} P_\theta(\tau_i) \right] \\ &= \mathbb{E}_{\tau_{-i}} [b_i \nabla_\theta 1] = 0, \end{aligned} \quad (12)$$

the middle step being $p \nabla \log p = \nabla p$ once more and $\sum_{\tau_i} P_\theta(\tau_i) = 1$. Subtracting $\sum_i b_i \nabla_\theta \log P_\theta(\tau_i)$ from (11) thus changes no expectation, for any such $\{b_i\}$.

The leave-one-out credit l1o. The baseline matched to the group objective is its value with sample i withheld, $b_i = \mathcal{R}(U_{-i})$, where $U_{-i} = \bigcup_{j \neq i, y_j=1} \uparrow S_j$. It uses only τ_{-i} and is admissible by (12). Substituting into (11) gives

$$\begin{aligned} \nabla_\theta J_K &= \mathbb{E} \left[\sum_{i=1}^K A_i \sum_t \nabla_\theta \log \pi_\theta(a_{i,t} | x_{i,t}) \right], \\ A_i &= \mathcal{R}(U_K) - \mathcal{R}(U_{-i}), \end{aligned} \quad (13)$$

which is the deletion credit (6).

C.3 Leave-two-out baseline

The l1o credit (13) is non-negative. A second admissible baseline lets it take negative values. For $j \neq i$ let $U_{-i,-j} = \bigcup_{k \neq i,j, y_k=1} \uparrow S_k$ and define the leave-two-out baseline

$$b_i = \mathcal{R}(U_{-i}) - \frac{1}{K-1} \sum_{j \neq i} \mathcal{R}(U_{-i,-j}),$$

the average marginal contribution inside the group that already excludes i . The baseline-adjusted credit is $\tilde{A}_i = A_i - b_i$.

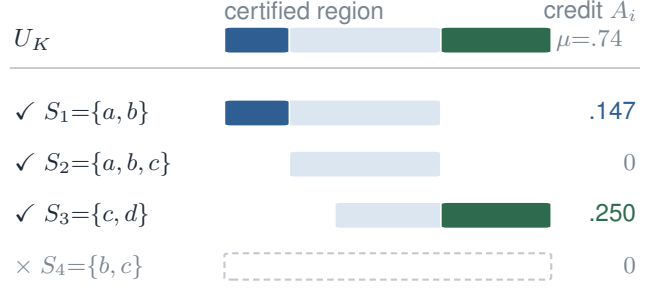


Figure 9: Deletion credit in the example of Figure 2, with minimal witnesses $\{a, b\}$ and $\{c, d\}$. Bar widths represent μ -measure. The top bar shows U_K , partitioned by overlap among successful up-sets. Each bar below shows a proposal's up-set; the solid portion contains the sets certified only by that proposal and has measure A_i . Here $S_1 = \{a, b\}$ is minimal and $S_3 = \{c, d\}$ is a second minimal witness, while the redundant $S_2 = \{a, b, c\}$ and the rejected $S_4 = \{b, c\}$ have zero credit, matching (14).

Both $\mathcal{R}(U_{-i})$ and every $\mathcal{R}(U_{-i,-j})$ depend only on $\{\tau_k\}_{k \neq i}$, so b_i is independent of τ_i . By (12), $\mathbb{E}[\tilde{A}_i \nabla_\theta \log P_\theta(\tau_i | c)] = \mathbb{E}[A_i \nabla_\theta \log P_\theta(\tau_i | c)]$. Summing over i gives the same aggregate expectation as (13), hence $\nabla_\theta J_K$. This estimator is labeled l2o throughout.

C.4 Support of deletion credit

Proposition 3 (Credit support). If $\mathcal{R} = \mathcal{R}_\mu$ for a full-support measure μ , then

$$A_i > 0 \iff y_i = 1 \text{ and } \nexists j \neq i : y_j = 1, S_j \subseteq S_i. \quad (14)$$

Proof. A trajectory adds at most its own up-set, so $U_K = U_{-i} \cup \uparrow S_i$ when $y_i = 1$ and $U_K = U_{-i}$ otherwise. Hence $A_i = y_i \cdot \mu(\uparrow S_i \setminus U_{-i}) \geq 0$. Since U_{-i} is an up-set, $\uparrow S_i \subseteq U_{-i} \iff S_i \in U_{-i} \iff \exists j \neq i : y_j = 1, S_j \subseteq S_i$; full support of μ gives (14). $\square$

Only unrepeated group-minimal successes receive nonzero credit.

D Base-measure families and scalable credit computation

D.1 Witness values

We characterize witness values on $2^{\mathcal{E}}$ that depend only on set size.

Proposition 2 (Rank-uniform refinement). Let ν depend on S only through $|S|$. The refinement ratio $\nu(|S|)/\nu(|S|+1)$ is independent of $|S|$ if and only if $\nu(S) = p^{|S|}$ for some $p \in (0, 1)$, realized by the product measure that includes each element independently with probability p .

Proof. Write $\nu(k)$ for the common value on sets of size k , normalized by $\nu(0) = \nu(\emptyset) = 1$, and put $r_k = \nu(k)/\nu(k+1)$. Suppose $r_k = r$ for every k . Then $\nu(k+1) = \nu(k)/r$, and induction gives $\nu(k) = r^{-k}$, which is p^k with $p =$

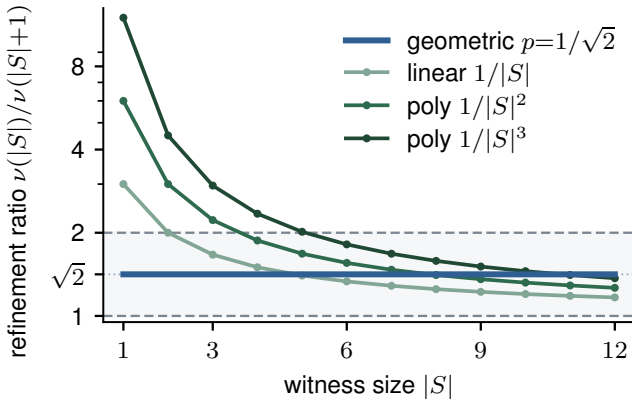


Figure 10: Per-element refinement ratio $\nu(|S|)/\nu(|S|+1)$ of the base measure. This ratio is $1/p$ for the geometric family and approaches one under polynomial reweighting as witness size grows. The band spans the two reference cases: strong refinement at $p = \frac{1}{2}$ and no refinement preference at $p = 1$. The experiments use $p = 0.7$, close to $1/\sqrt{2}$, which has equal logarithmic distances to the two reference cases.

$1/r$. Conversely $\nu(k) = p^k$ gives $r_k = 1/p$ for every k . Positivity and monotonicity of ν restrict p to $(0, 1)$. The product measure with $\Pr(t \in T) = p$ for each t satisfies $\nu(S) = \mu(\uparrow S) = p^{|S|}$ by the lemma below. $\square$

For polynomial reweightings of the layer-uniform measure, whose witness values decay as a power of $|S|$, the refinement ratio varies with set size. It approaches one as witnesses grow, and removing an element then gives less additional credit (Figure 10).

Lemma 1 (Product measures). *The uniform measure $\mu(T) = 2^{-n}$ gives $\nu(S) = 2^{-|S|}$. Any product measure $\mu = \bigotimes_t \text{Bern}(p_t)$ gives $\nu(S) = \prod_{t \in S} p_t$; if $\max_t p_t \leq \bar{p} < 1$ then $\nu(S) \leq \bar{p}^{|S|}$ decays exponentially in $|S|$.*

Proof. $T \supseteq S$ leaves the $n - |S|$ coordinates outside S free ($2^{n-|S|}$ sets); under independence $\Pr(T \supseteq S) = \prod_{t \in S} \Pr(t \in T)$. $\square$

Lemma 2 (Layer-uniform measure). *Draw $k \sim \text{Unif}\{0, \dots, n\}$ then T uniform in layer k , i.e. $\mu(T) = 1/((n+1)\binom{n}{|T|})$. Then $\nu(S) = 1/(|S|+1)$.*

Proof. With $m = |S|$, $\nu(S) = \frac{1}{n+1} \sum_{k=m}^n \binom{n-m}{k-m} / \binom{n}{k}$. Since $\binom{n-m}{k-m} / \binom{n}{k} = \binom{k}{m} / \binom{n}{m}$ and $\sum_{k=m}^n \binom{k}{m} = \binom{n+1}{m+1}$ (hockey-stick), the sum equals $\frac{1}{n+1} \frac{\binom{n+1}{m+1}}{\binom{n}{m}} = 1/(m+1)$. $\square$

The base measure controls the strength of refinement. Witness values decay exponentially with set size under product measures, while the layer-uniform measure assigns weight $1/(|S|+1)$. Both preserve the ordering induced by strict refinement.

D.2 Exact inclusion-exclusion credit

Fix a success S_i and let $D_i = \min_{\subseteq} \{S_j : j \neq i, y_j = 1\}$ be the minimal antichain of the other successes. Using the intersection identity $\uparrow A \cap \uparrow B = \uparrow(A \cup B)$,

$$\begin{aligned} A_i &= \mu(\uparrow S_i \setminus U_{-i}) \\ &= \nu(S_i) - \sum_{\emptyset \neq T \subseteq D_i} (-1)^{|T|+1} \nu\left(S_i \cup \bigcup_{x \in T} x\right). \end{aligned}$$

The expansion uses the witness values of Appendix D.1. If some $x \in D_i$ satisfies $x \subseteq S_i$, the credit is zero by (14). The exact sum costs $O(2^{|D_i|})$.

D.3 Shared-sample Monte Carlo credit

For large groups, $\mathcal{R}_\mu(U_K) = \Pr_{T \sim \mu}[\exists i : y_i = 1, T \supseteq S_i]$. Sampling $T^{(1)}, \dots, T^{(N)} \sim \mu$,

$$\hat{A}_i = \frac{1}{N} \sum_{m=1}^N \mathbf{1}[T^{(m)} \supseteq S_i] \prod_{x \in D_i} \mathbf{1}[T^{(m)} \not\supseteq x]$$

is an unbiased estimate of A_i . Computing the credit for one proposal costs $O(N|D_i|)$ and requires no additional verifier calls. The computation grows linearly with the number of group-minimal successes among the other proposals, while the exact inclusion-exclusion sum grows exponentially.

E MaxSAT experiments

E.1 Benchmark construction and ground truth

The MaxSAT benchmark provides an enumerable environment in which the complete witness family is known. An instance has Boolean variables $\mathcal{E} = \{1, \dots, n\}$ and positive clauses $C_1, \dots, C_m \subseteq \mathcal{E}$. A proposal $S \subseteq \mathcal{E}$ sets exactly the variables in S to true. Its verifier is

$$s(S) = \mathbf{1}[S \cap C_j \neq \emptyset \text{ for every } j \leq m], \quad (15)$$

and a proposal succeeds exactly when it intersects every clause. The minimal witnesses are therefore the prime implicants of the induced monotone Boolean function, equivalently the minimal hitting sets of the clause family.

We generate eight instances with $n = 14$, $m = 10$, and three variables per clause. Clauses are sampled uniformly without replacement within a clause, and duplicate clauses are discarded. Instance i uses random seed $1000 + i$. We rejection-sample formulas whose antichain size lies between 6 and 20; the eight retained instances contain 16 to 19 minimal witnesses, with mean 18.125. We evaluate all 2^{14} subsets and prune the sufficient sets by containment to obtain the antichain used for scoring.

E.2 Subset-construction MDP

Each formula defines a finite-horizon subset-construction MDP. At step t , the state is (S_t, t) , where S_t is the set of variables already selected. The policy observes the 14 membership bits of S_t and the normalized step index t/H . Its legal actions are

$$\mathcal{A}(S_t) = (\mathcal{E} \setminus S_t) \cup \{\text{STOP}\}. \quad (16)$$

Item	Setting
Instances	8 formulas, 14 variables, 10 clauses of length 3
Episode	add or STOP, horizon $H = 12$
Actor	$15 \rightarrow 128 \rightarrow 128 \rightarrow 15$, tanh activations
Rollout batch	8 groups per update, $K = 48$ proposals per group
Training	150 updates, 57,600 terminal verifier calls per formula
Optimization	Adam, learning rate 3×10^{-3} , entropy 0.02, gradient norm 1.0
MWRL	coverage measure with geometric μ , $p = 0.7$, l2o, mean-absolute rescaling, $N = 20,000$
Evaluation	128 fresh proposals per formula; policy seeds 0, 1, 2

Table 8: Common configuration for the MaxSAT policy-gradient comparison.

Choosing variable e produces the deterministic transition $S_{t+1} = S_t \cup \{e\}$. Choosing STOP terminates the episode at S_t . The horizon is $H = 12$, and an episode that reaches the horizon terminates at its current set. Intermediate rewards are zero. At termination, the environment returns the single bit $s(S_T)$ from Equation (15). Success does not terminate an episode automatically, and no variable is removed automatically. The policy must therefore learn both when a sufficient set has been reached and when further additions are redundant.

The main benchmark trains a fresh policy for each formula. The formula is fixed within a run and is not encoded in the observation. This isolates the behavior of the learning objective in the same way that a fixed maze isolates a navigation objective.

E.3 Training, baselines, and evaluation

All policy-gradient rows use the same masked categorical actor. The network maps the 15-dimensional observation through two 128-unit tanh layers to logits over the 14 add actions and STOP. Invalid add actions are masked before sampling. The GFlowNet baselines use the same actor architecture and share the policy-gradient configuration of Table 8.

One terminal proposal requires one call to the sufficiency verifier. Thus each policy-formula run uses $150 \times 8 \times 48 = 57,600$ training calls, followed by 128 evaluation calls. Samples drawn from μ for Monte Carlo credit do not query the verifier. Ground-truth enumeration is performed only for evaluation and is not supplied to any learned policy.

The scalar baselines differ only in the advantage assigned to the same rollout groups. MaxRL centers binary outcomes by the group mean and divides by that mean. GRPO centers by the group mean and divides by the group standard deviation. RLOO subtracts the mean outcome of the other $K - 1$ proposals. The PPO row applies the clipped update with threshold 0.2 to the MaxRL group advantage, while the remaining policy-gradient rows use the unclipped score-

function update. PPO with a size penalty applies the same clipped update to the group-standardized terminal reward $s(S) - 0.1|S|$. Recovery has converged even when the policy returns more than one witness: extending training by a factor of four changes only one of twenty-four runs. The second witness appears on formulas whose two smallest witnesses have the same size. Their penalized rewards are equal, so a distribution over the tied pair is itself optimal. The two GFlowNet rows use trajectory balance with a uniform backward policy. Their terminal rewards are $s(S)$ and $s(S)p^{|S|}$, respectively.

For each frozen policy, the born-minimal rate is the fraction of successful evaluation proposals that belong to the antichain. Antichain recall is the number of distinct minimal witnesses recovered in the 128 proposals divided by the antichain size. We also report the number of distinct minimal witnesses and the median size of successful terminal sets. Results are averaged first over the eight formulas and then over policy seeds.

E.4 Executed planning references

The two planning rows of Table 2 are executed on the same eight formulas. Scalar value iteration performs backward induction over (S, t) with terminal payoff

$$r_\lambda(S) = s(S) - \lambda|S|. \quad (17)$$

We use $\lambda = 0.1$ for the reported scalar planning row. The deterministic tie-breaking rule selects the lowest-index maximizing action. Repeating the resulting policy therefore returns the same minimum-size witness. With $\lambda = 0$, the planner terminates at non-minimal sufficient sets, whose mean size is 8.0.

Minimal-witness value iteration operates on pairs $(\mathcal{C}, b)$, where $\mathcal{C} \subseteq \mathcal{M}$ is the set of minimal witnesses already selected, which determines the certified region $F = \uparrow \mathcal{C}$, and b is the remaining proposal budget. For every $\mathcal{C}$, a subset-sum transform over witness-incidence masks computes $\mu(\uparrow \mathcal{C})$. Backward induction then evaluates

$$V_b(\mathcal{C}) = \max \left\{ \mathcal{R}_\mu(\uparrow \mathcal{C}), \max_{M \in \mathcal{M} \setminus \mathcal{C}} V_{b-1}(\mathcal{C} \cup \{M\}) \right\}. \quad (18)$$

At budget $b = |\mathcal{M}|$, the selected set contains the entire antichain on all eight instances. The implementation checks Equation (18) against brute-force enumeration of the selected sets for budgets up to four. It also checks the within-episode Bellman recursion against complete trajectory enumeration on a separate $n = 8$ instance. The maximum numerical discrepancy across these checks is below 10^{-14} .

E.5 Main learned-policy comparison

The prime-implicant comparison of Table 2 applies the preceding protocol independently to every formula and policy seed. We illustrate the terminal distributions induced by the three objective classes in Figure 1(a) to (c) and the sizes of their recovered families on the eight formulas in Figure 1(d). The complete antichain and both planning references make the terminal behavior of each objective directly observable.

E.6 Credit estimator

The credit-estimator experiment of Figure 5 isolates the estimator used after the deletion credit has been computed. We generate one enumerable instance with $n = 12$, eight clauses, horizon 10, and 11 minimal witnesses. An MWRL policy trains for 40 updates and is then frozen. For this policy and $K = 6$, we compute ∇J_K exactly by propagating its terminal distribution through all 2^{12} lattice states and differentiating the resulting grouped objective.

We compare group-mean centering, the l1o credit, and l2o on the same sampled groups. The experiment draws 8,192 groups and forms sixteen independent estimates from 512 groups each. We report the per-proposal credits on shared groups, their dispersion from the exact gradient, and their running relative error (Figure 5). The l1o credit and l2o converge to the exact gradient, while group-mean centering approaches a nonzero finite- K offset. We therefore use l2o in the main MaxSAT comparison.

E.7 Valuation and verifier access

We compare the two group measures of Appendix B on this benchmark (Table 7). The coverage measure requires only the terminal proposals, their verifier bits, and the chosen measure. The counting measure requires a canonicalizer Φ . For this monotone verifier, Φ makes one greedy pass over the selected variables and deletes a variable whenever the remaining set still satisfies all clauses. The deletion order is a deterministic function of the policy seed and terminal mask. It produces a minimal witness and adds at most $|S|$ verifier calls to a successful proposal.

The two measures place computation in different parts of the procedure. The coverage measure uses a problem-dependent base measure μ but learns refinement directly from the verifier bit. The counting measure assigns unit mass to each canonical witness and can use any deterministic search that returns a minimal witness as Φ . It avoids the choice of a base measure μ but requires running Φ before group credit is assigned. For the counting-measure row, both metrics are evaluated after Φ ; for the coverage-measure row, they are evaluated on raw terminal proposals.

E.8 Training group size and deployment budget

We separate the group size used to train a policy from the number of proposals drawn from it after training (Figure 6). We train on six of the benchmark formulas with policy seeds 0, 1, and 2. The training group sizes are $K \in \{2, 8, 32, 128\}$. Every training configuration uses 36,864 terminal verifier calls and eight groups per update. The corresponding numbers of updates are 2304, 576, 144, and 36, so larger groups receive neither more data nor more optimizer steps.

After training, each policy is frozen. We compute the exact terminal probability p_M of every $M \in \mathcal{M}$ by forward propagation over the complete subset lattice. The expected number of distinct minimal witnesses under deployment budget B is then

$$\mathbb{E}[N_B] = \sum_{M \in \mathcal{M}} [1 - (1 - p_M)^B]. \quad (19)$$

We evaluate this identity for B from 1 to 1024 and check it with prefixes of a 1024-rollout sample from each frozen policy. Small training groups produce concentrated terminal laws that saturate after few deployment proposals. Larger groups learn broader terminal laws and recover more of the antichain as B grows.

E.9 Antichain-geometry stress test

The geometry experiment replaces random clauses with a directly planted antichain $\mathcal{A}$ and uses the monotone verifier

$$s_{\mathcal{A}}(S) = \mathbf{1}[\exists M \in \mathcal{A} : M \subseteq S]. \quad (20)$$

This keeps the subset MDP and a monotone verifier while controlling the target family. We use $n = 20$ and cross four factors: antichain size $L \in \{8, 32\}$; balanced witnesses of size four or an equal split between sizes two and six; low or high pairwise overlap; and $L/K \in \{0.5, 2\}$. The overlap levels target the 0.1 and 0.9 quantiles between the minimum and maximum attainable mean overlap for each size profile.

Each of the sixteen conditions contains five planted predicates and three policy seeds. We train MWRL, a scalar policy gradient with a size penalty, and an up-set GFlowNet for 128 updates with 256 trajectories per update. Every learned run therefore uses 32,768 terminal verifier calls. The horizon is 10, evaluation uses 512 proposals, and MWRL estimates its credits from 5,000 Monte Carlo samples drawn from μ . The reported difference pairs MWRL with the stronger learned baseline within each condition after averaging policy seeds. Confidence intervals bootstrap the five predicate-level paired means.

The gain from deletion credit depends on the geometry of the target antichain (Figure 11). It is largest for large balanced families, where many incomparable witnesses compete for the same finite proposal budget. Size imbalance creates intrinsically favored witnesses and reduces the difference between objectives.

E.10 Base-measure choice for coverage credit

The base measure μ determines how strongly credit favors refinement. The uniform measure on $2^{\mathcal{E}}$ gives witness value $2^{-|S|}$. The layer-uniform measure gives $1/(|S| + 1)$. The geometric family gives $p^{|S|}$ and has constant per-element refinement ratio $1/p$. The main experiments use $p = 0.7$, close to $1/\sqrt{2}$, which has equal logarithmic distances to strong refinement ($p = \frac{1}{2}$) and to no refinement preference ($p = 1$). We compare this ratio with polynomial reweightings of the layer-uniform measure (Figure 12). Based on this comparison, we use the geometric measure as the default and keep p as a direct control of the trade-off between refinement and family coverage.

We vary p to measure how the base measure affects refinement and family recovery (Figure 7). The born-minimal rate stays between 0.71 and 0.79 across $p \in [0.5, 0.9]$. Every full-support measure assigns positive credit to the same group-minimal successes, so varying p preserves the incentive to remove redundant elements. As the per-element ratio $1/p$ approaches one, antichain recall rises from 0.55 to 0.87, and median proposal size rises from 3.8 to 4.2. A larger p

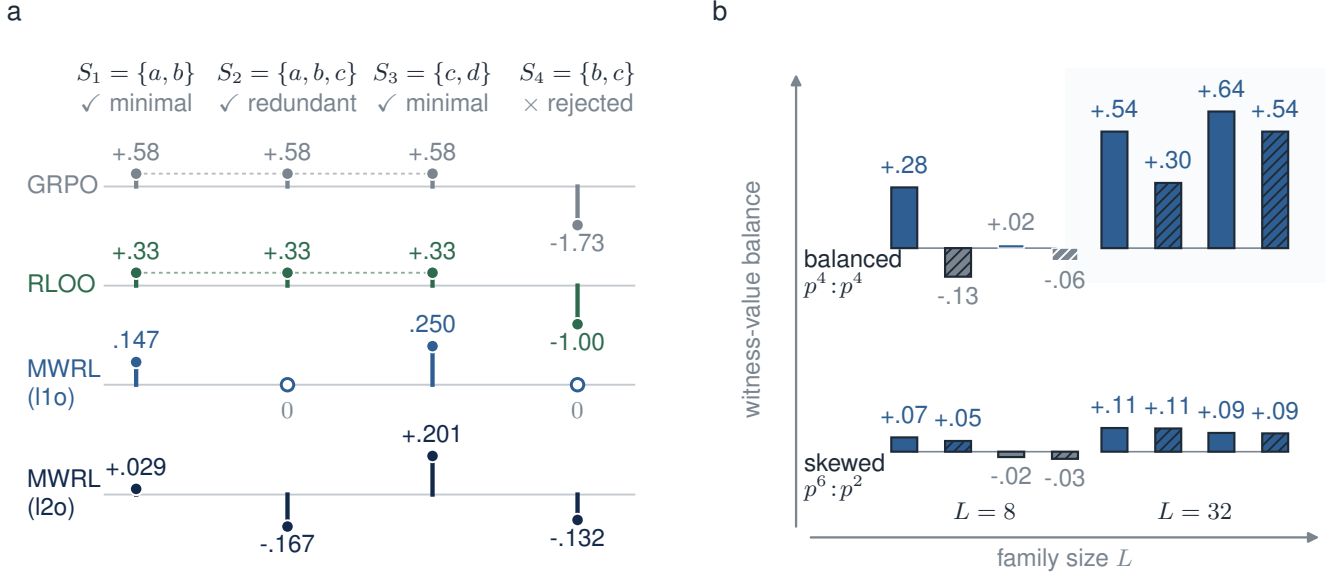


Figure 11: Deletion credit across planted antichain geometries. **(a)** The advantage each objective assigns inside one rollout group, under the planted verifier with $\mathcal{A} = \{\{a, b\}, \{c, d\}\}$ and the geometric measure at $p = 0.7$; stems are scaled per row and every value is printed. GRPO and RLOO use only the verifier bit: the redundant S_2 has the same advantage as the minimal witnesses S_1 and S_3 (dashed tie). Only the rejected proposal receives a different advantage. Raw deletion credit l1o assigns the mass certified only by each proposal: 0.147 and 0.250 for the two minimal witnesses, and zero for the redundant and rejected proposals. The l2o credit subtracts the leave-two-out baseline: S_3 retains positive credit, S_1 a small positive credit, and S_2 the lowest credit, below that of the rejected S_4 . **(b)** Sixteen conditions combine eight planted geometries with two group sizes. The axes vary family size L and the relative witness values: $|S_i| \equiv 4$ gives $p^4:p^4$, while $|S_i| \in \{2, 6\}$ gives $p^6:p^2$. Bars report the paired antichain-recall difference between MWRL and the strongest learned baseline after 32,768 matched verifier calls (5 predicates $\times$ 3 paired seeds). Bold edges mark 95% bootstrap intervals excluding zero; solid and hatched bars denote $K = 2L$ and $K = L/2$. Within each quadrant, bars order the low-overlap $q_{0.1}, q_{0.9}$ conditions followed by the high-overlap $q_{0.1}, q_{0.9}$ conditions.

MaxEnt RL, α	$\lambda = 0$	0.1	0.2	0.3	0.5	1
0.02	0.0 (0.00)	3.2 (0.77)	1.1 (0.87)	—	—	—
0.05	0.0 (0.00)	0.4 (0.00)	3.8 (0.49)	0.3 (0.24)	0.0 (0.00)	0.0 (0.00)
0.1	0.0 (0.00)	0.0 (0.00)	0.4 (0.00)	4.3 (0.12)	0.4 (0.20)	0.0 (0.00)
0.2	0.0 (0.00)	0.0 (0.00)	0.0 (0.00)	0.0 (0.00)	2.0 (0.02)	0.2 (0.13)

GFlowNet, λ	0.00	0.36	0.69	0.90	1.20	1.50	1.80	2.30
# found	0.2	0.8	2.0	3.0	4.3	3.6	3.1	3.0
born-min. rate	0.00	0.01	0.02	0.03	0.07	0.14	0.21	0.37

Table 9: Size-penalty sweeps for MaxEnt RL and GFlowNet on the eight MaxSAT formulas, each run with 57,600 verifier calls per formula. MaxEnt entries give the number of distinct minimal witnesses found and, in parentheses, the born-minimal rate. The GFlowNet penalties $\lambda = 0$ and $\lambda = 0.36 = \log(1/0.7)$ are the success and up-set rewards. Entries are means over three policy seeds after averaging the formulas; bold entries are reported in Table 2, and dashes mark settings not run.

brings the witness values of large and small minimal witnesses closer together, and the policy recovers more of this family, whose witness sizes range from 2 to 4. At $p = 1$, every witness value equals one, and the credit no longer favors smaller sets. We fixed $p = 0.7$ in advance based on these logarithmic distances, without per-domain tuning. Its born-minimal rate remains in this range; its column reproduces the MWRL row of Table 2.

E.11 Credit-rescaling sensitivity

Credit rescaling divides the credits of each group by a positive group statistic. It preserves the sign of every credit and rescales the gradient contribution of each group. We compare division by the mean absolute raw credit with division by the standard deviation of the baseline-adjusted credit. Both variants use the eight main formulas, policy seed 0, $K = 48$, and the same 57,600-query training budget. For both group-mean

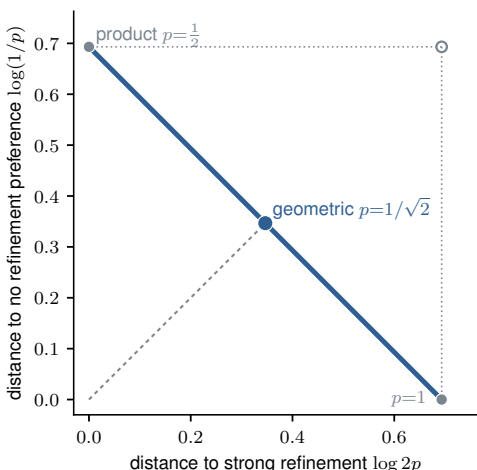


Figure 12: Logarithmic distances to the strong-refinement reference ($p = \frac{1}{2}$) and the limit of no refinement preference ($p = 1$). The distances sum to $\log 2$ and are equal at $p = 1/\sqrt{2}$; the experiments use $p = 0.7$. The dashed diagonal marks equal distances, and the open circle marks the unattainable ideal point.

centering and l2o, the two scales produce similar antichain recovery (Table 10); mean-absolute scaling gives the higher born-minimal rate, while standard-deviation scaling slightly increases l2o recall.

Centering	Rescale	born-min. rate $\uparrow$	antichain recall $\uparrow$	# found $\uparrow$
Group mean	Mean abs.	0.879	0.622	11.25
Group mean	Std. dev.	0.821	0.615	11.13
l2o	Mean abs.	0.819	0.675	12.25
l2o	Std. dev.	0.757	0.687	12.50

Table 10: Credit-rescaling sensitivity for coverage MWRL on the eight MaxSAT formulas. All rows use policy seed 0.

E.12 Size-penalized maximum-entropy RL and GFlowNet

GFlowNet and maximum-entropy (MaxEnt) RL spread probability over sufficient sets, and a size penalty moves that probability toward smaller sets. We add the penalty to both and sweep its strength. MaxEnt RL maximizes the entropy-regularized return (Haarnoja et al. 2018)

$$G(\tau) = s(S) - \lambda|S| - \alpha \sum_t \log \pi_\theta(a_t | x_t) \quad (21)$$

of a trajectory τ that ends at S . We standardize G within each group and apply the clipped update of PPO with a size penalty, keeping every other setting of Table 8; at $\alpha = 0$ and $\lambda = 0.1$ MaxEnt RL coincides with that baseline. GFlowNet uses trajectory balance (Malkin et al. 2022) with a uniform backward policy and the terminal reward $R(S) = s(S) e^{-\lambda|S|}$, which gives the success and up-set rewards of Appendix E.3 at $\lambda = 0$ and $\lambda = \log(1/p)$. A failed proposal

receives $R = \min(10^{-4}, 10^{-2}e^{-\lambda H})$, at least a hundredfold below any success within the horizon H .

Every run in this section uses the training budget of Table 8, 57,600 verifier calls per formula. GFlowNet uses 3,600 updates of 16 trajectories to reach this verifier-call budget, using Adam with step size 3×10^{-3} for the policy and 0.1 for $\log Z$. The success and up-set rows of Table 2 train on 150 updates of 16 trajectories with a single step size of 3×10^{-3} . At the full budget, the up-set reward recovers the same number of witnesses within one standard deviation (Table 9). We sweep $\alpha \in \{0.02, 0.05, 0.1, 0.2\}$ and $\lambda \in \{0, 0.1, 0.2, 0.3, 0.5, 1\}$ for MaxEnt RL and eight values of λ for GFlowNet, and we report in Table 2 the setting of each family that recovers the most distinct witnesses on the evaluation proposals.

Both objectives target laws proportional to a local reward, the second rule of Theorem 1. The reward weighs a minimal witness against its supersets only through their sizes. A weak penalty therefore leaves most probability on supersets, and a strong penalty concentrates it on the smallest witnesses. Recovery accordingly peaks at an intermediate strength in every row of Table 9. The peak moves to stronger penalties as α grows, since the entropy term rewards longer trajectories. Without a penalty, the successful proposals of MaxEnt RL reach the horizon at every α , because the maximum-entropy optimum weights each set by the number of add orders that reach it, a count that the uniform backward policy of GFlowNet divides out. At $\lambda = 1$ the penalty outweighs the success reward even for the smallest witnesses, and the policy stops before satisfying the formula. In both sweeps, higher born-minimal rates are accompanied by lower maximum witness recovery. At MWRL’s born-minimal rate, the strongest baseline setting recovers about one quarter as many witnesses as MWRL.

F Scientific variable identification experiments

F.1 Benchmark data and provenance

The benchmark combines published Suzuki–Miyaura reactions with simulated condition panels generated from mechanistic chemical rules. We train on the complete augmented grid as a closed-world reaction table. The release also includes the empirical partitions that anchor the augmentation; they retain their original schema.

The empirical data come from 2 published sources. The Perera flow-chemistry screen contributes the measured grid: 5,760 reactions over 15 substrate pairs on a complete $12 \times 8 \times 4$ ligand, base, and solvent factorial (Perera et al. 2018). The ChemRANG catalog contributes the reaction library (Cai et al. 2026), taken at repository commit dc0a14b: of 29,821 source records, 17,907 pass structure and yield quality control, curation of ambiguous or role-inconsistent entries leaves 16,659 unique canonical substrate pairs, and we store every rejected record in a quarantine table with a reason code. A task is one such pair, one single-site aryl or heteroaryl halide or triflate and one single-site organoboron partner. Every task identity, substrate structure, and product in the benchmark is therefore a literature reaction.

The augmentation supplies the condition panels that the sparse literature record does not contain. For each task, a mechanistic response simulator covering oxidative addition, transmetalation, reductive elimination, catalyst activation and deactivation, and the major side reactions generates yields over 14 discrete condition dimensions with one shared baseline. The panel enumerates every assignment that changes at most three dimensions. Each simulated yield averages four replicates with noise of standard deviation 0.2. No catalyst package dominates the others. The score tables encode documented trade-offs: bulky electron-rich monophosphine packages favor aryl-chloride and hindered couplings but tolerate chelating heteroaryls less well. Small-phosphine and bidentate ferrocenyl packages favor the heteroaryl-tolerant routes. As a result, different substrates require different sets of conditions for high yield. The empirical sources determine specific parts of the simulator. ChemRANG condition frequencies determine the available condition values and the shared baseline, and they are never read as success probabilities. Cleaned reported yields enter only through a weak task-level activity offset. The offset is proportional to the standardized median reported yield of the substrate pair, multiplied by $n/(n+2)$, where n is the number of literature records of the pair, and this factor shrinks the offset toward zero for pairs with few records. The Perera grid calibrates the yield distribution, its task heterogeneity, and the marginal and interaction structure of the factor effects; the response link is fitted on a fixed 32-task panel drawn before any outcome-conditioned selection, and the detection limit is set against simulated replicate means so that the zero-yield rate of the emitted table matches its calibration target. Each augmented task keeps explicit links to every strict ChemRANG record with the same substrates and product. These links establish the identity of each task and its reported successful conditions. The yield of a condition absent from the literature comes from the simulator. The augmented rows are simulator outputs anchored to real reactions, and the paper does not present them as measured yields.

A candidate task enters the release only if its induced ground truth passes the acceptance filters: an antichain of 2 to 80 minimal witnesses, at most a tenth of them singletons, a success fraction between 0.0005 and 0.6, and two stability conditions. First, the antichain computed from the noiseless simulator yields must equal the antichain computed from the released replicate means. Second, for every minimal witness, its best assignment must exceed the yield threshold by at least one point, and every assignment that changes only a strict subset of its dimensions must stay at least one point below the threshold. We evaluate the entire curated catalog of 16,659 tasks under generator seed 612, of which 1,381 pass the filters. We use a deterministic greedy procedure to select 1,024 tasks so that the minimal witnesses that occur across tasks are more evenly represented. We then use swap descent to bring the pooled yield aggregates within the calibration bands, subject to the popularity cap defined below. This selection changes which tasks appear in the release, not their simulated yields.

Before releasing the benchmark, we evaluate a substrate-blind popularity baseline. This baseline ranks minimal wit-

nesses by their frequency over the training side of an 80/20 task split and proposes the most frequent ones to every held-out task without using the substrate fingerprint; acceptance requires that this popularity strategy recover at most 0.45 of a typical held-out antichain at a budget of 16 proposals (measured curve in Table 11). The same variation across substrates occurs in the primary data: the thirteen substrate pairs of the Perera grid induce five distinct antichains, eight solved by a ligand swap alone, two with two alternative minimal witnesses, and three that require all three dimensions to move. Different substrates require different minimal sets of condition changes, so the amortization comparison tests whether the policy uses the substrate fingerprint to select those sets. If most tasks shared the same witnesses, a substrate-blind policy could recover them without the fingerprint. Because the fingerprint and the generator use the same mechanistic factors, the conditioned-versus-blind gap measures how much the fingerprint improves recovery within this benchmark.

The condition space of the released environment has 14 dimensions with one shared baseline protocol (Table 11). Every task exposes the same structured panel: the baseline row, all 28 singleton deviations, all 346 pairwise deviations, and all 2,504 triple deviations, giving 2,879 rows per task and 2,948,096 rows in total, with one global success threshold of 75 percent yield. The ground-truth antichain of a task is the family of inclusion-minimal successful deviation sets, between 2 and 35 members per task with mean 8.6; across the population the antichains draw on 201 distinct minimal witnesses in 986 distinct combinations. The evaluation derives the antichain from the table; the verifier never consults it. The Perera partition also defines a separate complete three-dimensional environment used for generator calibration and pilot runs, and no number in the paper comes from it.

F.2 Verifier, decision process, and policy

The verifier evaluates the sufficiency predicate. Opening a subset S of dimensions defines the subspace of condition assignments that deviate from the baseline only within S . In the reported benchmark, the verifier computes the closed-world verdict directly from the reaction table. It returns success if and only if a tabulated assignment with deviation set contained in S reaches the yield threshold. An assignment absent from the table is treated as a failure. This direct subset test is equivalent to exhaustively searching the tabulated assignments reachable within S , so the benchmark predicate is deterministic and monotone by construction. No Bayesian-optimization loop is executed during the reported training or evaluation. We compute coverage credit from the raw proposal and its verifier bit, so the policy must learn to produce minimal sets.

The table verifier is a computational abstraction of an experimental verifier. With wet experiments in the loop, the same interface can run Bayesian optimization over the assignments opened by S , using each experiment as a yield query. The benchmark uses direct lookup to represent the result of a converged Bayesian-optimization search, while making repeated policy training and controlled evaluation tractable.

The subset MDP uses add, remove, and stop actions over

Item	Setting
Tasks	1,024 substrate pairs; 205 held out per seed
Dimensions	catalyst system (6), base (6), solvent (6), water fraction (3), temperature (3), and binary time, catalyst loading, base equivalents, boron equivalents, concentration, ligand-to-palladium ratio, cosolvent, additive, and atmosphere
Panel per task	baseline, 28 singleton, 346 pairwise, and 2,504 triple deviations
Success	yield at least 75 percent; one global threshold
Antichain	2 to 35 minimal witnesses per task, mean 8.6; 201 distinct minimal witnesses, 986 distinct antichains
Popularity audit	held-out recall of the 16 most frequent minimal witnesses 0.42 (release criterion ≤ 0.45); 0.15, 0.25, 0.63 at budgets 4, 8, 32

Table 11: The released augmented Suzuki environment. Parentheses give the number of candidate values per dimension, including the baseline. The popularity audit row reports the substrate-blind frequency baseline used in the benchmark release criterion.

Item	Setting
Fingerprint Actor	33 substrate descriptors from SMILES hidden width 256, tanh; add, remove, or stop; full-set start; horizon 14
Group return	coverage measure with geometric μ , $p = 0.7$, l2o baseline, mean-absolute scaling
Amortized training	$K = 32$, 8 task groups per update, 4,800 updates
Per-substrate training	$K = 32$, 400 updates per task
Optimization	Adam, learning rate 5×10^{-4} , entropy 0.02, gradient norm 1.0
Evaluation	128 frozen-policy proposals per task; seeds 0, 1, 2

Table 12: Configuration for the Suzuki comparison.

the 14 dimensions. Episodes start from the full set and have horizon 14. The observation contains the current dimension mask, the normalized step index, and the task’s 33-dimensional substrate fingerprint. The fingerprint is computed from the halide and boron SMILES alone and records the leaving-group class, the organoboron class, heteroaryl and N–H content, the ortho substituent load at each coupling carbon, and polar-surface-area, mass, and lipophilicity proxies for both partners. These are the substrate properties that mechanistic Suzuki models treat as controlling oxidative addition, transmetalation, and catalyst poisoning. One policy is trained across the training substrates using the same coverage return as the other experiments, with the credit and optimization settings of Table 12.

F.3 Protocol and evaluation

We run the experiment with three seeds. Each seed determines the policy initialization and a shuffle of the 1,024 tasks, of which 205 are held out. The paper’s table reports mean and standard deviation over the seeds.

The four methods share the verifier, environment, and evaluation protocol. The two amortized variants train the same MWRL objective on the 819 training substrates and differ only in the observation: the conditioned policy receives the substrate fingerprint, and the substrate-blind policy receives zeros in its place. Scalar RL trains the same conditioned policy under best-of-group success credit in place of the MWRL group return. Per-substrate trains a fresh policy for every task and marks the recovery ceiling; it has no held-out entry because nothing is held out from it. One entry of that row is a complete training run, so the ceiling is measured on a fixed sample of 256 of the 1,024 substrates at a single seed; the sample gives its mean a 95 percent interval of ± 0.04 antichain recall.

Each frozen policy is evaluated within a window of 32 distinct condition sets per substrate: rollouts are drawn in order and deduplicated as sets, and the window closes once 32 distinct sets have been attempted, whether or not they verify. Antichain recall is the recovered fraction of the task’s ground-truth antichain within the window, using the same recovery criterion as the MaxSAT coverage rows: only raw terminal proposals that are themselves minimal count as recovered witnesses. The born-minimal rate is the fraction of successful attempts in the window whose raw terminal is already minimal, and the witness count is the number of distinct minimal witnesses recovered in the window. Recall is reported on training substrates and on held-out substrates; the born-minimal rate and the witness count are reported on the held-out split for the amortized policies and over the sampled substrates for the per-substrate ceiling. The evaluation stores 128 rollouts per task in proposal order with their verification outcomes, so every statistic is reproducible at any other budget from the released artifacts; elapsed training time for each method is stored with them.

F.4 Amortization on the fully synthetic library

Before the augmented experimental benchmark of this section, the amortization comparison was run on a fully synthetic Suzuki library: 172 substrate pairs over 18 discrete condition dimensions with structured coverage of every deviation of at most three dimensions, generated by the archived first-generation simulator. Each seed holds out 34 substrates, and the compared methods, credit rule, and optimization settings match Table 12 with the 128-proposal evaluation. The fully synthetic library gives the same ordering as the augmented experimental benchmark (Table 13). On held-out substrates, the conditioned policy matches the recall of the policies trained separately for each substrate. The substrate-blind ablation achieves less than half the conditioned policy’s recall, and the scalar-reward policy recovers none. The library is synthetic end to end, with a designed substrate-to-requirement structure; we report it here as a supporting instance.

Method	born-min.	recall $\uparrow$		#found $\uparrow$
	rate $\uparrow$	seen	held-out	
Per-substrate	0.58 $\pm$ 0.01	0.88 $\pm$ 0.00	—	6.4 $\pm$ 0.0
<i>Amortized MWRL</i>				
Conditioned	0.58 $\pm$ 0.03	0.86 $\pm$ 0.01	0.83 $\pm$ 0.04	6.1 $\pm$ 0.3
Substrate-blind	0.11 $\pm$ 0.05	0.41 $\pm$ 0.10	0.38 $\pm$ 0.08	2.7 $\pm$ 0.5
Scalar RL	0.00 $\pm$ 0.00	0.00 $\pm$ 0.00	0.00 $\pm$ 0.00	0.0 $\pm$ 0.0

Table 13: Amortized recovery on the fully synthetic 172-substrate library, mean and standard deviation over three seeds that control the substrate split and the policy. Per-substrate search retrains for each substrate and marks the recovery ceiling, so it has no held-out entry.

G LLM sparse-circuit discovery

Both experiments keep the language model frozen and search over attention heads and MLP blocks. An attention-head component is one query-head slice at the input to the attention output projection. An MLP component is the complete MLP output of one transformer layer. For each clean prompt x_i , we use another prompt from the same task as a counterfactual $\tilde{x}_i$. Removing a component replaces its clean activation with the aligned activation recorded on $\tilde{x}_i$.

Let P_i denote the clean next-token distribution, $P_{S,i}$ the distribution obtained when only components in S retain their clean activations, and $P_{\emptyset,i}$ the distribution when every component takes its counterfactual activation. For a probe set of size P , the reference KL divergence is

$$\text{KL}_{\emptyset} = \frac{1}{P} \sum_{i=1}^P D_{\text{KL}}(P_i \parallel P_{\emptyset,i}). \quad (22)$$

For a retained component set S , write

$$\text{KL}(S) = \frac{1}{P} \sum_{i=1}^P D_{\text{KL}}(P_i \parallel P_{S,i}). \quad (23)$$

The base verifier accepts S when

$$s_0(S) = \mathbf{1}[\text{KL}(S) \leq (1 - \alpha)\text{KL}_{\emptyset}], \quad \alpha = 0.8. \quad (24)$$

We report distributional faithfulness as $1 - \text{KL}(S)/\text{KL}_{\emptyset}$. We define the circuit canonicalizer Φ as an EAP-guided search within the proposed component set, followed by iterative single-component deletion. Every accepted deletion is checked by s_0 , and the returned canonical circuit $\Phi(S)$ is accepted by s_0 and 1-minimal. The search is deterministic for a fixed terminal mask and seed.

G.1 Qwen3-1.7B MMLU Bench10 and matched baselines

Benchmark construction. To construct Bench10, we evaluate the dense model on all 57 MMLU subjects and retain the ten on which it assigns higher probability to the correct answer token than to the cyclic foil on at least 0.92 of the discovery probes. The retained subjects are marketing, astronomy, computer security, high-school biology, high-school

chemistry, high-school computer science, international law, jurisprudence, management, and US foreign policy.

We use the test split of the `cais/mmlu_all` configuration. Each input is a zero-shot prompt containing the question, four choices labelled A to D, and the final cue `Answer: .` Discovery uses 12 questions per subject with random seed 0. Questions longer than 700 characters are removed. The selected set contains three questions for each answer letter and is ordered A, B, C, D. The next question in this cyclic order supplies the counterfactual prompt and foil answer.

Held-out evaluation uses 24 further questions per subject with random seed 1. It reconstructs the 12 discovery row identifiers, removes them before sampling, and selects six questions for each answer letter. Masks and baseline rankings are fixed from the discovery split before the held-out questions are loaded. A new verifier is then constructed on the held-out prompts, so $\text{KL}_{\emptyset}$ and the threshold in Equation (24) are recomputed on the evaluation split.

Model and subset MDP. Qwen3-1.7B has 28 layers and 16 query heads per layer. The component universe therefore has $28 \times 16 + 28 = 476$ elements. One policy is trained jointly across the ten subjects. Its task condition is the signed component-level EAP attribution vector, divided by its maximum absolute entry. The policy observes the current 476-bit component mask, the normalized step index, and this 476-dimensional task fingerprint. Its actions add one inactive component, remove one active component, or stop. Each episode starts from the full component set and has horizon 476.

The actor has two 512-unit tanh layers. The training return is the coverage measure with geometric μ on the raw terminal proposals with the raw verifier bit, adjusted by the l2o baseline and rescaled by the mean absolute credit, as recorded in Table 14; Φ is not applied during training. Evaluation deduplicates the raw terminals and counts the distinct 1-minimal circuits, checked by single-component removal tests. Each update samples one group of $K = 12$ trajectories for every subject. Training runs for 40 updates, giving 4,800 terminal proposals. The frozen policy is evaluated with 128 further proposals per subject (Table 14).

Baselines. For each subject, the smallest circuit found in the 128 frozen-policy proposals is the representative MWRL circuit and defines the matched size k . EAP retains the k components with largest absolute attribution. Magnitude ranks attention heads by their output-projection weight blocks and MLPs by their down-projection weights. The Wanda-style baseline ranks the mean norm of each component’s residual contribution on the discovery prompts. HardConcrete optimizes a differentiable component mask for 300 steps and retains the top k components in the resulting mask. Random samples five uniform k -component masks. ACDC-style greedy deletion starts from the full component set, tests deletions in EAP order, and retains its own converged size. Thus EAP, magnitude, Wanda-style, HardConcrete, and random use the same subject-specific k ; ACDC-style greedy is reported at the sparsity reached by its deletion procedure. The comparison matches sparsity; search cost differs by method

a

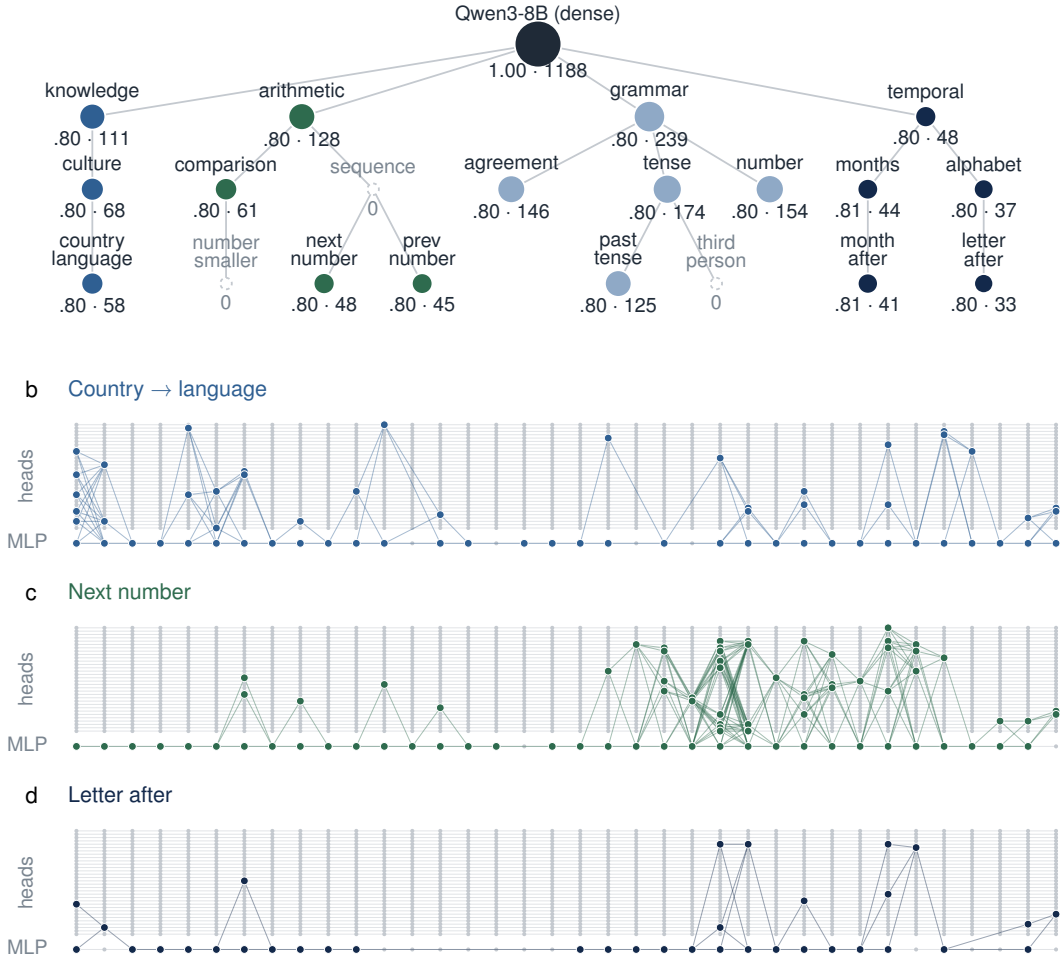


Figure 13: Hierarchical sparse circuits on Qwen3-8B. **(a)** The recovered portion of the taxonomy. Node area scales with circuit size; labels give distributional faithfulness and the median size of the recovered 1-minimal circuits. Dashed nodes denote tasks for which no circuit was recovered among the evaluation proposals. **(b–d)** Three leaf supports over the full component graph. Each support is the union of the successful canonical circuits in the 24 evaluation proposals for that leaf, covering 63, 81, and 36 of the 1,188 modules. Colored marks denote retained attention heads and MLP blocks; grey marks denote components given counterfactual activations.

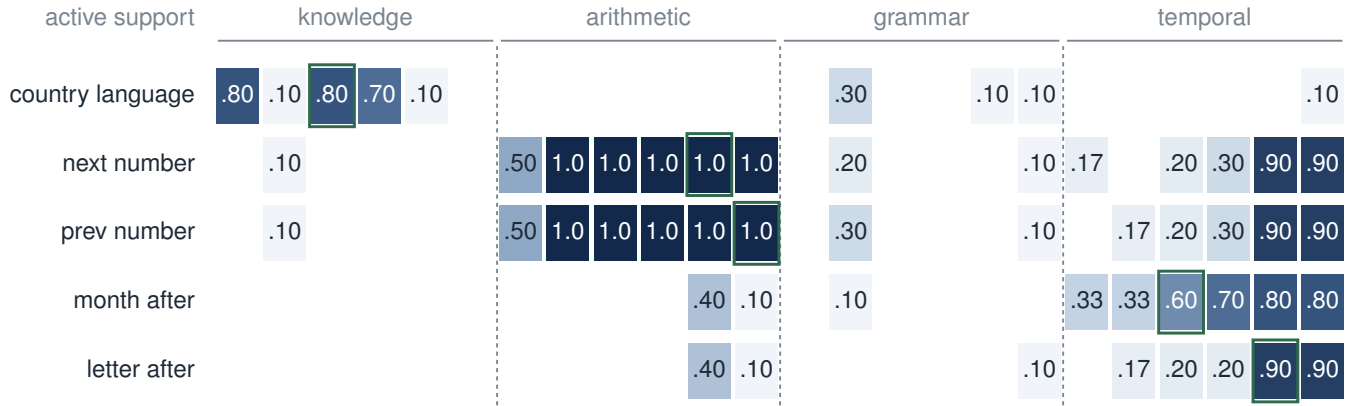
and is not equalized.

Evaluation. Discovery-side evaluation reports the born-minimal rate, the number of distinct 1-minimal circuits, the successful-proposal rate, median circuit size, and mean distributional faithfulness. The held-out evaluation reports four-choice accuracy, correct-versus-foil accuracy, distributional faithfulness, and the verifier bit for every frozen mask. For MWRL it also reports the fraction of the recovered circuit family that remains sufficient and, for a representative circuit that remains sufficient, the fraction of its components whose deletion breaks sufficiency. The cross-subject matrix activates subject j 's representative circuit and measures four-choice accuracy on subject i 's 24 held-out questions. Means

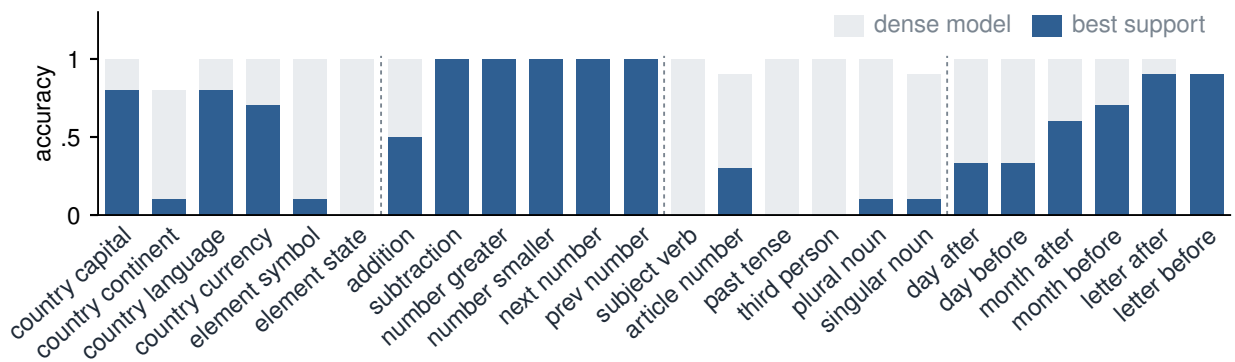
and MWRL-minus-baseline differences in four-choice accuracy use 10,000 paired bootstrap resamples over subjects.

Held-out four-choice accuracy is reported here for completeness. At the matched budget a mask keeps roughly a tenth of the components, and answer accuracy measures performance of the full question-answering pipeline, most of which is ablated; the verifier tests distributional faithfulness, a different quantity. The 24 held-out questions per subject are stratified by answer key, six per option, and chance is 0.25. Accuracy ranges from 0.13 for random- k and Hard-Concrete to 0.21 for MWRL, compared with 0.64 for the dense model. On this balanced answer key, the below-chance scores indicate a systematic preference for a plausible foil. This preference is as strong for random masks as for dis-

a



b



c

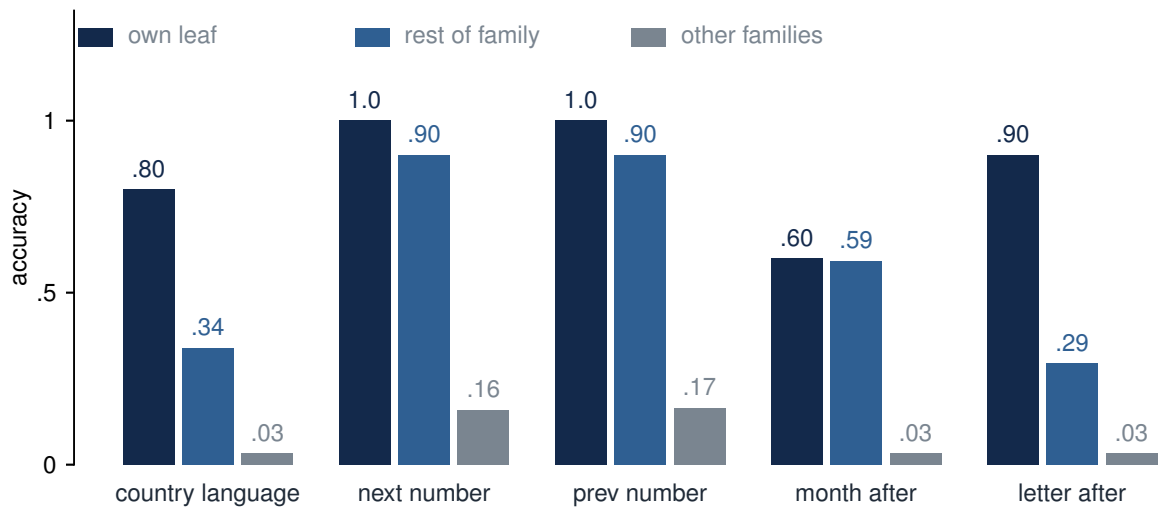


Figure 14: Task-indexed transfer of the Qwen3-8B leaf supports. **(a)** First-token correct-versus-foil accuracy of each recovered support, activated alone with every other component given its counterfactual activation, on every evaluation leaf. Columns group the leaves into the four task families, blank cells are zero, and boxes mark the support’s own leaf. **(b)** The highest accuracy any recovered support reaches on each leaf, against the dense model on the same leaf. Recovered supports match the dense model on most arithmetic leaves, but not addition; knowledge and temporal coverage is partial, and the search recovered no grammar support. **(c)** Each support’s mean accuracy on its own leaf, on the remaining leaves of its family, and on the leaves of the other families. Accuracy on a support’s own leaf averages 0.86 and accuracy outside its family averages 0.08.

Item	Setting
Model	Qwen3-1.7B, bfloat16, eager attention, frozen weights
Components	448 query heads and 28 MLP blocks
Verifier	raw verifier s_0 , $\alpha = 0.8$
Actor	$953 \rightarrow 512 \rightarrow 512 \rightarrow 953$, tanh activations
Episode	add, remove, or stop; full-set start; horizon 476
Group return	coverage measure with geometric μ , l2o baseline, mean-absolute scaling
Training	$K = 12$, ten groups per update, 40 updates
Optimization	Adam, learning rate 10^{-3} , entropy 0.02, gradient norm 1.0
Evaluation	128 frozen-policy proposals per subject; seed 0
Batching	576 prompt-mask sequences per model forward batch
Hardware	one NVIDIA L40 GPU

Table 14: Configuration for the Qwen3-1.7B Bench10 circuit-discovery experiment.

Item	Setting
Model	Qwen3-8B, bfloat16, eager attention, frozen weights
Components	1,152 query heads and 36 MLP blocks
Verifier	raw verifier s_0 , $\alpha = 0.8$
Actor	$2377 \rightarrow 512 \rightarrow 512 \rightarrow 2377$, tanh activations
Episode	add, remove, or stop; parent-support start; horizon 1,188
Group return	coverage measure with geometric μ , $p = 0.7$, l2o, mean-absolute scaling
Training per pass	$K = 10$, three groups per update, eight updates
Optimization	Adam, learning rate 10^{-3} , entropy 0.02, gradient norm 1.0
Support extraction	24 frozen-policy proposals per eligible task
Batching	512 prompt-mask sequences per model forward batch
Seed and hardware	seed 0; one NVIDIA L40 GPU

Table 15: Configuration for the three Qwen3-8B hierarchical passes.

covered circuits, pointing to heavily ablated decoding as its source. Four-choice accuracy does not reliably distinguish the methods at this budget: the MWRL-minus-EAP difference is $+0.02$, with a 95% paired bootstrap interval covering zero. Table 4 therefore reports distributional faithfulness, which does separate the method families on held-out questions, and discovery-prompt accuracy stays well above chance at a retention of 0.74.

Family	Subtask	Leaves
Knowledge	Geography	country capital; country continent
	Culture	country language; country currency
	Chemistry	element symbol; element state
Arithmetic	Basic	addition; subtraction
	Comparison	larger number; smaller number
Grammar	Sequence	next number; previous number
	Agreement	subject-verb agreement; article number
	Tense	past tense; third person
Temporal	Number	plural noun; singular noun
	Days	day after; day before
	Months	month after; month before
	Alphabet	letter after; letter before

Table 16: The complete Qwen3-8B behavior taxonomy used for hierarchical search.

G.2 Hierarchical circuit recovery on Qwen3-8B

On Qwen3-8B, we search a three-level taxonomy of four families, twelve subtasks, and twenty-four leaf behaviors over 1,188 attention-head and MLP components (Table 16). One policy pass searches the families; the next searches each subtask inside the support recovered for its parent family; the last restricts each leaf to its parent subtask’s support. The country-language leaf, for example, can select only components retained by Culture, whose search was restricted by Knowledge. This reduces the search space, but it also makes the child’s options depend on what its parents found. We therefore form each parent support from the union of its accepted circuits, preserving alternative circuits for the child search. We recover component supports for knowledge, arithmetic, and temporal tasks, but none for grammar (Figure 13).

To test whether a support is specific to the behavior for which it was found, we activate it on the other task probes and replace all components outside it with counterfactual activations (Figure 14). The supports retain much more accuracy on their own leaves than outside their task family. Arithmetic supports transfer across most of their leaves; the recovered supports cover only part of the knowledge and temporal tasks. For a leaf without a recovered support, this finite search did not find a circuit, and a sparse circuit for that leaf may still exist.

Taxonomy and probes. A subtask uses the pooled probes from its two leaves. A family uses the pooled probes from its three subtasks. The same probes thus define a parent task and the behaviors its support later constrains. Most leaves contain ten fixed probes. Element state contains eight, and each day relation contains six. The next prompt within each leaf supplies the cyclic counterfactual. Evaluation compares the first token of the correct answer with the first token of this counterfactual answer.

Hierarchical search. Qwen3-8B has 36 layers and 32 query heads per layer, giving $36 \times 32 + 36 = 1188$ components. The subset MDP uses the same add, remove, and stop actions as Bench10. Its observation has the current component mask, normalized step, and task-specific EAP fingerprint. The first pass trains one conditional policy over the four

family tasks. For each family, 24 frozen-policy proposals are canonicalized, and the union of their successful circuits forms a parent support. The second pass is warm-started from the family policy and trains jointly over the eligible subtasks. Each subtask search is restricted to its family’s support. The same procedure produces a support for each subtask. The third pass is warm-started from the subtask policy and restricts each leaf to its subtask support. A child support is therefore contained in the union of sampled circuits from its parent.

Each pass uses the same coverage return (Table 15). Since $K = 10$, deletion credits are computed by exact inclusion–exclusion.

Hierarchy and task-indexed evaluation. In Figure 13, a displayed family or subtask node summarizes the circuits recovered for that task.

To measure specialization, each nonempty leaf support is activated on every leaf probe set for which the dense model exceeds 0.5 correct-versus-foil accuracy. Each pairing of a support with a probe set gives one first-token correct-versus-foil accuracy. The task label selects the support in this evaluation (Figure 14).

H Related work

H.1 Symbolic and oracle minimal-set discovery

Minimal sufficient sets arise under different names across logic, diagnosis, and combinatorial enumeration. Prime implicants are the irredundant terms of Boolean functions (Quine 1952; McCluskey 1956; Crama and Hammer 2011). Minimal diagnoses, correction sets, and unsatisfiable subsets identify irreducible causes of inconsistency or repair in constraint systems (Reiter 1987; Marques-Silva, Janota, and Belov 2013; Liffiton et al. 2016). Minimal hypergraph transversals express the same antichain structure as hitting sets (Berge 1989; Eiter and Gottlob 1995; Eiter, Gottlob, and Makino 2003; Fredman and Khachiyan 1996; Murakami and Uno 2014). These methods exploit the representation of the underlying formula or hypergraph to enumerate the family. Other algorithms work through oracle access. Membership queries can identify the boundary of a monotone Boolean function (Bioch and Ibaraki 1995), while generic monotone-predicate procedures extract individual minimal sets (Junker 2004; Marques-Silva, Janota, and Belov 2013; Janota and Marques-Silva 2016). This literature provides algorithms for minimal-set discovery and ground truth on enumerable instances. MWRL addresses a different question: how a learned policy should allocate a finite rollout group across an unknown witness family from verifier feedback.

H.2 Sufficient explanations and sparse circuits

Several explanation frameworks define an explanation through sufficiency. Anchors search for high-precision sufficient rules (Ribeiro, Singh, and Guestrin 2018), and sufficient input subsets identify minimal observed features that preserve a prediction (Carter et al. 2019). Formal explainable AI characterizes sufficient reasons through prime implicants and abductive explanations (Shih, Choi, and Darwiche 2018;

Ignatiev, Narodytska, and Marques-Silva 2019). Mechanistic interpretability asks the analogous question inside neural networks. Manual circuit analysis and automated circuit discovery search for sparse subnetworks that retain a target behavior (Wang et al. 2023; Conmy et al. 2023), while attribution patching reduces the cost of ranking candidate edges (Syed, Rager, and Conmy 2024). Pruning methods such as Wanda and HardConcrete also construct sparse subnetworks, although their objective is model compression, and explanation is a separate goal (Sun et al. 2024; Louizos, Welling, and Kingma 2018). Circuit explanations need not be identifiable: distinct subnetworks can satisfy the same behavioral criterion (Méloux et al. 2025). MWRL makes this multiplicity the target by recovering a family of incomparable sufficient circuits under a common verifier.

H.3 Diverse and relational reinforcement learning

Score-function policy gradients optimize expected return from sampled trajectories (Williams 1992; Sutton et al. 1999), and PPO supplies a clipped update for this objective (Schulman et al. 2017). GRPO, RLOO, and maximum likelihood RL (MaxRL) couple sampled rollouts through group normalization or baselines (Shao et al. 2024; Ahmadian et al. 2024; Tajwar et al. 2026). Their credit does not depend on whether one terminal set contains another or repeats the same witness. GFlowNets learn diverse terminal distributions from a supplied positive reward (Bengio et al. 2021; Malkin et al. 2022).

Methods that pursue diversity differ in what they compare: individual proposals, action distributions, or archives of solutions. Exploration bonuses and intrinsic motivation add a per-proposal novelty term that decays as the space is visited (Pathak et al. 2017). The asymptotic objective stays proposal-separable, and Theorem 1 continues to apply. Entropy-regularized reinforcement learning (Haarnoja et al. 2018) and quality-diversity search (Lehman and Stanley 2011; Mouret and Clune 2015) do hold a distribution or an archive. Their measure of difference is supplied from outside the verifier, as entropy over actions or as a hand-designed behavior descriptor. A diversity measure that ignores containment counts redundant supersets as distinct discoveries. Under monotone sufficiency, $\{a, b, c, d\}$ succeeds whenever $\{a, b\}$ does, so counting both as discoveries does not identify an additional minimal witness. Entropy over the certified region is in fact maximized away from the antichain, since the up-set of a small witness is outweighed by the many larger sets above it (Figure 1a). All our baselines use an entropy bonus (Table 8), and the GFlowNet rows of Table 2 are the strongest member of this family that uses only the verifier bit.

Minimal-witness identification starts only with a sufficiency predicate. MWRL derives a relational return from the induced order structure, so one deletion credit rewards refinement along containment chains and recovery across incomparable witnesses, and a superset of a witness already in the group receives zero credit.

H.4 Submodular RL

Submodular reinforcement learning maximizes a submodular function of the states or trajectories a policy visits, and global RL extends this to general non-additive trajectory utilities (Prajapat et al. 2024; De Santi, Prajapat, and Krause 2024). Both evaluate collections of states or trajectories jointly. MWRL’s group return is a function of this kind. The valuation $\mu(\bigcup_i \uparrow S_i)$ is a monotone submodular coverage measure of the certified up-sets, and its deletion credit is the submodular marginal gain. The group policy gradient therefore ascends a submodular objective. These methods take the utility as part of the problem specification. MWRL constructs the utility from the verifier bits and the subset order, and the coverage measure and its marginals depend only on the up-sets certified by accepted proposals.

H.5 Hypervolume and set-level objectives

Multi-objective optimization values a set of points by the hypervolume of the region they dominate and scores each point by its exclusive contribution, the volume the set loses without it (Guerreiro, Fonseca, and Paquete 2021). The indicator drives population selection in evolutionary search (Beume, Naujoks, and Emmerich 2007), action selection in multi-objective reinforcement learning (Van Moffaert, Drugan, and Nowé 2013), and gradient-based Pareto set learning (Zhang et al. 2023). On the subset lattice the coverage measure under the geometric measure is a hypervolume. Map each successful proposal S to its indicator vector in $\{0, 1\}^{\mathcal{E}}$, minimize every coordinate, and place the reference point at $(1 - p)^{-1}$ in every coordinate. A set then dominates its supersets, and the antichain $\mathcal{M}(c)$ is the Pareto set of $F(c)$. Splitting every coordinate at 1 cuts the box $[0, (1 - p)^{-1}]^{\mathcal{E}}$ into cells indexed by the subsets $C \subseteq \mathcal{E}$ of coordinates above 1. The region dominated by S covers the cell C exactly when $S \subseteq C$, and that cell has volume $(p/(1 - p))^{|C|} = (1 - p)^{-|\mathcal{E}|} \mu(\{C\})$. The hypervolume of a group’s successes is therefore $(1 - p)^{-|\mathcal{E}|} \mathcal{R}_\mu(U_K)$, and each exclusive contribution is the deletion credit up to the same constant.

Theorem 2 shows that any group value unchanged by deleting duplicate or redundant successes can be written as a function of the certified region. Under the geometric measure its valuation coincides with a hypervolume. Hypervolume methods take objective vectors as given. In MWRL a proposal enters the dominated region only after the verifier accepts it, and the Pareto set of $F(c)$ is known only through these acceptances.

In reinforcement learning with verifiable rewards, pass@ k policy optimization values a group of samples through its best member (Walder and Karkhanis 2025), and a uniformity penalty over correct answers counters the indifference of group-relative objectives among them (Lochab, Li, and Zhang 2026). Under a binary reward the first depends on the policy only through its success probability, and the second targets a law proportional to a local reward, the verifier bit itself. Both fall under Theorem 1, together with maximum likelihood RL (Tajwar et al. 2026). When correctness is closed under supersets, a uniform law over correct an-

swers gives each redundant superset the weight of a minimal answer. Order-preserving GFlowNets learn a reward consistent with a partial order on the candidates, and training concentrates the sampler on the candidates highest in the order (Chen and Mauch 2024). The order enters through comparisons among sampled candidates, a relational signal outside the rules of Theorem 1. A hypergraph neural network agent trained with reinforcement learning reduces the satisfiability checks needed to enumerate minimal unsatisfiable subsets (Ijima and Yawata 2026), the minimal witnesses of a monotone predicate.

I Scope of application

Minimal-witness identification applies to tasks that ask which subsets suffice for an outcome and provide a yes-or-no verifier. We give the conditions for applying MWRL below, followed by examples of settings that meet them and settings outside the formulation.

I.1 Conditions for applying MWRL

A setting instantiates the problem of Section 2 when the following hold.

- **Subset structure.** The outcome depends on a finite ground set $\mathcal{E}$ whose elements can be supplied or withheld, so a proposal is an element of $2^{\mathcal{E}}$.
- **Binary verifier.** The environment answers a proposed subset with the bit $s(S)$. Additional verifier calls permit the counting measure of Appendix B, and the coverage measure needs the bit alone.
- **Monotonicity, exact or approximate.** Supersets of a witness remain sufficient. Under a verifier that violates this, a canonicalizer returns 1-minimal sets, which need not be minimal. We measure how often the raw verifier rejects supersets of accepted sets (Appendix A.4).
- **Plurality.** The antichain holds several incomparable witnesses on a non-negligible share of contexts. A predicate with one minimal witness is a special case of the formulation, where a size penalty already succeeds and recovering a family brings no advantage. Theorem 1 concerns predicates with multiple minimal witnesses.
- **Costly verification over related contexts.** Calls to s are expensive against the size of $2^{\mathcal{E}}$, and contexts arrive from a distribution over which a policy can amortize. For predicates exposed as formulas, the symbolic enumerators of Appendix H recover $\mathcal{M}$ without training.

The base measure μ encodes domain knowledge by assigning weights to sets in the lattice. These weights determine which incomparable witnesses an optimal finite-budget planner selects (Appendix B). Several domains supply information for choosing μ before any learning begins: reliability gives component failure probabilities, retrieval gives the probability that a passage is returned, and security gives the cost of acquiring a capability. In those settings the valuation follows from the domain and the recovered family reflects the domain’s own notion of importance.

Setting	ground set $\mathcal{E}$	verifier bit $s(S) = 1$	base measure μ
<i>Instantiated in this paper</i>			
MaxSAT witnesses	Boolean variables	setting the variables in S to true satisfies every clause	geometric, $p = 0.7$
Reaction conditions	condition dimensions opened from a baseline	a tabulated assignment inside the opened set reaches the yield threshold	geometric, $p = 0.7$
Sparse circuits	attention heads and MLP blocks	with every other component given its counterfactual activation, faithfulness stays at least α	geometric, $p = 0.7$
<i>Further settings admitted by the conditions</i>			
Coherent systems	components	the system fails	component failure law
Attack graphs	vulnerabilities and credentials	the attacker reaches the goal	exploit acquisition cost
Least privilege	permissions	the workload passes its tests	exposure weight per grant
Why-provenance	database tuples	the query returns the answer	tuple reliability
Candidate keys	relation attributes	the attributes determine every tuple	attribute acquisition cost
Configuration faults	feature flags, input fragments	the build reproduces the failure	flag prevalence
Evidence minimization	retrieved passages, prompt segments, exemplars	the answer is unchanged	retrieval probability
Metabolic design	genes, nutrients	growth above the viability threshold	perturbation feasibility
Combination therapy	compounds and doses	efficacy above the response threshold	toxicity or cost weight
Biomarker panels	assays	accuracy above the diagnostic threshold	assay cost
Data attribution	training examples	the behavior persists after retraining	sampling law of the corpus

Table 17: Instantiations of minimal-witness identification. Each row names the ground set over which proposals are formed, the event that the verifier bit indicates, and the information a domain supplies for choosing the base measure of Appendix D.

I.2 Settings where the antichain is already the object

A coherent system is defined by a monotone structure function, and the minimal cut sets of that function form its antichain (Barlow and Proschan 1975). System failure probability is the measure of the union of the cut sets’ up-sets under the product law of component failures, evaluated by inclusion and exclusion, and the contribution of one cut set is what that union loses when the cut set is removed. Those two quantities are the coverage measure of Section 3 and its deletion credit, computed by the identities of Appendix D. The same correspondence holds for fault trees, attack trees, and k -of- n redundancy. Attack graphs pose the question from the adversary’s side, where the minimal sets of vulnerabilities that reach a goal are incomparable and reachability grows with acquired capability (Sheyner et al. 2002). A defense that closes one minimal set leaves every incomparable set open, which is the operational reason the whole family is the answer.

Database provenance reaches the same object under the same name. Why-provenance defines the witnesses of a query answer as the subsets of the database that produce it, and the minimal witness basis is the antichain of those subsets (Buneman, Khanna, and Tan 2001). Candidate keys repeat the structure over attributes, since every superset of a key determines the relation, and their enumeration meets the dualization problem of Appendix H. The sufficient-component cause model of epidemiology treats a sufficient cause as a minimal set of component causes and takes the family of such causes as the scientific target (Rothman 1976). Model-based diagnosis arrives from the complementary direction,

where minimal diagnoses are the minimal hitting sets of the observed conflicts (Reiter 1987).

These fields supply ground truth on enumerable instances and independent evidence that the coverage measure is the quantity their practitioners compute. Their predicates are symbolic and are already served by enumeration.

I.3 Settings that require a learned policy

Configuration and input reduction. A failing build or a crashing compiler input is verified by running it, at the cost of a compile and a test suite. Delta debugging returns one failure-inducing subset from which no single element can be dropped, and states its guarantee as 1-minimality (Zeller and Hildebrandt 2002), the weakening our non-monotone verifiers also accept. Distinct irreducible fault combinations correspond to the incomparable members of the family, bug reports on one code base form the context distribution, and the ground set of flags or input fragments is far too large to enumerate.

Least privilege and attack surface. The ground set is the set of permissions granted to a workload and the verifier is its integration suite. An additional permission never breaks a passing run, so sufficiency is monotone, and the minimal permission sets that allow the workload to pass its integration tests differ in the exposure they create. Choosing among them is a decision made over the recovered antichain.

Evidence minimization. Retrieval and long-context pipelines ask which passages, prompt segments, or exemplars suffice for a correct answer, with an answer-match bit as the verifier and the retrieval law as μ . The predicate is approximately monotone, since added context occasionally

displaces the evidence a model attends to (Liu et al. 2024), which places it in the regime of Appendix A.4. The antichain measures evidential redundancy: an answer carried by several incomparable minimal evidence sets rests on more independent support than one carried by a single set.

Perturbation biology. Gene knockouts, media compositions, and reprogramming factors define ground sets whose sufficiency is a growth or phenotype bit, monotone in the genes or nutrients supplied and computable in silico by flux balance analysis (Orth, Thiele, and Palsson 2010). The factors for reprogramming to pluripotency were identified by withdrawing each candidate factor from a pool of 24 and testing whether the remaining factors were still sufficient (Takahashi and Yamanaka 2006). This procedure is the single-element deletion test that defines 1-minimality, run by hand at experimental cost, and it reduced the pool to 4 factors. Minimal drug combinations at an efficacy threshold and minimal assay panels at an accuracy threshold have the same subset structure. The full family matters because the choice among minimal witnesses depends on toxicity, cost, or deliverability.

Data attribution. Which training subsets suffice to induce a model behavior is a sufficiency question over examples, verified by retraining and testing (Koh and Liang 2017). Verification is the most expensive in this list, so amortization across behaviors is what makes the question affordable at all.

I.4 Settings outside the formulation

Monotone submodular maximization, influence maximization, and sensor placement optimize a graded utility and return a single set under a budget, so the antichain plays no role in their answer (Prajapat et al. 2024). Objectives of minimum cardinality, such as the smallest unsatisfiable core, rank candidate sets by size and discard the larger minimal witnesses by construction. A verifier that returns a continuous quality score provides information that the up-set construction does not use, and a valuation that uses that score is outside the verifier assumptions analyzed here. These cases fall outside the formulation, because their answer is a single set or depends on more than the subset order.

J Ethical statement

The formulation is domain-general. One setting that fits it is scientific variable identification, where the elements are experimental conditions, mutations, or components and the verifier is an assay. A policy that proposes minimal condition sets directs experiments when each benchmark lookup is replaced by a laboratory measurement (Appendix F).

The verifier specifies the outcome sought, its success threshold, and the admissible condition space. If a deployment allows hazardous reagents, unsafe protocols, or an inappropriate endpoint, it allows them in the verifier, where the domain’s own review and containment practices apply. The credit uses one bit and never observes the quantity behind it. MWRL can accelerate a search a laboratory has already sanctioned, and it cannot originate one.

A recovered family is guaranteed to satisfy the criterion evaluated by its verifier. For circuits, we test whether acceptance on the discovery probes extends to held-out questions

from the same subject (Appendix G.1). The recovered circuits fail to pass the verifier again on those questions, so sufficiency depends on the verifier and the probe distribution. Biological and chemical deployments should be verified again on the population of interest before a minimal set is read as a mechanism, and a wider claim demands a wider probe distribution behind it.

Recovering the whole antichain has a further benefit. Reporting one smallest witness presents a single account of an outcome as though it were the only one, and the redundant pathways and backup mechanisms of a predicate stay hidden (Méloux et al. 2025). The antichain makes the alternatives explicit, which weakens any single account’s claim to necessity and keeps the alternatives available when one route is unsafe, unavailable, or costly to realize.

We used AI tools mainly for language polishing, grammar correction, and novelty checks against prior work.