REWARD INTERACTIONS AND PRECISION GOVERN RESOURCE MEMORY BEFORE THE TASK

Anonymous authors
Paper under double-blind review

ABSTRACT

Adaptive computation can leave unfinished work whose cost depends on later choices. In language models with early exits, for example, a later deeper token may require missing representations from earlier tokens. When a resource state must be summarized once for objectives revealed later, the summary must preserve information before its intended use is known. We study this requirement in a deferred-cost model with exact budget safety and an irreversible first commitment. We prove matching upper and lower bounds showing how reward interactions and numerical precision jointly govern the worst-case message size. At fixed regret, interactions confined to small groups permit logarithmic memory growth with the horizon, whereas interactions spanning a constant fraction of the horizon can require linear memory. Limited precision caps this growth, so a large requirement needs both extensive interactions and fine cost resolution. The lower bounds already hold for constructed prediction tasks whose optimizer, given the resource state, checks only a fixed number of plans. Preparing information for an unknown objective thus has a complexity distinct from optimizing that objective. We also identify supplied public cost structure under which an exact private summary remains short as shared rate precision increases, while charging computation separately. Exact finite checks and encoding experiments examine the constructions and their computational limits. The results clarify when reusable resource interfaces must retain history and when earlier objective disclosure or prospective plan queries can avoid that information cost.

1 INTRODUCTION

Early-exit language models can leave keys and values unavailable at layers needed by later tokens. Earlier decisions therefore carry future resource liabilities. We ask what a reusable summary must preserve before its downstream objective is known. Figure 1 isolates why current spending alone can be insufficient.

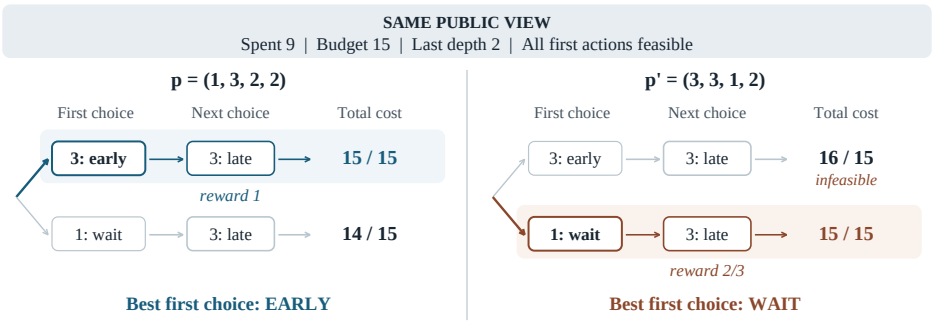

Figure 1: Same public view, different optimal commitments. Depth and rebate prices equal one, fees are zero, and the budget is 15. Depth three earns one third at the first future position and two thirds at the second. Highlighted plans maximize reward; the early-only plan earns one third. Appendix F verifies costs and reachability.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

SkipDecode avoids recomputing preceding KV caches through monotone exit points (Del Corro et al., 2023); CALM propagates a shallow hidden state to uncomputed layers (Schuster et al., 2022). We isolate a fixed-price model of deferred computation; these systems motivate it but do not establish the complete information interface below. Our summary is prepared before the objective is supplied and must support every objective in a declared class. Preparing that information can be difficult even when optimizing any one supplied objective is easy.

Both histories in Figure 1 have spent nine, end at depth two and share the same suffix maxima. Every first action is feasible, yet the best first choices differ. A later guard preserves budget safety but cannot recover reward lost by an early commitment. The modeled restriction is that no additional history-dependent exchange occurs before that commitment. Section 5 shows how earlier task disclosure or whole-plan feasibility queries can remove the need for a reusable descriptor.

We establish matching information laws governed by reward interactions and numerical precision. On one fixed physical family, bounded-size interactions permit logarithmic memory growth, while interactions over a constant fraction of the horizon can require linear memory. The precision law identifies when coarse numerical resolution prevents such a large requirement. Constructed prediction tasks with at most three noisy-parity heads already realize both laws, although a state-aware optimizer examines at most eight plans. Supports may grow with the horizon; the number of heads and candidate plans is constant.

The lower-bound argument uses standard information tools, but must realize their queries through legal computation histories. The rank construction keeps the action histogram, spending, initial guard mask and future price tape fixed across reward ranks. Its irreversible first-choice gap survives randomization and later exact feedback. This realization, together with the matching constructive order and precision cap, is the contribution beyond an abstract indexing problem.

We also give short exact messages under supplied public cost features with bounded integer coordinates in few dimensions. With three depths, two known liabilities and an upper bound on the third, this code preserves every feasible future plan. A two-rate implementation uses classical Euclidean arithmetic and charges shared precision separately from private message length. Finite checks and encoder controls test these mechanisms; a separate neural execution study reports its computational scope and unfavorable cases in the supplement.

2 Related work

The closest settings differ in what is unknown, what is represented, and when the objective becomes available (Table 1). Here the physical tape is known, the private input is a reachable resource history, and the utility arrives intact after encoding. The guarantee is uniform over a declared later utility class under hard safety and the specified feedback interface.

Table 1: Input and timing distinctions in representative related settings. Each row describes the cited model, not every variant of the area.

Setting	Hidden / represented object	Objective timing	Main guarantee
Advice (Böckenhauer et al., 2024)	Advice about future input	Online knapsack objective fixed	Competitive value
Valuation sketches (Cohavi and Dobzinski, 2017)	Valuation represented by a sketch	Valuation known to sketcher	Approximate set values
Reward-free RL (Jin et al., 2020)	Unknown transition dynamics	Rewards after exploration	Near-optimal policies
Unknown capacity (Disser et al., 2017)	Capacity learned by packing	Item values known	Robust packing value
Successor features (Barreto et al., 2017)	Expected feature occupancy	Linear rewards vary across tasks	Policy transfer
This work	Private resource history; public cost tape	Utility after message	Safe, low-regret decisions

Successor features separate dynamics from a linear reward representation to transfer policies across tasks (Barreto et al., 2017). They summarize expected discounted feature occupancy; our message summarizes a realized resource history for later utilities under exact safety. Adaptive computation (Graves, 2016) and Mixture-of-Depths (Raposo et al., 2024) choose where to spend compute. CALM, SkipDecode and LayerSkip (Schuster et al., 2022; Del Corro et al., 2023; Elhoushi et al., 2024) address early-exit execution and quality. We ask a different question: which resource distinctions a reusable interface must preserve before its objective is known.

The MPH hierarchy is due to Feige et al. (2015); MPH-1 is XOS and includes normalized monotone submodular valuations. The deletion inequality we use is explicit in Ezra et al. (2019, Lemma 6.2 of the 2018 preprint). Our contribution is a matching message law on one physically reachable history family: reward rank varies while public telemetry stays fixed (Theorem 1). The precision law then determines when limited numerical resolution prevents large information requirements even at high rank (Theorem 2). The proofs combine the deletion structure with irreversible threshold queries and an information lower bound. Parity complementarity (Krause and Guestrin, 2005, Example 1) supplies the prediction construction; the result here establishes the same resource-information law with few heads and an easy state-aware oracle.

Objective-independent knapsack formulations use large-item filtering and deletion repair (Bienstock, 2008, Theorem 1.1) (Pritchard, 2011, Theorem 2). They approximate a fixed integer hull for linear objectives; our message distinguishes hidden liabilities and supports feasible integral decisions. Superincreasing (SI) knapsack lexicography (Gupte, 2016), self-bounding inequalities (Vondrák, 2010), augmented-index arguments (Chakrabarti and Kondapally, 2011), and enumerative coding (Cover, 1973) supply tools. Few-weight algorithms (Etscheid et al., 2015), two-variable knapsack (Hirschberg and Wong, 1976) and fixed-dimensional integer programming (Lenstra, 1983) supply computational context. Our structured code converts a public superset of future costs into an exact liability alphabet, implemented with classical arithmetic.

Related settings include uncertain-capacity packing (Kawase et al., 2019; Klimm and Knaack, 2022), advice (Böckenhauer et al., 2025; Borodin et al., 2018), reward-agnostic exploration (Li et al., 2024), state abstraction (Abel et al., 2020; Allen et al., 2021; Wang et al., 2024), and deletion-robust summaries (Kazemi et al., 2018; Dütting et al., 2022). The relevant distinctions are the represented object, objective timing and allowed feedback. Weak submodularity (Thiery and Ward, 2022), zero-error side information (Witsenhausen, 1976), and action-dependent observations (Permuter and Weissman, 2011) provide further structural and coding context.

3 Model and information contract

A depth word $\mathbf{z} \in \{1, \dots, K\}^N$ records the computation depth chosen at each position. Each position has a depth price ℓ_i, a fixed fee c_i and a rebate Q_i when its chosen depth is below a later maximum. Its complete cost is

$$C(\mathbf{z}) = \sum_{i=0}^{N-1} \left[\ell_i \max_{j \geq i} z_j + c_i - Q_i \mathbf{1}\{z_i < \max_{j \geq i} z_j\} \right]. \tag{1}$$

Here ℓ_i prices one additional depth, c_i is a fixed fee, and Q_i rebates work omitted when an old position supplies only the dependencies needed by a later deeper position. In the audited projection/MLP ledger, normalized prices are $\ell = 6, Q = 5$; this excludes SDPA work and elapsed time (Appendix AB). Three depths isolate the first nontrivial conditional liability rather than claiming that deployed models have only three layers. Prices are fixed independently of actions, $\ell_i \geq Q_i \geq 0$, and $\ell_i + c_i \geq 0$. The suffix maximum represents deferred work: a later deeper computation can increase the cost attributable to an old position. For a prefix p, define its liability profile $g_p(M)$ as its old-position cost when the future reaches maximum M. If $D_{\mathrm{fut}}(w)$ is the standalone cost of a nonempty future word, then

$$C(pw) = g_p(\max w) + D_{\mathrm{fut}}(w), \qquad g_p(1) = C_p. \tag{2}$$

The profile is nondecreasing, all-shallow completion minimizes cost, and every feasible full plan respects all earlier prefix-spending bounds. Appendix A proves these accounting identities.

All logarithms and message lengths are in base two. The principal notation is summarized below; additional construction parameters are defined locally.

Table 2: Principal notation.

Symbol	Meaning
K, R	Number of depths; remaining horizon
$g_p(M), D_{\mathrm{fut}}(w)$	Past liability at future maximum M; standalone future cost
d, ρ, ε	MPH reward rank; utility regret; budget slack
P, Π, k	Normalized liability precision; future partition; witness sparsity
s_{f}, H	Public cost-feature dimension; coordinate bound

Handoff contract. Fix a finite prefix family, budget B, and future price tape of length $R = N - |p|$ before encoding. Every admitted prefix has a feasible completion. Public initial telemetry contains exact spending C_p and the exact viable first-action mask $A_B(p)$. An encoder then retains a fixed-length b-bit message; only afterwards is a utility $U : \{1, \dots, K\}^R \to [0, 1]$ supplied. Other task observations are common across histories compared by a lower bound. Later exact spending and guard feedback are permitted. Safety is almost sure, and the requirement is

$$\mathbb{E}U(W) \geq V_p^*(U; B - \varepsilon) - \rho, \qquad V_p^*(U; b) = \max_{w : C(pw) \leq b} U(w), \tag{3}$$

with empty maximum defined as zero. The main results use $\varepsilon = 0$, so they improve ordinary utility regret without extra budget. Random seeds are independent of history unless their information is charged to the message.

Every history-dependent datum crossing the handoff is charged to the message, including action logs, parameters, caches and auxiliary inputs. The information measure is the private message; the exact physical ledger, public prices, immutable algorithm, supplied utility and subsequent working memory are accounted for separately. In particular, streaming ledger reconstruction and peak RAM are different resource measures. Prospective whole-plan cost quotes provide an alternative communication interface (Section 5).

Information timing and the retained state. The information arrives in this order:

History ⟶ Message ⟶ Task ⟶ Action and later feedback

For each fixed public instance, the same encoder must support every admitted history and every later utility in the declared class. It may know that class and the target regret, but not the realized task. The receiver gets the intact task after handoff. Task-aware encoding and prospective whole-plan queries are alternative interfaces, analyzed in Section 5.

Three-depth conditional families. A prefix's suffix maxima partition it into consecutive buckets. At a fixed partition and exact spending, the possible three-depth profiles have the form

$$g_v = (g_1, g_2, \alpha - v), \qquad v \in \mathcal{V} \subseteq [v_{\min}, v_{\max}], \tag{4}$$

where g_1, g_2, α are public constants. The coordinate v is a tight-weight subset sum in the maximum-two bucket, conditional on the total tight weight already revealed by spending. Appendix A derives this anchored support from physical histories; Appendix C constructs the hard histories explicitly.

4 A MATCHING COMPLEMENTARITY–MEMORY LAW

Reward rank. A positive-hypergraph reward assigns nonnegative weight to groups of positions and earns each weight only when the entire group is activated at depth three. An MPH-d reward is the maximum of such rewards whose groups contain at most d positions (Feige et al., 2015); total reward is normalized to at most one. Rank bounds group size, not the number of groups. Rank one already includes nonadditive XOS rewards. Appendix W gives the formal definition. These rewards depend on activated positions; lattice-submodular rewards can also distinguish intermediate depths.

One physical family for every rank. For every horizon $R \geq 4$, we construct reachable histories that share their spending, depth histogram, last depth, suffix partition and initial feasible actions.

Their hidden future liabilities differ. The future prices and budget stay fixed while the reward rank varies. Appendix W specifies this common family using paired past buckets with dyadic prices. Let $b_d(\rho)$ be the least message length supporting every normalized MPH-d task on this entire family. The same encoder must serve them all.

Theorem 1 (Complementarity–memory law)**.** *For zero budget slack and fixed* $0 < \rho < 1/10$*,*

$$b_d(\rho) = \Theta_\rho(d \log_2(1 + R/d)), \qquad 1 \le d \le R - 1. \tag{5}$$

Constants are independent of d *and* R*. Encoders and actors may be randomized, and later exact spending and guards remain available. The same asymptotic order is sharp for the worst-case handoff information over arbitrary legal public price tapes at* $K = 3$*, conditional on public past partition and spending, with unrestricted computation.*

The hard family uses normalized liability precision $P = s = R - 2$, which is linear in the horizon. Matching means asymptotic order at fixed regret: the finite constants can be loose. At $R = 258$ and $\rho = 1/20$, the rank-one lower and upper bounds are about 1.32 and 96 bits; at rank 16 the upper bound reaches the 252-bit exact history index (Figure 2).

Why the upper and lower bounds match. The upper bound retains only the thresholds needed by a sparse set of costly positions. The established MPH deletion inequality (Ezra et al., 2019) ensures that if rounding excludes an optimal plan, deleting one of these positions repairs feasibility with controlled reward loss. Counting the possible sparse witnesses gives the upper order. At large rank, an exact index gives a shorter alternative. The arbitrary-price version provides the same information order, with unrestricted computation.

For the lower bound, physically reachable histories look identical through the public interface but disagree on which early commitment is best. A task rewards an early action and the later completion of a small group. Depending on the hidden liability, the actor should either take both rewards or reserve resources for the larger later reward. The wrong first choice loses a fixed amount, which later feedback cannot recover. Many ordered histories can be separated in this way. Their first choices therefore recover information about the hidden history; the ordered-bit argument remains valid under randomization. Appendix W gives the witnesses, entropy calculation and uniform constants.

The construction holds the physical family fixed across reward classes, so the change from logarithmic to linear memory isolates the effect of reward interactions. These are class worst cases at fixed regret; individual tasks can be easier. The next result limits the information requirement by the precision available to represent liabilities.

A matching precision cap. Let $\mathcal{B}_d(R, P, \rho)$ be the worst-case minimum message over legal public tapes and finite conditional families at $K = 3$, with public past partition and spending, where g_3 lies on a public affine grid of step 2^{-P} in an interval of width at most one. Thus P measures resolution over a normalized liability range, not the hardware mantissa.

Theorem 2 (Precision–complementarity law)**.** *For* $R \ge 4$*, integer* $P \ge 1$*,* $1 \le d \le R - 1$*, and fixed* $0 < \rho < 1/10$*, under the same randomized and later-feedback contract,*

$$\mathcal{B}_d(R, P, \rho) = \Theta_\rho(\min\{P, d \log_2(1 + R/d)\}). \tag{6}$$

Constants are uniform in R, P, d*. Computation is unrestricted.*

The precision ceiling follows directly from an exact grid index; the substantive lower-bound construction matches that ceiling jointly with reward rank. To match it from below, a mixed-radix price palette packs distinguishable liability thresholds into the available horizon and reward rank. Paired past positions keep public telemetry identical, and the same early-versus-late decisions force a fixed loss when histories are confused. Appendix Y proves reachability and the uniform constants.

Here the hard price tape may depend on rank and precision; the previous theorem uses one tape across ranks. When precision grows only logarithmically with the horizon, every reward rank has logarithmic worst-case information. Linear memory requires both precision and reward rank to grow at least linearly.

5 A PREDICTION WITNESS WITH AN EASY ORACLE

This constructed witness separates information preparation from optimization; it is not a deployment claim. Consider independent binary features hidden until their positions are acquired at depth three. Their distribution is public and independent of resource history. After the descriptor is fixed, a task specifies up to three prediction heads. Each head predicts the parity of at most d requested features, with independent label-noise probability $\eta < 1/2$. The loss is a nonnegative weighted average of the heads' classification errors. Their supports and the feature distribution are given; learning the parity functions is not part of the task.

With unbiased features, an incomplete head is unpredictable: a single missing feature leaves its parity equally likely to take either value. Acquiring its complete support reduces its Bayes error to the noise floor. Prediction therefore realizes exactly the completion rewards used by the lower bound. The Bayes-risk identity is derived in Appendix AC.

Corollary 5.1 (Compact prediction tasks)**.** *The orders in Theorems 1–2 hold unchanged when the required task class is restricted to these prediction heads and regret means excess Bayes risk at most $(1/2 - \eta)\rho$. For each unbiased task and a supplied exact resource profile, the Bayes-optimal plan is found among at most eight candidates in $O(R)$ arithmetic, with numerical bit costs separate.*

To optimize, enumerate unions of the heads' supports, acquire each union at depth three and leave other positions shallow. Every feasible word dominates the union of its completed heads, so one candidate is optimal. The lower bounds therefore separate the information needed by a reusable descriptor from the complexity of any one supplied task: each task has a compact description and a constant-size candidate menu.

The information orders tolerate label noise and sufficiently small feature bias, with the admissible regret range stated in Appendix AC. Bias gives partial observations some predictive value, so the eight-plan optimizer is guaranteed only for unbiased features. At the other extreme, deterministic features require no acquisition information. These distinctions separate robustness of the information law from tractability of a particular optimizer.

Controls identify when resource memory is needed. For a single unbiased head, acquiring each requested feature whenever the guard allows it is optimal with zero resource bits: there is no competing head to trade against. If the task reaches the encoder first, an m-bit index identifies the best of 2^m plans, requiring at most three bits for $m \leq 3$. At most eight whole-plan feasibility queries after task disclosure but before acting also remove the need for a pre-task descriptor. These controls isolate two ingredients of the separation: competing objectives and a representation fixed before their identity is known. A system designer can instead expose the task earlier or provide prospective plan queries, charging that additional communication and service. Appendix AC.5 proves the controls and gives a biased counterexample to the eight-plan optimizer.

6 ACHIEVABILITY

6.1 SPARSE-WITNESS COMPRESSION

The encoder groups future plans by their suffix-maximum partition Π. Each nonempty bucket above depth one has a mandatory final anchor. At other positions, a monotone-optimal plan uses either the bucket maximum or one depth below it. Call the former positions *tight*. For each partition, selecting tight positions is an optional-item knapsack: their saving prices are added to a fixed base cost. Across partitions there are quadratically many such knapsacks. Appendix B proves this reduction.

Write T for the selected optional tight positions and $f_\Pi(T)$ for the resulting utility. We require a bounded total deletion loss: the utility losses from removing each selected position individually must sum to at most a. Formally,

$$\sum_{i \in T} [f_\Pi(T) - f_\Pi(T \setminus \{i\})] \leq a \quad \text{for every canonical } \Pi, T. \tag{7}$$

For monotone lattice-submodular U in $[0, 1]$, $a = 1$, by the standard diminishing-returns telescoping inequality. Additive monotone rewards are included. Monotone DNFs with clause width d, and

nonnegative weighted sums of them with total weight at most one, have $a \leq d$ (Appendix O). These deletion bounds follow established self-bounding structure (Vondrák, 2010).

Theorem 3 (Sparse-witness compression)**.** *Assume* (4) *and the handoff contract. Suppose the first* $R-1$ *saving prices are positive and, when sorted increasingly, each is at least the sum of its predecessors. The final saving price need not satisfy this condition. For every integer* $k \geq 1$*, a deterministic task-independent code is safe at* B *and has regret at most* a/k *for all monotone utilities satisfying* (7)*. It uses the literal message bound*

$$b \leq 1 + \big(\min(k, R-1) + 2\big)\lceil \log_2(R+1) \rceil. \tag{8}$$

At $k \geq R-1$ *it has zero regret. Encoding takes* $O(R^3)$ *arithmetic and* $O(R)$ *working entries without an exponential codebook. Decoding and exact optimization of a monotone value oracle require* $O(R^3)$ *oracle calls and* $O(R^4)$ *explicit-word bookkeeping in the delivered implementation, with numerical bit costs and oracle evaluation charged separately.*

Why the approximation works. Record feasibility thresholds supported by at most k optional tight positions, then conservatively round the hidden resource state to one of these thresholds. Plans with at most that many tight positions remain feasible. For a larger optimum, the code preserves enough capacity for its k most expensive tight positions. Under the stated price ordering, removing any one of them also pays for the unrecorded tail. The deletion-loss condition ensures that some such removal loses at most a/k utility. Anchors remain in place, preserving the partition. This explains both safety at the original budget and the regret bound.

The encoder finds a sparse threshold in each partition using superincreasing knapsack lexicography (Gupte, 2016). Its literal packet, greedy search and monotone optimization are proved in Appendix B. Appendix N extends the code to every fixed depth bound, with a computational exponent depending on that bound. Arbitrary saving prices retain the information upper order via a tail allowance (Appendix Q), but efficient general encoding remains open.

6.2 Exact messages from public cost structure

The precision law is a worst case over tapes. A complementary question is when public cost structure permits exact compression even at high rate precision. The following regime preserves every feasible word, so its representation guarantee is independent of the utility class.

Exact encoding from public resource features. Suppose every depth price, fee and saving price is an integer linear combination of the same s_{f} positive rational rates. The rates and integer coefficient vectors are supplied publicly before history and task, and remain fixed across admitted histories. Every coefficient has magnitude at most H; the resulting physical prices must satisfy the model's legality conditions. This is an assumed cost representation, not a factorization that the encoder has to discover. Appendix AJ states the exact equalities.

Theorem 4 (Exact resource-feature code)**.** *Let* $K = 3$*, with public liabilities* g_1, g_2 *and a public bound* $\bar{g} \geq g_3$*. Under the supplied factorization above, if every feature coordinate has magnitude at most* $H \geq 1$*, a task-independent private message of* $O(s_{\mathrm{f}} \log(RH+1))$ *bits preserves the entire feasible future-word set at the same budget. Representation regret is zero for every word utility. For* $s_{\mathrm{f}} = 2$*, public setup uses* $O(Rs_{\mathrm{f}})$ *arithmetic operations; given* g_3*, encoding uses* $O(\beta)$ *integer operations, where* β *bounds normalized integer bit lengths, including intermediate values.*

Proof mechanism and charged resources. Sum the possible integer coefficient ranges across positions to obtain a public box containing every future cost. Each box point defines a liability threshold where a plan can change feasibility. Round the hidden third liability upward to the next such threshold, including its public upper bound as an endpoint. This preserves every feasible maximum-three word and admits no infeasible word; lower maxima use the two public liabilities. A box-point index is the message, so its length depends on the number of bounded integer coordinates rather than the precision of the shared rates.

For two rates, a Euclidean modular predecessor finds that index without enumerating the box. Appendix AJ proves exactness and charges numerical bit complexity. The private message excludes public rates, features and the exact physical ledger. Extracting the state from history, discovering a favorable basis and optimizing the later utility remain separate computational costs.

The private message is one resource measure. Public setup, state extraction, numerical arithmetic and optimization of the later utility remain distinct costs. Approximate arbitrary-price search and additive planning extensions are provided in the extended supplement; hardness of one exact subroutine does not rule out a better approximate encoder.

7 Computational evidence

The evidence directly examines the information laws and their constructive encoders. Exact finite checks verify physical reachability, separating first-choice gaps, Bayes identities and codec safety. They test specified finite domains rather than establish asymptotic theorems (Appendix AL).

Finite bounds and certified minima. Figure 2 distinguishes achievable bounds from the small-instance optimum. For the dyadic family, counting only reward-relevant plans tightens the rank-one upper bound to 96 bits at $R = 258$ and $\rho = 1/20$; the lower bound is 1.32 bits. At rank 16 the upper code still uses the full 252-bit history index. Appendix X reports several horizons.

We also certify the exact minimum in 23 of 36 specified small-instance configurations; 13 remain unresolved. Rational lower certificates exclude one fewer bit than the exact-history code, allowing randomized encoders and later feedback. At rank two and regret 0.01, the minima are one, two, three and four bits for horizons four through seven. Appendix X.2 proves this certificate and reports the complete grid. These small cases require all bits of the history index; they do not demonstrate finite-horizon compression or a rank separation.

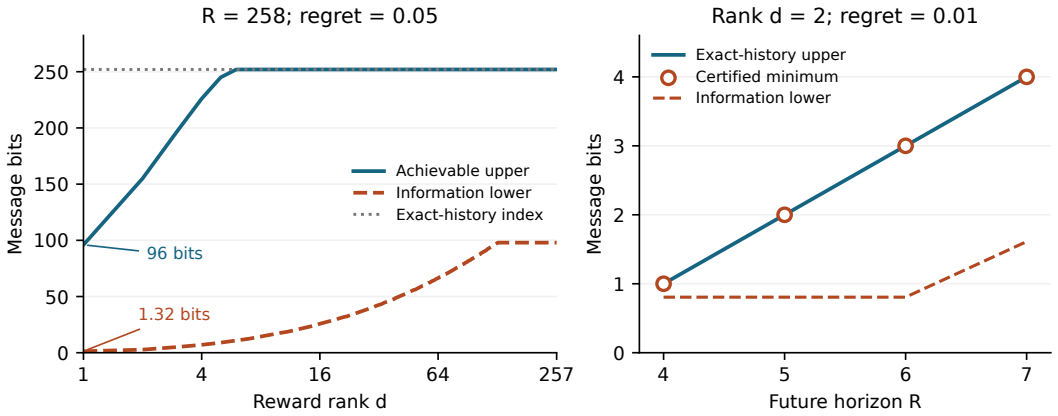

Figure 2: **Left:** proved lower and achievable upper bounds on the common dyadic family at $R = 258$, $\rho = 0.05$, compared with its exact history index. The finite gap remains substantial. **Right:** certified minimum bits for rank two, $\rho = 0.01$, under the same randomized and later-feedback contract. Circles mark exact optima; the line through them is the matching achievable bound. The complete grid includes unresolved cases (Appendix X.2).

Exact encoding at high public precision. The two-rate implementation reads 16,386 past actions and encodes a 10,000-position future with 4,096-bit rates in 43 logical message bits (six bytes). At horizons 64, 1,024 and 10,000, packet lengths are 21, 33 and 43 bits at each tested rate precision. Shared rates, features, the exact physical ledger and workspace are charged separately. This realizes the distinction between public numerical precision and private message length. A separate eight-position witness requires positive information below regret one fourteenth and uses a 20-bit exact packet; at eight-bit precision the simpler exact grid wins. Appendix AJ.5 reports every case and control. Small cases enumerate complete words; large-case feasibility follows from the exact-code proof.

A separate pretrained-model CPU study measures selective execution with complete plans supplied to all methods; its primary regime saves about 10% latency but its full-panel FP32 fidelity criterion fails (Appendix AM), so it supplies no evidence of a benefit caused by the resource message.

8 DISCUSSION

Designing the resource interface. The results separate interface design from representation design. In the constructed prediction tasks, earlier task disclosure reduces the retained information to a short plan index. If later tasks must share a representation, the matching laws instead specify the worst-case information budget needed for their reward interactions and liability precision. Under the structured-code assumptions on public costs and liabilities, an exact message preserves every feasible future plan, so the objective can change without re-encoding history. Encoder time, workspace and execution latency determine whether a chosen representation pays off in a system. The low-precision grid and high-precision two-rate encoder give different routes to a short exact message.

Scope and open directions. The matching laws describe class worst cases at fixed regret in the three-depth model; instance-specific optima and vanishing-regret limits remain open. Exact structured compression assumes supplied public features and lower liabilities. Discovering that structure and finding a fast, short code for unstructured high-precision prices remain algorithmic questions. A deployed service satisfying the complete pre-task handoff remains to be established: dynamic cache events, token identities and prospective worker queries change the interface. The supplementary systems scope compares these departures explicitly. Packet-compression benefit, online serving and predictive-quality gains therefore require separate evidence.

9 CONCLUSION

Preparing resource information has a complexity distinct from optimizing a known task: reward interactions and numerical precision determine matching worst-case message orders even for prediction tasks with an eight-candidate optimizer. Supplied public cost structure can instead keep exact private representations short, with computation accounted for separately. Designers should therefore specify the information interface first: expose the objective before encoding, provide prospective whole-plan feasibility queries, or budget a reusable message for the later reward class and liability precision.

REPRODUCIBILITY STATEMENT

Complete proofs of the main results and the finite verification domains appear in the compact appendices. Appendix AL maps claims to the source, frozen results and replay commands in the supplementary archive. The extended supplement contains algorithmic extensions and the complete comparisons supporting the reported computational results. Finite verifiers use exact arithmetic where specified; timing reruns measure repeatability on fixed inputs, not independent replication.

AI USE STATEMENT

We used generative AI tools for research ideas and constructions, proof and counterexample development, literature comparison, experiment design, implementation, test execution, analysis, writing and artifact preparation. The authors have manually reviewed all proofs and take full responsibility for the arguments, evidence, citations and final content. Exact finite verifiers and replayable results document the computational checks.

REFERENCES

David Abel, Nate Umbanhowar, Khimya Khetarpal, Dilip Arumugam, Doina Precup, and Michael Littman. Value preserving state-action abstractions. In *AISTATS*, PMLR 108:1639–1650, 2020. https://proceedings.mlr.press/v108/abel20a.html.

Cameron Allen, Neev Parikh, Omer Gottesman, and George Konidaris. Learning Markov state abstractions for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 34, 2021. https://papers.nips.cc/paper_files/paper/2021/hash/454cecc4829279e64d624cd8a8c9ddf1-Abstract.html.

André Barreto, Will Dabney, Rémi Munos, Jonathan J. Hunt, Tom Schaul, Hado P. van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017. https://papers.nips.cc/paper/2017/hash/350db081a661525235354dd3e19b8c05-Abstract.html.

Daniel Bienstock. Approximate formulations for 0-1 knapsack sets. *Operations Research Letters*, 36(3):317–320, 2008. https://www.columbia.edu/~dano/papers/bienstock-knap-final.pdf.

Mustafa Bilgic and Lise Getoor. Value of information lattice: Exploiting probabilistic independence for effective feature subset acquisition. *Journal of Artificial Intelligence Research*, 41:69–95, 2011. https://doi.org/10.1613/jair.3200.

Hans-Joachim Böckenhauer, Fabian Frei, and Peter Rossmanith. Removable online knapsack and advice. In *STACS*, LIPIcs 289:18:1–18:17, 2024. https://doi.org/10.4230/LIPIcs.STACS.2024.18.

Hans-Joachim Böckenhauer, Matthias Gehnen, Juraj Hromkovič, Ralf Klasing, Dennis Komm, Henri Lotze, Daniel Mock, Peter Rossmanith, and Moritz Stocker. Online unbounded knapsack. *Theory of Computing Systems*, 69, Article 14, 2025. https://doi.org/10.1007/s00224-025-10215-0.

Allan Borodin, Denis Pankratov, and Amirali Salehi-Abari. A simple PTAS for the dual bin packing problem and advice complexity of its online version. In *SOSA*, OASIcs 61:8:1–8:12, 2018. https://doi.org/10.4230/OASIcs.SOSA.2018.8.

Amit Chakrabarti and Ranganath Kondapally. Everywhere-tight information cost tradeoffs for augmented index. In *APPROX/RANDOM*, LNCS 6845:448–459, 2011. https://doi.org/10.1007/978-3-642-22935-0_38.

Keren Cohavi and Shahar Dobzinski. Faster and simpler sketches of valuation functions. *ACM Transactions on Algorithms*, 13(3), Article 30, 2017. https://doi.org/10.1145/3039871.

Thomas M. Cover. Enumerative source encoding.
IEEE Transactions on Information Theory, 19(1):73–77, 1973. https://doi.org/10.1109/TIT.1973.1054929.

Luciano Del Corro, Allie Del Giorno, Sahaj Agarwal, Bin Yu, Ahmed Awadallah, and Subhabrata Mukherjee. SkipDecode: Autoregressive skip decoding with batching and caching for efficient LLM inference. *arXiv:2307.02628*, 2023. https://arxiv.org/abs/2307.02628.

Yann Disser, Max Klimm, Nicole Megow, and Sebastian Stiller. Packing a knapsack of unknown capacity. *SIAM Journal on Discrete Mathematics*, 31(3):1477–1497, 2017. https://doi.org/10.1137/16M1070049.

Paul Dütting, Federico Fusco, Silvio Lattanzi, Ashkan Norouzi-Fard, and Morteza Zadimoghaddam. Deletion robust submodular maximization over matroids. In *ICML*, PMLR 162:5671–5693, 2022. https://proceedings.mlr.press/v162/duetting22a.html.

Mostafa Elhoushi, Akshat Shrivastava, Diana Liskovich, Basil Hosmer, Bram Wasti, Liangzhen Lai, Anas Mahmoud, Bilge Acun, Saurabh Agarwal, Ahmed Roman, Ahmed A. Aly, Beidi Chen, and Carole-Jean Wu. LayerSkip: Enabling early exit inference and self-speculative decoding. In *ACL*, pp. 12622–12642, 2024. https://aclanthology.org/2024.acl-long.681/.

Michael Etscheid, Stefan Kratsch, Matthias Mnich, and Heiko Röglin. Polynomial kernels for weighted problems. In *MFCS*, LNCS 9235:287–298, 2015. https://arxiv.org/abs/1507.03439.

Tomer Ezra, Michal Feldman, Eric Neyman, Inbal Talgam-Cohen, and S. Matthew Weinberg. Settling the communication complexity of combinatorial auctions with two subadditive buyers. In *FOCS*, pp. 249–272, 2019. https://doi.org/10.1109/FOCS.2019.00025. The cited Lemma 6.2 is in the 2018 preprint, titled *Complement-Free Couples Must Communicate*: https://arxiv.org/abs/1811.09871v1.

Uriel Feige, Michal Feldman, Nicole Immorlica, Rani Izsak, Brendan Lucier, and Vasilis Syrgkanis. A unifying hierarchy of valuations with complements and substitutes. In *AAAI*, 29(1):872–878, 2015. https://doi.org/10.1609/aaai.v29i1.9314.

Alex Graves. Adaptive computation time for recurrent neural networks. *arXiv:1603.08983*, 2016. https://arxiv.org/abs/1603.08983.

Akshay Gupte. Convex hulls of superincreasing knapsacks and lexicographic orderings. *Discrete Applied Mathematics*, 201:150–163, 2016. Author manuscript: *arXiv:1503.03742v1*, 2015. https://arxiv.org/abs/1503.03742v1.

D. S. Hirschberg and C. K. Wong. A polynomial-time algorithm for the knapsack problem with two variables. *Journal of the ACM*, 23(1):147–154, 1976. https://doi.org/10.1145/321921.321936.

Chi Jin, Akshay Krishnamurthy, Max Simchowitz, and Tiancheng Yu. Reward-free exploration for reinforcement learning. In *ICML*, PMLR 119:4870–4879, 2020. https://proceedings.mlr.press/v119/jin20d.html.

Yasushi Kawase, Hanna Sumita, and Takuro Fukunaga. Submodular maximization with uncertain knapsack capacity. *SIAM Journal on Discrete Mathematics*, 33(3):1121–1145, 2019. https://doi.org/10.1137/18M1174428.

Ehsan Kazemi, Morteza Zadimoghaddam, and Amin Karbasi. Scalable deletion-robust submodular maximization: Data summarization with privacy and fairness constraints. In *ICML*, PMLR 80:2544–2553, 2018. https://proceedings.mlr.press/v80/kazemi18a.html.

Max Klimm and Martin Knaack. Maximizing a submodular function with bounded curvature under an unknown knapsack constraint. In *APPROX/RANDOM*, LIPIcs 245:49:1–49:19, 2022. https://doi.org/10.4230/LIPIcs.APPROX/RANDOM.2022.49. Extended manuscript: *arXiv:2209.09668v3*, 2024. https://arxiv.org/abs/2209.09668v3.

Andreas Krause and Carlos Guestrin. Near-optimal nonmyopic value of information in graphical models. In *UAI*, pp. 324–331, 2005. https://www.cs.cmu.edu/~guestrin/Publications/UAI2005/uai2005.pdf.

H. W. Lenstra, Jr. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8(4):538–548, 1983. https://doi.org/10.1287/moor.8.4.538.

Gen Li, Yuling Yan, Yuxin Chen, and Jianqing Fan. Minimax-optimal reward-agnostic exploration in reinforcement learning. In *Proceedings of the 37th Conference on Learning Theory*, volume 247 of PMLR, pp. 3431–3436, 2024. https://proceedings.mlr.press/v247/li24a.html.

Haim H. Permuter and Tsachy Weissman. Source coding with a side information vending machine. *IEEE Transactions on Information Theory*, 57(7):4530–4544, 2011. https://doi.org/10.1109/TIT.2011.2145510.

David Pritchard. An LP with integrality gap $1+\epsilon$ for multidimensional knapsack. *arXiv:1005.3324v2*, 2011. https://arxiv.org/abs/1005.3324v2.

David Raposo, Sam Ritter, Blake Richards, Timothy Lillicrap, Peter Conway Humphreys, and Adam Santoro. Mixture-of-Depths: Dynamically allocating compute in transformer-based language models. *arXiv:2404.02258*, 2024. https://arxiv.org/abs/2404.02258.

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Q. Tran, Yi Tay, and Donald Metzler. Confident adaptive language modeling. In *Advances in Neural Information Processing Systems*, volume 35, 2022. https://papers.nips.cc/paper_files/paper/2022/hash/6fac9e316a4ae75ea244ddcef1982c71-Abstract-Conference.html.

Theophile Thiery and Justin Ward. Two-sided weak submodularity for matroid constrained optimization and regression. In *Proceedings of the 35th Conference on Learning Theory*, volume 178 of PMLR, pp. 3605–3634, 2022. https://proceedings.mlr.press/v178/thiery22a.html.

Jan Vondrák. A note on concentration of submodular functions. *arXiv:1005.2791v1*, 2010. https://arxiv.org/abs/1005.2791v1.

Zizhao Wang, Caroline Wang, Xuesu Xiao, Yuke Zhu, and Peter Stone. Building minimal and reusable causal state abstractions for reinforcement learning. In *AAAI*, 38(14):15778–15786, 2024. https://doi.org/10.1609/aaai.v38i14.29507.

Hans S. Witsenhausen. The zero-error side information problem. *IEEE Transactions on Information Theory*, 22(5):592–593, 1976. https://ieeexplore.ieee.org/document/1055607.

GUIDE TO THE APPENDICES

The appendices below contain the complete arguments supporting the main theorems, together with finite checks and concise execution scope. The three blocks follow the information laws, constructive algorithms and computational evidence. The guide below locates each argument.

Block	Contents
I. Information laws	A–C: accounting, sparse repair and hard histories; W: reward-rank law; Y: precision law; AC: prediction witness and controls.
II. Algorithms	N: fixed depth; Q: arbitrary prices; AJ: exact public-feature code and implementation; O: deletion certificates.
III. Evidence	F: opening example; X–Z: finite law checks; AB: operator accounting; AL: claim/evidence map; AM: neural execution and fidelity.

Further extensions and complete comparisons are provided in `Supplement_Extended.pdf` in the supplementary archive. Its sections use the prefix S followed by the original appendix letter (for example, S.AG). References with that prefix refer to that separate PDF. Main theorem proofs are complete in the present document.

I. Proofs of the information laws

A Complete accounting and conditional-state proofs

This appendix gives self-contained proofs. The physical identity, anchored fibers, conditional rank construction, and exact dyadic slack law provide the foundation for the sparse-witness, fixed-regret reward-class, fixed-histogram, and first-choice results. Finite verification supplies separate evidence about implementations.

A.1 The physical identity, monotonicity, and reachable histories

For a prefix $p = (k_0, \dots, k_{t-1})$, put $r_i = \max_{i \le j < t} k_j$. Define

$$g_p(M) = \sum_{i<t} \left[\ell_i \max(r_i, M) + c_i - Q_i \mathbf{1}\{k_i < \max(r_i, M)\} \right], \tag{9}$$

$$D_{\mathrm{fut}}(w) = \sum_{i=t}^{N-1} \left[\ell_i \max_{i \le j < N} w_{j-t} + c_i - Q_i \mathbf{1}\{w_{i-t} < \max_{i \le j < N} w_{j-t}\} \right]. \tag{10}$$

Splitting the positions of the complete word proves $C(pw) = g_p(\max w) + D_{\mathrm{fut}}(w)$ and $g_p(1) = C_p$. Applying the same split after appending k proves

$$g_{pk}(M) = g_p(\max(k, M)) + \ell_t \max(k, M) + c_t - Q_t \mathbf{1}\{k < M\}. \tag{11}$$

To check coordinatewise monotonicity of C, first consider positions before a raised action. If their suffix maximum increases by one, their contribution increases by either ℓ_i or $\ell_i - Q_i$; both are nonnegative. At the changed position, increasing an action below an unchanged suffix maximum does nothing until it becomes tight, when it adds Q_i. Raising a tight maximum adds ℓ_i. These cases exhaust a unit increase, and induction handles larger changes. Each position costs at least $\ell_i + c_i \ge 0$; appending an action cannot reduce the old contributions and adds a nonnegative contribution. Thus an all-shallow remaining word minimizes cost, every feasible full word is executable under the viability guard, and its earlier spending is no larger than its final spending. These statements require both stated price inequalities.

A.2 Anchored equal-spending fibers

The nonincreasing sequence (r_i) forms consecutive buckets $I_K, \dots, I_1$. Write

$$W_j = \sum_{i \in I_j} \ell_i, \qquad T_j = \sum_{i \in I_j : k_i = j} Q_i, \qquad G = \sum_{i<t} (c_i - Q_i).$$

A position in bucket j contributes $\ell_i \max(j, M) + c_i - Q_i$, except that a tight position adds back Q_i precisely when $j \ge M$. Therefore

$$g(M) = G + \sum_{j=1}^{K} \max(j, M) W_j + \sum_{j \ge M} T_j. \tag{12}$$

Every nonempty bucket ends at a position with action j. Conversely, choosing that anchor and any subset of the preceding bucket positions to have action j, and setting the other actions to one, realizes that bucket. For $j > 1$, its tight weight lies in the distinct support

$$\mathcal{S}_j = Q_{\max I_j} + \left\{ \sum_{i \in A} Q_i : A \subseteq I_j \setminus \{\max I_j\} \right\};$$

an empty bucket has support $\{0\}$. All actions in I_1 equal one, so T_1 is fixed. At a fixed partition and spending, the profiles are in bijection with

$$(T_2, \dots, T_K) \in \mathcal{S}_2 \times \cdots \times \mathcal{S}_K, \quad \sum_{j=2}^{K} T_j = C - G - \sum_j j W_j - T_1.$$

Indeed,

$$g(M+1) - g(M) = \sum_{j \le M} W_j - T_M$$

recovers $T_2, \ldots, T_{K-1}$, and the fixed sum recovers T_K. This also proves that subset multiplicities must not be counted as different profiles. Across partitions, one takes a union rather than adding potentially overlapping profile counts. At $K = 3$, $g(1), g(2)$ are fixed and $g(3) = \alpha - v$, where $v = T_2$.

A.3 Rank characterization and its exact boundary

Let $\mathcal{D}_m = \{D_{\mathrm{fut}}(w) : \max w = m\}$ be sorted distinct future costs. For each profile define

$$r_m^0(g) = |\{d \in \mathcal{D}_m : g(m) + d \le B\}|,$$
$$r_m^-(g) = |\{d \in \mathcal{D}_m : g(m) + d \le B - \varepsilon\}|,$$
$$I_g = \prod_m [r_m^-(g), r_m^0(g)] \cap \mathbb{Z}^K.$$

Within each public class $Y = y$, put an edge between profiles with disjoint boxes. A proper coloring is sufficient for safety at B and zero regret against the $B - \varepsilon$ optimum. To prove this, pairwise intersecting boxes satisfy, coordinate by coordinate,

$$\max_{g \text{ in cell}} r_m^-(g) \le \min_{g \text{ in cell}} r_m^0(g).$$

Choose an integer q_m between these endpoints. The common menu with cost rank at most q_m is safe in every cell member and contains every tightened-feasible word of every member. Optimize the supplied utility over that menu. If it is empty, the common all-shallow completion is safe and the tightened benchmark is zero. This is elementary box intersection and classical side-information coding, not a new graph-coloring principle (Witsenhausen, 1976).

If each compared history has at least two $B - \varepsilon$-viable first actions, the chromatic number is also necessary for deterministic policies at regret $\rho < 1/2$, and for deterministic encoders with randomized policies at $\rho < 1/4$. An edge supplies a word w feasible at $B - \varepsilon$ in one history and infeasible even at B in another. In the latter choose a tightened-feasible v with $v_1 \neq w_1$. Set $U(w) = 1, U(v) = 1/2$, and zero elsewhere. A shared first action equal to w_1 gets zero in the latter world; any other first action gets at most $1/2$ in the former. For probability a of the first class, the two regrets are at least $a/2$ and $(1 - a)/2$. Later feedback cannot undo the first action. This proves necessity with the stated constants. The requirement is substantive: Section S.E replaces it only at zero budget slack and for the full bounded-utility class.

At three depths and a fixed partition/spending, the only variable coordinate is $g(3)$, so boxes reduce to integer intervals. Sort them by right endpoint, stab the earliest uncovered endpoint, and repeat. Each newly selected witness interval is disjoint from all previous witness intervals. Every stabbing solution needs a point in each witness; the greedy points cover all intervals. Thus stabbing number, maximum disjoint packing, and the relevant chromatic number coincide. No general-K polynomial coloring algorithm follows from this fact.

B SPARSE-WITNESS COMPRESSION: FULL PROOF AND IMPLEMENTATION

B.1 CANONICAL PARTITIONS TURN DEFERRED COSTS INTO OPTIONAL TIGHT WEIGHTS

Fix $K = 3$ and a future tape of length $R \geq 1$. A suffix-maximum partition is specified by (n_3, n_2, n_1) with sum R. There are $\binom{R+2}{2}$ such partitions. At every nonempty bucket of maximum $m > 1$, its last position is a mandatory depth-m anchor. Every other position can be either tight (depth m) or loose. For a monotone utility, raising a loose action to $m - 1$ preserves both its cost and the suffix maxima and cannot reduce utility. Bucket-one positions all have depth one. Consequently, some optimal word is canonical: anchors are tight, loose optional positions have depth $m - 1$, and a subset T of optional positions is tight.

Let O_Π be the optional positions in partition Π, and let $w(\Pi, T)$ denote this canonical word. Let $D_\Pi = D_{\mathrm{fut}}(w(\Pi, \varnothing))$. Then

$$D_{\mathrm{fut}}(w(\Pi, T)) = D_\Pi + \sum_{i \in T} Q_i. \tag{13}$$

The final future position is never optional. Thus the hypothesis that the first $R - 1$ saving prices, in increasing order, are positive and weakly superincreasing applies to every O_Π. A subsequence retains the inequality. The final saving price can duplicate another price; it is an anchor, not an optional item.

B.2 THE UTILITY ASSUMPTION AND STANDARD EXAMPLES

Write $f_\Pi(T) = U(w(\Pi, T))$. The hypothesis of Theorem 3 is

$$\sum_{i \in T} \big[f_\Pi(T) - f_\Pi(T \setminus \{i\}) \big] \leq a \quad \text{for every } \Pi, T \subseteq O_\Pi. \tag{14}$$

This is a bounded total deletion-loss assumption, related to classical self-bounding properties, not a newly proposed general function class (Vondrák, 2010). For a monotone lattice-submodular utility, the restriction f_Π is monotone submodular because the canonical embedding preserves coordinatewise meet and join. If $T = \{i_1, \ldots, i_n\}$, diminishing returns gives

$$f_\Pi(T) - f_\Pi(T \setminus \{i_j\}) \leq f_\Pi(\{i_1, \ldots, i_j\}) - f_\Pi(\{i_1, \ldots, i_{j-1}\}).$$

Summing telescopes to $f_\Pi(T) - f_\Pi(\varnothing) \leq 1$. Hence $a = 1$, including additive monotone utilities. More generally, for nonnegative weighted sums of monotone conjunctions, each involving at most d coordinates and with weights summing to at most one, a satisfied clause is destroyed by at most d single optional deletions. Interchanging the two finite sums proves $a \leq d$. A completion conjunction over a growing target set need not have constant a.

B.3 THRESHOLD ROUNDING AND THE REPAIR ARGUMENT

The family has a fixed partition/spending, hence a profile $(g_1, g_2, \alpha - v)$ with fixed g_1, g_2, α and $v \in [v_{\min}, v_{\max}]$. The interval endpoints are public, history-independent configuration. Every admitted history has a feasible completion. For $k \geq 1$ let

$$\mathcal{T}_k = \{v_{\min}\} \cup \left(\left\{ \alpha + D_\Pi + \sum_{i \in H} Q_i - B : \max \Pi = 3,\ H \subseteq O_\Pi,\ |H| \leq k \right\} \cap [v_{\min}, v_{\max}] \right). \tag{15}$$

Round v downward to its predecessor $\widehat{v}$ in this set. Decode $\widehat{g} = (g_1, g_2, \alpha - \widehat{v})$. Since $\widehat{v} \leq v$, every decoded-feasible word is physically feasible. The all-shallow completion remains feasible since its liability is the unchanged g_1.

Choose a true optimal canonical word (Π, T). If its maximum is below three, it remains feasible. If $|T| \leq k$, its threshold is either recorded in (15), or is below $v_{\min}$. In both cases $\widehat{v}$ is at least that threshold and the word remains feasible. Otherwise let H be the k largest saving prices in T. The same argument applied to H shows that the decoded capacity

$$\widehat{c} = B - \alpha + \widehat{v} - D_\Pi \quad \text{satisfies} \quad \widehat{c} \geq \sum_{i \in H} Q_i.$$

All weights in $T \setminus H$ precede the smallest member of H in the superincreasing order. Therefore

$$\sum_{i \in T} Q_i - \widehat{c} \leq \sum_{i \in T \setminus H} Q_i \leq \min_{i \in H} Q_i.$$

Deleting *any* one member of H makes the word feasible under the conservative profile, without deleting an anchor or changing the partition. Equation (14) supplies a member whose deletion loses at most a/k. The decoded optimum is no worse than that repaired word. Thus safety holds at the original budget, with no resource augmentation or budget tightening, and utility regret is at most a/k. If $k \geq R - 1$, every optional subset has size at most k, so regret is zero. This proves the approximation assertion of Theorem 3.

B.4 A LITERAL FIXED-LENGTH MESSAGE WITH NO EXPONENTIAL CODEBOOK

Let $k' = \min(k, R - 1)$ and $w = \lceil \log_2(R + 1) \rceil$. Instead of storing a rank in the potentially large threshold set, send a witness for its predecessor. The all-zero message means $v_{\min}$. Every other valid message has a one-bit flag, the two partition lengths (n_3, n_2), and k' optional-position indices in increasing order, padded by the sentinel R. Each integer field has exactly w bits. The message has

$$b_{\text{lit}} = 1 + (k' + 2)w \tag{16}$$

bits, including the flag. The decoder reconstructs D_Π from public prices and computes the threshold from the witness. Invalid padded strings need not occur; the implementation rejects them. In particular, no high-precision value of v is hidden in a parameter, a cache, a filename, or a private codebook. The price tape and $\alpha, B, v_{\min}, v_{\max}, k$ are common across all histories.

To find the best witness, enumerate partitions. Within each maximum-three partition, maximize the mass of at most k' optional items under capacity $B - \alpha + v - D_\Pi$. The descending greedy rule is exact for positive weakly superincreasing weights: if the largest item fits and $k' > 0$, all subsets omitting it have mass at most its weight. Thus a maximum-mass solution can include it; recurse on residual capacity and one fewer slot. If it does not fit, every feasible subset omits it. Induction proves the rule, including equality cases. Taking the largest resulting threshold above $v_{\min}$ gives exactly the predecessor in (15). The algorithm visits $O(R^2)$ partitions and spends $O(R)$ arithmetic per partition after one $O(R \log R)$ public sorting step. Encoding uses $O(R^3)$ arithmetic and $O(R)$ working entries, not an enumerated threshold table.

B.5 POLYNOMIAL OPTIMIZATION OF THE DECODED MONOTONE UTILITY

We spell out the standard superincreasing structure rather than hiding an exponential optimizer. For n optional items in decreasing weight order, greedily compute the full-cardinality maximum feasible bit vector θ. The feasible vectors are exactly the lexicographic initial segment ending at θ (Gupte, 2016, Section 2, Proposition 1). Here is a direct proof adequate for the present use. If a vector first exceeds θ at position j, its matching larger prefix leaves less capacity than item j; it is infeasible. A vector first below θ at j replaces that item by smaller items whose total weight is no greater, so it is feasible.

Every feasible vector is coordinatewise dominated by either θ, or one of the vectors that match its larger prefix, turn a selected $\theta_j = 1$ into zero, and set all smaller positions to one. There are at most $n + 1$ such candidates, all feasible. Evaluating a monotone utility only on these candidates gives its exact optimum for that partition, with no submodularity assumption needed for this optimization step. Repeat over all $O(R^2)$ partitions and apply the appropriate liability cap for their maximum. There are $O(R^3)$ utility-oracle calls. The delivered implementation explicitly constructs each length-R word, giving $O(R^4)$ bookkeeping/arithmetic in addition to oracle evaluation, and $O(R)$ working entries beyond the oracle and public input. These are not constant-time hardware claims. Encoding prices and rewards as rational numbers charges their arithmetic bit lengths; unrelated input denominators can require a common denominator with length proportional to the sum of their bit lengths. All bounds remain polynomial in the explicitly represented input and R, k, but not independent of numerical precision.

C SAME-TAPE LOWER BOUNDS AND THE REWARD-CLASS SEPARATION

C.1 LEGAL POSITIVE DYADIC HISTORIES AND TASKS

For precision $p_{\mathrm{prec}} \geq 1$, let $\delta_{p_{\mathrm{prec}}} = 2^{-p_{\mathrm{prec}}}$. Each of two consecutive past buckets has prices $\ell = Q$ equal to

$$(\delta_{p_{\mathrm{prec}}}, 2\delta_{p_{\mathrm{prec}}}, \ldots, 2^{p_{\mathrm{prec}}-1}\delta_{p_{\mathrm{prec}}}, \delta_{p_{\mathrm{prec}}}),$$

with zero fees. The last position is its anchor and the total mass is one. Each anchored support is $\delta_{p_{\mathrm{prec}}}\{1, \ldots, 2^{p_{\mathrm{prec}}}\}$. Set $T_2 + T_3 = 1 + \delta_{p_{\mathrm{prec}}}$; then $v_i = T_2 = i\delta_{p_{\mathrm{prec}}}$ for $1 \leq i \leq 2^{p_{\mathrm{prec}}}$. Equation (12) gives

$$C = 4 + \delta_{p_{\mathrm{prec}}}, \qquad g_i = (C, C, C + 1 - v_i).$$

Optional positions in bucket two are tight according to the binary expansion of $i - 1$; complementary optional positions in bucket three are tight. This explicitly constructs every history on the same fixed price tape.

For $s \geq 0$, let $R = s + 2$, $\delta_s = 2^{-s}$, and take future prices

$$\ell = Q = (1, \delta_s, 2\delta_s, \ldots, 2^{s-1}\delta_s, \delta_s), \qquad B = C + 6. \tag{17}$$

The middle list is empty at $s = 0$. First action three followed by shallow actions costs $g_i(3) + 4 \leq C + 5 < B$, so all initial masks are full. Every listed prefix is reachable without violating B, since its all-shallow future is feasible and the physical cost is prefix-monotone.

Let S contain the last future position, exclude the first, and have saving weight T. A word with first action three and depth three on all of S has minimum standalone cost $5 + T$: the last requested depth three makes all suffix maxima three, the total price mass is two, and the tight weight is $1 + T$. Omitting the early request gives minimum $4 + T$. Coordinatewise monotonicity proves minimality. Both requests therefore fit at B exactly when $T \leq v$; waiting fits for every $T \leq 1$ and every $v > 0$.

C.2 THE ORDERED-BIT INFORMATION LEMMA

Suppose m ordered hidden states have common predecision telemetry. For every cut between consecutive selected states there is a legal task, identical throughout that cut, under which a wrong binary first-action class loses at least $\gamma > 0$. Select 2^d states in order, $d = \lfloor \log_2 m \rfloor$, and index them by a uniform $X \in \{0, 1\}^d$. Given $X_{<j}$, the decoder chooses the cut between the two corresponding halves. The first-action class estimates X_j with error at most ρ/γ under a worst-case regret guarantee ρ.

Let Z be history-independent public randomness and H the entire b-bit message, allowing private encoder randomness. The binary error-entropy inequality and the chain rule imply

$$b \geq H(H \mid Z) \geq I(X; H \mid Z) \tag{18}$$

$$= \sum_{j=1}^{d} I(X_j; H \mid Z, X_{<j}) \geq d\,[1 - h_2(\rho/\gamma)]\,, \qquad \rho/\gamma < 1/2. \tag{19}$$

A randomized actor cannot beat its posterior-optimal deterministic bit estimator. The additional prefix $X_{<j}$ is decoder side information in this reduction, not an uncharged part of the controller's resource state. The actual task chosen for each cut does not depend on the unknown suffix of X. This is the standard augmented-index information argument (Chakrabarti and Kondapally, 2011); the contribution is the fixed-price allocation witness, not the entropy inequality.

C.3 TWO-TARGET ADDITIVE WITNESSES GIVE A LOGARITHMIC LOWER BOUND

Assume $p_{\mathrm{prec}} \geq s \geq 1$. Use ordered states

$$v_0 = \delta_s, \qquad v_{j+1} = (1 + 2^j)\delta_s \quad (0 \leq j < s).$$

These are all legal multiples of $\delta_{p_{\mathrm{prec}}}$. For cut j, request the future optional position of weight $2^j\delta_s$ and the final anchor of weight δ_s. Reward first-action-three by $1/5$ and each of these two later depth-three choices by $2/5$. This is monotone additive and bounded by one. Its threshold is $T_j = (1 + 2^j)\delta_s$.

For $v \geq T_j$, all three rewards are feasible, and waiting loses at least $1/5$. For $v < T_j$, waiting earns $4/5$, whereas first choosing three precludes completing both targets and earns at most $3/5$. Thus the wrong first class loses at least $1/5$ in every world on the corresponding side, regardless of subsequent exact guards or spending. Equation (19) with $m = s + 1$ gives

$$b_{\mathrm{add}}(\rho) \geq [1 - h_2(5\rho)]\lfloor \log_2(s+1) \rfloor, \qquad 0 \leq \rho < 1/10. \tag{20}$$

For each fixed $0 < \rho < 1/10$, take $k = \lceil 1/\rho \rceil$ in the sparse-witness code. The first $R-1$ saving prices of (17) are superincreasing in increasing order. Theorem 3 gives $O_\rho(\log R)$ bits for all monotone lattice-submodular utilities. Additive utilities are a subclass, so the lower and upper bounds match for both classes. This argument uses zero budget slack throughout.

C.4 Completion witnesses retain a linear lower bound

For any anchored target set of weight $T = j\delta_s$, use

$$U_S(w) = \tfrac{1}{3}\mathbf{1}\{w_1 = 3\} + \tfrac{2}{3}\mathbf{1}\{w_i = 3 \text{ for every } i \in S\}.$$

Every grid weight $j\delta_s$, $1 \leq j \leq 2^s$, is realized by selecting the optional binary weights of $(j-1)\delta_s$ plus the final anchor. If $v_- < T \leq v_+$, the richer history can earn one; waiting loses $1/3$. In the poorer history waiting earns $2/3$, but choosing three first prevents completion and gives at most $1/3$. Hence the first-action error costs at least $1/3$. For $p_{\mathrm{prec}} \geq s$, the full family has at least 2^s ordered separable states. Equation (19) gives $\Omega_\rho(s)$ bits at fixed $\rho < 1/6$; the exact rank code below uses at most $s + 1 = R - 1$ bits. This is the retained constant-regret completion reduction, now contrasted with the additive/submodular upper and matching lower bounds.

C.5 One fixed action-histogram fiber preserves the separation

Set $p_{\mathrm{prec}} = s \geq 2$, $d = \lfloor s/2 \rfloor$, and restrict histories to

$$\mathcal{I}_d = \{i \in \{1, \ldots, 2^s\} : \mathrm{popcount}(i-1) = d\}.$$

There are $\binom{s}{d}$ histories. In every one, the numbers of actions at depths $(1, 2, 3)$ are exactly $(s, d+1, s-d+1)$. Time is $2s+2$, last depth is two, the entire past suffix-maximum partition is $3^{s+1}2^{s+1}$, spending is C, and the initial guard is full. These observations may all be public without changing the lower bounds.

For completion utilities, any adjacent v values in this fiber have a legal separating grid threshold. Thus

$$b_{\mathrm{comp}}(\rho) \geq [1 - h_2(3\rho)] \left\lfloor \log_2 \binom{s}{d} \right\rfloor = \Omega_\rho(s).$$

For additive utilities, put $a_0 = 2^{d-1} - 1$ and select

$$i_j = 1 + a_0 + 2^j, \qquad j = d-1, \ldots, s-1.$$

Each $i_j - 1$ has exactly d one-bits. For two consecutive representatives $j-1, j$, the two-target threshold $T_j = (1 + 2^j)\delta_s$ satisfies

$$i_{j-1}\delta_s < T_j \leq i_j\delta_s,$$

since $a_0 < 2^{j-1}$. It also separates all earlier from all later representatives. There are $s - d + 1$ representatives, so

$$b_{\mathrm{add}}(\rho) \geq [1 - h_2(5\rho)]\lfloor \log_2(s - d + 1) \rfloor.$$

The same sparse-witness upper code remains valid for this restricted family. Thus the logarithmic–linear separation occurs *inside one fixed public counter class*, not by comparing different counter values. At zero regret the restricted fiber's exact count is $\binom{s}{d}$, rather than the full family's 2^s; the positive-regret order separation is unchanged. The central binomial coefficient is at least $2^s/(s+1)$ by averaging over all $s+1$ binomial coefficients, which proves the stated linear lower order without an asymptotic approximation.

W A MATCHING LAW ACROSS THE MPH HIERARCHY

This section proves Theorem 1. The MPH hierarchy is prior (Feige et al., 2015). We use its supporting hypergraphs to identify a matching resource-information order, with constants uniform in rank at fixed regret. The underlying physical tape, entire conditional history family, and disclosure contract are identical for all ranks. All bounds are on additional fixed-length handoff bits. They do not count the independent utility's representation, the physical ledger, or subsequent workspace.

W.1 THE COMMON PHYSICAL FAMILY AND REWARD CLASSES

Let $s \geq 2$, $R = s + 2$, $h = \lfloor s/2 \rfloor$, and $\delta = 2^{-s}$. For $u \in \{0, \dots, 2^s - 1\}$ with h one-bits, let u_j be its bit at position j. On the first past bucket, use depth one if $u_j = 1$ and depth three otherwise, followed by a depth-three anchor. On the second bucket, use depth two if $u_j = 1$ and depth one otherwise, followed by a depth-two anchor. In each bucket the optional prices are $\delta, 2\delta, \dots, 2^{s-1}\delta$ and the anchor price is δ; $\ell = Q$ and fees are zero. Thus the tight masses are $T_2 = (u + 1)\delta = v$ and $T_3 = 1 + \delta - v$. The accounting identities give

$$C = 4 + \delta, \qquad g = (C, C, C + 1 - v).$$

All histories have histogram $(s, h + 1, s - h + 1)$, time $2s + 2$, last depth two, and suffix partition $3^{s+1}2^{s+1}$. The common future tape and budget are

$$\ell = Q = (1, \delta, 2\delta, \dots, 2^{s-1}\delta, \delta), \qquad B = C + 6. \tag{21}$$

The full initial guard and legal reachability follow as in Appendix C. In particular, the histories do not differ in any of the public counters used here. Their count is $\binom{s}{h}$.

A PH-d function is $g(S) = \sum_{\emptyset \neq E \subseteq S,\, |E| \leq d} c_E$, with $c_E \geq 0$. An MPH-d function is the pointwise maximum of a finite collection of PH-d functions. Normalization here means $f([R]) \leq 1$. We allow arbitrary collections and nonnegative edge weights; the algorithm may evaluate the resulting intact utility after handoff. Define $U_f(w) = f(\{i : w_i = 3\})$. This is a monotone word utility. Let $b_d(\rho)$ be the least number of bits allowing almost-sure safety and worst-case expected regret at most ρ for every such utility and every history in the common family. The encoder knows the declared class and ρ, but not the later utility. Shared random seeds are history independent; private randomization is also permitted.

W.2 DELETION BOUND AND THE CONSTRUCTIVE UPPER ORDER

Fix S and a supporting PH-d component g attaining $f(S)$. Since $f(S \setminus \{i\}) \geq g(S \setminus \{i\})$,

$$\sum_{i \in S} \big[f(S) - f(S \setminus \{i\})\big] \leq \sum_{i \in S} \big[g(S) - g(S \setminus \{i\})\big] \tag{22}$$

$$= \sum_{E \subseteq S} |E| c_E \leq d\, g(S) = d\, f(S) \leq d. \tag{23}$$

This is the established MPH deletion inequality: apply Ezra et al. (2019, Lemma 6.2 of the 2018 preprint) to the restriction of f to S. We include the short proof for completeness, not as a new self-bounding principle. A supremum representation gives the same inequality by approximating a supporting component. Deleting an optional tight position in a canonical word either removes its one depth-three activation or leaves the activation set unchanged. Anchors are not deleted. Consequently (7) holds with $a = d$ on every canonical partition.

The first $R - 1$ saving prices in (21), sorted increasingly, are superincreasing. Set $k = \min(R - 1, \lceil d/\rho \rceil)$. Theorem 3 and the enumerative packet of Appendix N.3 give zero regret when $k = R - 1$, and at most $d/k \leq \rho$ otherwise. There are at most $\binom{R+2}{2}$ partitions and at most $R - 1$ optional positions per maximum-three partition. Thus an achievable message alphabet is bounded by

$$A_{R,k} = 1 + \binom{R + 2}{2} \sum_{j=0}^{k} \binom{R - 1}{j}. \tag{24}$$

Unused packet indices can be padded. Alternatively, the exact combinatorial rank of u within the common histogram needs $\lceil \log_2 \binom{s}{h} \rceil$ bits and recovers the exact profile. Both descriptions have effective rank/unrank algorithms. The exact profile can use the same monotone SI actor. Therefore

$$b_d(\rho) \le \min\left\{ \left\lceil \log_2 \binom{s}{h} \right\rceil, \lceil \log_2 A_{R,k} \rceil \right\}. \tag{25}$$

These are literal achievable counts, not an assertion that either is the smallest finite alphabet.

For $1 \le k \le (R-1)/2$, the binomial-sum inequality gives $\sum_{j=0}^{k} \binom{R-1}{j} \le (e(R-1)/k)^k$. Taking logarithms in (24) gives $O(\log R + k \log(eR/k))$ bits. At fixed ρ, with $k = \lceil d/\rho \rceil$ in this range, this is $O_\rho(d \log(1 + R/d))$; the additive $\log R$ is absorbed for $d \ge 1$. If $k > (R-1)/2$, then $d = \Omega_\rho(R)$ for all sufficiently large R, and the exact $O(R)$ alternative has the same required upper order. Enlarging the constants handles the remaining finite horizons. This proves the upper bound uniformly in d and R at fixed ρ.

W.3 A SHARPER FINITE ALPHABET FOR ACTIVATION REWARDS

The count in (24) also covers plans that are unnecessary for the activation-only utilities used here. For a nonempty activation set S, let $j = \max S$. A cheapest word with this activation set uses depth two at unactivated positions before j and depth one afterwards. This preserves every MPH-d reward. It needs only one depth-three bucket, with last anchor j. With at most k optional activations, one endpoint and all such witnesses number at most

$$1 + \sum_{j=0}^{R-1} \sum_{i=0}^{\min(k,j)} \binom{j}{i} = \sum_{i=0}^{\min(k+1,R)} \binom{R}{i}.$$

Positions are indexed from zero in this count.

For the tape (21), write P_j for the price prefix through j. The total future price is two, and the cheapest word above costs $2 + P_j + Q(S)$. Its feasibility threshold is consequently $v \ge P_j + Q(S) - 3$. A positive threshold must include the first position (price one). Its last anchor must be either the largest dyadic position or the final tiny anchor: earlier prefixes have $P_j < 3/2$, so even activating every position gives $2P_j - 3 < 0$. Without the first position, the threshold is nonpositive at either remaining anchor as well.

The first position uses one optional slot. At the final anchor, thresholds are $(t+1)\delta$, where t selects at most $k-1$ of the s dyadic bits. At the largest-dyadic anchor, they are $(2^{s-1} + t - 1)\delta$, with at most $k-1$ of the smaller $s-1$ bits. Removing duplicate and out-of-range thresholds can only reduce the count. Thus the following explicit alphabet is sufficient:

$$A^{\mathrm{act}}_{s,k} = 1 + \sum_{i=0}^{\min(k-1,s)} \binom{s}{i} + \sum_{i=0}^{\min(k-1,s-1)} \binom{s-1}{i}. \tag{26}$$

Round v down to the predecessor among these thresholds and the public lower endpoint. Words with at most k optional activations retain their feasibility. For a larger optimum, retain its k largest optional saving prices. The original superincreasing repair deletes any one of them to restore conservative feasibility; the anchor stays fixed and the canonical word remains in the reduced class. Equation (23) then supplies a deletion losing at most d/k. This proves

$$b_d(\rho) \le \min\left\{ \left\lceil \log_2 \binom{s}{h} \right\rceil, \lceil \log_2 A^{\mathrm{act}}_{s,k} \rceil \right\}, \qquad k = \min(R-1, \lceil d/\rho \rceil). \tag{27}$$

The result is an achievable message count for this structured family. It does not assert a new running-time bound for computing the merged threshold index, or apply the activation-only reduction to arbitrary lattice utilities. At $R = 258$, $\rho = 1/20$, the upper counts are 96, 155 and 226 bits at ranks one, two and four, respectively.

W.4 LOWER BOUND FOR RANKS AT LEAST TWO

Let

$$q = \min(d-1, h), \qquad m = h - q.$$

For every q-element set $J \subseteq \{m, \dots, s-1\}$ define

$$r_J = \sum_{j \in J} 2^j, \qquad u_J = r_J + 2^m - 1, \qquad v_J = (u_J + 1)\delta. \tag{28}$$

The q high bits and m low padding bits are disjoint, so u_J has exactly h one-bits. These are distinct legal histories inside the same public histogram fiber. Their number is

$$L = \binom{s-m}{q} = \binom{s-h+q}{q}.$$

Sort them by r_J. Distinct r_J values differ by at least 2^m. For the threshold $T_J = (r_J + 1)\delta$, every earlier J' satisfies

$$v_{J'} = (r_{J'} + 2^m)\delta \le r_J\delta < T_J \le (r_J + 2^m)\delta = v_J.$$

Thus this threshold separates all earlier representatives from J and all later ones. This padding argument is what keeps the full public histogram fixed while allowing a bounded-width query.

Let S_J consist of the final future anchor and the q optional positions with weights $2^j\delta$, $j \in J$. Its weight is T_J, it excludes the first position, and $|S_J| = q + 1 \le d$. Choose the utility

$$U_J(w) = \tfrac{1}{3}\mathbf{1}\{w_1 = 3\} + \tfrac{2}{3}\mathbf{1}\{w_i = 3 \text{ for every } i \in S_J\}. \tag{29}$$

It is PH-d, hence MPH-d, with total weight one. The physical costs of early-plus-completion and waiting-plus-completion are respectively $B + T_J - v$ and $B - 1 + T_J - v$. The latter always fits because $T_J \le 1$ and $v > 0$. Above the threshold, early can earn one and waiting earns at most $2/3$. Below it, waiting can earn $2/3$ and early earns at most $1/3$. Every wrong first-action class loses at least $1/3$. Depth two is in the waiting class. Later exact feedback cannot restore the lost early reward or undo the commitment, so the gap holds even for an otherwise optimal adaptive continuation.

Select 2^n ordered representatives, where $n = \lfloor \log_2 L \rfloor$, and index them by a uniform $Z \in \{0,1\}^n$. Given the previous rank bits $Z_{<j}$, a decoder uses the threshold at the midpoint of that remaining block and predicts Z_j from the actor's first action. The query depends only on the known previous bits, not the unknown suffix. Pointwise regret bounds its error by 3ρ. Write H for the entire handoff message and W for shared randomness, which is independent of Z. Binary Fano and the chain rule yield

$$b_d(\rho) \ge I(Z; H \mid W) = \sum_{j=1}^{n} I(Z_j; H \mid W, Z_{<j}) \tag{30}$$

$$\ge n\,[1 - h_2(3\rho)]. \tag{31}$$

One may replace the actor's randomized estimate by the optimal posterior binary estimator; private coins cannot improve that estimator. The error bound is averaged over independent shared and private coins, and conditional binary Fano uses the same average (or Jensen's inequality). No conditioning on a favorable seed is required. This is the standard ordered/augmented-index information argument (Chakrabarti and Kondapally, 2011), here instantiated by the fixed-price physical cuts. For fully randomized schemes this yields

$$b_d(\rho) \ge [1 - h_2(3\rho)] \left\lfloor \log_2 \binom{s-h+q}{q} \right\rfloor. \tag{32}$$

Here h_2 denotes binary entropy.

If $2 \le d \le h + 1$, then $q = d - 1$ and

$$\log_2 L \ge q \log_2\left(1 + \frac{s-h}{q}\right) = \Omega(d \log_2(1 + R/d)).$$

For example, the first inequality follows termwise from $\binom{a+q}{q} = \prod_{i=1}^{q}(1 + a/i) \ge (1 + a/q)^q$. Here $q \ge d/2$ and $s - h$ is a constant fraction of R. If $d > h + 1$, then $q = h$ and $L = \binom{s}{h} \ge 2^s/(s+1)$, while $d \log_2(1 + R/d) = O(R) = O(s)$. The coefficient $1 - h_2(3\rho) > 0$ at the specified fixed regret. Since $L \ge 2$, flooring its logarithm changes these lower orders by at most a constant factor; finite small s are again absorbed.

W.5 THE RANK-ONE ENDPOINT AND WHAT THE ORDER MEANS

The completion construction with $q = 0$ gives no rank-one lower bound. Instead take $a_0 = 2^{h-1} - 1$ and representatives $u_j = a_0 + 2^j$, for $j = h-1, \dots, s-1$. They all have h one-bits. For consecutive representatives, the threshold $(1 + 2^j)\delta$ separates them. Reward the first depth-three choice by $1/5$, the position of weight $2^j\delta$ by $2/5$, and the final anchor by $2/5$. These additive PH-1 tasks have wrong-first-class loss at least $1/5$, as proved in Appendix C. There are $s - h + 1$ representatives, so

$$b_1(\rho) \geq [1 - h_2(5\rho)]\lfloor \log_2(s - h + 1) \rfloor.$$

For fixed $0 < \rho < 1/10$ this matches the upper logarithmic order. Together with the $d \geq 2$ construction, it proves (5).

In particular, fixed d, $d = \Theta(\sqrt{R})$, and $d = \Theta(R)$ give $\Theta_\rho(\log R)$, $\Theta_\rho(\sqrt{R}\log R)$, and $\Theta_\rho(R)$, respectively. These are worst cases over the declared class; large edge width with negligible total weight need not be hard. The law is not a sharp joint characterization as ρ tends to zero, and its hard family uses $p_{\mathrm{prec}} = s$. MPH rank is one sufficient reward-structure parameter. The ridge deletion certificate does not establish that every ridge objective has small MPH rank or inherits this lower bound.

W.6 WORST-CASE ARBITRARY-PRICE INFORMATION COROLLARY

For clarity, define $\mathfrak{b}_d(R, \rho)$ as the supremum of the minimum required bits over all legal fixed public price tapes at $K = 3$ and finite admitted history families with public fixed past partition and spending. Thus g_1, g_2 are public and only g_3 may be hidden. Each admitted history has a feasible continuation. The same task-after-encoding and later-feedback contract applies, and the supremum imposes no bound on price precision or computation.

The arbitrary-price threshold construction in Appendix Q has, for n positive optional prices,

$$L(n, k) = \sum_{j=0}^{\min(k-1,n)} \binom{n}{j} + \mathbf{1}\{k \leq n\}(n - k + 1)\binom{n}{k}.$$

For $n \leq R - 1$ it is at most $R\sum_{j=0}^{k} \binom{R-1}{j}$. Including all partitions and the endpoint, a valid alphabet has size at most

$$1 + R\binom{R+2}{2}\sum_{j=0}^{k}\binom{R-1}{j}.$$

The relative deletion bound (23) supplies $a = d$; the same choice of k as above therefore yields the logarithmic binomial-sum bound. Independently, rank $B - g_3$ among the distinct standalone costs of all maximum-three future words. There are at most 3^R such costs. A fixed-length index of at most $\lceil \log_2(3^R + 1) \rceil$ bits preserves every such feasibility comparison. Since g_1, g_2 are public, this preserves the whole feasible language and gives an $O(R)$ exact alternative. The entropy argument above now gives

$$\mathfrak{b}_d(R, \rho) = O_\rho(d \log_2(1 + R/d)).$$

Our fixed dyadic tape and common histogram are members of this broader class and supply the matching lower bound. This proves the arbitrary-price part of Theorem 1. It is a worst-case information corollary, not a lower bound on each price tape. Its enumeration need not be efficient and does not solve generic-oracle planning.

Y PRECISION CAPS THE COMPLEMENTARITY HIERARCHY

This section proves Theorem 2 independently of the finite checks. It retains the accounting model and information timing. The extra condition concerns the resolution of the hidden liability over a public range; it does not assert that any hardware word size supplies that condition. The lattice ceiling and positional-number representation are elementary tools. The result combines a precision cap with the MPH rank law and provides reachable witnesses with positive prices and identical public counters.

Y.1 THE PRECISION CLASS AND THE UPPER BOUND

Fix $K = 3$, $R \geq 4$, integer $P \geq 1$, and $1 \leq d \leq R-1$. In each admitted finite conditional history family, the past partition and spending are public, hence g_1, g_2 are known. Require g_3 to lie on a public affine grid of step $\Delta = 2^{-P}$ within a public interval of width at most one. Let $\mathcal{B}_d(R, P, \rho)$ be the supremum, over these instances and their public tapes, of the least fixed-length handoff message supporting all normalized MPH-d rewards. We allow randomized encoders and actors, zero budget slack, and later exact spending/guards, with the same independent-randomness accounting as before. Encoding and planning computation are unrestricted.

There are at most $2^P + 1$ possible grid liabilities in that interval. Their exact index needs at most $P + 1$ bits. It recovers the profile and therefore the entire feasible language. The arbitrary-price part of Theorem 1 independently supplies $O_\rho(d \log_2(1 + R/d))$ bits. Taking the public minimum of the two codes proves the claimed upper order because $P + 1 \leq 2P$.

For a grid quantum Δ and hidden spread H in unnormalized units, the analogous elementary count is $\lfloor H/\Delta \rfloor + 1$. Thus the relevant quantity is numerical range divided by quantum. A scalar's hardware mantissa, a ratio of largest to smallest prices, or the number of price fields alone cannot replace it. If the fixed past ℓ, Q lie on a Δ grid, common fees supply only an affine offset to g_3; a spread bound still has to be checked. Future prices need not obey the same grid for this upper argument. The construction below does put *all* prices on it.

Y.2 A RADIX PALETTE FOR BOUNDED-WIDTH QUERIES

For $d \geq 2$, set $q = \min(d-1, P)$; for $d = 1$, set $q = 1$. Since $d \leq R-1$, in both cases $q \leq R-2$. Define

$$a = \min\left\{ \lfloor P/q \rfloor, \left\lfloor \log_2 \left(1 + \frac{R-2}{q}\right) \right\rfloor \right\}, \quad A = 2^a, \quad t = aq, \quad M = 2^t, \quad \delta = M^{-1}. \tag{33}$$

Then $a \geq 1$, $t \leq P$, and $q(A-1) \leq R-2$. Use an initial future position of weight one. In the middle place one position of weight $jA^g\delta$ for each $g \in \{0, \ldots, q-1\}$ and $j \in \{1, \ldots, A-1\}$. Fill any unused middle positions with weight δ, and finish with a δ anchor. Let W_f be the total future weight. Set $\ell = Q$ to these weights and all fees to zero. Every price is strictly positive, at most one, and a multiple of 2^{-P}. All filler positions are before the final anchor; none precedes the valuable initial choice as an extra observation opportunity.

Every $r \in \{0, \ldots, M-1\}$ has an A-ary representation. Select the one position corresponding to each nonzero digit and include the last anchor. The resulting set S_r has at most $q+1$ members and total weight

$$T_r = (r+1)\delta.$$

When $d \geq 2$, $|S_r| \leq d$; when $d = 1$ and $r \geq 1$, it has exactly two members. The future tape generally is not superincreasing. This palette supplies existence witnesses for the arbitrary-price information claim, not an efficient generic optimization algorithm.

Y.3 POSITIVE-PRICE PAIRS KEEP THE ENTIRE HISTOGRAM FIXED

Each of two past buckets contains the pairs $(2^j\delta + \delta, \delta)$ for $j = 0, \ldots, t-1$, followed by a δ anchor. A bucket has total weight $W_p = 1 + 2t\delta$. For $u \in \{0, \ldots, M-1\}$, the second bucket makes the first member of pair j tight at depth two if bit $u_j = 1$, and the second member otherwise; its other member has depth one. In the first bucket the complementary member is tight at depth three and the other has depth one. The first and second bucket anchors have depths three and two. This yields M distinct actual histories with the same histogram $(2t, t+1, t+1)$, time $4t+2$, last depth two, and suffix partition $3^{2t+1}2^{2t+1}$. All past prices too are in $(0, 1]$ on the allowed grid; no zero-price past padding is needed.

The two tight masses satisfy

$$T_2 = (t+1+u)\delta, \qquad T_3 = 1 + (t-u)\delta, \qquad T_2 + T_3 = W_p + \delta.$$

Direct accounting gives $C = 4W_p + \delta$ and

$$v = (u+1)\delta, \qquad \alpha = 1 + t\delta, \qquad g = (C, C, C + \alpha - v). \tag{34}$$

The hidden spread is $1-\delta<1$ on the allowed grid, so these families belong to the defined precision class. Take $B=C+\alpha+2W_f+1$. All-shallow futures are feasible. First-three followed by shallow actions has cost $g_3+W_f+2\le B$ since $W_f\ge 1$ and $v>0$. Thus every initial viable-action mask is full. Coordinatewise and prefix monotonicity of physical cost prove reachability and safety of every listed past prefix at the same budget.

Y.4 Queries, randomness, and a uniform lower constant

For $d\ge 2$ and cutoff $r\in\{1,\dots,M-1\}$, use

$$U_r(w)=\tfrac{1}{3}\mathbf{1}\{w_1=3\}+\tfrac{2}{3}\mathbf{1}\{w_i=3 \text{ for all } i\in S_r\}.$$

This is PH-d. Because the last position is a depth-three target, the minimum cost of completing both terms is $B+T_r-v$. Waiting and completing the later targets costs $B-1+T_r-v$ and always fits. Thus $u\ge r$ permits value one by choosing three first; waiting loses at least $1/3$. For $u<r$, waiting earns $2/3$ but early three permits at most $1/3$. These statements optimize over all later actions, including adaptive continuations using the permitted feedback.

For $d=1$, the one radix group supplies one nonzero digit position for every cutoff. Give early-three reward $1/5$ and each later target reward $2/5$. This is additive PH-1. Above the threshold, early can earn one while waiting gets at most $4/5$; below it, waiting earns $4/5$ while early gets at most $3/5$. The wrong first class loses at least $1/5$. A conjunction of two targets is never treated as a rank-one valuation.

There are exactly $M=2^t$ ordered states and a constant-loss query for every cut. Applying the conditional binary-Fano argument from Appendix W, without restricting shared or private randomization, gives

$$b_I\ge\begin{cases}[1-h_2(5\rho)]t, & d=1,\\ [1-h_2(3\rho)]t, & d\ge 2.\end{cases}\tag{35}$$

Shared coins are independent of history and conditioned upon. A query at a rank-bit cut depends on preceding rank bits, not the unknown suffix. The expected-regret guarantee bounds the average bit error; there is no demand to realize every query with a single random seed.

For $d\ge 2$, if $q=P$, then $a=1$ and $t=P$. Otherwise $q=d-1\ge d/2$. Each argument under the floor in (33) is at least one, so

$$t\ge\tfrac{1}{2}\min\left\{P, q\log_2\left(1+\frac{R-2}{q}\right)\right\}\tag{36}$$

$$\ge\tfrac{1}{8}\min\{P, d\log_2(1+R/d)\}.\tag{37}$$

The second line uses $R-2\ge R/2$ and $\log_2(1+x/2)\ge\frac{1}{2}\log_2(1+x)$. For $d=1$, $t=\min\{P,\lfloor\log_2(R-1)\rfloor\}\ge\frac{1}{4}\min\{P,\log_2(1+R)\}$. Thus $[1-h_2(5\rho)]/8$ is a common valid lower coefficient at $0<\rho<1/10$. With the upper bound this proves Theorem 2. Constants are loose; no exact finite message optimum is asserted.

Y.5 Interpretation and quantifier limits

At fixed ρ, fixed P caps every rank at $O(P)$. When $P=\Theta(\log R)$, the worst-case order is $\Theta_\rho(\log R)$ for every rank $d\ge 1$; increasing complementarity alone cannot create a linear-in-R memory requirement in that regime. A linear worst-case requirement needs both $P=\Omega(R)$ and $d=\Omega(R)$. Intermediate requirements are cut off by the smaller resource. This explains the role of precision rather than removing it.

The lower tape may depend on (R,P,d). This theorem therefore does not claim that one tape simultaneously attains all capped ranks. The earlier same-tape Theorem 1 has that stronger physical comparison at its stated precision. Neither minimax statement is a pointwise lower bound for every instance: an instance with one history or an early revealing quote may need zero additional bits. Nor does this proof provide a fast arbitrary-price encoder/actor, certify real ML deployment, or resolve joint sharpness as $\rho\to 0$.

AC PREDICTION TASKS WITH COMPACT DESCRIPTIONS AND EASY ORACLES

This extension is a reduction from a known Bayes prediction calculation to the retained resource-information laws. XOR complementarity is classical (Krause and Guestrin, 2005, Example 1); cost-sensitive acquisition of feature sets is also established (Bilgic and Getoor, 2011). The additional consequence here is that the matching resource-bit orders already hold for at most three prediction heads, even when the state-aware optimizer examines at most eight plans. This is a constructed prediction model, not a newly admitted deployment or an independently new information law.

AC.1 INFORMATION TIMING AND THE BAYES IDENTITY

Keep the fixed-price CBR cost, public past partition and spending, hard safety and later feedback of Section 3. Draw a fresh vector $X \in \{-1,1\}^R$ independently of the resource history, with independent coordinates of public means μ_i. Choosing depth three at position i reveals X_i; shallower actions do not. Supports and weights arrive after the resource descriptor but before acquisition. Neither labels nor unacquired feature values are supplied to the actor. Observed values may be used causally, and random coins remain independent.

A task has $1 \le m \le 3$ nonempty supports E_j, $|E_j| \le d$, and weights $w_j \ge 0$ summing to one. It predicts $Y_j = Z_j \prod_{i \in E_j} X_i$, where the independent sign noise satisfies $\Pr(Z_j = -1) = \eta < 1/2$. Loss is the weighted sum of the heads' zero-one errors. It is not joint exact-match loss on the whole target vector. The task specification and distribution are known; learning parity supports from labels is not part of the model.

For a transcript revealing S, an unobserved coordinate retains its prior distribution: nonanticipating acquisition decisions depend only on already observed values and independent coins. Conditional means give

$$\mathbb{E}[Y_j \mid X_S] = (1-2\eta) \prod_{i \in E_j \cap S} X_i \prod_{i \in E_j \setminus S} \mu_i.$$

Hence the conditional weighted Bayes risk is

$$L_\mu(S) = \tfrac{1}{2} - \lambda F_\mu(S), \quad \lambda = \tfrac{1}{2} - \eta, \quad F_\mu(S) = \sum_{j=1}^{m} w_j \prod_{i \in E_j \setminus S} |\mu_i|. \tag{38}$$

The empty product is one. Its value depends on the acquired set, not the signs of acquired values. An adaptive strategy is thus a distribution over feasible words and cannot improve on the best feasible full word when the resource profile is known. Its excess Bayes risk is exactly $\lambda[\max_{w:C(pw)\le B} F_\mu(S(w)) - \mathbb{E}F_\mu(S(W))]$. A non-Bayes predictor can only have greater loss. Here and below normalized excess risk ρ means actual excess risk $\lambda\rho$; the noise floor and this normalization are explicit.

For unbiased coordinates, $F_0(S) = \sum_j w_j \mathbf{1}\{E_j \subseteq S\}$, a PH-$d$ valuation. The corresponding Bayes risk is

$$L(S) = \tfrac{1}{2} - (\tfrac{1}{2} - \eta) \sum_j w_j \mathbf{1}\{E_j \subseteq S\}. \tag{39}$$

More generally write $\nu_i = |\mu_i|$ and expand

$$\prod_{i \in E} [\nu_i + (1-\nu_i)\mathbf{1}\{i \in S\}] = \sum_{T \subseteq E} \left(\prod_{i \in T} (1-\nu_i) \prod_{i \in E \setminus T} \nu_i \right) \mathbf{1}\{T \subseteq S\}. \tag{40}$$

All coefficients are nonnegative, have degree at most d, and sum to one. Subtracting the public empty-set constant makes F_μ a normalized PH-d valuation without changing any regret. The existing upper codes therefore apply to every later task in this prediction class. Their actor need not materialize the polynomial: evaluating the given three products takes $O(R)$ arithmetic. Computation of the general resource encoder remains a separate question.

AC.2 THE MATCHING ORDERS PERSIST IN THIS SMALLER CLASS

Define $b_d^{\mathrm{pred}}(\rho)$ using the common physical family of Theorem 1, but only the unbiased prediction tasks above. Define $\mathcal{B}_d^{\mathrm{pred}}(R, P, \rho)$ analogously to Theorem 2. The bit budget covers all tasks in the class; it need not be required by any single task on its own.

Corollary AC.1 (Three-head prediction laws)**.** *At fixed* $0 < \rho < 1/10$ *and* $0 \le \eta < 1/2$*,*

$$b_d^{\mathrm{pred}}(\rho) = \Theta_\rho(d \log_2(1 + R/d)), \qquad \mathcal{B}_d^{\mathrm{pred}}(R, P, \rho) = \Theta_\rho(\min\{P, d \log_2(1 + R/d)\}).$$

They concern excess Bayes classification risk $\lambda\rho$*, permit randomization, subsequent guard/spending feedback and acquired feature values, and retain the respective public interfaces and worst-case quantifiers of the original theorems.*

Proof. The Bayes identity and PH-d inclusion give the upper bounds. For $d \ge 2$, every hard cut in the original proofs uses two heads: the first position with weight $1/3$ and the later conjunction of at most d positions with weight $2/3$. For rank one, the existing additive query uses three singleton heads with weights $1/5, 2/5, 2/5$. The same physical histories, public counters and price tapes therefore instantiate legal tasks in this smaller class. The wrong-first-action utility gaps are at least $1/3$ and $1/5$ respectively. Equation (38) multiplies them by λ in actual risk. Independent feature values neither reveal the hidden resource profile before the first choice nor improve the best value within a fixed first-action class. The ordered-cut/Fano argument in the retained proofs applies unchanged to normalized risk, giving the claimed lower orders. □

AC.3 FULL-INFORMATION PLANNING NEEDS AT MOST EIGHT CANDIDATES

For unbiased features, enumerate all subsets $J \subseteq [m]$ of heads. Let $A_J = \bigcup_{j \in J} E_j$ and form w^J with depth three on A_J and depth one elsewhere. Evaluate its cost with the supplied exact profile and its actual reward, including heads incidentally completed. For any feasible word w, collect the heads it completes. The corresponding union word is coordinatewise no larger than w, so is feasible by cost monotonicity and has at least its reward. One of the $2^m \le 8$ candidates is therefore optimal. The all-shallow word ensures a candidate exists. This takes $O(R)$ arithmetic at $m \le 3$, with bit precision separately charged. It proves that neither a large task description nor hard full-information optimization is needed for the information lower bound. The hard tasks use at most three R-bit support masks and fixed weights; learning their supports is not required. It does not provide an efficient encoder. For biased features, partially observed heads may have positive value; the eight-candidate guarantee is deliberately not asserted there.

AC.4 BIAS ROBUSTNESS AND A ZERO-INFORMATION BOUNDARY

Fix any public mean vector with $|\mu_i| \le \beta < 1/5$. Every incomplete head has value at most β, hence $0 \le F_\mu(S) - F_0(S) \le \beta$. In each hard query, the correct first class retains at least its old optimum, whereas the wrong class gains at most β. Its gap remains at least $\gamma - \beta$, with $\gamma = 1/5$ sufficient for all ranks. For fixed $0 < \rho < (1/5 - \beta)/2$, Fano's error parameter $\rho/(1/5 - \beta)$ is below $1/2$, preserving both lower orders with constants depending on ρ, β. The PH expansion above retains the upper bounds without an additional bias-error term. Noise multiplies actual gaps by λ; no noise-dependent accuracy improvement or jointly sharp limit is inferred. Conversely, if $|\mu_i| = 1$ for all coordinates, the noise-floor prediction is available without acquisition and zero resource bits suffice. These restrictions distinguish a worst-case prediction family from arbitrary ML data distributions.

AC.5 INTERFACE CONTROLS AND THE ROLE OF COMPETING HEADS

These controls explain the assumptions of Corollary AC.1; they are elementary consequences, not additional novelty claims. They use unbiased features except for the explicit negative example.

One head, unchanged interface. Let E be its known support. At each $i \in E$, acquire at depth three if that action is viable, and otherwise choose depth one; outside E choose depth one. No resource message is used. If the minimal word that acquires all of E is feasible, each of its target actions is viable along this policy, because its remaining suffix witnesses feasibility. The policy then completes

the head and attains the noise floor. If that word is infeasible, coordinatewise cost monotonicity implies that no feasible word can complete E; every plan has Bayes risk $1/2$. The policy remains safe by the viable-action guard and is optimal in either case. Thus even a large single parity interaction does not by itself require resource information. The lower-bound tasks use competing heads and a descriptor that must support every later task.

Task disclosure before encoding. If both parties know the task before the resource message, the encoder can evaluate the 2^m union candidates with its exact profile and send the index of an optimal feasible candidate. Public tie-breaking and candidate ordering give a fixed m-bit packet, at most three bits. The receiver reconstructs a complete safe plan from the task and index. This is a changed information timing, not a smaller code for the original task-independent contract.

Whole-plan queries before commitment. If the actor instead has access to a resource service that answers complete-plan feasibility questions after task disclosure, it can ask about all $2^m \leq 8$ candidates and select the best feasible one. No pre-task resource descriptor is then necessary. Up to eight Boolean answers, the plan requests, the service's exact state, computation and latency are additional resources. This does not assert zero total communication or a free oracle. The original next-action guard does not provide these prospective whole-plan answers.

Bias invalidates the union-only optimizer. For the unit-price prefix $p = (1, 3, 2, 2)$, two unit-price future positions and $B = 14$, the costs of $(3, 3), (3, 1), (1, 3)$ are $15, 13, 14$. Take one head on both features, $\mu_1 = \mu_2 = 1/20$, and $\eta = 0$. The union menu contains only the empty and full supports; only the empty one fits, giving $F_\mu = 1/400$. Acquiring either single feature fits and gives $F_\mu = 1/20$. The union-only optimizer therefore incurs excess Bayes risk $\frac{1}{2}(1/20 - 1/400) = 19/800$. This refutes an extension of that optimizer even at small bias; it does not refute the biased information law, whose upper bound permits a different actor.

Exact interface-control checks. A separate protocol and standalone physical-cost verifier were hashed before execution. At $R = 3, \ldots, 6$, eight declared past histories, three tapes (unit, non-superincreasing, and zero savings with legal fees) and five distinct cost-rank budgets give 480 instances. The verifier enumerates 25,920 full-word costs, checks every nonempty one-head support (13,920 cases), and 110,400 task/instance cases for each of the task-aware packet and whole-plan-query controls. All 73,680 chosen-prefix guards pass. For $R \leq 4$ the multi-head domain includes all combinations of up to three distinct nonempty supports; larger horizons use the frozen cyclic fixtures. The biased counterexample is evaluated with exact fractions. Two fresh processes and a clean archive replay match the scientific JSON. These are dependent finite mathematical checks, not sampled ML outcomes or evidence that the changed interfaces occur in a natural deployment.

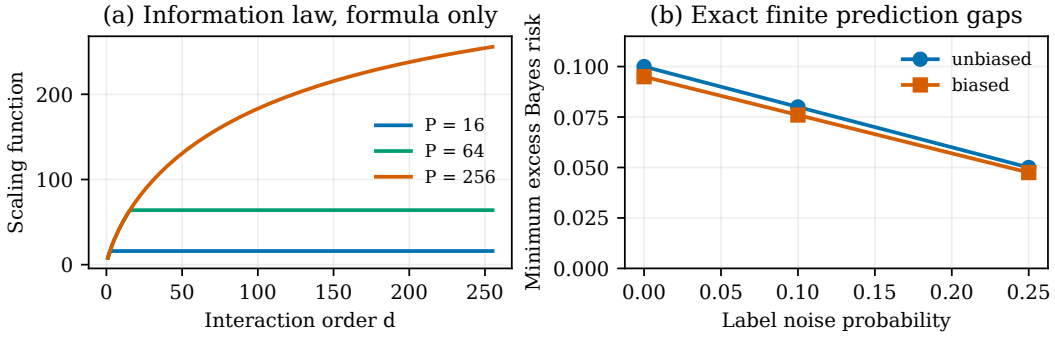

Figure 3: Left: the proven scaling function at $R = 256$, with several precision caps; these are formula values, not measured minimum bits. Right: exact minimum wrong-first-class excess Bayes risk over all 1,704 constructed state/query pairs at each noise level. The biased panel uses alternating feature means $\pm 1/20$. No fitted model or independent statistical sampling is involved.

AC.6 PROSPECTIVE EXACT VERIFICATION

The analytical note, protocol, verifier, unmodified reference construction and environment were hashed before the new runs. All arithmetic is exact rational arithmetic. Independent joint-mass enumeration groups assignments by observed values and sums the smaller of the two label masses; it does not substitute the product formula. For $R = 1, \ldots, 6$ it covers every nonempty head and observation mask, four declared mean profiles and $\eta \in \{0, 1/10, 1/4\}$. The physical suite covers $R = 4, \ldots, 7$, $P = 1, \ldots, 4$, every rank, constructed state and nontrivial cutoff, enumerating all 3^R words with the unchanged physical engine.

Check	Count
Joint-mass Bayes identities	64,008
Positive-hypergraph expansion checks	21,336
Physical word decompositions	357,696
Eight-candidate optima versus exhaustive search	1,704
First-class gaps, each of unbiased and small-bias panels	1,704
Raw risk gaps across three noise rates and both panels	10,224
Deterministic-feature zero-information controls	1,704
Selected-plan prefix guards	10,736

All checks pass. Two fresh processes and an extracted-archive replay reproduce the scientific JSON exactly; timing is kept separate. These dependent enumerations check finite identities, not statistical significance, minimum measured bits, an independently certified proof or a learned predictor. They supply no natural application validation.

II. Constructive algorithms

N FIXED-DEPTH EXTENSION AND COMPACT WITNESS CODING

The main separation uses three depths because that already supplies a single physical family with different information orders. The constructive argument, however, extends to every fixed K. The essential property is that a complete plan consults the old liability only at its *one terminal maximum*. It does not combine all hidden coordinates into a multidimensional knapsack.

More than three depths. At fixed past partition and spending, g_1, g_2 remain public while up to $K-2$ liability coordinates vary. They can be encoded separately without accumulating regret across coordinates.

Theorem 5 (Any fixed number of depths)**.** *Let $K \geq 3$, keep the fixed-partition/spending contract, and impose positive superincreasing saving prices on the first $R-1$ future positions. For monotone utilities with total deletion loss at most a on every canonical partition, a task-independent safe code has regret at most a/k, zero when $k \geq R-1$, and at most*

$$\sum_{M=3}^{K} \left[1 + \big(M - 1 + \min(k, R-1)\big)\lceil \log_2(R+1)\rceil\right] \tag{41}$$

bits. Given the liability profile, for fixed K encoding takes $O_K(R^K)$ arithmetic and optimization takes $O_K(R^K)$ monotone-oracle calls, with $O_K(R^{K+1})$ explicit-word bookkeeping. Bit precision and oracle costs are charged separately.

For each terminal maximum M, round g_M upward to the nearest sparse feasibility threshold. Any complete plan uses only one such coordinate, so its repair loses at most a/k, not Ka/k. The rounded vector may be unrealizable as a past history; it remains a valid conservative certificate for planning a full safe word. Appendix N proves the construction and implementation bounds. The degree grows with K; this does not settle the regime where K is part of the input.

Encoding the witness. Enumerative ranking (Cover, 1973) replaces padded indices by the logarithm of the partition/subset alphabet. Ranking and unranking need no exponential codebook; counting and workspace are charged separately (Appendix N.3). A worked example in Appendix S.K also shows when exact state is cheaper.

N.1 CANONICAL PLANS AT ARBITRARY DEPTH

A nonincreasing suffix-maximum word has at most K consecutive buckets, with lengths $(n_K, \dots, n_1)$ summing to R. The last position of every nonempty bucket of maximum $m > 1$ is an anchor at depth m. Every other position in that bucket is optional: a monotone optimum can use $m-1$ or m, since smaller depths have the same cost as $m-1$ and no larger utility. Positions in the maximum-one bucket have depth one. Thus

$$C(hw(\Pi, T)) = g_h(M(\Pi)) + D_\Pi + \sum_{i \in T} Q_i,$$

where $M(\Pi)$ is the largest nonempty bucket and the final position is never optional. There are $\binom{R+K-1}{K-1} = O_K(R^{K-1})$ partitions. The same deletion-loss assumption is imposed on every canonical set function f_Π. Lattice-submodular utilities into $[0,1]$ still give $a = 1$.

N.2 SEPARATE CONSERVATIVE THRESHOLDS, ONE REPAIR

Fix the past suffix-maximum partition and exact spending. Appendix A shows that $g(1)$ and $g(2)$ are public constants. For every $M \geq 3$, let $\bar{g}_M$ be a public upper bound on $g_h(M)$ throughout the family. Such a bound requires no exponentially sized history table: with the past maxima r_i public, $\sum_{i<|h|}[\ell_i \max(r_i, M) + c_i]$ is valid because the discounts subtracted in the cost are nonnegative.

For each M, form thresholds

$$\mathcal{T}_M = \{\bar{g}_M\} \cup \left\{ B - D_\Pi - \sum_{i \in H} Q_i : M(\Pi) = M,\ H \subseteq O_\Pi,\ |H| \le k \right\}.$$

Decode $\widehat{g}_M$ as the smallest threshold at least the true $g_h(M)$; the endpoint ensures existence. Set $\widehat{g}_1 = g_1$ and $\widehat{g}_2 = g_2$. The resulting vector need not be the liability of any physical past. This is harmless: it is a conservative certificate for the full-word planner, not a claimed reconstructed history or an updateable ledger.

Proof of Theorem 5. Safety follows from $g_h(M) \le \widehat{g}_M$ for every M. An all-shallow completion remains available because g_1 is unchanged. Consider a canonical true optimum in partition Π with maximum M. If $M \le 2$, its old liability is exact. Otherwise, if its optional set T has $|T| \le k$, the threshold for T preserves it exactly. If $|T| > k$, let H be the k largest selected weights. Its threshold yields decoded optional capacity at least $\sum_{i \in H} Q_i$. Superincreasing prices give $\sum_{i \in T \setminus H} Q_i \le \min_{i \in H} Q_i$. Any one deletion from H repairs feasibility, and the deletion-loss inequality supplies one with utility loss at most a/k. Anchors remain, so the same maximum and partition are preserved. The decoded optimizer does at least as well as this plan. There is *no factor K in regret*: only the true optimum's maximum is used in this argument. At $k \ge R - 1$ all optional sets are represented.

For each $M = 3, \dots, K$, encode either the endpoint flag or a witnessing partition, its $M - 1$ lengths $(n_M, \dots, n_2)$ and at most $k' = \min(k, R - 1)$ indices padded by R. Concatenation of these public-width fields gives the displayed bit bound. Maximizing affordable optional mass with at most k items uses the same descending greedy argument as for three depths. Enumerating all partitions takes $O_K(R^K)$ arithmetic operations. Within each decoded partition the lexicographic frontier has at most R members, so optimization takes $O_K(R^K)$ monotone-oracle calls and $O_K(R^{K+1})$ explicit-word bookkeeping. Partitions and candidates can be streamed, using $O(R + Kk + K^2)$ working entries; no exponential static table is assumed. Arithmetic precision and utility-evaluation costs remain separate. The degree grows with K; this is not polynomial uniformly in input K. These bounds take the liability profile as input. Computing that profile from an explicit past and its suffix maxima adds $O(K|h|)$ arithmetic; the encoder is not an online sketch maintaining an exact ledger in the packet bits. □

The same proof and the sequential repair of Proposition S.J.1 give the harmonic regret bound at any fixed K when prefix mass is bounded by β. Outside superincreasing prices we still make no efficient arbitrary-monotone-oracle decoder claim. Dynamic programming by numeric capacity alone does not describe a generic set-value oracle: unlike additive values, that utility need not decompose across partial subsets. Explicit additive tables do admit a different profit-indexed algorithm without a partition factor (Appendix S.S).

N.3 Fixed-length enumeration of legal witnesses

The padded representation wastes invalid combinations of partition lengths and indices. We apply standard enumerative coding (Cover, 1973) to the public set of legal witnesses. This changes the representation, not the rounded liability or the policy menu. Put

$$A_k(n) = \sum_{j=0}^{\min(k,n)} \binom{n}{j}, \qquad N_M = \sum_{\Pi : M(\Pi) = M} A_k(|O_\Pi|).$$

One endpoint label and one label per legal witness require exactly $\lceil \log_2(1 + N_M) \rceil$ fixed bits for this unmerged alphabet. Different witnesses may yield the same threshold; we do not claim optimality after merging such duplicates. There is no prior, variable-length message, hidden numerator, or utility-dependent codebook.

To count the alphabet without enumerating subsets, let b be the number of nonempty buckets at depth at least two and u their total length. There are $\binom{M-2}{b-1}$ choices of their depths (including M), $\binom{u-1}{b-1}$

positive compositions, and $u - b$ optional positions. Thus

$$N_M = \sum_{b=1}^{\min(M-1,R)} \binom{M-2}{b-1} \sum_{u=b}^{R} \binom{u-1}{b-1} A_k(u-b). \tag{42}$$

Order partitions lexicographically, then subsets by cardinality and lexicographic order. Counts A_k give partition offsets; binomial counts rank and unrank a subset without listing the alphabet. The public counting preprocessing is charged explicitly. The compact implementation also evaluates exact binomial coefficients for ranking and unranking; their polynomial bit costs are additional to the padded-layout arithmetic count. Encoder and decoder store arbitrary-precision integer indices whose bit lengths are the reported packet lengths.

At $K = 3$, $R = 34$, the padded counts 19 and 37 bits for $k = 1, 4$ become 14 and 23 bits, with identical decoded liabilities. For $\rho = 0.05, 0.02, 0.01$, taking $k = 20, 50, 100$ moves the first strict crossover with $R - 2$ exact bits from $R = 180, 472, 1126$ to $R = 66, 138, 253$. These crossovers are exact formula evaluations; integer widths can cause recrossings and no monotone crossover behavior is assumed. Compression of the packet does not itself imply lower regret than another descriptor with the same number of bits, lower total RAM, or faster planning.

Q ARBITRARY-PRICE INFORMATION AND GENERIC-DECODER COST

This extension uses the physical CBR contract in Section 3, including its additive utility regret and permission for later guard/spending feedback. It does not replace that contract by a no-feedback scalar-capacity problem. The constructed planner is deterministic and does not need extra feedback, which is permitted by the original contract.

Theorem 6 (Arbitrary saving prices)**.** *Under the fixed-depth contract of Appendix N, allow arbitrary $Q_i \geq 0$. For integer $k \geq 1$ and monotone utilities satisfying* (7)*, a safe task-independent descriptor has regret at most a/k, zero for $k \geq R-1$, using $O_{K,k}(\log(R+1))$ bits. Encoding and optimization time are unrestricted.*

Q.1 THRESHOLD ALPHABET AND PROOF

Fix a canonical partition Π with terminal maximum $M(\Pi)$. Set all optional positions with $Q_i = 0$ tight: this costs nothing and cannot reduce a monotone utility. Include those choices in its base word. Let O_Π^+ be the remaining positive-saving positions, $n_\Pi = |O_\Pi^+|$, and D_Π its base standalone cost. Then the cost of a canonical word is

$$C(pw(\Pi, T)) = g_p(M(\Pi)) + D_\Pi + \sum_{i \in T} Q_i, \qquad T \subseteq O_\Pi^+.$$

The deletion bound remains valid when its nonnegative summands are restricted to positive-saving positions. Define

$$\mathcal{A}_\Pi = \left\{ \sum_{i \in H} Q_i : |H| < k \right\} \cup \left\{ \sum_{i \in H} Q_i + t \min_{i \in H} Q_i : |H| = k,\ 0 \leq t \leq n_\Pi - k \right\},$$

where $H \subseteq O_\Pi^+$ and the second set is empty when $k > n_\Pi$. For $M \geq 3$ take the liability alphabet

$$\mathcal{L}_M = \{\bar{g}_M\} \cup \{B - D_\Pi - \tau : M(\Pi) = M,\ \tau \in \mathcal{A}_\Pi\}.$$

The public upper endpoint $\bar{g}_M$ from Appendix N.2 ensures that upward rounding of $g_p(M)$ is defined. Decode $g(1), g(2)$ exactly and all other coordinates to their least larger alphabet entry.

Proof of Theorem 6. Conservative liability coordinates make every decoded-feasible full word truly feasible. An all-shallow completion remains available because $g(1)$ is unchanged. Canonicalize a true optimum, set zero-saving optional positions tight, and write S for its positive optional set in partition Π. If $|S| \leq k$, its exact sum belongs to $\mathcal{A}_\Pi$, preserving the word. Otherwise let H contain its k largest positive weights and put

$$q = \min_{i \in H} Q_i > 0, \qquad t = \left\lfloor \frac{\sum_{i \in S \setminus H} Q_i}{q} \right\rfloor.$$

Every tail weight is at most q, so $0 \leq t \leq n_\Pi - k$. For $\tau = \sum_{i \in H} Q_i + tq$,

$$\tau \leq \sum_{i \in S} Q_i < \tau + q.$$

The liability threshold $B - D_\Pi - \tau$ is at least $g_p(M(\Pi))$ and hence is no smaller than its upward-rounded value. The decoded optional capacity is therefore at least τ. Deleting any item of H restores feasibility. The additive deletion inequality supplies one with loss at most a/k. Anchors and zero-saving tight positions remain, so the maximum and partition are unchanged. Exact decoded planning does at least as well as this repaired word. Only the true optimum's terminal maximum is used, so the regret is a/k, not Ka/k. If $k \geq R-1$ every optional set is preserved exactly. □

To count witnesses, put

$$L(n,k) = \sum_{j=0}^{\min(k-1,n)} \binom{n}{j} + \mathbf{1}\{k \leq n\}(n-k+1)\binom{n}{k}, \qquad A_M = 1 + \sum_{\Pi : M(\Pi) = M} L(n_\Pi, k),$$

with the second term omitted when $k > n$. An indexed witness, including the upper endpoint, uses at most $\sum_{M=3}^{K} \lceil \log_2 A_M \rceil$ bits. The public order lists a partition, a subset and, where applicable, the integer t; equal thresholds need not have distinct indices in an optimized implementation. At fixed K, k, the number of partitions and witnesses is polynomial in R, so their indices have logarithmic length. Materializing the alphabet is unnecessary for the information statement but finding its predecessor may still be expensive. Direct witness enumeration takes $O_{K,k}(R^{K+k})$ arithmetic operations; no polynomial bound is asserted for exact optimization of an arbitrary-price decoded menu. The separate additive approximation in Appendix S.S does not change this witness-enumeration encoder.

The construction adapts classical large-item filtering and inexpensive deletion repair (Bienstock, 2008; Pritchard, 2011). Its role here is to separate a price assumption needed for the efficient implementation from one needed for the information upper bound. It does not remove the high precision of the matched lower-bound family, prove computational optimality, or establish a new generic knapsack approximation method.

Q.2 Exact physical-engine integration checks

The prospective protocol uses $K \in \{3, 4\}$, $R \in \{3, 4, 6\}$ and histories $(1, K, 2, 2)$ and $(K, K, 1, 2)$, with past $\ell_i = Q_i = 1, c_i = 0$. They have identical past suffix maxima and public $g(1), g(2)$. Future saving tapes are equal positive prices, a mixed tape containing zeros, and positive rational prices; they need not be superincreasing. Future prices obey $\ell_i = Q_i + 1$ and $c_i = (i \bmod 3) - 1$. Four budget offsets and all registered $k \in \{1, 2, 3, R - 1\}$ are retained. Utilities are normalized additive depth, coverage, and a width-two completion indicator, with deletion bounds $1, 1, 2$.

The verifier enumerates every full future word, checks its physical cost decomposition, verifies canonical optimum equality, then compares true and decoded optima. It also checks every prefix-spending guard of each selected word. Two fresh CPU processes reproduce the same exact scientific output.

Check	Count
Full-word physical decompositions	31,518
Canonical deletion inequalities	8,601
Canonical optimum equalities	432
Decoded regret comparisons	1,440
Selected-word prefix guards	6,480

All 1,440 regret comparisons are ties with the true optimum on this panel. This verifies integration without demonstrating tightness of the loss bound or a compression advantage. In particular, the two histories can be indexed exactly with one bit, while the generic witness bounds on these small panels range from 4 to 19 bits. These are finite mechanism tests, not ML samples or evidence of superiority over an exact representation. No GPU or natural workload is evaluated.

AJ EXACT FEASIBILITY FROM PUBLIC RESOURCE FEATURES

This appendix proves Theorem 4 in the unchanged three-depth physical ledger. It adds a supplied public structure assumption; it does not replace the unrestricted-price information law. The reduction to a public rectangular cost superset is the construction analyzed here. Low-dimensional integer optimization is classical (Hirschberg and Wong, 1976; Lenstra, 1983); neither its arithmetic nor a general knapsack algorithm is claimed as a new contribution.

AJ.1 INPUTS, INFORMATION AND THE FIXED-WIDTH PACKET

The receiver knows the legal future tape, budget B, liabilities g_1, g_2, and a public bound $\bar{g} \geq g = g_3$. Every admitted history has a feasible all-shallow completion. Rates $a \in \mathbb{Q}_{>0}^{s_f}$ and integer vectors e_i, f_i, q_i satisfy the exact public equalities

$$\ell_i = \langle a, e_i \rangle, \qquad c_i = \langle a, f_i \rangle, \qquad Q_i = \langle a, q_i \rangle.$$

Coordinates may be signed. Input validation checks these equalities and $\ell_i \geq Q_i \geq 0$, $\ell_i + c_i \geq 0$ separately. The rates, features and their encoding are fixed public inputs before the hidden history and subsequent task. Finding a favorable factorization is not supplied by the theorem. Rates need not be rationally independent, so this is integer description structure, not a claim about rational matrix rank.

Let $\mathcal{P} = \{(1,0),(2,0),(2,1),(3,0),(3,1)\}$. Define

$$\begin{aligned} l_{it} &= \min_{(m,r)\in\mathcal{P}} (me_{it} + f_{it} - rq_{it}), \quad & u_{it} &= \max_{(m,r)\in\mathcal{P}} (me_{it} + f_{it} - rq_{it}), \\ L_t &= \sum_{i=0}^{R-1} l_{it}, & M_t &= \sum_{i=0}^{R-1} (u_{it} - l_{it}), & D_0 &= \langle a, L \rangle. \end{aligned} \tag{43}$$

For any word w, the pair consisting of its suffix maximum and rebate indicator is in $\mathcal{P}$ at every position. Subtracting the coordinate minima therefore gives

$$D_{\mathrm{fut}}(w) = D_0 + \langle a, x(w) \rangle, \qquad x(w) \in \mathcal{X} := \prod_{t=1}^{s_f} \{0, \ldots, M_t\}.$$

The box need not equal the realizable cost set. It can contain negative costs or many unrealizable points; only containment is used. Define the public alphabet

$$\mathcal{A} = \{\bar{g}\} \cup \{B - D_0 - \langle a, x \rangle : x \in \mathcal{X}\}, \qquad \widehat{g} = \min\{v \in \mathcal{A} : v \geq g\}.$$

The endpoint makes the minimum well defined. Assign label zero to $\bar{g}$ and label $1 + \sum_{t=1}^{s_f} x_t \prod_{j=t+1}^{s_f} (M_j + 1)$ to a box point. Choose a deterministic witness if costs coincide. The fixed logical message length is

$$b_{\mathrm{feat}} = \left\lceil \log_2 \left(1 + \prod_{t=1}^{s_f} (M_t + 1) \right) \right\rceil. \tag{44}$$

The implementation stores $\lceil b_{\mathrm{feat}}/8 \rceil$ bytes, rejects wrong lengths, unused labels and nonzero padding, and decodes the label using public integers. This packet is the private history-dependent message, not all storage at either endpoint.

AJ.2 EXACTNESS AND SAFETY

Proof of the representation part of Theorem 4. Consider a truly feasible word with maximum three. Its cost gives $B - D_{\mathrm{fut}}(w) \in \mathcal{A}$ and $B - D_{\mathrm{fut}}(w) \geq g$. By minimality,

$$g \leq \widehat{g} \leq B - D_{\mathrm{fut}}(w).$$

It is still feasible under the decoded liability. Conversely, $\widehat{g} + D_{\mathrm{fut}}(w) \leq B$ implies $g + D_{\mathrm{fut}}(w) \leq B$ because $\widehat{g} \geq g$. Words with maxima one or two use the unchanged public liabilities. Thus every word, not just an optimal or monotone normal form, has the same feasibility status. Maximizing any supplied $U : \{1,2,3\}^R \to [0,1]$ over this identical set gives the same value. An approximate planner retains its own approximation loss.

Legal spending is nondecreasing under execution. An old position whose suffix maximum rises by at least one gains at least ℓ_i and can gain a discount of at most $Q_i \le \ell_i$; otherwise its old cost is unchanged. A newly appended position costs at least $\ell_i + c_i \ge 0$. Hence a complete feasible word also passes every prefix budget bound in the original ledger.

If the absolute value of every coordinate is at most H, then $|me_{it} + f_{it} - rq_{it}| \le 5H$. Consequently $M_t \le 10RH$, and (44) is $O(s_{\mathrm{f}} \log(RH+1))$ for $R, H \ge 1$. The guarantee imposes neither an MPH rank bound nor monotonicity of U. □

This is exact feasibility at the same budget, without a margin assumption. Utilities that depend on other hidden history, unmodeled outcomes or a different physical ledger require a different information contract. The bound is not a minimum-message theorem: a smaller exact grid or even a zero-bit descriptor may be preferable on some inputs.

AJ.3 Two-rate encoding without capacity enumeration

After exact public rational normalization, let the two positive integer rates be A, D and let $C = \lfloor B - g - D_0 \rfloor$ in these normalized units. Encoding reduces to

$$p^* = \max\{Ax + Dy \le C : 0 \le x \le M_1,\ 0 \le y \le M_2,\ x, y \in \mathbb{Z}\}. \tag{45}$$

If $C < 0$ there is no box witness, so use the endpoint. Otherwise select the smaller of $\bar{g}$ and $B - D_0 - p^*$, with the associated label. The floor handles an exact rational capacity; the distributed executable accepts normalized integers only, rejects binary floats, and does not implement an automatic rational-normalization front end.

If $AM_1 + DM_2 \le C$, return the upper corner. Otherwise put $y_{\max} = \min(M_2, \lfloor C/D \rfloor)$. For fixed y, the best coordinate is $x = \min(M_1, \lfloor (C - Dy)/A \rfloor)$. On the saturated region $y \le \lfloor (C - AM_1)/D \rfloor$, the largest admitted y is best. On the remaining interval $y_0 \le y \le y_{\max}$, the value is $C - ((C - Dy) \bmod A)$. One modular minimum, plus comparison with the saturated candidate, therefore solves (45) exactly, returning both coordinates.

Lemma AJ.1 (Euclidean modular minimum)**.** *For integers $n, m \ge 1$, $u, v \in \mathbb{Z}$, an index minimizing $(ui + v) \bmod m$ over $0 \le i < n$ can be found in $O(\log(m+1))$ integer arithmetic operations.*

Proof. First reduce u, v modulo m. If $n = 1$ or $u = 0$, index zero suffices. If $u > m/2$, reverse the progression: set $v \leftarrow (v + u(n-1)) \bmod m$, $u \leftarrow m - u$ and map a returned index i back to $n - 1 - i$. Thus assume $0 < u \le m/2$. Let $J = \lfloor (u(n-1) + v)/m \rfloor$. If $J = 0$, no wrap occurs and index zero minimizes the residues.

Otherwise any strict improvement on residue v occurs at the first index after a wrap $j = 1, \ldots, J$:

$$i_j = \left\lceil \frac{jm - v}{u} \right\rceil, \qquad (ui_j + v) \bmod m = (v - jm) \bmod u.$$

Between wraps the residues increase. The J candidate wrap residues form a smaller modular-minimum instance with length J, modulus u, slope $-m$ and intercept $v - m$, indexed by $j - 1$. Compare its minimum with v and recover the witness using i_j. Each recursive modulus is at most half its predecessor. A constant number of integer operations per level proves the bound. The code stores recovery frames explicitly instead of relying on a language recursion limit. □

This recurrence is a specialization of classical two-variable integer optimization (Hirschberg and Wong, 1976); the proof here specifies the implemented bounded predecessor, including witnesses. It does not use an array indexed by capacity, enumerate box points, or enumerate action words or suffix partitions.

Charged computation and storage. Public exact validation and box setup use $O(Rs_{\mathrm{f}})$ arithmetic operations. At $K = 3$, raw past extraction is the single pass

$$g_3 = 3 \sum_{i<|p|} \ell_i + \sum_{i<|p|} c_i - \sum_{i<|p|} Q_i \mathbf{1}\{p_i < 3\},$$

and costs $O(|p|)$ operations. After setup the two-rate encoder uses $O(\beta)$ arithmetic operations. Here β bounds the bit lengths of normalized integers and intermediate sums/products, including budgets, coordinate bounds and the hidden liability; it is not just the rate precision P. With elementary quadratic-time integer arithmetic a conservative end-to-end bound is

$$O((|p| + R)s_{\mathrm{f}}\beta^2 + \beta^3),$$

plus exact rational normalization when needed. Persistent codec state uses $O(s_{\mathrm{f}})$ integers; the temporary Euclidean stack uses $O(\beta)$ frames of $O(\beta)$ bits each. Input storage, public metadata, task representation and planner workspace are separate. Fixed $s_{\mathrm{f}} > 2$ admits classical fixed-dimensional integer optimization in $f(s_{\mathrm{f}})\,\mathrm{poly}$(input bit length) time (Lenstra, 1983); only $s_{\mathrm{f}} = 2$ is implemented and timed here.

Representing arbitrary prices with a unit rate and enormous integer coordinates increases $\log H$. Likewise discovering an economical basis is outside the algorithm. High precision in a few supplied rates is therefore distinct from arbitrary high-precision price diversity. No universal fast encoder or fast arbitrary-utility planner follows.

AJ.4 A NONZERO-INFORMATION WITNESS

The scale tests alone do not establish that their message is necessary. The following separate witness rules out the explanation that every structured instance is already solvable with zero bits. It asserts positive information, not optimality of the feature packet.

For $P \geq 8$ form a bucket with saving prices $(2^j + 1, 1)_{j=0}^{P-1}$ followed by a price-one anchor. Use two copies, with $\ell = Q$ and zero fees. For bit u_j of $0 \leq u < 2^P$, choose upper-bucket actions $(1, 3)$ if $u_j = 1$ and $(3, 1)$ otherwise; choose lower-bucket actions $(2, 1)$ if $u_j = 1$ and $(1, 2)$ otherwise. Append anchors three and two, respectively. Each bucket has $2P + 1$ positions and total price $W = 2^P + 2P$. The upper and lower tight weights are $2^P + P - u$ and $P + u + 1$. Consequently all these histories share spending $C_* = 4W + 1$, histogram, last depth and suffix partition, with

$$g_1 = g_2 = C_*, \qquad g_3 = C_* + 2^P + P - 1 - u.$$

Set $a = 2^{P-1} + 3$, $b = 2^{P-2} + 5$ and future horizon $R = 8$. Let $Q_i = a + bi$ and $\ell_i = L = (4R + 1)a + 2R(R - 1)b = 4\sum_i Q_i + a$, with zero fees. The common feature vectors are $e_i = (4R + 1, 2R(R - 1))$, $q_i = (1, i)$, $f_i = (0, 0)$. Choose histories $u_{\mathrm{lo}} = 2^P - 1$ and $u_{\mathrm{hi}} = 2^P - 1 - a$. They have $g_{\mathrm{hi}} = g_{\mathrm{lo}} + a$. Write

$$D_{\mathrm{last}} = 3RL - \sum_{i=0}^{R-2} Q_i, \quad T = Q_0 + Q_1 + Q_2, \quad B = g_{\mathrm{lo}} + D_{\mathrm{last}} + T.$$

All three initial actions have a feasible shallow completion in both worlds: even choosing three first costs only $g_{\mathrm{hi}} + (R + 2)L$, which is below B. Thus their initial guard masks coincide.

Reward the activated positions $0, 1, 2, 7$ by weights $1, 2, 2, 2$ divided by seven. If the final position is active, the exact future cost is $D_{\mathrm{last}} + \sum_{i<7:w_i=3} Q_i$; non-tight depths can be one or two without changing it. The low-liability world can activate all four rewarded positions, attaining one. The high-liability world can activate $1, 2, 7$, attaining $6/7$, but cannot afford all four. If it activates position zero, it can include at most one of $1, 2$ when position seven is active, so its reward is at most $5/7$. If position seven is inactive, total available reward is also at most $5/7$. Conversely, skipping position zero limits the low world to $6/7$.

A zero-bit actor has the same probability π of choosing depth three first in both worlds. Even if the full hidden state is revealed after that choice, its worst-case expected regret is at least

$$\max\{\pi/7, (1 - \pi)/7\} \geq 1/14.$$

Hence no zero-bit actor meets regret below $1/14$ on this pair, while the feature code preserves both feasible sets exactly. This is an additive task in the original ledger; no new feedback restriction is imposed. It does not show that 20 bits are necessary or that every large scaling fixture needs nonzero information.

AJ.5 EXECUTED EVIDENCE, CONTROLS AND REPLAY

All counts below describe constructed verification domains, not independent statistical trials or natural ML workloads. The frozen `resource_feature` receipt covers 140 legal tapes of lengths one through seven, with signed fee features, 65,580 complete ternary-word feasibility comparisons and 3,279 independent raw past-plus-future cost checks. The separate `resource_additional` receipt covers 39,366 complete-word comparisons on the witness above for $P = 8, 64, 512$, checks equality of public telemetry and initial guards, and rejects floating-point inputs at the strict interface. Both deterministic result files matched fresh-copy replay.

Future R	Distinct rebates	Feature packet	Exact grid, $P = 64/512/4096$
64	64	21 bits / 3 bytes	64 / 512 / 4096 bits
1,024	1,024	33 bits / 5 bytes	64 / 512 / 4096 bits
10,000	10,000	43 bits / 6 bytes	64 / 512 / 4096 bits

Table 3: All nine executed scaling cases. For zero-based i, features are $e_i = (i + 2, 3)$, $q_i = (i, 1)$, $f_i = 0$. Rate precision changes arithmetic but not the public coordinate widths. The paired-history family has exactly 2^P consecutive liabilities, so its exact offset-grid baseline uses P bits; a general closed unit interval with both endpoints can instead require $P + 1$ bits.

The witness packet uses 20 logical bits at all three tested precisions. At $P = 8$, the eight-bit exact grid wins; at $P = 64, 512$, the feature packet is smaller.

An additional end-to-end case actually reads 16,386 past actions, with 10,000 future positions and 4,096-bit rates, and emits 43 logical bits in six bytes. Original single-run CPU times were 0.0536 seconds for public setup and 0.00334 seconds for history extraction plus encoding. These are engineering feasibility observations, not calibrated latency comparisons or ML speedups. Shared prices, rates, features, budget, public liabilities and the endpoint bound are not included in the six-byte message. The large case checks raw profile extraction and conservative decoding; it does not enumerate its 3^{10000} words. Full feasibility equality there follows from the proof, not exhaustive execution.

Artifact mapping. The unchanged implementation is `resource_feature_codec.py`, with the strict integer interface `resource_feature_codec_v2.py` and arithmetic in `two_price_codec.py`, all under `studies/public_features/code/`. Result and timing JSON are provided separately. The top-level `reproduce.py` runs the `resource_feature` and `resource_additional` suites in fresh directories and compares their deterministic outputs. Finite checks and same-host replay do not replace proof or independent novelty review.

O UTILITY CERTIFICATES AND PUBLIC QUERY ACCOUNTING

Bounded-width monotone DNFs. Suppose a Boolean monotone utility on a canonical optional set is an OR of AND clauses, each using at most d items. When it equals one, choose any satisfied clause. Every item whose deletion would change the utility to zero must belong to that clause. There are at most d such critical deletions; when the utility is zero all losses are zero. Hence the total deletion loss is at most d, irrespective of the number or overlap of clauses. A nonnegative weighted sum of these utilities with total weight at most one also has $a \le d$. Depth-threshold versions inherit the same argument on each canonical partition. This extends the assumption beyond submodularity without confusing an OR of conjunctions with OR coverage. At the other extreme, the AND of all $R-1$ optional items has deletion sum $R-1$ at the full set, so a constant a cannot cover arbitrary monotone objectives.

Finite-output public queries. In the dyadic family, an exact prospective action-three quote $C+4-v$ reveals v and removes the need for a further descriptor. One real-valued query is therefore sufficient, but its precision can carry linearly many bits. More generally suppose additional public telemetry Z is supplied *before the utility*, has at most 2^q possible values, and an additional descriptor has at most 2^b messages. The ordered-bit proof applies to the joint transcript (Z, E), whose entropy is at most $q + b$. Therefore the same-tape bounds become

$$b + q \ge [1 - h_2(5\rho)]\lfloor \log_2(s+1) \rfloor \quad \text{(additive)}, \qquad b + q \ge [1 - h_2(3\rho)]\lfloor \log_2 L \rfloor \quad \text{(completion)}.$$

This identifies where information travels; it is not a characterization of all minimal query primitives. A utility-adaptive query after the handoff changes the one-way reduction and is not covered by this statement.

III. Finite checks and execution scope

F COUNTEREXAMPLES AND ALGORITHMIC LIMITS

A scalar and guard do not always suffice. With three depths, depth and rebate prices $\ell_i = Q_i = 1$ and fees $c_i = 0$ at every past and future position, $p = (1, 3, 2, 2)$ and $p' = (3, 3, 1, 2)$ have spending nine, last depth two, and suffix-maximum partition $(3, 3, 2, 2)$, but profiles $(9, 9, 9)$ and $(9, 9, 10)$. With two future positions and budget fifteen, routes $(3, 3), (1, 3), (3, 1)$ cost respectively $(15, 14, 13)$ after p and $(16, 15, 14)$ after p'. All initial actions are viable. Additive rewards $1/3$ early and $2/3$ late for depth three force deterministic worst-case regret at least $1/3$, or randomized regret at least $1/6$, under a common message. A static state-blind first decision and a maximum-feasible-depth policy are included in this failure.

A language conflict can disappear before any choice. With two depths, three past prices $(\ell, Q, c) = (2, 1, 0)$, and prefixes $(1, 1, 2)$ and $(2, 2, 1)$, spending is ten and profiles are $(10, 10)$ and $(10, 11)$. Future prices are $(10, 10, 0), (1, 1, 0)$ and $B = 22$. Both initial masks are $\{1\}$. Route $(1, 2)$ costs 22 in one history and 23 in the other; a static rank graph has an edge. Nevertheless, take the forced action one and then optimize using the final guard, which is $\{1, 2\}$ or $\{1\}$. Zero bits suffice for every utility. Theorem S.1 gives $M_* = 1$ because the stopped transcripts differ.

Counters and ordinary fixed-grid rounding. In the fixed-histogram family at $s = 8, d = 4$, histories $v = 1/16$ and $v = 3/32$ share all declared counters and round down to the same value on a grid of width $1/4$. The two-target threshold $17/256$ lies between them, forcing wrong-first-class additive regret at least $1/5$. This is a counterexample to that grid, not to every conservative rounding scheme: the sparse code itself is price-adapted conservative rounding. By contrast, the exact prospective cost after action three is $C + 4 - v$ in this family. If exposed before acting, it reveals v and eliminates additional handoff memory. The information travels through the quote rather than disappearing.

Superincreasing greedy is not general knapsack greedy. At weights $(2, 3, 4)$, capacity five, and cardinality two, descending greedy picks mass four whereas weights two and three attain five. The implemented constructor rejects such non-superincreasing input. This refutes generalizing this greedy step, not all possible compression algorithms for arbitrary prices. A numeric-capacity dynamic program would still have a state range that can grow exponentially with binary price length.

Other retained boundaries. A final-action mask is sufficient when $R = 1$, regardless of the number of exact profiles. For $K = 2$ and $\ell = Q$, $g(2) = g(1)$, so spending determines the whole profile. With $\ell_i = Q_i = \beta^{-i}$ and integer $\beta > K$, spending digits encode coefficients $d_i = r_i - \mathbf{1}\{k_i < r_i\}$, and

$$g(M) = \sum_i \beta^{-i} \max(d_i, M - 1).$$

The terminating radix representation is unique, so no additional profile information is needed; the exact scalar itself contains growing precision. Allowing a zero-price leading probe can reveal $g(3)$ after a monotone-safe action, illustrating why a lower bound must account for observations before the first irreversible choice. Scaling every price and budget in the six-position witness by an arbitrarily small positive number leaves its utility regret unchanged, refuting a same-budget regret bound based only on absolute cost distance. Approximate closeness is nontransitive; taking its transitive closure can violate any distortion target. Violating $\ell \geq Q$ can make raising a depth cheaper; violating $\ell + c \geq 0$ can make a final-feasible word overspend at an earlier prefix. The supplementary code includes these counterexamples and their finite checks.

X VERIFICATION AND FINITE BOUNDS FOR THE HIERARCHY

The finite checks use Python integers and exact rational arithmetic for physical costs, feasibility and utility. Entropy values in the bound plots are floating-point evaluations of the proved formulas. The domains below are fixed and all cases are reported.

Table 4: Declared verification domains and executed checks. These test finite constructions and implementations, not the asymptotic theorem or deployed-system performance.

Check	Count
Representatives, all $s = 2, \ldots, 16$ and $d = 2, \ldots, s + 1$	249,814
Adjacent separating cuts in those representative lists	249,679
Physical words, all common histories and $s = 2, \ldots, 5$	27,135
Completion first-action gaps, exhaustive small words	568
Rank-one additive first-action gaps	26
MPH relative-deletion set checks	2,528
Unchanged encoder/actor versus exhaustive optimal utility	852
Chosen-word prefix guard checks	5,424

For the combinatorial checks, all q-subsets are enumerated at every declared s, d; cardinality, bit count, and adjacent separation are checked directly. At $s = 2, \ldots, 5$, the implementation builds each actual past word and enumerates every word in $\{1, 2, 3\}^{s+2}$. Full physical cost is compared with the liability decomposition, and common telemetry and initial masks are verified. The first-action check compares the best feasible reward with first depth three against the best feasible waiting reward; no learned policy or significance estimate is involved.

The utility fixtures are maxima of three PH-d components, at every $d = 1, \ldots, R - 1$ and seeds 20260923 and 20260924. Each component uses six distinct nonempty edges of width at most d (or all available if fewer) with integer weights from one through five, normalized to total one. All 2^R set deletions are checked. The existing SI encoder and monotone actor are then run for every common-fiber history and distinct $k \in \{1, 2, 3, R - 1\}$. All exact safety and regret bounds pass. Twelve of the 852 cases have positive regret, with maximum $4/11$. The coarse k choices do not uniformly instantiate the theorem's small-ρ regime, and this is not a new empirical comparison claiming a compression advantage. Source and full per-case values are supplied.

X.1 FINITE BOUNDS

Figure 4 evaluates (27) and the information lower bounds across ranks. At $R = 258$ and $\rho = 1/20$, the rank-one lower and upper bounds are about 1.32 and 96 bits; at rank 16 they are 25.75 and 252, and at rank 257 they are 97.93 and 252. The exact common-histogram code uses 252 bits. Thus the matching law concerns order, not tight finite constants. Packet comparisons for different history families retain their own stated baselines and are not interchangeable with this histogram fiber.

X.2 CERTIFIED MINIMUM BITS ON SMALL INSTANCES

The preceding physical checks alone do not determine the best encoder. Here we certify exact minima for part of a specified finite grid while retaining the original randomized and later-feedback contract. A finite set of utility witnesses provides a lower bound for the full MPH-d class; the exact-history code provides its upper bound.

Witness tasks. Let $N = \binom{s}{h}$ and order the common histories by v. For a cut t before history t, take its h selected dyadic positions and the final anchor as targets. Their total price is v_t. Partition them into $m = \lceil (h + 1)/d \rceil$ nonempty groups of size at most d. Give the first depth-three action reward a, and completion of each group reward $c = (1 - a)/m$, with $0 < a < 1/(m + 1)$. This is a normalized PH-d utility. Early action plus all targets has minimum cost $B + v_t - v$; waiting plus all targets costs $B - 1 + v_t - v$ and is always feasible. Above the cut, waiting loses a. Below it, an early action misses at least one group and loses at least $c - a$ relative to waiting. Even complete state disclosure after the first action cannot undo these losses.

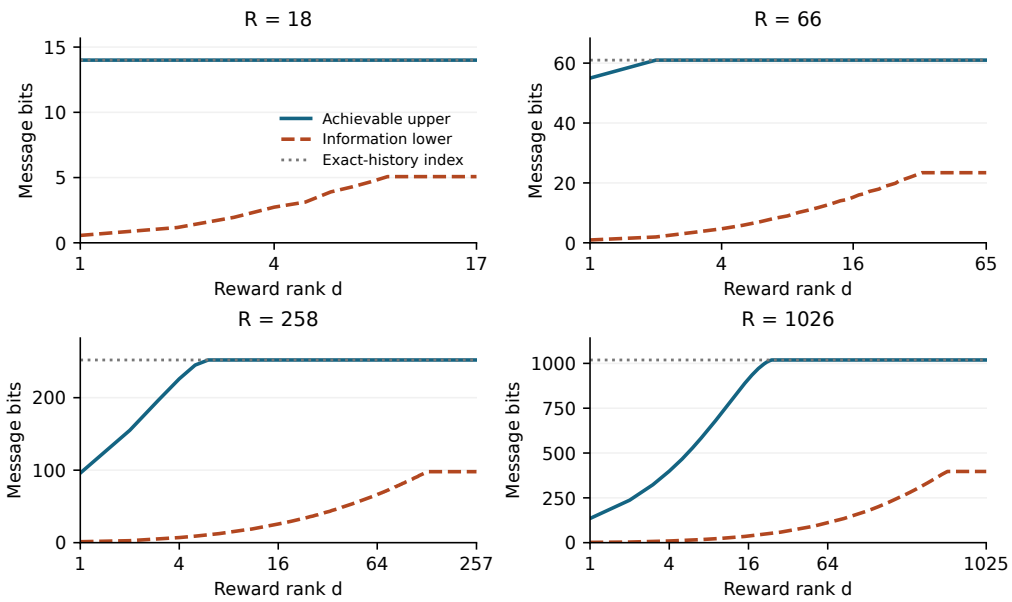

Figure 4: Proved bounds across all ranks at several horizons, with $\rho = 1/20$. Solid and dashed curves are achievable upper and information lower bounds; the dotted line is the exact common-histogram index. These are formula evaluations, not measured minimum message lengths.

An exact lower certificate. Fix $M < N$ messages and let $\mathcal{P}_M$ be all partitions of the N histories into M nonempty cells. Write q_t for the cut index of task t. For nonnegative rational weights α_{it} summing to one over history/task pairs, define

$$L_M(\alpha) = \min_{\mathcal{P}\in\mathcal{P}_M} \sum_{C\in\mathcal{P}} \sum_t \min\left\{ (c_t - a_t) \sum_{\substack{i\in C \\ i<q_t}} \alpha_{it}, a_t \sum_{\substack{i\in C \\ i\geq q_t}} \alpha_{it} \right\}. \tag{46}$$

Different reward weights at the same cut define distinct tasks. For each cell/task, the two terms are the weighted losses from choosing the early and waiting classes. A randomized decision cannot improve their minimum. Fixing the encoder's independent coins gives a partition; revealing its private coins to the decoder only helps. Partitions with fewer than M cells can be refined, and mixing over seeds cannot reduce the lower bound below the minimum over partitions. Hence every original M-message scheme has worst-case expected regret at least $L_M(\alpha)$, even with arbitrary computation and later feedback.

Set $b_* = \lceil \log_2 N \rceil$ and $M = 2^{b_*-1}$. If $L_M(\alpha) > \rho$, one fewer bit is impossible. The exact-history index attains zero regret for every admitted utility with b_* bits; therefore $b_d(\rho) = b_*$. A certificate that fails this strict test leaves the full minimum unresolved. The optimum of the finite witness relaxation is not identified with the full-class minimum.

Execution and complete results. The grid is $s = 2, \ldots, 5$, every $d = 1, \ldots, R-1$, and $\rho \in \{0.01, 0.05\}$. At every adjacent cut use $a \in \{1/(2m+1), j/[4(m+1)] : j = 1, 2, 3\}$. A linear program finds weights; their nonnegative rational reconstruction is normalized to sum one. A separate standard-library checker evaluates (46) exactly over every partition, independent of the numerical optimizer. Thus floating-point solver accuracy is not assumed in the certificate. Ten distinct witnesses cover the grid; ranks with the same m share a certificate. The largest case enumerates 750 partitions and uses an LP with 21,871 variables and 21,480 inequalities.

The certificates give exact minima in 23 configurations; 13 remain unresolved. All unresolved intervals have lower endpoint one: two additive representative histories with opposite optimal first actions and gap $1/5$ force zero-message regret at least $1/10$. The exact-history index supplies the displayed upper endpoints. These small, stringent-regret cases require every bit of that index; they do not demonstrate a rank separation or efficient compression at realistic horizons.

Table 5: Full minimum-bit grid. A single number is certified exact; an interval is unresolved. Shared/private randomization and later exact feedback remain permitted. Grouped rows expand to 36 configurations.

R	N	Rank d	M tested	$\rho = 0.01$	$\rho = 0.05$
4	2	1–3	1	1	1
5	3	1	2	2	$[1, 2]$
5	3	2–4	2	2	2
6	6	1–5	4	3	$[1, 3]$
7	10	1	8	$[1, 4]$	$[1, 4]$
7	10	2–6	8	4	$[1, 4]$

The implementation enumerates 27,135 history/future-word pairs and 2,880 task/history gaps for the new witnesses. For the refined alphabet, all $s = 2, \ldots, 10$ and $k = 1, \ldots, R - 1$ pass exact threshold-set equality and 233,717 nontrivial one-deletion repairs. Fresh-process replay reproduces the scientific outputs. Code, the complete grid and rational certificates are included in the supplementary artifact.

Z EXACT CHECKS OF THE PRECISION EXTENSION

The proof, protocol, source, copied reference engine, and environment were frozen before execution. All declared parameter cases are retained. The two fresh processes and one extracted-archive replay have identical scientific JSON; runtime receipts are separate. All physical costs and utilities use integer-scaled or rational arithmetic. Only the asymptotic-order inequality and displayed entropy formulas use floating point evaluation, with tolerance 10^{-12} in those checks.

Table 6: Precision-extension checks. Counts concern deterministic finite constructions and algorithms, not independent statistical samples.

Check	Count
Parameter triples: $R = 4, \ldots, 128$, $P = 1, \ldots, 32$, all ranks	260,000
Radix targets: $R = 4, \ldots, 24$, $P = 1, \ldots, 8$, all ranks	87,918
Paired profiles checked with the unchanged reference engine	312
Full physical words and profile decompositions	357,696
First-action class comparisons at all small cutoffs	1,704
MPH deletion-set inequalities	10,112
General-price encode–act comparisons	1,872
Selected-word prefix guards	11,232

The physical and algorithmic domain uses $R = 4, \ldots, 7$, $P = 1, \ldots, 4$, and every $d = 1, \ldots, R-1$. Each construction enumerates all 2^t paired histories and all 3^R words. The first action check optimizes each action class over those full words and checks every nontrivial cutoff against every history. The general-price code enumerates the sparse-tail alphabet from Appendix Q, at distinct $k \in \{1, 2, R-1\}$. It is deliberately an exhaustive small reference, not an SI encoder applied outside its assumptions or a fast large-instance method. For seeds 20260923 and 20260924, each utility is the maximum of three PH-d components, each with at most six distinct nonempty edges and integer weights in $\{1, \ldots, 5\}$ divided by 30. All sets, exact conservative optima, and selected prefixes are checked. All bounds pass; 23 of 1,872 cases have positive regret, with maximum $2/5$, under their coarse d/k guarantees. These are not a baseline win or tests of every small-regret deployment regime.

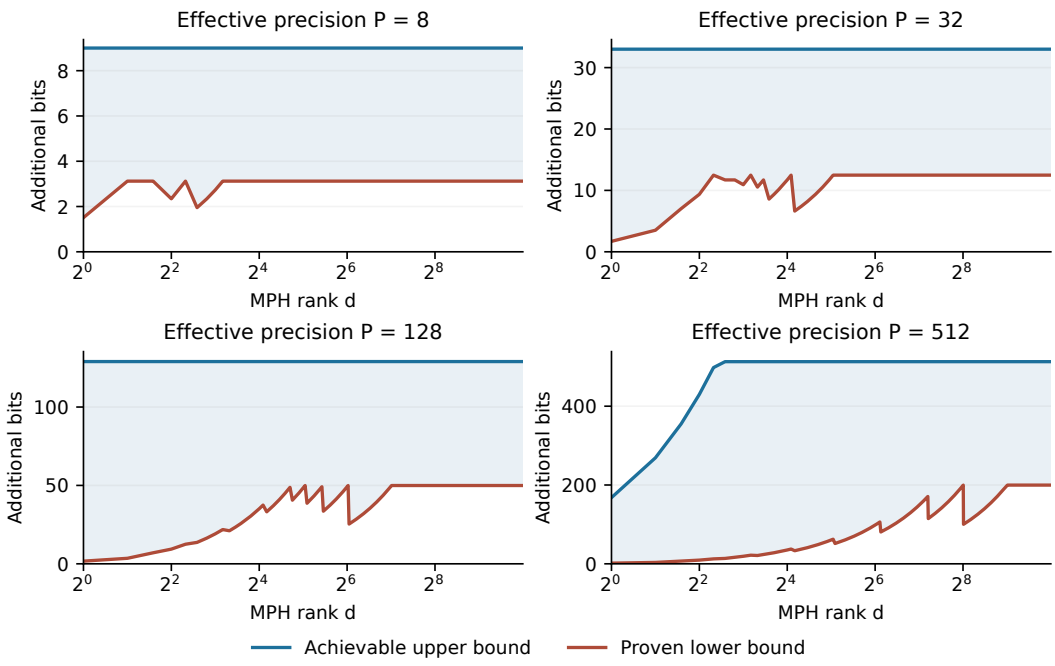

Figure 5: **Finite formula bounds at $\rho = 1/20$, not measured minimum bits.** Lines use (35) and the minimum of the exact grid, exact future-language, and sparse-tail upper counts. Each panel includes every rank at $R = 1024$. The cap refers to normalized liability resolution P, not a hardware word size. Gaps between the proven lower and achievable upper remain visible.

The supplied formula output retains $R \in \{64, 256, 1024\}$ and $P \in \{8, 16, 32, 64, 128, 256, 512, 1024\}$ at every rank; the figure shows four labeled precision panels at $R = 1024$ for legibility. It does not select empirical wins or fit a scaling exponent. The first execution takes 2.64 seconds and peaks at 65.39 MiB working set in the recorded Python 3.13.5 environment. This is verifier cost, not a production encoder benchmark. Mathematical proofs remain open to independent review and are not certified by finite replay.

AB Resource accounting and the scope of neural execution

The projection/MLP ledger isolates dense linear operations from attention and elapsed time. With hidden width h and MLP width f, the KV dependencies cost $A = 2h^2$ multiply-accumulates per block and position; the omitted body costs $q = 2h^2 + 2hf$. Thus the depth price is $\ell = A + q$ and the rebate is $Q = q$. At the audited ratio $f = 4h$, these reduce to normalized integer prices $\ell = 6, Q = 5$. This exact count excludes SDPA work and does not make an arbitrary deferred-cost price tape a faithful latency model.

Increasing model width scales work but need not increase normalized resource-state complexity. A direct integer counter can then encode the relevant liabilities cheaply. This is a substantive boundary for applications of the information lower bound: expensive computation alone does not imply an expensive resource descriptor.

The complete operator calculation, implementation audit and stronger native baselines are retained in the extended supplement's Appendix S.AB. The selective-execution experiment in Appendix AM uses supplied complete plans, so its latency differences do not test the benefit of compressed history.

AL PROOF AND EVIDENCE GUIDE

The supplementary archive separates complete proofs, exact finite checks, encoding comparisons and systems diagnostics. Its `CLAIM_EVIDENCE_MAP.csv` and `README.md` identify the source modules, frozen results and replay commands. File hashes check content integrity; they are not proof certificates or independent timestamps.

Table 7: Where to verify each central claim. An S-prefixed appendix is in `Supplement_Extended.pdf`; other appendices are in this PDF.

Claim or question	Argument and evidence
Physical resource model	Appendix A: decomposition, monotonicity and conditional profiles; Appendix F: the opening example.
Complementarity law	Appendix W: one common-histogram family, ordered queries and randomized information bound; Appendix X: finite checks, bound curves and certified minima.
Precision law	Appendix Y: mixed-radix palette and uniform constants; Appendix Z: exact verification domains.
Prediction witness	Appendix AC: Bayes identity, eight-plan oracle, bias limits and interface controls.
Constructive encoders	Appendices B, N, Q: packets, regret and computational costs.
Public resource features	Appendix AJ: exactness, Euclidean predecessor and scaling controls, including the low-precision loss.
Further comparisons	Appendices S.P, S.R, S.AG: full domains, runtime costs, skips and unfavorable cases.
Neural execution	Appendices AB, AM: operator accounting and numerical-fidelity scope; Appendix S.AM: the full panel.

Finite suites compare physical costs, feasibility, utility and prediction identities on their declared domains. Exhaustive means exhaustive only within those domains. Formula plots evaluate proved bounds; the separate rational certificates in Appendix X.2 establish the stated small-instance minima. Timing reruns use the same inputs and are not independent workload samples. Protocol deviations and failed criteria are retained with the associated comparisons. The complete neural panel reports numerical-fidelity limits and execution scope separately from resource-message claims.

AM PRETRAINED-MODEL EXECUTION AND NUMERICAL FIDELITY

The frozen CPU study compares native full, native prefix, fused-selective and split-selective executors on 72 Pythia-14M/70M cases: two fixed text windows, lengths 128/512/1024 and six schedules. All methods receive unchanged weights and identical complete plans. Two fresh processes reverse case order, with two warmups and seven retained calls per case. The timed call includes planning, masks, forward execution and extraction; shared loading/tokenization and vocabulary-head fidelity checks are excluded. Execution uses one CPU thread on an Intel Core i5-12500H. Full environment and protocol records accompany the source.

Table 8: Prespecified six-case regime: Pythia-70M, 1024 tokens, alternating/ascending/last-six and both windows. Ratios are geometric means of per-case median ratios against the faster native full/prefix.

Measure	Run 1	Run 2
Native / fused-selective latency	1.111	1.107
Call-latency reduction	10.01%	9.70%
Extra speed from split / fused	0.56%	1.22%
Fused activation/readout checks	6/6 pass	6/6 pass

The six primary cases pass performance and fidelity criteria, but the full panel fails its registered FP32 fidelity gate: 32/216 non-self method/case activation comparisons fail per process, and three NLL comparisons on one case exceed tolerance 10^{-3}. All 34,032 top-1 comparisons agree. Post-outcome FP64 diagnosis supports roundoff on the failed cases without changing the FP32 decision. Projection splitting fails its separate 5% increment criterion and stores an additional 18.9 MB of weights for 70M.

The full 72-case results, slower cases, raw timings, memory probes, operator counts, recorded deviations and archive replay appear in Appendix S.AM of the extended supplement. The result concerns CPU activation extraction, not generation, deployed serving or a benefit from resource-message compression.