SPARSE QUERY COMPILATION FOR REVERSE-CYCLIC MULTIPLAYER NIM

Anonymous authors
Paper under double-blind review

ABSTRACT

Exact symbolic reasoning in multiplayer games must preserve more than a binary win/loss outcome. We study the phase-zero reverse-cyclic recurrence for Nim, representing a background position by its response to one additional heap. Let $m \geq 3$ be the number of cyclic outcome classes and let $q \geq 0$ bound the background mass above $2m$. For backgrounds with p distinct heap heights below m and binary-encoded multiplicities, we give a uniform deterministic algorithm for exact response queries. A shielding lemma, arithmetic descriptions of unblocked coordinates, and an existential rank–history identity compile each threshold query into $(p + q + 2)^{O(q+1)}$ integer-linear systems, each with at most $2q + 1$ variables. Fixed-dimensional integer programming yields polynomial bit time for every fixed q, with exact values, requested positive-label coordinates or absence, and preference-maximizing actions. The bound is XP in upper excess, not an FPT result in q alone. The construction is supported by a sparse-response theorem and an explicit obstruction to composing response signatures. Definition-level tests and large-integer checks corroborate the interfaces; existing implementations cover restricted layers rather than the full uniform algorithm. Our results separate compact representation, composition, and exact inference in a multi-valued sequential problem.

1 INTRODUCTION

Exact strategy reasoning is sensitive to what a state representation forgets. Classical two-player Nim is exceptionally well behaved: an integer associated with each component supports outcome evaluation and composition by XOR. In a multiplayer preference game, a response may still be compact even when its summary does not compose. Moreover, a representation with few entries relative to the number of players need not have polynomial size when that number is binary encoded. These distinctions matter for exact symbolic inference and for constructing trustworthy targets or counterexamples for strategy synthesis.

We study ordinary Nim moves under the phase-zero reverse-cyclic outcome rule

$$u_m(\varnothing) = m - 1, \qquad u_m(P) = \big(1 + \max_{P \to Q} u_m(Q)\big) \bmod m \quad (P \neq \varnothing). \tag{1}$$

A move reduces exactly one heap. Labels belong to $\{0, \ldots, m-1\}$. A *preference-maximizing action* attains the maximum inside (1); we do not identify it with a guaranteed personal win or with a literal final placement under a different payoff model. The background-plus-observer response is $V_G(x) = u_m(G + *x)$, where $*x$ is one additional Nim heap.

Our main result is a uniform query algorithm for succinctly encoded backgrounds. Let p count distinct background heights, all below m, and suppose total background mass is at most $2m + q$. Although m, the multiplicities, and the observer size are binary encoded, exact thresholds are decidable in polynomial bit time for every fixed upper-excess bound q. The dependence on q is XP, not FPT in q alone. The algorithm also returns exact values, the unique coordinate of any requested positive label when present, and legal maximizing actions.

The key step is to turn a sequential prefix maximum into an existential arithmetic query. We first retain only the background children that can affect the maximum: small deletions, child tops, and specific penultimate-label events. Their top coordinates form a sparse union of intervals with at

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.

most one missing residue per piece. Chinese remaindering describes the unblocked coordinates, and a rank–history identity compiles their prefix recurrence into fixed-dimensional integer feasibility. We bound the number of branches, variables, and coefficient bits separately.

Our contributions are (i) a sparse-response representation and a concrete failure of additive composition; (ii) a uniform fixed-upper-excess query compiler with an explicit bit-complexity bound and constructive action recovery; and (iii) computational checks that distinguish mathematical guarantees from the scope of implemented modules. Complete structural and algorithmic proofs are collected in the appendices.

The connection to symbolic AI is representation and exact inference for a non-Boolean sequential task. Existing exact evaluators can be used within a synthesis loop, but we do not claim a completed learning system, empirical generalization, or superiority to learned baselines. The main result is an exact, structurally restricted algorithm.

2 Related work

Bouton's theory gives the classical XOR solution to two-player Nim (Bouton, 1901–1902). Multiplayer variants depend on preferences and termination conventions. Li (1978) studies n-person Nim and Moore's games; the forward arithmetic in this line counts binary columns modulo the player count, rather than applying ordinary XOR to base-m digits.

Suetsugu (2019) studies the reverse-preference model relevant here and proves unique one-coordinate completion for the distinguished outcome. We do not claim to introduce that model or the completion-uniqueness theorem. Our contribution concerns response representation and exact computation under explicit encoding and parameter restrictions. The multiplayer-preference discussion in Nowakowski (2025) provides further context, but does not itself formulate or resolve our sparse bit-complexity problem.

Exact generalized strategy synthesis is studied by Wu et al. (2020) using SMT, and by Fang et al. (2024) using exact binary classification. These are methodological predecessors, not reverse-multiplayer Nim evaluators. Their conventional two-player winning/losing semantics differ from (1). An elementary multiclass encoding alone would not establish a new synthesis method; our substantive claims are the structural compression and size-controlled arithmetic compiler.

We use fixed-dimensional integer programming (Lenstra, 1983) to decide each compiled conjunction. That result does not control how many conjunctions a game reduction produces. Accounting for their expansion is essential to our XP classification. We make no unrestricted polynomial-time or literature-priority claim for multiplayer Nim.

3 Model and sparse response information

An impartial game is a finite acyclic directed graph with a designated root; its options are its outgoing neighbors. A disjunctive sum permits a move in one component. The Nim heap $*x$ has options $*0, \dots, *(x-1)$; $*0$ is terminal. Extend (1) to these games and put $d = m - 1$. For $r \in \{1, \dots, d\}$, write $\sigma_r(G)$ for the coordinate with $V_G(x) = r$, when it exists, and $\perp$ otherwise.

Theorem 3.1 (Sparse one-heap response)**.** *For every finite impartial game, every positive label occurs in its one-heap response at most once, and label d occurs exactly once. Defined labels and their coordinates are strictly increasing. Thus $\mu_m(G) = (\sigma_1(G), \dots, \sigma_d(G))$ describes the response, which is zero elsewhere.*

Proof. For nonterminal G, let $M_G(x) = \max_{0 \le z < x} V_G(z)$ and $E_G(x) = \max_{H \in \mathrm{Opt}(G)} V_H(x)$. At $x = 0$, use zero for the missing observer maximum. Splitting the legal moves gives

$$V_G(x) = (1 + \max\{M_G(x), E_G(x)\}) \bmod m. \tag{2}$$

If a positive label r occurs, every later observer maximum is at least r, so the next positive label is strictly larger. The terminal row is d at zero and zero thereafter. Inducting on the game graph, each child row has finite positive support. Beyond its last event, each pre-top coordinate increases the running label by one. A top must therefore occur; after it, all responses are zero. □

The proof gives a sparse scan: initialize a running label at zero, take its maximum with the child event at each coordinate, emit zero if this maximum is d, and otherwise increment it by one. Stop at label d. If a nonterminal game has at most Δ distinct children, then

$$\sigma_r(G) \le \Delta + r - 1 \quad \text{whenever } \sigma_r(G) \neq \bot. \tag{3}$$

Indeed, each preceding zero requires a distinct child top, and there are at most $r-1$ preceding positive entries. For Nim, Δ is at most $D(G)$, the sum of distinct positive heap heights; repeated heaps do not multiply the number of distinct child positions.

At $m = 2$, the scan is mex and the single marker is the Sprague–Grundy number (Sprague, 1935). This does not imply additive composition for more players. Already at $m = 3$, direct evaluation gives

$$\begin{aligned} \mu_3(1,1,1) = \mu_3(1,2) &= (\bot, 0), \\ \mu_3(1,1,1,1) = (1,2), \qquad & \mu_3(1,1,2) = (2,3). \end{aligned} \tag{4}$$

Equal signatures become unequal after adding the same component. Thus no binary operation on these signatures alone computes $\mu_3(G+H)$ for all games. This rules out that particular summary as an additive state, not every richer representation.

Appendix A characterizes all formal signatures and their information content. Every admissible increasing label/coordinate vector is realizable by a finite impartial game; the exact number with top at most C is $\binom{C+d}{d}$. This is a representation statement, not a runtime lower bound. Since the marker count can be exponential in $\log m$, our main algorithm provides queries rather than materializing the entire vector.

4 Uniform exact-query compilation

4.1 Input and guarantee

Let $m \ge 3$ and let q be a nonnegative integer. A canonical sparse background is

$$\begin{aligned} G = ((h_1, n_1), \dots, (h_p, n_p)), \qquad & 1 \le h_1 < \cdots < h_p < m, \quad n_i \ge 1, \\ S(G) = \sum_i h_i n_i \le 2m + q. & \end{aligned} \tag{5}$$

All integers and multiplicities are binary encoded. The observer $x \ge 0$ may be arbitrarily large; only the background heights are bounded by m. Write $N = p + q + 2$ and let L include the root and concrete query encodings. A loose q may be replaced by $\max(0, S(G) - 2m)$.

Theorem 4.1 (Fixed-upper-excess threshold queries)**.** *There is a deterministic uniform algorithm deciding $V_G(x) \ge a$ on (5) in*

$$g(q) N^{8q + C_0} L^{C_1} \tag{6}$$

bit operations, for computable g and absolute constants C_0, C_1. At truncated excess $r = \max(0, S(P) - 2m) \le q$, a threshold compiles to at most $c_0 N (c_4 N^6)^r$ integer-linear conjunctions, each with at most $2r+1$ existential integer variables, $O(r+1)$ rows, polynomial-bit coefficients, and explicit finite signed variable bounds. The base $r = 0$ needs no existential variable.

Hence the threshold problem is XP parameterized by upper excess and polynomial in the sparse binary input for every fixed excess bound. The same construction gives $\mathrm{Top}(G)$, $V_G(x)$, the coordinate of a requested positive label or *absent*, and one maximizing action; it returns *no move* only at the empty terminal. The public object supports concrete queries, not an explicit list of $m-1$ markers or necessarily a single materialized free-parameter formula.

The player parameter is not held fixed in this bit-complexity guarantee. For example, $m = 3 \cdot 2^b$ and $G = (2^{b+1} + 1)^3$ have exact upper excess three for every $b \ge 1$, while every background height remains below m. A method enumerating observer coordinates, heap magnitudes, or all player labels would be exponential in b. Conversely, for a fixed player count and growing mass, q grows; the theorem does not then promise a polynomial bound independent of q.

4.2 SPARSE BASE AND LARGE-DELETION SHIELDING

For $S(P) \le 2m$, the exact base has $O(p+1)$ disjoint positive-response records. On a bounded interval a record is either

$$V_P(x) = x + \delta \quad \text{or} \quad V_P(x) = b + x - \left\lceil \frac{C-x}{n} \right\rceil, \quad n \ge 1. \tag{7}$$

The latter has threshold condition

$$na - (n+1)x \le nb - C. \tag{8}$$

This follows from $\lceil y/n \rceil \le z \iff y \le nz$ for integer z, including negative z. The base separately returns true for $a \le 0$ and false for $a \ge m$. Appendix D derives all low-mass, strict-subdouble, and exact-double cases. In the resonant exact-double case, follower-top coverage certifies a zero region and computes a delayed top; those coverage pieces are not positive parent records.

For an upper state $S(P) = 2m + r$, $r \ge 1$, deletions of size $1 \le \delta \le r$ reach smaller excess and are prepared recursively. Deletions $\delta \ge r+1$ are compressed rather than enumerated. The shielding lemma in Appendix E shows that a large-deletion event of value at most $m-3$ is dominated by the immediately shallower child from the same source. Only small children, large-child tops, and exact value-$(m-2)$ triggers are needed to preserve the parent envelope.

To illustrate the arithmetic geometry, remove a source heap h and put $B = P - *h$. Large targets obey $0 \le y \le h - r - 1$. Their child tops have form

$$T(y) = h - r - 1 - y + t(y), \qquad t(y) = \min\{c \ge 1 : A_B(c) + \min(y, c) \ge m + 1\}, \tag{9}$$

where $A_B(c) = \sum_{h \in B} \min(h, c)$, counting multiplicity. Sparse capacity profiles make $t(y)$ constant or inverse-ceiling on clipped intervals. The varying map has the form $z + \lceil z/n \rceil$, whose image is an interval missing one residue modulo $n+1$. Across all sources there are $O((p+q)^2)$ full or one-missing-residue pieces. Their monotone inverse recovers a legal source/target.

Interior triggers are child tops shifted down by one coordinate. The boundary target $y = h - r - 1$ needs the distinct test

$$A_Q(t_Q) \ge m + 2, \qquad t_Q = \min\{c \ge 1 : A_Q(c) \ge m + 1\}, \tag{10}$$

on its child Q, which has mass $2m - 1$. Using the parent's capacity is wrong: for $m = 3$, $P = (1, 1, 1, 2, 2)$ and move $2 \to 0$, the parent capacity at 1 is 5 but the child's is 4; the child response at zero is 0, not 1.

4.3 OPEN RANK AND THE RANK–HISTORY IDENTITY

Call a coordinate blocked if some background child has top label there. Equation (2) then gives parent value zero. Split the line only at blocker-piece endpoints. On a cell, a full blocker leaves no open point. Otherwise being outside every holey blocker means satisfying all its missing-residue congruences. Generalized CRT gives one progression or the empty set. Rank is retained on empty-open cells as a constant; no residue period is enumerated.

List open coordinates as $z_1 < z_2 < \cdots$, with child envelope $e_i \le m - 2$. Define unwrapped history

$$H_0 = 0, \quad H_i = 1 + \max\{H_{i-1}, e_i\} = i + \max\left\{0, \max_{j \le i}(e_j - j + 1)\right\}. \tag{11}$$

Let $R_P(x)$ count open coordinates through x, and set $H_P(x) = H_{R_P(x)}$. Let $\mathrm{ChildAtLeast}_P(w, a)$ denote existence of a background child whose response at w is at least a.

Lemma 4.2 (Rank–history threshold identity). *For $\ell > 0$,*

$$\begin{aligned} H_P(x) \ge \ell \quad \iff \quad & R_P(x) \ge \ell \ \textit{or} \\ & \exists\, \textit{open}\ w \le x : \mathrm{ChildAtLeast}_P\big(w, R_P(w) + \ell - R_P(x) - 1\big). \end{aligned} \tag{12}$$

Proof. In (11), the baseline is $R_P(x)$. The term associated with $w = z_j$ reaches ℓ exactly when $e_j \ge j + \ell - R_P(x) - 1$. □

Algorithm 1 Uniform preparation from a raw sparse background

Require: $m \geq 3$, canonical P, heights below m
Ensure: Concrete response-query interface and exact top for P
if $S(P) \leq 2m$ **then**
  return the sparse base records and their query interface
end if
$r \leftarrow S(P) - 2m$
for each source height h and $1 \leq \delta \leq \min(r, h)$ **do**
  prepare $P - *h + *(h - \delta)$ recursively; retain its parent edge
end for
compile all $\delta \geq r + 1$ into large top and trigger pieces
form the child-top union and its CRT open-rank cells
construct the raw-history template using (12)
$T \leftarrow$ first raw-history crossing of $m - 1$ in $[0, S(P)(m - 1)]$
return T and the threshold interface given by (13)

Generated nonpositive child thresholds are true and cannot be omitted; for action queries they retain a real parent edge. Thresholds at least $m - 1$ are false on open coordinates.

The first history crossing of $m - 1$ is exact and is the parent top. For nonempty P, (3) supplies $\mathrm{Top}(P) \leq D(P) + m - 2 \leq S(P)(m - 1)$. For $1 \leq a \leq m - 1$,

$$V_P(x) \geq a \iff x \leq \mathrm{Top}(P), \quad x \text{ open}, \quad H_P(x) \geq a. \tag{13}$$

Raw history is monotone; the public response is not. Top and label searches therefore use (12).

For a concrete example, take $m = 3$ and $G = 1^7$. Its only background child is 1^6, whose top is at zero. The open coordinates are therefore $1, 2, \ldots$, the child envelope there is zero, and $R_G(x) = H_G(x) = x$ for $x \geq 1$. The first history crossing of two is at $x = 2$. The actual response is $(0, 1, 2, 0, 0, \ldots)$: continuing the monotone history without the cutoff would not evaluate the game.

4.4 Uniform preparation, bit complexity, and actions

Preparation follows strict lower excess: prepare the small children, compile the large pieces, assemble rank cells, build raw history, determine the current top, and only then add the response cutoff. Thus raw construction does not call its own top or response. A path has at most q recursive steps and support at most $p + q$.

Algorithm 1 is the mathematical construction, not a claim that the current prepared-state bridge implements every line for arbitrary q. Its loops enumerate support and upper-excess choices; large numeric deletion ranges are handled only by the piece compiler.

The complete compiler proof in Appendix E bounds prepared states by $A_q \leq (1 + qN)^{q+1}$ and branches per threshold by $M_r \leq c_0 N (c_4 N^6)^r$. Their product is at most $2^{q+1} c_0 c_4^q N^{8q+3}$. Source, state, cell, and trigger choices are DNF alternatives, not hidden integer selectors. Each visited upper layer introduces at most one current-rank quotient and one history-witness quotient. A missing-residue trigger may add one quotient only at the terminal branch, giving at most $2r + 1$ variables.

Affine substitution creates no product of unknowns. CRT modulus lengths are bounded by the sum of participating modulus lengths. With root encoding length Λ, a safe coefficient/bound length is $O((r + 1)N^2(\Lambda + \log(q + 2)))$, plus concrete query bits. Deterministic fixed-dimensional integer programming decides each conjunction (Lenstra, 1983), establishing Theorem 4.1. The $N^{O(q+1)}$ branch expansion remains; small ILP dimension alone does not imply FPT in q.

For action recovery, satisfiability needs the correct provenance. At a blocker, use its child-top action; after the top, reduce the observer to that top. At an open $x \leq \mathrm{Top}(P)$, compare the exact prior maximum $p_0 = H_P(x - 1)$ (zero at $x = 0$) with the current child envelope. If positive prior history controls, find the first coordinate z attaining that exact value and return $x \to z$. The existential witness w in (12) need not be this target. For $m = 5$, $P = 1^{11}$, $x = 3$, a witness $w = 1$ can certify threshold 3, yet $V_P(1) = 1$ and $V_P(2) = 2$; the maximizing observer target is 2. If a current child

Suite	Domain and result
Sparse structure	Random-DAG and constructed-vector suites: 232,922 marker assertions and 2,875 formal-vector/minimum-degree checks; all passed.
Base thresholds	All admissible $m = 3, \ldots, 11$, $S \leq 2m$, $x = 0, \ldots, 2m+1$, $a = -2, \ldots, m+1$: 8,958 backgrounds; 2,714,852 thresholds; no mismatches.
Upper recurrence	All bounded-part backgrounds $m = 3, \ldots, 7$, exact excess $1, \ldots, 4$: 1,148 roots; 94,989 response/action pairs and 808,835 threshold checks; all passed.
Finite DNF	Seven complete selected formulas, 10,437 branches, and 18,353,876 finite free assignments.
Large arithmetic	4,097-bit player parameter, sampled $q \in \{1, 3, 7\}$ families: 12 sources, 21 pieces, 57 targets, 63 top and 63 trigger inversions.

Table 1: Deterministic correctness checks and their actual domains. They support interfaces, not universal proofs or performance comparisons.

controls, recover a child at the exact maximum and return its stored parent edge or inverted large piece. If both maxima are zero, a real background edge suffices.

A decision-only feasibility backend recovers signed bounded witnesses by binary-searching each variable's least feasible value. Coordinate searches use logarithmically many decisions in the numeric interval width, but every decision may scan all M_q conjunctions. These are not logarithmic overall running times.

5 Verification interface and computational checks

For fixed m and fixed heap arity, let candidate outcome predicates $\Phi_0, \ldots, \Phi_{m-1}$ partition the nonnegative integer tuples. Write $E_j(x) = \exists y\,(\mathrm{Opt}(x, y) \wedge \Phi_j(y))$. The terminal condition is $\Phi_{m-1}(0)$, and the exact nonterminal constraints are

$$\Phi_0 \longleftrightarrow E_{m-1}, \qquad \Phi_j \longleftrightarrow E_{j-1} \wedge \bigwedge_{r=j}^{m-1} \neg E_r \quad (j > 0). \tag{14}$$

Necessity follows from the maximum in (1); sufficiency follows by induction on heap mass. Presburger candidates can therefore be verified by a decidable first-order theory. This neither supplies such candidates for unrestricted Nim nor makes verification efficient. We view it as an interface for future exact synthesis, not an evaluated synthesis method.

The checks in Table 1 target formulas, boundary cases, and witness semantics. In finite domains, a separate memoized literal recurrence enumerates legal reductions. An action is checked against the maximum over *all* legal children, not only for legality. Large-integer checks instead inspect arithmetic identities and integer certificates; they are not brute-force truth oracles at that scale. Suite domains overlap, so counts must not be added as disjoint samples.

A selected end-to-end excess-three SMT fixture uses $m = 3 \cdot 2^{4095}$, $h = 2^{4096} + 1$, and $G = (h, h, h)$. It returns $\mathrm{Top}(G) = 2^{4095} - 2$ and a maximizing one-stone source move at observer $x = \mathrm{Top}(G)$, using 4,612 top leaves across four upper states and 20,499 solver calls, with no UNKNOWN results. It demonstrates a specific large-integer path, not the deterministic worst-case backend of Theorem 4.1.

The implementation boundary is material. The compressed prepared-state bridge supports only $q \leq 4$, needs a supplied upper-state top provider, and lacks the full public upper query API. It is not the all-q raw-input algorithm proved here. The exact-excess induction closes that dependency mathematically, while production integration remains future work. A recorded approximately 30,000-branch performance abort also prevents interpreting fixed-q polynomiality as uniformly small practical cost.

No neural comparison, statistical generalization, or speedup is claimed. Appendix F records reproducibility boundaries and additional regression cases. Generic proof conclusions rely on the mathematical induction, not on these tests.

6 LIMITATIONS AND CONCLUSION

The main compiler requires phase zero, background heights below m, and fixed upper excess. It allows one distinguished arbitrarily large heap, not two arbitrary large heaps. Other phases and explicitly listed residual-core inputs admit complementary results in Appendix B, but use different parameters and encodings.

The sparse signature summarizes one-heap response, not arbitrary-context equivalence. Its realizability theorem ranges over finite impartial games, not exclusively Nim. The explicit three-player non-composition example does not establish an obstruction for every richer representation. Information-counting results likewise do not prove algorithmic hardness.

Our contribution is a restricted exact symbolic algorithm, not a trained model or a new generic synthesis framework. The next empirical step is a full raw-input implementation and a controlled comparison within a multiclass synthesis pipeline. Neither is assumed in the present claims.

In summary, reverse-cyclic Nim separates compact response representation from additive composition and efficient inference. Shielding and rank–history compilation give exact, constructive queries in polynomial sparse bit time for every fixed upper-excess bound, while keeping the model, encoding, parameter dependence, and executable scope explicit.

REPRODUCIBILITY STATEMENT

The recurrence, parameter restrictions, and public query semantics are explicit in Sections 3–4. Appendices give the structural, base-band, and upper-compilation proofs. Section 5 and Appendix F distinguish definition-level enumeration from sampled large-integer certificates and declare the current implementation's limitations. A complete standalone all-q compiler is not claimed as an artifact.

ETHICS STATEMENT

The experiments study abstract games and use no human-subject or personal data. The principal integrity concerns are correctness, reproducibility, attribution, and disclosure of AI assistance. A maximizing action has only the mathematical meaning specified in (1), not a recommendation for real-world strategic conduct.

AI USE STATEMENT

OpenAI Codex and its underlying language models assisted with literature discovery, research ideation, mathematical formulations and proof development, computational cross-checks, implementation of evaluators and verification scripts, experimental analysis, and manuscript drafting and editing. This assistance was not limited to language polishing. Synthetic finite test instances were generated by AI-assisted scripts; their ground-truth labels came from explicit game recurrences, not language-model predictions. Separate AI-assisted proof audits and counterexample searches are not independent human peer review. The human authors retain responsibility for the final claims, proofs, citations, and reported experiments.

REFERENCES

Charles L. Bouton. Nim, a game with a complete mathematical theory. *Annals of Mathematics*, 3 (1/4):35–39, 1901–1902. doi: 10.2307/1967631.

Liangda Fang, Meihong Yang, Dingliang Cheng, Yunlai Hao, Quanlong Guan, and Liping Xiong. Generalized strategy synthesis of infinite-state impartial combinatorial games via exact binary classification. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pp. 562–570. IFAAMAS, 2024. URL https://www.ifaamas.org/Proceedings/aamas2024/pdfs/p562.pdf.

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

S.-Y. R. Li. N-person Nim and n-person Moore's games. *International Journal of Game Theory*, 7 (1):31–36, 1978. doi: 10.1007/BF01763118.

Richard J. Nowakowski. Unsolved problems in combinatorial games. In Urban Larsson (ed.), *Games of No Chance 6*, volume 71 of *Mathematical Sciences Research Institute Publications*, pp. 55–98. Cambridge University Press, 2025. doi: 10.1017/9781009565486.005.

R. Sprague. Über mathematische kampfspiele. *Tohoku Mathematical Journal, First Series*, 41:438–444, 1935. URL https://www.jstage.jst.go.jp/article/tmj1911/41/0/41_0_438/_article.

Koki Suetsugu. Multiplayer games as extension of misère games. *International Journal of Game Theory*, 48(3):781–796, 2019. doi: 10.1007/s00182-019-00662-8.

Kaisheng Wu, Liangda Fang, Liping Xiong, Zhao-Rong Lai, Yong Qiao, Kaidong Chen, and Fei Rong. Automatic synthesis of generalized winning strategies of impartial combinatorial games using SMT solvers. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pp. 1703–1711, 2020. doi: 10.24963/ijcai.2020/236. URL https://www.ijcai.org/proceedings/2020/236.

A SPARSE RESPONSES, REALIZABILITY, AND DEGREE-SENSITIVE REPRESENTATION

This appendix supplies the structural proofs used in the main text. Unless explicitly stated otherwise, the terminal seed is phase zero. Fix $m \geq 2$, write $d = m - 1$, and define, on finite acyclic impartial games,

$$u_m(0) = d, \qquad u_m(G) = \left(1 + \max_{H \in \mathrm{Opt}(G)} u_m(H)\right) \bmod m \quad \text{if } \mathrm{Opt}(G) \neq \varnothing. \tag{15}$$

Here 0 denotes a terminal game, $+$ is disjunctive sum, and $*c$ is the ordinary Nim heap of size $c \in \mathbb{N}_0$. We set $V_G(c) = u_m(G + *c)$.

A.1 EXACT SPARSE RESPONSE AND ITS RECURSIVE COMPUTATION

Theorem A.1 (Sparse response)**.** *For every finite impartial game G, each label $r \in \{1, \ldots, d\}$ occurs at most once in the row V_G, and label d occurs exactly once. Let $\sigma_r(G)$ be the coordinate of label r, or $\perp$ if the label is absent. The vector*

$$\mu_m(G) = (\sigma_1(G), \ldots, \sigma_d(G))$$

determines the entire row. Its defined labels have strictly increasing coordinates. If G is nonterminal and has n distinct options, then

$$\sigma_r(G) \neq \perp \quad \Longrightarrow \quad \sigma_r(G) \leq n + r - 1. \tag{16}$$

Proof. For a nonterminal background, the options of $G + *c$ split into a move in G and a reduction of the appended heap. Define

$$M_G(c) = \max\big(\{V_G(z) : 0 \leq z < c\} \cup \{0\}\big), \qquad E_G(c) = \max_{H \in \mathrm{Opt}(G)} V_H(c).$$

Since the second maximum is nonempty, the zero baseline in $M_G(0)$ is harmless. The recurrence gives

$$V_G(c) = \big(1 + \max\{M_G(c), E_G(c)\}\big) \bmod m. \tag{17}$$

If $V_G(c) = r > 0$, the maximum on the right is exactly $r - 1$. At every later coordinate the history maximum is at least r. Consequently the same positive label cannot recur, and every subsequent positive label is strictly larger. Once d occurs, the history maximum remains d and all subsequent outputs are zero.

For the terminal background, the row is $d, 0, 0, \ldots$: at a positive heap size, reducing that heap to zero reaches label d. Now induct on the rank of a nonterminal game. By induction every option row has finite positive support, so E_G has finite support. If the parent has not yet emitted d, then beyond the final child event each coordinate increases the running positive label by one. A top marker therefore occurs after finitely many coordinates. This establishes existence as well as uniqueness and order.

For the coordinate bound, consider the coordinates strictly before $c = \sigma_r(G)$. At most $r - 1$ of them have a positive label. At each remaining coordinate the output is zero while the history maximum is strictly below d, so an option must contribute label d there. An option has only one top marker. Hence at most n preceding coordinates are zero, and $c \leq n + r - 1$. □

The proof is constructive. Given all option vectors, form the finite event function

$$e_G(c) = \max\Big(\{r : \sigma_r(H) = c,\ H \in \mathrm{Opt}(G)\} \cup \{0\}\Big).$$

Initialize a running label $b = 0$. Scan coordinates in increasing order and put $t = \max\{b, e_G(c)\}$. If $t = d$, emit zero and leave b unchanged. Otherwise emit $t + 1$, record $\sigma_{t+1}(G) = c$, and set $b = t + 1$. Stop when $b = d$. The terminal vector is initialized separately: only its top marker is defined, at zero. In particular, the construction does not apply the nonterminal update to an empty option set.

Proposition A.2 (Explicit-DAG computation)**.** *On an explicitly supplied follower DAG with q vertices and E edges, all marker vectors can be computed in $O(d(E+q))$ word operations using $O(dq)$ coordinate words of storage.*

Proof. Process the vertices in increasing rank. At a nonterminal vertex with n options, discard option events beyond $n + d - 1$. By (16), no parent marker lies beyond that coordinate. Bucket the remaining events and scan the interval $\{0, \dots, n + d - 1\}$. Accumulating the events costs $O(dn)$ operations, and the scan costs $O(n + d)$. Summing over vertices gives the stated time bound. Each vertex retains at most d coordinates, while a temporary bucket array has at most $n + d$ entries. □

This is a bound for an explicit graph. It does not imply polynomial time in the binary encoding of a Nim position, whose follower graph can be exponentially large. When $m = 2$, the update has a single marker and reduces exactly to the ordinary mex recursion. For three players the marker vector already fails to be a disjunctive-sum congruence, as witnessed in the main text; that example alone makes no separate assertion for every fixed $m \geq 4$.

A.2 Pure top markers and isolated lower events

A vector is *formally admissible* if its defined labels and coordinates can be written

$$1 \leq r_1 < \cdots < r_k = d, \qquad 0 \leq c_1 < \cdots < c_k. \tag{18}$$

In game constructions below, $\{G_1, \dots, G_s\}$ denotes a root whose options are precisely the displayed games.

Lemma A.3 (Pure top games)**.** *For every $c \in \mathbb{N}_0$ there is a finite impartial game A_c whose only positive marker is $\sigma_d(A_c) = c$.*

Proof. Set $A_0 = 0$. For $c \geq 1$, suppose $A_0, \dots, A_{c-1}$ have already been constructed. Define

$$H_{1,c} = \{A_0, \dots, A_{c-1}\},$$
$$H_{s,c} = \{A_0, \dots, A_{c-1}, H_{s-1,c}\}, \qquad 2 \leq s \leq d.$$

The pure top options block precisely the coordinates $0, \dots, c - 1$. For $s = 1$, the scan then emits labels $1, \dots, d$ at consecutive coordinates $c, \dots, c + d - 1$. Induction on s gives

$$\sigma_r(H_{s,c}) = \begin{cases} \bot, & r < s, \\ c + r - s, & s \leq r \leq d. \end{cases} \tag{19}$$

Indeed, $H_{s-1,c}$ supplies height $s - 1$ at c and the successive heights needed by the scan. Set $A_c = H_{d,c}$. Dependencies are acyclic: first increase c, then s. When $d = 1$, only $H_{1,c}$ is needed, and this is the usual mex construction. The case $c = 0$ was seeded separately and does not use an empty list in $H_{1,c}$. □

Lemma A.4 (Horizon-isolated events)**.** *Assume $d \geq 2$. For every $C \geq 0$, $0 \leq c \leq C$, and $1 \leq r \leq d - 1$, there is a finite impartial game $J^C_{r,c}$ whose only positive marker within $\{0, \dots, C\}$ is label r at c.*

Proof. Use the option set

$$\mathcal{B}_{C,c} = \{A_j : 0 \leq j \leq C,\ j \neq c\} \cup \{A_{C+1}\}.$$

The final option makes the set nonempty even when $C = c = 0$ and contributes no event inside the horizon. Let

$$\mathrm{Opt}(J^C_{1,c}) = \mathcal{B}_{C,c}, \qquad \mathrm{Opt}(J^C_{r,c}) = \mathcal{B}_{C,c} \cup \{J^C_{r-1,c}\} \quad (2 \leq r \leq d - 1).$$

Every horizon coordinate except c is blocked. At c, the first game emits label one; the preceding isolator then supplies the height required to emit label r in each successive construction. More explicitly, the scan gives

$$\sigma_s(J^C_{r,c}) = \begin{cases} \bot, & s < r, \\ c, & s = r, \\ C + s - r + 1, & s > r. \end{cases}$$

All spill markers lie strictly beyond C. The construction increases r and therefore introduces no cycle. □

Theorem A.5 (Complete realizability)**.** *Every formally admissible vector is $\mu_m(G)$ for some finite acyclic impartial game G.*

Proof. Write the desired vector as in (18) and let $C = c_k$. If $d = 1$, the pure top game A_C is sufficient. Assume $d \geq 2$. Give the desired root the following options:

1. A_x at each coordinate $x \in \{0, \ldots, C-1\} \setminus \{c_1, \ldots, c_{k-1}\}$;
2. $J^C_{r_j-1,c_j}$ for each desired label $r_j \geq 2$;
3. the harmless beyond-horizon game A_{C+1}.

At an undesired coordinate before C, a child top blocks the output. At c_j, the event height is $r_j - 1$, with zero as the default when $r_j = 1$. The preceding running label is at most $r_j - 1$, so the scan emits exactly r_j. At C it emits d and terminates. Spill events do not interfere inside the horizon. Every child is a game constructed earlier, proving finiteness and acyclicity. This argument includes $C = 0$. □

Proposition A.6 (Exact count at a marker horizon)**.** *The number of distinct response vectors with $\sigma_d(G) \leq C$ is*

$$\sum_{k=1}^{d} \binom{d-1}{k-1}\binom{C+1}{k} = \binom{C+d}{d}. \tag{20}$$

Proof. If k labels occur, choose $k-1$ lower labels from $d-1$ possibilities and k increasing coordinates from $\{0, \ldots, C\}$. Every choice is realizable by Theorem A.5. The final identity is Vandermonde's identity, using the convention that an out-of-range binomial coefficient is zero. □

A.3 Minimum root degree

For a formal vector as in (18), put $r_0 = 0$, $C = c_k$, and define

$$Z = \{0, \ldots, C-1\} \setminus \{c_1, \ldots, c_{k-1}\}, \qquad z = |Z| = C + 1 - k, \tag{21}$$

$$J = \{c_j : r_j > r_{j-1} + 1\}. \tag{22}$$

Thus Z consists of the zero coordinates before the top, and J consists of the coordinates at which the desired label skips at least one label.

Theorem A.7 (Exact minimum root outdegree)**.** *The minimum root outdegree among nonterminal realizations of this vector is*

$$\delta_{\mathrm{nt}} = \begin{cases} z, & J = \varnothing \text{ and } Z \neq \varnothing, \\ z, & J \neq \varnothing \text{ and some } x \in Z \text{ satisfies } x > \max J, \\ z+1, & \text{otherwise}. \end{cases} \tag{23}$$

The phase-zero terminal vector has actual minimum degree zero. Every other vector has the actual minimum degree in (23)*.*

Proof. Each coordinate in Z requires a child top, and distinct such coordinates require distinct children. This gives the lower bound z. At a jump c_j, the preceding running label is $r_{j-1} \leq r_j - 2$. A child must therefore supply label $r_j - 1$ at c_j, and its own top marker lies strictly later. If $c_j = \max J$ and no zero lies after it, every coordinate from $c_j + 1$ through C is a desired positive coordinate. A child top at any of those coordinates would block the parent. Hence the jump child must have its top beyond C and is different from the z forced blockers. If $J = Z = \varnothing$, a nonterminal game still requires at least one option. These arguments prove all lower bounds in (23).

They are attained as follows. If $J = \varnothing$ and $Z \neq \varnothing$, use precisely the pure top games A_x for $x \in Z$. The unblocked coordinates then emit the consecutive labels $1, \ldots, d$.

If $J \neq \varnothing$ and $x \in Z$ lies after the last jump, use Theorem A.5 to construct a child K with lower markers $\sigma_{r_j-1}(K) = c_j$ for all $c_j \in J$ and top marker $\sigma_d(K) = x$. Its prescribed labels and

coordinates are strictly increasing, so the vector is formally admissible. Use K together with A_y for $y \in Z \setminus \{x\}$. These z options provide exactly the required jumps and blockers.

In every remaining case, put the jump child's top at $C+1$ instead and use it with all z pure blockers. When no jump occurs, take $K = A_{C+1}$. This uses $z+1$ options and gives the desired scan. The only degree-zero game type is the terminal vector, whose minimum nonterminal degree is one. For $d = 1$, the no-jump cases give degree C when $C \geq 1$ and nonterminal degree one when $C = 0$, as required. □

A.4 Degree enumeration and the representation lower bound

Theorem A.8 (Exact root-degree budget)**.** *For $\Delta \geq 1$, the exact number of response vectors realizable by finite impartial games with at most Δ root options is*

$$B_d(\Delta) = 1 + \sum_{j=1}^{d} \binom{d}{j} \binom{\Delta + j - 1}{j}. \tag{24}$$

For $\Delta = 0$ there is one vector.

Proof. We work with formal power series, so convergence is not needed. Let y mark z, the number of zero coordinates before the top, and put $R(y) = (2-y)/(1-y)$. For k defined labels, the k gaps before successive markers are independent nonnegative integers summing to z. Hence the baseline series, before accounting for an additional child, is

$$A(y) = \sum_{k=1}^{d} \binom{d-1}{k-1} (1-y)^{-k} = \frac{(2-y)^{d-1}}{(1-y)^d}. \tag{25}$$

The no-jump, no-zero vector contributes one exceptional type. For every other type requiring an extra child, let h be the number of selected labels after the final jump. The selected labels end in $d-h, d-h+1, \ldots, d$, with $d-h$ the final jump label and $0 \leq h \leq d-2$. Earlier selected labels form an arbitrary subset of $\{1, \ldots, d-h-2\}$. If there are j of them, the coordinates up to and including the final jump have $j+1$ free zero gaps, while all later gaps vanish. The exceptional-type series is therefore

$$\begin{aligned} E(y) &= 1 + \sum_{h=0}^{d-2} \sum_{j=0}^{d-h-2} \binom{d-h-2}{j} (1-y)^{-j-1} \\ &= 1 + \frac{1}{1-y} \sum_{h=0}^{d-2} R(y)^{d-h-2} = R(y)^{d-1}. \end{aligned} \tag{26}$$

When $d = 1$, both sums are empty and $E(y) = 1$, so the same expression remains valid.

Moving each exceptional type from degree z to degree $z+1$ gives the series for minimum nonterminal degree:

$$A(y) + (y-1)E(y) = yR(y)^d.$$

The terminal vector has actual degree zero instead of one. Thus the series for actual minimum degree is

$$T_d(y) = 1 - y + yR(y)^d.$$

Divide by $1-y$ to count degree at most Δ. Since $R(y) = 1 + (1-y)^{-1}$, for $\Delta \geq 1$ its coefficient is

$$[y^\Delta] \frac{T_d(y)}{1-y} = \sum_{j=0}^{d} \binom{d}{j} \binom{\Delta + j - 1}{j} = 1 + \sum_{j=1}^{d} \binom{d}{j} \binom{\Delta + j - 1}{j}.$$

The constant coefficient is one. The realizations and lower bounds of Theorem A.7 make this a count of attainable response types, rather than merely a count of encodings. □

Define $G \equiv_1 H$ if $u_m(G + *c) = u_m(H + *c)$ for every $c \in \mathbb{N}_0$. The sparse response theorem gives $G \equiv_1 H$ if and only if $\mu_m(G) = \mu_m(H)$. Therefore a fixed-length exact encoding of the response quotient with root-degree budget Δ needs at least $\lceil \log_2 B_d(\Delta) \rceil$ bits. For fixed d,

$$B_d(\Delta) = \frac{\Delta^d}{d!} + O_d(\Delta^{d-1}), \qquad \log_2 B_d(\Delta) = d \log_2 \Delta - \log_2(d!) + O_d(\Delta^{-1}).$$

The coordinate bound provides an encoding in at most $d\lceil \log_2(\Delta + d + 1) \rceil$ bits, including an absent-marker symbol. Thus the leading dependence on Δ is optimal. For arbitrary variable-length encodings without a prefix-free convention, the worst-case lower bound differs by at most an additive constant. These are information bounds for one-heap responses, not computational lower bounds or counts of arbitrary-context equivalence classes. Realizability throughout this appendix is over all finite impartial games and is not restricted to ordinary Nim.

B A COMPLEMENTARY ALGORITHM PARAMETERIZED BY RESIDUAL CORE SIZE

This section concerns an explicitly listed Nim position: each heap is given in binary, but equal heaps are listed individually. It is a different input model and parameter regime from the sparse-multiplicity query theorem in the main text. We allow any fixed normalized phase $i \in \{0, \dots, m-1\}$, with terminal seed $m - 1 - i$ and the same nonterminal maximum recurrence.

For a nonempty input, distinguish a largest heap of size c. Write the remaining position as $R + s \cdot *1$, where every heap in the nonunit core R has size at least two. Define

$$\begin{gathered} A = \sum_{h \in R} h, \qquad D(R) = \sum_{h \text{ a distinct height in } R} h, \\ H = \max\{1, D(R) + \mathbf{1}_{\{s>0\}}\}, \qquad B = H + m - 2. \end{gathered} \tag{27}$$

In particular $B \le A + m - 1$. The empty input is handled directly by the terminal seed.

Lemma B.1 (Observer truncation)**.** *Every positive response marker of $R + s \cdot *1$ lies at a coordinate at most B. Hence if $c > B$, the input has normalized label zero and reducing c to the prefix top is a legal action attaining the maximum child label.*

Proof. The positive-label uniqueness and coordinate-bound arguments above do not depend on the terminal seed. In phase i, the terminal row instead has top coordinate i: starting with $m-1-i$ at zero, its positive labels increase to $m - 1$, and all later coordinates are zero. For a nonterminal prefix, the number of distinct options is at most $D(R) + \mathbf{1}_{\{s>0\}}$. Repeated heaps of the same height create the same possible child multisets and need not be counted repeatedly. The coordinate bound therefore places its top, and every lower marker, at or before B. If the prefix is empty, its top coordinate i is at most $m - 1 = B$, which is why H includes a baseline of one. Beyond the top the response is zero, and reducing the observer to that top reaches label $m - 1$. □

Proposition B.2 (Residual-core dynamic program)**.** *Let $\mathcal{F}(R)$ be the unordered follower closure of R, with Q vertices and E edges. After constructing this graph, the input outcome and an action attaining the maximum child label can be computed in*

$$O\big((s+1)(B+1)(E+Q)\big) \tag{28}$$

word operations using $O(Q(B+1))$ words of live storage. Consequently the problem is fixed-parameter tractable in (m, A) for explicitly listed binary heaps.

Proof. For $T \in \mathcal{F}(R)$, $0 \le t \le s$, and $0 \le z \le B$, compute

$$U[T, t, z] = u_{m,i}(T + t \cdot *1 + *z).$$

Process t in increasing order; within a layer, process T in increasing game rank and z in increasing order. The legal child labels come from

$$U[K, t, z] \quad (K \in \mathrm{Opt}(T)), \qquad U[T, t-1, z] \quad (t > 0), \qquad U[T, t, w] \quad (0 \le w < z).$$

The last group is represented by a running prefix maximum in z and one attaining coordinate. The other groups are available from lower-rank vertices in the current layer or from the previous unit layer. Only $(T, t, z) = (0, 0, 0)$ has no legal move; initialize it to $m - 1 - i$. Duplicate descriptions of the same legal child do not affect a maximum.

Across one unit layer and one observer coordinate, scanning all core edges and updating the prefix maxima costs $O(E + Q)$ operations. This proves (28). Only the current and preceding unit layers are needed. Store an attaining edge, unit deletion, or observer reduction along with each maximum when an action is requested. If $c \le B$, the root entry at (R, s, c) gives the answer directly. If $c > B$, find the root top among the computed coordinates and use Lemma B.1. The table need not contain every marker of every follower: it evaluates their outcomes exactly on the downward-closed observer interval $0 \le z \le B$, which suffices for the root query.

Every follower of R is a partition of mass at most A. If $p(a)$ is the partition function and $\Pi(A) = \sum_{a=0}^{A} p(a)$, then $Q \le \Pi(A)$ and $E \le AQ$. The finite follower graph can be enumerated in time depending only on A, with canonicalization costs absorbed in that dependence. Since the input lists its unit heaps individually, s is at most its encoding length. Also $B \le A + m - 1$. All stored labels and indices have bit lengths bounded in terms of m, A, and the input length, so the word-operation bound gives a bit-time bound of the form $f(m, A)\,\mathrm{poly}(L)$. □

If the residual excess is

$$\varepsilon = \sum_{h \in R} (h - 1),$$

then $A \le 2\varepsilon$ because every nonunit heap satisfies $h \le 2(h - 1)$. Thus the same algorithm is fixed-parameter tractable in the *combined* parameter (m, ε). The bound above does not establish fixed-parameter tractability in ε alone with an unrestricted binary-encoded m. Nor is it polynomial in a binary-encoded unit multiplicity: its factor $s + 1$ uses the explicit-list input convention. These distinctions prevent this complementary dynamic program from being mistaken for the sparse upper-excess compiler of the main text.

C An exact-total family of distinguishable Nim responses

The realizability theorem permits arbitrary finite impartial games. The following separate construction gives a quadratic number of distinguishable responses within ordinary Nim, all at the same total mass. Its player count grows with that mass.

Lemma C.1 (Largest-gap shell)**.** *Let $m \ge 3$. Let R have at least three positive heaps, total mass $m + 1$, and largest heap L. If $M \ge L$ and $1 \le q = M - L + 1 \le m - 1$, then*

$$u_m(R + {*}M) = q.$$

Proof. We first recall the small-mass recurrence. Every positive singleton has label zero because it can move to the terminal top. A position with at least two positive heaps and total $S \le m$ has label $S - 1$. For the latter statement, induct on S: every child is either a singleton of label zero or a smaller multiheap position of label at most $S - 2$. A one-stone move attains $S - 2$, preserving two heaps when $S > 2$; when $S = 2$, either singleton child attains zero. There is no wrap before $S = m + 1$.

Now induct on q. Consider a move in R deleting $\delta \ge 1$ stones. Since $\delta \le L \le M$, the resulting child can reduce the distinguished heap from M to $\delta - 1$. This is a legal move to total mass m. At least two of the original heaps of R remain positive, so the reached position has label $m - 1$. Thus every child obtained by moving in R has label zero.

If the distinguished heap is reduced to $M' < L$, a heap of size L in R can be reduced by $M' + 1$, again reaching total m with at least two other positive heaps. This child also has label zero. If $L \le M' < M$, induction gives its label $M' - L + 1 < q$. For $q = 1$ all children therefore have label zero. For $q > 1$, reducing M by one attains label $q - 1$, and no child exceeds it. The recurrence returns q because $q \le m - 1$. Ties at $M = L$ are included in the base case; the argument distinguishes heap copies only to describe legal moves. □

Theorem C.2 (Quadratic exact-total response family)**.** *For each integer $t \geq 1$, put $m = 8t$ and $S = 12t$. The family*

$$\mathcal{E}_t = \{(x, y, z) : t \leq x \leq 2t, \quad 4t \leq y \leq 5t, \quad z = 12t - x - y\}$$

contains $(t+1)^2$ distinct three-heap Nim positions of total S, pairwise distinguishable by one appended Nim heap. For each member, the first positive response coordinate and its label are

$$c_0 = 8t - x - y + 1, \qquad r_0 = z - y + 1. \tag{29}$$

Proof. The parameter ranges imply $1 \leq x \leq y \leq z$ and $c_0 \geq t + 1 > 0$. At $0 \leq c < c_0$, the position (x, y, z, c) exceeds mass m by $4t + c \leq z$. Deleting exactly that many stones from the heap z reaches mass m, retaining the positive heaps x and y. The reached label is $m - 1$, so the response at c is zero.

At $c = c_0$, the residual triple (x, y, c_0) has total $m + 1$ and maximum y, since

$$y - c_0 = x + 2y - 8t - 1 \geq t - 1 \geq 0.$$

The distinguished heap z satisfies $z \geq y$, and

$$1 \leq z - y + 1 = 12t - x - 2y + 1 \leq 3t + 1 \leq 8t - 1.$$

Lemma C.1 gives the positive label r_0 at c_0. Thus (29) is exactly the first-positive signature, including the endpoint $t = 1$ and the possible ties $c_0 = y$ and $z = y$.

The signature recovers the prefix:

$$z = c_0 + 4t - 1, \qquad y = z - r_0 + 1, \qquad x = 12t - y - z.$$

There are $t + 1$ independent choices of each of x and y. If two first-positive coordinates differ, their smaller coordinate separates the responses. If they agree, distinct prefixes have different labels there. This proves pairwise distinguishability and the count. □

Every heap in this family is below m (in particular $z \leq 7t < 8t$), and $S < 2m$. Thus it lies within the base regime of the query algorithm. The number of response classes is $(t+1)^2 = S^2/144 + O(S)$, but this is a joint-parameter statement with $m = 2S/3$. It establishes neither a superlinear asymptotic lower bound at fixed m nor a running-time lower bound.

An independent defining-recurrence audit, rerun for this draft, checked all residual shell partitions for $3 \leq m \leq 11$ in the stated gap range, including the necessity of the arity restriction, and checked the family for $1 \leq t \leq 4$. It passed 1,816 shell assertions, 394 pre-marker zero checks, 54 first-positive checks, and 462 pair-separation checks. These finite checks audit the formulas and boundaries; the proofs above establish the statements for all parameters.

D THE EXACT SPARSE BASE THROUGH MASS $2m$

Fix $m \geq 3$. Throughout this appendix, a background G has all heap heights below m and total $S \leq 2m$. Multiplicities are binary encoded. The observer x need not be below m.

D.1 LOW MASS AND CAPACITY BEFORE THE FIRST WRAP

A positive singleton has label zero, because it can move to the empty top. By induction on mass, a position with at least two positive heaps and total $R \leq m$ has label $R - 1$: all children have label at most $R - 2$, and a one-stone deletion attains it. At $R = 2$ the resulting singleton has label zero, so the boundary is consistent.

For a position P of total $R > m$ containing at most one heap of size at least m, define

$$\begin{aligned} F_P(c) &= \sum_{h \in P} \min(h, c) - c, \qquad w(P) = R - m - c(P), \\ c(P) &= \min\{c \geq 1 : F_P(c) \geq m + 1\}. \end{aligned} \tag{30}$$

An empty crossing set gives $c(P) = \infty$.

Lemma D.1 (First-ramp capacity)**.** *If $c(P) = \infty$, then $u_m(P) = 0$. If the crossing is finite and $1 \le w(P) \le m$, then $u_m(P) = w(P)$ for $w(P) < m$, and $u_m(P) = 0$ for $w(P) = m$.*

Proof. Choose a largest heap M, and let $B = R - M$. The maximum of F_P is B, so infinite capacity is equivalent to $B \le m$. A singleton reaches the empty top. Otherwise reduce M by $R - m = M + B - m$, which is positive and at most M. The result has total m and at least two positive heaps. If $B < m$, the changed heap remains positive. If $B = m$ and the remainder were a singleton of size m, then both it and M would be at least m, violating the hypothesis. The child has label $m - 1$, giving parent label zero.

Suppose capacity is finite. Then $B \ge m + 1$ and $R - m > M$. Every legal child stays above mass m. Finite crossing is attained by $c(P) \le M$, so every finite-capacity position in this class has $w(P) \ge 1$. Deleting δ stones cannot increase capped sums; hence a finite-capacity child P' satisfies $c(P') \ge c(P)$ and $w(P') \le w(P) - \delta$.

If $w(P) \ge 2$, a one-stone reduction can preserve $c(P)$. Reduce a heap above $c(P)$ if one exists. Otherwise all heaps are at most $c(P)$ and $F_P(c(P)) = R - c(P) = m + w(P) \ge m + 2$, so any one-stone reduction leaves enough slack at the crossing. No earlier crossing appears because capped sums only decrease. This child has $w(P') = w(P) - 1$.

If $w(P) = 1$, no finite-capacity child can exist, since it would have positive $w(P') \le 0$. All children then have label zero. Strong induction on total mass proves the successive labels through $w(P) = m$, where the maximum child label is $m - 1$ and the recurrence wraps to zero. □

This proof supplies actions. Infinite capacity uses the mass-m reduction above. Finite capacity with $w \ge 2$ uses a one-stone capacity-preserving reduction; with $w = 1$, every legal child has label zero.

D.2 Profiles and all positive records

Write distinct background heights as $e_1 < \cdots < e_p = H$ with multiplicities n_j, and put $e_0 = 0$. Define

$$\begin{aligned} A(c) &= \sum_j n_j \min(e_j, c), & D(c) &= A(c) - c, \\ P_j &= \sum_{i<j} n_i e_i, & Q_j &= \sum_{i \ge j} n_i, \\ L_j &= e_{j-1} + 1, & U_j &= e_j. \end{aligned} \tag{31}$$

For $L_j \le c \le U_j$, $A(c) = P_j + Q_j c$ and $D(c) = P_j + (Q_j - 1)c$; the latter expression also holds at $L_j - 1$. Put $D(0) = 0$.

When $S > m$, let $t = \min\{c \ge 1 : A(c) \ge m + 1\}$. For $k_j = Q_j - 1 > 0$, set

$$I_j = [m + 1 - D(U_j),\, m - D(L_j - 1)] \cap [0, t - 1], \qquad c_x = \left\lceil \frac{m + 1 - x - P_j}{k_j} \right\rceil. \tag{32}$$

The function D is nondecreasing on $[0, H]$, because each slope $Q_j - 1$ is nonnegative, and it decreases after H. On a nonempty I_j, c_x is the first c with $D(c) \ge m + 1 - x$. This is precisely the inversion of $D(L_j - 1) < m + 1 - x \le D(U_j)$. Flat profiles $k_j = 0$ cannot contain a new first crossing. The intervals partition the finite inverse-capacity prefix coordinates. Moreover $c_x > x$ when $x < t$: otherwise monotonicity gives $D(x) \ge m + 1 - x$, hence $A(x) \ge m + 1$, contradicting the definition of t.

For low mass the complete positive support is

G	observer interval	$V_G(x)$
$\varnothing$	$x = 0$	$m - 1$
singleton, $1 \le S \le m$	$1 \le x \le m - S$	$x + S - 1$
not singleton, $1 \le S \le m$	$0 \le x \le m - S$	$x + S - 1$.

(33)

Empty intervals are omitted. The low-mass recurrence proves these values; a larger observer reduces to the displayed top, and a singleton at observer zero reduces its background to the terminal.

For $m < S < 2m$, put $r = S - m$. The complete positive records are

$$V_G(x) = \begin{cases} r + x - c_x, & x \in I_j, \\ r + x - t, & t \le x \le t + 2m - 1 - S. \end{cases} \tag{34}$$

For $x < t$, the capacity of $G + *x$ must exceed x, and its crossing equation is $D(c) + x \ge m + 1$; when finite it equals c_x. Lemma D.1 gives the first record. For $x \ge t$, the first capacity is t, and its w rises from r to $m - 1$ over the displayed tail. The top is

$$C = t + 2m - 1 - S. \tag{35}$$

Larger coordinates reduce to C and have value zero. The finite inverse records are positive and disjoint from the tail.

At $S = 2m$, the inverse prefix still has

$$V_G(x) = m + x - c_x \qquad (x \in I_j). \tag{36}$$

If $A(t) \ge m + 2$, then at $x = t - 1$ the first inverse crossing is t, and the response is $m - 1$. Thus $\mathrm{Top}(G) = t - 1$ is already part of the inverse record, and all $x \ge t$ give zero. If $A(t) = m + 1$, the top is delayed and is computed below.

All positive records therefore belong to six types: low diagonal; strict-subdouble inverse; strict-subdouble tail; nonresonant exact-double inverse; resonant exact-double inverse; resonant delayed-top singleton. The inverse records have the form $b + x - \lceil (C - x)/n \rceil$, so (8) produces one quantifier-free linear threshold inequality per record. There are $O(p + 1)$ records.

D.3 Resonant exact-double mass

Assume $S = 2m$ and $A(t) = m + 1$. At least two heaps exceed t: if exactly one did, its height would be $t + (2m - A(t)) = t + m - 1 \ge m$, contradicting the height restriction. Hence $A(t+1) \ge m + 3$, and the inverse prefix gives $c_{t-1} = t + 1$ and $V_G(t - 1) = m - 2$.

Every background move deletes δ stones with $1 \le \delta \le m - 1$, reaching a child Q of strict-subdouble mass. Its top is

$$\tau(Q) = \delta + t_Q - 1, \qquad t_Q = \min\{c \ge 1 : A_Q(c) \ge m + 1\}. \tag{37}$$

Let $Z = \{\tau(Q) : G \to Q\}$. Then the delayed top is

$$T = \min\{x \ge t : x \notin Z\}. \tag{38}$$

For $t \le x < T$, a child top forces zero. At T, no child top occurs, while observer history contains label $m - 2$, so the response is $m - 1$. Later observers reduce to T and have value zero. A one-stone deletion above t gives $t \in Z$. Moreover $\delta, t_Q \le m - 1$, so $Z \subseteq [0, 2m - 3]$ and $t < T \le 2m - 2$.

We compute (38) without enumerating δ or m. Fix a source h, let $a_c = A(c) - m - 1$, and set $\mathrm{loss}_k(\delta) = \min(h, k) - \min(h - \delta, k)$. The exact conditions for child crossing $c > t$ are

$$\mathrm{loss}_{c-1}(\delta) \ge a_{c-1} + 1, \qquad \mathrm{loss}_c(\delta) \le a_c. \tag{39}$$

Solving them yields

$$\begin{cases} h - c + a_{c-1} + 2 \le \delta \le \min(h, h - c + a_c), & c \le h, \\ a_{c-1} + 1 \le \delta \le \min(h, a_c), & c > h. \end{cases} \tag{40}$$

Empty ranges are omitted. At $c = t$, zero-loss deletions are $1 \le \delta \le h - t$ for $h > t$; their top union is $[t, H - 1]$.

On a profile $A(c) = P + Qc$, put $z = c + \delta - 1$. For $c \le h$ and $a_c \le c$, the image block is

$$Qc + h + P - Q - m \le z \le Qc + h + P - m - 2. \tag{41}$$

Across consecutive c, these blocks fill their endpoint interval except residue $h + P - m - 1$ modulo Q. At $Q = 1$ the blocks are empty. A clipped block is nonempty exactly when $1 \le a_c - c \le Q - 2$. Since $a_c - c$ increases by $Q - 1$, at most one clipped c contributes, producing a full interval.

For $c > h$ and $a_c \le h$, the block is

$$(Q+1)c + P - Q - m - 1 \le z \le (Q+1)c + P - m - 2. \tag{42}$$

Its consecutive union misses residue $P - m - 1$ modulo $Q+1$. The clipped condition $1 \le a_c - h \le Q - 1$ again allows at most one contributing c, because $a_c - h$ increases by Q. Splitting sparse profiles at h and at these clipping thresholds gives $O(p^2)$ full or one-missing-residue pieces.

Sort the piece endpoints within $[t, 2m-2]$, using right endpoints plus one. Active pieces are fixed in each cell. A full piece covers the cell. Otherwise a point outside their union must lie in every active missing residue. Generalized CRT returns one progression or the empty set, including incompatible non-coprime cases. One ceiling division finds the first point of a nonempty progression in the cell. The first successful cell gives T.

A conservative implementation uses $O(p^4)$ big-integer/CRT operations. Combined modulus length is at most the sum of active modulus lengths; all quantities have polynomial bit length. Thus this is sparse bit-polynomial preprocessing rather than an arithmetic model that expands a numeric period.

The positive support is the inverse prefix and the singleton $(T, m-1)$. The covering-piece inverse is explicit. In (41), put $\alpha = h + P - Q - m$; for a nonempty ordinary block ($Q \ge 2$), recover $c = \lfloor (z - \alpha)/Q \rfloor$. In (42), put $\gamma = P - Q - m - 1$ and recover $c = \lfloor (z - \gamma)/(Q+1) \rfloor$. A clipped or $c = t$ piece stores its single threshold c. In all cases set $\delta = z - c + 1$.

For actions at $x < t$, use the capacity-lemma move. For $t \le x < T$, apply that inverse with $z = x$ and reduce its stored source by δ. The defining inequalities guarantee legality and child label $m-1$. At $x = T$, reduce the observer to $t-1$, attaining $m-2$; beyond T, reduce it to T.

D.4 Top, requested labels, and thresholds at base roots

Interval records are disjoint and contain all positive coordinates. Their scan returns the top; a requested positive label is located by inverting the relevant monotone affine or inverse-ceiling record and checking equality. If no record contains it, return $\perp$. Threshold decisions use the record bounds and (8); a satisfying record retains its exact value and production branch. A parent invoking a base-child threshold owns the edge to that child, not an action internal to it.

For a base-root action, use the low-mass or capacity move, the resonant covering-piece inverse, or an observer reduction to a known earlier top as specified above. The only no-move case is the truly empty position. These constructions close the base interface required by the upper-excess induction.

E Upper-excess query compilation: detailed proofs

Throughout this appendix, $m \ge 3$, the terminal label is $m-1$, and

$$u_m(P) = \big(1 + \max_{P \to Q} u_m(Q)\big) \bmod m$$

for nonterminal positions. All background heaps are strictly smaller than m. For a background P, write $S(P)$ for its total mass, $A_P(c) = \sum_{h \in P} \min(h, c)$, and $V_P(x) = u_m(P + *x)$. Multiplicity is included in this sum, although equal source heights give only one anonymous move type for each target height. The input root has p distinct heights and $S(G) \le 2m + q$, where $q \ge 0$ is an integer. Set $N = p + q + 2$ and let Λ be its complete binary encoding length. The supplied q may first be replaced by $\max(0, S(G) - 2m)$.

We use the base-band formulas proved in the preceding appendix in precisely the following form. If $m < S(Q) < 2m$, put $e = S(Q) - m$ and $t_Q = \min\{c \ge 1 : A_Q(c) \ge m + 1\}$. For $x < t_Q$, a finite inverse capacity

$$c_x = \min\{c \ge 1 : A_Q(c) - c \ge m + 1 - x\}$$

gives $V_Q(x) = e + x - c_x$; otherwise that prefix value is zero. Its tail is

$$V_Q(x) = e + x - t_Q \quad (t_Q \le x \le t_Q + m - 1 - e),$$

and every later value is zero. If $S(Q) = 2m$, the same inverse prefix has additive constant m and is valid for every $x < t_Q$, both in the resonant and nonresonant cases. The base response has $O(s+1)$ disjoint affine or inverse-ceiling records at support size s, with quantifier-free linear positive-threshold predicates, an exact top, and constructive root queries. Only these established base statements are used below.

E.1 IMMEDIATE-SHALLOWER SHIELDING AT EVERY POSITIVE UPPER EXCESS

Lemma (shielding). Let $S(P) = 2m + r$ with $r \geq 1$. Fix a present source h and a legal deletion $d \geq r + 1$. Put $y = h - d$, $Q = P - *h + *y$, and $Q^+ = P - *h + *(y + 1)$, omitting zero heaps. Then

$$1 \leq V_Q(x) \leq m - 3 \quad \Longrightarrow \quad V_{Q^+}(x) \geq V_Q(x). \tag{43}$$

There is no additional hypothesis that a capacity threshold exceed one.

Proof. Since $d \leq h \leq m-1$, both children have mass greater than m; Q has mass at most $2m-1$ and Q^+ has mass at most $2m$. The deletion $d-1$ defining Q^+ is legal because $d \geq r+1 \geq 2$. Write $A = A_Q$, $A^+ = A_{Q^+}$ and let their thresholds be t, t^+. Increasing one target by one gives

$$A(c) \leq A^+(c) \leq A(c) + 1$$

at every integer $c \geq 1$, so $t^+ \leq t$. If $t^+ < t$, threshold minimality forces $A(t^+) = m$ and $A^+(t^+) = m + 1$. As $S(Q) > m$, some heap of Q exceeds t^+; thus $A(t^+ + 1) \geq m + 1$. Consequently

$$t^+ \in \{t, t - 1\}. \tag{44}$$

First assume $d \geq r+2$, so both siblings are strict-subdouble. Put $e = m+r-d$. Their tail endpoints are

$$C = t + d - r - 1, \qquad C^+ = t^+ + d - r - 2.$$

If a low event in (43) lies in the tail of Q, then $e + x - t \leq m - 3$ implies $x \leq C - 2$. By (44), $x \leq C^+$, and $x \geq t \geq t^+$. It therefore lies in the tail of Q^+ and

$$V_{Q^+}(x) - V_Q(x) = 1 + t - t^+ \geq 1.$$

If instead $x < t$, the positive event has finite inverse capacity $c_x > x$. When $x < t^+$, pointwise $A^+(c) - c \geq A(c) - c$ gives $c_x^+ \leq c_x$, and the two inverse formulas give

$$V_{Q^+}(x) = e + 1 + x - c_x^+ \geq V_Q(x) + 1.$$

The only remaining prefix case is $t^+ = t - 1$ and $x = t - 1$. This is the start of the Q^+ tail, whose value is $e + 1$; on the other hand, $c_x > x$ implies $V_Q(x) \leq e - 1$. This proves the interior case.

Now assume $d = r + 1$. Then $S(Q) = 2m - 1$ and $S(Q^+) = 2m$. The tail of Q contains only its top at t, so every event under consideration is in its prefix. If $x < t^+$, the strict and exact-double inverse formulas give

$$V_{Q^+}(x) = m + x - c_x^+ \geq m + x - c_x = V_Q(x) + 1.$$

Again the only other formal possibility is $t^+ = t - 1$, $x = t - 1$. In this case $t^+ \geq 1$ itself implies $t \geq 2$, and the threshold drop gives $A_Q(t-1) = m$. The mass above cap $t - 1$ is $m - 1$. It cannot all belong to one heap, since that heap would have height $(t - 1) + (m - 1) = m + t - 2 \geq m$. At least two heaps exceed $t - 1$, whence $A_Q(t) \geq m + 2$. The inverse threshold at $x = t - 1$ is exactly t: $A_Q(t-1) - (t-1) = m - t + 1$ is too small, while $A_Q(t) - t \geq m + 2 - t$ reaches the required level. Hence $V_Q(t-1) = m - 2$, contrary to the low-event premise. This completes the proof, including threshold-one backgrounds. □

Corollary (the only retained large events). For an interior deletion $d \geq r + 2$, the only event of value $m - 2$ is

$$t_Q + d - r - 2 = \mathrm{Top}(Q) - 1. \tag{45}$$

For the boundary deletion $d = r + 1$, such an event exists exactly when

$$A_Q(t_Q) \geq m + 2; \tag{46}$$

its coordinate is then $t_Q - 1$.

Proof. In the interior case $e = m + r - d \leq m - 2$. Every prefix value is at most $e - 1 \leq m - 3$, while the last two tail labels are $m - 2, m - 1$. At the boundary $e = m - 1$, the tail consists only of the top. In the prefix, label $m - 2$ requires $c_x = x + 1$. If $x \leq t_Q - 2$, minimality of t_Q gives $A_Q(x + 1) \leq m$, so this is impossible. At $x = t_Q - 1$, the condition $c_x = t_Q$ is exactly (46). □

The capacity in (46) belongs to the child. For example, when $m = 3$, $P = (1, 1, 1, 2, 2)$ and the boundary move is $2 \to 0$, the parent has $A_P(1) = 5$ but the child $Q = (1, 1, 1, 2)$ has $A_Q(1) = 4$ and $V_Q(0) = 0$. Testing the parent's overshoot would create a false trigger.

E.2 EXACT LARGE-FAMILY CLIPPING, IMAGES, AND INVERSION

Fix an upper state P of excess r and support size s, and a source h. If $h \le r$, the large family is empty. Otherwise put

$$B = P - *h, \quad H = h - r - 1, \quad C_0 = h - r - 1, \quad Q_y = B + *y \quad (0 \le y \le H).$$

The strict-subdouble top formula gives

$$T(y) = \mathrm{Top}(Q_y) = C_0 - y + t(y), \qquad t(y) = \min\{c \ge 1 : A_B(c) + \min(y, c) \ge m + 1\}. \tag{47}$$

Let $t_P = \min\{c \ge 1 : A_P(c) \ge m + 1\}$.

Lemma (exact target intervals). Targets $y \in [\max(0, t_P), H]$ have $t(y) = t_P$ and yield one full image interval. For $y < t_P$, use one positive-slope profile $A_B(c) = \beta + nc$ on $L \le c \le R$, also valid at $L - 1$, and put $K = m + 1 - \beta$. Exactly the targets whose first threshold belongs to that profile form the interval

$$\max(0, K - nR) \le y \le \min\left(H, t_P - 1, K - n(L-1) - 1, \left\lfloor \frac{K-1}{n+1} \right\rfloor\right). \tag{48}$$

An empty interval is omitted. On a nonempty interval, $t(y) = \lceil (K - y)/n \rceil$.

Proof. If $y \ge t_P$, both the original source h and the target contribute c at every $c \le t_P$, so the child capped sum equals the parent's there. Its first threshold is exactly t_P. If $y < t_P$, the child and parent capped sums agree at $c = y$ and are at most m, so $t(y) > y$. Also $S(B) = 2m + r - h \ge m + r + 1 \ge m + 2$. A threshold exists within the positive-slope profiles of A_B; a final flat profile is unnecessary. For a proposed first crossing $c > y$, the condition is $A_B(c) + y \ge m + 1$, or $\beta + nc + y \ge m + 1$ on this profile. Routing its first crossing into $[L, R]$ is precisely

$$\beta + n(L-1) + y < m + 1 \le \beta + nR + y.$$

These are the bounds $K - nR \le y \le K - n(L-1) - 1$. The additional requirement $\lceil (K-y)/n \rceil > y$ is equivalent, for integer y, to $K - y > ny$, or $y \le \lfloor (K-1)/(n+1) \rfloor$. Intersecting these conditions with $0 \le y \le H$ and $y < t_P$ proves (48), including both endpoints. □

Lemma (one missing residue). On a nonempty interval in (48), set $z = K - y$. Then

$$T(y) = C_0 - K + z + \lceil z/n \rceil. \tag{49}$$

Its image is the interval between its endpoint values with exactly the residue $C_0 - K + 1 \pmod{n+1}$ removed. In particular, T is strictly decreasing in y.

Proof. Use Euclidean division $z = an + b$ with $0 \le b < n$, also for negative integers. If $b = 0$, then $z + \lceil z/n \rceil = (n+1)a$. If $1 \le b < n$, its value is $(n+1)a + b + 1$. Thus its image on all integers is exactly the complement of residue one modulo $n + 1$, in increasing order. Consecutive image values differ by one or two. Restricting a strictly increasing sequence of this form to a consecutive input interval removes no additional interior points beyond that residue class. Adding $C_0 - K$ proves the assertion and strict monotonicity. □

There are at most s positive-slope profiles of A_B and one constant threshold interval, so one source produces at most $s + 1$ top pieces. Retain the source and its target interval with each piece. Membership in a piece guarantees a unique inverse target: binary search the strictly decreasing map (47) on that interval and verify its value. This costs polynomially many binary arithmetic operations, not a scan of targets. Overlap between pieces from different sources causes no problem; any covering source supplies a legal action.

Interior triggers are obtained by running the same compiler with target upper bound $H - 1$ and shifting the images by -1. Their inverse adds one to the queried trigger coordinate and uses the top inverse. The boundary $y = H$ contributes the single point in (46), when that test holds, with its stored source and target. Therefore all large top and trigger data use $O(s^2)$ pieces. Sparse prefix sums, binary threshold search, exact integer division, and sorting suffice to construct them. No multiplicity, source height, or residue period is expanded.

E.3 EXACT RETAINED ENVELOPES AND OPEN-RANK VIEWS

Small deletions satisfy $1 \le d \le \min(r, h)$. Their children have strictly smaller excess and will be fully prepared by induction. Store each small child's top as a singleton. Together with the large top pieces these represent exactly every background-child top. Their count is at most

$$K_P \le sr + s(s+1) = O(N^2). \tag{50}$$

A coordinate is *open* when it is not in this union. On an open coordinate every background child has label at most $m-2$.

Lemma (retained envelope). On an open coordinate w, let $E_P(w)$ be the maximum over all legal background-child responses. For every integer a its threshold is exactly

$$\mathsf{Child}_P(w,a) = \begin{cases} \text{true}, & a \le 0, \\ \bigvee\limits_{Q \text{ small}} [V_Q(w) \ge a] \ \vee \bigvee\limits_{J \text{ large trigger}} [w \in J], & 1 \le a \le m-2, \\ \text{false}, & a \ge m-1. \end{cases} \tag{51}$$

The true nonpositive branch can retain a fixed legal one-stone parent edge.

Proof. All displayed alternatives correspond to real child values. Conversely, suppose a large child has a positive value at most $m-3$ at w. The shielding lemma supplies a dominating immediately shallower sibling. If this sibling is a top, w is blocked, a contradiction. If its value is $m-2$, it is retained by the trigger classification, or lies in a small child. Otherwise repeat the comparison while its deletion is large. The deletion decreases each time and the chain stops no later than $d = r$, a legal retained small deletion. Thus discarded positive events never increase the maximum. Zero values do not affect a positive threshold, and nonpositivity is true because the nonempty upper background has a legal child and every response is nonnegative. □

For every nonempty state, the sparse-marker coordinate bound gives

$$\mathrm{Top}(P) \le D(P) + m - 2 \le S(P) + m - 2 \le B_P := S(P)(m-1).$$

Here $D(P)$ is the sum of distinct positive heights, and the last inequality uses $S(P) \ge 1$. Work on $[0, B_P]$ and split it at all top-piece left endpoints and right endpoints plus one. There are at most $2K_P + 2$ cells. Clip pieces to the working interval first.

In a cell with an active full piece there are no open points. Otherwise a point is open exactly when it lies in the missing residue of every active holey piece. Generalized CRT either gives an inconsistent system or one residue class modulo a positive least common multiple. With no active pieces, use modulus one. Intersect with the cell. An empty intersection is stored as a zero-open cell. A nonempty intersection is written

$$f + Mk, \qquad 0 \le k < n_C,$$

where f is its first point. Store also b_C, the number of open points strictly before the cell. Prefix rank is important even in zero-open cells.

The following views represent arbitrary current coordinates without enumeration. On a zero-open cell $[L_C, U_C]$, rank is the constant b_C. On the prefix $[L_C, f-1]$ of a nonempty cell it is also b_C. On the remaining interval introduce one integer quotient k and impose

$$\begin{aligned} &L_C \le X \le U_C, \quad 0 \le k < n_C, \\ &f + Mk \le X \le f + M(k+1) - 1, \qquad R_P(X) = b_C + k + 1. \end{aligned} \tag{52}$$

The equality for R_P means syntactic substitution of that affine expression, not a new variable. These views cover every coordinate exactly once. For an open current-coordinate view, replace the floor inequalities by $X = f + Mk$, using the same quotient for rank and openness. Splitting the short prefixes at most doubles the number of views, so it remains $O(N^2)$.

A witness open cell introduces one quotient u, substitutes $w = f + Mu$, and uses $R_P(w) = b_C + u + 1$ with $0 \le u < n_C$. Membership in a full trigger uses only endpoint inequalities. For a holey trigger $[L_J, U_J] \setminus \{x : x \equiv \rho \pmod M\}$, normalize $0 \le \rho < M$ and use

$$L_J \le X \le U_J, \qquad \rho + Mv + 1 \le X \le \rho + M(v+1) - 1 \tag{53}$$

for one integer v. Modulus-one holey pieces are empty and are omitted. For $M \ge 2$, the intervals in (53) are precisely the integers between consecutive excluded residues, so the encoding is exact.

E.4 Raw history and the acyclic preparation algorithm

List the open coordinates as $z_1 < z_2 < \cdots$, and put $e_i = E_P(z_i) \le m-2$. Define the unwrapped history

$$H_0 = 0, \qquad H_i = 1 + \max(H_{i-1}, e_i).$$

Induction gives

$$H_i = i + \max\big(0, \max_{1\le j\le i}(e_j - j + 1)\big). \tag{54}$$

Indeed, substituting the formula for H_{i-1} in the recurrence gives the displayed maximum with its new $j = i$ term. Put $H_P(x) = H_{R_P(x)}$ and handle $H_P(-1) = 0$ as a constant convention. For $a > 0$, (54) is equivalent to

$$H_P(x) \ge a \iff R_P(x) \ge a \ \vee\ \exists \text{ open } w \le x : \mathsf{Child}_P\big(w, R_P(w) + a - R_P(x) - 1\big). \tag{55}$$

If $R_P(x) < a$, a maximum term in (54) reaches a exactly under the second alternative; if $R_P(x) \ge a$ the first alternative suffices. This proves the equivalence without any assumption on regularity of the child envelope.

The first raw-history crossing of $m-1$ is open and equals $m-1$. Before it, both prior history and the open child value are at most $m-2$, so one step cannot overshoot $m-1$. The sparse-marker theorem and the observer maximum identify this crossing with the parent top: blocked coordinates emit zero, open coordinates before it emit their raw history, and after it an observer reduction reaches label $m-1$. For $1 \le a \le m-1$, therefore,

$$V_P(x) \ge a \iff 0 \le x \le \mathrm{Top}(P), \quad x \text{ open}, \quad H_P(x) \ge a. \tag{56}$$

Use the constant true answer for $a \le 0$ and false for $a \ge m$.

Prepare states in increasing upper excess. The base has $S(P) \le 2m$. At an upper state, prepare only its small children, compile the large families by (48), build the rank views, construct (55), and binary-search its first crossing on $[0, B_P]$ to freeze the top. Only then construct (56). The current raw-history formula uses strict-child response formulas, not the current top. The order

$$\text{strict children} \longrightarrow \text{pieces and ranks} \longrightarrow \text{raw history} \longrightarrow \text{current top} \longrightarrow \text{current response}$$

is acyclic. This is a construction from raw sparse input; no upper-state top oracle is a mathematical premise.

E.5 Complete branch-count and dimension ledger

Along a small-deletion path the excess decreases by at least one and the support increases by at most one per edge. Thus path length is at most q, support is at most $p + q < N$, and each node has at most qN small children. Even without memoization, the number of state nodes is bounded by

$$A_q \le (1 + qN)^{q+1}. \tag{57}$$

This follows from $A_0 = 1$ and $A_j \le 1 + qNA_{j-1}$. Every such state has mass at most the root's and heights below m.

For each prepared state, retain an internal two-formal-parameter template

$$\mathsf{Resp}_P(X, A) = \bigvee_b \exists z_b \in \mathbb{Z}^{d_b} : \ell_b \le z_b \le u_b, \quad C_b z_b + c_b X + d_b' A \le e_b, \tag{58}$$

allowing linear equalities as well. X, A are formal inputs, not existential variables. A concrete public query substitutes integers. A parent substitutes affine expressions. All discrete choices are expanded as DNF alternatives. Keep the branch's state, predicate kind, and parent-edge information with the inequalities.

Let M_j bound the flattened branch count of a response template and of an upper raw-history template at excess at most j. At $j = 0$ only the base response interface is needed, and $M_0 \le c_0 N$. Let $C_j = O(N^2)$ count current-rank views, $J_j = O(N^2)$ count trigger pieces, and $S_j \le qN$ count small children. A history branch makes exactly the following choices:

1. One current rank view. A direct rank-baseline branch then stops.

2. Otherwise, one witness open cell, with its quotient and $w \le X$.
3. One nonpositive child branch, one retained trigger piece, or one small child followed by one branch of its prepared response.

The resulting count, including constant threshold guards, is at most

$$\begin{aligned} F_j &\le cC_j + cC_j^2(1 + J_j + S_j M_{j-1}) \\ &\le c_3 N^6 + c_3 q N^5 M_{j-1}. \end{aligned} \tag{59}$$

For a response template its chosen open current view already supplies the rank quotient; the construction does not choose a second independent current-rank view. The top cutoff adds a constant number of inequalities. Thus it obeys the same bound. A direct current-child threshold uses no more choices than this count. This accounts for every factor: N^4 from the two views, N^2 from terminal trigger choices, or qN times the lower branch count from small-child choices.

Since $q \le N$ and $M_{j-1} \ge 1$, increasing an absolute constant gives

$$M_j \le c_4 N^6 M_{j-1}, \qquad M_j \le c_0 N (c_4 N^6)^j. \tag{60}$$

The uniform informal form is $N^{O(q+1)}$, which also covers $q = 0$. Because $1 + qN \le 2N^2$, combining (57) and (60) yields

$$A_q M_q \le 2^{q+1} c_0 c_4^q N^{8q+3}. \tag{61}$$

This is a safe eager materialization bound, not a claim that lazy storage must attain the product.

For dimension, a layer contributes at most one current-coordinate/rank quotient and one witness quotient. Write its witness as $w = f + Mu$ and its rank as $b_C + u + 1$; neither w nor its rank becomes another variable. Substitution into a small child keeps the two parent variables already counted and introduces only the child's own quotients. A holey-trigger branch contributes one extra quotient and terminates; it cannot also recurse into a small child. A full trigger and a nonpositive child add none, and the base is quantifier-free. Consequently

$$D_0 = 0, \qquad D_j \le 2 + \max(1, D_{j-1}), \qquad D_j \le 2j + 1 \quad (j \ge 1).$$

For a path visiting k upper layers, the four termination types have at most $2k - 1$, $2k$, $2k$, and $2k + 1$ variables, respectively: direct rank baseline, nonpositive child, base/full trigger, and holey trigger. No selector, coordinate, per-blocker CRT, or ceiling variable is hidden in this count. Each layer contributes a constant number of linear rows: view endpoints, quotient bounds, the witness-order inequality, threshold guards, and possibly the top cutoff. Terminal membership contributes a constant number and the base has a constant number per record. Thus one branch has $O(j + 1)$ rows.

E.6 Signed bounds, coefficient growth, and bit complexity

Set $U = S(G)(m - 1)$ for a nonempty root; the empty root is handled by the base. It bounds the working coordinate of every upper state. An open or floor-rank quotient in (52) has the explicit interval $0 \le k \le n_C - 1$; zero-open and short-prefix branches introduce no quotient. Witness quotients have the same kind of interval. In (53), $0 \le X \le U$ and $0 \le \rho < M$. Every nonexcluded X has a unique quotient between -1 and $U + 1$: the gap containing X is the one following the preceding excluded residue. Thus adding

$$-1 \le v \le U + 1$$

does not remove any solution. The same bound can be used after affine substitution, since the original coordinate guards remain in the branch. All lower-state variables retain their previously explicit bounds. Every emitted existential variable therefore has a finite signed binary interval; no search over that interval is part of formula construction.

Put $\lambda = \Lambda + \lceil \log_2(q + 2) \rceil + 1$. Along a recursion path, support grows by at most q, multiplicities change by at most q, and heights and total mass do not increase. All individual state fields and sparse capped-profile sums therefore have $O(\lambda)$ bits. Primitive large-piece endpoints and moduli are formed by a bounded number of products, sums, differences, and exact divisions of such integers, and have $O(\lambda)$ bits. CRT combines at most $O(N^2)$ primitive moduli on a cell. Since

$$\log_2 \operatorname{lcm}(M_1, \dots, M_t) \le \sum_{i=1}^{t} \log_2 M_i,$$

its normalized residue and step have $O(N^2\lambda)$ bits. Counts and prefix ranks require no more. Thus a common primitive bound is $\mu = cN^2\lambda$ for an absolute c.

Consider a lower-template row

$$c_0 + \sum_i c_i z_i + c_X X + c_A A \le 0.$$

Substitute $X = \alpha_0 + \sum_i \alpha_i u_i$ and $A = \beta_0 + \sum_i \beta_i u_i$, where the parent coefficients have at most μ bits. Each resulting coefficient is a sum of the form

$$c_i + c_X\alpha_i + c_A\beta_i.$$

Multiplication adds bit lengths, and a signed sum adds only logarithmically many carry bits. There is no product of two unknowns. At layer j the coefficient bound therefore satisfies

$$b_j \le b_{j-1} + O(\mu + \log(j+2)), \qquad b_j = O\big((j+1)N^2(\Lambda + \log(q+2))\big).$$

Equalities obey the same calculation. Native current-layer constraints have at most $\mu + O(1)$ bits, and duplicating rows across alternatives changes the branch count but not this bit bound. The inherited finite bounds are also covered. In particular, with

$$\mathrm{sbit}(a) = 1 + \lceil \log_2(|a| + 1) \rceil,$$

one may take the safe witness bound

$$W_j = \max_i \{\mathrm{sbit}(\ell_i), \mathrm{sbit}(u_i), \lceil \log_2(u_i - \ell_i + 1) \rceil\} = O(b_j + \log(U+1) + 1),$$

with maximum zero for a variable-free branch.

If the concrete query arguments have η bits, substituting them changes the row encoding length by only a polynomial in $L = \Lambda + \eta + 1$, with an absolute exponent. One conjunction has $O((q+1)^2)$ coefficient entries, each bounded as above. Its total bit length is therefore $\mathrm{poly}(q, N)L$ with absolute polynomial degree. The same is true of building a branch: sparse sorting, gcd, extended gcd, multiplication, and ceiling division have polynomial bit cost on the already bounded integers. No numeric loop is needed over a heap height, multiplicity, observer interval, or residue period.

Apply deterministic fixed-variable integer programming separately to each conjunction. Its input-length polynomial has an absolute exponent; its dimension-dependent factor depends only on $2q+1$. Computing the top of each state adds $O(\log(U+1))$ whole-DNF decisions, and the derived queries below add only bit-length searches and bounded witness recovery. Multiplying these costs by (61) gives absolute constants C_0, C_1 and a computable function g such that preparation and one query take

$$g(q)N^{8q+C_0}L^{C_1}$$

bit operations. This proves uniform XP in upper excess and polynomial bit time for each fixed excess. It does not give FPT in q alone. The combined-parameter corollary follows by $N \le (p+1)(q+2)$; with the conservative classical $2^{O(d^3)}$ fixed-variable dimension factor, the parameter $\kappa = q + \lceil \log_2(p+1) \rceil$ has the safe envelope $2^{O((\kappa+1)^3)}\,\mathrm{poly}(L)$.

E.7 Constructive values, labels, and actions

First, a feasibility-only integer-programming procedure is sufficient. In a feasible branch with explicit bounds $\ell_i \le z_i \le u_i$, the predicate "the branch remains feasible after adding $z_i \le t$" is monotone nondecreasing in t, also for negative endpoints. Binary-search its least true integer, fix z_i to that integer, and continue to the next coordinate. Each fixing preserves feasibility. The result is the lexicographically least integer solution, using at most

$$\sum_i \lceil \log_2(u_i - \ell_i + 1) \rceil \le D_q W_q$$

additional conjunction-feasibility calls. Empty bounds are rejected before solving. This is bit self-reduction, not enumeration of the integer box.

Top search uses only the monotone raw-history predicate. A value query first checks the top cutoff and openness and then binary-searches labels. For $1 \le \ell \le m-1$, a requested-label query finds

the first raw crossing of ℓ and tests exact equality, openness, and the cutoff. The equality check is necessary because history may jump over a label. The base-root case instead uses its exact record scan and inversion.

For action recovery, retain with a successful branch its state identifier, predicate kind, integer assignment, and the outermost parent-edge token. A small-child alternative certifies the value of the child reached by that edge; an action internal to the child is not an action at the parent. A large top or trigger uses the piece inverse already proved above. The following procedure exhausts all upper-state coordinates:

1. If $x > \mathrm{Top}(P)$, reduce the observer to $\mathrm{Top}(P)$. The reached label is $m-1$, the largest possible child label.
2. If $x \le \mathrm{Top}(P)$ is blocked, select a covering small-child top singleton or large top piece. Its stored or inverted background action reaches label $m-1$ at the current coordinate.
3. Otherwise x is open and $x \le \mathrm{Top}(P)$, including equality. Compute exactly $e = E_P(x)$ using (51), and $b = H_P(x-1)$ using raw history, with $H_P(-1) = 0$. Before the top, actual earlier positive responses equal raw history, so b is the maximum earlier observer-child label, and the maximum of all legal child labels is $\max(e, b)$.
4. If $b > 0$ and $b \ge e$, separately binary-search the first $z < x$ satisfying $H_P(z) \ge b$. Since b is itself an attained prior-history value, that first crossing is open and equals b. Return the observer action $x \to z$.
5. If $e > b$, retain or rerun a successful *current-child* branch at (P, x, e). Its outermost edge or large-piece inverse reaches label at least e, and by definition exactly e.
6. If $e = b = 0$, use a fixed legal one-stone background edge. Every background child is then zero, so this edge is maximizing.

This proves both legality and maximality. The terminal query has no edge and is handled by the base interface with the answer "no move."

It is essential that step 4 search for the exact prior maximum, rather than reuse an arbitrary witness w in (55). For $m = 5$ and $P = 1^{11}$, the response at $x = 0, \dots, 5$ is $(0, 1, 2, 3, 4, 0)$. The only background child is 1^{10}, with response $(4, 0, 0, 0, 0)$ at $x = 0, \dots, 4$, so $R_P(x) = x$ for $x \ge 1$. At $x = 3$, level $a = 3$, the existential formula accepts $w = 1$ because its generated child threshold is $1 + 3 - 3 - 1 = 0$. Yet the observer move $3 \to 1$ reaches only label one, while $3 \to 2$ reaches the maximizing label two. Thus a threshold certificate alone does not determine an action. The procedure above uses $b = H_P(2) = 2$ and correctly returns $3 \to 2$.

For completeness, write $L_B = \lceil \log_2(B_P + 1) \rceil$ and $L_m = \lceil \log_2 m \rceil$. Preparing one top uses at most $L_B + 1$ raw-history decisions. With that top cached, safe whole-DNF decision bounds are $L_m + 1$ for a value, $L_B + 2$ for a requested label, and $2L_m + L_B + O(1)$ for an action. Every such decision scans at most M_q conjunctions, and witness recovery adds at most $D_q W_q$ feasibility calls on a selected branch. These are decision counts, not claims of logarithmic total running time. The resulting cost is already covered by the factored XP bound above.

F Computational evidence, regressions, and implementation boundaries

This appendix distinguishes three evidence types. A finite recurrence check can compare every legal option in a declared domain. An arithmetic interface check validates transformations and witness inverses on selected inputs. A large-integer solver certificate validates its returned assignment, but is not an exhaustive game-tree oracle. None replaces the all-input proofs.

F.1 Literal reference semantics

Canonical positions are sorted multisets of positive integers. The reference evaluator memoizes (1) and generates children by replacing one positive heap h by every integer in $\{0, \dots, h-1\}$, deleting zeros and identifying equal child multisets. For each returned action it checks legality and

recomputes the largest label among all legal children. Positive-label coordinates are checked by scanning the finite observer horizon justified by the marker bound.

The base suite covers every bounded-part background with $m = 3, \ldots, 11$, mass at most $2m$, and maximum height below m. Observers range from 0 through $2m + 1$, and thresholds from -2 through $m + 1$. It records 8,958 backgrounds, 15,754 positive records, 2,714,852 threshold checks, and 543,484 inverse-ceiling linearization checks without a mismatch.

The upper definition-level suite covers every bounded-part background with $m = 3, \ldots, 7$, exact excess $1, \ldots, 4$, and maximum height below m. It records 1,148 roots, 94,989 response-coordinate and maximizing-action checks, 808,835 threshold checks, 89,042 retained-envelope equalities, and 6,019 positive-label checks. Its structural checks include 2,939 shielded low events, 2,272 large-child top formulas, 1,217 holey images, 422 constant full images, 1,105 interior triggers, and 1,167 boundary triggers. These all passed.

The finite DNF evidence consists of seven selected complete formulas, totaling 10,437 branches and 18,353,876 finite free-assignment tests. It is not exhaustive over all upper-excess states. The second-stage semantic suite separately includes all 46 exact-excess-three/four roots for $m = 3, 4$, totaling 95 DAG states; three selected $m = 5$ roots are additional cases, not exhaustive coverage.

F.2 Structural checks

Separate random-DAG and constructed-vector programs rechecked the sparse scan and realizability construction. The former records 232,922 marker assertions, 245,637 direct states, 456 two-player SG comparisons, and 35,280 line-sparsity checks. The latter records 2,875 formal-vector/minimum-degree checks, 198 pure-top rows, 1,341 isolator windows, and 58,008 direct cells. Random DAGs are sampled, not an enumeration of all finite games.

The degree enumerator separately checks 12,375 formal types, 55 independently minimized small envelopes, 1,789 envelope states, 432 phase-aggregate coefficients, and 32 exact rational generating-function identities. The exact-total Nim family in Appendix C has finite checks of 1,816 shell instances, 394 zero-prefix coordinates, 54 family members, and 462 pair-separation comparisons. These test small boundaries; the family theorem follows from its shell proof for every integer parameter.

The residual-core algorithm was checked for all phases at $m = 2, \ldots, 7$, nonunit-core partitions through mass seven, and unit counts through six. It records 2,835 root rows, 1,342,809 follower-layer cell comparisons, 37,653 direct comparisons, and 11,340 oversized-heap zero checks.

F.3 Regression cases that constrain the proof interface

The following small examples are retained because the wrong abstraction can pass simpler tests.

1. *Marker noncomposition.* At $m = 3$, $(1, 1, 1)$ and $(1, 2)$ have equal signatures, but adding one unit heap yields $(1, 2)$ and $(2, 3)$. Adding two units gives scalar labels 1 and 0.
2. *Boundary overshoot.* At $m = 3$, parent $(1, 1, 1, 2, 2)$ and boundary child $(1, 1, 1, 2)$ have different capacity overshoots. Testing the parent creates a false penultimate-label trigger at observer zero.
3. *Threshold evidence is not an action target.* At $m = 5$, background 1^{11} has $V(1) = 1$, $V(2) = 2, V(3) = 3$. A raw-history witness at 1 can certify the threshold at 3, but reduction to 1 is not maximizing. Exact prior-history attainment must be recovered separately.
4. *Empty-prefix phase exception.* In normalized phase i, the empty-prefix top is i. At $m = 2, i = 1$ it is one, so the nonterminal bound $S + m - 2$ cannot be applied to the empty prefix. The residual algorithm explicitly treats that seed.

Signed-bound self-reduction also has a 500-case arithmetic check, including negative endpoints. Finite label jumps are checked by requiring equality after the first raw-history crossing rather than assuming every positive label exists.

F.4 LARGE INTEGERS AND PRACTICAL SCOPE

Structural large-integer checks use a 4,097-bit player parameter and $q \in \{1, 3, 7\}$ on selected families: 12 sources, 21 pieces, 57 sampled targets, 63 top inversions, and 63 trigger inversions. The instrumentation records zero expanded giant coordinate ranges, residue periods, or multiplicities.

The selected exact-excess-three SMT fixture in Section 5 is an end-to-end arithmetic path. It does not instantiate all possible support shapes or excess levels. Solver-returned assignments are checked against the compiled arithmetic constraints; correctness for the game also uses the proved compiler semantics. No literal recurrence is run over its astronomically large game grid.

The prepared-state bridge is capped at $q \leq 4$, requires an upper-state top provider, and does not implement the entire public upper query API. The mathematical algorithm in Theorem 4.1 must not be identified with that bridge. In particular, the classical fixed-dimensional ILP theorem is not a runtime guarantee for unrestricted SMT search.

The recorded approximately 30,000-branch abort is a practical limitation, not a counterexample to correctness. We omit timing-based performance claims because a controlled benchmark with common hardware, software versions, baselines, and timeout policy has not been completed. No statistical or learned-policy evaluation is included.