Absorbing States in Greedy Search: When Endogenous Constraints Break Sequential Extraction

Jack Chaudier Gaffney

Abstract

We study constrained extraction of weighted event subsequences under a sequential phase grammar with an endogenous turning point. The turning point is defined by an `argmax` over the selected subset, so feasibility constraints depend on the evolving solution itself. We prove a sufficient-condition impossibility result for a common greedy policy: if the number of development-eligible events before the greedy-forced turning point is smaller than the required prefix length, greedy cannot produce any valid sequence. The proof uses an absorbing-state argument in the product automaton of event graph and grammar DFA. Empirically, the result is exact in its domain: across 11,400 boundary instances, false positives are zero under the development-eligible prefix measure, with only known measurement artifacts under alternate counters. We then present a complete failure taxonomy and a constructive algorithm hierarchy. The hierarchy reveals a partial order with sharp boundaries, including a constraint antagonism regime where span-only viability filtering collapses to 0% validity on multi-burst+gap graphs. A TP-conditioned dynamic program recovers 94% (multi-burst+gap) and 96% (bursty+gap), with zero degradations versus BVAG. Across valid cases, approximation quality is consistently high, showing a feasibility-dominated landscape.

1 Introduction

Greedy selection is the default strategy for sequential extraction in many settings because it is simple, fast, and often effective. In hereditary systems with suitable exchange structure, greedy can enjoy strong guarantees. In many practical systems, however, feasibility constraints are *endogenous*: a distinguished selected element determines how all other elements are interpreted. This coupling can invalidate classical greedy assumptions.

We study extraction from temporally ordered event graphs under a phase grammar with ordered states {SETUP, DEVELOPMENT, TURNING_POINT, RESOLUTION}. The grammar requires at least k DEVELOPMENT events before the TURNING_POINT. The turning point itself is endogenous: it is the maximum-weight focal-actor event in the extracted sequence.

Our contributions are:

1. **Prefix-Constraint Impossibility Theorem.** A sufficient-condition failure criterion with zero false positives in the validated regime (11,400+ instances).

2. **Complete failure taxonomy.** Four classes with mechanism, prevalence, and constructive fix or structural boundary.

3. **Constructive hierarchy.** From myopic greedy to viability-aware variants to TP-conditioned dynamic programming, including a constraint antagonism result.

4. **Feasibility-dominated quality result.** When valid sequences exist, approximation ratios are consistently high (typically > 0.95).

5. **Invariance validation.** The theorem persists across actor count, event density, weight families, and topology sweeps.

The empirical program was motivated by a predecessor system (3,250+ deterministic runs), but the framework studied here is domain-agnostic and uses only event, actor, timestamp, and grammar primitives. Experiment numbering in this paper is the base range (Exps 1–39); companion streaming and compression papers continue the same index for traceability [Gaffney, 2026b,a].

2   Problem Formulation

Definition 1 (Event Graph). *An event graph is $G = (V, E, t, w, a)$ where V is a finite event set, $E \subseteq V \times V$ is a set of directed causal edges (time-respecting: $(u, v) \in E \Rightarrow t(u) < t(v)$), $t : V \to \mathbb{R}$ is a timestamp function inducing a temporal order (with deterministic tie-breaking by event ID), $w : V \to \mathbb{R}_{>0}$ is a weight map, and $a : V \to A$ assigns actors.*

Definition 2 (Candidate Pool). *Given focal actor a^*, let $e_0 \in V$ be an anchor event selected independently of e^* (baseline: e_0 is chosen from the top-n events by global weight). For that anchor, the candidate pool is*

$$P = \{v \in V : \textit{there exists a path from } e_0 \textit{ to } v \textit{ in } (V, E)\} \cup \{e^*\},$$

where $e^* = \arg\max_{v:a(v)=a^*} w(v)$ is injected regardless of reachability. Pool construction is a modeling choice: BFS pools enforce causal locality; unrestricted pools ($P = V$) maximize temporal coverage.

In this paper, E is used only to construct P; once P is fixed, feasibility and scoring depend on timestamps, weights, actor identity, and grammar constraints.

Definition 3 (Phase Grammar). *A phase grammar is a DFA $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ over*

$$\Sigma = \{\textit{SETUP}, \textit{DEVELOPMENT}, \textit{TURNING\_POINT}, \textit{RESOLUTION}\},$$

with monotone phase transitions and acceptance requiring at least k DEVELOPMENT labels before TURNING_POINT.

Definition 4 (Phase Classifier). *Given subsequence S and designated turning point $\tau \in S$, a phase classifier $\phi_\tau : S \to \Sigma$ assigns phases subject to: (i) monotone phase progression in temporal order, (ii) $\phi_\tau(\tau) = \textit{TURNING\_POINT}$, (iii) post-$\tau$ events receive RESOLUTION. The assignment of pre-τ events to SETUP or DEVELOPMENT is a degree of freedom constrained by the grammar's acceptance condition ($\geq k$ DEVELOPMENT labels). A* baseline *classifier uses a fixed position-based SETUP allocation rule; a* grammar-aware *classifier reserves k DEVELOPMENT slots before assigning SETUP.*

Definition 5 (Endogenous Turning Point). *For focal actor a^*, the turning point is*

$$\tau(S) = \arg\max_{v \in S, \, a(v)=a^*} w(v).$$

Thus, TP identity depends on the selected set.

Definition 6 (Extraction Problem). *Given G, focal actor a^*, grammar parameter k, length limit L, minimum timespan fraction s, and maximum temporal gap $\Delta_{\max}$, find*

$$S^* = \arg\max_{S \subseteq V} \sum_{v \in S} w(v)$$

subject to: (i) $|S| \leq L$, (ii) S accepted by $\mathcal{A}$ under $\phi_{\tau(S)}$, and (iii) auxiliary temporal constraints.

Definition 7 (Greedy Policy). *For each anchor-defined candidate pool, pre-select $e^* = \arg\max_{v:a(v)=a^*} w(v)$ as TURNING_POINT (forced TP), inject e^* into the pool if needed, and construct the remaining sequence by forward weight-maximizing selection.*

This differs from standard regular-language constrained optimization because the automaton pivot is endogenous rather than fixed.

Synthetic generators. We evaluate on two graph topologies. The *bursty* generator places high-weight events preferentially in the first ϵ fraction of the timeline (front-loading parameter $\epsilon \in [0, 1]$); higher ϵ increases the probability that the max-weight focal event occurs early, activating the absorbing state. The *multi-burst* generator creates two or more temporal clusters of high-weight events separated by sparse valleys containing only low-weight bridge events. When the grammar includes a maximum temporal gap constraint $\Delta_{\max}$, sequences must include bridge events from these valleys, creating the constraint antagonism studied in §5.2.

Proposition 1 (Non-hereditary structure). *Under the extraction problem with endogenous TP, the feasible family $\mathcal{F} = \{S \subseteq V : S \text{ accepted by } \mathcal{A} \text{ under } \phi_{\tau(S)}\}$ violates both the hereditary axiom and the exchange axiom of greedoids.*

Proof sketch. For hereditary violation: let $S \in \mathcal{F}$ and let $v = \tau(S)$ be the TP. Removing v changes $\tau(S \setminus \{v\})$, reassigning all phase labels and potentially making $S \setminus \{v\} \notin \mathcal{F}$. For exchange violation: let $S_1, S_2 \in \mathcal{F}$ with $|S_1| < |S_2|$. Adding any $v \in S_2 \setminus S_1$ to S_1 can change $\tau(S_1 \cup \{v\})$, invalidating the phase structure. Empirically, both axioms fail on all 40 tested instances (n=20, N_{actors}=3). $\square$

This places the system outside the Korte–Lovász greedoid framework [Korte and Lovasz, 1981, Bjorner and Ziegler, 1992], in the regime of general independence systems where greedy has no approximation guarantees.

3 Prefix-Constraint Impossibility Theorem

Definition 8 (Development-Eligible Event). *Fix a turning point τ and a phase-classification policy (Definition 4). Event $v \in P$ is* development-eligible *if $t(v) < t(\tau)$ and the policy labels v as DEVELOPMENT. If the policy imposes no mandatory minimum SETUP count, every pre-τ event is development-eligible; under position-based policies with a forced SETUP prefix, only pre-τ events outside that prefix are eligible.*

This makes explicit that j_{dev} depends on classifier policy: Definition 4 determines which pre-TP events remain available for DEVELOPMENT labeling.

Theorem 1 (Prefix-Constraint Impossibility). *Let greedy force $e^* = \arg\max_{v:a(v)=a^*} w(v)$ as TURNING_POINT. Let grammar require $k \geq 1$ DEVELOPMENT events before TURNING_POINT. Let j_{dev} be the number of development-eligible events (any actor) with timestamp strictly before e^* in the candidate pool under the chosen phase-classification policy. If $j_{dev} < k$, greedy produces zero valid sequences.*

Proof. Use product automaton $\mathcal{P} = G \times \mathcal{A}$. (1) By Definition 7, greedy pre-selects e^* as TURNING_POINT in each candidate evaluation. (2) Once e^* is consumed as TURNING_POINT, DFA monotonicity forbids transitions back to DEVELOPMENT. (3) Any greedy pre-TP prefix can include at most j_{dev} development-eligible events. (4) If $j_{dev} < k$, the required prefix count is unreachable. Hence no greedy completion can satisfy acceptance. $\square$

Remark 1. *The converse is false: $j_{dev} \geq k$ does not imply success.*

Remark 2. *Using focal-only counters yields false positives. Counting development-eligible prefix events across* all *actors is required for zero-FP diagnosis.*

Bridge to context-algebra notation ($j_{dev} \leftrightarrow d_{\text{pre}}$). This paper uses count-based development capacity, where each development-eligible pre-TP event contributes one unit. In that unit-mass regime, the finite counter equals pre-pivot capacity: $j_{dev} = d_{\text{pre}}$. More generally, if events carry non-unit development mass $m(v)$, then

$$d_{\text{pre}} = \sum_{v:\, t(v) < t(e^*),\ \text{DEV-eligible}} m(v),$$

and the impossibility boundary becomes $d_{\text{pre}} < k$. This is the same absorbing condition used in Paper 03's context algebra.
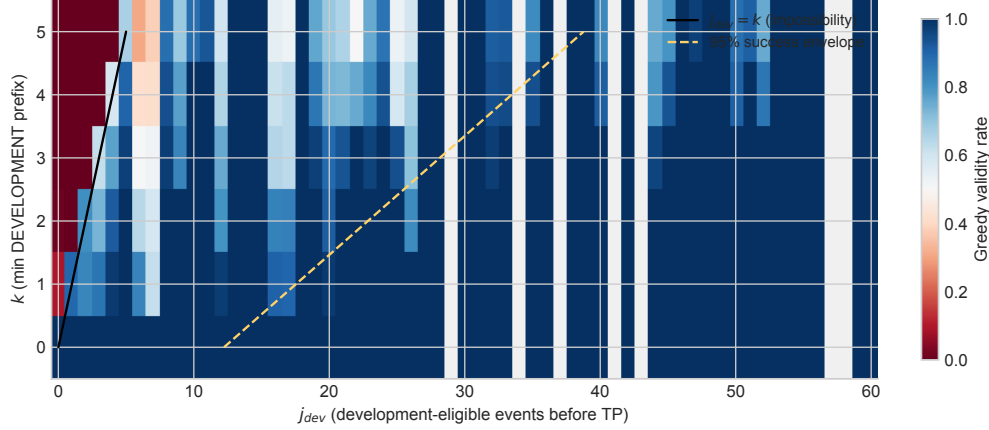
Figure 1: Greedy validity rate as a function of prefix requirement k and development-eligible count j_{dev} (pool-based: development-eligible events in the BFS+injection candidate set). Solid line: impossibility-theorem boundary ($j_{\text{dev}} = k$); validity is exactly zero below. Dashed line: empirical 95% success envelope ($j_{\text{dev}} \approx 5.31 \cdot k + 12.23$, fit by logistic regression on bursty generator with BFS+injection pool). The stochastic zone between the two boundaries contains the non-absorbing failure modes (Classes B–D).

Empirical validation. From the final boundary sweep (11,400 instances, $k \in \{0, \ldots, 5\}$, $\epsilon \in \{0.05, \ldots, 0.95\}$, 100 seeds per ϵ): (1) zero false positives for the output-based development counter; (2) for pool-based j_{dev}, false positives are zero for $k \geq 2$ and eight at $k = 1$, all diagnosed as timestamp-tie artifacts on one seed; (3) below-diagonal validity is exactly 0.000 for all cells with $k \geq 3$. The boundary is sharp: Figure 1 shows the transition from zero validity below the diagonal to high validity above it. Figure 2 shows how validity, absorption rate, and theorem accuracy co-vary with front-loading ϵ.

Exogenous TP validation. To isolate the role of endogenous TP selection, we repeated the boundary sweep with TP fixed exogenously at the median-timestamp focal event rather than the max-weight focal event. The diagonal barrier $j_{\text{dev}} < k \Rightarrow$ zero validity persists identically—it is a structural property of the grammar, not of TP selection. However, the probability of landing below the barrier drops sharply: under endogenous TP with $\epsilon = 0.80$, $P(j_{\text{dev}} < k) = 13.6\%$; under exogenous TP, $P(j_{\text{dev}} < k) = 0.0\%$. This confirms that the absorbing state is activated by the interaction between weight-based TP selection and temporal front-loading, not by the grammar alone.

Additionally, when 11 non-absorbing baseline failure cases at ϵ=0.80, k=1 (later classified as Classes C/D within the 138 baseline failures) are re-run with TP forced to the oracle's TP selection, 9/11 produce valid sequences. The 2 residual cases fail with `insufficient_timespan`, a constraint orthogonal to TP selection. In this prefix-only regime, most failures localize to TP selection rather than sequence assembly.

4 Failure Taxonomy

Across the baseline evaluation (ϵ=0.80, k=1, 150 seeds), 138 instances exhibit greedy failure where the exact oracle succeeds—these are the false negatives that the taxonomy decomposes. The four failure classes form a progression from structural impossibility through pipeline artifacts to algorithmic ceilings, each with a precise constructive resolution or proved boundary (Table 1).

Class A (Absorbing State) captures the theorem's domain: when $j_{dev} < k$, no greedy sequence can satisfy the prefix constraint. This is not an algorithmic weakness but a structural impossibility given the endogenous TP. The only resolution is exogenous TP reassignment, which the hierarchy addresses at Level 3.

Class B (Pipeline Coupling) is a classifier artifact: the default position-based phase rule (implementation: first $\lceil 0.2 \cdot n_{\text{pre}} \rceil$ pre-TP events labeled SETUP) allocates too many pre-TP events to SETUP,
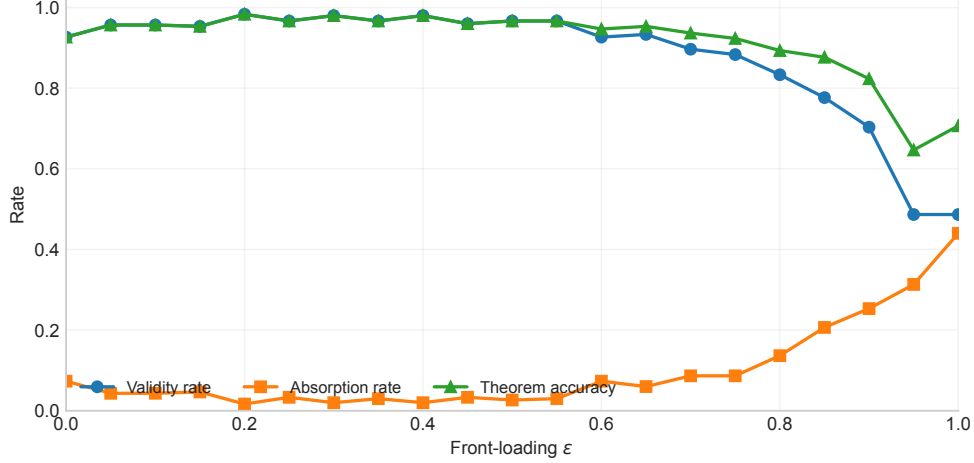
Figure 2: Position sweep: validity, absorption, and theorem accuracy versus front-loading ϵ (100 seeds per ϵ). As ϵ increases, more high-weight events concentrate early, raising absorption probability. Theorem accuracy uses the pool-based development-eligible counter (j_{dev}).

Class	Mechanism	Observed prevalence	Fix	Recovery
A	Absorbing state ($j_{dev} < k$); TP enters post-TP DFA region before enough DEVELOPMENT capacity exists.	Theorem domain (conditioned on k, ϵ).	Not repairable within endogenous greedy; requires TP reassignment.	Structural impossibility.
B	Pipeline coupling in classifier (SETUP allocation consumes scarce pre-TP slots).	63/138 initial failures (45.7%).	Grammar-aware classifier preserving min DEVELOPMENT slots.	63/63.
C	Commitment timing: bridge/span infrastructure events exist but are selected too late; plus pool-boundary subcases.	Initial TP misselection: 18/138 (13.0%); gap cohorts: 20 (multi-burst) and 26 (bursty) BVAG-fail/exact-valid.	TP-conditioned solver (plus explicit pool-boundary diagnosis).	17/20 and 21/26.
D	Assembly compression: weight-maximizing assembly collapses timespan below threshold.	57/138 initial failures (41.3%).	Span-VAG viability filter.	55/57.

Table 1: Failure taxonomy with mechanism and constructive boundary.

leaving zero DEVELOPMENT slots even when development-eligible events exist. A grammar-aware classifier that reserves k DEVELOPMENT slots before assigning SETUP recovers all 63 affected cases. This class is fully eliminated by a one-line fix.

Classes C and D represent genuine algorithmic ceilings where the failure persists despite $j_{dev} \geq k$ and a correct classifier. Class D (assembly compression) is solved by span-aware viability filtering: the greedy assembler compresses timespan below the minimum threshold, and a one-step lookahead prevents this (55/57 recoveries). Class C (commitment timing) is the true frontier—greedy commits to high-weight events before recognizing the need for bridge or span-extending infrastructure. The TP-conditioned solver addresses this by factoring the endogenous TP out of the inner optimization.
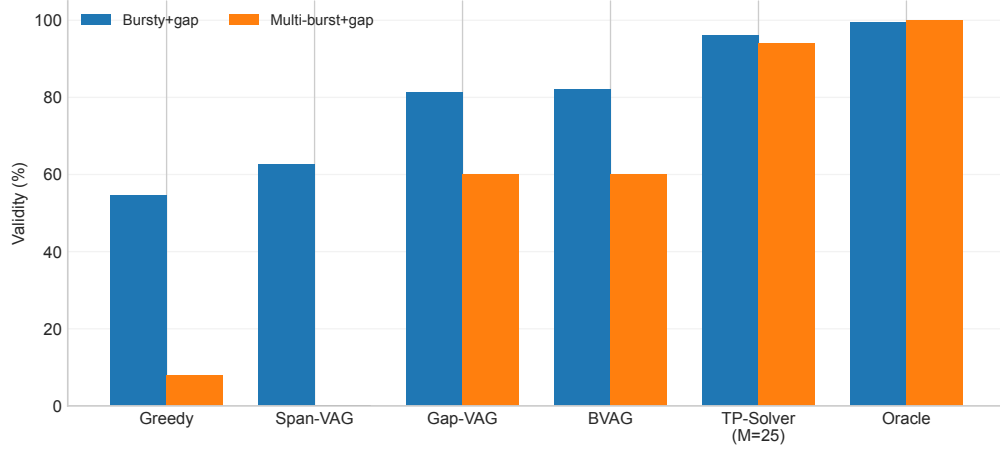
Figure 3: Validity by algorithm on bursty+gap and multi-burst+gap. Note the span-VAG adversarial collapse: multi-burst+gap validity drops to 0%, below myopic greedy (8%).
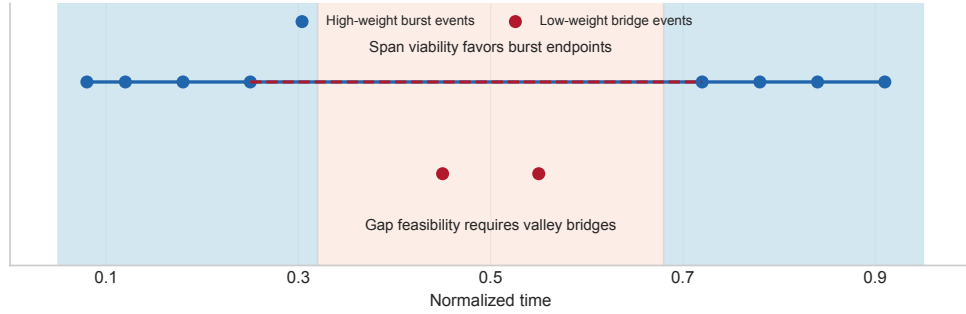

Figure 4: Constraint antagonism on multi-burst topology. Span viability favors burst endpoints (blue), skipping valley bridge events (red) required for gap feasibility.

5 Constructive Algorithm Hierarchy

5.1 Viability-Aware Greedy (VAG)

Span-VAG enforces one-step timespan viability and recovers 55/57 Class-D cases (Figure 3). Gap-aware VAG enforces bridge existence per gap and reaches 81.3% (bursty+gap) and 60% (multi-burst+gap). BVAG adds a global bridge-budget lower bound and improves bursty from 81.3% to 82.0% (+3 cases), with no multi-burst gain.

5.2 Constraint Antagonism

Observation 1 (Constraint Antagonism). *Span-VAG achieves 0% validity on multi-burst+gap, below myopic greedy (8%).*

Mechanism: span viability prefers temporally distant endpoints, which skips valley bridge events needed for local gap feasibility (Figure 4). This demonstrates that the hierarchy is a **partial order**: span-VAG and gap-VAG branch rather than forming a monotone chain.

5.3 TP-Conditioned Dynamic Programming

The TP-conditioned solver factors endogenous TP coupling into an outer loop over top-M TP candidates and solves each fixed-TP case by label-setting DP with dominance on (score, slots, first_time). Fixing TP makes phase labels deterministic in the inner problem, yielding an RCSPP-style formulation [Irnich and Desaulniers, 2005].

Level	Algorithm	Search type	TP selection	MB+gap	Bursty+gap	Complexity	Median runtime
0	Myopic greedy	Forward, myopic	Endo. (max-w)	8%	54.7%	$O(n^2)$	1.3 ms
1	Span-VAG	Forward, 1-step span	Endogenous	0%	62.7%	$O(n^2)$	9 ms
2	Gap-aware VAG	Forward, depth-1 bridge	Endogenous	60%	81.3%	$O(n^2 \cdot \text{pool})$	10 ms
2.5	BVAG	Forward, budget bound	Endogenous	60%	82.0%	$O(n^2 \cdot q)$	10 ms
3	TP-solver ($M{=}25$)	Multi-pass DP	Outer loop (top-M)	94%	96.0%	$O(ML_{\max}n^2 d)$	2.3 s / 5.7 s
∞	Exact oracle	Exhaustive DP	All focal events	100%	99.3%	$O(n^2 S)$	4.4 s / 3.1 s

Table 2: Algorithm hierarchy from myopic greedy to exhaustive oracle. MB = multi-burst. Runtime shown as multi-burst / bursty. S = exhaustive search space size. TP-solver is faster than oracle on multi-burst but slower on bursty at $M{=}25$; adaptive M with early exit (not shown) restores competitive runtime.

At $M = 25$, recovery is 94% (multi-burst+gap) and 96.0% (bursty+gap), with no cases where BVAG outperforms the TP-solver. Residuals are diagnosed as either TP-candidate truncation (3 cases) or pool-construction boundary (5 bursty cases), not DP incompleteness. The exact oracle searches over all events in V (unrestricted pool); all other algorithms are restricted to P. The oracle's 99.3% value on bursty+gap reflects a small set of structurally infeasible instances where no grammar-valid extraction exists even under exhaustive search.

6 Approximation Quality

Observation 2 (Bimodal quality). *When valid sequences are produced, approximation ratios are near one:*

- *Bursty (no gap): mean 0.996, minimum 0.975.*

- *Multi-burst (no gap): mean 0.9905, minimum 0.9512.*

- *Zero cases below 0.90 in either no-gap topology.*

Interpretation: the landscape is feasibility-dominated. The dominant failure mode is inability to enter the feasible region; optimization quality inside that region is high.

7 Invariance Results

The theorem is stable across controlled sweeps:

- **Actor count** ($N = 2$ to 50): 0 FPs; mean j_{dev} constant at 48.264.
- **Event density** ($n = 50$ to 1000): 0 FPs; mean j_{dev} scales linearly (10.640, 21.392, 48.264, 101.280, 201.048).
- **Weight distributions** (uniform, power-law, bimodal): 5 FPs total (0.67%), all diagnosed as TP-identity shift artifacts under weight replacement.
- **Multi-burst topology**: 0 FPs; theorem accuracy 0.948.

These results support structural robustness of the theorem under topology and parameter variation.

8 Related Work

Greedy guarantees in combinatorial optimization rely on structural assumptions such as hereditary/exchange properties [Korte and Lovasz, 1981, Bjorner and Ziegler, 1992]. Our feasible family violates those assumptions because TP identity is endogenous (Proposition 1).

Process mining aligns event traces to process models, typically with fixed model semantics [van der Aalst, 2016]. Grammar-constrained decoding in language models uses automata or grammars during generation [Scholak et al., 2021, Willard and Louf, 2023]. Our setting differs because grammar semantics depend on an endogenous pivot selected by the solution itself.

With TP fixed, the inner problem maps to resource-constrained shortest path formulations [Irnich and Desaulniers, 2005]. The $\arg\max$ coupling itself has been studied as a constraint-programming primitive [Gange and Stuckey, 2020]; our contribution is its composition with REGULAR-like sequential grammar constraints under greedy commitment.

9   Discussion and Future Work

TP-decoupling as Markovization. The central mechanism underlying the entire hierarchy is endogenous TP coupling: because $\tau(S) = \arg\max_{v \in S,\, a(v)=a^*} w(v)$, modifying any element of S can change the TP identity and trigger global phase reassignment. This coupling makes the search state *non-local*: the feasibility of a partial sequence depends on its eventual completion through the TP identity. Factoring TP selection into an explicit outer loop *Markovizes* the inner problem: with τ and n_{pre} fixed, phase labels become deterministic functions of timestamp relative to τ, and the inner problem reduces to a resource-constrained shortest path where label-setting DP applies with standard dominance. Note that fixing TP does not restore hereditary structure—the minimum-count grammar constraint ($\geq k$ development events) remains non-hereditary even under exogenous TP. What TP-conditioning removes is the *global coupling* whereby any element change can cascade into full phase reclassification.

Failure decomposition. The observed failure rate under any TP-selection policy π and candidate-set function P decomposes as:

$$P(\text{fail}) \;=\; P(j_{\text{dev}}(\tau_\pi) < k) \;+\; \big(1 - P(j_{\text{dev}}(\tau_\pi) < k)\big) \cdot P(\text{other failure} \mid j_{\text{dev}}(\tau_\pi) \geq k).$$

The first term is the absorbing-state probability, controlled by the TP policy's interaction with the weight-timestamp distribution. The second term is the residual failure rate among non-absorbed instances, weighted by the probability of escaping absorption. Endogenous TP selection under front-loading inflates the first term; TP-conditioning reduces it by searching over multiple τ candidates. This decomposition clarifies what each level of the hierarchy addresses: VAG variants reduce the conditional residual rate, while TP-conditioning attacks the absorbing-state probability.

Forward filtering ceiling. Depth-1 viability filters (span-VAG, gap-VAG, BVAG) improve specific failure classes but cannot close commitment-timing failures globally. The BVAG null result (0/20 multi-burst recoveries beyond gap-VAG) empirically demonstrates this ceiling. The fundamental limitation is that forward filters make irrevocable selections: once a high-weight event is chosen, the remaining budget and constraint state may preclude feasible completion, and no local lookahead can detect this without exploring the full constraint interaction.

Constraint antagonism. The 0% span-VAG result on multi-burst+gap is a structural phenomenon, not a corner case. When two constraints interact through incompatible local objectives, optimizing for one actively worsens the other under greedy composition. Span viability pushes selections toward temporally distant endpoints to satisfy timespan, but these selections skip valley bridge events required for gap feasibility. This antagonism is topology-dependent and manifests on multi-burst graphs with sparse valleys. The general lesson is that composing constraint-aware filters requires analyzing their interaction structure, not just their individual correctness.

Proposition 2 (Pool Temporal Coverage Barrier). *Let $P \subseteq V$ be a candidate pool. Define its temporal coverage $\text{cov}(P) = \max_{v \in P} t(v) - \min_{v \in P} t(v)$ and the required span $s \cdot T$ where $T = \max_{v \in V} t(v) - \min_{v \in V} t(v)$. If $\text{cov}(P) < s \cdot T$, then no sequence $S \subseteq P$ can satisfy the minimum timespan constraint, regardless of search strategy.*

Proof. For any $S \subseteq P$, $\text{span}(S) \leq \text{cov}(P) < s \cdot T$. $\square$

Pool construction boundary. The TP-solver's residual failures are pool-boundary cases: the exact oracle's optimal sequence includes span-extending events at temporal extremes (very early or very late timestamps) outside the BFS-reachable candidate set. These events typically belong to non-focal actors with no causal path from the focal actor. The solver's best attempts within the BFS pool fail with `insufficient_timespan`—not gap violations. Notably, the Pool Temporal Coverage Barrier (Proposition 2) does not fully explain these residuals: the pool's aggregate temporal extent exceeds the required span (1.6–3.2$\times$ the minimum). However, the solver's weight-optimized sequences cluster in the high-weight temporal core and do not reach the pool's temporal extremes. The oracle

succeeds by selecting specific low-weight endpoint events outside the BFS pool. This is a modeling choice—BFS-based pools enforce causal locality (appropriate for narrative or incident domains), while unrestricted pools include span-extending events at the cost of causal interpretability. The choice should be explicit in any deployment.

Applications. The endogenous-pivot failure mode is plausible in any sequential extraction system where a distinguished element determines how other elements are interpreted: incident diagnosis systems anchoring on a root-cause event, constrained generation workflows committing to a high-salience token, and process mining alignments fixing a reference activity. The constructive hierarchy suggests a general recipe: when greedy fails due to endogenous coupling, factor the pivot into an outer enumeration and solve the residual as a constrained shortest path.

10   Conclusion

We isolate a precise failure mechanism for greedy extraction under endogenous sequential constraints, prove a sufficient-condition impossibility theorem, validate it with zero false positives in its measured domain, and map the constructive hierarchy needed to recover feasibility. The main empirical pattern is sharp: failures are structural and deterministic; successes are near-optimal.

References

Anders Bjorner and Gunter M. Ziegler. Introduction to greedoids. In Neil White, editor, *Matroid Applications*, volume 40 of *Encyclopedia of Mathematics and its Applications*, pages 284–357. Cambridge University Press, 1992. doi: 10.1017/CBO9780511662041.011.

Jack Chaudier Gaffney. The validity mirage: Context algebra for endogenous semantics under memory compression, 2026a. arXiv preprint, submitted concurrently.

Jack Chaudier Gaffney. Streaming oscillation traps in endogenous-pivot sequential extraction, 2026b. arXiv preprint, submitted concurrently.

Graeme Gange and Peter J. Stuckey. The argmax constraint. In *Principles and Practice of Constraint Programming (CP 2020)*, volume 12333 of *Lecture Notes in Computer Science*, pages 323–337. Springer, 2020. doi: 10.1007/978-3-030-58475-7_19.

Stefan Irnich and Guy Desaulniers. Shortest path problems with resource constraints. In Guy Desaulniers, Jacques Desrosiers, and Marius M. Solomon, editors, *Column Generation*, pages 33–65. Springer, 2005. doi: 10.1007/0-387-25486-2_2.

Bernhard Korte and Laszlo Lovasz. Mathematical structures underlying greedy algorithms. In *Fundamentals of Computation Theory (FCT'81)*, volume 117 of *Lecture Notes in Computer Science*, pages 205–209. Springer, 1981. doi: 10.1007/3-540-10854-8_22.

Torsten Scholak, Nathan Schucher, and Dzmitry Bahdanau. Picard: Parsing incrementally for constrained auto-regressive decoding from language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9895–9901. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EMNLP-MAIN.779.

Wil van der Aalst. *Process Mining: Data Science in Action*. Springer, 2 edition, 2016. doi: 10.1007/978-3-662-49851-4.

Brandon T. Willard and Rémi Louf. Efficient guided generation for large language models. *arXiv preprint arXiv:2307.09702*, 2023. URL https://arxiv.org/abs/2307.09702.

#	Experiment	Key finding
1–17	Failure taxonomy phase	Position sweep, oracle divergence, classifier coupling (63), TP misselection (18), assembly compression (57), beam saturation, repair null.
18–27	Constructive phase	Exact oracle baseline, non-hereditary verification, VAG variants, gap adversarial tests, BVAG marginal gain (+3 bursty, +0 multi-burst).
28–29	Boundary mapping	k–j formalization and measurement companion (counter-definition mismatch resolved).
30–36	Final validation	FP diagnosis, final k–j boundary, invariance suite, weight-FP diagnosis, TP-solver evaluations ($M=10, 25$), pool-boundary diagnosis.
37	Exogenous TP validation	Diagonal barrier persists under exogenous TP; $P(j_{dev} < k)$ drops from 13.6% to 0.0%; 9/11 Class-C/D cases recovered with oracle TP.
38	Empirical success envelope	Logistic regression on boundary data; 95% contour at $j_{dev} \approx 5.31k + 12.23$; AUC = 0.92.
39	Pool barrier verification	Pool Temporal Coverage Barrier (Prop. 2) does not explain residuals (pool span 1.6–3.2× minimum); solver fails with `insufficient_timespan` due to weight-optimized sequences not reaching temporal extremes.

Table 3: Condensed list of all 39 experiments from the canonical context.

Hypothesis	Result
Per-actor weight decomposition determines pool contamination	Rejected ($r = 0.033$, 0/9 fixed).
Constraint relaxation is monotone-beneficial	Rejected (domain-dependent; monotone on synthetic, non-monotone in predecessor system).
Beam width resolves assembly compression	Rejected (saturates at width 2; 0/57 Class-D recoveries).
Local repair resolves timespan compression	Rejected (0/57).
BVAG closes multi-burst gap frontier	Rejected (0/20 additional recoveries).
Span-VAG is monotone improvement over greedy	Rejected (0% on multi-burst+gap).

Table 4: Negative results that constrain the theory and algorithm design space.

A   Full Experiment List (39)

B   Disproven Hypotheses

C   TP-Conditioned Solver

Algorithm 1 gives the full pseudocode for the TP-conditioned label-setting solver. The key insight is that fixing the TP candidate τ and pre-TP event count n_{pre} makes phase labels deterministic, reducing the problem to a resource-constrained shortest path. In the pseudocode, $\ell \oplus v$ denotes extending label ℓ by appending event v: incrementing the score by $w(v)$, incrementing the slot count, updating the earliest timestamp, and advancing the DFA state.

Dominance. Label A dominates B in the same bucket if $A.score \geq B.score$, $A.slots \leq B.slots$, and $A.t_1 \leq B.t_1$. This three-dimensional criterion ensures that extending A yields a sequence at least as good as extending B along every constrained dimension. Because DFA state is part of the bucket key, slot count acts as remaining-capacity resource within a fixed automaton state, so the $\leq$ dominance direction is valid.

Algorithm 1 TP-Conditioned Label-Setting Solver

Require: Event graph $G=(V, t, w, a)$, focal actor a^*, grammar $\mathcal{A}$ with prefix k, length limit $L_{\max}$, candidate count M
Ensure: Best valid sequence S^* or $\emptyset$

1: $\mathcal{T} \leftarrow$ top-M focal-actor events by weight
2: $S^* \leftarrow \emptyset$; $best \leftarrow -\infty$
3: **for** $\tau \in \mathcal{T}$ **do**
4:     **for** $n_{\mathrm{pre}} = 0$ **to** $L_{\max}-1$ **do**
5:         Assign deterministic phase labels given (τ, n_{pre})
6:         $\mathcal{L} \leftarrow \{(score=0,\ slots=0,\ t_1=\infty,\ last=\bot,\ dfa=q_0)\}$
7:         **for** event v in temporal order within candidate pool **do**
8:             **for** label $\ell \in \mathcal{L}$ with valid DFA transition $\delta(\ell.dfa, \phi(v))$ **do**
9:                 $\ell' \leftarrow \ell \oplus v$: $\ell'.score \leftarrow \ell.score + w(v)$; update slots, t_1, dfa
10:                 **if** ℓ' not dominated in bucket $(last=v,\ dfa=\ell'.dfa)$ **then**
11:                     Add ℓ'; remove labels it dominates
12:                 **end if**
13:             **end for**
14:         **end for**
15:         **for** accepting $\ell \in \mathcal{L}$ satisfying all constraints **do**
16:             **if** $\ell.score > best$ **then**
17:                 $S^* \leftarrow$ Reconstruct(ℓ); $best \leftarrow \ell.score$
18:             **end if**
19:         **end for**
20:     **end for**
21: **end for**
22: **return** S^*

Complexity. The outer loops contribute $O(M \cdot L_{\max})$. The inner label-setting DP processes $O(n)$ events, each extending $O(|\mathcal{L}|)$ active labels with $O(d)$ DFA states, giving $O(M \cdot L_{\max} \cdot n^2 \cdot d)$ worst case. Empirically at $M=25$: median 536K labels (multi-burst), 1.54M labels (bursty); median runtime 2.3 s and 5.7 s respectively.