

Persistent Scaffold Computation: Causal Presentations, Sewing, and Feedback

Joshua Gay
FPRD Lab, Insight Forge

21 August 2026

Abstract

Scaffolding automata, introduced by Loff, Moreira, and Reis to characterize parsing expression grammars, build one bounded-degree persistent graph node per input symbol. Their language-theoretic power is known, but the internal mathematics of the histories they generate is largely implicit. We develop a causal presentation calculus for those histories.

For fixed degree and observation distance, construction histories are words over a finite row alphabet. Rooted port-word unfolding separates the physical history from its behavioral figure. We prove that generated figures are reachability chains with regressive, hence $\mathcal{R}$ -trivial, navigation. On finite weakly acyclic figures, guarded pair folds generate the full bisimulation quotient. Lifting that static fact through time gives the main general theorem: least endpoint detours and one literal **SELF**-unroll rule form a terminating confluent graph-local presentation of complete strong-trace equivalence, with a linear row-move bound.

Final observation is a strictly coarser problem because it can erase earlier figures. We isolate its replay support, prove that each causal-garbage fibre has a cubical path presentation, give a two-row sewing theorem on the unexposed acyclic fragment, and prove a whole-space final normal form at degree one and distance one. These calibrated results are not promoted to an unrestricted final-equivalence theorem.

Finally, we use the calculus to revisit a demanding application. A direct even-palindrome construction needs a fresh occurrence record to own a fresh receipt rope whose terminal receipt points back to the owner. This exposes a same-step cycle missed by an earlier topological allocation model. We prove that bounded simultaneous record batches, including arbitrary finite fresh cycles, compile through physical **SELF** plus finite slot tags; an opaque-element staging lemma explains why a strict persistent sequence can tie the receipt knot only at that boundary. Palindromes are therefore a stress test of the algebra, not its organizing theorem. The paper is released for specialist review and makes no novelty claim for established coalgebraic, automata-minimization, or term-graph ingredients.

Contents

1	From a finite presentation to a history algebra	3
1.1	What is new here, and what is inherited	3
2	Histories, unfoldings, and observations	4
2.1	The finite row alphabet	4

2.2	Port-word behavior	4
2.3	Full abstraction	5
2.4	The canonical behavioral figure	5
3	Chronological order and navigation	6
4	Static guarded folding	6
5	The causal presentation calculus	7
5.1	Strong equivalence and semantic sewing	7
5.2	One-state extensions	8
5.3	The two primitive strong moves	8
5.4	SELF-eager prefixes	8
6	Final observation and replay support	10
6.1	A noncommuting boundary	10
7	Two calibrated sewing theorems	11
7.1	Unexposed acyclic two-row factorizations	11
7.2	The unary whole space	12
7.3	A path-level quotient	13
8	Feedback, receipts, and the PC-83 defect	14
8.1	Typed receipts	14
8.2	Bounded simultaneous record transducers	14
8.3	Staging a strict persistent sequence	15
9	What the FPRD approach contributed	17
10	Open boundaries	17
11	Assurance and nonclaims	18
12	Conclusion	18
A	Move and preservation table	19
B	The central implication map	19

1 From a finite presentation to a history algebra

Loff, Moreira, and Reis (LMR) introduced scaffolding automata as a machine model for ordinary parsing expression grammars (PEGs) [7]. At each input position the machine reads a fixed-radius neighborhood of the current top of a persistent graph, chooses a finite label, and appends one new node. Each new port is missing, points to the fresh node itself, or follows a bounded port word from the preceding top. LMR prove that a language has a PEG if and only if its reversal is decided by such a machine.

The usual observation is acceptance. This paper keeps more of the generated object visible. A run has a physical graph at every time, a behavioral figure at every time, and a final figure. Those are different quotients.

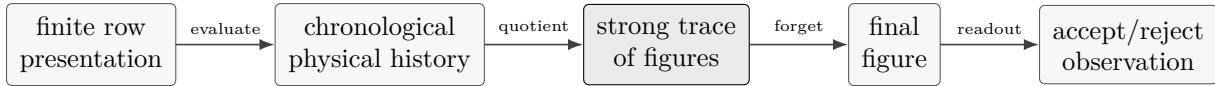


Figure 1: The observation ladder. Every arrow can identify presentations. No reverse implication is automatic.

This is the FPRD viewpoint used throughout:

$$\text{finite presentation} \longrightarrow \text{generated dynamics} \longrightarrow \text{chosen observation}.$$

Changing an observation may merge histories without changing the rows that generated them. Changing a presentation may alter the physical history while preserving every behavioral figure. The algebra begins by keeping both phenomena explicit.

1.1 What is new here, and what is inherited

Bisimulation quotients and functional bisimulation are standard coalgebraic objects [8]. Bottom-up merging of equivalent states and incremental minimization of acyclic deterministic automata are established [2]. Cyclic term graphs, tree unwinding, copying, hidden nodes, and confluence have a substantial theory [1]; static term-graph sharing also has a categorical presentation [3]. We do not rename those subjects as new scaffold theory.

The target here is the intersection of four restrictions:

1. one immutable layer is appended per input symbol;
2. every old address is read from the preceding top within one fixed radius;
3. a strong move must preserve the figure after every prefix; and
4. soundness must survive every common future continuation.

The resulting local rules live in the generated scaffold graph. They need not be bounded contiguous substring rules in the linear word of rows.

2 Histories, unfoldings, and observations

2.1 The finite row alphabet

Fix a degree $d \geq 1$, distance $k \geq 0$, and finite working alphabet Γ . Write $P = \{0, \dots, d-1\}$ for the port alphabet and

$$\Delta_{d,k} = \{\text{MISSING}, \text{SELF}\} \cup P^{\leq k}.$$

A *row* is an element of

$$\text{Row}_{d,k,\Gamma} = \Gamma \times \Delta_{d,k}^d.$$

The empty word $\epsilon \in P^{\leq k}$ is a valid descriptor: it names the preceding top.

Definition 2.1 (Evaluation). A history

$$H = (a_t; \delta_{t,0}, \dots, \delta_{t,d-1})_{t=1}^n$$

is evaluated from an unlabelled edgeless base node 0. At time t , append node t with label a_t . Its port i is missing if $\delta_{t,i} = \text{MISSING}$, points to t if $\delta_{t,i} = \text{SELF}$, and otherwise points to the endpoint obtained by following $\delta_{t,i}$ from node $t-1$ in the old scaffold. An invalid path gives a missing edge.

The set of histories is the free monoid $\text{Row}_{d,k,\Gamma}^*$ as syntax. Appending a row is a total right action on evaluated histories because invalid paths have the defined missing outcome.

2.2 Port-word behavior

For a rooted evaluated scaffold (S, t) , adjoin a missing value $\perp_e$ and define the partial right action of P^* by following ports. Its *behavioral unfolding* is

$$\mathcal{U}_{S,t} : P^* \longrightarrow \Gamma \cup \{\text{UNLABELLED}, \perp_e\},$$

where a valid word returns the endpoint label and an invalid word returns $\perp_e$. The unlabelled base is therefore observable and is not a missing edge.

Two rooted scaffolds are behaviorally equivalent, written $(S, t) \simeq (T, u)$, when their unfoldings agree. For a history H , let

$$\text{Fig}_j(H) = M(\mathcal{U}_{S_j,j})$$

denote the canonical minimal figure of its length- j prefix, defined below. The *strong trace* and *final observation* are

$$\text{Tr}(H) = (\text{Fig}_1(H), \dots, \text{Fig}_n(H)), \quad \pi(H) = \text{Fig}_n(H).$$

Definition 2.2 (Two congruences). Equal-length histories are *strongly equivalent*, $H \equiv_s H'$, when $\text{Tr}(H) = \text{Tr}(H')$. They are *finally equivalent*, $H \equiv_f H'$, when $\pi(H) = \pi(H')$.

Strong equivalence implies final equivalence. The converse fails because a later row may make an earlier difference unreachable or behaviorally irrelevant.

2.3 Full abstraction

Theorem 2.3 (Future-context full abstraction). *For rooted degree- d scaffolds over one working alphabet:*

1. *equal unfoldings, equality of every finite rooted neighborhood, and deterministic labelled bisimilarity are equivalent;*
2. *unfolding equivalence is preserved by every common scaffold row and hence by every common continuation; and*
3. *unequal unfoldings are distinguished after finitely many steps by a distance-two unary scaffold continuation.*

Proof. The depth- m neighborhood is exactly the restriction of the unfolding to port words of length at most m , which proves the first equivalence.

Suppose the old unfoldings agree and append the same row. The observed depth- k neighborhoods agree. For a query word starting at fresh port i , a missing descriptor fails on both sides; a SELF descriptor returns to the respective fresh root; and an old descriptor p reduces the comparison to $\mathcal{U}_{S,t}(pw) = \mathcal{U}_{T,u}(pw)$. The fresh labels agree. Induction on the remaining query length proves equality of the new unfoldings. Iterating proves continuation stability.

For the converse, choose a shortest port word $p_1 \cdots p_m$ on which the unfoldings differ. If $m = 0$, the current labels already differ. Otherwise a unary tester stores a cursor in one port. Its first row points along p_1 ; at the next step the distance-two path through the cursor and p_2 advances it; and so on. All proper prefixes behave identically by minimality. The last neighborhood exposes the first label or missing-edge difference, which finite control records in acceptance. $\square$

Corollary 2.4. *Both $\equiv_s$ and $\equiv_f$ are stable under a common right suffix of rows.*

2.4 The canonical behavioral figure

For a valid port word p , write $\mathcal{U}_p(w) = \mathcal{U}(pw)$. Let $\mathcal{R}(\mathcal{U})$ be the finite set of distinct valid residual unfoldings.

Theorem 2.5 (Minimal residual figure). *Every finite rooted scaffold has a canonical finite behavioral figure $M(\mathcal{U})$ whose states are $\mathcal{R}(\mathcal{U})$, whose root is $\mathcal{U}_\epsilon$, and whose port i sends $\mathcal{U}_p$ to $\mathcal{U}_{pi}$ when defined. The physical reachable graph maps onto $M(\mathcal{U})$ by a surjective functional bisimulation. Two rooted scaffolds are behaviorally equivalent if and only if their minimal figures are isomorphic.*

Proof. Every residual is represented by a physical reachable node, so there are at most as many residuals as nodes. Residualization commutes with every port, which makes labels and successors well defined and makes the node-to-residual map a functional bisimulation. Distinct residual functions have distinct unfoldings and cannot be bisimilar. The construction depends only on $\mathcal{U}$, proving minimality and uniqueness up to rooted labelled port-graph isomorphism. $\square$

Remark 2.6. Physical aliasing is finer than behavior. Two ports may point to one node in one history and to two bisimilar nodes in another; their minimal figures still agree. Alias identity becomes observable only when a presentation explicitly stores it in finite labels.

3 Chronological order and navigation

Write $x \preceq y$ when x is reachable from y , including equality. A finite port graph is *weakly acyclic* when its only directed cycles are self-loops. Its rooted component is a *reachability chain* when $\preceq$ is total there.

Theorem 3.1 (Generated-chain theorem). *Let S be a legally generated chronological scaffold with current top t .*

1. *Its physical graph and its minimal figure are weakly acyclic.*
2. *The nodes reachable from t , and the states of the minimal figure, are totally ordered by reachability.*
3. *Conversely, every finite behaviorally reduced labelled rooted reachability-chain port graph, with an optional least unlabelled edgeless state, has an exact legal scaffold presentation at some finite distance.*

Proof. A non-SELF edge of node t reaches an older node, so creation time never increases along a port path and can stay fixed only on a self-loop. If the behavioral quotient had a cycle through two classes, repeating a word around the cycle from a physical representative would give an infinite path of nonincreasing times. It is eventually constant, so every later edge is one physical self-loop and all classes on the alleged cycle coincide.

Let R_j be the component reachable from the top after step j . Every old target of the new top is already in R_{j-1} , so $R_j \setminus \{j\} \subseteq R_{j-1}$. Induction makes R_j a chain, and quotienting preserves total comparability.

Conversely enumerate the desired states $q_1 \prec \dots \prec q_m$. When q_j is appended, each strict successor is some q_i below q_{j-1} and is therefore reached from q_{j-1} by a port word. Use that word, SELF for self-loops, and missing for absent edges. A shortest simple word has length at most $j - 2$, so one distance at most $m - 1$ realizes all rows. $\square$

Corollary 3.2 (Regressive navigation). *After adjoining a least missing sink to a generated minimal figure and numbering its chain, every port acts regressively. Its port-word transition monoid is a submonoid of the full regressive transformation monoid and is $\mathcal{R}$ -trivial.*

Proof. Each successor is at or below its source. If transformations f, g are $\mathcal{R}$ -related, write $f = gu$ and $g = fv$. Pointwise regressiveness gives $xf \leq xg$ and $xg \leq xf$ for every state x , hence $f = g$. $\square$

Boundary. This is the algebra of historical port navigation. A separate finite root-routing interface can contain partial permutations and Coxeter relations. The historical navigation itself is irreversible and is not a braid group.

4 Static guarded folding

Let G be a finite deterministic labelled partial port graph and let E be a bisimulation equivalence already imposed on it. Work in G/E .

Definition 4.1 (Elementary guarded fold). Two distinct quotient states x, y admit a guarded fold when their labels agree and, at every port, the two successors are both missing, are the same quotient state, or lie in $\{x, y\}$. Equivalently, identifying x and y makes their one-step records equal.

The last case is the guarded recursive clause. It permits self-reference at either member of the pair but no larger unresolved cycle.

Theorem 4.2 (Guarded folds generate bisimilarity). *On every finite weakly acyclic deterministic labelled partial port graph, the equivalence generated by elementary guarded folds is exactly the greatest bisimulation equivalence.*

Proof. Each fold is sound because corresponding successors become equal in the new quotient.

For completeness, suppose a sound quotient still has a nonsingleton greatest bisimulation class. Choose such a class C minimal in strict reachability of the behavioral quotient. Every strict successor class below C is already a singleton.

The edges between distinct current states inside C form a finite DAG. If it has two sinks x, y , an internal successor of either sink can only be its own self-loop, while corresponding external successors are the same singleton; therefore x, y are guarded-foldable. If it has one sink x , choose a minimal state y of $C \setminus \{x\}$. Every strict internal successor of y is x , and every internal successor of x is its self-loop. External successors are again equal, so x, y are guarded-foldable.

Another sound fold is therefore available until the greatest bisimulation quotient is reached. $\square$

Corollary 4.3 (Static normal form). *Orient each guarded fold toward its quotient. The system terminates and is confluent up to rooted labelled port-graph isomorphism. Its normal form is the minimal residual figure.*

Proof. Every fold reduces the state count. An irreducible graph is behaviorally reduced by Theorem 4.2; every reduction preserves the unfolding; and Theorem 2.5 gives one reduced figure up to isomorphism. $\square$

This theorem contains the usual bottom-up acyclic merge as a special case. The extra local case is a self-recursive pair. Static normalization alone, however, may remove a physical state and therefore does not yet define a time-respecting rewrite of construction histories.

5 The causal presentation calculus

5.1 Strong equivalence and semantic sewing

A *semantic step replacement* changes one row over a fixed preceding prefix when the two possible fresh roots are behaviorally equivalent.

Proposition 5.1 (Semantic causal sewing). *Two strongly equivalent length- n histories are connected by at most n semantic step replacements.*

Proof. Align the rows from left to right. Once the prefixes through $j - 1$ are literally equal, replaying the old j th row over that prefix preserves its behavior by Theorem 2.3; the target row has the same j th figure by strong equivalence. Replace it and continue. $\square$

This removes any semantic obstruction. The real problem is to replace the global behavioral side condition by a fixed family of bounded graph-local moves.

5.2 One-state extensions

Let Q be a behaviorally reduced old live figure. A new row has the guarded one-variable form

$$R = (a; t_0, \dots, t_{d-1}), \quad t_i \in Q \cup \{X, \perp_e\},$$

where X denotes the fresh root.

Lemma 5.2 (One-state extension dichotomy). *Suppose two such fresh roots have the same behavior.*

1. *If the behavior is not already a state of Q , their missing and SELF positions agree, and their old target agrees at every other port.*
2. *If the behavior is an old state $q \in Q$, each row is a guarded unroll of q : it copies every missing and strict-successor port, while a self-loop of q may target either q or the fresh variable X .*

Proof. In the first case, a fresh SELF on one side and an old target q on the other would make the common fresh behavior equal to q , a contradiction. Thus the SELF masks agree. Corresponding old targets are bisimilar and are therefore the same state of reduced Q ; missing positions also agree.

In the second case, compare the fresh root with q . A non-SELF target must be the unique reduced successor of q . A fresh SELF is possible exactly where that successor is q itself. These conditions are also sufficient: identity on Q together with the fresh-root/ q pair is a bisimulation. $\square$

5.3 The two primitive strong moves

Fix a total order

$$\text{MISSING} < \text{SELF} < p \quad (p \in P^{\leq k})$$

and a total order on the port words.

Definition 5.3 (Endpoint detour D). Replace one old-top word by another bounded word with the same physical endpoint. All invalid words share the missing endpoint. Orient the move toward the least name of that endpoint.

Definition 5.4 (Literal guarded unroll U). Suppose a fresh row presents an old residual q . It must copy every strict successor of q . At a self-loop port of q , it may point to the old physical q or use fresh SELF. If at least one such old pointer appears, replace all of them by SELF and choose least endpoint names at the other ports. Orient in that direction.

The simultaneous recursive part of U is *absorption* A ; its remaining changes are endpoint detours. The guard is physical and bounded: it names q within the observed scaffold and inspects its fixed port tuple.

These are mathematical graph-local rewrites with explicit endpoint-equality certificates. The calculus does not assert that an LMR automaton can discover or execute a normalizing move from finite control, nor that the move occupies a bounded contiguous interval in the chronological row word.

5.4 SELF-eager prefixes

Call a canonical history *SELF-eager* when it uses SELF at every available self-loop coordinate and least names elsewhere.

Lemma 5.5 (Canonical history exists). *Every realizable finite strong trace has a unique lexicographically least history. It is obtained from left to right by appending the least row that realizes the next figure over the already chosen prefix.*

Proof. Assume the canonical prefix through time $j - 1$ has been chosen and take any original realization of the full trace. Its length- $(j - 1)$ prefix has the same rooted unfolding as the canonical prefix. Replay the original j th row over the canonical prefix. Theorem 2.3 says that the new root has the required j th figure, so the finite set of realizing rows is nonempty and has a unique least element. Induction constructs the history; leastness at the first differing row proves uniqueness. $\square$

Lemma 5.6 (Reduced live prefixes). *Every root-reachable component at every prefix of a SELF-eager history is behaviorally reduced.*

Proof. Induct on prefix length. If the fresh behavior is new, it adds one new residual above a reduced old component. If it repeats an old residual q , Lemma 5.2 says that every strict successor lies below q and that only a self-loop coordinate can choose between old q and fresh SELF. SELF-eagerness chooses the fresh root. It therefore reaches itself and strict descendants but not the old physical q . The fresh root replaces q as the unique live representative. $\square$

Theorem 5.7 (Convergent strong-trace presentation). *For fixed finite d, k, Γ , oriented endpoint detours and literal guarded unrolls form a terminating confluent graph-local presentation of strong equivalence. Two histories have the same strong trace if and only if they reduce to the same SELF-eager history. A length- n history normalizes in at most dn row moves when U counts as one whole-row move.*

Proof. Both moves are sound. A detour preserves a physical endpoint. For an unroll, identity on the old prefix together with the fresh-root/ q pair is a bisimulation. Theorem 2.3 carries either replacement through the unchanged suffix.

Lexicographically order the chronological descriptor word. A detour strictly decreases one path name. An unroll replaces at least one old path by smaller SELF and never increases an earlier descriptor. Thus reduction terminates.

Induct left to right through an irreducible history. Its old prefix is already SELF-eager and reduced by Lemma 5.6. Compare the fresh row with the least row supplied by Lemma 5.5 for the same next figure. If the residual is new, Lemma 5.2 gives the same missing/SELF mask and the same old residual targets. Reducedness turns those targets into the same physical nodes; any remaining difference is a detour redex. If the residual repeats old q , a path back to q at a self-loop is an unroll redex, and a nonleast path to any fixed strict successor is a detour redex. Therefore an irreducible row is exactly the SELF-eager least row.

Each strong-trace class has one irreducible history, so termination gives confluence and completeness. A new residual needs at most d detours. A repeated residual needs at most d detours or one whole-row unroll; because $d \geq 1$, the total is at most dn . $\square$

Remark 5.8 (What “coherence” means here). Theorem 5.7 is object-level Church–Rosser coherence: equal strong traces have one normal form. It does not identify every two reduction paths modulo a finite list of two-dimensional cells. We reserve *path-level coherence* for results that actually present those path relations.

6 Final observation and replay support

Final equivalence forgets intermediate figures. A row that is unreachable in the final rooted graph can nevertheless have been queried while a later descriptor was evaluated. Ordinary graph garbage is therefore too small.

Definition 6.1 (Causal endpoint support). Let R_H be the created nodes reachable from the final top. Support every label and every outgoing port cell of a node in R_H . Whenever a supported port cell was created by following an old-top path, support every older port cell actually queried along that evaluation, including the cell at which an invalid path fails. Close backward to the least such set C_H .

Theorem 6.2 (Replay invariance). *Suppose an equal-length history H' agrees with H on the labels of R_H and on every descriptor cell in C_H . Then the final rooted concrete scaffolds of H and H' are identical. Moreover $R_{H'} = R_H$ and $C_{H'} = C_H$.*

Proof. Proceed chronologically through supported descriptor cells. Every dependency of a supported cell is an earlier supported cell, so induction makes its evaluation traverse the same physical endpoints on both sides. The outgoing edges and labels of all final-reachable nodes are therefore identical, giving the same rooted concrete scaffold and the same reachable set. Recomputing the least backward closure follows the same descriptor traces, hence gives the same support. $\square$

Definition 6.3 (Garbage fibre). $\mathcal{G}(H)$ is the set of histories agreeing with H on the data named in Theorem 6.2. A G -move changes any collection of the remaining data.

Theorem 6.4 (Causal-garbage cube). *Every garbage fibre is the Cartesian product of the finite value sets of its unsupported label and descriptor coordinates. Simultaneous G factors into single-coordinate changes. Fill a triangle for three values of one coordinate and a square for changes of two distinct coordinates. The resulting path two-complex is simply connected.*

Proof. Support stability in Theorem 6.2 makes every unsupported coordinate independent. Order the coordinates. Commuting squares move all changes of the first coordinate to the front, then the second, and so on. Same-coordinate triangles compose each block to the direct change between its endpoint values. Every path reduces to one coordinate-ordered path; a loop reduces to the empty path. $\square$

6.1 A noncommuting boundary

Let S denote strong normalization and let J choose least values outside causal support. Both preserve the final figure, but in general

$$S \circ J \neq J \circ S.$$

An absorption can move a repeated residual to a newer physical carrier. Junking first may erase the label needed to perform that absorption; absorbing first makes the older carrier garbage. Thus the two normalizers can reach distinct common fixed points with one final figure already in the unary distance-one three-row fragment.

Concretely, take

$$H = (1, \text{SELF}); (1, \epsilon); (0, 0).$$

Strong normalization first absorbs the second row and replay moves the last endpoint to that newer carrier; subsequent junking gives

$$H_{SJ} = (0, \text{MISSING}); (1, \text{SELF}); (0, \epsilon).$$

Junking first keeps the second descriptor because the last row queries it, but erases its unsupported label, giving the already strong-normal history

$$H_{JS} = (1, \text{SELF}); (0, \epsilon); (0, 0).$$

Both final roots are labelled 0 and point to a 1-labelled self-loop, but the two histories are distinct and irreducible under both normalizers.

Boundary. Object-level confluence for strong traces plus cubical coherence inside each garbage fibre does not present their join. The missing operation changes which chronological node carries a final residual and therefore changes the support boundary itself.

7 Two calibrated sewing theorems

This section does not assert a complete final-equivalence calculus at arbitrary degree and distance. It proves one local two-row theorem and one global unary theorem. Their different hypotheses show where the remaining problem lies.

7.1 Unexposed acyclic two-row factorizations

Fix a behaviorally reduced old prefix with top q , append a router r , then a consumer z . Assume no consumer port points to r by the empty descriptor. Whenever a consumer word begins with router slot i , assume that slot contains an old-prefix path α_i , not SELF or missing. Thus a consumer word $i\beta$ reaches

$$q_{\alpha_i\beta}.$$

Call such a block *unexposed and acyclic*.

Three final-only moves are useful.

1. A *routing crossing* injectively reassigns used router slots and changes each first selector to follow the reassignment.
2. A *cut slide* moves one letter between α_i and β when both bounded descriptors remain legal.
3. A *dedicated relation exchange* changes (α, β) to (α', β') in a slot selected by exactly one consumer port, under the guard $q_{\alpha\beta} = q_{\alpha'\beta'}$.

The first two refactor a fixed composite word. The third uses a relation in the pointed old navigation action and is the move that crosses between different words with one old endpoint.

Theorem 7.1 (Unexposed two-row sewing). *Fix $d, k \geq 1$, and a behaviorally reduced old prefix. Consider two unexposed acyclic two-row extensions with the same consumer label, the same valid nonmissing path-demanding consumer coordinates, and literal agreement as missing or SELF on every other consumer coordinate. If corresponding path demands have the same final old residual behavior, the blocks are connected by endpoint detours, causal-garbage moves, continuation-aware routing crossings, and dedicated relation exchanges. Every common suffix is preserved.*

Proof. Reducedness makes equal old residual behaviors the same physical old state. Canonicalize every missing consumer coordinate to literal missing by a detour.

If several path demands use one first selector, fewer router slots than demands are currently used. Copy that router descriptor into an unsupported slot by G and retarget one consumer word to the copy by a detour; its complete physical endpoint is unchanged. Iteration makes the used selectors injective. Routing crossings put both blocks into one fixed injective slot placement.

Now each demand has a dedicated slot. Its current and target factorizations reach the same old physical state. One dedicated relation exchange changes that pair without affecting another demand. Process all demands, align the unused cells and router label by G , and reverse the target-side selector-splitting steps. Every move preserves the consumer endpoint vector, so the consumer-rooted concrete graph is identical and every common suffix replays identically. $\square$

Remark 7.2. Dedicated relation exchange has a bounded graph-local guard for fixed d, k , but it is not a rowwise endpoint detour. It explicitly cites equality between two composite old paths across the router/consumer cut. Whether it is derivable by refactoring older rows is open. Theorem 7.1 is therefore a presentation *relative to the pointed endpoint congruence of the old prefix*; it is not a finite relation list independent of that old action.

7.2 The unary whole space

Now set $d = k = 1$. Every descriptor is one of

$$\text{MISSING}, \quad \text{SELF}, \quad \epsilon, \quad 0.$$

After absorbing repeated live presentations of a terminal recursive residual, the final live figure is a reduced chain.

Lemma 7.3 (Unique relay). *If consecutive states $u > v$ of that final reduced chain are carried at nonadjacent construction times, the supported descriptor segment between them is, up to endpoint detours at a recursive bottom,*

$$\epsilon 0 0 \cdots 0.$$

Proof. The first row after the lower carrier selects that carrier by ϵ . Every later supported relay row, including the upper carrier's edge to the lower residual, must use 0 to pass the same endpoint upward; missing or SELF would terminate the relay, while a later ϵ would make the immediately preceding row final-reachable and insert another live residual between u and v . If the endpoint is a self-loop residual, ϵ and 0 can name the same old endpoint and one detour chooses the displayed form. $\square$

A *temporal carrier slide* exchanges the adjacent relay patterns

$$(\epsilon, 0) \longleftrightarrow (0, \epsilon)$$

when the two potential carrier rows have the same label. The endpoint calculation is literal: the left side reaches the old endpoint through the newer clone, while the right side makes that row a bridge and reaches the older carrier. The final figure is unchanged although the intermediate figure and support carrier may change.

Theorem 7.4 (Unary final presentation). *Fix a length n , a nonempty finite working alphabet, degree one, and distance one. Two histories have the same final behavioral figure if and only if they are connected by endpoint detours, absorption, causal-garbage changes, and temporal carrier slides. Every final fibre contains one literal right-packed normal form determined by n and the final figure.*

Proof. Soundness has been proved for each move. For completeness, absorb every final-live finite unroll of a recursive residual. The remaining live chain is behaviorally reduced.

Apply Lemma 7.3 between each two consecutive live residuals. Prepare the first hidden relay label by a garbage move. An inverse temporal slide advances the lower carrier through that relay time; if it is recursive, absorption restores the SELF presentation. Each slide decreases the sum of gaps between consecutive essential carriers. Working from the final root downward terminates with the whole reduced live chain in the latest possible consecutive rows.

Canonicalize endpoint spellings by detours. Below the last essential carrier there are three cases. A missing terminal leaves only causal garbage; a recursive terminal leaves a canonical SELF bottom block; and a terminal at the distinguished base has the unique supported relay $\epsilon 0 \cdots 0$, with only its unused labels free. Garbage moves choose the least remaining data. The result depends only on n and the final figure, and is the right-packed history $RP_n(F)$. Histories in one final fibre both reach that form. $\square$

7.3 A path-level quotient

In the unary theorem, quotient a final fibre by its internal strong and garbage moves. The remaining coordinates are the creation times of the essential residual carriers, omitting the arbitrary physical carrier of a terminal SELF residual. Temporal slides move one carrier into one adjacent vacant time.

Proposition 7.5 (Carrier-surgery coherence). *For a fixed unary distance-one final figure and history length, the temporal slide graph on strong/garbage chambers is the undirected Hasse graph of the finite lattice of allowed increasing carrier tuples. Filling every diamond formed by two independent slides makes the quotient two-complex simply connected.*

Proof. The unique-relay lemma shows that strong and garbage moves preserve the essential tuple and that all histories with one tuple lie in one chamber. The temporal endpoint calculation realizes exactly an adjacent move into a vacancy. Componentwise order on fixed-size subsets is the order-ideal lattice of a rectangle.

Orient its edges upward. At a maximal-rank vertex of a loop, equal incident edges cancel as a backtrack. Otherwise the two downward moves remove distinct maximal boxes and form a commuting lattice diamond. Replacing the peak by the other two sides strictly lowers the maximal-rank profile. Iteration contracts the loop. $\square$

Boundary. Proposition 7.5 is path-level coherence only after collapsing each strong/garbage chamber. Theorem 6.4 gives path coherence inside a garbage fibre. A finite elementary presentation of all mixed strong, garbage, and temporal derivations is not proved here.

8 Feedback, receipts, and the PC-83 defect

The preceding calculus was developed intrinsically. A direct palindrome construction supplies a deliberately hostile application because it forces a fresh feedback edge.

8.1 Typed receipts

The direct recognizer for

$$\text{EPAL}_2 = \{ww^R : w \in \{0,1\}^*\}$$

maintains the longest even-palindromic suffix of the current prefix and a reversed context for its current occurrence. A *typed receipt* is not merely a palindrome type or an untyped historical endpoint. It points to the particular old occurrence record whose immutable suffix-link and transition ropes will be used later. Persistent receipt ropes transport these pointers across pop, prepend, reset, and catenation updates. The full recurrence and its representation-specific escape lemmas are proved in the separate direct paper [4].

The load-bearing fresh-record equation has the form

$$\Omega_b(P) = (H; \mathcal{T}(a, P)).$$

The new occurrence owner P stores $\Omega_b(P)$, while the rope's terminal receipt points back to P . Thus the same input step contains the logical cycle

$$P \longrightarrow \Omega_b(P) \longrightarrow P.$$

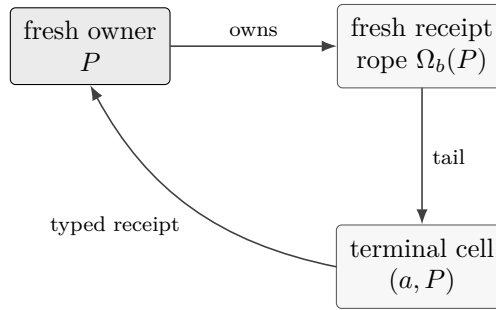


Figure 2: The same-step receipt knot. A topologically ordered fresh allocator cannot construct this finite strongly connected component.

An earlier PC-83 checkpoint allowed a fresh field to name only an *already constructed* fresh slot. Its compiler checker imposed the same topological order. Figure 2 is therefore a genuine proof defect in that version: the claimed source transition was outside its own model.

8.2 Bounded simultaneous record transducers

Definition 8.1 (Simultaneous record batch). Fix constants s (roots), A (fresh slots), q (pointer fields per slot), and r (old dereference radius), together with finite control, tag, and payload alphabets. A transition may inspect tags and payloads reached within r old logical field dereferences from its roots. From that old observation it chooses the next control and one finite batch of at most A

records. Each fresh field and next root targets missing, a bounded old address, or *any* active slot of the same batch. No fresh slot is inspected while the plan is selected, and control does not branch on physical pointer identity.

The last clause permits self, forward, and mutual fresh references. The batch is a finite labelled graph specified all at once.

Theorem 8.2 (Simultaneous-batch compiler). *Every bounded simultaneous record transducer with parameters (s, A, q, r) has an exact letter-synchronous scaffolding-automaton implementation of degree*

$$d = s + Aq$$

and observation/update distance at most $r + 1$.

Proof. One physical scaffold node represents the whole logical batch. Its finite label records the active mask; every active tag and payload; and, for every root and field, the finite target-slot tag.

Ports $0, \dots, s - 1$ represent roots. Port $s + iq + j$ represents field j of logical slot i . A decoded pointer is a pair (v, i) of a physical historical node and a slot. An old logical address beginning at root u and following fields $j_1, \dots, j_m$, $m \leq r$, compiles to the physical word

$$u, \quad s + i_0q + j_1, \quad \dots, \quad s + i_{m-1}q + j_m,$$

where each next slot i_h is read from the finite target tag at the preceding edge. The observed depth- $(r + 1)$ neighborhood contains this information.

Use a missing descriptor for missing, the compiled word for an old target, and physical **SELF** for every fresh target. In the last case the new label supplies the requested target slot.

Assume the old physical heap decodes correctly. Induction along a logical old address proves that its physical word reaches the same old pointer. Missing decodes immediately. A fresh descriptor reaches the node currently being appended and its tag selects exactly the requested fresh slot. Applying these three cases to every root and field makes the decoded new finite adjacency table identical to the source batch, including all cycles. This is equality of finite labelled graphs; no recursive graph unfolding or fixed point is needed. Run induction gives exact simulation. All alphabets and plan branches are finite, proving the stated degree and distance bounds. $\square$

Remark 8.3 (Where the cycle lives). There is no contradiction with Theorem 3.1. The physical scaffold still has only backward edges and self-loops. A logical record pointer is the larger coordinate (v, i) , not the physical node v alone. Following a physical **SELF** edge keeps v fixed while its finite edge tag changes the logical slot from i to i' . An arbitrary finite cycle in the decoded fresh batch is therefore represented by one physical self-loop together with finite slot dynamics. The coordinate realization has moved cyclicity from physical time into the finite decoder state.

8.3 Staging a strict persistent sequence

The compiler solves the target representation problem but not the source evaluation problem. A cycle is safe only if the program can choose its allocation skeleton without following the fresh feedback edge.

Theorem 8.4 (Opaque-element staging). *Let a strict purely functional sequence operation:*

1. use only finite constructor tests and a worst-case bounded number of fixed-arity constructor, projection, and pointer steps;
2. treat stored elements as opaque atoms, with no equality, hashing, pattern match, identity test, or traversal through an element; and
3. return a fixed finite tuple of roots and removed elements.

For fixed old roots and symbolic new elements, its control branch and fresh structural allocation graph are determined independently of the symbolic values. If one symbol denotes a fresh owner that stores a resulting sequence root, the owner and structural graph form one bounded simultaneous batch and compile by Theorem 8.2.

Proof. Run the bounded operation with distinct symbols in place of new elements. Induct over its bounded primitive steps. A constructor test or projection reads only an old structural record or an earlier fresh structural record. Element opacity prevents the control path from following or branching on a symbol. A constructor step adds one fixed-arity structural record whose fields are old pointers, earlier structural pointers, or symbolic elements. Thus the branch and an acyclic bounded structural allocation graph are fixed before element substitution.

Add the fresh owner slot and replace its symbol by a pointer to that slot. The substitution can create a cycle from the owner through the sequence structure and back through an opaque element field, but it cannot change the selected branch or work bound. Theorem 8.2 represents every edge of the resulting finite graph using old paths, missing, or SELF plus a slot tag. $\square$

Kaplan and Tarjan define “purely functional” for their catenable structures as using only `car`, `cons`, and `cdr`, explicitly excluding memoization. Their Theorem 5.1 gives worst-case constant-time push, pop, inject, and catenation for regular catenable steques [6]. Elements at the base type are parameters to those structural operations. The representation may inspect its own fixed pair and buffer constructors at recursive levels, but it does not inspect the typed receipt pointer stored at a payload leaf. Hence their theorem supplies the source premise required by Theorem 8.4, not merely an asymptotic slogan.

Corollary 8.5 (Corrected receipt application). *The fresh owner/receipt cycle in Figure 2 is implementable by one bounded simultaneous scaffold batch. Combined with the typed-receipt recurrences and strict catenable-steque implementation proved in the direct paper, it yields a finite scaffolding automaton for EPAL_2 . Since EPAL_2 is reversal invariant, LMR Theorem 16 gives $\text{EPAL}_2 \in \text{PEG}$.*

Proof. Receipt-rope updates are a fixed finite composition of head/tail, singleton, inject, and catenation operations on opaque typed pointers. Theorem 8.4 produces their bounded structural batch without inspecting the fresh receipt. Add the occurrence owner and tie every symbolic occurrence of that receipt to its fresh slot. Theorem 8.2 compiles the resulting cycle. The direct paper proves the semantic receipt recurrences and acceptance test. LMR supplies the final scaffold-to-PEG implication. $\square$

Remark 8.6. Corollary 8.5 repairs the allocation seam; it is not a first even-palindrome membership result. FPRD Lab already has an independent strict-real-time-machine transfer proof [5].

9 What the FPRD approach contributed

The point of the investigation was not to attach algebraic words to a solved language-membership exercise. The FPRD separation in Figure 1 repeatedly changed the mathematical target.

1. Treating physical endpoint equality and behavioral equality as different observations produced the unfolding quotient and exposed pure aliases as presentation data.
2. Retaining every prefix figure, rather than only the final figure, produced a right-stable strong congruence and the detour/unroll normal form.
3. Forgetting prefixes only after that theorem exposed temporal carrier motion and causal support as new final-observation coordinates.
4. Treating a receipt as a typed coordinate, rather than a semantic existence assertion, exposed the same-step allocation cycle.
5. Moving from topological allocation to a simultaneous coordinate realization turned that apparent impossibility into ordinary **SELF** feedback with finite slot state.

The method can be summarized as

$$\boxed{\text{presentation} \longrightarrow \text{exact dynamics} \longrightarrow \text{observation} \longrightarrow \text{change of representation.}}$$

Each arrow is a theorem obligation. In particular, a universal graph embedding is not automatically a bounded coordinate realization, and final language equality does not imply equality of causal histories.

10 Open boundaries

The proved general object is strong-trace equivalence. The broader program begins where final observation forgets that trace.

1. **Unrestricted final presentation.** Determine whether a finite graph-local move family presents final behavioral equivalence at all fixed degrees and distances. Dedicated relation exchange may be a primitive or may be derivable by refactoring older rows.
2. **Elementary higher coherence.** The strong normalizer gives one object normal form, garbage fibres are cubical, and unary carrier chambers form distributive lattices. A finite set of cells presenting all mixed derivations is open.
3. **Support handoff.** A temporal slide can transfer one visible label obligation across a causal cut. Classify its critical pairs with detours and absorption.
4. **Navigation monoids.** Characterize which finitely generated submonoids of the regressive chain monoid occur at fixed degree, distance, and uniform finite control.
5. **Functional sharing versus branching.** Separate ordinary immutable record DAGs from unrestricted scaffold navigation first at the dynamic or observational level, before seeking a language separation.
6. **Comparator theorem.** Construct the precise functor, if one exists, from the guarded causal presentation calculus to an established cyclic term-graph or polygraphic rewriting system. Novelty should be judged only after that comparison.

11 Assurance and nonclaims

The theorem proofs in this paper are ordinary written mathematical proofs. Executable enumeration is used separately to find counterexamples and corroborate finite instances. No proof assistant is used as a stronger label for the mathematics here.

The public package includes a theorem inventory, proof audit, source audit, finite audit code selected from the research corpus, and exact output. Those layers answer different questions:

Layer	What it establishes
Written proof	the stated theorem for all objects satisfying its hypotheses
Finite audit	exact behavior over one explicitly bounded search domain
Source audit	the external theorem or model says what the paper attributes to it
Specialist review	an independent reader has attacked correctness and positioning
Novelty review	the contribution has been compared broadly enough to support priority language

At this review stage no novelty claim is made. In particular:

- coalgebraic minimization and functional bisimulation are inherited;
- ordinary acyclic-state merging is inherited;
- general cyclic term-graph fold/unfold and confluence are inherited;
- Coxeter relations for partial routing and cubical coordinate changes are inherited mathematical forms;
- no unrestricted final-equivalence or finite higher-coherence theorem is claimed; and
- the palindrome corollary is not the first proof of membership.

12 Conclusion

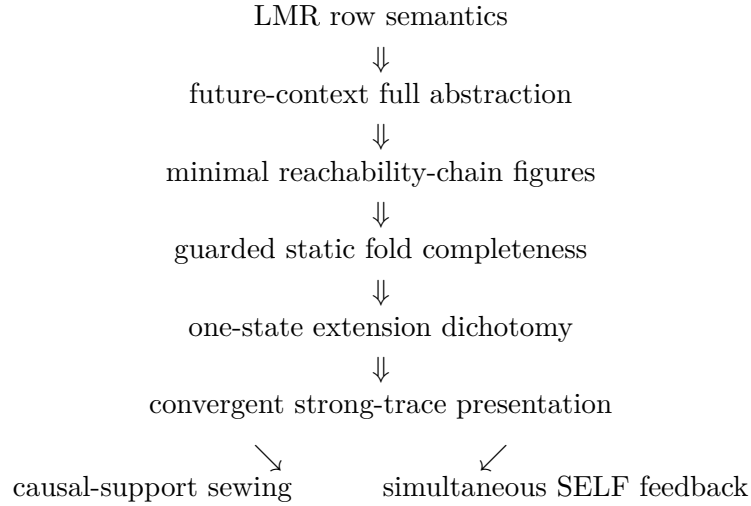
Persistent scaffold computation has an intrinsic algebra before one asks which language a machine recognizes. Its generated behavioral figures are ordered residual systems. Guarded static folds collapse equivalent presentations; chronological detours and one **SELF**-unroll lift that collapse to a convergent presentation of full strong traces. Final observation then reveals a second layer—causal support, garbage cubes, temporal carriers, and cross-cut sewing—whose unrestricted presentation remains open.

The receipt knot explains why this distinction matters computationally. The semantic record exists, but a topologically ordered presentation cannot build it. In a simultaneous scaffold coordinate system the same cycle is a finite batch of **SELF** edges with slot tags. That change of representation repairs the palindrome application, but more importantly it exhibits the general principle: complexity can move between semantic structure, presentation dynamics, and observation without disappearing.

A Move and preservation table

Move	Local action	Guaranteed preservation
D	rename one bounded word with the same physical endpoint	every prefix figure and every common suffix
A/U	replace an old recursive carrier by fresh SELF	every prefix figure and every common suffix
G	change data outside one source-certified replay support	final rooted concrete scaffold
T	move a unary residual carrier across one adjacent time cut	final behavioral figure
R	reassign demanded router slots injectively and transport continuations	final concrete consumer endpoints
E^{rel}	exchange two dedicated factorizations with one old endpoint	final concrete consumer endpoints

B The central implication map



References

- [1] Zena M. Ariola and Jan Willem Klop. Equational term graph rewriting. *Fundamenta Informaticae*, 26(3–4):207–240, 1996. <https://doi.org/10.3233/FI-1996-263401>.
- [2] Jan Daciuk, Stoyan Mihov, Bruce W. Watson, and Richard E. Watson. Incremental construction of minimal acyclic finite-state automata. *Computational Linguistics*, 26(1):3–16, 2000. <https://aclanthology.org/J00-1002/>.
- [3] Tobias Fritz and Wendong Liang. Free gs-monoidal categories and free Markov categories. arXiv:2204.02284, 2022. <https://arxiv.org/abs/2204.02284>.

- [4] Joshua Gay. Persistent receipts and simultaneous records: a direct scaffold for even palindromes. FPRD Lab review paper, 2026. <https://gitlab.com/insight-forge/boxlab/fprd-lab/-/tree/main/publications/direct-palindrome-scaffolding>.
- [5] Joshua Gay. Real-time multitape languages transfer to parsing expression grammars. FPRD Lab review paper, 2026. <https://gitlab.com/insight-forge/boxlab/fprd-lab/-/tree/main/publications/persistent-stacks-scaffolding>.
- [6] Haim Kaplan and Robert E. Tarjan. Purely functional, real-time dequeues with catenation. *Journal of the ACM*, 46(5):577–603, 1999. <https://doi.org/10.1145/324133.324139>.
- [7] Bruno Loff, Nelma Moreira, and Rogério Reis. The computational power of parsing expression grammars. arXiv:1902.08272v2, 2020. <https://arxiv.org/abs/1902.08272>.
- [8] Jan J. M. M. Rutten. Universal coalgebra: a theory of systems. *Theoretical Computer Science*, 249(1):3–80, 2000. [https://doi.org/10.1016/S0304-3975\(00\)00056-6](https://doi.org/10.1016/S0304-3975(00)00056-6).