June 2026

Switch Corridors and Boundary-Rigid Lenses in Even Dumbbell Thrackles: A Local Reduction and Minimal-Witness Strategy

Luca Blanchi

Abstract

We give a detailed reduction framework for the irreducible even-dumbbell stage of Conway's thrackle program. The starting point is a non-T_3 even dumbbell thrackle. A finite domain-incidence argument first selects two ordinary same-cycle vertices p, q with separated incident-domain data. The two complementary p-to-q paths are then treated as anchored traces rather than as free local corridors: every subsequent simplification must preserve the original endpoint data or record a precise lower subproblem. From this anchored setup we construct typed carriers, prove a no-backtracking theorem by a well-founded minimal-witness cost, and compile the remaining reachability data into ordered switch corridors. The corridor analysis separates regular switching cells from the genuinely singular residuals: common-vertex marker data, support-gate disks containing original vertices, blocked R3 moves, replacement lenses, and endpoint failures. Each residual is routed to a terminal contradiction, a strict descent, a lower named subproblem, or a finite local verification. The internal closure theorem states that, once the listed finite local verifications are supplied, no unresolved irreducible non-T_3 even-dumbbell obstruction survives. Thus the manuscript reduces the internal Conway step to a finite collection of explicit local checks together with the standard external reductions to the even-dumbbell and T_3 cases.

Contents

1. Overview of the Reduction

The target of this paper is the internal even-dumbbell step in the usual strategy for Conway's thrackle conjecture. We use the following standard external route as background:

$$\text{Conway false} \Rightarrow \text{even dumbbell thrackle} \Rightarrow \text{irreducible even dumbbell thrackle}$$

The external reductions and the theorem that T_3-thrackles satisfy the Conway bound are not reproved here. The object of the paper is the middle implication:

$$\text{there is no irreducible non-}T_3\text{ even dumbbell thracklesurviving the local reducti}$$

The reduction has six conceptual stages.

First, Section 3 replaces the non-T_3 hypothesis by finite domain data. The ordinary vertices determine a bipartite domain graph. By Koenig's theorem, the failure of a three-domain cover supplies explicit matchings. These matchings give two ordinary same-cycle vertices p, q whose incident-domain pairs are disjoint. The point is deliberately modest: this step only produces anchored endpoint data and the two complementary graph paths between p and q.

Second, Section 4 turns one of those graph paths into an anchored carrier. This is where the proof must avoid a common pitfall. A smaller local corridor is not useful if it forgets the original p, q endpoints or their domain, orientation, marker, and replacement data. The carrier is therefore a typed object with terminal sections built into the data. A finite layered reachability graph records which typed boundary states can be connected through the carrier. A minimal-witness argument then removes backtracking: every non-monotone return contains a smaller typed disk, a lower carrier, or a named residual subproblem.

Third, Section 5 regularizes the resulting ordered corridor. Regular alternating cells are kept; nonregular strips are not simply discarded. They are either simplified by a typed identity or same-relation splice, shown to lower the global cost, or exported to one of the residual analyses. This is the main place where the switch-corridor language enters.

Fourth, Sections 6 and 7 handle the residuals caused by the graph itself. The common vertex is treated by a rank-one marker record: passages through v are singular events and cannot be silently replaced by ordinary continuations. Support-gate disks containing ordinary original vertices are reduced by port graphs, short-port chains, and blocked-R3 alternatives. The role of these sections is not to declare every residual terminal; it is to assign each residual a controlled destination.

Fifth, Sections 8 and 9 handle replacement and endpoint phenomena. A proposed R3 move or edge replacement is useful only when the required local data are explicit: endpoint matching, crossing control, absence of hidden vertices or marker events, and persistence of the non-T_3 obstruction. When such data are missing, the first failure is routed to a smaller local problem or to endpoint discharge. Endpoint discharge prevents whole-witness attempts, rejected replacements, and near-whole captures from being counted as progress.

Finally, Section 10 assembles the local reductions by a single well-founded cost. This is the only role played by the presentation-complexity language in the paper. The proof uses a finite description of the current obstruction package and a cost $\mathfrak{K}^*$ that records the global drawing size, marker and anchor defects, carrier complexity, support-gate and ported-disk data, blocked moves, replacement ranks, and the finite multiset of active subproblems. A local step is admissible only if it is terminal, strictly lowers this cost, or replaces the active subproblem by lower-ranked subproblems. This converts the usual minimal-counterexample method into a bookkeeping statement strong enough to rule out cycles among the different local minimalities.

For reference, we organize all local outputs into five classes.

type	meaning	allowed use
T	terminal contradiction	contradicts the thrackle condition, non-T_3 status, or irreducibility
D	strict descent	produces a witness package of strictly lower $\mathfrak{K}^*$
R	routed subproblem	enters a named downstream analysis without losing typed data
C	finite verification	becomes T, D, R, or N after checking a finite list of local cases
N	whole-witness non-descent	rejected as a non-progress move; never used as a contradiction

This taxonomy is part of the proof. A routed output is not a contradiction merely because it has been named. It must be sent to its downstream analysis, verified locally, or shown to lower the declared cost. A finite verification is not terminal until all of its rows have been evaluated. A whole-witness non-descent is bookkeeping, not progress.

We use the word **certificate** only in the standard finite-witness sense: a certificate is explicitly stored finite data, such as a matching in the domain graph or a local replacement table, whose verification is part of the proof. It is not a black-box assertion that a desired local move exists.

The main internal theorem is stated in Section 10. Its proof uses one well-founded cost throughout, so that the different local minimalities of the construction cannot cycle.

2. Witness Packages and the Global Cost

This section formalizes the bookkeeping used in the minimal-counterexample argument. The terminology is taken from presentation complexity only where it does real work: we describe each obstruction by a finite combinatorial description, define a well-founded cost on such descriptions, and require every nonterminal local move to lower that cost or to replace the current problem by lower-ranked subproblems. No probabilistic, algorithmic, or Kolmogorov complexity is being invoked.

After this section, a phrase such as "smaller corridor", "smaller blocker", or "proper endpoint object" has a precise meaning: it is either a strict decrease in the resource scale below, or a handoff to an active subproblem of strictly lower rank.

2.1. Finite descriptions of Conway obstruction packages

Let $\mathcal{X}_{\mathrm{Con}}$ be the class of typed Conway obstruction packages considered up to PL homeomorphism preserving all graph, side, orientation, domain, support, gate, marker, anchor, port, and replacement labels. A description of such a package is a finite planarized combinatorial encoding

$$d = (T, p, q, \gamma_0, \gamma_1, C, \mathcal{L}, \mathcal{M}, \mathcal{D}, \mathcal{B}, \mathcal{O}).$$

The entries have the following roles. The drawing T is the planarized even dumbbell drawing. The vertices p, q are same-cycle ordinary vertices supplied by the domain graph. The paths γ_0, γ_1 are the two complementary p-to-q paths. The carrier C is an anchored carrier, ordered corridor, support-gate carrier, or the local carrier currently being evaluated. The finite layered graph $\mathcal{L}$ records typed reachability states. The marker record $\mathcal{M}$ stores the rank-one common-vertex marker data. The routed disk $\mathcal{D}$ is the current support-gate, ported, endpoint, or dirty-boundary disk, when one is active. The blocker relation $\mathcal{B}$ records blocked R3 and replacement obligations. Finally, $\mathcal{O}$ is the finite multiset of active local obligations still requiring evaluation.

Let $\mathcal{D}_{\mathrm{Con}}$ be the class of all such finite descriptions satisfying the local typing rules. The realization map

$$\rho_{\mathrm{Con}} : \mathcal{D}_{\mathrm{Con}} \longrightarrow \mathcal{X}_{\mathrm{Con}}$$

forgets the chosen encoding and keeps the realized typed obstruction package. The presentation system is

$$\Gamma_{\mathrm{Con}} = (\mathcal{D}_{\mathrm{Con}}, \rho_{\mathrm{Con}}, \kappa_{\mathrm{Con}}),$$

where the cost κ_{Con} is the value $\mathfrak{K}^*$ defined below. This notation is only a compact way to say that we work with finite encodings and a well-founded cost. The relevant bounded part is

$$\mathcal{X}^{\leq b}_{\Gamma_{\mathrm{Con}}} = \rho_{\mathrm{Con}}\{d \in \mathcal{D}_{\mathrm{Con}} : \mathfrak{K}^*(d) \leq b\}.$$

A description is **unresolved** if it realizes a nonterminal obstruction package and has at least one active subproblem not yet evaluated by the reduction scheme. Terminal contradictions, verified descents, and inert whole-witness rejections are not unresolved states.

2.2. The resource scale

The base scale is the lexicographic product

$$B_{\mathrm{base}} = \mathbb{N}^{12}_{\mathrm{lex}}.$$

For a description d write

$$\kappa_0(d) = (G, M, A, R, H, S, E, Q, L, P, \mathsf{B}_{\mathrm{blk}}, U).$$

The coordinates are, in order:

1. G: global obstruction size, for example the original vertex/edge size of the even dumbbell drawing;
2. M: marker complexity at the common vertex;
3. A: anchor defect, measuring loss or displacement of the original p, q anchoring;
4. R: unresolved reachability or backtracking defect count;
5. H: Reeb-height or sweep-event complexity of essential lanes;
6. S: layered typed state count;
7. E: carrier edgelet or planarized crossing count;
8. Q: support-gate loop complexity;
9. L: train or alternating-cell complexity;
10. P: ported-disk complexity, including ordinary original vertices and boundary ports;
11. $\mathsf{B}_{\mathrm{blk}}$: blocker complexity for R3 and replacement candidates;
12. U: residual finite tie-breaker.

The symbol $\mathsf{B}_{\mathrm{blk}}$ is deliberately not denoted B_0, because B_{base} is the base resource scale. This avoids a collision between the scale and the blocker coordinate.

There are two further families of local verification ranks. Let Δ be the well-founded rank set used by diagonal-free alternating enclosures, with rank $\delta(P_i)$ for the i-th active enclosure. Let $\mathcal{R}_{\mathrm{rep}}$ be the well-founded rank set used by pending replacement-edge or R2 obligations, with rank $\varrho(R_j)$ for the j-th active replacement obligation. These ranks are not observables by themselves; they are finite records attached to the corresponding active local problem.

For a well-founded ordered set A, write

$$\mathrm{MSet}_{\mathrm{fin}}(A)$$

for the finite-multiset extension of A. The witness cost before adding active local obligations is

$$\mathfrak{K}(d) = \big(\kappa_0(d), \{\{\delta(P_i)\}\}, \{\{\varrho(R_j)\}\}\big) \in B_{\mathrm{base}} \times \mathrm{MSet}_{\mathrm{fin}}(\Delta) \times \mathrm{MSet}_{\mathrm{fin}}(\mathcal{R}_{\mathrm{rep}}).$$

An active obligation $O \in \mathcal{O}(d)$ is a typed subobject still requiring evaluation: a B3/B4 routed object, marker endpoint, ported B6 disk, blocked R3 candidate, dirty boundary cycle, B7 verification candidate, endpoint object, or whole-witness candidate not yet rejected. Its obligation rank is

$$\lambda(O) = \big(\mathrm{scope}(O), \mathrm{kind}(O), \mathrm{size}(O), \mathrm{cert}(O)\big) \in \mathcal{A}.$$

Here $\mathcal{A}$ is a well-founded obligation-rank set, concretely a finite lexicographic product of finite label ranks and natural-number size ranks. The scope coordinate distinguishes proper cells, proper boundary intervals, proper caps, proper routed disks, whole-boundary cycles, and whole-witness candidates. Proper subobjects are ranked below whole-boundary and whole-witness candidates, because replacing a global pending object by a proper localized obligation is progress only when the localization data is recorded in Λ. The kind coordinate records the router priority, ordered so that already localized splice/collapse, marker/anchor endpoint, B7 verification, blocked-R3/B6b, dirty-cycle/B4p, and same-complexity pending states appear in increasing order of priority. The size coordinate is the relevant local finite size: boundary length, number of ports or original vertices, blocker graph size, endpoint candidate count, or typed state count. The verification coordinate records whether the finite check is unchecked, checked-routed, checked-terminal, or checked-rejected. Checked-rejected whole-witness candidates are removed from $\mathcal{O}(d)$ rather than kept as active obligations.

The final scale is

$$B_* = B_{\mathrm{base}} \times \mathrm{MSet}_{\mathrm{fin}}(\Delta) \times \mathrm{MSet}_{\mathrm{fin}}(\mathcal{R}_{\mathrm{rep}}) \times \mathrm{MSet}_{\mathrm{fin}}(\mathcal{A}),$$

ordered lexicographically. The final cost is

$$\mathfrak{K}^*(d) = \big(\mathfrak{K}(d), \Lambda(d)\big), \qquad \Lambda(d) = \{\{\lambda(O) : O \in \mathcal{O}(d)\}\}.$$

Lemma 2.1 (Well-foundedness of the Conway scale). The strict order induced by B_* is well-founded.

Proof. The lexicographic order on $\mathbb{N}^{12}$ is well-founded. The local rank sets Δ and $\mathcal{R}_{\mathrm{rep}}$ are finite or finite lexicographic products of natural-number ranks, hence well-founded. The obligation rank set $\mathcal{A}$ is well-founded by construction. The finite-multiset extension of a well-founded order is well-founded, and a finite lexicographic product of well-founded orders is well-founded. Therefore B_* is well-founded.

2.3. Observables versus costs

The following data are observables: the T_3 domain-cover data, the domain-incidence graph, the boundary relation of a carrier, the marker word, the Reeb graph type of a lane, the blocker dependency graph, and the braid word of a train segment. They may select a local analysis or a finite verification, but they are not descents by themselves.

For example, "the boundary relation is nontrivial" is not a proof step. It becomes a proof step only after the typed subcarrier theorem produces a terminal contradiction, a lower-cost description, or a lower-ranked active obligation. Likewise, "an R3 candidate is blocked" is merely an observation until the blocker analysis lowers P, $\mathsf{B}_{\mathrm{blk}}$, U, δ, ϱ, or the active-obligation multiset.

2.4. Cost-controlled local outputs

A local block is admissible only if every output has one of the following forms.

type	cost meaning
T	terminal contradiction; no unresolved description is produced
D	strict descent; an unresolved description d' is produced with $\mathfrak{K}^*(d') < \mathfrak{K}^*(d)$
R	routed subproblem; the output is a named downstream obligation, with no loss of typed data, and either lowers $\mathfrak{K}$ or replaces the active obligation by a finite multiset strictly smaller in the multiset order on Λ
C	finite verification; a finite check whose every row is T, D, R, or N
N	whole-witness non-descent; the attempted object is proved to be the whole active witness, is marked inert, and is never counted as proof progress

Thus an R-output is not a theorem, and a C-output is not a theorem. They are instructions to continue inside a finite, lower-ranked evaluation problem. An N-output is allowed only as a rejection of a whole-witness attempt. It cannot be used to dispose of a proper residual subobject.

In the terminology of presentation complexity, a nonterminal local row is a description-level transfer inside the same presentation system Γ_{Con}: it replaces the current description, or one active obligation inside it, by a new finite description with all typed data needed for the next local analysis. Since the transfer stays inside the same presentation system, the ambient overhead is the identity on B_*; admissibility requires strict descent in that scale. Concretely, a D-row lowers $\mathfrak{K}^*$; an R-row either lowers $\mathfrak{K}$ or replaces the active obligation by a finite multiset of lower λ-rank obligations; a C-row is a finite table whose rows are evaluated in the same sense; and an N-row removes a whole-witness candidate from the active record. Thus every continued nonterminal branch has strictly smaller B_*-cost in the same presentation system Γ_{Con}.

Theorem 2.2 (No cycle of local reductions). Assume each invoked local block is admissible in the sense above. Then there is no infinite chain of productive local evaluations starting from a fixed unresolved description.

Proof. Consider one nonterminal evaluation step after expanding any finite verification table. If it is a D-step, then $\mathfrak{K}^*$ strictly decreases. If it is an R-step and $\mathfrak{K}$ decreases, then $\mathfrak{K}^*$ decreases. If it is an R-step preserving $\mathfrak{K}$, then one active obligation is replaced by a finite multiset of obligations strictly smaller in the multiset order on Λ, so again $\mathfrak{K}^*$ decreases. If it is an N-step, the candidate has been proved to be a whole-witness candidate; it is removed from the active ledger and cannot be reused as progress. This also strictly decreases the Λ coordinate unless no active obligation remains, in which case the description is no longer unresolved. Therefore every continued nonterminal evaluation decreases the well-founded value $\mathfrak{K}^*$. An infinite chain would give an infinite descending sequence in B_*, contradicting Lemma 2.1.

Theorem 2.3 (Minimal-witness principle). Let $\mathcal{U} \subseteq \mathcal{D}_{\mathrm{Con}}$ be the set of unresolved descriptions surviving the external reductions to the irreducible non-T_3 even-dumbbell case. Suppose that every $d \in \mathcal{U}$ admits a closed admissible local evaluation: starting from some active obligation of d, the reduction catalogue produces only T, D, R, C, and N outputs as above, and every R or C output is sent to its named downstream admissible evaluation. Then $\mathcal{U}$ is empty.

Proof. If $\mathcal{U}$ were nonempty, Lemma 2.1 would give a $\mathfrak{K}^*$-minimal element $d_0 \in \mathcal{U}$. Apply the closed admissible evaluation to an active obligation of d_0. A terminal output T resolves the obstruction, contradicting $d_0 \in \mathcal{U}$. A strict descent D produces an unresolved description d' with $\mathfrak{K}^*(d') < \mathfrak{K}^*(d_0)$, contradicting the minimal choice of d_0. A routed or finite-verification output cannot persist forever by Theorem 2.2; its finite downstream evaluation must therefore end in T, D, or N. The first two cases have already been excluded. In the N case the attempted object has been proved to be the whole active witness and is removed as inert; if no active obligation remains, the description is resolved. If another active obligation remains, the description with the inert candidate removed has strictly smaller active-obligation multiset, hence strictly smaller final cost, contradicting the minimality of the original choice. Thus d_0 cannot be unresolved. This contradiction proves $\mathcal{U} = \varnothing$.

Remark 2.4 (Relation with bounded-counterexample arguments). This is the bounded-counterexample method with the bound chosen internally. If a bad irreducible non-T_3 even-dumbbell obstruction exists, then the bad set has an element of minimal B_*-cost. The local catalogue is the exclusion theorem for that minimal bounded part: every possible local failure at that bound is terminal, strictly lower, or inert. The role of the remaining sections is therefore precise. They must prove that the geometric B3, B4, B5, B6, B7, and endpoint blocks are admissible in the sense of this section.

3. Domain-Pair Separation

The first step is purely finite. It extracts the endpoint data used by the later corridor routers, and it does so without invoking any geometric "inner domain" reduction. The output is a B1/B2 setup certificate: a same-cycle pair of ordinary vertices p, q with separated incident-domain pairs, together with the two complementary graph paths between them. This section is not a descent argument and not a corridor construction. It is a finite compiler that adds certified endpoint data to the Conway description.

Let $P(T)$ be the planarization of the even dumbbell drawing T. Its vertices are the original vertices of the dumbbell together with the crossing vertices. Its faces are called **domains**. The common original vertex is denoted v; the remaining original vertices are called **ordinary**. Write $\mathrm{Ord}(T)$ for the set of ordinary original vertices.

For a domain set S and a set X of original vertices, say that S covers X if every $x \in X$ is incident with at least one domain in S. The drawing is T_3 if some set of at most three domains covers all original vertices, including v.

For an ordinary vertex x, let

$$I(x) = \{D_x^-, D_x^+\}$$

be the two domains incident with the two local sectors at x. These two domains are distinct. Indeed, $P(T)$ is a connected Eulerian plane graph: crossing vertices have degree 4, ordinary original vertices have degree 2, and the common vertex has degree 4. A connected Eulerian plane graph has no bridges, so the two local sectors at an ordinary degree-two passage are not the same global face. Its faces admit a two-coloring, and the two local sectors at an ordinary degree-two passage have opposite face colors.

Define the ordinary domain graph $G_D(T)$ as the finite bipartite multigraph whose vertices are the domains of $P(T)$ and whose edges are

$$e_x = D_x^- D_x^+, \qquad x \in \mathrm{Ord}(T).$$

Parallel edges are retained, because different ordinary vertices are different covering constraints. The face two-coloring of $P(T)$ is a bipartition of $G_D(T)$.

Write $\tau(G_D)$ for the minimum vertex-cover size of G_D and $\nu(G_D)$ for its maximum matching size.

Lemma 3.1 (Domain-cover dictionary). A set of domains S covers the ordinary vertices of T if and only if S is a vertex cover of $G_D(T)$. Consequently the ordinary vertices are coverable by at most r domains if and only if the vertex-cover number $\tau(G_D(T))$ is at most r.

Proof. The edge e_x of $G_D(T)$ has endpoints exactly the two domains in $I(x)$. Thus S covers the ordinary vertex x exactly when S contains at least one endpoint of e_x. This is precisely the vertex-cover condition, applied to every ordinary vertex. Parallel edges cause no ambiguity: they represent distinct ordinary vertices with the same incident-domain pair, and a domain set covers each of them exactly when it covers the corresponding parallel edge.

A **B1 domain certificate** is one of the following finite objects:

certificate	finite data
ordinary-domain certificate	a matching of size 4 in $G_D(T)$
v-domain certificate	for every $d \in I_T(v)$, a matching of size 3 in $G_D(T) - d$

Here $I_T(v)$ is the finite set of domains incident with v, and $G_D(T) - d$ is obtained by deleting the domain vertex d and all ordinary-domain edges incident with it. The certificate stores the displayed matching or matchings as finite data; it does not merely assert that they exist.

The certificate proves $T \notin T_3$. In the ordinary-domain case, a matching of size 4 prevents any three-domain cover of the ordinary vertices by Koenig's theorem. In the v-domain case, any three-domain cover of all original vertices would contain some $d \in I_T(v)$; deleting d would leave a two-domain cover of $G_D(T) - d$, contradicting the stored size-three matching.

Theorem 3.2 (B1 domain certificate and separated endpoints). Let T be a non-T_3 even dumbbell drawing with common vertex v. Then T admits a B1 domain certificate. Moreover the certificate supplies same-cycle ordinary vertices with separated incident-domain pairs:

1. If the ordinary vertices are not coverable by three domains, then $G_D(T)$ has a matching of size at least 4. Among the four corresponding ordinary vertices, two lie on the same original cycle. For those vertices p, q, $I(p) \cap I(q) = \varnothing$.
2. If the ordinary vertices are coverable by three domains, then the obstruction to being T_3 is at v. For every $d \in I_T(v)$, the graph $G_D(T) - d$ has a matching of size at least 3. Among the three corresponding ordinary vertices, two lie on the same original cycle. For those vertices p_d, q_d, $I(p_d) \cap I(q_d) = \varnothing$ and $d \notin I(p_d) \cup I(q_d)$.

Proof. By Lemma 3.1 and Koenig's theorem for finite bipartite multigraphs, the minimum number of domains covering the ordinary vertices is the matching number of $G_D(T)$:

$$\tau(G_D(T)) = \nu(G_D(T)).$$

Assume first that the ordinary vertices are not coverable by three domains. Then $\tau(G_D(T)) \geq 4$, hence $G_D(T)$ has a matching of size at least 4. The four matched edges correspond to four distinct ordinary vertices whose incident-domain pairs are pairwise disjoint. These vertices lie on the two original cycles of the dumbbell, so two of them lie on the same cycle. Call them p, q. Since their matching edges are disjoint, $I(p) \cap I(q) = \varnothing$.

Assume now that the ordinary vertices are coverable by three domains. Since T is not T_3, no such three-domain cover can also cover v. The ordinary vertices are not coverable by two domains: if D_1, D_2 covered all ordinary vertices, then D_1, D_2 together with any domain $d \in I_T(v)$ would cover all original vertices, contradicting $T \notin T_3$. Hence $\tau(G_D(T)) = 3$.

Fix $d \in I_T(v)$. If $G_D(T) - d$ had a vertex cover C of size at most 2, then $C \cup \{d\}$ would cover every ordinary vertex: edges incident with d are covered by d, and all remaining edges are covered by C. It would also cover v, because d is incident with v. This would make T a T_3 drawing, contradiction. Therefore $\tau(G_D(T) - d) \geq 3$, and Koenig's theorem gives a matching of size at least 3 in $G_D(T) - d$.

The three matched ordinary vertices again have pairwise disjoint incident-domain pairs, and none of those pairs contains d. Among three vertices on two original cycles, two lie on the same cycle. These are p_d, q_d, and they satisfy both displayed conditions.

Lemma 3.3 (B2 complementary paths). Let p, q be any same-cycle pair supplied by Theorem 3.2. Then the original cycle containing p, q contains exactly two complementary simple graph paths γ_0, γ_1 from p to q. Their union is that cycle and their intersection is $\{p, q\}$. The endpoint data inherited from B1 is

$$I(p) \cap I(q) = \varnothing,$$

and, in the v-certificate branch, the chosen v-domain d is absent from both endpoint pairs.

Proof. A simple cycle with two distinct vertices p, q becomes two open path components after deleting p and q. Restoring the endpoints gives the two complementary simple p-to-q graph paths. The endpoint-domain statements are exactly the matching disjointness statements proved in Theorem 3.2. No regular-neighborhood or disk claim is being made here; the drawn image of either path may self-interact after planarization.

Remark 3.4 (Complexity role of B1/B2). The domain graph, its matchings, and the T_3 cover tests are observables of the planarized drawing. In the presentation-complexity language, B1/B2 are finite setup compilers: they add certified endpoint data to the initial obstruction package. They are not descent steps and do not construct a switch corridor. The images of γ_0, γ_1 may self-interact after planarization; the passage from these paths to an anchored carrier is the B3 problem of the next section.

4. Anchored Carriers and the B3 Handoff

B1/B2 supply a same-cycle pair p, q and two complementary graph paths γ_0, γ_1. The role of B3 is to turn one of these finite p-to-q traces into either an anchored corridor for B4 or a named lower router. The key correction from the audit is that minimization must preserve the p, q anchors; a smaller local corridor that forgets the original endpoints is not a substitute for the B1/B2 problem.

Fix one of the two complementary paths and call it γ. Traversing γ in the drawing gives a finite planarized walk $\Gamma_\gamma \subset P(T)$. A **terminal section** at p or q is a small transverse interval meeting the trace in the corresponding endpoint germ and carrying the incident-domain pair, side, orientation, endpoint-role, marker, anchor, and replacement labels.

A **typed state** is the full label read on such a section or on an intermediate slicing level:

$$\sigma = (\mathbf{side}, \mathbf{orient}, \mathbf{dom}, \mathbf{supp}, \mathbf{gate}, \mathbf{end}, \mathbf{mark}, \mathbf{anch}, \mathbf{rep}).$$

Only full typed states may be compared. In particular, two geometrically equal sections are not identical unless their marker, anchor, endpoint, and replacement data also agree.

Lemma 4.1 (Finite raw switch trace). The planarized walk Γ_γ carries a finite nonconstant lateral-domain sequence from the endpoint data at p to the endpoint data at q.

Proof. The path γ has finitely many graph edges, and each edge is cut by finitely many crossings in the planarization. Hence the planarized walk has finitely many edgelets. Each edgelet has two lateral domains, so the walk determines a finite sequence of lateral-domain pairs. The initial pair is $I(p)$ and the terminal pair is $I(q)$. By B1, $I(p) \cap I(q) = \varnothing$, so the reduced lateral sequence cannot be constant.

4.1. Carriers and Reachability

A **typed anchored carrier** for γ is a compact PL disk C equipped with:

1. terminal sections Σ_p, Σ_q identified with the original p, q endpoint sections;
2. a finite planarized trace inside C carrying the relevant part of Γ_γ;
3. the full typed labels on boundary states, slicing states, marker passages, anchor data, and replacement obligations;
4. a full typed boundary relation $\mathcal{R}_C \subseteq \Sigma_p \times \Sigma_q$ recording which terminal states are connected through the complement of the trace with compatible labels.

The carrier is **anchored** because the terminal sections are part of the data. An operation on C is allowed inside B3 only if it preserves these sections and their full typed endpoint relation, or else exports the failure as an active obligation.

The following finite reachability description is the safe replacement for the old global composition assertion.

Lemma 4.2 (Layered reachability graph). Let C be a typed anchored carrier with a generic PL height function $h : C \to [0, 1]$ whose level 0 contains Σ_p and whose level 1 contains Σ_q. Choose finitely many regular levels separating all critical heights of the refined planar graph. Build a finite graph $\mathcal{L}(C)$ whose vertices are:

1. typed states on the chosen levels; and
2. relative-domain components inside the strips between consecutive levels.

Join a level-state vertex to a strip-component vertex when the state lies in the closure of the component with matching type. Then two terminal states are related by $\mathcal{R}_C$ if and only if they lie in the same connected component of $\mathcal{L}(C)$.

Proof. Every path in $C \setminus P(T)$ from one terminal state to another crosses a finite sequence of strips and levels. Recording the component of each strip and the state met on each level gives a path in $\mathcal{L}(C)$. Conversely, a path in $\mathcal{L}(C)$ is a finite chain of incidences between level states and strip components; inside each strip component, the two incident states are connected by definition. Concatenating these local connections gives a path in $C \setminus P(T)$ with the same full typed labels. Thus the boundary relation is exactly finite graph reachability.

The reachability graph is not yet an ordered corridor. It allows arbitrary backtracking. The next lemma is the PL mechanism that detects such backtracking and turns it into a routed subobject.

Lemma 4.3 (PL backtracking subdisk). Let Λ be a relative-domain component participating in an essential boundary relation of a typed anchored carrier C. If the essential Reeb graph of $h|_\Lambda$ is not a height-monotone interval from Σ_p to Σ_q, then C contains a compact typed routed subdisk D whose boundary is made of level segments, lane-continuation arcs, carrier-boundary arcs, and support or gate arcs. The subdisk has one of the following types:

1. full typed identity or pure-ear data;
2. same-relation splice data;
3. smaller anchored carrier or smaller support-gate loop;
4. marker or common-vertex data;
5. ordinary-original-vertex ported disk data;
6. B7 replacement, blocked-R3, or certificate data;

7. whole-witness return.

Proof. Put C, $P(T) \cap C$, and h in generic PL position. After subdividing at all relevant regular levels, the union of the trace, the carrier boundary, and the chosen levels is a finite plane graph.

If the essential Reeb graph is not a monotone interval, then one of the following finite events occurs: an essential branch terminates away from the opposite terminal section, a lane meets the same regular level twice, two essential continuations merge or split, an interval branch has an interior height extremum, or the Reeb graph contains a cycle. In each case choose the first such event in the sweep order and then choose an innermost return.

Equivalently, there is an essential arc $\alpha \subset \overline{\Lambda}$ whose endpoints lie on the same regular level $h^{-1}(t)$ and whose interior lies on one side of that level. Let β be the level segment between the endpoints, subdivided at its intersections with the trace. The closed walk contained in $\alpha \cup \beta$ and the refined plane graph contains an embedded innermost simple cycle. Let D be the disk bounded by that cycle.

The boundary of D consists only of the listed typed pieces: level segments, lane arcs, carrier boundary, and support/gate arcs. Its interior classification is exhaustive. If it contains no graph, marker, original-vertex, or replacement data, it is identity, pure-ear, same-relation, or smaller-carrier data according to its boundary relation. If it contains v or marker transitions, it is B5/B6c data. If it contains ordinary original vertices, it is B6 ported-disk data. If it contains replacement or R3 ledger data, it is B7 data. If the return is the whole active witness rather than a proper subdisk, it is marked as a whole-witness non-descent.

4.2. Reduction and No-Backtracking

A carrier is **B3-reduced** if every compact typed subdisk produced by Lemma 4.3 has already been evaluated in the sense of Section 2: full typed identity pieces are deleted, same-relation pieces are spliced, smaller anchored carriers or support-gate loops are installed with lower cost, proper marker/original/replacement pieces are exported as active B5/B6/B7 obligations, and whole-witness returns are marked inert.

Theorem 4.4 (Complexity-controlled no backtracking). Let d be a minimal unresolved description with respect to $\mathfrak{K}^*$, and suppose its active B3 carrier is B3-reduced. Then every essential lane in the carrier has height-monotone essential Reeb graph.

Proof. Suppose an essential lane is not height-monotone. Lemma 4.3 gives a compact typed routed subdisk D.

If D has full typed identity data, deleting it preserves the terminal typed relation and lowers one of the carrier coordinates R, H, S, E or the tie-breaker U. If D has same-relation splice data, the splice again preserves the full anchored boundary relation and lowers the same carrier part of $\mathfrak{K}^*$. If D is a smaller anchored carrier or smaller support-gate loop, replacing the active object lowers the corresponding carrier or obligation rank. Each of these contradicts minimality.

If D contains marker, common-vertex, ordinary-original-vertex, replacement, blocked-R3, or certificate data, then D is a proper B5/B6/B7 obligation. That contradicts the assumption that the carrier was B3-reduced unless the obligation has already been exported, in which case B3 has produced a router output rather than an unresolved carrier. If the return is a whole-witness non-descent, it is removed from the active ledger and cannot persist in a minimal unresolved B3 carrier. Therefore no non-height-monotone essential lane remains.

Theorem 4.5 (Reachability-to-corridor normal form). Let C be a B3-reduced typed anchored carrier in which every essential lane is height-monotone. Then the essential part of $\mathcal{L}(C)$ is equivalent, with the same full typed boundary relation, to an ordered typed corridor

$$\mathcal{C} = (\Sigma_0, Q_1, \Sigma_1, \dots, Q_N, \Sigma_N; \tau, R_1, \dots, R_N),$$

where $\Sigma_0 = \Sigma_p$, $\Sigma_N = \Sigma_q$, the Q_i are the strips between consecutive regular levels, τ records the full state type, and R_i is the local typed relation in Q_i.

Proof. By height-monotonicity, an essential lane meets each chosen regular level in the sweep order and does not return to an earlier level. Therefore every essential boundary connection determines a chain

$$\Sigma_0 \sim Q_1 \sim \Sigma_1 \sim \cdots \sim Q_N \sim \Sigma_N$$

in the layered reachability graph. Reading the strip components in order gives a compatible sequence of local relations $R_1, \dots, R_N$.

Conversely, any compatible ordered chain of local strip relations is a path in $\mathcal{L}(C)$, hence gives a connection in the carrier by Lemma 4.2. Components of $\mathcal{L}(C)$ not meeting both terminal sections either do not affect the boundary relation, have already been deleted/spliced in the B3-reduced carrier, or have been exported to a downstream router. Thus the essential typed boundary relation of C is exactly the ordered composition of the strip relations, and the displayed data form the required ordered typed corridor.

4.3. The Handoff Router

Theorem 4.6 (B3-to-B4 handoff router). Let d be a minimal unresolved description after B1/B2 have supplied p, q and γ_0, γ_1. Then B3 has one of the following admissible outputs:

1. T: a certified terminal contradiction or certified local terminal move;
2. D: an unresolved description d' with $\mathfrak{K}^*(d') < \mathfrak{K}^*(d)$;
3. R: an explicit active B5/B6/B7 or endpoint obligation, with full typed data preserved;
4. R_{B4}: an ordered p, q-anchored typed corridor $\mathcal{C}$ satisfying Theorem 4.5, recorded as the named B4p obligation.

Proof. Start with the finite raw trace Γ_γ from Lemma 4.1. If its image graph has no relevant cycle, a small regular neighborhood gives an anchored carrier whose terminal sections are the original p, q sections. If the image graph has a repeated vertex, repeated state, or relevant cycle, choose an innermost such cycle in the finite plane graph. The disk it bounds is typed. If it is full identity or same-relation data, deleting or splicing it lowers the carrier part of the cost. If it carries marker, ordinary-original-vertex, replacement, blocked-R3, endpoint, or whole-witness data, it is exported to the corresponding active obligation or marked inert. Since the trace is finite and every identity deletion lowers finite carrier size, this process terminates. Its output is either T, D, R, or a typed anchored carrier to be minimized in the next paragraph.

In the anchored-carrier branch, minimize only among carriers preserving the terminal sections Σ_p, Σ_q and the full typed endpoint relation, including marker, anchor, endpoint, and replacement ledgers. This anchored restriction is part of the proof: a smaller local subcorridor that forgets p, q is not an allowed replacement.

If anchored minimization finds a preserving deletion, splice, or smaller carrier, the output is D. If it finds a proper marker, original-vertex, replacement, blocked-R3, or endpoint object, the output is R. In the residual case the carrier is B3-reduced. The no-backtracking theorem then makes all essential lanes height-monotone, and Theorem 4.5

compiles the finite reachability graph to an ordered typed corridor. This is the R_{B4} output: the active B3 obligation has been replaced by the named B4p obligation with the same p, q anchors and full typed boundary relation.

Remark 4.7 (Status of B3). B3 closes the anchored-carrier stage as an admissible router. Its genuine mathematical content is that the proof no longer uses an arbitrary minimal local corridor and no longer uses ordered composition before no-backtracking has been proved. The downstream obligations exported by B3 are the B5/B6/B7 and endpoint routers.

5. Typed Corridor Regularization

B4 is the normal-form compiler for the ordered corridors produced by B3. It is invoked only after Theorem 4.6 has supplied a p, q-anchored ordered typed corridor

$$\mathcal{C} = (\Sigma_0, Q_1, \Sigma_1, \ldots, Q_N, \Sigma_N; \tau, R_1, \ldots, R_N),$$

with $\Sigma_0 = \Sigma_p$, $\Sigma_N = \Sigma_q$, full state type τ, and ordered relation

$$R_{\mathcal{C}} = R_N \circ \cdots \circ R_1.$$

This ordered composition is not assumed for raw disks. It is the output of the B3 no-backtracking and reachability compiler. The role of B4 is to replace such an ordered corridor by a typed regular corridor, or else to output a certified terminal move, a strict descent in $\mathfrak{K}^*$, or a named downstream obligation.

A **critical strip** is a connected component of some Q_i after deleting monotone bands whose full typed boundary relation is identity or same-relation. The strip is **essential** if it changes the support, side, gate order, endpoint role, marker record, or replacement ledger of an essential lane. A corridor is **B4-reduced** if every inessential monotone band, proper identity ear, same-relation splice, and proper same-boundary subcorridor has already been evaluated as a terminal local move, a strict descent, or a named active obligation in the ledger Λ.

A **typed regular alternating cell** is an embedded H_{2m} cell, with $m \geq 2$, equipped with stable typed input and output gates, one-to-one gate continuation, no returning lane, no branching or merging lane, no repeated gate identification, and no hidden marker, original-vertex, endpoint, replacement, identity, or same-relation obligation in its interior.

Thus the square-switch H_4 is a binary regular cell. The two-gate case H_2 is not regular; it must be identity, same-side, lower-cost, or routed.

A corridor is **typed regular** if every essential critical strip is a typed regular alternating cell, distinct cells meet only along full typed gates, no three cells share a gate, terminal gates lie on endpoint sections, and every non-B4 datum has been exported to B5, B6, endpoint discharge, or B7.

Lemma 5.1 (Critical-strip normal form). Let $\mathcal{C}$ be an ordered typed corridor supplied by Theorem 4.6. Then its essential part admits a finite critical-strip decomposition. In a B4-reduced minimal package, every component outside the essential critical strips has already produced a typed identity deletion, same-relation splice, strict descent, or named downstream obligation.

Proof. The corridor has finitely many strips Q_i, and each strip contains finitely many planarized edgelets, gate states, and critical heights. Subdividing at these heights gives finitely many monotone components. A component whose full typed boundary relation is identity deletes without changing the ambient typed relation and lowers the carrier coordinates H, S, E or the tie-breaker U. A component with the same typed relation as the adjacent boundary data splices and lowers the same finite part of $\mathfrak{K}^*$. A component whose typed relation is nontrivial but proper is not discarded; by the definition of B4-reduced it has already been promoted to a lower active carrier or exported as an active B5/B6/B7 or endpoint obligation. What remains is exactly the finite family of essential critical strips.

Call an essential critical strip **bad** if it is nonembedded, nonalternating without an extracted genuine typed lens, an H_2 strip, gate-nonregular, or cell-intersection-nonregular. Under the revised convention, a square-switch H_4 satisfying the regularity hypotheses is not bad; it is the $m = 2$ regular cell.

Lemma 5.2 (Proper bad-strip router). Let K be a bad critical strip inside a B4-reduced minimal ordered typed corridor. If K is contained in a proper typed subcorridor, then B4 produces one of the following outputs:

1. a typed identity deletion or same-relation splice;
2. a witness package d' with $\mathfrak{K}^*(d') < \mathfrak{K}^*(d)$;
3. an explicit B5/B6/B7 or endpoint obligation with full typed data preserved.

Proof. Choose the smallest consecutive block of strips containing K and all incident gates needed to read its full typed boundary relation. Since the block is proper, its carrier size, state count, height-event count, or finite tie-breaker is strictly smaller than that of the active corridor, unless its relation is empty.

There are three relation cases. If the full typed relation is identity, delete the block; the terminal sections and their labels are unchanged, and H, S, E or U decreases. If the block has the same typed boundary problem as the active corridor, promote it to the active carrier of a new witness package; no earlier coordinate increases, while a carrier coordinate strictly decreases. If the block has different typed boundary data, it is not a contradiction and not a free descent. Its first non-B4 feature is finite: marker or common-vertex data gives B5, ordinary original-vertex data gives B6, endpoint data gives endpoint discharge, and replacement, blocked-R3, or certified edge-removal data gives B7. These alternatives exhaust a proper finite typed block.

Lemma 5.3 (Global monochromatic corridor router). Let $\mathcal{C}$ be a B4-reduced minimal ordered typed corridor with no pending B5/B6/B7 or endpoint obligation. Suppose a nonalternating bad strip is coextensive with the whole active corridor and contains no v, marker data, ordinary original-vertex data, endpoint object, replacement data, or blocked-R3 data. Then it routes to typed identity/splice, a same-side gate-path output, a closed-train output, or a strict descent.

Proof. After the listed data have been excluded, a monochromatic run is made only of typed support and gate information. If the run meets an endpoint section, then following its same-color continuation cannot encounter a proper lens or proper bad strip; otherwise Lemma 5.2 would apply. Therefore the continuation returns to the same endpoint side before producing a cross-endpoint alternating ladder, or else it closes. The former is the same-side gate-path configuration of Lemma 5.5 below. The latter is a closed train or support-cycle. If the run meets no endpoint section, finiteness and typed gate continuation imply that it is either relation-empty, hence deletable, or a closed same-color component, hence a train/support-cycle. In every case the output is one of the listed router outputs.

Lemma 5.4 (Short whole-corridor strips). In a B4-reduced minimal no-pending package, no residual cell-clean whole-corridor H_2 remains. A residual cell-clean square-switch H_4 satisfying stable gates, one-to-one cross-endpoint continuation, no repeated support, and no hidden marker/original/replacement data is a typed regular binary cell.

Proof. An H_2 strip has one essential channel between two typed boundary gates. If the two gates lie on the same endpoint side, it is a same-side gate-path output. If they lie on opposite endpoint sides, the channel is a monotone typed band. Its full relation is identity, same-relation, or different endpoint data. The first two cases delete, splice, or lower the active carrier; the third is endpoint or downstream router data, contradicting the no-pending residual hypotheses.

For H_4, first remove the nonregular alternatives. Same-side external gates are handled by the same-side gate-path router. Repeated support is handled by the train/support-shift router. Non-one-to-one gate pairing is a typed normality defect. Identity or same-relation boundary data deletes, splices, or lowers $\mathfrak{K}^*$. Hidden marker, ordinary original-vertex, endpoint, replacement, or blocked-R3 data is exported. The only remaining case is the square-switch with four stable typed gates, two on each endpoint side, one-to-one cross-endpoint continuation, no repeated support, and nontrivial alternating square relation. This is precisely an embedded H_{2m} cell with $m = 2$, so it is regular under the revised definition.

Lemma 5.5 (Typed same-side gate-path router). Let $\mathcal{C}$ be a B4-reduced typed regular corridor. If an essential gate-graph path has both external endpoints on the same endpoint section, then B4 produces a typed identity deletion, same-relation splice, strict descent, closed-train output, or B5/B6/B7 or endpoint obligation. Consequently, in a minimal no-pending package, no essential same-side path component remains.

Proof. If the gate path contains a cycle, that cycle is a closed alternating train. Otherwise the path, together with the interval of the endpoint section between its two external gates, forms a closed walk in the planarized carrier. Choose an innermost simple cycle in this closed walk and call its disk D. The disk cannot contain the opposite endpoint section; otherwise the gate path would separate the two terminal sections and would have to carry a cross-corridor continuation, contradicting that both external endpoints lie on the same endpoint section.

Thus D is a proper routed subdisk. If its full typed relation is identity, delete it. If it has the same typed boundary problem, promote it to a lower-cost witness package. If it has same-relation boundary data, splice it. If it contains v or marker data, route to B5. If it contains ordinary original vertices, route to B6. If it contains endpoint data, route to endpoint discharge. If it contains replacement or blocked-R3 data, route to B7. If the part inside D is a smaller closed train, lower the train coordinate L. These are exactly the allowed outputs.

For a typed regular corridor, form the finite gate graph whose vertices are stable gates and whose edges are typed cell continuations. An essential component is one that participates in the cross-corridor relation.

Proposition 5.6 (Gate-graph ladder/train dichotomy). In a B4-reduced minimal typed regular corridor with no pending routed obligation, every essential gate-graph component is either a cross-endpoint local ladder or a closed alternating train.

Proof. In a typed regular cell, every internal gate has one incoming and one outgoing continuation, so the essential gate graph has internal degree 2. A finite component with a cycle contains a closed alternating train. If an essential component is acyclic, it is a path with external endpoints on endpoint sections. Lemma 5.5 removes the case in which both external endpoints lie on the same endpoint section. Therefore every remaining acyclic essential component connects opposite endpoint sections and is a local ladder.

Proposition 5.7 (Anchored complementary ladder gluing). Suppose B3 and B4 applied to the two complementary p-to-q paths produce anchored typed regular corridors $\mathcal{C}_0, \mathcal{C}_1$, and suppose each contains a local ladder L_0, L_1. Then $L_0 \cup L_1$ closes to a closed alternating train, unless an endpoint-local typed deletion, same-relation splice, strict descent, endpoint discharge, or B6/B7 obligation is produced at p or q.

Proof. Let x be one of the ordinary endpoints p, q. A small disk about x meets exactly the two incident edge germs of the original cycle. Its complement has two sectors, and these sectors are the two domains in $I(x)$. Since the two graph paths are complementary arcs of the same original cycle, the two ladder ends at x approach through the two complementary sides of that cycle.

If the typed endpoint data match these two sectors, the continuation through x is forced. If they do not match, the mismatch is finite and local. A pure typed ear deletes, a same-relation endpoint splice lowers the carrier, a gate identification or branching defect gives a strict descent or typed splice, ordinary original-vertex data gives B6, endpoint data gives endpoint discharge, and replacement or blocked-R3 data gives B7. Since $p, q \neq v$, no new B5 marker is created at the endpoint itself; if the failure disk reaches v, it is no longer endpoint-local and is routed to B5/B6.

After both endpoint pairings are made, every gate in $L_0 \cup L_1$ has degree 2: internal gates by regularity, endpoint gates by the forced two-sector pairing. Therefore the union contains a cyclic gate-graph component, namely a closed alternating train.

Theorem 5.8 (B4p typed corridor regularization router). Let d be a minimal unresolved description whose active B3 output is an ordered p, q-anchored typed corridor $\mathcal{C}$. Then B4 has one of the following admissible outputs:

1. R_{reg}: $\mathcal{C}$ is typed regular, and its essential gate-graph components are local ladders or closed alternating trains as in Proposition 5.6; this replaces the active B4 obligation by the named ladder/train support-gate obligations;
2. T or D: there is a certified terminal/local move or an unresolved description d' with $\mathfrak{K}^*(d') < \mathfrak{K}^*(d)$;
3. R: there is an explicit B5/B6/B7 or endpoint obligation with full typed data preserved;
4. N: an attempted whole-corridor or whole-witness object is rejected as an inert non-descent and removed from the active ledger.

Proof. Start with the ordered typed corridor supplied by Theorem 4.6, so the relation $R_{\mathcal{C}} = R_N \circ \cdots \circ R_1$ is meaningful. Apply Lemma 5.1 to obtain the finite critical-strip decomposition.

If a bad critical strip is proper, Lemma 5.2 gives a terminal local move, a strict descent, or a downstream routed obligation. Hence no proper bad strip survives in the residual minimal no-pending branch. If a nonalternating bad strip is whole-corridor, first export any hidden v, marker, ordinary original-vertex, endpoint, replacement, or blocked-R3 data. In the remaining pure case Lemma 5.3 routes the global monochromatic corridor to identity/splice, same-side gate path, train, or strict descent. If the whole-corridor bad strip is H_2 or H_4, Lemma 5.4 removes H_2 and promotes the residual square-switch H_4 to a regular binary cell.

Gate nonuniqueness, returning lanes, branching or merging, same-side external gate paths, nonregular cell intersections, and triple gate sharing are handled by Lemma 5.5 and by the same proper-subdisk classification used above. Once these outputs are excluded by minimality and no-pending hypotheses, every essential critical strip is an embedded typed alternating cell H_{2m} with $m \geq 2$, gate continuation is one-to-one, cells meet only along typed gates, and no hidden non-B4 datum remains. Thus $\mathcal{C}$ is typed regular. Proposition 5.6 gives the ladder/train dichotomy for its essential gate graph.

All outputs are admissible in the sense of Section 2: terminal/local certified operations resolve the active obligation, strict descents lower $\mathfrak{K}^*$, routed objects enter B5/B6/B7 or endpoint discharge with full type preserved, regular ladders/trains replace the B4 obligation by lower-ranked named support-gate obligations, and whole-witness rejections are removed from Λ as inert non-progress. Therefore B4 is closed as an admissible router.

Remark 5.9 (Complexity role of B4). B4 is a normal-form compiler inside the bounded-counterexample method. It says that a minimal unresolved ordered corridor cannot keep arbitrary local irregularity: every irregularity is either visible as a finite typed certificate, lowers the presentation cost, or becomes a named lower router. The downstream dependencies are explicit: B3 must supply the ordered corridor, and B5/B6/B7 plus endpoint discharge must evaluate the exported obligations.

6. Common-Vertex Parity and Marker Ledger

Let the two even cycles be $A = C_{2a}$ and $B = C_{2b}$, with common vertex v. Throughout this section we use the standard topological thrackle convention: adjacent edges meet exactly at their common endpoint and do not also cross elsewhere, while nonadjacent edges meet exactly once in a transverse crossing. The purpose of B5 is local and typed. It excludes the separated rank-zero model at v, defines the surviving rank-one marker calculus, and states exactly which marker operations are legal. It is not a terminal proof engine for all marker-containing support-gate objects; those objects are routed to endpoint discharge, B6, or B7 unless a named marker cancellation or descent is supplied.

Lemma 6.1 (Rotation parity at the common vertex). Around a sufficiently small circle about v, the two A-half-edges and the two B-half-edges do not alternate. Equivalently, the local cyclic order A, B, A, B is impossible.

Proof. Choose a closed disk D around v so small that D meets only the four incident edge germs and contains no other vertex or crossing. Delete the interior of D. The cycle A becomes a properly embedded arc A' with endpoints on ∂D, and B becomes a properly embedded arc B' with endpoints on ∂D.

For two properly embedded arcs in a disk with boundary, the mod-2 intersection number is determined by the boundary order of their endpoints: alternating endpoints give odd intersection number, and nonalternating endpoints give even intersection number. This follows by closing one arc along either boundary interval between its endpoints and applying the Jordan curve parity test to the other arc.

Now count intersections of A' and B' in the thrackle drawing. There are $(2a)(2b)$ pairs consisting of one edge of A and one edge of B. Exactly four pairs use an edge of A incident with v and an edge of B incident with v; these pairs are adjacent at v and, by the standard convention, meet only at v, hence contribute no intersection outside D. Every other such edge pair is nonadjacent and crosses exactly once outside D. Therefore

$$|A' \cap B'| \equiv (2a)(2b) - 4 = 4(ab-1) \equiv 0 \pmod{2}.$$

Thus A' and B' meet evenly, so their endpoints on ∂D are not alternating. The order A, B, A, B would make them alternate, contradiction.

Definition 6.2 (Rank-one sector model). After Lemma 6.1, the four half-edges may be cyclically ordered, up to reversal and exchanging A, B, as

$$A_1, A_2, B_1, B_2.$$

Let

$$S_A = (A_1, A_2), \qquad S_1 = (A_2, B_1), \qquad S_B = (B_1, B_2), \qquad S_2 = (B_2, A_1).$$

The same-cycle local passages are ordinary continuations

$$c_A : S_A \to S_A, \qquad c_B : S_B \to S_B.$$

The mixed passages through v are singular rank-one transitions and are recorded as markers

$$m_A : S_2 \Rightarrow S_1, \qquad m_B : S_1 \Rightarrow S_2.$$

The double arrow means that the passage is not an ordinary v-free support-switch continuation. It is a typed event at the common vertex and must remain visible in the marker ledger.

A local boundary state near v has the form

$$\sigma = (\ell, s, \epsilon, \rho, \mu),$$

where ℓ records the local support, gate, lane, or sector; s records side or endpoint role; ϵ records orientation and coorientation data; ρ records the transported domain/support relation; and μ is the marker word in the alphabet $\{m_A, m_B\}$, with any endpoint-orientation refinement needed for typed equality. Equality of typed states includes equality of μ.

Lemma 6.3 (Marker persistence). Ordinary v-free support-switch normalization, ordinary lane continuation, full typed identity deletion, and same-relation splice preserve the marker record. A transition through v cannot be replaced by an ordinary continuation unless a named marker-discharge certificate is supplied.

Proof. Ordinary v-free moves are supported away from the common vertex, so they do not create, destroy, or alter singular passages through v. Ordinary lane continuations are arrows inside the typed state space and therefore preserve the marker coordinate.

A full typed identity deletion is legal only when the entering and exiting typed boundary states agree. Since typed equality includes μ, such a deletion cannot remove an occurrence of m_A or m_B. A same-relation splice has the same restriction: the two-terminal typed boundary relation must agree, including marker data. Thus a splice that changes a nonempty marker word into the empty word is not a same-relation splice; it is an additional marker-discharge theorem and must be stated with its own hypotheses.

Consequently, the marker word can change only by a named typed marker cancellation, by a rank-one descent whose decreasing coordinate includes M, by endpoint discharge lowering the marker ledger, or by a certified replacement/edge-removal theorem whose certificate explicitly includes the marked sector data.

Lemma 6.4 (Marker normal form). In a minimal reduced witness package, every non-discharged local marker endpoint object has primitive word $m_A m_B$ or $m_B m_A$.

Proof. A closed marker word alternates because m_A maps S_2 to S_1 and m_B maps S_1 to S_2. After choosing the initial mixed sector, it is of the form $(m_A m_B)^r$ or $(m_B m_A)^r$, with $r \geq 1$.

If $r > 1$, the first return to the starting mixed sector cuts off a proper closed marker subpackage with shorter marker word. By Lemma 6.3, this subpackage cannot be erased by ordinary moves. In a minimal reduced witness, it must therefore be a typed same-relation marker cancellation, a lower marker-complexity witness, a certified B7 operation, or a routed endpoint/B6/B7 object. If none occurs, the original package was not minimal in the marker coordinate M. Hence the only local marker endpoint residual has $r = 1$.

Lemma 6.5 (Primitive marker cap-relation table). Suppose a primitive marker pair is consecutive and cap-eligible: the two marker transitions lie in one normalized v-

neighborhood, no ordinary carrier segment lies between them, the mixed-sector cap between the marked detour and the short unmarked sector arc contains no graph, endpoint, anchor, or replacement data, and the two terminal states have been normalized. Then the pair cancels as a typed marker-cap cancellation and lowers M.

Proof. Let T_i^-, T_i^+ be the two terminal typed states on a mixed sector S_i, and let $u_i : T_i^- \to T_i^+$ be the short unmarked sector arc. The two possible caps are

marked detour	short arc	relation after deleting markers
$m_B \circ m_A : T_2^- \to T_2^+$	$u_2 : T_2^- \to T_2^+$	u_2
$m_A \circ m_B : T_1^- \to T_1^+$	$u_1 : T_1^- \to T_1^+$	u_1

In each row, the marked detour and the corresponding short arc are the two boundary arcs of a disk contained in the normalized v-neighborhood. By cap eligibility, the disk interior contains no graph data, no ordinary carrier segment, no endpoint/anchor mismatch, and no replacement boundary. The two arcs have the same terminal typed states by construction. All non-marker labels in the boundary relation are read at those terminal states and along an empty sector cap, so the only difference between the two boundary relations is the explicit primitive marker word. Removing that word by the named marker-cap operation leaves exactly the short-sector relation u_i. This strictly lowers the marker coordinate M and does not increase any earlier coordinate of $\mathfrak{K}^*$.

If any alleged cap row differs after deleting the marker word, then at least one cap-eligibility hypothesis was false. The failure routes as follows.

source of failure	output
terminal states do not match	endpoint/anchor defect, lower A or endpoint discharge
ordinary carrier segment lies between markers	separated marker object, routed to endpoint/B6/B7
hidden graph or ordinary original-vertex data lies in the cap	B6 routed object
support, side, or gate label is unstable	B3/B4 endpoint router or strict descent

source of failure	output
replacement or blocked-R3 boundary is involved	B7

There is therefore no additional purely local "primitive cap-relation mismatch" residual.

Theorem 6.6 (B5 marker router). Let a minimal unresolved description contain an active common-vertex marker object. Then B5 has one of the following admissible outputs:

1. R: a same-cycle ordinary continuation c_A or c_B with marker record preserved, replacing the marker obligation by ordinary typed continuation data;
2. D: a typed marker-cap cancellation lowering M;
3. D: a strict descent in $\mathfrak{K}^*$;
4. R: an endpoint, B6, or B7 obligation with full marker, endpoint, anchor, and replacement data preserved;
5. N: a whole-witness marker return rejected as inert non-progress.

In particular, B5 never licenses replacing m_A or m_B by an ordinary support-switch move without one of these outputs.

Proof. Lemma 6.1 excludes the alternating rank-zero local order. Definition 6.2 gives the surviving local sector model: same-cycle passages are the ordinary continuations c_A, c_B, while mixed passages are the marked arrows m_A, m_B. Lemma 6.3 makes the marker word invariant under all ordinary v-free operations, so any legal removal of a marker must be named.

If the marker object has a nonprimitive local closed marker word, Lemma 6.4 cuts off a shorter marker subpackage; minimality then forces terminal cancellation, strict descent, B7 verification, or a routed endpoint/B6/B7 object. If the residual primitive pair is consecutive and cap-eligible, Lemma 6.5 cancels it and lowers M. If it is not cap-eligible, the failure table in Lemma 6.5 names the downstream local analysis or descent. If the marker trace is separated by ordinary carrier data, it is not a purely local B5 object; it is an endpoint, near-whole/anchor, B6, or B7 obligation. A whole-witness marker return is recorded as N and removed from the active record. All outputs are therefore admissible in the sense of Section 2.

Remark 6.7 (Scope of B5). B5 is a safety and normalization block. It proves that the common vertex is singular, that mixed passages must be carried as rank-one marker data, and that local cap-eligible primitive pairs discharge by a named marker-cap cancellation. It does not by itself prove that every marker-containing support-gate disk is terminal; nonlocal marker objects are deliberately exported to endpoint discharge, B6, or B7.

7. Support-Gate and Ported-Disk Routers

This section is the B6 part of the proof. Its role is not to prove, by one geometric stroke, that every support-gate disk with original vertices is reducible. That implication is false as stated: inside a pure even dumbbell, an ordinary-vertex disk may contain only ported path pieces, not a smaller dumbbell or an automatic replacement edge. The correct statement is a router theorem. A B6 object is either cell-clean and v-free, ordinary-original-vertex ported, common-vertex/marker data, or near-whole/whole-capture data. Each case is terminal, strictly lowers $\mathfrak{K}^*$, or is exported with full typed data to B5, B7, endpoint discharge, or B4p.

A **typed support-gate disk** is a compact PL disk whose boundary is a simple closed support-gate curve, with full boundary state data: side, orientation, domain pair, support, gate, endpoint role, marker word, anchor state, and replacement ledger. It is **active** only as an obligation in the ledger Λ of Section 2. A support-gate operation is legal only when it preserves the full typed boundary relation or produces an explicitly named obligation of lower ledger rank.

Lemma 7.1 (Cell-clean refinement). Let B be a simple support-gate loop in a finite PL witness complex. After subdividing support and gate arcs at the finitely many points where B changes type or meets existing cell-boundary vertices, B is a subcomplex of the one-skeleton. In the refined complex no support-switch cell crosses B transversely: every cell interior lies entirely inside B, entirely outside B, or is disjoint from B and meets it only along boundary vertices or boundary arcs.

Proof. The refinement only subdivides existing support and gate arcs. It does not change lane continuation, labels, terminal sections, marker data, endpoint data, or the number of original vertices. Once refined, B is a simple closed PL subcomplex. By the Jordan curve theorem, its complement has an inside and an outside. The interior of each support-switch cell is connected and disjoint from the one-skeleton containing B, so it lies on one side. Hence no cell interior crosses through B. Any remaining intersection is on the boundary of the cell and is, after refinement, a union of boundary vertices or boundary arcs. If such a refinement were impossible, the obstruction would already be a gate-identification, cell-overlap, returning-lane, or non-support-gate normality defect.

Lemma 7.2 (B6a cell-clean support-gate router). Let D be an innermost typed support-gate disk in a minimal reduced witness package. Assume

$$D \cap V_{\mathrm{orig}} = \varnothing, \qquad v \notin D,$$

and assume that D contains no pending marker, endpoint, replacement, or edge-removal obligation. Then D routes to one of the following:

1. full typed identity deletion;
2. same-relation splice;
3. smaller typed support-gate loop or smaller typed corridor;
4. smaller closed train;
5. direct typed normality contradiction;
6. a strict descent in $\mathfrak{K}^*$.

Proof. By Lemma 7.1, refine so that the boundary of D is a subcomplex. Since D contains no original vertices, no v, and no pending replacement or marker data, its interior is a finite typed carrier made only of crossing vertices, support edgelets, gate intervals, lane components, and typed boundary states.

If the two boundary readings give the full typed identity relation, deletion is legal because all labels agree, including marker, endpoint, anchor, and replacement coordinates. The deletion removes cells or edgelets and lowers one of the carrier coordinates of $\mathfrak{K}^*$. If the boundary readings give the same active relation, splice the two boundary intervals; again the full typed relation is unchanged and the carrier size decreases.

If the boundary relation is nontrivial, choose an innermost component of the finite relation graph realizing it. If it has the same boundary problem as the active object, promote it to the active support-gate object; it is proper, so Q, S, E, L, or U decreases. If it has different typed boundary data, it is an explicitly smaller typed support-gate loop or corridor and is routed with lower obligation rank. If the finite gate graph inside D has a cyclic component, that component is a closed typed train with fewer cells than the ambient object, lowering the train coordinate L.

The remaining possibilities in the finite incidence graph are repeated support, double crossing, gate identification, terminal lane, returning lane, or branching/merging. A double crossing is terminal. The others either cut off a smaller cell-clean disk or are typed normality defects. Thus no independent cell-clean B6 residual remains.

Definition 7.3 (Port graph of an original-vertex disk). Let D be a typed support-gate disk. The port graph P_D has as vertices the original vertices lying in D together with the boundary ports where original supports meet ∂D. Its edges are connected original-support pieces inside D between original vertices and boundary ports. Interior ordinary vertices have degree 2 in the original dumbbell; the common vertex v has degree 4; boundary ports have degree 1 in P_D.

A v-free ported path component has **short length** $0, 1$, or 2 when it is respectively a single ordinary vertex, a one-edge ordinary path, or a two-edge ordinary path. It is **long** when it contains four consecutive ordinary vertices

$$x_0, x_1, x_2, x_3$$

on one original cycle, hence a graph 3-path $x_0x_1x_2x_3$.

Lemma 7.4 (Pure dumbbell port-graph classification). Let D be an innermost label-essential support-gate disk in a pure even dumbbell $A \cup B$ with $A \cap B = \{v\}$. Then every original-vertex component of P_D is one of the following:

1. a v-free ported path segment lying in $A \setminus \{v\}$ or $B \setminus \{v\}$;
2. a singular v-arm package, routed to B5/B6c with marker data preserved;
3. a boundary-closed support-gate object using original arcs and support-gate boundary arcs, routed to the ordinary support-gate router;
4. a near-whole or whole-cycle capture, routed by Lemma 7.10 below;
5. label-inessential original material, deletable or splicable only with a full typed certificate.

In particular, an ordinary-original-vertex disk does not automatically contain a smaller T_3-critical dumbbell core.

Proof. The original graph has exactly two graph-theoretic cycles, A and B, and both contain v. Therefore a connected component of P_D avoiding v lies in a cycle with v removed, hence in an ordinary path. Cutting by ∂D turns it into a ported path segment, possibly together with support-gate boundary arcs if the witness closes around it.

If a component contains v, the local degree is four and the rank-one common-vertex ledger is active; this is not an ordinary ported path and must be routed to B5/B6c. If a disk captures essentially all of one cycle, then it is a near-whole or whole-cycle endpoint/anchor object, not a smaller dumbbell counterexample. If it captures both cycles, it is the whole original witness and is a non-descent unless a separate marker, anchor, B7, or replacement certificate is supplied. These cases exhaust subgraphs of two cycles sharing a single vertex.

Lemma 7.5 (Short-port chain router). Let P be a v-free short ported path component of a minimal reduced B6 disk. After exporting certified boundary R3 candidates, blocked boundary R3 candidates, and endpoint/anchor defects, the remaining two-terminal typed relation of P is the unique ordinary degree-two chain continuation. Hence it is a typed identity or ordinary splice if the terminal states match, and a typed normality defect otherwise.

The finite short-port table is:

short type	possible nontrivial event	route
x_i	ordinary degree-two turn	typed continuation, splice, or normality defect
$x_i x_{i+1}$	boundary 3-path using both adjacent port edges	B7 if certified; otherwise B6b/B7 blocker or endpoint/anchor router
$x_i x_{i+1} x_{i+2}$	left or right boundary 3-path using one adjacent port edge	B7 if certified; otherwise B6b/B7 blocker or endpoint/anchor router

Consequently, if every ordinary ported component of a v-free B6 disk has length at most 2, then the disk is not a residual ordinary-original-vertex disk: it routes to B7, B6b/B7 blocker routing, endpoint/anchor discharge, typed continuation/splice, typed normality failure, or the cell-clean router of Lemma 7.2 after the short components have been removed.

Proof. At a single ordinary vertex $x \neq v$, a small disk meets exactly two edge germs and has two sectors. There is no marker transition and no support-switch choice at x; the local passage is forced by the ordinary degree-two continuation. If the typed terminal states agree with this continuation, the component is deletable or splicable. If they do not, the failure is endpoint, anchor, side/orientation, or normality data.

For a length-one or length-two short path, compose these forced local continuations along the ordinary path. Since the path is v-free, the marker word is unchanged. Since the path is short, the only graph 3-paths involving it are the boundary candidates listed in the table, obtained by adjoining adjacent port edges. Once certified and blocked boundary-

R3 cases and endpoint/anchor failures are exported, no independent short-port relation remains. Removing all matched short components leaves either a cell-clean disk or an already named routed obligation.

Lemma 7.6 (R3 candidate and saturation router). Let a v-free ported disk contain a long ported path component with a graph 3-path

$$P = x_0 x_1 x_2 x_3,$$

where x_1, x_2 are ordinary degree-two vertices. Let Q be the crossing of the first and third edges, and let $\Delta(P)$ be the standard edge-removal triangle bounded by the middle edge $x_1 x_2$ and the subarcs from x_1, x_2 to Q. Then exactly one of the following occurs:

1. $\Delta(P)$ is empty of original vertices and label-essential data, and the B7 edge-removal certificate applies;
2. $\Delta(P)$ contains a proper label-essential support-gate disk, yielding a lower Q, S, E, L, or U obligation;
3. $\Delta(P)$ contains a proper ordinary-original-vertex ported disk, yielding a lower P or $\mathrm{B}_{\mathrm{blk}}$ B6 obligation;
4. $\Delta(P)$ contains v or marker data, routed to B5/B6c;
5. $\Delta(P)$ contains endpoint, anchor, near-whole, or replacement data, routed to endpoint discharge, Lemma 7.10, or B7;
6. $\Delta(P)$ contains only label-inessential material, which collapses by a full typed identity certificate and then returns to the first row.

Thus a long ported path gives an R3 candidate, not an R3 theorem. If the candidate is not certified, its first obstruction is a named lower router or a strictly lower obligation.

Proof. The first and third edges of P are nonadjacent original edges in a thrackle drawing, so they cross exactly once. This defines the standard R3 triangle. If the triangle is empty and all typed labels on its boundary satisfy the edge-removal and obstruction-persistence certificate, Section 8 applies. Otherwise inspect the first datum preventing that certificate. A label-essential support-gate subdisk or ordinary-vertex ported disk inside $\Delta(P)$ is proper because it lies inside the triangle. Marker, endpoint, anchor, near-whole, and replacement data are not R3 certificates; they are precisely the B5, endpoint, B6d, and B7 ledgers. If the interior data are label-inessential, the full typed identity calculus deletes them before the same R3 candidate is tested again.

Lemma 7.7 (Blocked-triangle normal form). In a minimal residual v-free B6b disk, choose a blocked R3 candidate P whose active triangle-content rank

$$\theta(P) = \big(|\Delta(P) \cap V_{\mathrm{orig}}|, \#\{\text{ported components meeting } \Delta(P)\}, \#\{\text{boundary exit}$$

is lexicographically minimal after the earlier coordinates of $\mathfrak{K}^*$. Then every blocker inside $\Delta(P)$ either routes to B5/B6c, B7, endpoint/anchor discharge, Lemma 7.5, or a lower $P, \mathsf{B}_{\mathrm{blk}}, U$ obligation, or else belongs to a triangle-saturated blocker dependency cycle preserving θ.

Equivalently, after condensing the finite blocker dependency graph into strongly connected components, every terminal residual component is θ-rigid: every internal blocker arrow preserves active triangle rank. Every rank-changing arrow or SCC exit is a descent or an already named router output.

Proof. Consider an innermost connected original-vertex component or labelled obstruction inside $\Delta(P)$. If it exits through a boundary port or replacement side, the blocked triangle has produced endpoint, anchor, or B7 data. If it contains v, it is B5/B6c. If all ordinary ported pieces in it are short, Lemma 7.5 applies. If it contains a long ported subpath, its R3 candidate has either a certified triangle, handled by B7, or a blocked triangle. If the new blocked triangle has smaller active content rank, minimality of $\theta(P)$ is contradicted; hence any residual blocker must have the same rank and continue the dependency chain.

The blocker graph is finite. Its condensation is a finite DAG. A terminal SCC cannot contain a lowering arrow, by minimality. It also cannot contain a strictly raising arrow without a lowering arrow somewhere on the directed cycle returning to the starting rank. Therefore every terminal residual SCC is θ-rigid. Nonterminal exits are exactly the routed outputs already listed.

Theorem 7.8 (B6 ordinary-original-vertex router integration). Let D be a minimal reduced typed support-gate disk containing ordinary original vertices. Then B6 has one of the following admissible outputs:

1. D: typed identity deletion, same-relation splice, typed continuation, smaller support-gate loop, smaller corridor, or smaller train, hence strict descent in $\mathfrak{K}^*$;
2. C or R: a verified R3 or verified replacement obligation for B7;
3. R or D: a blocked-R3 obligation whose first obstruction routes to B5/B6c, B6d, endpoint discharge, B7, or a lower B6 obligation;

4. R: a θ-rigid blocker SCC, which the SCC compiler turns into a certified-diagonal-free alternating blocker enclosure with recorded δ-rank;
5. R, D, or N: a θ, δ, ϱ-rigid whole-boundary dirty cycle, which either has an admissible cut state and opens to an ordered typed boundary corridor for B4p, fails to open by visible endpoint, marker, replacement, near-whole, whole-capture, typed-normality, or lower-cost data, or is proved to be the whole active witness;
6. N: a whole-witness capture or whole-witness return marked as inert non-progress.

No proper ordinary-original-vertex disk remains as an independent residual.

Proof. First apply the port-graph classification. Components containing v are B5/B6c, and near-whole or whole-capture components are handled by Lemma 7.10. A v-free ordinary component is a ported path. If all such components are short, Lemma 7.5 removes the disk as a residual. Therefore any remaining v-free ordinary-original-vertex disk contains a long ported path and hence an R3 candidate.

Apply Lemma 7.6 to a long candidate. A certified empty triangle is a B7 output. Any proper support-gate or original-vertex obstruction inside the triangle lowers the corresponding coordinate of $\mathfrak{K}^*$. Marker, endpoint, anchor, near-whole, and replacement data are exported with full typed labels. Hence the only residual case is a blocked R3 candidate with no lower first obstruction.

Choose a blocked candidate of minimal θ. Lemma 7.7 shows that the terminal residual blocker graph is θ-rigid. The SCC compiler is finite: if a blocker edge leaves the SCC or changes rank, it gives a descent or a named router. A terminal rank-rigid SCC induces an alternating blocker enclosure. Certified ears, certified diagonals, and double crossings are terminal or descending; geometric-only diagonals are recorded as diagonal-obstruction data and are not used as proof cuts. The diagonal-obstruction coordinate δ and replacement-obligation coordinate ϱ remove every non-rigid first-blocker return by descent or router output.

Thus the only geometric remainder is a θ, δ, ϱ-rigid whole-boundary dirty cycle. If it has an admissible cut state, cut it there and read the full typed boundary relation as an ordered typed boundary corridor; B4p then applies. If it has no admissible cut state, the first obstruction to cutting is visible in the typed ledger: endpoint, marker, replacement, near-whole, whole-capture, normality, or a lower coordinate. That is an admissible router output. A whole-witness return is recorded as N and removed from the active ledger, not counted as proof progress. Therefore no independent B6 ordinary-vertex residual remains.

Remark 7.9 (Scope of the B6 integration theorem). The theorem is an integration statement for the ordinary-vertex residuals. It does not assert that every blocked R3 triangle is itself terminal, nor that a whole-boundary dirty cycle is a contradiction. It asserts that each such object has a finite typed destination: a B7 verification, endpoint discharge, B5/B6c, B6d, B4p, a strict descent, or an inert whole-witness rejection. Section 8 supplies the B7 local checks, and Section 9 discharges the endpoint outputs.

Lemma 7.10 (Near-whole anchor router). A near-whole capture of one original cycle routes as follows:

case	output
captured cycle length at least 6	internal R3 candidate, hence certified B7 or B6b/B7 blocker router
captured cycle length 4	finite two-edge port table
ports cap through v	B5 marker endpoint
whole original witness	non-descent, not proof progress

Proof. A literal whole-cycle capture contains v, so any cap through the missing vertex is marker/common-vertex data and belongs to B5/B6c. If the near-whole capture is v-free and the captured even cycle has length at least 6, then the captured ordinary path contains a graph 3-path not using v. Lemma 7.6 makes it an R3 candidate; the outcome is certified B7, blocked-R3/B6b routing, endpoint/anchor data, or lower complexity.

If the cycle has length 4, the captured path has the form $x_1x_2x_3$ with ports on the two edges incident with v. The only boundary graph 3-paths are the two candidates obtained by adjoining one adjacent v-edge at a time. Each is certified empty, blocked, or endpoint/marker defective. After those rows are exported, Lemma 7.5 gives the unique ordinary degree-two chain relation, hence typed continuation/splice or typed normality failure. If the capture is the whole active original witness, it is recorded as N and cannot be used as a descent or terminal contradiction.

8. Certified Edge Removal and Replacement

The B7 block records the finite data required before an edge removal, an R3 move, or a replacement arc may be used in the minimal-witness argument. The basic principle is simple: a proposed local surgery is a proof step only after its local hypotheses and its effect on the non-T_3 obstruction have both been checked. If the required data are present, the surgery is either terminal or strictly decreases $\mathfrak{K}^*$. If some datum is missing, the first failure becomes a named lower local problem rather than an implicit contradiction.

The edge-removal principle used below is the standard three-path criterion. A graph $\mathbf{3}$-path

$$x_0x_1x_2x_3$$

with degree-two middle vertices and the prescribed crossing between the first and third edges may be reduced only when the associated edge-removal triangle is empty of original vertices and all typed obstruction data persist. A long ported path produces such a candidate, not by itself a valid edge removal.

Lemma 8.1 (B7 obligation normal form). Every B7 output in a minimal reduced witness falls into exactly one of the following rows, after applying the first row whose hypotheses are satisfied:

row	status
double crossing of the same original support pair	terminal contradiction
full typed identity, same-relation splice, or typed collapse	strict descent
verified R3 edge removal	terminal or lower G
blocked R3 candidate	route by first obstruction to B6b, B5/B6c, B6d, endpoint discharge, or nested B7
carrier-local strong replacement	removes or lowers a replacement obligation
graph-changing strong replacement with obstruction persistence	graph-level descent
uncertified replacement/R2 candidate	pending replacement obligation
rejected whole-witness or T_3-collapsing replacement	inert non-descent, followed by pullback routing if proper

Proof. A double crossing of the same original support pair contradicts the thrackle condition. A full typed identity, same-relation splice, or typed collapse is legal only because the entire boundary relation agrees, including marker, anchor, endpoint, domain, support/gate, and replacement labels; applying it removes finite carrier data and lowers $\mathfrak{K}^*$.

A verified R3 edge removal requires a real graph $\mathbf{3}$-path, degree-two middle vertices, the prescribed crossing, an empty or fully controlled edge-removal triangle, and persistence of the non-T_3 obstruction or production of a lower witness. Without those data it is only a blocked R3 candidate, and the first obstruction in the triangle determines the downstream analysis: ordinary original vertices go to B6b, v or marker data to B5/B6c, near-whole or anchor data to B6d or endpoint discharge, and replacement data to B7.

A replacement is terminal or descending only with strong local data and obstruction persistence. If it is carrier-local, the abstract obstruction package is unchanged and only the active replacement record improves. If it changes the graph, it must supply fresh non-T_3 data and a comparison map lowering $\mathfrak{K}^*$. Otherwise the proposed replacement is merely an active obligation.

A pending replacement obligation is recorded as

$$R = (\partial R, \alpha_R, \beta_R, \tau_R, \omega_R),$$

where ∂R is the typed boundary interval, α_R is the old arc or edge segment, β_R is the proposed replacement arc, τ_R is the full typed endpoint data, and ω_R is the first obstruction to verification. Its active rank is

$$\varrho(R) = (\mathrm{span}(R), \mathrm{int}(R), \mathrm{obs}(R), \mathrm{return}(R), \mathrm{bdry}(R)).$$

Here span is the affected boundary length, int counts unresolved interior events before the first stable obstruction, obs records the obstruction type, return records whether the attempt is terminal, routed, descending, or same-complexity, and bdry is the shortest boundary interval needed to read the obligation. The multiset of active ϱ-ranks is part of $\mathfrak{K}^*$, so a nested or improved replacement obligation is a genuine descent only when this multiset strictly decreases.

Lemma 8.2 (Strong replacement data). Let $R = (\partial R, \alpha_R, \beta_R, \tau_R, \omega_R)$ be a pending replacement obligation. A strong replacement datum consists of the following finite data:

1. typed endpoint match: the endpoints of α_R and β_R are the same two stable full typed states;
2. replacement lens: α_R, β_R, and two endpoint caps bound a disk D_R in the carrier;
3. no hidden data: the interior of D_R contains no original vertex, v-marker event, anchor endpoint, near-whole or whole-capture data, or unresolved replacement obligation;
4. controlled crossings: every support meeting D_R is a through-strand meeting α_R and β_R in the same prescribed crossing class, and no support meets β_R twice;
5. endpoint-germ compatibility: adjacent original edges keep their allowed common endpoint contacts;
6. obstruction persistence: either the carrier-level obstruction package is unchanged and R is removed or lowered, or a graph-level output package is supplied with a B1 non-T_3 certificate and strictly lower $\mathfrak{K}^*$.

If R has these data, applying the replacement preserves the thrackle condition and either removes/lowers R in the same witness package or produces a strictly lower non-T_3 witness.

Proof. Supports not meeting the replacement lens are unchanged. The endpoint match and endpoint-germ compatibility preserve adjacent-edge incidences at the ends of the replacement. Controlled crossings preserve the crossing pattern with all nonadjacent through-strands and prevent a second crossing with the same support. The no-hidden-data row prevents the replacement from jumping over graph vertices or typed ledger data.

Thus the local drawing after replacement is still a thrackle drawing. The obstruction-persistence row is what makes the move useful for this proof. In the carrier-level case, $T, p, q, \gamma_0, \gamma_1$, the anchored boundary relation, the marker record, the anchor record, and all active local records are unchanged except that R is removed or replaced by lower ϱ data. In the graph-level case, the finite data supply a new even dumbbell drawing T', new B1/B2 data, a new anchored obstruction package, and a comparison map proving strict decrease of $\mathfrak{K}^*$. Hence the replacement is an admissible B7 transfer.

Lemma 8.3 (Local replacement-failure reduction). A local failure of strong replacement data routes to B4p, B5/B6c, B6b, B6d, nested B7 with lower ϱ, endpoint/anchor discharge, double crossing, or lower $\mathfrak{K}^*$.

More precisely:

failed row	output
typed endpoint mismatch	endpoint/anchor defect, typed normality failure, or lower A
no proper replacement lens	B4p/B6b boundary regularization or typed normality defect
hidden ordinary original vertex	B6b ported-disk or blocked-R3 router
hidden v or marker event	B5/B6c
near-whole or whole-capture data	B6d or endpoint discharge
hidden replacement data	nested B7 obligation, lower ϱ, or replacement-rigid dirt
support meets β_R twice	double crossing or B6b blocker
obstruction does not persist	not a local proof step; keep as B7 unless Lemma 8.4 or Lemma 8.5 applies

Proof. Inspect the first failed row in the smallest proper typed region where it is visible. Endpoint mismatch is endpoint or anchor data, because the proposed replacement cannot even identify its typed terminal states. Failure of the lens condition is a boundary regularization problem: if the region is cell-clean it is B4p, if it contains ordinary vertices it is B6b, and if it contains v or marker data it is B5/B6c.

Hidden original vertices, marker data, near-whole data, and replacement data are precisely the named records in Sections 6, 7, and 8. If hidden replacement data is proper, its span, interior-event count, obstruction type, or boundary-reading interval is smaller, so the active ϱ-multiset decreases. If the same support pair is forced to meet twice, the thrackle condition is contradicted; if another ordinary obstruction mediates the second encounter, it is the B6b blocker problem. The only failure not decided locally is obstruction persistence, which is handled by the next two lemmas.

Lemma 8.4 (Domain-data persistence). Let $T \rightsquigarrow T'$ be a graph-changing admissible even-dumbbell surgery with strong local data. Suppose B1 domain data for $T \notin T_3$ survive outside the surgery lens:

1. all ordinary matched vertices survive as ordinary vertices of T';

2. the replacement lens is disjoint from their small neighborhoods and from the common vertex;
3. the outside face bijection preserves the two incident domains of every matched vertex;
4. in the v-certificate case, the four domains incident to v are identified with the four domains incident to v'.

Then the transported data prove $T' \notin T_3$. If the surgery lowers the graph coordinate G, it gives graph-level descent persistence.

Proof. In the ordinary B1 case, the certificate is a matching of size 4 in the ordinary domain graph. The four matched ordinary vertices survive, and their incident domain pairs are transported by the outside face bijection. Therefore the same four domain edges are pairwise disjoint in $G_D(T')$. Since $G_D(T')$ is bipartite, a three-domain cover of the ordinary vertices would be a vertex cover of size at most 3, contradicting the matching of size 4.

In the v-certificate case, fix $d' \in I_{T'}(v')$ and let $d \in I_T(v)$ be the corresponding transported domain. The stored certificate in $G_D(T) - d$ is a matching of size 3 using vertices untouched by the surgery. Transporting it gives a matching of size 3 in $G_D(T') - d'$. Hence no two domains away from d' cover the ordinary vertices after deleting d'. This holds for every $d' \in I_{T'}(v')$, so no three-domain cover can cover both all ordinary vertices and v'. Thus $T' \notin T_3$.

Lemma 8.5 (Domain-critical replacement and full-propagation test). Let $R : T \rightsquigarrow T'$ be a residual strong graph-changing replacement after local failures and certificate-preserving surgeries have been removed. If no B1 certificate survives outside the affected interval, then the outside part of the domain graph is T_3-critical in the following finite sense:

1. in the ordinary B1 case, the preserved ordinary vertices are coverable by at most three preserved domains;
2. in the v-certificate case, if the v-domain ledger is preserved, then for some $d \in I_T(v)$ the preserved ordinary vertices outside the affected interval are coverable by d and at most two further preserved domains.

Let S_R be such an outside cover and let J_R be the affected original-cycle interval. If S_R fails to propagate across J_R, the first failure event is endpoint/anchor, marker/common-vertex, uncovered ordinary vertex, crossing face-bijection change, replacement ledger, near-whole/whole-capture, or double-crossing data; the corresponding output is endpoint discharge, B5/B6c, B6b, nested B7 or lower ϱ, B6d, direct terminal contradiction, or lower $\mathfrak{K}^*$.

If S_R propagates across all of J_R and all endpoint caps are compatible, then all ordinary vertices of T' are covered by at most three transported domains. Exactly one of the following occurs:

1. the transported cover meets $I_{T'}(v')$, so T' is T_3 and the replacement is rejected as non-descending;
2. the cover misses $I_{T'}(v')$ and every $G_D(T') - d$ has matching number at least 3, giving a fresh B1 v-certificate and graph-level descent persistence;
3. the cover misses $I_{T'}(v')$ and some $G_D(T') - d$ has matching number at most 2, so Konig's theorem gives a three-domain cover of all original vertices and T' is T_3.

Proof. If an ordinary B1 certificate does not survive outside the affected interval, then the preserved outside domain graph has no matching of size 4; by Konig's theorem it has a vertex cover of size at most 3. In the v-certificate case, absence of a surviving certificate means that for some preserved v-domain d, the graph outside the affected interval after deleting d has no matching of size 3; by Konig, it has a two-domain cover. Adding d gives the stated outside cover.

Propagation of S_R is locally constant along open edgelets of the affected interval. Hence a failure has a first finite event. It cannot occur at a smooth point, so it occurs at an endpoint cap, ordinary vertex, crossing, marker/common vertex, replacement boundary, near-whole boundary, or attempted repeated support crossing. These are exactly the router rows listed in the statement.

If no first failure occurs, the cover propagates over all ordinary vertices of T'. If one transported domain is incident with v', then the same at-most-three domains cover every original vertex, and T' is T_3. If the transported cover misses $I_{T'}(v')$, the B1 case-2 test is finite. When every $G_D(T') - d$ has a matching of size at least 3, those matchings are a fresh v-certificate for $T' \notin T_3$. If some $G_D(T') - d$ has matching number at most 2, Konig gives a cover C_d of size at most 2 after deleting d, and $C_d \cup \{d\}$ covers all original vertices of T'. Hence T' is T_3.

Lemma 8.6 (Rejected replacement pullback). If the full-propagation test rejects a strong graph-changing replacement because T' is T_3, then pulling the resulting three-domain cover of T' back to T either produces a local B5/B6b/B6d/B7 or endpoint router, lowers ϱ, or proves that the affected interval was the whole active witness and hence a non-descent.

Proof. Let S' be the three-domain cover of T' obtained in Lemma 8.5. Pull it back through the outside face bijection on the complement of the affected interval J_R. The pulled-back cover is well-defined and covers every original vertex outside J_R. It cannot extend to all of T, because $T \notin T_3$.

Read along J_R from either endpoint cap. Since the event list is finite, either there is a first obstruction to extending the pulled-back cover, or J_R is the whole active witness. A first obstruction cannot occur at a smooth point of a preserved edgelet. If it is an ordinary vertex or a crossing/through-strand face change, it is B6b ported-disk or blocked-R3 data. If it is v or a marker event, it is B5/B6c. If it is an endpoint cap, anchor mismatch, or near-whole boundary, it is endpoint or B6d data. If it is replacement-ledger data, it is a nested B7 obligation; when proper, its ϱ-rank is lower. If no proper first obstruction exists, the attempted replacement was a whole-witness transformation and is recorded as N, not as a contradiction or descent.

Remark 8.7 (Scope of B7). B7 closes only after its finite local data have been checked. Verified R3, full typed deletion/splice, double crossing, carrier-local replacement, and graph-changing replacement with transported or fresh B1 data are terminal or descending. Blocked R3, unverified replacement, rejected T_3 output, and whole-witness replacement are not terminal; they must route through B5, B6, endpoint discharge, nested B7, or the inert record.

9. Endpoint Discharge

Endpoint discharge is the router for failures concentrated at a terminal section, an anchor cap, a marker cap, or the boundary of a replacement/R3 obligation. It is the place where several downstream outputs meet:

$$\text{B5 marker endpoint}, \qquad \text{B6d near-whole anchor object}, \qquad \text{B7 endpoint oblig}$$

This phrase is useful only if endpoint objects are finite, typed, and controlled by the same global witness complexity. An endpoint object is therefore recorded as

$$E = (\partial E, \tau, \mu, a, \rho, \Pi),$$

where ∂E is the typed boundary trace entering and leaving the endpoint region, τ records side, orientation, domain, support, gate, and endpoint role data, μ is the common-vertex marker word when present, a records the anchor state, ρ is the replacement or blocked-R3 ledger, and Π is the finite multiset of active B7 candidates attached to the endpoint.

The endpoint cost is the endpoint projection of the global scale, not a new minimality:

$$\eta(E) = (M_E, A_E, R_E, U_E),$$

where M_E is marker complexity, A_E is anchor defect, R_E is the active replacement/R3 multiset, and U_E is finite boundary tie-breaker data. More precisely, after Section 8,

$$R_E^\star = \big(\{\{\varrho(R) : R \in \Pi(E)\}\}, |\Pi(E)|, U_E\big)$$

is the endpoint replacement coordinate. Rejected whole-witness B7 candidates are removed from $\Pi(E)$ and stored only in the inert ledger. Thus an endpoint step is productive only if it is terminal, lowers $\mathfrak{K}^*$, enters a named router with full typed data, or removes a whole-witness non-descent from the active ledger.

Lemma 9.1 (Endpoint normal form). Before applying the near-whole router and the B7 productive ledger, every non-discharged endpoint object in a minimal reduced witness is one of:

1. a primitive marker pair $m_A m_B$ or $m_B m_A$;
2. an anchor-rigid near-whole cycle capture with no proper anchored subobject of lower A_E;
3. a pending B7 replacement/R3 obligation or blocked R3 obligation carried by $\Pi(E)$.

Proof. If the marker word has nonprimitive local length, Lemma 6.4 cuts off a shorter marker package. Lemma 6.3 prevents this package from being erased by ordinary moves. Therefore a reduced endpoint cannot retain a nonprimitive marker word unless it has already produced a marker cancellation, strict descent, B7 certificate, or routed endpoint/B6/B7 object. The only local marker residual is a primitive pair.

A whole-capture endpoint containing or capping through v is marker/common-vertex data and belongs to B5/B6c, not to an ordinary endpoint residual. If it is v-free but contains all ordinary vertices of one original cycle except the two ports near v, then it is a near-whole anchor object. If it had a proper anchored subobject with the same full typed boundary relation, then the anchor defect A_E, ported-disk coordinate P, or tie-breaker U_E would decrease. Hence the residual case is anchor-rigid.

Finally, an endpoint may carry an R3 or replacement candidate. If the R3 triangle is certified empty and obstruction-persistent, or if the replacement has the certificate of Lemma 8.2, Section 8 makes it terminal or descending. If the triangle is blocked or the replacement is uncertified, it is exactly a pending B7 obligation in $\Pi(E)$. These alternatives exhaust the typed data (μ, a, ρ, Π) of a reduced endpoint object.

Lemma 9.2 (Productive endpoint discipline). Evaluating an active B7 candidate in $\Pi(E)$ has exactly one productive outcome:

1. certified terminal/descent;
2. local router or lower $\mathfrak{K}^*$;
3. nested B7 with strictly lower ϱ;
4. whole-witness rejection, which removes the candidate from $\Pi(E)$ but is not terminal progress.

After all active B7 candidates of E have been evaluated, either E has produced one of the first three outputs, or E has no productive B7 output and must be discharged by marker, anchor, near-whole, B6b blocked-R3, typed continuation/splice, or whole-witness non-descent rows.

Proof. Certified R3, full typed splice/collapse, carrier-local replacement, and obstruction-persistent graph-changing replacement are exactly the terminal or descending rows of Lemma 8.1. Local certificate failures route by Lemma 8.3. Domain-critical discontinuities and rejected graph-changing replacements route or become whole-witness non-descents by Lemmas 8.5 and 8.6. A nested replacement obligation is productive only if its span, interior-event count, obstruction type, return type, or boundary-reading interval is lower; equivalently, the active ϱ multiset decreases. If none of these coordinates decreases, the same-complexity data is replacement-rigid dirt already recorded in the B6/B4p cyclic boundary ledger, not an endpoint terminal move.

The only nonproductive B7 status is an explicitly rejected whole-witness candidate. Removing it from $\Pi(E)$ lowers the finite active candidate count and prevents the proof from using the same rejected branch again as if it were terminal. Since $\Pi(E)$ is finite, repeated endpoint B7 evaluation terminates. If no active B7 candidate remains, the endpoint must be routed by its non-B7 data (τ, μ, a), not declared solved by B7.

Lemma 9.3 (No proper inert-only endpoint residual). After primitive marker normalization, rank-one cap cancellation, near-whole routing, short-port routing, and productive B7 evaluation, a proper endpoint object cannot have all B7 candidates inert and no remaining marker, anchor, B6b, B6d, B7, or typed-continuation output.

Proof. By Lemma 9.1, a non-discharged endpoint object is a primitive marker pair, an anchor-rigid near-whole capture, or a B7/blocked-R3 obligation.

For a primitive marker pair, the consecutive cap-eligible case cancels by Lemma 6.5 and lowers M_E. Failure of cap eligibility is not inert: terminal-state mismatch is endpoint/anchor data, ordinary carrier data between the markers is a separated marker or B6/B7 object, hidden graph data gives B6, and replacement boundary data gives B7. Thus a primitive marker endpoint always lowers M_E or routes.

For an anchor-rigid near-whole capture, Lemma 7.10 gives the length dichotomy. Captures of length at least 6 contain internal R3 candidates and route to B6b/B7. The length-4 case is the finite two-edge port table: caps through v are B5 marker data, boundary R3 rows are B7 or B6b/B7, endpoint/anchor failures lower or repair A_E, and the remaining ordinary chain is typed continuation/splice or normality failure. Whole-witness captures are moved to the inert ledger and are not proper endpoint residuals.

For B7 data, Lemma 9.2 evaluates every active candidate. A certified candidate is productive; a local or rejected candidate routes by Lemma 8.3, Lemma 8.5, or Lemma 8.6; a nested candidate lowers ϱ; and a whole-witness rejection is removed from $\Pi(E)$. A blocked R3 obligation is, by definition, a B6b/B7 router output rather than an inert endpoint. Hence a proper endpoint with no active B7 candidate and no marker, anchor, B6b, B6d, B7, or typed-continuation output contradicts the normal form.

Theorem 9.4 (Endpoint discharge router). Every proper endpoint object routes to terminal/lower $\mathfrak{K}^*$, B5/B6b/B6d/B7 productive routing, anchor repair, typed continuation/splice, or whole-witness non-descent. Endpoint discharge has no independent residual beyond the B6b/B7 routers to which it legitimately exports.

Proof. Let E be a reduced endpoint object. If it is not in normal form, the reduction to Lemma 9.1 gives a lower endpoint projection of $\mathfrak{K}^*$ or a named router. In normal form, there are three cases.

Primitive marker pairs are discharged by the cap-relation table of Lemma 6.5, unless a cap-eligibility hypothesis fails; every failure is endpoint/anchor, separated-marker, B5/B6/B7, or lower-cost data. Anchor-rigid near-whole captures route by Lemma 7.10 and either produce B5 marker data, B6b/B7 blocker data, endpoint/anchor repair, typed continuation/splice, or whole-witness non-descent. B7 candidates are evaluated by Lemma 9.2; by Lemma 9.3, once inert whole-witness candidates are removed, no proper endpoint can remain with no productive output.

Thus every proper endpoint either lowers $M_E, A_E, R_E^\star$, or U_E, supplies a certified terminal/descent, enters B5, B6b, B6d, endpoint/anchor repair, or B7 with full typed data, or is explicitly marked as a whole-witness non-descent. These are exactly the admissible outputs in Section 2.

Remark 9.5 (Endpoint scope). Endpoint discharge is a router, not an additional terminal theorem. It closes the bookkeeping gap that would otherwise allow rejected replacements, whole-cycle captures, or primitive marker pairs to be reused as fake progress. The only endpoint non-output is an explicitly inert whole-witness non-descent; every proper endpoint remains productive or strictly lower.

10. Internal Minimal-Witness Closure

This section is the internal closure step. It does not introduce another geometric obstruction. It records the point at which the preceding local reductions, once supplied with the finite verifications explicitly named in the statements, become a single minimal-counterexample argument.

Let $\mathcal{U}_{\mathrm{Con}} \subset \mathcal{D}_{\mathrm{Con}}$ be the set of unresolved descriptions surviving the external reductions to irreducible non-T_3 even dumbbell thrackles. Thus $d \in \mathcal{U}_{\mathrm{Con}}$ carries the B1/B2 endpoint data, the anchored-carrier data produced by B3 when available, and the active-obligation multiset $\Lambda(d)$ of Section 2.

Definition 10.1 (Admissible Conway reduction catalogue). A Conway reduction catalogue is admissible for a description d if the B1/B2 setup initializes the description and every subsequent local block in Sections 4--9 is a cost-controlled description-level transfer for Γ_{Con} with only the following outputs:

block	admissible outputs
B1/B2 domain and path extraction	finite setup data for the initial description; not a descent output
B3 anchored-carrier handoff	T, D, or R, where the anchored ordered corridor is the named R_{B4} handoff

block	admissible outputs
B4 typed corridor regularization	T, D, R, or N, where a typed regular corridor is the named R_{reg} handoff
B5 common-vertex marker discharge	D, R, or N
B6 support-gate and ordinary-vertex reductions	T, D, R, C, or N
B7 replacement and finite-verification reductions	T, D, R, C, or N
endpoint discharge	T, D, R, or N; typed continuations, splices, and anchor repairs are D when they lower cost and R when they create a named downstream obligation

Here T, D, R, C, N have the meanings of Section 2.4. In particular, an R-output is not a completed proof step: it is a named downstream obligation with full typed data preserved. A C-output is not terminal until its finite verification table has been evaluated. An N-output is admissible only for a candidate proved to be the whole active witness, and it is removed from $\Lambda(d)$.

Lemma 10.2 (Catalogue admissibility of the developed blocks). Subject to the finite verification rows explicitly named in Sections 5--9, the local blocks developed in this manuscript form an admissible Conway reduction catalogue.

Proof. The assertion is a bookkeeping verification against the output taxonomy of Section 2.4.

The B1/B2 block of Section 3 is a finite setup step. It adds the domain matching data and complementary paths; it is not used as a descent. The B3 block, Theorem 4.6, either gives a terminal contradiction, gives a strict descent in $\mathfrak{K}^*$, exports a named active obligation, or produces the anchored ordered corridor needed by B4. The proof of Theorem 4.6 preserves the original p, q terminal labels, so the transfer is a transfer on Conway descriptions rather than on an unrelated local corridor.

The B4 block, Theorem 5.8, is admissible for the same reason. Identity bands, same-relation bands, and proper subcorridors lower the carrier coordinates H, S, E, U only when the full typed boundary relation, including marker, anchor, endpoint, and replacement labels, is preserved. Nonregular outputs are not discarded: marker data is sent to Section 6, support-gate or ordinary-vertex data to Section 7, replacement data to Section 8, and endpoint data to Section 9; the regular branch is the R_{reg} normal-form handoff to the ladder/train support-gate obligations, not an unstated terminal case.

The common-vertex block is Section 6. Lemmas 6.1--6.5 reduce nonprimitive marker data to separated marker obligations, B6/B7 obligations, endpoint data, or a strict marker descent. Theorem 6.6 proves that a proper marker residual cannot remain once cap-eligible primitive pairs have been evaluated. Whole-witness marker returns are inert and are not counted as progress.

The support-gate and ordinary-vertex block is Section 7. Cell-clean disks are handled by Lemmas 7.1 and 7.2; pure short-port chains by Lemma 7.5; R3 candidate and saturation data by Lemmas 7.6 and 7.7; ordinary-vertex integration by Theorem 7.8; and near-whole capture by Lemma 7.10. Each nonterminal branch lowers a recorded carrier, port, blocker, diagonal, or obligation rank, or else exports a fully typed B5/B6d/B7/endpoint obligation. The ordinary-vertex reduction therefore supplies T, D, R, C, N outputs only.

The replacement block is Section 8. Lemma 8.1 puts every replacement obligation in terminal, descent, routed, pending, or inert normal form. Lemma 8.2 identifies the finite data required for a strong replacement move. Lemmas 8.3, 8.5, and 8.6 handle local failure, domain-critical failure, and rejected replacement by routing, descent, lower ϱ, or inert whole-witness rejection. No replacement is used as a contradiction without obstruction persistence and the strong finite-verification rows.

Finally, endpoint discharge is Section 9. Lemma 9.1 gives endpoint normal form, Lemma 9.2 evaluates active B7 candidates with respect to the endpoint projection $(M_E, A_E, R^{\star}_E, U_E)$, and Lemma 9.3 excludes a proper inert-only endpoint residual. Theorem 9.4 then routes every proper endpoint object to terminal/lower $\mathfrak{K}^*$, B5/B6b/B6d/B7, anchor repair, typed continuation/splice, or whole-witness non-descent.

Thus every local output is one of the admissible T, D, R, C, N outputs, and every R or C output is assigned to its named downstream evaluation with the relevant typed labels and local ranks still present.

Theorem 10.3 (Internal minimal-witness closure). Assume the external reductions have produced an irreducible non-T_3 even dumbbell obstruction, and assume the local finite verification rows invoked in Lemma 10.2. Then $\mathcal{U}_{\mathrm{Con}}$ is empty. Equivalently, no unresolved Conway obstruction package survives the admissible reduction catalogue.

Proof. Suppose, for contradiction, that $\mathcal{U}_{\mathrm{Con}} \neq \varnothing$. Since B_* is well-founded by Lemma 2.1, choose $d_0 \in \mathcal{U}_{\mathrm{Con}}$ of minimal $\mathfrak{K}^*$.

Apply the B1/B2 setup from Section 3. This supplies the certified same-cycle endpoint data p, q, γ_0, γ_1 and does not alter the minimality problem. Now run the B3 handoff. If Theorem 4.6 gives T, the assumed irreducible non-T_3 thrackle obstruction is contradicted. If it gives D, it produces an unresolved description of smaller $\mathfrak{K}^*$, contradicting the choice of d_0. If it gives R, the output is sent to its named downstream router. In the remaining branch B3 supplies an anchored ordered corridor, so B4 applies.

Run B4. The terminal and descent cases are again impossible for d_0. Every exported marker, support-gate, ordinary-vertex, replacement, endpoint, dirty-cycle, or whole-capture object is a named obligation in $\Lambda(d_0)$ with the full typed boundary relation preserved. The regular-corridor branch feeds the train/support-gate analysis of Section 5: regular cell data, nonregular cell data, ordinary-vertex disks, replacement candidates, marker endpoints, and near-whole captures are precisely the B5, B6, B7, or endpoint objects listed in Lemma 10.2.

It remains only to see that the downstream reduction chain cannot be a same-cost cycle. This is exactly Theorem 2.2 applied to the catalogue verified in Lemma 10.2. A productive step that preserves $\mathfrak{K}$ replaces the active obligation by a finite multiset of lower λ-rank; a finite-verification step is a finite table of such steps; and an inert whole-witness rejection removes the candidate from Λ rather than creating progress. Hence every nonterminal evaluation strictly decreases $\mathfrak{K}^*$.

The finite evaluation starting from any active obligation of d_0 must therefore end in T, D, or N. The first case contradicts the standing obstruction hypotheses. The second contradicts minimality of d_0. In the third case the attempted object has been proved to be the whole active witness and is removed as inert. If no active obligation remains, d_0 is resolved and was not in $\mathcal{U}_{\mathrm{Con}}$. If another active obligation remains, the same description with the inert candidate removed has smaller Λ and hence smaller $\mathfrak{K}^*$, again contradicting minimality.

All possible final states are impossible. Therefore $\mathcal{U}_{\mathrm{Con}} = \varnothing$.

Remark 10.4 (Scope of the closure theorem). Theorem 10.3 is the internal minimal-counterexample closure of the local reduction scheme. In the language of finite descriptions and well-founded cost, the bad set is excluded from the minimal bounded part of Γ_{Con} because every possible local failure is terminal, strictly cheaper, or inert. The theorem becomes a proof of the Conway reduction once the remaining finite verification rows are written in full and the external reductions to the irreducible non-T_3 even-dumbbell case, together with the T_3 bound, are cited or reproved.

11. Remaining Integration Tasks

The present reduction isolates a finite list of local verifications rather than a new family of global obstructions. To turn the internal closure theorem into a complete proof, the following points must be supplied in full.

1. Write the finite-verification appendices. The manuscript still needs the full tables for B4p cyclic cut states, B6b short-port and blocked-R3 outcomes, the SCC compiler, strong replacement data, rejected B7 pullbacks, near-whole length-four captures, primitive marker caps, and endpoint B7 evaluations.
2. Audit every descent coordinate. Each identity splice, same-relation splice, marker cancellation, anchor repair, support-gate extraction, R3 move, replacement, endpoint discharge, and whole-witness rejection must name the coordinate of $\mathfrak{K}^*$ it lowers, or the exact downstream obligation it creates.
3. Preserve the global anchors and records. No local transfer may silently change the p, q endpoints, the B1 non-T_3 data, the marker record, the anchor record, the replacement record, or the active θ, δ, ϱ ranks.
4. Synchronize the manuscript artifacts. The Markdown proof now contains the expanded local reduction through the internal closure theorem; the TeX and PDF versions must be regenerated and checked after the remaining finite-verification tables are inserted.
5. Insert the external reductions. The final proof must cite or reprove the reduction from a Conway counterexample to an irreducible non-T_3 even dumbbell obstruction, and the theorem that T_3-thrackles satisfy Conway's bound.
6. Perform the final no-hidden-router audit. After the TeX rewrite, every exported B5/B6/B7/endpoint object must be traced to a terminal verification, a strict $\mathfrak{K}^*$ descent, a lower active obligation, or an inert whole-witness rejection. No routed object may be treated as terminal merely because it has been named.

Thus the current manuscript is best read as a detailed local-reduction strategy with an internal minimal-counterexample closure theorem. The remaining work needed for a complete proof is finite but substantial: write the local tables in full, verify each transfer against $\mathfrak{K}^*$, and splice in the external Conway reductions.

References

[LPS97] L. Lovasz, J. Pach, and M. Szegedy, *On Conway's thrackle conjecture*, Discrete & Computational Geometry 18(4) (1997), 369-376.

[CN00] G. Cairns and Y. Nikolayevsky, *Bounds for generalized thrackles*, Discrete & Computational Geometry 23(2) (2000), 191-206.

[MN18] G. Misereh and Y. Nikolayevsky, *Annular and pants thrackles*, Discrete Mathematics & Theoretical Computer Science 20(1) (2018), Article #16.

[FP11] R. Fulek and J. Pach, *A computational approach to Conway's thrackle conjecture*, Computational Geometry: Theory and Applications 44(6-7) (2011), 345-355.

[Xu21] Y. Xu, *A new upper bound for Conway's thrackles*, Applied Mathematics and Computation 389 (2021), Article 125573.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work, ChatGPT, by OpenAI, was used to assist with mathematical drafting, formalization, review, and editing. This work is shared as a preliminary AI-assisted mathematical note. The mathematical content may have been only partially reviewed and may contain errors; it should not be treated as peer-reviewed or as a fully verified manuscript.