

Gap-Free Rigidity and Robust Certificates for Extremal Growth in Finite Positive Templates

Drew Alexander Higgins

February 20, 2026

Abstract

We present a gap-free rigidity mechanism for extremal (minimal) growth in finite families of positive template operators. The core outcome is a finite certificate dichotomy: for every reduced descriptor instance, either one produces a concrete obstruction witness (OG1–OG4), or one produces an explicit Doeblin word and a recurrence modulus (RC1) that together yield uniform projective contraction along all tight trajectories. We then upgrade the pipeline to a robust form (RU-A–RU-E): each certificate is accompanied by explicit perturbation margins, so the decision is stable under small changes of template entries. The resulting package yields a deterministic, checkable route from reduction to quantitative Hilbert-metric contraction, with constants that can be read off from the certificate itself.

1 Overview and reading guide

This paper is built for two complementary goals.

- **Structural goal:** explain why “optimality forces regularity” in finite positive-template growth problems.
- **Operational goal:** give a finite certificate contract that either returns a specific obstruction (OG1–OG4) or returns quantitative contraction data that can be verified directly.

The technical spine proceeds in three layers.

- **Reduction and tight core:** the descriptor reduction produces a tight subgraph $G^{(0)}$ and local face data.
- **Gap-free forcing and contraction:** calibrated minimizers concentrate on the tight core; slack and return-cost arguments yield positive-density forcing; Doeblin and recurrence then give projective contraction on minimizers.
- **Certificates and robustness:** each step is packaged as a finite certificate with explicit robustness margins.

1.1 Decidable island for robust switching stability

In the general theory of switching systems and joint spectral radius (JSR), many natural decision problems are not decidable in full generality. For instance, even boundedness questions for products of a fixed finite matrix set can be undecidable in broad rational classes [1]. Accordingly, progress

on “where stability is decidable” typically takes the form of identifying structurally constrained subclasses in which stability reduces to finite witnesses.

The certificate program developed here provides such a subclass. Rather than attempting to decide *JSR* in complete generality, we decide a stronger, certificate-based property: whether the instance admits a *robust uniform contraction mechanism* that is both checkable and stable under perturbation.

Definition 1.1 (Robust contraction certificate property). *Fix a perturbation model and norm as in Section 17. An instance satisfies RU-Stable if the pipeline returns a contraction certificate whose fields include:*

- a tight aperiodic SCC H on which the tight-core data are valid,
- a Doeblin word witness w with margin $\delta > 0$ on the SCC face support,
- a recurrence modulus (RC1) for (H, w) , yielding bounded-gap recurrence of w along every tight minimizer in H ,
- an explicit projective contraction-rate pack (τ, C, κ) ,
- an explicit robustness radius $\varepsilon^* > 0$ such that all of the above remain valid (with degraded constants) for perturbations of size at most ε^* .

Theorem 1.2 (Decidability frontier in the certified class). *For every instance in the certified finite positive-template class considered in this paper, the pipeline terminates and returns exactly one of the following finite, verifiable artifacts.*

- A finite obstruction witness (*OG1–OG4*), certifying that the data required by Definition 1.1 cannot be obtained from the instance by the certificate rules. In particular, *OG4* returns an explicit directed cycle in the avoidance automaton, producing a tight periodic trajectory that avoids the Doeblin word forever.
- A robust contraction certificate establishing RU-Stable for the instance, including an explicit robustness radius ε^* and explicit rate degradation formulas.

Consequently, membership in the property RU-Stable is decidable on this class.

Remark 1.3 (Relation to JSR and switching stability). *Theorem 1.2 does not claim a decision procedure for JSR in unrestricted classes. Instead, it identifies a concrete decidable island: within the present positive-template setting, one can decide whether a robust, quantitative uniform contraction mechanism exists, and one can locate precisely where this mechanism fails via explicit witnesses. This is consistent with, and complementary to, general undecidability barriers [1].*

1.2 Resolution map for neighboring open-problem families

Several neighboring research programs study joint spectral radius, extremal products, ergodic optimization, and synchronization phenomena in broad generality. In full generality many decision questions are provably undecidable, and many structural questions are answered only at the level of existence or genericity. The present work contributes a complementary kind of result: a *certificate calculus* that turns a range of these themes into finite objects whose validity is decidable on the certified positive-template class.

The table below records, for each commonly cited open-problem family, the concrete statement resolved *within the certified class* and the pipeline artifact that realizes the statement.

Open-problem family	Resolved statement in the certified class	Pipeline artifact
Decidability of robust switching stability / JSR-type questions [1, 3]	Membership in RU-Stable is decidable: the pipeline returns either a robust contraction certificate (with ε^* and explicit rate degradation) or an explicit obstruction witness (OG1–OG4), including the avoidance-cycle witness for recurrence failure (OG4).	<code>robust_certificate</code> or <code>obstruction_witness</code>
Mather-set structure for matrix sequences (ergodic optimization for JSR) [2]	The extremal set is represented effectively by a finite symbolic system: a tight-core subshift (Definition 3.6) refined by the avoidance/recurrence automaton (Section 11), yielding an explicit sofic description of extremal itineraries in the certified regime.	Tight-core graph H + avoidance automaton
Generic finiteness behavior for JSR and chamber regularity [4]	Away from certificate-failure surfaces (notably $\gamma \rightarrow 0$ and $\delta \rightarrow 0$) the tight core and obstruction regime are locally constant (RU-B/RU-C), so extremal combinatorics and rates are stable on chambers. Within a chamber, periodicity and recurrence are decided by the obstruction dichotomy.	$\gamma, \delta, \varepsilon^*$ + chamber report
Devil’s-staircase boundary complexity vs. tame regions [4]	Complex behavior is confined to explicit boundary loci where the certificate margins collapse. The robustness layer identifies and localizes these loci, separating stable chambers from boundary pathology by explicit, checkable inequalities.	Exceptional-set report (region mode)

Table 1: How the robust certificate program resolves several open-problem families within the certified positive-template class (part 1).

Interpretation. The point is not to decide joint spectral radius in unrestricted families. Instead, the paper identifies a structurally defined subclass in which the relevant stability and extremal-structure conclusions reduce to finite, checkable certificates. The certificate margins then make these conclusions robust and enable region certification.

Decidability frontier for robust switching stability. Theorem 1.2 gives a dichotomy: either a robust certificate exists, producing explicit contraction rates and a robustness radius ε^* (Definition ??), or an obstruction witness exists (OG1–OG4), and the witness is finite and verifiable. This realizes a decidable island inside the broader JSR landscape where global decision problems are known to be hard and can be undecidable [1].

Effective Mather-set and extremal-language description. Within the certified class, the tight-core subshift (Definition 3.6) and the avoidance/recurrence automata (Section 11) produce an explicit sofic model for extremal itineraries. This is the finite-state analogue of the qualitative Mather-set structure developed in the ergodic-optimization literature [2].

Generic periodicity and chamber stability. The chamber machinery (Section ??) states that, away from explicit tie and margin-collapse loci, the tight combinatorics and the obstruction regime

are locally constant. In those chambers, the pipeline decides whether extremals are periodic (unique tight cycle) or genuinely aperiodic (tight aperiodic SCC with certified recurrence), and it produces the corresponding finite model.

Devil’s-staircase boundaries as certificate-failure surfaces. The region certification layer (Section ??) turns the informal slogan “complexity lives on the boundary” into a checkable statement: nontrivial parameter dependence can only occur where the robustness margins $(\gamma, \delta, \varepsilon^*)$ collapse. This separates stable chambers from boundary pathology in a way that complements known devil’s-staircase phenomena in broader families [4].

Synchronizing bounds for induced automata. The Doeblin search automaton (OG3) and the avoidance/recurrence automaton (OG4) are built explicitly and come with finite state counts. In the acyclic-avoidance regime, the return modulus B is computed (Algorithm ??), which is already a quantitative recurrence statement. When the induced machines fall into tractable synchronizing subclasses (aperiodic, trivial group in the transition monoid, or related checkable hypotheses), known Černý-type bounds become available for these induced automata [5, 6].

Quantitative bounds on Doeblin-word length. When a Doeblin block exists, the pipeline produces an explicit positive word and margin δ (OG3), and the robustness layer converts δ into contraction data. In the certified setting, standard exponent bounds for primitive directed graphs can be used as plug-in upper bounds for the worst-case search radius on the core support [7].

Constructive extremal geometry and extremal-norm analogues. The calibrated section and rate pack (Section ??) provide a concrete geometric object that governs the extremal dynamics together with quantitative contraction and robustness radii. This aligns with the extremal-norm narrative around Barabanov norms, but in the certified class the object is produced and verified by finite witnesses [8].

Bounded-language semigroup decidability template. The certified language of admissible products is not arbitrary: it is the tight-core language refined by explicit avoidance constraints. This is precisely the kind of language restriction under which semigroup problems can become decidable in settings where the unrestricted problems are intractable [9].

Frontier mapping and explicit witnesses. Finally, the obstruction taxonomy (OG1–OG4) and the robustness margins provide an operational map of why certification fails, returning concrete finite witnesses rather than leaving failure as an opaque non-result. This is the main deliverable that makes the subclass practically usable: it either certifies stability with quantitative control or explains non-certifiability with an explicit obstruction.

2 Canonical Normalization and the Tight Subgraph

This section records the structural normalization that underlies the rest of the paper. It converts the extremal-growth problem into the study of a canonically defined zero-set (the *tight* transitions). Subsequent rigidity and obstruction statements are properties of this tight geometry.

2.1 Reduced costs

Let $G = (V, E)$ be the reduced finite graph produced by the descriptor reduction, with edge costs $c : E \rightarrow \mathbb{R}$. Define the minimal cycle mean

$$\lambda_* := \min_{\gamma \text{ cycle}} \frac{1}{|\gamma|} \sum_{e \in \gamma} c(e).$$

Theorem 2.1 (Canonical normalization by a potential). *There exists a function (potential) $\phi : V \rightarrow \mathbb{R}$ such that the reduced costs*

$$c_\phi(e) := c(e) - \lambda_* + \phi(\text{tail}(e)) - \phi(\text{head}(e))$$

satisfy $c_\phi(e) \geq 0$ for all $e \in E$.

Proof. Let $\tilde{c}(e) := c(e) - \lambda_*$. For any directed cycle γ of length k , $\sum_{e \in \gamma} \tilde{c}(e) = k(\bar{c}(\gamma) - \lambda_*) \geq 0$, so $(G, \tilde{c})$ has no negative cycles. Fix a root v_0 in each reachable component and define $\phi(v)$ to be the minimum $\tilde{c}$ -cost of a directed path from v_0 to v . Then for any edge $e : u \rightarrow v$, optimality of $\phi(v)$ yields $\phi(v) \leq \phi(u) + \tilde{c}(e)$, i.e. $\tilde{c}(e) + \phi(u) - \phi(v) \geq 0$. This inequality is exactly $c_\phi(e) \geq 0$. $\square$

2.2 The tight subgraph

Definition 2.2 (Tight edges and tight subgraph). *Fix a potential ϕ as in Theorem 2.1. Define the tight edges*

$$E^{(0)} := \{e \in E : c_\phi(e) = 0\}, \quad G^{(0)} := (V^{(0)}, E^{(0)}),$$

where $V^{(0)}$ are vertices incident to $E^{(0)}$.

2.3 Period and aperiodicity of a tight SCC

Definition 2.3 (Period of a strongly connected digraph). *Let H be a strongly connected directed graph. Its period is*

$$\text{per}(H) := \gcd\{|\gamma| : \gamma \text{ is a directed cycle in } H\}.$$

We call H aperiodic if $\text{per}(H) = 1$, and imprimitive otherwise.

2.4 Minimizers are asymptotically tight

A (one-sided) path $\pi = (e_0, e_1, \dots)$ has length- n average cost

$$\text{Avg}_n(\pi) := \frac{1}{n} \sum_{t=0}^{n-1} c(e_t).$$

We call π *minimizing* if $\liminf_{n \rightarrow \infty} \text{Avg}_n(\pi) = \lambda_*$.

Proposition 2.4 (Asymptotic tightness of minimizing paths). *Let π be a minimizing path and fix ϕ as above. Then the set of times t with $e_t \notin E^{(0)}$ has asymptotic density 0. Equivalently, $E^{(0)}$ is visited with asymptotic density 1 along π .*

Proof. Because E is finite and $c_\phi(e) \geq 0$, define

$$\alpha := \min\{c_\phi(e) : e \in E \setminus E^{(0)}\},$$

with the convention $\alpha = +\infty$ if $E^{(0)} = E$. When $E^{(0)} \neq E$, we have $\alpha > 0$. Let N_n be the number of indices $t < n$ with $e_t \notin E^{(0)}$. Then $\sum_{t=0}^{n-1} c_\phi(e_t) \geq \alpha N_n$. On the other hand, telescoping the potential gives

$$\sum_{t=0}^{n-1} c_\phi(e_t) = \sum_{t=0}^{n-1} (c(e_t) - \lambda_*) + \phi(v_0) - \phi(v_n),$$

where v_n is the vertex reached after n steps. Since ϕ is bounded on the finite reachable vertex set, $|\phi(v_0) - \phi(v_n)| \leq C$. Divide by n and take $\liminf$. Minimizing implies $\liminf_{n \rightarrow \infty} \frac{1}{n} \sum_{t=0}^{n-1} (c(e_t) - \lambda_*) = 0$, hence $\liminf_{n \rightarrow \infty} \frac{1}{n} \sum_{t=0}^{n-1} c_\phi(e_t) = 0$. Therefore $\liminf_{n \rightarrow \infty} \alpha N_n/n = 0$, which forces $N_n/n \rightarrow 0$. $\square$

2.5 A tight-core dichotomy

For a subset $A \subseteq \mathbb{N}$, write its *lower asymptotic density* as

$$\underline{d}(A) := \liminf_{n \rightarrow \infty} \frac{|A \cap \{0, 1, \dots, n-1\}|}{n}.$$

Proposition 2.5 (Tight-core dichotomy). *Let π be a minimizing path and let $T := \{t \geq 0 : e_t \in E^{(0)}\}$ be its set of tight times. For each SCC H of the tight graph $G^{(0)}$, set*

$$T_H := \{t \in T : v_t \in H\},$$

where v_t is the tail vertex of e_t . Then there exists at least one SCC H with $\underline{d}(T_H) > 0$. Moreover, exactly one of the following holds:

(P) every SCC H with $\underline{d}(T_H) > 0$ is imprimitive, i.e. $\text{per}(H) > 1$ (Definition 2.3);

(A) there exists an aperiodic SCC H of $G^{(0)}$ with $\underline{d}(T_H) > 0$, i.e. $\text{per}(H) = 1$.

Proof. Let $T := \{t \geq 0 : e_t \in E^{(0)}\}$ be the set of tight times along π . By Proposition 2.4, T has asymptotic density 1, hence lower density 1. For each strongly connected component H of the tight graph $G^{(0)}$, set

$$T_H := \{t \in T : \text{the time-}t \text{ vertex } v_t \text{ lies in } H\}.$$

The sets $(T_H)_H$ form a *finite* partition of T indexed by the SCCs of $G^{(0)}$. If every T_H had lower density 0, then their finite union T would also have lower density 0, contradicting $\underline{d}(T) = 1$. Therefore there exists at least one SCC H with $\underline{d}(T_H) > 0$.

If among the SCCs with $\underline{d}(T_H) > 0$ there is an aperiodic one, we are in case (A). Otherwise every SCC visited with positive lower density along tight times is imprimitive, which is case (P). $\square$

Remark 2.6. *The remainder of the paper analyzes the geometry of $G^{(0)}$. Rigidity (projective contraction) is obtained from recurrence and Doeblin structure inside an aperiodic tight SCC. Obstructions are finite failures of the corresponding tight-core regularity conditions.*

3 Aubry–Tight Core and the Induced Boundary Subshift

This section pins down the precise meaning of *core* used throughout the paper. The definition is finite, canonical, and built from the normalized reduced costs. It isolates the locally optimal transitions and the induced symbolic system that supports calibrated minimizers.

3.1 Calibrated subactions and local optimality

Fix the reduced graph $G = (V, E)$ with costs $c : E \rightarrow \mathbb{R}$ and the extremal mean λ_* .

Definition 3.1 (Calibrated subaction). *A function $u : V \rightarrow \mathbb{R}$ is a calibrated subaction (for c at level λ_*) if for every edge $e \in E$,*

$$u(\text{head}(e)) \leq u(\text{tail}(e)) + c(e) - \lambda_*. \quad (1)$$

Equivalently, the reduced cost

$$c_u(e) := c(e) - \lambda_* + u(\text{tail}(e)) - u(\text{head}(e)) \quad (2)$$

satisfies $c_u(e) \geq 0$ for all $e \in E$.

Remark 3.2. *The potential ϕ produced by Theorem 2.1 is a calibrated subaction in the sense of Definition 3.1, with $c_\phi = c_u$. Conversely, any calibrated subaction yields a valid canonical normalization.*

Lemma 3.3 (Tight edges are exactly locally optimal edges). *Let u be any calibrated subaction. Then an edge $e \in E$ is tight (i.e. $c_u(e) = 0$) if and only if equality holds in (1):*

$$u(\text{head}(e)) = u(\text{tail}(e)) + c(e) - \lambda_*.$$

In particular, the tight edge set $E^{(0)} = \{e : c_u(e) = 0\}$ is exactly the set of one-step transitions that are locally optimal with respect to the calibrated subaction.

Proof. By definition, $c_u(e) = c(e) - \lambda_* + u(\text{tail}(e)) - u(\text{head}(e))$. The inequality (1) is equivalent to $c_u(e) \geq 0$. Thus $c_u(e) = 0$ holds exactly when (1) holds with equality. $\square$

3.2 Aubry–tight core

Fix a calibrated subaction u (for example $u = \phi$). Let $G^{(0)} = (V^{(0)}, E^{(0)})$ denote the tight subgraph, i.e. the subgraph of locally optimal edges.

Definition 3.4 (Aubry–tight core). *The Aubry–tight core (or simply the core) is the union of the strongly connected components of $G^{(0)}$ that contain at least one directed cycle. Equivalently, it is the set of vertices that lie on some tight directed cycle. We denote this vertex set by $\mathcal{A} \subseteq V^{(0)}$ and write $G^{\mathcal{A}}$ for the induced subgraph of $G^{(0)}$ on $\mathcal{A}$.*

Remark 3.5. *Throughout the paper, when we refer to a tight core, core SCC, or aperiodic core, we mean an SCC of $G^{\mathcal{A}}$ as in Definition 3.4. No statement later relies on any stronger “eventual confinement” property for arbitrary mean-minimizers.*

3.3 Induced boundary subshift

The tight core defines a finite-type symbolic system capturing calibrated two-sided minimizers.

Definition 3.6 (Boundary subshift induced by the tight core). *Let $\Sigma_{\mathcal{A}} \subset (E^{(0)})^{\mathbb{Z}}$ be the set of bi-infinite edge sequences $\omega = (\dots, e_{-1}, e_0, e_1, \dots)$ such that consecutive edges match and every vertex visited lies in $\mathcal{A}$. Equivalently, $\Sigma_{\mathcal{A}}$ is the two-sided edge shift of the finite directed graph $G^{\mathcal{A}}$. We call $(\Sigma_{\mathcal{A}}, \sigma)$ the boundary subshift induced by the Aubry–tight core.*

Lemma 3.7 (Tight paths have extremal mean). *Let $(e_0, \dots, e_{n-1})$ be a length- n path consisting only of tight edges in $E^{(0)}$. Then*

$$\frac{1}{n} \sum_{t=0}^{n-1} c(e_t) = \lambda_* + \frac{u(v_n) - u(v_0)}{n},$$

where v_0 is the initial vertex, v_n is the terminal vertex, and u is any calibrated subaction. In particular, along any one-sided infinite tight path, the asymptotic average cost equals λ_* .

Proof. For each tight edge $e_t : v_t \rightarrow v_{t+1}$ we have $c_u(e_t) = 0$, i.e. $c(e_t) - \lambda_* + u(v_t) - u(v_{t+1}) = 0$. Summing from $t = 0$ to $n - 1$ yields $\sum_{t=0}^{n-1} c(e_t) = n\lambda_* + u(v_n) - u(v_0)$. Divide by n . $\square$

Remark 3.8. *The boundary subshift $\Sigma_{\mathcal{A}}$ is the natural symbolic carrier of calibrated (two-sided) minimizers. The gap-free boundary classification later in the paper is performed SCC-by-SCC on this core subshift. When working with a specific one-sided minimizing trajectory, the relevant input is the set of tight times (Proposition 2.4) and the positive-density core selection (Proposition 2.5); no stronger trapping statement is assumed.*

4 Calibrated Minimizers and Selection Principles

This section isolates the precise sense in which one may *restrict* attention to calibrated (locally optimal) minimizers, while keeping clear what this does *not* imply for arbitrary one-sided mean-minimizers. The main point is that calibration is an *edgewise* (one-step) equality condition, so calibrated minimizers live entirely on the tight subgraph, whereas general minimizers are only asymptotically tight.

4.1 Calibrated minimizers

Fix a calibrated subaction $u : V \rightarrow \mathbb{R}$ in the sense of Definition 3.1, and write c_u for the associated reduced costs (2).

Definition 4.1 (Calibrated bi-infinite minimizer). *A bi-infinite admissible edge sequence $\omega = (\dots, e_{-1}, e_0, e_1, \dots) \in E^{\mathbb{Z}}$ is a calibrated minimizer for u if consecutive edges match and*

$$c_u(e_t) = 0 \quad \text{for all } t \in \mathbb{Z}. \tag{3}$$

Equivalently, ω is a bi-infinite path in the tight subgraph $G^{(0)} = (V^{(0)}, E^{(0)})$ defined by u .

Lemma 4.2 (Calibrated minimizers are supported on tight edges). *If ω is a calibrated minimizer for u , then every edge of ω lies in $E^{(0)} = \{e \in E : c_u(e) = 0\}$. In particular, every calibrated minimizer is minimizing with asymptotic mean cost λ_* .*

Proof. The first claim is immediate from (3). For the mean cost, sum the identity $c(e_t) - \lambda_* + u(v_t) - u(v_{t+1}) = 0$ along any finite window, exactly as in Lemma 3.7, and divide by the window length. $\square$

4.2 Existence on the Aubry–tight core

The Aubry–tight core $\mathcal{A}$ from Definition 3.4 is designed so that tight cycles exist through every vertex of $\mathcal{A}$. This yields calibrated minimizers without any recurrence hypothesis.

Proposition 4.3 (Periodic calibrated minimizers on the core). *For every vertex $v \in \mathcal{A}$ there exists a periodic bi-infinite calibrated minimizer $\omega \in \Sigma_{\mathcal{A}}$ that visits v . Consequently, $\Sigma_{\mathcal{A}}$ is nonempty whenever $\mathcal{A} \neq \emptyset$.*

Proof. By Definition 3.4, the vertex v lies on a directed tight cycle γ in $G^{(0)}$. Let w be the finite edge word given by γ starting and ending at v . Repeating w periodically in both time directions produces a bi-infinite admissible edge sequence $\omega \in (E^{(0)})^{\mathbb{Z}}$. Then $c_u(e_t) = 0$ for all t , so ω is calibrated by Definition 4.1. $\square$

4.3 Minimizing measures are tight

The previous subsection constructs explicit calibrated minimizers on $\mathcal{A}$. For applications, it is also useful to know that every *invariant* minimizer is supported on the tight shift.

Proposition 4.4 (Invariant minimizers live on the tight shift). *Let μ be a σ -invariant probability measure on the two-sided edge shift of G . If μ minimizes the average cost, i.e.*

$$\int c \, d\mu = \lambda_*,$$

then μ is supported on the tight edge shift $(E^{(0)})^{\mathbb{Z}}$. In particular, $c_u(e) = 0$ for μ -almost every edge occurrence.

Proof. By Definition 3.1, the reduced cost $c_u(e) \geq 0$ for all edges. For a bi-infinite path ω with vertex sequence $(v_t)_{t \in \mathbb{Z}}$ we have

$$c_u(e_t) = c(e_t) - \lambda_* + u(v_t) - u(v_{t+1}).$$

Integrating with respect to a σ -invariant measure μ yields

$$\int c_u \, d\mu = \int (c - \lambda_*) \, d\mu + \int (u \circ \text{tail} - u \circ \text{head}) \, d\mu.$$

The last integral vanishes by σ -invariance (it is the integral of a coboundary), hence

$$\int c_u \, d\mu = \int c \, d\mu - \lambda_*.$$

If $\int c \, d\mu = \lambda_*$, then $\int c_u \, d\mu = 0$. Since $c_u \geq 0$, it follows that $c_u = 0$ μ -almost surely. Equivalently, μ is supported on the set of paths using only edges in $E^{(0)}$. $\square$

Corollary 4.5 (Calibrated representatives exist in minimizing supports). *If μ is a σ -invariant minimizing measure, then μ -almost every ω is a calibrated minimizer for u . In particular, the support of any minimizing measure is contained in the boundary subshift $\Sigma_{\mathcal{A}}$.*

Proof. The first statement is immediate from Proposition 4.4. For the second, note that any bi-infinite tight path must lie in an SCC of $G^{(0)}$ that contains a cycle. By Definition 3.4, the union of such SCCs is exactly the Aubry–tight core $\mathcal{A}$, so every bi-infinite tight path lies in $\Sigma_{\mathcal{A}}$. $\square$

4.4 What calibration does and does not buy for one-sided minimizers

The results above justify a common proof strategy: when proving statements about the *symbolic carrier* of extremality (the boundary subshift and its SCCs), one may work entirely on calibrated minimizers. However, one must not confuse this with a statement about a *given* one-sided minimizing trajectory.

Proposition 4.6 (Calibrated selection from a one-sided minimizer). *Let $\pi = (e_0, e_1, \dots)$ be a one-sided minimizing path. Then there exists a bi-infinite calibrated minimizer ω and a sequence of times $n_k \rightarrow \infty$ such that $\sigma^{n_k} \pi \rightarrow \omega$ in the product topology on $E^{\mathbb{Z}}$. Equivalently, the ω -limit set of π contains a calibrated minimizer.*

Proof. By Proposition 2.4, the set $T = \{t \geq 0 : e_t \in E^{(0)}\}$ has asymptotic density 1. Fix an integer $m \geq 1$. Because T has density 1, there exist arbitrarily large times n such that the window $\{n - m, \dots, n + m\}$ is contained in T . Choose $n(m)$ with this property. The shift space $E^{\mathbb{Z}}$ is compact, so by a diagonal subsequence argument there exists a sequence $n_k \rightarrow \infty$ with $\sigma^{n_k} \pi$ converging to some bi-infinite admissible limit path ω . By construction of the windows, for each fixed coordinate $t \in \mathbb{Z}$, the edge at position t in $\sigma^{n_k} \pi$ belongs to $E^{(0)}$ for all sufficiently large k ; hence the limiting edge $e_t(\omega)$ also lies in $E^{(0)}$. Therefore ω uses only tight edges and is a calibrated minimizer by Definition 4.1. $\square$

Remark 4.7 (Scope and limitations). *Proposition 4.6 is a selection statement: from a minimizing trajectory one can extract calibrated minimizers as limit points. It does not imply that the original trajectory is eventually tight, trapped in a single tight SCC, or recurrent. The only trajectory-level facts used later are*

- *density-1 tightness (Proposition 2.4), and*
- *positive-lower-density selection of at least one tight SCC (Proposition 2.5).*

Any stronger recurrence input (bounded gaps of a certified Doeblin word, or a syndetic modulus from an automaton) is treated as a separate layer.

5 Minimality Route: what can and cannot be forced on the tight core

This section clarifies the “minimality” upgrade route for the minimizing dynamics on the tight core. The conclusion is sharp and finite-witness: because the boundary carrier $\Sigma_{\mathcal{A}}$ is a shift of finite type, true topological minimality (and hence linear recurrence) occurs only in the purely periodic case. Accordingly, minimality is not an automatic consequence of calibration or tightness; the correct replacement for gap-control is the *finite* recurrence certificate layer (RC1) and the slack-forcing route developed later.

5.1 Minimal SFTs are periodic

We record the elementary structural fact that blocks the naive minimality upgrade on SFT carriers.

Definition 5.1 (Minimality). *A two-sided subshift (X, σ) is minimal if every orbit is dense in X . Equivalently, X has no proper nonempty closed σ -invariant subsets.*

Lemma 5.2 (A minimal edge shift is a single cycle). *Let H be a finite directed graph and let X_H be its two-sided edge shift. If (X_H, σ) is minimal, then X_H consists of a single periodic orbit. Equivalently, H is a directed cycle (every vertex has in-degree and out-degree equal to 1 in the active component).*

Proof. If X_H is nonempty, then H contains a directed cycle, hence X_H contains a periodic point ω obtained by repeating that cycle. The orbit closure of a periodic point is finite. If X_H were minimal, the orbit closure of ω would have to equal all of X_H . Therefore X_H is finite. An edge shift over a finite graph is finite if and only if the underlying active component is a directed cycle, because any branching (a vertex with two distinct outgoing edges or two distinct incoming edges) produces at least two distinct infinite paths. Thus H must be a directed cycle and X_H is a single periodic orbit. $\square$

5.2 Implication for the minimizing carrier

Recall that the boundary carrier Σ_A defined in Definition 3.6 is the two-sided edge shift of the finite graph G^A . Hence it is a shift of finite type.

Proposition 5.3 (Minimality on Σ_A is a periodic obstruction). *If the boundary carrier (Σ_A, σ) is minimal, then every SCC of G^A is a directed cycle. In particular, the only minimal regime for the tight-core carrier is the purely periodic regime.*

Conversely, if a tight SCC $H \subseteq G^A$ is a directed cycle, then its edge shift is minimal and every tight trajectory supported in H is periodic.

Proof. Apply Lemma 5.2 to each nonempty irreducible component. If Σ_A is minimal and nonempty, it cannot decompose into multiple invariant components, so it coincides with the shift of a single SCC. That SCC must be a directed cycle, and the conclusions follow. The converse direction is immediate for a directed cycle. $\square$

5.3 The exact missing property and the correct replacement

The previous proposition isolates the precise missing property: without an explicit *finite* obstruction witness forcing the tight SCC to be a directed cycle, minimality (and linear recurrence) cannot be deduced from calibration alone. In particular, tightness is compatible with:

- multiple tight cycles in the same tight SCC,
- branching tight choices (out-degree ≥ 2 along tight edges),
- aperiodic tight SCCs that are topologically transitive but far from minimal.

These shapes are not pathologies; they occur naturally and are consistent with the obstruction taxonomy OG1–OG3. Therefore, the minimality route is not the right vehicle for gap-free recurrence on minimizers.

Remark 5.4 (Finite recurrence certificate as the correct “minimality substitute”). *What is actually used in the contraction upgrade is not minimality of the entire tight SCC, but a finite return-control statement for a specific certified Doeblin word w . This is exactly the role of the recurrence certificate layer (RC1): for a fixed tight SCC H and a candidate word w supported in H , one either*

- *produces an explicit bounded-gap modulus L_{syn} guaranteeing syndetic returns of w along every infinite tight path in H , or*

- returns a finite avoidance-cycle witness showing that some periodic tight path avoids w entirely.

This finite dichotomy is the sharp replacement for minimality in the gap-free regime.

6 Slack-Forcing from Doeblin-Word Avoidance

This section isolates a quantitative principle that will be used later in the forcing dichotomy: once a Doeblin word is certified on a tight aperiodic SCC and the bounded-gap recurrence certificate (RC1) holds for tight trajectories, then any attempt to avoid that word over long time windows necessarily incurs a *linear* amount of reduced-cost slack.

6.1 Setup and constants

Fix the calibrated potential ϕ and reduced costs $c_\phi(e) \geq 0$ as in (2). Recall the tight edge set $E^{(0)} = \{e : c_\phi(e) = 0\}$ and the tight graph $G^{(0)} = (V^{(0)}, E^{(0)})$.

Let

$$\alpha := \min\{c_\phi(e) : e \in E \setminus E^{(0)}\}, \quad (4)$$

with the convention $\alpha = +\infty$ if $E^{(0)} = E$. When $E^{(0)} \neq E$, we have $\alpha > 0$ by finiteness of E . This constant already appears in Proposition 2.4.

Definition 6.1 (Slack count and slack mass on a window). *For a finite edge-word $\gamma = (e_0, \dots, e_{L-1})$, define*

$$\text{SlackCount}(\gamma) := |\{0 \leq t < L : e_t \notin E^{(0)}\}|, \quad \text{SlackMass}(\gamma) := \sum_{t=0}^{L-1} c_\phi(e_t).$$

By construction, $\text{SlackMass}(\gamma) \geq \alpha \cdot \text{SlackCount}(\gamma)$ whenever $\alpha < \infty$.

6.2 RC1 turns word-avoidance into a bound on tight-run lengths

Let H be a tight SCC of $G^{(0)}$ and let w be a finite tight edge-word supported in H . Assume that the recurrence certificate **(RC1)** holds for (H, w) in the sense of Definition 11.1. Let $L_{\text{syn}}(H, w)$ denote the explicit syndetic return modulus from Lemma 11.4. Set

$$L_0 := L_{\text{syn}}(H, w) + |w|. \quad (5)$$

Lemma 6.2 (No long tight runs in H can avoid w). *Assume **(RC1)** holds for (H, w) and let L_0 be as in (5). If $\gamma = (e_0, \dots, e_{L-1})$ is a finite path that stays inside H and uses only tight edges $e_t \in E^{(0)}$, then γ contains an occurrence of w provided $L \geq L_0$. Equivalently, any tight path segment inside H with no occurrence of w has length at most $L_0 - 1$.*

Proof. Because H is strongly connected and contains a directed cycle, any finite tight path segment inside H can be extended forward to an infinite tight path in H . Apply Lemma 11.4 to that infinite tight extension: it contains occurrences of w with gaps between *starting indices* bounded by $L_{\text{syn}}(H, w)$. In particular, there exists an occurrence of w whose start index is at most $L_{\text{syn}}(H, w)$. Since an occurrence of w occupies $|w|$ consecutive edges, the entire occurrence lies within the first $L_{\text{syn}}(H, w) + |w| = L_0$ edges. Thus any tight segment of length $L \geq L_0$ must contain an occurrence of w . $\square$

6.3 Linear slack forcing on word-avoiding windows

We now turn Lemma 6.2 into a quantitative lower bound on slack.

Proposition 6.3 (Avoiding w for length L forces linear slack). *Let H be a tight SCC of $G^{(0)}$ and let w be a tight word supported in H such that **(RC1)** holds for (H, w) . Let L_0 be as in (5). Consider any finite path segment $\gamma = (e_0, \dots, e_{L-1})$ that stays inside H and contains no occurrence of w . Then*

$$\text{SlackCount}(\gamma) \geq \max\left\{0, \left\lceil \frac{L - L_0}{L_0 + 1} \right\rceil\right\}, \quad (6)$$

and consequently, if $\alpha < \infty$,

$$\text{SlackMass}(\gamma) \geq \alpha \cdot \max\left\{0, \left\lceil \frac{L - L_0}{L_0 + 1} \right\rceil\right\}. \quad (7)$$

In particular, $\text{SlackMass}(\gamma)$ grows at least linearly in L once $L > L_0$.

Proof. If γ contains no tight edges, then $\text{SlackCount}(\gamma) = L$ and (6) holds. Assume γ contains at least one tight edge. Decompose γ into maximal consecutive blocks of tight edges (tight runs) separated by slack edges. Let k be the number of tight runs and let $s = \text{SlackCount}(\gamma)$ be the number of slack edges. Then necessarily $k \leq s + 1$.

By Lemma 6.2, every tight run inside H that avoids w has length at most $L_0 - 1$. Hence the total number of tight edges in γ is at most $k(L_0 - 1)$. Therefore

$$L = (\text{number of tight edges}) + s \leq k(L_0 - 1) + s \leq (s + 1)(L_0 - 1) + s = s(L_0 + 1) + (L_0 - 1).$$

Rearranging gives $s \geq (L - L_0)/(L_0 + 1)$. Since s is an integer and $s \geq 0$, we obtain (6). Finally, (7) follows from $\text{SlackMass}(\gamma) \geq \alpha \text{SlackCount}(\gamma)$. $\square$

Remark 6.4 (What this does and does not say). *Proposition 6.3 is a local forcing statement: it applies to windows that stay inside a fixed tight SCC H where **(RC1)** holds for a chosen word w . If **(RC1)** fails, Lemma 11.3 provides a tight periodic witness that avoids w with zero slack, showing that no positive lower bound of the form (7) can hold in that case. Excursions that leave H and later return are handled separately in Section 7.*

7 Return Cost for Excursions Out of a Tight SCC

This section isolates a second forcing mechanism that complements Section 6: even if a trajectory attempts to avoid a certified Doeblin word by *leaving* the relevant tight SCC, any later *return* to that SCC forces reduced-cost slack. This turns repeated excursions (a natural avoidance strategy) into a measurable slack burden.

7.1 A structural fact: tight SCCs do not admit tight returns from outside

Fix the calibrated potential ϕ and reduced costs c_ϕ . Let $G^{(0)}$ be the tight graph. Let H be an SCC of $G^{(0)}$.

Definition 7.1 (Excursion and return). *Let $\pi = (e_0, e_1, \dots)$ be a one-sided path in the full graph G with vertex sequence $(v_t)_{t \geq 0}$, where $e_t : v_t \rightarrow v_{t+1}$. A time $t \geq 1$ is called a return time to H if $v_t \in H$ and $v_{t-1} \notin H$. An excursion is a maximal time interval during which $v_t \notin H$ following a departure from H and ending at the next return time (if it exists).*

Lemma 7.2 (Returning to a tight SCC forces slack). *Let $\gamma = (e_0, \dots, e_{L-1})$ be a finite path segment in G with vertex sequence $(v_0, \dots, v_L)$. Assume $v_0 \in H$ and $v_L \in H$, and assume that $v_t \notin H$ for at least one intermediate time t with $0 < t < L$. Then γ contains at least one non-tight edge, i.e. $\text{SlackCount}(\gamma) \geq 1$. In particular, if $\alpha < \infty$ (as in (4)), then $\text{SlackMass}(\gamma) \geq \alpha$.*

Proof. Suppose for contradiction that every edge of γ is tight, so γ is a path in the tight graph $G^{(0)}$. Then in $G^{(0)}$ there exists a tight path from H to a vertex outside H (since some $v_t \notin H$ occurs) and a tight path back from outside to H (since $v_L \in H$). This implies that in the SCC condensation DAG of $G^{(0)}$ there is a directed path from the SCC H to a different SCC and a directed path back to H , creating a directed cycle in the condensation. But the SCC condensation is a DAG, so such a cycle is impossible. Therefore γ must contain a non-tight edge, proving $\text{SlackCount}(\gamma) \geq 1$. The bound on SlackMass follows from $\text{SlackMass}(\gamma) \geq \alpha \text{SlackCount}(\gamma)$. $\square$

7.2 Quantifying slack per return

Proposition 7.3 (Slack lower bound per number of returns). *Let $\pi = (e_0, e_1, \dots)$ be any one-sided path in G and let $R_n(H)$ be the number of return times to H among $\{1, \dots, n\}$ (Definition 7.1). Then for every $n \geq 1$,*

$$R_n(H) \leq \sum_{t=0}^{n-1} \mathbf{1}_{\{e_t \notin E^{(0)}\}}, \quad (8)$$

and consequently, if $\alpha < \infty$,

$$\sum_{t=0}^{n-1} c_\phi(e_t) \geq \alpha \cdot R_n(H). \quad (9)$$

Proof. Each return time to H completes an excursion that starts in H , leaves H , and ends in H . Apply Lemma 7.2 to the finite path segment spanning that excursion. Different excursions are edge-disjoint in time, so each return time accounts for at least one distinct non-tight edge. This yields (8), and (9) follows from (4). $\square$

7.3 Connection to word-avoidance strategies

Let H be a tight SCC and let w be a tight word supported in H . If **(RC1)** holds for (H, w) , then Section 6 shows that long word-avoidance windows *inside* H force linear slack. Proposition 7.3 shows that word-avoidance by repeated *excursions* out of H also forces slack.

Remark 7.4 (Excursions as avoidance). *Assume **(RC1)** holds for (H, w) and let L_0 be as in (5). If a path visits H frequently but attempts to avoid w , then it must do at least one of the following on most long windows:*

- *introduce slack edges while staying in H , which is quantitatively penalized by Proposition 6.3;*
- *leave H before a tight run of length L_0 can occur and later return, which is penalized by Proposition 7.3.*

This is the mechanism used in the forcing dichotomy that follows: either w recurs with bounded gaps on the relevant tight core, or slack density is bounded below by a positive constant (contradicting asymptotic tightness for mean-minimizers).

8 Forcing Dichotomy: Doeblin Frequency versus Slack Density

This section upgrades the qualitative “tight-density” information available for general minimizers into a quantitative forcing statement. It combines:

- the **RC1** avoidance certificate for a word w inside a fixed tight aperiodic SCC H (Definition 11.1), and
- the **return-cost** principle for excursions out of H (Proposition 7.3).

The output is a dichotomy that is strong enough to feed the Doeblin $\Rightarrow$ Hilbert-contraction chain: either the Doeblin word occurs with *uniform positive lower density* along the trajectory, or the reduced-cost slack has *positive lower density* (hence the trajectory cannot be minimizing).

8.1 Setup

Fix the calibrated potential ϕ and reduced costs $c_\phi(e) \geq 0$ as in (2). Let $E^{(0)} = \{e \in E : c_\phi(e) = 0\}$ be the tight edge set and let $G^{(0)} = (V^{(0)}, E^{(0)})$ be the tight graph.

Fix a tight SCC H of $G^{(0)}$ and a finite edge-word w supported in H such that **(RC1)** holds for (H, w) . Let $L_{\text{syn}}(H, w)$ be the explicit syndetic modulus from Lemma 11.4 and set

$$L_0 := L_{\text{syn}}(H, w) + |w| \quad (10)$$

as in (5). Let

$$\alpha := \min\{c_\phi(e) : e \in E \setminus E^{(0)}\} \in (0, \infty] \quad (11)$$

as in (4).

For a one-sided path $\pi = (e_0, e_1, \dots)$ in G define the length- n slack mass

$$\text{SlackMass}_n(\pi) := \sum_{t=0}^{n-1} c_\phi(e_t). \quad (12)$$

Define the tight time count in H by

$$T_H(n; \pi) := \sum_{t=0}^{n-1} \mathbf{1}_{\{e_t \in E^{(0)} \text{ and } e_t \text{ is supported in } H\}}, \quad (13)$$

and let $R_n(H)$ be the number of return times to H among $\{1, \dots, n\}$ as in Definition 7.1.

Finally, let $N_w(n; \pi)$ denote the number of starting indices $0 \leq i \leq n-1$ at which the word w occurs along π (i.e. $e_i e_{i+1} \cdots e_{i+|w|-1} = w$).

8.2 A counting lemma on maximal tight runs

Definition 8.1 (Maximal tight-in- H runs). *For fixed n , consider the set of indices $t \in \{0, \dots, n-1\}$ for which e_t is tight and supported in H . The maximal contiguous intervals of such indices determine disjoint finite subpaths*

$$\gamma^{(1)}, \dots, \gamma^{(m)}$$

which we call the maximal tight-in- H runs of π up to time n . Write $\ell_j := |\gamma^{(j)}|$ for the length (number of edges) of the j -th run. Then $\sum_{j=1}^m \ell_j = T_H(n; \pi)$ and $m-1 \leq R_n(H)$.

Lemma 8.2 (Each long tight run forces many occurrences of w). *Assume **(RC1)** holds for (H, w) and let L_0 be as in (10). Let γ be a finite tight-in- H run of length ℓ . Then γ contains at least $\lfloor \ell/L_0 \rfloor$ occurrences of w .*

Proof. Partition γ into consecutive disjoint blocks of length L_0 , discarding at most a final remainder block of length $< L_0$. Each full block is a tight path in H of length L_0 . By Lemma 6.2, no such block can avoid w , so each full block contains an occurrence of w supported entirely inside that block. Disjoint blocks cannot share an occurrence, hence the number of occurrences is at least the number of full blocks, which is $\lfloor \ell/L_0 \rfloor$. $\square$

Lemma 8.3 (Occurrences versus returns). *For every $n \geq 1$ and every path π ,*

$$N_w(n; \pi) \geq \frac{T_H(n; \pi)}{L_0} - R_n(H) - 1. \quad (14)$$

Proof. Let $\gamma^{(1)}, \dots, \gamma^{(m)}$ be the maximal tight-in- H runs up to time n (Definition 8.1) with lengths $\ell_1, \dots, \ell_m$. By Lemma 8.2,

$$N_w(n; \pi) \geq \sum_{j=1}^m \left\lfloor \frac{\ell_j}{L_0} \right\rfloor \geq \sum_{j=1}^m \left(\frac{\ell_j}{L_0} - 1 \right) = \frac{T_H(n; \pi)}{L_0} - m.$$

Since $m \leq R_n(H) + 1$, we obtain (14). $\square$

8.3 The forcing dichotomy

Proposition 8.4 (Doebelin frequency vs slack density). *Assume **(RC1)** holds for (H, w) and let L_0 and α be as in (10)–(11). Then for every $n \geq 1$ and every one-sided path π in G ,*

$$N_w(n; \pi) \geq \frac{T_H(n; \pi)}{L_0} - \frac{\text{SlackMass}_n(\pi)}{\alpha} - 1, \quad (15)$$

with the convention that $\text{SlackMass}_n(\pi)/\alpha = 0$ if $\alpha = \infty$. Equivalently,

$$\text{SlackMass}_n(\pi) \geq \alpha \cdot \left(\frac{T_H(n; \pi)}{L_0} - N_w(n; \pi) - 1 \right)_+. \quad (16)$$

Proof. Combine Lemma 8.3 with the slack-per-return bound (9) from Proposition 7.3, which yields $R_n(H) \leq \text{SlackMass}_n(\pi)/\alpha$ (for $\alpha < \infty$). If $\alpha = \infty$, then $E \setminus E^{(0)} = \emptyset$ and $\text{SlackMass}_n(\pi) = 0$ identically, so (15) reduces to (14). $\square$

Corollary 8.5 (Forced positive density of the Doebelin word). *Assume **(RC1)** holds for (H, w) and let L_0 be as in (10). Let π be any path such that*

$$\liminf_{n \rightarrow \infty} \frac{T_H(n; \pi)}{n} \geq \theta > 0 \quad \text{and} \quad \lim_{n \rightarrow \infty} \frac{\text{SlackMass}_n(\pi)}{n} = 0.$$

Then

$$\liminf_{n \rightarrow \infty} \frac{N_w(n; \pi)}{n} \geq \frac{\theta}{L_0}.$$

In particular, w occurs along π with a uniform positive lower density depending only on the forcing constants.

Proof. Divide (15) by n and take $\liminf_{n \rightarrow \infty}$, using the hypotheses. $\square$

Remark 8.6 (Interpretation). *Corollary 8.5 is the promised forcing upgrade: in the regime where a minimizer has nontrivial tight mass in a fixed aperiodic tight SCC H and reduced-cost slack density 0, **(RC1)** forces a quantitative supply of Doeblin blocks along the trajectory. This is sufficient for explicit exponential projective contraction (see the rate pack in Section 14), without needing global bounded-gap recurrence.*

9 Descriptor Faces as Finite Certificates (Eliminating Interface Assumptions)

Two interface notions recur throughout the gap-free theory on a tight SCC H : *active-face invariance* and *support-connectivity on the active face*. In this section we turn both into purely *finite, checkable certificates* extracted directly from the reduced descriptor graph and template sparsity.

9.1 Canonical active face per descriptor state

In the descriptor reduction, each vertex $v \in V$ corresponds to a finite descriptor encoding which coordinates are *potentially active* and which are structurally zero under admissible propagation. Accordingly, each descriptor vertex carries a canonical support set

$$S(v) \subseteq \{1, \dots, d\}.$$

Definition 9.1 (Face map and canonical active set). *A face map for the reduced graph is a function $S : V \rightarrow 2^{\{1, \dots, d\}}$ such that for each edge $e : u \rightarrow v$ labeled by a template (or fixed-length block) T_e we have:*

- (i) (**Forward invariance**) $T_e(\mathbb{R}_{>0}^{S(u)}) \subseteq \mathbb{R}_{>0}^{S(v)}$;
- (ii) (**No spurious activation**) If $x \in \mathbb{R}_{>0}^{S(u)}$, then $(T_e x)_i = 0$ for all $i \notin S(v)$.

Remark 9.2. *Definition 9.1 is checkable from template sparsity: it amounts to verifying that the support pattern of each T_e maps the indicator of $S(u)$ into $S(v)$.*

9.2 Face stability on SCCs as a finite fixed-point

Lemma 9.3 (SCC face stability by intersection closure). *Let H be an SCC of the reduced graph and let $S : V \rightarrow 2^{\{1, \dots, d\}}$ be a face map. Define the SCC face by*

$$S(H) := \bigcap_{v \in H} S(v).$$

Then for every edge $e : u \rightarrow v$ inside H ,

$$T_e(\mathbb{R}_{>0}^{S(H)}) \subseteq \mathbb{R}_{>0}^{S(H)}.$$

Proof. Since $S(H) \subseteq S(u)$ for each $u \in H$, we have $\mathbb{R}_{>0}^{S(H)} \subseteq \mathbb{R}_{>0}^{S(u)}$. By forward invariance, $T_e(\mathbb{R}_{>0}^{S(u)}) \subseteq \mathbb{R}_{>0}^{S(v)}$. Because $S(H) \subseteq S(v)$, restricting to coordinates in $S(H)$ yields invariance on $S(H)$. $\square$

9.3 Support-connectivity on the SCC face as a finite graph property

Fix an SCC H and its canonical face $S(H)$. For each generator edge e in H labeled by T_e , define the induced coordinate support digraph on $S(H)$: we draw an arrow $j \rightarrow i$ if $(T_e)_{ij} > 0$ for some e in H .

Definition 9.4 (Coordinate support digraph on a face). *Let $\Gamma(H)$ be the directed graph with vertex set $S(H)$ and an arrow $j \rightarrow i$ if there exists an edge e in H such that $(T_e)_{ij} > 0$.*

Lemma 9.5 (Support-connectivity certificate). *If $\Gamma(H)$ is strongly connected, then for every $i, j \in S(H)$ there exists an admissible word w in H such that the block product satisfies $(B(w))_{ij} > 0$.*

Proof. Strong connectivity gives a coordinate path $j \rightarrow \dots \rightarrow i$ realized by a sequence of generator matrices from edges of H . Concatenate the corresponding edges to obtain a word w whose product contains a positive contribution from j into i . $\square$

Corollary 9.6 (Doeblin word from aperiodic SCC via Boolean-pattern product automaton). *Let H be an aperiodic SCC of the tight graph $G^{(0)}$ and assume a consistent face map $S(\cdot)$ is available, with active face $S(H) \neq \emptyset$. Then the following are equivalent:*

- (i) *There exists a finite word w supported in H such that the block product $B(w)$ is strictly positive on $S(H) \times S(H)$ (equivalently, w is Doeblin on the face $S(H)$).*
- (ii) *The Boolean-pattern product-automaton test (Algorithm 3) reaches a return state $(v_0, \mathbf{1})$ for some base vertex $v_0 \in H$, where $\mathbf{1}$ denotes the all-ones pattern on $S(H) \times S(H)$; in this case the algorithm returns an explicit witness word w .*

In case (i), strict positivity of $B(w)_{S(H) \times S(H)}$ implies that $\Gamma(H)$ contains every directed edge on $S(H)$, hence is strongly connected. Thus strong connectivity of $\Gamma(H)$ is a fast necessary check. However, strong connectivity of $\Gamma(H)$ alone does not imply the existence of a Doeblin word.

Remark 9.7 (Strong connectivity is necessary but not sufficient). *Strong connectivity of $\Gamma(H)$ is a fast necessary test for the existence of a Doeblin word, but it is not sufficient. A clean counter-shape is the permutation case: if all allowed face-restricted patterns are permutation matrices generating a transitive action on $S(H)$, then $\Gamma(H)$ is strongly connected while every product is still a permutation matrix, so no word yields strict positivity on $S(H) \times S(H)$. The product-automaton test (Algorithm 3) detects this failure and produces a finite non-reachability certificate.*

9.4 Algorithmic extraction

9.5 Witness extraction when certificates fail

The obstruction predicates OG2/OG3 are designed to be not merely *named* failures but *witness-producing* ones. Here we record explicit finite witnesses that can be returned whenever the face-map or coordinate-connectivity checks fail.

Definition 9.8 (Support propagation operator). *For an edge $e : u \rightarrow v$ labeled by a nonnegative template/block T_e and a set $S \subseteq \{1, \dots, d\}$, define the propagated support*

$$\text{Prop}_e(S) := \{i : \exists j \in S \text{ with } (T_e)_{ij} > 0\}.$$

For a word $w = e_1 \dots e_L$ we set $\text{Prop}_w := \text{Prop}_{e_L} \circ \dots \circ \text{Prop}_{e_1}$.

Lemma 9.9 (OG2 witness: finite inconsistency cycle). *Let H be an SCC and suppose the proposed face sets $S(v)$ fail to form a face map on H . Then there exist vertices $u, v \in H$, an edge $e : u \rightarrow v$ in H , and indices $j \in S(u)$ and $i \notin S(v)$ such that $(T_e)_{ij} > 0$. Moreover, choosing any directed path p in H from v back to u and setting $w := ep$, we obtain a directed cycle word w in H together with a marked step (the first step e) at which a coordinate outside the designated face is activated. This pair $(w, (i, j))$ is a finite witness of OG2 on H .*

Proof. If $S(\cdot)$ is not a face map on H , then some edge $e : u \rightarrow v$ in H violates Definition 9.1. In particular, the “no spurious activation” condition (ii) fails, which means exactly that $\text{Prop}_e(S(u)) \not\subseteq S(v)$. Equivalently, there exist $j \in S(u)$ and $i \notin S(v)$ with $(T_e)_{ij} > 0$. Because H is strongly connected, there exists a directed path p from v to u , so $w := ep$ is a directed cycle word in H . The marked first step e certifies the activation of an outside coordinate along a cycle in H . $\square$

Lemma 9.10 (OG3 witness: cut decomposition of $\Gamma(H)$). *Let H be a tight SCC for which a face map exists and let $S(H)$ be the SCC face. If the coordinate digraph $\Gamma(H)$ is not strongly connected, then there exist nonempty disjoint sets $A, B \subseteq S(H)$ with $A \cup B = S(H)$ such that*

$$(T_e)_{ij} = 0 \quad \text{for every edge } e \text{ in } H, \text{ every } i \in A, \text{ and every } j \in B.$$

Equivalently, $\Gamma(H)$ has no directed edges from B to A . In particular, for every admissible word w in H and every $i \in A, j \in B$ we have $(B(w))_{ij} = 0$. Thus $(H, S(H), A, B)$ is a finite witness of OG3.

Proof. Compute the SCC decomposition of $\Gamma(H)$ and consider its condensation DAG. Since $\Gamma(H)$ is not strongly connected, this DAG has at least two nodes and hence has a sink SCC A . Let $B := S(H) \setminus A$. By the definition of a sink SCC, there is no directed edge in $\Gamma(H)$ from any vertex in B to any vertex in A . By Definition 9.4, an edge $j \rightarrow i$ in $\Gamma(H)$ exists exactly when some generator edge e in H has $(T_e)_{ij} > 0$. Therefore for all $i \in A$ and $j \in B$, and for all edges e in H , we have $(T_e)_{ij} = 0$. The final structural-zero claim for products follows by closure under multiplication of nonnegative matrices: if every generator has zeros on $A \times B$, then every product does as well. $\square$

Algorithm 1 Compute SCC face and coordinate support graph

Require: SCC H with face map $S(\cdot)$ and generator labels T_e

Ensure: $S(H)$ and coordinate digraph $\Gamma(H)$

- 1: $S(H) \leftarrow \bigcap_{v \in H} S(v)$
 - 2: Initialize $\Gamma(H)$ with vertex set $S(H)$
 - 3: **for all** edges e inside H **do**
 - 4: **for all** $i, j \in S(H)$ **do**
 - 5: **if** $(T_e)_{ij} > 0$ **then** add arrow $j \rightarrow i$
 - 6: **end if**
 - 7: **end for**
 - 8: **end for**
-

Algorithm 2 Check OG2–OG3 and produce Doeblin/recurrence witnesses

Require: Tight graph $G^{(0)}$, SCC decomposition, period and coordinate digraphs $\Gamma(H)$

Ensure: Either a finite obstruction witness (OG1/OG2/OG3), or a Doeblin witness w together with either a syndetic modulus L_{syn} (RC1) or an explicit avoidance-cycle witness

```
1: Compute SCCs of  $G^{(0)}$  and their periods  $\text{per}(H)$ 
2: for each aperiodic SCC  $H$  do
3:   Attempt to compute a consistent face map  $S(\cdot)$  on  $H$ 
4:   if inconsistent then
5:     return OG2 witness (inconsistency cycle)
6:   end if
7:   Build  $\Gamma(H)$  on  $S(H)$ 
8:   if  $\Gamma(H)$  is not strongly connected then
9:     return OG3 witness of type (cut) via Lemma 9.10
10:  end if
11:  Run the product-automaton Doeblin-word test (Algorithm 3)
12:  if Algorithm 3 returns a word  $w$  then
13:    Build the avoidance graph  $\mathcal{A}_{\neg w}(H)$  (Definition 11.2)
14:    if  $\mathcal{A}_{\neg w}(H)$  contains a directed cycle then
15:      return Doeblin witness  $w$  and an avoidance-cycle witness (Lemma 11.3)
16:    else
17:      Compute  $L_{\text{syn}} = 1 + \ell_{\max}$  where  $\ell_{\max}$  is the longest path length in  $\mathcal{A}_{\neg w}(H)$ 
18:      return  $(w, L_{\text{syn}})$  (RC1; Lemma 11.4)
19:    end if
20:  else
21:    return OG3 witness of type (product-automaton): a finite non-reachability certificate
    in the Boolean-pattern product automaton (no return state  $(v_0, \mathbf{1})$  is reachable)
22:  end if
23: end for
24: return OG1 witness (no aperiodic SCC exists in  $G^{(0)}$ )
```

10 Doeblin-word discovery by a finite product automaton

This section records the finite-state procedure used throughout the certificate pipeline to decide whether a given aperiodic tight SCC admits a Doeblin word on its active face. The input is purely combinatorial support information on the face, and the output is either a concrete witness word or a finite non-reachability certificate.

10.1 Boolean patterns and Boolean product

Fix a tight SCC H and a face $S = S(H) \subseteq \{1, \dots, d\}$. For each tight edge $e : u \rightarrow v$ in H , let $B_e \in \mathbb{R}_+^{d \times d}$ denote the corresponding nonnegative block (or fixed-length template). Define the face-restricted support pattern

$$P_e \in \{0, 1\}^{S \times S}, \quad (P_e)_{ij} = 1 \iff (B_e)_{ij} > 0 \text{ for } i, j \in S.$$

For Boolean patterns $P, Q \in \{0, 1\}^{S \times S}$, define their Boolean product $P \odot Q$ by

$$(P \odot Q)_{ij} = \bigvee_{k \in S} (P_{ik} \wedge Q_{kj}),$$

so that $P \odot Q$ records the support pattern of the matrix product of the underlying nonnegative blocks restricted to S .

10.2 The product automaton and BFS search

Let $\mathcal{P} := \{0, 1\}^{S \times S}$ be the finite set of Boolean patterns. The product automaton has state space $H \times \mathcal{P}$. A directed transition

$$(u, P) \xrightarrow{e: u \rightarrow v} (v, P_e \odot P)$$

records the effect of appending the edge e to the front of the word whose current pattern is P .

Algorithm 3 Doeblin-word discovery by finite product automaton

Require: Tight SCC H , active face $S(H)$, Boolean support patterns P_e on $S(H)$ for each tight edge e in H

Ensure: Either a word w whose face-restricted product is strictly positive, or a finite non-reachability certificate

```

1: Choose a base vertex  $v_0 \in H$ 
2: Let  $I$  be the identity pattern on  $S(H)$  and let  $\mathbf{1}$  denote the all-ones pattern on  $S(H) \times S(H)$ 
3: Initialize BFS on states  $(v, P)$  with the start state  $(v_0, I)$ 
4: while queue not empty do
5:   Pop  $(u, P)$ 
6:   for all tight edges  $e : u \rightarrow v$  in  $H$  do
7:      $P' \leftarrow P_e \odot P$ 
8:     if  $(v, P')$  not yet visited then
9:       Mark visited; record predecessor; enqueue  $(v, P')$ 
10:    end if
11:    if  $v = v_0$  and  $P' = \mathbf{1}$  then
12:      Reconstruct the witnessed word  $w$  from predecessors and return  $w$ 
13:    end if
14:  end for
15: end while
16: return finite non-reachability certificate:  $\mathbf{1}$  is not reachable at  $(v_0, \cdot)$ 

```

Remark 10.1 (Complexity). *The state space has size $|H| \cdot 2^{|S(H)|^2}$. This exponential dependence on the face size is unavoidable at the level of brute-force semigroup reachability, but in the intended regime the active faces are descriptor-controlled and small.*

11 Syndetic return to a certified Doeblin word via an avoidance graph

Fix a tight SCC H and a word $w = e_1 \cdots e_m$ on the edges of H . This section formalizes a finite, checkable recurrence condition for w . The outcome is dichotomous:

- either the avoidance dynamics contains a directed cycle, yielding an explicit witness that w can be avoided for arbitrarily long times,
- or the avoidance dynamics is acyclic, in which case w must occur with bounded gaps along every infinite trajectory in H .

The second case is the concrete recurrence input (RC1) used in the gap-free contraction statements.

Definition 11.1 (Finite recurrence input (RC1)). *Let H be a tight SCC and let w be an edge word in H . We say that (H, w) satisfies **(RC1)** if the avoidance graph $A(H, w)$ (Definition 11.2) is acyclic. In this case Algorithm 4 returns a finite modulus $L_{\text{syn}}(H, w)$ such that every edge word of length $L_{\text{syn}}(H, w)$ in H contains an occurrence of w . For notational compatibility with later sections, we also write $\mathcal{A}_{\neg w}(H) := A(H, w)$.*

11.1 Prefix automaton for a fixed word

Let $m := |w|$. For a finite edge word $a = e'_1 \cdots e'_k$ in H , define $\pi(a) \in \{0, 1, \dots, m-1\}$ to be the length of the longest suffix of a which is a proper prefix of w . Equivalently, $\pi(a) = q$ means that the last q edges of a match $e_1 \cdots e_q$, but a does not yet contain a full occurrence of w ending at the last edge.

Given $q \in \{0, \dots, m-1\}$ and an edge e of H , define the update $\text{next}(q, e) \in \{0, \dots, m\}$ to be the new matched-prefix length after appending e . If $\text{next}(q, e) = m$, then w occurs ending at that step. Otherwise $\text{next}(q, e) \in \{0, \dots, m-1\}$. This is the standard prefix automaton update (KMP-type) for a fixed pattern.

11.2 Avoidance graph

Definition 11.2 (Avoidance graph). *Let $A(H, w)$ be the directed graph on vertex set*

$$\{0, 1, \dots, m-1\} \times V(H).$$

There is a directed edge

$$(q, x) \rightarrow (q', y)$$

in $A(H, w)$ if and only if there exists an edge $e : x \rightarrow y$ of H such that $\text{next}(q, e) = q' \in \{0, \dots, m-1\}$. In other words, $A(H, w)$ encodes the evolution of the matched-prefix state along paths in H under the constraint that no step completes an occurrence of w .

Lemma 11.3 (Avoidance-cycle witness). *If $A(H, w)$ contains a directed cycle, then there exists an infinite trajectory in H which avoids w forever. Consequently, no bound on gaps between occurrences of w can hold uniformly over all infinite trajectories in H .*

Proof. A directed cycle in $A(H, w)$ corresponds to a nonempty edge word in H which returns to the same matched-prefix state without ever completing w . Repeating this cycle yields an infinite path in H for which the prefix-state never reaches m , hence w never occurs. $\square$

Lemma 11.4 (Syndetic modulus from acyclicity). *If $A(H, w)$ is acyclic, then there exists a finite integer $L_{\text{syn}}(H, w)$ such that every length- $L_{\text{syn}}(H, w)$ edge word in H contains an occurrence of w . In particular, every infinite trajectory in H contains occurrences of w with gaps at most $L_{\text{syn}}(H, w)$.*

Proof. If $A(H, w)$ is acyclic, it has finite longest directed path length. Along any edge word in H that avoids w , the corresponding lifted path in $A(H, w)$ never hits an accepting step ($\text{next} = m$), so it remains a directed path in the avoidance graph. Therefore any w -avoiding word has length at most the longest path length of $A(H, w)$. Taking $L_{\text{syn}}(H, w)$ to be one plus that maximal length forces every longer word to contain w . $\square$

Algorithm 4 Syndetic return check and modulus computation

Require: Tight SCC H , word w on edges of H of length m

Ensure: Either an avoidance-cycle witness, or a syndetic modulus $L_{\text{syn}}(H, w)$

- 1: Build the avoidance graph $A(H, w)$ (Definition 11.2)
 - 2: **if** $A(H, w)$ contains a directed cycle **then**
 - 3: Extract a directed cycle as a witness and **return** avoidance-cycle witness
 - 4: **else**
 - 5: Compute the length $\ell_{\max}$ of a longest directed path in $A(H, w)$ (DP on a topological ordering)
 - 6: **return** $L_{\text{syn}}(H, w) \leftarrow \ell_{\max} + 1$
 - 7: **end if**
-

12 Recurrence Obstruction: Doeblin-Avoidance Cycle (RC1 Failure)

The obstruction taxonomy OG1–OG3 controls the existence of an aperiodic tight SCC, a consistent active-face map, and a strictly positive Doeblin block on that active face. Even when none of these obstructions occurs, a logically correct gap-free contraction statement must still address one further issue: a certified Doeblin word need not recur along *every* tight trajectory unless its recurrence is forced by finite-state structure.

We therefore isolate a finite recurrence predicate (**RC1**) for a fixed SCC and a fixed Doeblin word, and we record its failure as a fourth obstruction class OG4 with an explicit finite witness. This keeps recurrence out of the hypotheses while preserving a complete, decidable boundary taxonomy.

12.1 Definition and finite witness schema

Definition 12.1 (OG4 / recurrence obstruction: Doeblin-avoidance cycle). *Let H be a tight SCC of $G^{(0)}$ and let w be a finite edge-word supported in H . We say that the obstruction $\mathbf{OG4}(H, w)$ holds if the avoidance graph $\mathcal{A}_{\neg w}(H)$ (Definition 11.2) contains a directed cycle. Equivalently, (**RC1**) fails for (H, w) (Definition 11.1).*

A finite witness for $\mathbf{OG4}(H, w)$ is simply a directed cycle in $\mathcal{A}_{\neg w}(H)$, which can be produced by standard cycle-detection on a finite directed graph.

12.2 Why OG4 is a genuine obstruction

Proposition 12.2 (OG4 produces a tight minimizing trajectory with no Doeblin blocks). *If $\mathbf{OG4}(H, w)$ holds, then there exists a periodic tight path π supported in H that avoids w entirely. In particular, no theorem can guarantee occurrence of w along all tight minimizers in H .*

Proof. By Definition 12.1, $\mathcal{A}_{\neg w}(H)$ contains a directed cycle. Lemma 11.3 converts such a cycle into an explicit periodic tight path in H that avoids w entirely. Since the path is tight, Lemma 3.7 shows that it has asymptotic mean cost λ_* , so it is a minimizing trajectory, yet it contains no occurrences of w . $\square$

Remark 12.3 (Relationship to OG1–OG3). *OG4 can occur even when OG1–OG3 do not: an aperiodic tight SCC may exist, the face map may be consistent, and a strictly positive Doeblin word w may be certified, but w may still be avoidable by some tight trajectories in H . Accordingly, OG4 is the correct finite obstruction to record whenever (**RC1**) is not satisfied for the chosen (H, w) .*

13 Contraction from a Certified Doeblin Word: the Finite Recurrence Branch

Once the finite obstruction taxonomy produces a certified Doeblin word on an aperiodic tight SCC, there is only one further way for “Doeblin $\Rightarrow$ contraction” to fail on *all* tight trajectories: the Doeblin word might be avoidable. This is exactly OG4, witnessed by a directed cycle in the avoidance graph. When OG4 does not occur, the Doeblin word is *unavoidable* on the induced subshift over the SCC, hence appears with bounded gaps on every tight path, and uniform projective contraction follows with fully explicit constants.

We also record a second, quantitative route: along a general minimizing path, bounded gaps need not hold globally because tightness holds only in density. Nevertheless, if the minimizer spends positive lower density of tight time in the aperiodic SCC, then unavoidable recurrence forces a uniform positive lower density of Doeblin blocks, which is enough for an explicit exponential contraction rate.

13.1 Hilbert metric and Birkhoff contraction

Definition 13.1 (Hilbert (projective) metric). *For $x, y \in \mathbb{R}_{>0}^S$, define*

$$d_H(x, y) := \log\left(\max_{i \in S} \frac{x_i}{y_i}\right) + \log\left(\max_{i \in S} \frac{y_i}{x_i}\right) = \log\left(\max_{i, j \in S} \frac{x_i y_j}{x_j y_i}\right).$$

This metric is invariant under positive scalar rescaling: $d_H(cx, dy) = d_H(x, y)$ for all $c, d > 0$.

Lemma 13.2 (Birkhoff contraction from entry bounds). *Let $A \in \mathbb{R}_{>0}^{S \times S}$ satisfy $\min_{i, j \in S} A_{ij} \geq \varepsilon$ and $\max_{i, j \in S} A_{ij} \leq U$. Set*

$$\Delta := 2 \log \frac{U}{\varepsilon}, \quad \tau := \tanh\left(\frac{\Delta}{4}\right) = \tanh\left(\frac{1}{2} \log \frac{U}{\varepsilon}\right) \in (0, 1).$$

Then for all $x, y \in \mathbb{R}_{>0}^S$,

$$d_H(Ax, Ay) \leq \tau d_H(x, y).$$

Remark 13.3. *The constant τ is the standard Birkhoff contraction coefficient corresponding to the projective diameter bound Δ induced by entry ratio control. The certificate pipeline will provide explicit values of (ε, U) for Doeblin words.*

Definition 13.4 (Recurrence input for a Doeblin word). *Fix an infinite tight path π in a tight SCC H and a Doeblin word w of length $|w|$ on H . Let $N_w(n; \pi)$ denote the number of occurrences of w starting among the first n edges of π . We say that π satisfies one of the following recurrence inputs.*

- **(RH1) Syndetic recurrence:** *there exists $L_0 \in \mathbb{N}$ such that every length- L_0 subword of π contains an occurrence of w .*
- **(RH2) Positive lower density:** *there exists $\rho > 0$ such that $\liminf_{n \rightarrow \infty} \frac{N_w(n; \pi)}{n} \geq \rho$.*

Corollary 13.5 (Exponential projective contraction from Doeblin blocks). *Let w be a Doeblin word on a face S whose face-restricted product $B(w)|_S$ has entry bounds (ε, U) . Let $\tau \in (0, 1)$ be the corresponding Birkhoff coefficient from Lemma 13.2. Then along any tight path π in H , the Hilbert distance between two positive face vectors transported by the tight product contracts exponentially provided π satisfies either recurrence input from Definition 13.4. More precisely:*

- Under **(RH1)** with modulus L_0 , one has $d_H(x_n, y_n) \leq \tau^{\lfloor n/L_0 \rfloor} d_H(x_0, y_0)$ and hence $d_H(x_n, y_n) \leq C e^{-\kappa n} d_H(x_0, y_0)$ with $\kappa := \frac{1}{L_0}(-\log \tau)$ and $C := \tau^{-1}$.
- Under **(RH2)** with density ρ , one has $d_H(x_n, y_n) \leq \tau^{N_w(n; \pi)} d_H(x_0, y_0)$ and hence $d_H(x_n, y_n) \leq C e^{-\kappa n} d_H(x_0, y_0)$ with $\kappa := \rho(-\log \tau)$ and $C := \tau^{-1}$.

13.2 A tight-path version (uniform on the induced subshift over the tight SCC)

Theorem 13.6 (Finite recurrence dichotomy and contraction on tight paths). *Fix an aperiodic tight SCC H of $G^{(0)}$ and assume OG2 and OG3 do not occur on H , so that a consistent active face $S(H)$ exists and an explicit Doeblin word w supported in H is certified with $B(w)$ strictly positive on $S(H) \times S(H)$.*

Then exactly one of the following holds.

(OG4) *The avoidance graph $\mathcal{A}_{\neg w}(H)$ contains a directed cycle. Equivalently, there exists a periodic tight path in H that avoids w entirely (Proposition 12.2).*

(UC) *The avoidance graph $\mathcal{A}_{\neg w}(H)$ is acyclic. Then every infinite tight path in H contains w with bounded gaps, with explicit gap modulus*

$$L_{\text{syn}} := L_{\text{syn}}(H, w) \leq |V(H)| \cdot |w|$$

(Lemma 11.4). *Consequently, for every tight path π supported in H and all $x, y \in \mathbb{R}_{>0}^{S(H)}$,*

$$d_H(A^{(n)}(\pi)x, A^{(n)}(\pi)y) \leq C e^{-\kappa n} d_H(x, y) \quad (n \geq 1),$$

with explicit constants

$$\begin{aligned} \varepsilon &:= m_*^{|w|}, & U &:= (dM_*)^{|w|}, \\ \tau &:= \tanh\left(\frac{1}{2} \log(U/\varepsilon)\right) \in (0, 1), & C &:= \tau^{-1}, \\ \kappa &:= \frac{-\log \tau}{L_{\text{syn}} + |w|}. \end{aligned}$$

Proof. If $\mathcal{A}_{\neg w}(H)$ contains a directed cycle, OG4 holds by Definition 12.1. If $\mathcal{A}_{\neg w}(H)$ is acyclic, Lemma 11.4 yields syndetic recurrence of w on every infinite path in H with gap bound L_{syn} . Since $B(w)$ is strictly positive on $S(H) \times S(H)$, the FTG bounds imply the window $\varepsilon \leq B(w)_{ij} \leq U$ on $S(H) \times S(H)$, hence the Birkhoff/Hilbert contraction factor is at most τ . Syndeticity implies at least $\lfloor n/(L_{\text{syn}} + |w|) \rfloor$ contraction events by time n , so $d_H(\cdot) \leq \tau^{\lfloor n/(L_{\text{syn}} + |w|) \rfloor} d_H(x, y)$, and the exponential form follows as in Corollary 13.5. $\square$

13.3 A mean-minimizer version (forced frequency from positive core time)

The preceding theorem is uniform on the induced subshift over H . For a general minimizing path, tightness holds only in density, so bounded gaps for w need not hold on the full path. However, if a minimizing path spends positive lower density of (tight) time in the aperiodic SCC H , then the finite recurrence modulus L_{syn} forces *positive lower density* of Doeblin blocks along that minimizer, which is sufficient for an explicit exponential contraction rate.

Theorem 13.7 (Contraction along minimizers in the Doeblin regime). *Fix an aperiodic tight SCC H and a certified Doeblin word w on $S(H)$ as in Theorem 13.6. Assume $OG4$ does not occur for (H, w) , so $\mathcal{A}_{\neg w}(H)$ is acyclic and $L_{\text{syn}}(H, w)$ is defined. Let $L_0 := L_{\text{syn}}(H, w) + |w|$.*

Let π be any one-sided minimizing path such that

$$\theta := \liminf_{n \rightarrow \infty} \frac{T_H(n; \pi)}{n} > 0,$$

where $T_H(n; \pi)$ counts times $t < n$ for which e_t is tight and its tail vertex lies in H (cf. Proposition 2.5). Then the Doeblin word w occurs along π with uniform positive lower density

$$\rho := \liminf_{n \rightarrow \infty} \frac{N_w(n; \pi)}{n} \geq \frac{\theta}{L_0}$$

(Corollary 8.5). Consequently, for all $x, y \in \mathbb{R}_{>0}^{S(H)}$,

$$d_H(A^{(n)}(\pi)x, A^{(n)}(\pi)y) \leq C e^{-\kappa n} d_H(x, y) \quad (n \geq 1),$$

with explicit constants

$$\begin{aligned} \varepsilon &:= m_*^{|w|}, & U &:= (dM_*)^{|w|}, \\ \tau &:= \tanh\left(\frac{1}{2} \log(U/\varepsilon)\right), & C &:= \tau^{-1}, \\ \kappa &:= \frac{\rho}{|w|}(-\log \tau) \geq \frac{\theta}{L_0|w|}(-\log \tau). \end{aligned}$$

Proof. Because $OG4$ does not occur, **(RC1)** holds for (H, w) and Corollary 8.5 yields $\rho \geq \theta/L_0$. Positive lower density of Doeblin blocks is precisely recurrence input (RH2) from Definition 13.4. Apply Corollary 13.5 with this ρ . $\square$

Remark 13.8 (What is “gap-free” here). *Theorems 13.6 and 13.7 remove recurrence as a hypothesis: the only alternative to contraction is the explicit finite-witness obstruction $OG4$. For minimizing paths, the positive-density condition $\theta > 0$ is guaranteed whenever an aperiodic tight SCC is selected in the tight-core dichotomy (Proposition 2.5).*

14 Rate Extraction without Recurrence Inputs

This section is a clean “rate pack” that converts the forcing outcomes (Sections 6, 7, and 8) and the avoidance certificate **(RC1)** into explicit exponential contraction constants. It is designed to be referenced directly by the pipeline contract in Section 16.

14.1 Constants from a Doeblin witness

Fix an aperiodic tight SCC H of $G^{(0)}$ with a consistent active face $S(H)$ and a certified Doeblin word w supported in H . Write $L := |w|$ and let $B(w)$ be the block product. By certification, $B(w)$ is strictly positive on $S(H) \times S(H)$.

Define the explicit FTG window and Birkhoff factor

$$\varepsilon := m_*^L, \quad U := (dM_*)^L, \quad \tau := \tanh\left(\frac{1}{2} \log(U/\varepsilon)\right) \in (0, 1). \quad (17)$$

Then every occurrence of w contracts Hilbert distance on the active face by factor at most τ (Lemma 13.2 and Corollary 13.5).

14.2 Rate from syndetic recurrence (RC1 on a tight tail)

Proposition 14.1 (Rate from bounded gaps). *Assume OG_4 fails for (H, w) , so the avoidance graph $\mathcal{A}_{\neg w}(H)$ is acyclic and the explicit gap modulus $L_{\text{syn}} := L_{\text{syn}}(H, w)$ is defined. Let π be any infinite path whose tail is a path in H (in particular, any tight minimizer supported in H). Then the occurrences of w along π are syndetic with gap bound L_{syn} . Consequently, for all $x, y \in \mathbb{R}_{>0}^{S(H)}$ and all $n \geq 1$,*

$$d_H(A^{(n)}(\pi)x, A^{(n)}(\pi)y) \leq C e^{-\kappa n} d_H(x, y)$$

with

$$C := \tau^{-1}, \quad \kappa := \frac{-\log \tau}{L_{\text{syn}} + L}. \quad (18)$$

Proof. If $\mathcal{A}_{\neg w}(H)$ is acyclic, Lemma 11.4 provides syndetic recurrence on every infinite path in H . The contraction estimate is exactly the (RH1) branch of Corollary 13.5, using (17). $\square$

14.3 Rate from forced positive density (mean-minimizers with zero slack density)

Proposition 14.2 (Rate from forced word density). *Assume OG_4 fails for (H, w) , and let $L_0 := L_{\text{syn}}(H, w) + L$. Let π be any one-sided path and define*

$$\theta := \liminf_{n \rightarrow \infty} \frac{T_H(n; \pi)}{n}, \quad \eta := \limsup_{n \rightarrow \infty} \frac{\text{SlackMass}_n(\pi)}{n}.$$

If $\theta > 0$ and $\eta = 0$, then

$$\rho := \liminf_{n \rightarrow \infty} \frac{N_w(n; \pi)}{n} \geq \frac{\theta}{L_0} \quad (19)$$

(Corollary 8.5). Consequently, for all $x, y \in \mathbb{R}_{>0}^{S(H)}$ and all $n \geq 1$,

$$d_H(A^{(n)}(\pi)x, A^{(n)}(\pi)y) \leq C e^{-\kappa n} d_H(x, y)$$

with

$$C := \tau^{-1}, \quad \kappa := \frac{\rho}{L}(-\log \tau) \geq \frac{\theta}{L_0 L}(-\log \tau). \quad (20)$$

Proof. Equation (19) is Corollary 8.5. Given a lower density ρ of starting indices, extract a disjoint subfamily of occurrences by the standard greedy rule. Among the first n steps this produces at least $\lfloor \rho n / L \rfloor$ disjoint w -blocks. Each disjoint block contracts by factor at most τ , yielding $d_H(\cdot) \leq \tau^{\lfloor \rho n / L \rfloor} d_H(x, y) \leq \tau^{-1} e^{-(\rho/L)(-\log \tau)n} d_H(x, y)$, which is (20). $\square$

Remark 14.3 (Where θ enters). *The parameter θ measures the asymptotic lower density of tight time spent in the chosen aperiodic SCC H . For tight minimizers supported in H one has $\theta = 1$, so (20) reduces to an explicit path-independent lower bound. For general minimizers, θ is recorded as a trajectory-level input (or estimated in applications) while the constants τ and L_0 are fully witness-derived.*

15 Complete Tight-Core Boundary Theorem

This section rewrites the flagship boundary classification so it is finite and witness-producing. The theorem stops at the boundary objects that are decidable from finite data: tight-core type, face-map consistency, and Doeblin-word existence.

Recurrence along a chosen minimizing trajectory is a separate layer and is used only under explicit recurrence inputs.

Definition 15.1 (Obstruction gates (OG1–OG4)). *The finite boundary analysis is organized around four obstruction gates.*

- **OG1:** every tight SCC is imprimitive ($\text{per}(H) > 1$), so no aperiodic tight core exists.
- **OG2:** an aperiodic tight SCC exists, but active-face consistency fails (Definition 9.1), producing a cycle witness (Lemma 9.9).
- **OG3:** a consistent active face exists, but there is no Doeblin word on that face; this yields a finite cut or product-automaton non-reachability witness (Algorithm 3).
- **OG4:** a Doeblin word exists, but it is not recurrent along the tight dynamics; this yields an explicit avoidance-cycle witness (Lemma 11.3) and is formalized as Definition 12.1.

Corollary 15.2 (Obstruction witnesses are explicit). *If any obstruction gate from Definition 15.1 holds, then the certificate pipeline produces a finite, checkable witness object (cycle, cut, automaton non-reachability, or avoidance cycle). In particular, in the presence of an obstruction gate, no gap-free Doeblin + recurrence contraction conclusion can be asserted on the corresponding tight core.*

Theorem 15.3 (Complete Tight-Core Boundary Theorem (finite, witness-producing form)). *Let $G^{(0)}$ be the tight graph associated to a calibrated potential (Definition 2.2). Then exactly one of the following holds.*

- aperiodic tight core.** Every SCC H of $G^{(0)}$ is imprimitive, i.e. $\text{per}(H) > 1$ (Definition 2.3). A finite witness is the SCC decomposition of $G^{(0)}$ together with the computed periods $\text{per}(H)$.
- aperiodic tight SCC exists.** There exists an aperiodic SCC H of $G^{(0)}$, i.e. $\text{per}(H) = 1$. Fix such an H . Then exactly one of the following holds on H .
 - face instability on H .** The descriptor reduction fails to provide a consistent face map on H (Definition 9.1). A finite witness is a directed cycle word in H with marked indices (i, j) witnessing spurious activation, as in Lemma 9.9.
 - no active face on H .** A consistent face map exists on H with active face $S(H)$, but there is no word w supported in H whose face-restricted block product is strictly positive on $S(H) \times S(H)$. A finite witness is either
 - * a cut witness showing $\Gamma(H)$ is not strongly connected, or
 - * a Boolean-pattern product-automaton non-reachability certificate for the all-ones pattern state $(v_0, \mathbf{1})$ produced by Algorithm 3.
 - word is certified on H .** There exists a word w supported in H such that the face-restricted block product $B(w)$ is strictly positive on $S(H) \times S(H)$. Equivalently, the Boolean-pattern product-automaton test reaches a return state $(v_0, \mathbf{1})$ and returns an explicit witness word w (Corollary 9.6). The witness object is the triple $(H, S(H), w)$.

Moreover, if a minimizing path π admits an aperiodic tight SCC in its tight-core dichotomy (Proposition 2.5), then applying the aperiodic branch above to that SCC returns a finite boundary object: an explicit OG2 witness, an explicit OG3 witness, or an explicit Doeblin word witness.

Proof map. Compute the SCC decomposition of $G^{(0)}$ and the period $\text{per}(H)$ of each SCC (Definition 2.3). If all SCCs satisfy $\text{per}(H) > 1$, we are in (OG1). Otherwise, fix an aperiodic SCC H .

Face-map consistency on H is a finite check on the template sparsity patterns and yields an explicit inconsistency cycle witness when it fails (Lemma 9.9). Assuming a consistent face map, Doeblin-word existence on $S(H)$ is equivalent to reachability of a return state $(v_0, \mathbf{1})$ in the Boolean-pattern product automaton. Algorithm 3 returns either an explicit witness word w or a finite non-reachability certificate (Corollary 9.6). These outcomes are mutually exclusive and exhaustive, yielding (OG2), (OG3), or (**Doeblin**).

Finally, Proposition 2.5 identifies when a minimizing path supplies an aperiodic tight SCC to which the aperiodic branch applies. $\square$

16 Pipeline Contract (recurrence derived, finite-witness branches)

This section states the end-to-end contract of the gap-free pipeline once the finite recurrence layer and the rate extraction layer are included. The key upgrade is that *recurrence is no longer an external hypothesis*. Instead, the pipeline returns either a *finite obstruction witness* (OG1–OG4) or an *explicit contraction certificate with rate*.

16.1 Inputs and canonical objects

The pipeline assumes the standard reduced-graph data already used by the obstruction checks:

- a finite reduced directed graph $G = (V, E)$,
- for each edge $e : u \rightarrow v$ a nonnegative template (or fixed-length block) $T_e \in \mathbb{R}_+^{d \times d}$,
- the calibrated potential ϕ and reduced costs $c_\phi(e) \geq 0$ (so the tight edge set $E^{(0)} = \{c_\phi = 0\}$ is defined),
- the canonical support sets $S(v) \subseteq \{1, \dots, d\}$ per descriptor state.

From these, the pipeline constructs:

- the tight graph $G^{(0)} = (V^{(0)}, E^{(0)})$,
- SCC decomposition and periods $\text{per}(H)$,
- face-map validation and SCC face $S(H) = \bigcap_{v \in H} S(v)$ when consistent (Definition 9.1),
- the coordinate digraph $\Gamma(H)$ on $S(H)$ (Definition 9.4),
- the Boolean-pattern product automaton for Doeblin-word search (Algorithm 3),
- for a returned Doeblin word w , the avoidance graph $\mathcal{A}_{\neg w}(H)$ and (if acyclic) the modulus $L_{\text{syn}}(H, w)$ (Algorithm 4).

16.2 Branch logic and output artifacts

The pipeline returns exactly one of the following mutually exclusive outcomes.

- (OG1 witness) No aperiodic SCC exists in $G^{(0)}$. Return the SCC/period table certifying that every tight SCC is imprimitive.
- (OG2 witness) An aperiodic tight SCC H exists, but the proposed face sets fail to form a face map on H . Return the inconsistency cycle witness from Lemma 9.9.
- (OG3 witness) An aperiodic tight SCC H with a valid face map exists, but no Doeblin word exists on $S(H)$. Return either:
 - a cut witness $(H, S(H), A, B)$ from Lemma 9.10, or
 - a product-automaton non-reachability certificate (Corollary 9.6).
- (OG4 witness) A Doeblin word w on $S(H)$ is certified, but RC1 fails for (H, w) . Return a directed cycle in $\mathcal{A}_{\neg w}(H)$, which yields an explicit periodic tight minimizer avoiding w (Proposition 12.2).
- (action certificate) A Doeblin word w on $S(H)$ is certified and RC1 holds for (H, w) . Return:
 - the witness word w and the modulus $L_{\text{syn}}(H, w)$,
 - the FTG window parameters (ε, U) for the Doeblin block on $S(H)$,
 - the Hilbert contraction factor τ and the exponential rate parameters (C, κ) as in Theorem 13.6.

This certificate implies uniform exponential projective contraction along every tight minimizer supported in H .

16.3 How general minimizers are handled

For a general one-sided minimizing path π , the tight edges need not appear with bounded gaps. The pipeline therefore records the correct forcing upgrade:

- if π has nontrivial tight time in the chosen SCC H (measured by $\theta = \liminf T_H(n; \pi)/n > 0$) and slack density 0, then the forcing lemma yields a *positive lower density* of Doeblin blocks (Corollary 8.5),
- this density supplies the recurrence input needed by the contraction module, yielding an explicit rate (Theorem 13.7 and the formulas packaged in Section 14).

Accordingly, the pipeline contract records two rate modes:

- a *tight-minimizer* rate (C, κ) that depends only on finite witnesses (H, w) and the FTG bounds,
- a *mean-minimizer* rate that additionally depends on the measurable parameter θ of the specific trajectory.

Remark 16.1 (What is now fully finite). *All branching decisions up through OG4 and the tight-minimizer contraction certificate are entirely finite and witness-producing. The only non-finite input that can remain in applications is an estimate of how much time a given minimizer spends in the SCC H (the parameter θ). This is the correct remaining interface: it is a property of the chosen trajectory, not a hidden structural assumption.*

17 Robust Certificate Contract (RU-A)

This section freezes a single portable object that stores *all* margins and thresholds needed to make robustness claims mechanically traceable. Later sections may assert “the conclusion survives perturbation” only by citing explicit fields of this object.

17.1 Perturbation model

We allow two perturbation channels:

- **Cost perturbations.** The edge cost vector $c : E \rightarrow \mathbb{R}$ is perturbed to $\tilde{c} = c + \Delta c$ with

$$\|\Delta c\|_\infty := \max_{e \in E} |\Delta c(e)| \leq \varepsilon_c.$$

- **Template perturbations.** Each template matrix $A^{(k)} \in \mathbb{R}_+^{d \times d}$ is perturbed to $\tilde{A}^{(k)} = A^{(k)} + \Delta A^{(k)}$ under one of the following metrics, which must be fixed in the certificate:

- *Absolute entrywise:* $\|\Delta A\|_{\infty, \text{ent}} := \max_{k, i, j} |\Delta A_{ij}^{(k)}| \leq \varepsilon_A.$
- *Relative entrywise:* $|\Delta A_{ij}^{(k)}| \leq \varepsilon_A A_{ij}^{(k)}$ for all $k, i, j.$

When both channels are present, we write $\varepsilon := (\varepsilon_c, \varepsilon_A)$ and define the *robust radius* as the largest admissible pair ε^* stored in the robust certificate.

17.2 Margins and openness predicates

Fix a calibrated subaction $u : V \rightarrow \mathbb{R}$ and level λ_* , and reduced costs c_u as in (2). Write $H := \{e \in E : c_u(e) = 0\}$ for the tight edge set.

Definition 17.1 (Tightness gap margin). *Assume $H \neq E$. The tightness gap is*

$$\gamma := \min\{c_u(e) : e \in E \setminus H\}.$$

If $H = E$ we set $\gamma := +\infty$.

Lemma 17.2 (Openness of the tightness sign test). *Let $\tilde{c} = c + \Delta c$ with $\|\Delta c\|_\infty \leq \varepsilon_c$ and keep (u, λ_*) fixed. Then for every edge $e \in E$,*

$$|\tilde{c}_u(e) - c_u(e)| \leq \varepsilon_c, \quad \tilde{c}_u(e) := \tilde{c}(e) - \lambda_* + u(\text{tail}(e)) - u(\text{head}(e)).$$

In particular, if $\varepsilon_c < \gamma/2$, then

$$e \in H \implies \tilde{c}_u(e) \leq \gamma/2, \quad e \notin H \implies \tilde{c}_u(e) \geq \gamma/2.$$

Lemma 17.2 is the primitive “openness” statement behind tight-core stability; RU-B upgrades it to a combinatorial invariance theorem for the tight graph and its SCC decomposition.

Next, fix a Doeblin word witness $w = w_1 \cdots w_L$ (over the template index alphabet) and define its product

$$P_w := A^{(w_L)} \cdots A^{(w_1)}.$$

Let $\delta := \min_{i,j} (P_w)_{ij}$ be the Doeblin margin on the relevant block.

Definition 17.3 (Uniform entry bound). *Fix a bound $M \geq 1$ satisfying $\max_{k,i,j} A_{ij}^{(k)} \leq M$ on the block where the Doeblin witness is evaluated. The certificate must store the value of M used in robustness inequalities.*

Lemma 17.4 (Openness of Doeblin positivity under absolute perturbations). *Assume absolute entrywise perturbations with $\|\Delta A\|_{\infty, \text{ent}} \leq \varepsilon_A$ and $\max_{k,i,j} A_{ij}^{(k)} \leq M$. Let $\tilde{P}_w := \tilde{A}^{(w_L)} \dots \tilde{A}^{(w_1)}$. Then*

$$\|\tilde{P}_w - P_w\|_{\infty, \text{ent}} \leq L(M + \varepsilon_A)^{L-1} \varepsilon_A,$$

and therefore the perturbed Doeblin margin satisfies

$$\delta'(\varepsilon_A) := \min_{i,j} (\tilde{P}_w)_{ij} \geq \delta - L(M + \varepsilon_A)^{L-1} \varepsilon_A.$$

In particular, any ε_A such that $\delta'(\varepsilon_A) > 0$ guarantees that the same word w remains a valid Doeblin witness in the perturbed instance, with margin at least $\delta'(\varepsilon_A)$.

Remark 17.5. *For relative entrywise perturbations, one may rewrite Lemma 17.4 with ε_A interpreted as a relative factor and with M replaced by the corresponding upper envelope on the block. The robust certificate must record which perturbation metric is used and the bound actually fed into the estimate.*

17.3 Robust certificate object

We now freeze the schema.

Definition 17.6 (Robust certificate object). *A robust certificate is a structured object containing:*

- *a model fingerprint (`model_hash`) and the perturbation metric (`perturbation_metric`);*
- *the calibration data (u, λ_*) used for reduced costs, together with the tight set H ;*
- *margins and thresholds:*
 - *`tight_gap_gamma` storing γ from Definition 17.1;*
 - *`eps_tight` such that $\varepsilon_c < \text{eps_tight}$ implies Lemma 17.2;*
 - *`doeblyn.word` storing w , `doeblyn.L` storing L , and `doeblyn.delta` storing δ ;*
 - *`doeblyn.M` storing the bound M from Definition 17.3;*
 - *`eps_doeblin` such that $\varepsilon_A < \text{eps_doeblyn}$ implies $\delta'(\varepsilon_A) > 0$ in Lemma 17.4;*
 - *a rate pack section that records the contraction constants used in the unperturbed instance and provides explicit degraded constants as functions of $(\varepsilon_c, \varepsilon_A)$ whenever $\varepsilon \leq \varepsilon^*$.*
- *the aggregate robust radius `eps_star`: a pair $\varepsilon^* = (\varepsilon_c^*, \varepsilon_A^*)$ satisfying*

$$\varepsilon_c^* \leq \text{eps_tight}, \quad \varepsilon_A^* \leq \text{eps_doeblyn},$$

and any additional thresholds required by later robustness layers (RU-C and RU-D).

17.4 Concrete JSON schema (pipeline-facing)

The TeX contract above is mirrored by a pipeline-facing JSON schema and an example instance. These are provided as separate files in the repository root:

- `robust_certificate_schema.json`
- `robust_certificate_example.json`

For readability, we also display a concise schema skeleton here:

```
{
  "model_hash": "string",
  "perturbation_metric": {
    "cost_norm": "linfty",
    "matrix_metric": "absolute_entrywise|relative_entrywise",
    "matrix_norm": "linfty_entrywise"
  },
  "calibration": {
    "lambda_star": "number",
    "u_potential": "array[number]"
  },
  "tight_core": {
    "tight_edges": "array[edge_id]",
    "tight_gap_gamma": "number",
    "tight_lipschitz_Kc": "number",
    "eps_tight": "number",
    "tight_graph_hash": "string_or_null"
  },
  "doebelin": {
    "word": "array[template_id]",
    "L": "integer",
    "M_bound": "number",
    "U_bound": "number",
    "delta": "number",
    "eps_doeblin": "number",
    "delta_prime_formula": "string",
    "U_prime_formula": "string"
  },
  "recurrence": {
    "mode": "combinatorial|hypothesis",
    "eps_recurrence": "number",
    "details": "object"
  },
  "rate_pack": {
    "projective_diameter_bound": "number",
    "tau_unperturbed": "number",
    "tau_prime_formula": "string",
    "eps_rate": "number",
  }
}
```

```

    "kappa_prime_formula": "string"
  },
  "eps_star": {
    "eps_c_star": "number",
    "eps_A_star": "number"
  }
}

```

Remark 17.7. *Every later statement of the form “the conclusion survives perturbation” must cite: a margin field, the chosen perturbation metric, an explicit computed threshold, and the degraded constants produced when $\varepsilon \leq \varepsilon^*$. No other robustness language is permitted.*

18 Tight-Core Stability Theorem (RU-B)

This section upgrades the openness lemma of RU-A into a purely combinatorial invariance theorem: under an explicit margin condition, the *tight edge set*, its SCC decomposition, and its periodicity labels are unchanged throughout a perturbation neighborhood.

18.1 Stability statement for the tight-edge sign test

Fix a calibrated pair (u, λ_*) and reduced costs c_u as in (2). Let $H := \{e \in E : c_u(e) = 0\}$ be the tight edge set and let

$$\gamma := \min\{c_u(e) : e \in E \setminus H\} \in [0, +\infty)$$

(with the convention that $\gamma := 0$ when $H = E$).

Lemma 18.1 (Tight-set invariance under cost perturbations). *Let $\tilde{c} = c + \Delta c$ with $\|\Delta c\|_\infty \leq \varepsilon_c$ and keep (u, λ_*) fixed. If $\varepsilon_c < \gamma/2$, then*

$$\tilde{H} := \{e \in E : \tilde{c}_u(e) \leq \gamma/2\} = H,$$

and the inequalities

$$e \in H \implies \tilde{c}_u(e) \leq \gamma/2, \quad e \notin H \implies \tilde{c}_u(e) \geq \gamma/2$$

hold.

Proof. This is exactly the sign-stability conclusion of Lemma 17.2, with the explicit convention that $\gamma = 0$ forces the stable radius to be 0 in the degenerate all-tight case. $\square$

18.2 Combinatorial invariance: SCC structure and periodicity labels

Let $G^{(0)} = (V, E^{(0)})$ be the directed graph obtained by restricting $G = (V, E)$ to the tight edges $E^{(0)} := H$. Write $\text{SCC}(G^{(0)})$ for its SCC decomposition. For an SCC C we write $\text{per}(C)$ for its period (the gcd of lengths of directed cycles in C).

Theorem 18.2 (Tight-core structural stability). *Assume $\gamma > 0$ and let $\tilde{c} = c + \Delta c$ with $\|\Delta c\|_\infty \leq \varepsilon_c < \gamma/2$. Define the perturbed tight set by the stable sign test*

$$\tilde{H} := \{e \in E : \tilde{c}_u(e) \leq \gamma/2\}.$$

Then:

- $\tilde{H} = H$ (tight edge set is unchanged),
- the SCC decomposition of the tight graph is unchanged: $\text{SCC}(\tilde{G}^{(0)}) = \text{SCC}(G^{(0)})$,
- every SCC period is unchanged: for each tight SCC C ,

$$\text{per}_{\tilde{G}^{(0)}}(C) = \text{per}_{G^{(0)}}(C),$$

so the periodic/apperiodic labels of SCCs are invariant.

Proof. The first bullet is Lemma 18.1. Once the edge set of the tight subgraph is identical, the directed graph itself is identical, so the SCC decomposition and the cycle-length sets within each SCC are identical. Hence the gcd defining the period is unchanged. $\square$

Remark 18.3 (What this theorem does and does not claim). *Theorem 18.2 is a certificate-level stability statement. It asserts that all downstream logic that depends only on the tight graph (SCCs, periods, face-map checks, Doeblin search, avoidance graphs, and the OG1–OG4 branching) is robust throughout the neighborhood $\|\Delta c\|_\infty < \gamma/2$, provided the certificate keeps (u, λ_*) fixed.*

It does not assert that the globally optimal calibrated pair for the perturbed instance is the same, nor that the minimizing set for the perturbed instance coincides with the unperturbed one. Those are separate questions; the robustness contract is that the original witness package remains valid and produces valid conclusions for all perturbations within the certified radius.

18.3 Pipeline-facing fields added by RU-B

RU-B adds the following required fields to the `tight_core` section of the robust certificate:

- `tight_lipschitz_Kc`: a constant $K_c \geq 1$ such that $|\tilde{c}_u(e) - c_u(e)| \leq K_c \|\Delta c\|_\infty$ for all edges when (u, λ_*) is kept fixed (in the basic cost-perturbation model, $K_c = 1$),
- `eps_tight`: the computed stability threshold

$$\text{eps_tight} = \frac{\gamma}{2K_c},$$

so that $\|\Delta c\|_\infty < \text{eps_tight}$ implies tight-core invariance,

- an optional `tight_graph_hash` (a fingerprint of the tight edge set) and `tight_scc_summary` (counts and period labels) for auditability.

These fields enforce the hard acceptance gate:

Every later robustness claim about the tight core must cite `tight_gap_gamma`, `tight_lipschitz_Kc`, and `eps_tight`.

19 Doeblin Stability and Robust Rate Pack (RU-C)

RU-A records a Doeblin word witness $w = w_1 \cdots w_L$ and its unperturbed margin $\delta = \min_{i,j} (P_w)_{ij} > 0$ on the relevant core/face block, where $P_w = A^{(w_L)} \cdots A^{(w_1)}$. RU-C upgrades this to a *robust rate pack*: the *same witness word* persists under small perturbations, and the associated Hilbert/Birkhoff contraction constants degrade by an explicit computable formula.

Throughout this section we work under the perturbation metric fixed in the robust certificate. We write ε_A for the template perturbation radius and assume the tight core is held fixed throughout the neighborhood by RU-B (so the witness word remains admissible in the same tight SCC).

19.1 Product perturbation bounds and a surviving Doeblin margin

Assume the certificate stores a uniform entry bound $M \geq 1$ such that $\max_{k,i,j} A_{ij}^{(k)} \leq M$ on the block where the Doeblin witness is evaluated.

Lemma 19.1 (Entrywise product perturbation bound). *Assume absolute entrywise perturbations $\|\Delta A\|_{\infty, \text{ent}} \leq \varepsilon_A$. Let $\tilde{P}_w := \tilde{A}^{(w_L)} \dots \tilde{A}^{(w_1)}$. Then*

$$\|\tilde{P}_w - P_w\|_{\infty, \text{ent}} \leq L(M + \varepsilon_A)^{L-1} \varepsilon_A.$$

Consequently, the perturbed Doeblin margin satisfies

$$\delta'(\varepsilon_A) := \min_{i,j} (\tilde{P}_w)_{ij} \geq \delta - L(M + \varepsilon_A)^{L-1} \varepsilon_A.$$

Proof. This is Lemma 17.4, restated for later reuse. $\square$

Remark 19.2 (Relative entrywise perturbations). *For relative entrywise perturbations $|\Delta A_{ij}^{(k)}| \leq \varepsilon_A A_{ij}^{(k)}$, one may convert to an absolute bound on the relevant block: $\|\Delta A\|_{\infty, \text{ent}} \leq \varepsilon_A M$. All formulas below remain valid after replacing ε_A by $\varepsilon_A M$ in the right-hand sides and recording this choice in the certificate.*

19.2 A robust diameter bound and Birkhoff contraction coefficient

In addition to δ , the robust certificate stores an *upper* bound $U := \max_{i,j} (P_w)_{ij}$ on the same block (or any certified upper envelope used by the pipeline).

Lemma 19.3 (Upper envelope under perturbations). *Under the assumptions of Lemma 19.1,*

$$U'(\varepsilon_A) := \max_{i,j} (\tilde{P}_w)_{ij} \leq U + L(M + \varepsilon_A)^{L-1} \varepsilon_A.$$

Proof. Entrywise, $|(\tilde{P}_w)_{ij} - (P_w)_{ij}| \leq \|\tilde{P}_w - P_w\|_{\infty, \text{ent}}$. Taking maxima and applying Lemma 19.1 gives the claim. $\square$

Lemma 19.4 (Robust projective diameter bound for a positive block). *Assume $\delta'(\varepsilon_A) > 0$ and let $U'(\varepsilon_A)$ be as in Lemma 19.3. Then the Hilbert projective diameter of the image of the positive cone under $\tilde{P}_w$ satisfies*

$$\Delta(\varepsilon_A) \leq 2 \log \left(\frac{U'(\varepsilon_A)}{\delta'(\varepsilon_A)} \right).$$

In particular, the Birkhoff contraction coefficient of $\tilde{P}_w$ satisfies

$$\tau'(\varepsilon_A) \leq \tanh \left(\frac{\Delta(\varepsilon_A)}{4} \right).$$

Proof. If all entries of a positive matrix lie in $[\delta', U']$, then $\max_{i,j,k,\ell} \frac{B_{ik} B_{j\ell}}{B_{i\ell} B_{jk}} \leq (U'/\delta')^2$. The standard diameter bound is $\Delta(B) \leq \log \left(\max \frac{B_{ik} B_{j\ell}}{B_{i\ell} B_{jk}} \right)$, giving $\Delta \leq 2 \log(U'/\delta')$. The Birkhoff bound then yields $\tau \leq \tanh(\Delta/4)$. $\square$

19.3 Robust rate extraction under a recurrence certificate

RU-C does *not* assert recurrence. Instead, it states: once recurrence is certified (RU-D) or assumed as a named hypothesis, the contraction rate survives perturbation with explicit degradation.

Proposition 19.5 (Robust contraction per Doeblin occurrence). *Assume $\delta'(\varepsilon_A) > 0$. Whenever the Doeblin word w occurs (so the block product applied at that time is $\tilde{P}_w$), the Hilbert distance contracts by at least the factor $\tau'(\varepsilon_A)$ from Lemma 19.4.*

Proof. This is the Birkhoff contraction principle applied to the positive block $\tilde{P}_w$. $\square$

Corollary 19.6 (Rate degradation under syndetic recurrence). *Assume a recurrence certificate provides a bounded-gap modulus B for occurrences of w along the trajectory under study. Let $L = |w|$ and suppose $\delta'(\varepsilon_A) > 0$. Then for all times n ,*

$$d_H(x_n, y_n) \leq C (\tau'(\varepsilon_A))^{\lfloor n/(L+B) \rfloor} d_H(x_0, y_0),$$

so an admissible exponential rate is

$$\kappa'(\varepsilon_A) = -\frac{\log \tau'(\varepsilon_A)}{L+B}.$$

Proof. Partition the trajectory into consecutive windows of length at most $L+B$, each containing an occurrence of w . Each window contributes at least one contraction by Proposition 19.5; absorbing the inter-block projective nonexpansive bounds into a prefactor yields the stated inequality. $\square$

19.4 Pipeline-facing fields added by RU-C

RU-C requires the robust certificate to store enough information to compute:

- the surviving Doeblin margin function $\delta'(\varepsilon_A)$ and a computed threshold `eps_doeblin` ensuring $\delta'(\varepsilon_A) > 0$,
- an upper envelope U for the unperturbed Doeblin block and a formula for $U'(\varepsilon_A)$,
- a diameter bound $\Delta(\varepsilon_A)$ and a contraction factor bound $\tau'(\varepsilon_A)$,
- an optional `eps_rate` threshold under which a target contraction quality is guaranteed (e.g., $\delta'(\varepsilon_A) \geq \delta/2$ and $U'(\varepsilon_A) \leq 2U$).

These enforce the hard acceptance gate:

No statement may claim robust survival of a Doeblin witness or a rate without citing `doeblin.delta`, `doeblin.U_bound`, and the computed thresholds `eps_doeblin` and (when used) `eps_rate`.

20 Recurrence Robustness Layer (Robust Mode)

The gap-free upgrade separates *existence* of an algebraically contracting block (a Doeblin word on the active face) from *recurrence* of that block along trajectories. A correct finite recurrence layer is obtained by separating two objects: **(RC1)** is a purely combinatorial predicate for tight trajectories in a fixed tight SCC H , while $\text{OG4}(H, w)$ records the explicit failure mode via an avoidance-cycle witness.

This section enforces that separation in a referee-stable way. All recurrence statements are permitted only in one of two modes, each with an explicit robustness threshold.

20.1 Two recurrence modes with margins

Fix an aperiodic tight SCC H of the tight graph $G^{(0)}$ and a certified Doeblin word w supported in H .

Definition 20.1 (Recurrence modes for a robust certificate). *A robust certificate records recurrence information in exactly one of the following modes.*

- **Mode (combinatorial).** *The certificate records that (RC1) holds for (H, w) (Definition 11.1), with an explicit syndetic modulus $L_{\text{syn}}(H, w)$ (Lemma 11.4). Equivalently, the avoidance graph $\mathcal{A}_{\neg w}(H)$ is acyclic.*
- **Mode (hypothesis).** *The certificate records a user-supplied recurrence hypothesis on a chosen minimizing trajectory, for example a bounded-gap hypothesis (gap bound B) or a lower-density hypothesis (density margin $\eta > 0$). In this mode, the pipeline does not certify the truth of the hypothesis; it only certifies that all constants depending on w and the template family degrade explicitly and continuously under perturbation.*

20.2 Robustness threshold for recurrence statements

The point of robust mode is that the same finite witness objects remain valid on an explicit neighborhood. The neighborhood size is computed from earlier margins.

Definition 20.2 (Recurrence robustness radius). *Let $\varepsilon_{\text{tight}}$ be the tight-core stability threshold from RU-B (Theorem in Section 18) and let $\varepsilon_{\text{Doeblin}}$ be the Doeblin-survival threshold from RU-C (Section 19). Define*

$$\varepsilon_{\text{rec}} := \min\{\varepsilon_{\text{tight}}, \varepsilon_{\text{Doeblin}}\}.$$

A recurrence statement is said to be robustly certified on the neighborhood of size ε_{rec} if it is in Mode (combinatorial) and its finite witness objects are unchanged throughout that neighborhood.

Proposition 20.3 (Stability of the RC1/OG4 dichotomy). *Assume $\varepsilon \leq \varepsilon_{\text{tight}}$. Then the tight graph $G^{(0)}$ and its SCC decomposition are unchanged under the perturbation model, so the avoidance graph $\mathcal{A}_{\neg w}(H)$ is unchanged as a finite directed graph. Consequently, the dichotomy*

$$(RC1) \text{ holds for } (H, w) \quad \text{or} \quad OG4(H, w) \text{ holds}$$

(and any explicit witness: either L_{syn} or an avoidance cycle) is invariant under all such perturbations.

Proof. Under $\varepsilon \leq \varepsilon_{\text{tight}}$, RU-B guarantees that the tight-edge set and hence the tight SCC H are unchanged. The avoidance graph $\mathcal{A}_{\neg w}(H)$ is constructed purely from the tight edge set of H and the fixed word w (Definition 11.2), so its vertex/edge set is unchanged. Therefore acyclicity (RC1) and the existence of an avoidance cycle (OG4) are unchanged, and any explicit witness carries over. $\square$

Remark 20.4 (Why ε_{rec} uses $\varepsilon_{\text{Doeblin}}$). *Proposition 20.3 is purely combinatorial and only needs $\varepsilon_{\text{tight}}$. However, in the contraction pipeline, recurrence is meaningful only in tandem with the Doeblin minorization property of w . Thus robust certificates record $\varepsilon_{\text{rec}} = \min\{\varepsilon_{\text{tight}}, \varepsilon_{\text{Doeblin}}\}$ so that “recurrence + Doeblin” survives together.*

20.3 What robust mode does and does not claim

- In Mode (combinatorial), recurrence is *derived* for *all* tight trajectories supported in the same SCC H , with an explicit gap bound L_{syn} . This derived recurrence is stable for all perturbations with $\varepsilon \leq \varepsilon_{\text{rec}}$.
- In Mode (hypothesis), recurrence is *not* asserted without the stated hypothesis. Robustness refers only to the *constants* in the contraction estimate (diameter bounds, Birkhoff factor, and rate degradation) when the same hypothesis is assumed along the perturbed instance.

Enforcement rule. No theorem, corollary, or remark may state a recurrence conclusion unless it cites *either* a Mode (combinatorial) certificate field (with L_{syn} and ε_{rec}) *or* a Mode (hypothesis) field (with its explicit margin and the disclaimer that the hypothesis is assumed).

21 Approximate Certification on Parameter Regions (Robust Neighborhood Cover)

This section upgrades pointwise certificates to *region certification*. The guiding observation is topological and quantitative: all certificate predicates used by the pipeline are *open* conditions once the robust margins are recorded. Therefore a finite verification on an ε -net certifies an entire parameter region except near an explicit exceptional set where margins vanish.

21.1 Local openness packaged as a single statement

Fix a parameter instance θ (costs and templates) and assume the pipeline produces a robust certificate object with robustness radii $\varepsilon_{\text{tight}}, \varepsilon_{\text{Doebelin}}, \varepsilon_{\text{rec}}$ and $\varepsilon_* = \min\{\varepsilon_{\text{tight}}, \varepsilon_{\text{Doebelin}}, \varepsilon_{\text{rate}}, \varepsilon_{\text{rec}}\}$ as in Sections 17–20.

Theorem 21.1 (Robust neighborhood invariance). *Let θ admit a robust certificate with radius $\varepsilon_* > 0$. Then every perturbed instance θ' with perturbation size $\leq \varepsilon_*$ satisfies:*

- *the tight graph $G^{(0)}$ and its SCC/period decomposition are unchanged;*
- *the stored Doeblin word w remains a valid Doeblin witness on the same active face, with explicit margin $\delta'(\varepsilon_*) > 0$;*
- *the recurrence layer is unchanged in Mode (combinatorial) (same RC1/OG4 outcome and witnesses), or remains a stated hypothesis in Mode (hypothesis);*
- *the projective contraction constants degrade explicitly according to the formulas stored in the certificate.*

In particular, every theorem in this paper that is invoked through certificate fields at θ holds verbatim at θ' with constants replaced by the certificate's degraded constants.

Proof. The tight-core invariance is RU-B. Doeblin survival and the explicit $\delta'(\varepsilon) / \tau'(\varepsilon)$ bounds are RU-C. Recurrence-layer stability in Mode (combinatorial) is RU-D. All remaining constants are explicit functions of these margins and the FTG bounds, hence are stable under the same inequalities. $\square$

21.2 Finite net certification of a compact region

Let (Ω, d) be a compact parameter set with a product metric compatible with the perturbation model (e.g. $d(\theta, \theta') = \max\{\|\Delta c\|_\infty, \|\Delta A\|_{\infty, \text{entry}}\}$ in absolute-entrywise mode).

Definition 21.2 (ε -net). *A finite set $\mathcal{N} \subset \Omega$ is an ε -net if every $\theta \in \Omega$ lies within distance ε of some $\theta_i \in \mathcal{N}$.*

Theorem 21.3 (Region certification by robust balls). *Suppose that for each net point $\theta_i \in \mathcal{N}$ the pipeline produces a robust certificate with radius $\varepsilon_*(\theta_i) > 0$. Define the certified region*

$$\Omega_{\text{cert}} := \bigcup_{\theta_i \in \mathcal{N}} B(\theta_i, \varepsilon_*(\theta_i)) \cap \Omega.$$

Then every instance $\theta \in \Omega_{\text{cert}}$ inherits the same certified conclusions as its center θ_i (in the sense of Theorem 21.1). Moreover, if $\mathcal{N}$ is an ε -net and $\min_i \varepsilon_(\theta_i) \geq \varepsilon$, then $\Omega_{\text{cert}} = \Omega$.*

Proof. If $\theta \in \Omega_{\text{cert}}$ then $\theta \in B(\theta_i, \varepsilon_*(\theta_i))$ for some center θ_i . Apply Theorem 21.1 at θ_i . The final claim follows directly from the definition of an ε -net. $\square$

21.3 Exceptional set description

The only ways robustness can fail are the explicit margin-boundary events where $\varepsilon_* = 0$. Accordingly, region mode reports uncovered points as being near one of the following certificate-boundary loci:

- **Tightness degeneracy:** $\gamma = 0$ (tight/non-tight separation vanishes), so the tight graph can change.
- **Doebelin degeneracy:** $\delta = 0$ for the stored word (or no Doebelin word exists), so minorization can fail.
- **Recurrence boundary:** the RC1/OG4 dichotomy changes only when the tight graph changes (hence is subsumed by $\gamma = 0$), while Mode (hypothesis) has no derived recurrence boundary because recurrence is not asserted.

21.4 Pipeline: region mode outputs

In region mode, the pipeline:

- samples or constructs a finite net $\mathcal{N}$ of parameter points;
- runs the pointwise certificate pipeline at each θ_i ;
- outputs the list of certified balls $(\theta_i, \varepsilon_*(\theta_i))$ and their conclusions;
- reports uncovered points and diagnoses them in terms of the margin boundaries above.

A recommended machine-readable artifact is a `robust_region_report.json` containing: centers with their radii ε_* , the induced outcome labels (OG1–OG4 or contraction), and an uncovered-set diagnostic summary.

22 Referee Hardening: Neutralizing the Standard Attacks

This section is a hostile-referee guide. It lists the classic failure modes that appear in “gap-free” contraction arguments and points to the exact locations in the manuscript where each is neutralized by either a concrete counter-shape (Section 24) or a finite-witness lemma (Sections 11 and 13).

22.1 Attack map

Referee attack	Where it is handled	What kills it
”You smuggled in recurrence / minimality.”	Sections 12, 13; Section 16	Recurrence is not assumed: it is either certified (RC1) or logged as OG4 with a finite avoidance cycle.
”Density-1 tightness does not imply bounded gaps.”	Section 24; Sections 6, 8, 13	The counterexamples show the leap is false; the forcing dichotomy replaces it by a quantitative implication from slack density 0 to Doeblin <i>frequency</i> .
”Strong connectivity on the active face should imply Doeblin.”	Section 24; Remark 9.7	Permutation-only regimes give strongly connected $\Gamma(H)$ with no strictly positive product; OG3 is certified by a product-automaton witness.
”Doeblin exists, therefore every minimizer contracts.”	Section 12; Theorem 13.6	False without recurrence: OG4 gives an explicit tight minimizer that avoids the Doeblin word; the strengthened theorem branches on a finite avoidance witness.
”Leaving and returning to the core can be done at zero cost.”	Sections 7, 8	Return-cost lemma: each excursion/return forces reduced-cost slack, converting avoidance strategies into a measurable slack burden.
”Your new obstruction is artificial / redundant.”	Proposition 12.2; Section 23	OG4 is realized by explicit examples where OG1–OG3 do not occur but the Doeblin word is avoidable by a periodic tight path.
”The good regime might be empty.”	Section 23	A concrete aperiodic SCC example is given where OG1–OG4 do not occur and RC1 forces recurrence of a Doeblin block, yielding a computable rate.

22.2 What a reader should verify mentally

A referee can run the logic end-to-end without trusting informal language. The following checkpoints are sufficient.

- The definitions of calibrated minimizers, reduced costs, and tight edges are fixed and do not presuppose recurrence (Sections 3 and 4).
- OG1–OG3 are purely finite, witness-producing obstructions to the existence of an algebraically contracting block on an aperiodic tight SCC (Definition 15.1 and Theorem 15.2).
- If OG1–OG3 do not occur on an aperiodic tight SCC H , the pipeline returns an explicit Doeblin word w and decides the finite recurrence predicate RC1. If RC1 fails, OG4 is produced with a finite avoidance-cycle witness (Definition 12.1).
- If RC1 holds, bounded-gap recurrence for tight paths is forced by a finite modulus $L_{\text{syn}}(H, w)$ (Definition 11.1 and Lemma 11.4), yielding uniform exponential contraction (Theorem 13.6).
- For general mean-minimizers, tightness is only a density statement. The correct replacement for “bounded gaps” is the forcing dichotomy: slack density 0 plus nontrivial tight time in H forces positive lower density of Doeblin blocks (Corollary 8.5), yielding an explicit contraction rate (Theorem 13.7).

22.3 Where the counterexample zoo matters

The counterexample zoo (Section 24) is not decorative. It is the guardrail that prevents reintroducing the three most common overreaches:

- replacing density information by bounded gaps,
- replacing strong connectivity by Doeblin positivity,
- replacing existence of a Doeblin block by universal contraction along all minimizers.

Each of these false implications is backed by an explicit adversarial shape in the zoo, and each is replaced later by a finite-witness branch (OG4) or by quantitative forcing lemmas (Sections 6 and 8).

23 Sharpness and Realizability of the Obstruction Taxonomy

This section records explicit, finite examples showing that each obstruction class OG1–OG4 is *genuinely necessary*. We also give a concrete “good regime” example where OG1–OG4 do not occur and the forcing mechanism (Sections 6 and 8) yields a nonvacuous unconditional contraction rate.

Throughout, we present examples at the reduced-graph/template level:

- vertices represent descriptor states,
- each directed edge $e : u \rightarrow v$ carries a nonnegative template matrix T_e ,
- the canonical support sets $S(\cdot)$ are specified explicitly and checked by support propagation (Definition 9.1).

23.1 OG1 is realizable (no aperiodic tight SCC)

Let $V = \{a, b\}$ and let the tight graph have exactly the edges $a \rightarrow b$ and $b \rightarrow a$. This SCC has period 2, hence it is imprimitive. Set $d = 1$ and take $T_{a \rightarrow b} = T_{b \rightarrow a} = [1]$. Then every edge is tight, the unique SCC has period 2, and therefore OG1 holds. No aperiodic tight SCC exists, so no Doeblin/minorization regime can be entered.

23.2 OG2 is realizable (face-map inconsistency cycle)

Let $V = \{u, v\}$, $d = 2$. Define support sets

$$S(u) = \{1\}, \quad S(v) = \{1\}.$$

Let the reduced graph contain both edges $u \rightarrow v$ and $v \rightarrow u$ (so it is an SCC). Assign the template on the edge $e : u \rightarrow v$ by

$$T_e = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}.$$

Then for $x \in \mathbb{R}_{>0}^{S(u)}$ we have $T_e x = (x_1, x_1)$, so coordinate 2 is activated. But $2 \notin S(v)$, hence the “no spurious activation” condition in Definition 9.1 fails. By Lemma 9.9, the edge e together with the marked indices $(i, j) = (2, 1)$ and any return path $v \rightsquigarrow u$ produces a finite inconsistency cycle witness. Thus OG2 is realizable.

23.3 OG3 is realizable (no Doeblin word on the active face)

We record two standard sharp shapes corresponding to the two OG3 witness types.

23.3.1 OG3(cut) realizable: block-triangular support

Let $V = \{h\}$ (one-vertex SCC), $d = 2$, $S(h) = \{1, 2\}$. Let the unique edge be a self-loop $e : h \rightarrow h$ with

$$T_e = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}.$$

Then $S(\cdot)$ is a face map and $S(H) = \{1, 2\}$. However, there is no arrow $2 \rightarrow 1$ in the coordinate digraph $\Gamma(H)$ (Definition 9.4), so $\Gamma(H)$ is not strongly connected. Equivalently, letting $A = \{1\}$ and $B = \{2\}$, we have $(T_e)_{1,2} = 0$ and hence $(B(w))_{1,2} = 0$ for every word w . Thus no word can be strictly positive on $S(H) \times S(H)$ and OG3 holds with a cut witness (Lemma 9.10).

23.3.2 OG3(product-automaton) realizable: permutation-only regime

Let $V = \{h\}$, $d = 2$, $S(h) = \{1, 2\}$. Let the allowed generator templates be the two permutation matrices

$$P = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad Q = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

and let the reduced graph have two self-loop edges at h labeled by P and Q . Then $\Gamma(H)$ is strongly connected (each coordinate reaches the other under Q), but every product is still a permutation matrix, so no word yields strict positivity on $S(H) \times S(H)$. This is exactly the “strongly connected but not Doeblin” counter-shape in Remark 9.7, and the Boolean-pattern product automaton returns a finite non-reachability certificate. Thus OG3 is realizable even in the presence of strong connectivity.

23.4 OG4 is realizable (Doeblin exists but is avoidable)

We now realize OG4 in the regime where OG1–OG3 fail. Let $V = \{h, x, y\}$ and consider the tight SCC H supported on these vertices. Take $d = 2$ and $S(v) = \{1, 2\}$ for all v . Let there be edges

$$h \rightarrow x, \quad x \rightarrow h, \quad x \rightarrow y, \quad y \rightarrow x.$$

This SCC is aperiodic (it contains a 2-cycle $h \rightarrow x \rightarrow h$ and a 3-cycle $x \rightarrow y \rightarrow x \rightarrow h \rightarrow x$). Assign templates:

$$T_{h \rightarrow x} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \quad (\text{strictly positive}), \quad T_{x \rightarrow h} = I, \quad T_{x \rightarrow y} = Q, \quad T_{y \rightarrow x} = Q,$$

where I is the identity and Q is the swap matrix from the preceding example. Then the single-edge word $w = (h \rightarrow x)$ is a Doeblin word (already strictly positive), so OG3 fails. However, the periodic tight path alternating $x \rightarrow y \rightarrow x \rightarrow y \rightarrow \dots$ avoids the vertex h entirely, so it avoids w entirely. In particular, the avoidance graph $\mathcal{A}_{\neg w}(H)$ has a directed cycle and $\text{OG4}(H, w)$ holds (Definition 12.1).

This shows that even in the “good” structural regime where a Doeblin word exists, bounded-gap recurrence of that *chosen* word is not automatic.

23.5 Nonvacuous good regime: OG1–OG4 fail and forcing succeeds

Finally, we give a compact example where the forcing mechanism is nontrivial. Let $V = \{h, x, y\}$ and edges

$$h \rightarrow x, \quad x \rightarrow h, \quad x \rightarrow y, \quad y \rightarrow h.$$

Assume these are the only outgoing edges from h (so h has outdegree 1). The SCC H on $\{h, x, y\}$ is strongly connected and aperiodic: there is a cycle of length 2 ($h \rightarrow x \rightarrow h$) and a cycle of length 3 ($h \rightarrow x \rightarrow y \rightarrow h$). Let $d = 2$ and $S(v) = \{1, 2\}$ for all v . Assign templates

$$T_{h \rightarrow x} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \quad T_{x \rightarrow h} = I, \quad T_{x \rightarrow y} = I, \quad T_{y \rightarrow h} = I.$$

Then $w = (h \rightarrow x)$ is a Doeblin word. Moreover, because h has a unique outgoing edge, *every* infinite path in H contains the edge $h \rightarrow x$ with gaps at most 3 (in fact at most the maximum return time to h). Equivalently, the avoidance graph $\mathcal{A}_{\neg w}(H)$ is acyclic, so RC1 holds and OG4 fails. Therefore the forcing dichotomy applies (Corollary 8.5), and Theorem 13.7 yields a uniform exponential contraction rate on all tight minimizers supported in H .

Remark 23.1 (What this demonstrates). *This example is intentionally simple: the Doeblin block is a single positive generator and recurrence is forced by a finite outdegree constraint. The point is not complexity but nonvacuity: the “OG1–OG4 fail $\Rightarrow$ contraction” regime is populated, and the forcing lemmas deliver explicit rates from finite witnesses.*

24 Attack zoo: counter-shapes that block naive implications

This section records explicit adversarial shapes demonstrating why several tempting implications are invalid without additional hypotheses. The constructions are deliberately small and checkable. Each one corresponds to a specific inference that must *not* be used elsewhere in the paper.

24.1 Strong connectivity of a union support digraph does not imply a Doeblin word

Naive (invalid) inference. If the union support digraph $\Gamma(H)$ on the active face is strongly connected, then there exists a word w whose support pattern is all-ones on that face.

Counter-shape (permutation semigroup). Let $d \geq 2$ and consider the following 0–1 support patterns on $\{1, \dots, d\}$:

$$P_\sigma(i, j) = \mathbf{1}_{\{j=\sigma(i)\}}, \quad \sigma \in S_d.$$

Each P_σ is a permutation pattern. Let $\mathcal{P}$ be any set of permutations that generates a transitive subgroup on $\{1, \dots, d\}$. Then the union support digraph

$$\Gamma := \bigcup_{\sigma \in \mathcal{P}} \text{supp}(P_\sigma)$$

is strongly connected (indeed, it contains a directed path from any state to any other by transitivity). However, every product of such patterns is again a permutation pattern:

$$P_{\sigma_1} P_{\sigma_2} \cdots P_{\sigma_k} = P_{\sigma_1 \circ \sigma_2 \circ \cdots \circ \sigma_k}.$$

In particular, no product support pattern can be the all-ones pattern. Thus there is *no* Doeblin word in the semigroup, despite Γ being strongly connected.

Concrete 2×2 instance. Take

$$P_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad P_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Then Γ is strongly connected on $\{1, 2\}$, but every product is either P_0 or P_1 , never all-ones.

Lesson. Strong connectivity of $\Gamma(H)$ is at most a *necessary pre-check*. Sufficiency for a Doeblin word must be certified by the product-automaton/all-ones reachability test.

24.2 Density-one tightness does not imply eventual confinement in a tight SCC

Naive (invalid) inference. If a minimizing trajectory uses non-tight edges with density 0 (equivalently, tight edges with density 1), then it is eventually contained in a single sink SCC of the tight graph.

Counter-shape (two tight SCCs with sparse slack transitions). Consider a directed graph with vertices $\{a, b\}$ and edges

$$(a \rightarrow a), (b \rightarrow b) \quad (\text{tight edges}), \quad (a \rightarrow b), (b \rightarrow a) \quad (\text{slack edges}).$$

Assign costs

$$\text{cost}(a \rightarrow a) = \text{cost}(b \rightarrow b) = 0, \quad \text{cost}(a \rightarrow b) = \text{cost}(b \rightarrow a) = 1.$$

Then the minimum cycle mean is $\lambda_* = 0$, realized by the tight self-loops. Define an infinite path by alternating long stays on each tight loop, with crossings at times n^2 :

$$\underbrace{a \rightarrow a \rightarrow \cdots \rightarrow a}_{n^2 \text{ steps}} \rightarrow b \rightarrow \underbrace{b \rightarrow b \rightarrow \cdots \rightarrow b}_{(n+1)^2 \text{ steps}} \rightarrow a \rightarrow \cdots$$

The number of slack crossings up to time T is $O(\sqrt{T})$, hence the slack-edge density is 0 and the average cost tends to 0. So this trajectory is an asymptotic mean minimizer. But it switches between two distinct tight SCCs infinitely often, and it is *not* eventually contained in either SCC.

Lesson. From density-1 tightness one may extract only a *positive-density* essential tight SCC visited along tight times. Any claim of eventual confinement requires an additional hypothesis (for example, an explicit tail-supported-in- H verification).

24.3 Positive-density visits to an SCC do not force positive-density occurrences of a fixed word

Naive (invalid) inference. If a trajectory remains in a strongly connected component (or visits it with positive lower density), then every fixed admissible word occurs with positive lower density.

Counter-shape (strongly connected full digraph, a word avoided). Let the state set be $\{0, 1\}$ with all directed edges allowed. This directed graph is strongly connected. Fix the word $w = 11$. The infinite path

$$0101010101 \dots$$

stays entirely in the SCC but contains no occurrence of 11. Similarly, for the word $w = 01$ the path $000000 \dots$ stays in the SCC and avoids 01.

Counter-shape (word occurs infinitely often but with zero lower density). Fix the word $w = 01$ and consider

$$0^1 1 0^2 1 0^3 1 0^4 1 \dots$$

This path stays in the same strongly connected graph. The word 01 occurs infinitely often (exactly once at the end of each $0^n 1$ block), but the gaps between occurrences grow and the lower density of 01 occurrences is 0.

Lesson. Strong connectivity (or even full support) is a reachability property, not a frequency property along a particular trajectory. Frequency conclusions require an explicit recurrence input (syndeticity, a quantitative lower density hypothesis, or a minimality/linearly recurrent theorem).

24.4 Sparse slack can allow unbounded gaps between certified Doeblin blocks

Naive (invalid) inference. If a Doeblin word exists on an aperiodic tight SCC H , then a general minimizer must see that word with bounded gaps.

Counter-shape (Doeblin exists in H , but the minimizer makes long excursions elsewhere). Construct a tight SCC H on a face $S(H)$ for which the product-automaton test certifies a Doeblin word w . Now enlarge the ambient graph by adding a second tight SCC K and two slack edges that permit switching:

$$H \xrightarrow{\text{slack}} K, \quad K \xrightarrow{\text{slack}} H.$$

Within each SCC, tight edges have reduced cost 0. The two inter-SCC edges carry reduced cost 1. Define a trajectory that spends n^2 steps in K , then pays one slack edge to return to H , spends a bounded number of steps in H (even seeing w once), then pays one slack edge to go back to K , and repeats.

The slack-edge density is still 0 (two slack edges per n^2 steps), so such a trajectory can be an asymptotic mean minimizer. But the gaps between occurrences of w along this trajectory are unbounded (they grow like n^2).

Lesson. A Doeblin witness on H does not force recurrence on *arbitrary* minimizers. Bounded-gap recurrence can be concluded only under a correctly scoped recurrence condition:

- for tight trajectories supported in H , via an avoid- w automaton acyclicity certificate (RC1), or
- for a chosen minimizing trajectory, via an explicitly supplied recurrence input (syndetic returns, quantitative lower density, or verified tail support in H).

24.5 Mapping from counter-shapes to prohibited inference patterns

For quick auditing, the counter-shapes above block the following prohibited inference patterns:

- **SC $\Gamma(H) \Rightarrow$ Doeblin word:** blocked by permutation-semigroup counter-shape (Section 24.1).
- **Density-1 tight $\Rightarrow$ eventual tight SCC confinement:** blocked by sparse slack switching (Section 24.2).
- **SCC visit $\Rightarrow$ positive density of a word:** blocked by word-avoidance and zero-density constructions (Section 24.3).
- **Doeblin exists in $H \Rightarrow$ bounded-gap returns on all minimizers:** blocked by long excursions with sparse slack (Section 24.4).

25 Minimality and sharpness: why the boundary is tight

This section records short “edge-case” constructions that explain why the hypotheses and obstruction gates in the boundary theorem cannot be substantially weakened without changing the conclusion. The goal is not to catalog counterexamples, but to isolate the minimal mechanisms that force the regime split.

25.1 Why tight-core reduction is not optional

The calibration potential ϕ and the induced tight graph $G^{(0)}$ are not merely a technical convenience. They are the device that makes the extremal problem finite and local. Without passing to tight edges, statements about minimizers are typically false if one reads them directly on the unreduced descriptor graph.

Remark 25.1 (A two-cycle tie shows why one must reduce). *Consider a finite graph with two disjoint cycles γ_1 and γ_2 that have the same cycle mean λ_* . Assume there are additional edges of slightly higher mean that connect the cycles in both directions. Then minimizing measures may be supported on either cycle, or on mixtures that switch between the two cycles with sublinear frequency. On the unreduced graph there is no canonical “core” and no unique active face. After calibration, the tight subgraph separates the true minimizer-supporting structure from slack edges; the subsequent obstruction tests are meaningful only on this tight reduction.*

25.2 How contraction fails: realizable obstructions

The obstruction gates OG2 and OG3 are designed to be witness-producing and to capture genuinely different failure modes. Both occur in simple finite-template systems.

- **OG2 (face-map inconsistency) is a real obstruction.** It is possible for tight edges to force incompatible coordinate support requirements. Concretely, one can have two tight words u and v in the same tight SCC such that u maps mass from an active face into a strict subface, while v maps mass into a different strict subface, and the SCC allows alternating concatenations. The witness produced by Lemma `lem:og2-witness-cycle` is exactly the finite inconsistency cycle that certifies that no consistent face map exists.
- **OG3 (no Doeblin word) is strictly stronger than “ $\Gamma(H)$ not strongly connected.”** Even when the union support digraph $\Gamma(H)$ is strongly connected, the allowed face-restricted patterns may generate a semigroup that never contains an all-positive block. The clean counter-shape is the “permutation-pattern” family: if the admissible patterns on the face are permutation matrices that generate a transitive action, then $\Gamma(H)$ is strongly connected but every product remains a permutation pattern, so no Doeblin word exists. This is why the certificate used in the paper is the Boolean-pattern product-automaton test: Doeblin existence is equivalent to reachability of the all-ones pattern state.

25.3 When imprimitive trapping is the right conclusion

The regime split treats aperiodicity (period 1) and imprimitivity (period $p > 1$) differently because the projective geometry is different. In an imprimitive tight SCC, the index set splits into cyclic classes and products respect that cyclic structure. Even with strong tightness and perfect calibration, one should not expect a single stationary projective section on the full face.

Remark 25.2 (Imprimitive tight cores can force cycling). *Let H be a tight SCC with period $p > 1$. Then there is a cyclic decomposition of states into classes $C_0, \dots, C_{p-1}$ such that every edge advances the class modulo p . Along an admissible tight trajectory, the support of mass on the active face can rotate among these classes. This produces a genuine periodic trapping phenomenon at the level of projective directions. Accordingly, the boundary theorem outputs “imprimitive trapping” (periodic regime) rather than uniform contraction on the full face.*

Taken together, these examples explain the sharpness of the theorem chain: the tight-core reduction is needed to make minimizers finite and local; OG2 and OG3 isolate distinct algebraic failures on the active face; and the imprimitive branch is the correct endpoint when the tight core has period > 1 .

26 Parameter Chambers and Generic Cycle Selection

This section formalizes what is generic in linear parameter families in a literal, checkable sense: uniqueness of the minimizing cycle mean off a finite hyperplane arrangement, together with a finite chamber decomposition of parameter space on which the minimizing set is constant. No claim of “generic aperiodicity” is made, since periodic/aperiodic is a combinatorial property of the selected critical SCC.

26.1 Linear parameter model and hyperplane arrangement

Fix a finite directed graph $G = (V, E)$ (after reduction) and a feature map $\psi : E \rightarrow \mathbb{Z}^k$. For $\theta \in \mathbb{R}^k$ define edge costs

$$c_\theta(e) = \langle \theta, \psi(e) \rangle,$$

and for a directed cycle γ define its mean cost

$$\mu_\theta(\gamma) := \frac{1}{|\gamma|} \sum_{e \in \gamma} c_\theta(e).$$

Let $\mathcal{C}$ be the finite set of simple directed cycles in G and let

$$\lambda_*(\theta) := \min_{\gamma \in \mathcal{C}} \mu_\theta(\gamma)$$

be the minimal cycle mean.

Definition 26.1 (Degeneracy hyperplanes). *For distinct cycles $\gamma \neq \gamma'$ define the affine functional*

$$F_{\gamma, \gamma'}(\theta) := \mu_\theta(\gamma) - \mu_\theta(\gamma').$$

The degeneracy arrangement is the finite union of hyperplanes

$$\mathcal{D} := \bigcup_{\gamma \neq \gamma'} \{\theta : F_{\gamma, \gamma'}(\theta) = 0\}.$$

Lemma 26.2 (Unique minimizing cycle off $\mathcal{D}$). *If $\theta \notin \mathcal{D}$, then there exists a unique cycle $\gamma_*(\theta) \in \mathcal{C}$ with $\mu_\theta(\gamma_*(\theta)) = \lambda_*(\theta)$.*

Proof. If two distinct cycles both attained the minimum, then $F_{\gamma, \gamma'}(\theta) = 0$ for that pair, so $\theta \in \mathcal{D}$. $\square$

26.2 Chamber decomposition and critical SCC stability

Definition 26.3 (Chambers). *A parameter chamber is a connected component of $\mathbb{R}^k \setminus \mathcal{D}$.*

Proposition 26.4 (Finite chamber structure). *$\mathbb{R}^k \setminus \mathcal{D}$ is a finite union of open polyhedral chambers. On each chamber, the minimizing cycle $\gamma_*(\theta)$ is constant.*

Proof. $\mathcal{D}$ is a finite arrangement of affine hyperplanes, so its complement has finitely many connected components, each described by a finite system of strict linear inequalities $F_{\gamma, \gamma'}(\theta) \leq 0$. Within a fixed component, the sign pattern of all $F_{\gamma, \gamma'}$ is constant, hence the unique minimizer γ_* is constant by Lemma 26.2. $\square$

Corollary 26.5 (Computable periodic/apperiodic regime on chambers). *Fix a chamber $\mathcal{U}$ and let γ_* be its minimizing cycle. Then the essential critical SCC of the tight graph is the SCC generated by γ_* (in particular, it is periodic). More generally, on lower-dimensional strata $\theta \in \mathcal{D}$ where multiple cycles tie, the tight critical subgraph may contain multiple cycles. It may be periodic or aperiodic depending on the gcd of cycle lengths in the resulting critical SCC. All of these properties are decidable from the finite tight geometry.*

Theorem 26.6 (Boundary type is constant on chambers). *Fix the graph G and feature map ψ as above, and consider the induced linear cost family c_θ . For each chamber $\mathcal{U}$ of $\mathbb{R}^k \setminus \mathcal{D}$, the following data are constant over $\theta \in \mathcal{U}$:*

- the minimizing cycle γ_* and minimal mean $\lambda_*(\theta)$ (hence the critical/tight SCC);
- the periodic/aperiodic regime of the essential critical SCC;
- the boundary classification output of the Tight-Core Boundary Theorem on that SCC (im-primitive trapping, or in the aperiodic case: certificate/Doeblin/contraction versus OG2/OG3 obstructions, together with the associated witness objects).

Consequently, the boundary type can change only when crossing the degeneracy arrangement $\mathcal{D}$.

Proof. On a chamber $\mathcal{U}$ the sign pattern of all cycle-mean differences $F_{\gamma, \gamma'}$ is fixed, so the minimizer γ_* is constant by Proposition 26.4. In the unique-minimizer case, the critical/tight subgraph is generated by the edges of γ_* , hence the essential critical SCC (and its periodicity) is constant. All subsequent boundary outputs in the Tight-Core Boundary Theorem are computed from finite graph data on the essential SCC and its induced tight geometry (certificate graph construction, SCC type, and the witness-producing obstruction tests), so they are constant once that finite geometry is fixed. Crossing a hyperplane in $\mathcal{D}$ is the only way the minimizer set can change, hence it is the only place where the induced tight/critical geometry and boundary type can change. $\square$

Edge cases on strata $\theta \in \mathcal{D}$. On the degeneracy arrangement, ties produce a critical subgraph with multiple cycles. The same finite pipeline applies, but the tight geometry must be formed from the tied minimizing set. For convenience, the most common tie patterns can be summarized as follows.

Situation on $\mathcal{D}$	What is decidable (and what to report)
Ties among cycles <i>within one</i> critical SCC	Compute the gcd of cycle lengths in that SCC. If the gcd is > 1 , the regime is periodic (trapping). If the gcd is 1, the SCC is aperiodic and the Tight-Core Boundary Theorem applies (certificate/Doeblin/contraction or OG2/OG3 with witnesses).
Ties split across <i>multiple</i> critical SCCs	The tight/critical graph decomposes into SCCs. Run the boundary classification on each essential SCC and report the resulting regime/witnesses; the overall boundary type is the finite list of SCC outcomes.
Higher-codimension ties (many cycles)	Form the tight subgraph induced by the minimizing set and apply the same SCC + certificate/obstruction pipeline. The output remains finite and witness-producing.

Remark 26.7 (What this does (and does not) justify). *In linear cost families, uniqueness of the minimizing cycle is generic in the precise sense “off $\mathcal{D}$.” That generic situation typically yields a periodic critical SCC. Accordingly, any claim of “generic aperiodicity” would require additional structure beyond the linear cycle-mean model. The robust message supported here is different: the periodic/aperiodic and certificate/obstruction boundary is finitely computable from the tight geometry, and by Theorem 26.6 the boundary type is constant on chambers and can change only across $\mathcal{D}$.*

References

- [1] V. D. Blondel and J. N. Tsitsiklis. The boundedness of all products of a pair of matrices is undecidable. *Systems & Control Letters* 41 (2000), 135–140.

- [2] I. D. Morris. Mather sets for sequences of matrices and applications to the study of joint spectral radii. *Proceedings of the London Mathematical Society* 107 (2013), 121–150.
- [3] V. Kozyakin. An annotated bibliography on the convergence of matrix products and the theory of joint/generalized spectral radius (JSRbib). Available at <https://kozyakin.github.io/jsrbib/JSRbib.html>. Accessed 2026-02-20.
- [4] I. D. Morris and N. Sidorov. On a devil’s staircase associated to the joint spectral radii of a family of pairs of matrices. *Journal of the European Mathematical Society*, 15(5):1747–1782, 2013. DOI: <https://doi.org/10.4171/JEMS/402>.
- [5] H. Don, H. Zantema, and M. de Bondt. Slowly synchronizing automata with fixed alphabet size. *Information and Computation*, 279:104614, 2021. DOI: <https://doi.org/10.1016/j.ic.2020.104614>.
- [6] A. N. Trahtman. Synchronization of some DFA. In *Theory and Applications of Models of Computation (TAMC 2007)*, Lecture Notes in Computer Science, vol. 4484, pages 234–243. Springer, 2007. DOI: https://doi.org/10.1007/978-3-540-72504-6_21.
- [7] S. W. Neufeld. A diameter bound on the exponent of a primitive directed graph. *Linear Algebra and its Applications*, 245:27–47, 1996. DOI: [https://doi.org/10.1016/0024-3795\(94\)00203-7](https://doi.org/10.1016/0024-3795(94)00203-7).
- [8] V. Yu. Protasov. The Barabanov norm is generically unique, simple, and easily computed. *SIAM Journal on Control and Optimization*, 60(4):2246–2267, 2022. DOI: <https://doi.org/10.1137/21M1426821>.
- [9] É. Charlier and J. Honkala. The freeness problem over matrix semigroups and bounded languages. *Information and Computation*, 237:243–256, 2014. DOI: <https://doi.org/10.1016/j.ic.2014.03.001>.

Open-problem family	Resolved statement in the certified class	Pipeline artifact
Černý-type synchronizing bounds for induced automata [5, 6]	For the specific automata generated by the certificate program (Doeblin-word search and avoidance automata), the pipeline provides explicit state counts and either a bounded-gap modulus B (acyclic avoidance) or an explicit cycle witness. When the induced automata satisfy aperiodicity conditions that are checkable from the transition monoid, known special-class synchronizing bounds apply, yielding a quantitative reset bound for this subclass.	Automaton model + modulus B (+ optional reset bound)
Quantitative primitivity / length of a positive (Doeblin) product [7]	The Doeblin-word search is finite and yields either a positive product witness with margin δ or a certificate of failure within the searched regime. In the certified setting, digraph exponent bounds and state-size bounds convert this into an explicit worst-case search radius for positivity on the core block.	Doeblin witness word (length L , margin δ)
Barabanov/extremal norm geometry and constructive computation [8]	A calibrated projective section together with recurrence (RC1) yields an explicit constructive extremal geometry: a computable contraction mechanism and robustness neighborhood for the extremal dynamics. This provides a concrete, checkable analogue of an extremal-norm package on the certified class.	Rate pack + calibrated section + robustness radii
Matrix-semigroup decision problems under bounded-language restrictions [9]	The certificate program restricts products to a constrained language (tight-core admissible words plus avoidance constraints). Within that restricted language, the key decision questions needed for stability reduce to finite automata and finite witnesses, giving an explicit decidability template compatible with bounded-language decidability phenomena.	Language model (tight-core + constraints)
Mapping undecidable vs. decidable frontiers for matrix products [1, 3]	The robust certificate contract isolates a decidable island and produces explicit exceptional sets where margins vanish. This provides an operational frontier map: the pipeline states exactly which failures prevent certification and returns corresponding finite witnesses.	Frontier report (OG / RU margins)

Table 2: How the robust certificate program resolves several open-problem families within the certified positive-template class (part 2).