MATHDISCOVERIES
Mathematical Preprint

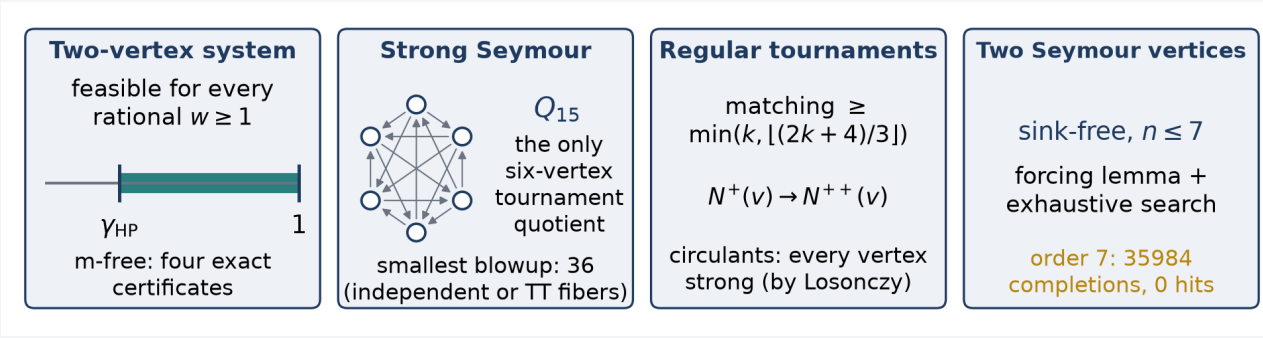
Working Draft

Seymour's Second Neighborhood Problem: Huang–Peng above γ_{HP}, Small Blowups, and Two Seymour Vertices through Order Seven

Tyler B. Hughes*,†

†Independent Researcher, United States

Disclosure: AI-assisted working draft directed by Tyler B. Hughes. Not for submission as is. No human mathematician has yet refereed the proof.



ABSTRACT: Seymour conjectured that every oriented graph has a vertex v with $d^{++}(v) \geq d^{+}(v)$; its 2026 proofs are preprints. For rational μ and w with $\gamma_{\mathrm{HP}} < \mu \leq 1 \leq w$, Huang and Peng's two-vertex system has a rational solution. Every m-free oriented graph, $m = 3, \ldots, 6$, has a vertex with $d^{++}(v) \geq 0.7203\, d^{+}(v)$ (Seacrest's are larger). A k-regular Seymour-tight oriented graph on at most $2k+2$ vertices has a Seymour-tight converse; a blowup of $C_n(1, \ldots, k)$ with nonempty fibers whose sizes have period $\gcd(n,k)$ is tight exactly when its fibers are. For $k \geq 1$, k-regular tournaments have a matching of $\min(k, \lfloor (2k+4)/3 \rfloor)$ arcs from $N^+(v)$ to $N^{++}(v)$. Of six-vertex tournaments only Q_{15} has a blowup with nonempty independent or transitive fibers and no strong Seymour vertex; such blowups have at least 36 vertices, and one has 36. Of oriented graphs on 1 to 6 vertices, only Q_{15} and Q_{14}. Nonempty sink-free oriented graphs of order $n \leq 7$ have at least two Seymour vertices. By Losonczy, circulant tournaments have only strong Seymour vertices. AI research agents built and directed by Tyler B. Hughes generated and checked this work. No human mathematician has yet refereed the proof.

Verification Scope (G11): WARN: Human referee: pending
Main results: 4 of 9 in Lean Kernel (44%) · 3 Cert Only · 2 written
Abstract claims: 5 of 10 in Lean Kernel (50%) · 3 Cert Only · 2 written

Full Trust Boundary: Section 5.

MSC 2020: 05C20, 05C69, 90C57, 68V20 **Keywords:** second neighborhood conjecture, Seymour vertex, oriented graphs, tournaments, exact certificates, Lean 4

1. INTRODUCTION AND MAIN RESULTS

Draw arrows between some pairs of points, never a loop and never both arrows between the same two points. Seymour conjectured in 1990 that some point has at least as many points exactly two arrows away as it has points one arrow away. Formally, an *oriented graph* D is a finite set of vertices with a set of arcs such that (v,v) is never an arc and (u,v), (v,u) are never both arcs. For a vertex v write $N^+(v)$ for the out-neighbors and $N^{++}(v)$ for the vertices other than v that are not in $N^+(v)$ and are reached by a path of two arcs; write $d^+(v) = |N^+(v)|$ and $d^{++}(v) = |N^{++}(v)|$. The *second neighborhood conjecture* (SNC) states that every nonempty oriented graph has a *Seymour vertex*, a vertex with $d^{++}(v) \geq d^{+}(v)$.

Status of the conjecture. The conjecture is stated in the paper of Dean and Latka [1] [to check]. Fisher proved it for tournaments [2] [to check], Havet and Thomassé proved that a tournament without a sink has two Seymour vertices by median orders [3] [to check], and Kaneko and Locke proved it for oriented graphs of minimum out-degree at most 6 [4] [to check]. Chen, Shen and Yuster [5] [to check] gave a vertex with $d^{++}(v) \geq \lambda d^{+}(v)$ for the positive root $\lambda = 0.6572\ldots$ of $2x^3 + x^2 - 1$ in every oriented graph, and Huang and Peng [6] raised the constant to $\gamma_{\mathrm{HP}} = 0.7155\ldots$, the root in $[0,1]$ of $8x^5 + 4x^4 - 12x^3 - 7x^2 + 2x + 4$. In September 2026 OpenAI posted a proof of the conjecture [12], with a Lean 4 file that we did not rebuild, and in November 2026 Bourneuf, Charbit, Coulomb and Thomassé [13] posted a short alternative proof. Both are preprints, not refereed as far as we know. If either is correct, it gives every nonempty oriented graph the constant 1, so Theorem 2 below is then weaker than a proved statement; the same holds for Theorem 9 if the statement that Bourneuf et al. announce without proof is correct. Theorems 3, 4, 5, 6, 7 and 8 concern other properties and do not follow from the conjecture.

The two-vertex constraint system. Huang and Peng show that an oriented graph without a γ-Seymour vertex gives,

for each weight $w \geq 1$, a real vector (which may depend on w) that satisfies an explicit system of inequalities (their Lemma 2.2). They prove that the system has no solution at $\mu = \gamma_{\mathrm{HP}}$ and $w = \gamma_{\mathrm{HP}}^2 + 2\gamma_{\mathrm{HP}}^3$; Lemma 18 is the step of that proof on their reduced family. In Section 2 we write the system as a feasibility problem in eleven variables $x_{11}, \dots, x_{34}$ with parameters μ and w and the quadratic $F(x; w)$ of Definition 16. They write, without a proof, that the constant is optimal under the constraints of their analysis; the next theorem proves the part that concerns the system with all variables nonnegative.

Theorem 1 (The Huang–Peng system is feasible above γ_{HP}). *For all rational numbers μ and w with $\gamma_{\mathrm{HP}} < \mu \leq 1$ and $w \geq 1$ there are nonnegative rational numbers*

$$x_{11}, x_{12}, x_{13}, x_{14}, x_{21}, x_{22}, x_{23}, x_{24}, x_{32}, x_{33}, x_{34}$$

with $x_{11} > 0$, $d_2 < \mu d_1$, $d_3 < \mu^2 d_1$, $x_{12} + x_{22} + x_{32} < \mu(x_{11} + x_{21})$, $x_{12} + x_{13} + x_{14} \leq x_{21}$ and $F(x; w) > 0$, where $d_1 = x_{11} + x_{12} + x_{13} + x_{14}$, $d_2 = x_{21} + x_{22} + x_{23} + x_{24}$ and $d_3 = x_{32} + x_{33} + x_{34}$.

Here F is the quadratic form of Definition 16. In other words, the system of their Lemma 2.2 is satisfiable for every rational μ in this range and every rational $w \geq 1$, so the two-vertex system by itself gives no bound beyond γ_{HP}. This is not a lower bound for γ_{HP} as a constant: it concerns a system, not a graph.

Liang and Xu [7] used the number of missing arcs in the out-neighborhood of a vertex to give a bound that depends on the length of the shortest directed cycle. An oriented graph is *m-free* if it has no directed cycle of length at most m. We put their lemma into the Huang–Peng system.

Theorem 2 (Bounds for m-free oriented graphs). *For $m = 3, 4, 5, 6$, every nonempty m-free oriented graph has a vertex v with $d^{++}(v) \geq c_m\, d^{+}(v)$, where $c_3 = 7203/10000$, $c_4 = 361/500$, $c_5 = 1807/2500$ and $c_6 = 3617/5000$.*

For $m = 3, 4, 5$ the constant c_m is larger than the root λ_m^{LX} of Liang and Xu, and for $m = 6$ it is smaller (Table 1). Seacrest [9] [to check] proved larger constants than ours for every m, the root of $x^m + x^{m-1} = 1$, so Theorem 2 shows what the two-vertex system gives and is weaker than the known bounds. If the SNC preprints are correct, the constant 1 holds for every oriented graph.

Converses and blowups. We call a graph *Seymour-tight* if $d^{++}(v) = d^{+}(v)$ at every vertex, and *Eulerian* if $d^{+}(v) = d^{-}(v)$ at every vertex. Guo, Kang and Zwaneveld [10] conjecture that every k-regular Seymour-tight orientation has a Seymour-tight converse (Conjecture 8.2 there [to check]), and prove it for $k = 1, 2$.

Theorem 3 (Converses of dense regular tight graphs). *Let k be a natural number and let D be a k-regular Seymour-tight oriented graph with at most $2k+2$ vertices. Then the converse of D is Seymour-tight.*

Theorem 4 (Circulant blowups with periodic fiber sizes). *Let $k \geq 1$ and $n \geq 2k + 1$, let $G_0, \dots, G_{n-1}$ be nonempty finite oriented graphs with s_i vertices, and let O be the lexicographic blowup of the circulant $C_n(1, \dots, k)$ by $G_0, \dots, G_{n-1}$. Put $d = \gcd(n, k)$. If $s_{i+d} = s_i$ for all i modulo n, then O is Seymour-tight if and only if every G_i is Seymour-tight, O is Eulerian if and only if every G_i is Eulerian, and the converse of O is Seymour-tight if and only if the converse of every G_i is. If every G_i has a Seymour vertex and O is Seymour-tight, then $s_{i+d} = s_i$ for all i modulo n.*

The implication from tight fibers to a tight blowup is the case of circulants of Theorem 4.6 and Lemma 4.7 of Guo, Kang and Zwaneveld [10] [to check]; the reverse implication, the Eulerian and converse statements, and the necessity of periodic sizes are added here.

Strong Seymour vertices. Bai, Li and Park [11] call v a *strong Seymour vertex* if the arcs from $N^{+}(v)$ to $N^{++}(v)$ contain a matching that covers $N^{+}(v)$. They proved that every oriented graph of minimum out-degree at most 5 has one, and Dzitsoev's 36-vertex blowup of the tournament Q_{15} shows that the statement fails for oriented graphs in general (Remark 3.1 there [to check]).

Theorem 5 (Matching bound for regular tournaments). *Let $k \geq 1$, let T be a tournament in which every vertex has out-degree and in-degree k, and let v be a vertex. Then the arcs from $N^{+}(v)$ to $N^{++}(v)$ contain a matching with $\min(k, \lfloor (2k+4)/3 \rfloor)$ arcs.*

Theorem 6 (The only six-vertex tournament quotient). *Let Q be a tournament on 6 vertices and n a vector of 6 positive integers. If $Q[n]$ or $Q[\mathrm{TT}_n]$ has no strong Seymour vertex, then Q is isomorphic to Q_{15} and $n_0 + \cdots + n_5 \geq 36$.*

Theorem 7 (All oriented quotients on one to six vertices). *Let Q be an oriented graph on one to six vertices. Then Q has a vector n of positive integers for which $Q[n]$ has no strong Seymour vertex exactly when Q is isomorphic to Q_{15} or to Q_{14}, and the same holds with $Q[\mathrm{TT}_n]$ in place of $Q[n]$.*

Theorem 8 (Circulant tournaments; a corollary of Losonczy's theorem). *Every vertex of every circulant tournament is a strong Seymour vertex.*

The graph Q_{14} is Q_{15} without the arc $(0, 5)$; the blowups are defined in Section 2. The uniqueness of Q_{15} among six-vertex tournaments and the minimum 36 were obtained earlier with exact certificates in the repository of njallskarp [27], whose Lean project covers the Hall criterion of blowups only; Theorem 6 is a Lean proof of a statement of the same content. Theorem 5 with unit weights was obtained by Gibbons [22], whose barrier $W \geq 5w(v) + 6\delta$ has the same consequence for regular tournaments. The statement for regular tournaments that every vertex is a Seymour vertex is Cary's Theorem 2 [18] [to check].

Theorem 8 is an immediate corollary of published work and is not a contribution of this paper. Losonczy [19] [to check] proved that for every abelian group G and every

nonempty finite $A \subseteq G \setminus \{0\}$ there is a bijection $\pi\colon A \to A$ with $a + \pi(a) \notin A$ for all $a \in A$; Eliahou and Lecouvey [20] (Theorem 3.2) proved the same for every group. Take $G = \mathbb{Z}_n$ and $A = S$, the connection set. For $u \in S$ and $b = \pi(u)$ we have $u + b \notin S$, and $u + b \neq 0$ since $-u \notin S$, so $u + b \in -S$. Hence $-b - u \in S$, that is, $u \to -b$, and $-b \in -S$ is neither 0 nor in S, so $-b \in N^{++}(0)$. The map $u \mapsto -\pi(u)$ is therefore the matching required at the vertex 0, and translations carry it to every vertex. We keep a direct written proof in Section 3, which does not use these theorems and was supplied by the blind referee of this draft, an AI model, and the proof of the finding, which covers prime orders only. A note in the repository of Labianco [25] proves with Kneser's theorem that every vertex of a Cayley oriented graph of an abelian group is a Seymour vertex [to check]; that is a statement on the sizes of N^{+} and N^{++}, not on matchings.

The number of Seymour vertices. A graph is *sink-free* if every vertex has an out-neighbor.

> **Theorem 9** (Two Seymour vertices through order seven). *Every nonempty sink-free oriented graph on at most 7 vertices has at least two Seymour vertices.*

In their preprint [13], Bourneuf, Charbit, Coulomb and Thomassé announce without proof that every nonempty sink-free oriented graph has at least two Seymour vertices; if that holds, it contains Theorem 9. The proof here is independent of it and of OpenAI's manuscript, and uses no weighted form of the conjecture; Section 3 also gives two lemmas on a unique Seymour vertex that assume the weighted form.

How the results are checked. Main results: 4 of 9 in Lean (44%) · 3 by certificate · 2 by written proof. The abstract covers the nine main results and Proposition 37. Section 5 lists every numbered statement with its tier and Lean declaration. Theorems 2, 7 and 9 rest on exact integer or rational programs in `cert/`, whose reductions are written proofs. Figure 1 shows how the parts fit together.

Concept in Action (verbatim in `recompute.py`)

```
from itertools import combinations
out, n = [50, 56, 11, 17, 36, 12], (7, 3, 11, 3, 9, 3)
N = lambda i: {j for j in range(6) if out[i] >> j & 1}
for i in range(6):
    far = {k for j in N(i) for k in N(j)} - N(i) - {i}
    S = [T for r in range(1, 7) for T in combinations(N(i), r)
         if sum(n[k] for k in far if any(k in N(j) for j in T))
         < sum(n[j] for j in T)]
    assert S  # a Hall violation at vertex i
assert sum(n) == 36  # no strong vertex
```

The program builds Q_{15}, whose out-neighborhoods are the bits of the masks, puts empty fibers of sizes n on its vertices, and finds for every fiber a set of out-fibers whose union is larger than the set of fibers that it reaches at distance two; by Lemma 34 no vertex is then a strong Seymour vertex. Section 4 checks the same blowup by the arcs of the 36-vertex graph itself.

Roadmap. Section 2 fixes the definitions in the form of the Lean library. Section 3 proves the main results and the lemmas behind them. Section 4 records the computations, and Section 5 lists what the Lean library proves.

2. DEFINITIONS, NOTATION, AND SETUP

All graphs are finite. An oriented graph may be empty, unless a statement asserts that a vertex exists, in which case it is assumed nonempty.

Definition 10 (Oriented graphs). An oriented graph on a vertex set V is a relation $u \to v$ with $v \not\to v$ for every v and not both $u \to v$ and $v \to u$. Put $N^{+}(v) = \{u : v \to u\}$, $N^{-}(v) = \{u : u \to v\}$, $d^{+} = |N^{+}|$, $d^{-} = |N^{-}|$; $N^{++}(v)$ is the set of $w \neq v$ with $w \notin N^{+}(v)$ and $v \to u \to w$ for some u, $N^{--}(v)$ is the same set in the converse (all arcs reversed), and $d^{++} = |N^{++}|$, $d^{--} = |N^{--}|$. The number of arcs is the sum of the out-degrees. The graph is a *tournament* if any two distinct vertices are joined by an arc, *strongly connected* if every vertex reaches every vertex, *Eulerian* if $d^{+} = d^{-}$ at every vertex, *k-regular* if $d^{+} = d^{-} = k$ at every vertex, and *isomorphic* to another if a bijection of the vertex sets preserves and reflects arcs.

Definition 11 (Seymour vertices, tight graphs, Pisa graphs). A vertex v is a Seymour vertex if $d^{+}(v) \leq d^{++}(v)$, and a tight vertex if $d^{+}(v) = d^{++}(v)$; the graph is Seymour-tight if all vertices are tight. A *Pisa graph*, after Halkiewicz [17], is strongly connected, satisfies $d^{++}(v) \leq d^{+}(v)$ for every v and has equality for some v. The slack of v is $\sigma(v) = d^{++}(v) - d^{+}(v)$.

Definition 12 (Strong Seymour vertices). A vertex v is a strong Seymour vertex if some injective map f from $N^{+}(v)$ into $N^{++}(v)$ satisfies $u \to f(u)$ for every u. The largest size of a matching of arcs from $N^{+}(v)$ into $N^{++}(v)$ is the maximum matching number of v.

Definition 13 (Weights). A weight on V is a function η from V to the positive rationals; $\eta(S)$ is its sum over S. The graph satisfies the *weighted second neighborhood property* for η if $\eta(N^{+}(v)) \leq \eta(N^{++}(v))$ for some v. For a vertex u_0 the weight that is $3/2$ at u_0 and 1 elsewhere is written η_{u_0}.

Definition 14 (Circulants, blowups, Hall violations). For $S \subseteq \mathbb{Z}_n \setminus \{0\}$ with $S \cap (-S) = \emptyset$, the circulant $C_n(S)$ has vertices $\mathbb{Z}_n$ and arcs $i \to i + s$, $s \in S$; $D_6 = C_6(1,2)$. For oriented graphs Q and G_i ($i \in V(Q)$) the lexicographic blowup $Q[(G_i)_i]$ has vertices (i,x) and arcs $(i,x) \to (j,y)$ when $i \to j$, or $i = j$ and $x \to y$ in G_i. $Q[n]$ is the blowup whose fiber over i has n_i vertices and no arc, and $Q[\mathrm{TT}_n]$ the blowup whose fibers are the transitive tournaments on n_i vertices with arcs from the smaller to the larger index. A Hall violation at i is a nonempty $S \subseteq N^{+}(i)$ with $\sum_{j \in N^{+}(S) \cap N^{++}(i)} n_j < \sum_{j \in S} n_j$, where $N^{+}(S) = \bigcup_{u \in S} N^{+}(u)$.

Definition 15 (Named graphs). The tournament Q_{15} has vertices $0, \dots, 5$ and 15 arcs; the out-neighborhoods of its vertices are given by the binary masks $50, 56, 11, 17, 36, 12$, in which the bit of j stands for the arc to j. $Q - (u,v)$ is Q without the arc (u,v), and $Q_{14} = Q_{15} - (0,5)$. The vectors $(7,3,11,3,9,3)$ and $(8,7,16,3,14,3)$ are those used at Q_{15} and Q_{14}. R_{11} is the 5-regular tournament on $0, \dots, 10$ with

(i) What was found	(ii) How it was found	(iii) Basis
Huang-Peng system feasible above γ_{HP}	rational solutions in two regimes	Lean
m-free constants for m = 3, 4, 5, 6	m-free form, exact 3x3 minors	Certificate
converses of dense regular tight graphs	counting at n = 2k+1 and 2k+2	Lean
tight circulant blowups	second differences of fiber sizes	Written
matching bound in regular tournaments	Hall and König counting	Lean
the only six-vertex tournament quotient	Hall, independent or TT fibers	Lean
all oriented quotients on 1 to 6 vertices	Hall systems, integer duals	Certificate
circulant tournaments: every vertex strong	corollary of Losonczy; direct proof	Written
sink-free, n ≤ 7: two Seymour vertices	forcing lemma, exhaustive search	Certificate
sink-free, n ≥ 8: one Seymour vertex?	open question (1)	Unsettled

Figure 1. Discovery and proof architecture. Left: what was found. Middle: the way each statement was found and checked. Right: the basis of each statement. Statements marked unsettled are questions, not claims of this paper.

arcs $0 \Rightarrow \{1,\dots,5\}$, $\{6,\dots,10\} \Rightarrow 0$, directed triangles $1,2,3$ and $8,9,10$, and $\{1,2,3\} \Rightarrow \{4,5,6,7\}$, $\{8,9,10\} \Rightarrow \{1,2,3\}$, $\{4,5,6,7\} \Rightarrow \{8,9,10\}$, $4 \to 5$, $6 \to 7$, $5 \to 6$, $5 \to 7$, $4 \to 6$, $7 \to 4$. P_6 is the graph on $0,\dots,5$ with the thirteen arcs $(0,4)$, $(0,5)$, $(1,4)$, $(1,5)$, $(2,0)$, $(2,1)$, $(3,0)$, $(3,1)$, $(4,2)$, $(4,3)$, $(5,2)$, $(5,3)$, $(5,4)$.

Definition 16 (The two-vertex system). Let μ and w be real. The system has variables x_{ij} as in Theorem 1, and

$$\begin{aligned} F(x;w) = {} & x_{21}(x_{32}-x_{11}-x_{13}) + x_{22}(x_{32}+x_{23}+x_{33}) \\ & + x_{23}x_{12} - x_{14}(x_{12}+x_{21}+x_{22}) + \tfrac{1}{2}(x_{21}+x_{22})^2 \\ & - \tfrac{1}{2}(wx_{11}^2 + x_{12}^2) + (w-1)x_{11}x_{12}. \end{aligned}$$

Put $d_1 = x_{11}+x_{12}+x_{13}+x_{14}$, $d_2 = x_{21}+x_{22}+x_{23}+x_{24}$ and $d_3 = x_{32}+x_{33}+x_{34}$. A vector is feasible if $x_{11} > 0$, the other ten coordinates are nonnegative, and $d_2 < \mu d_1$, $d_3 < \mu^2 d_1$, $x_{12}+x_{22}+x_{32} < \mu(x_{11}+x_{21})$, $x_{12}+x_{13}+x_{14} \le x_{21}$ and $F(x;w) > 0$. γ_{HP} is the root in $[0,1]$ of $P(x) = 8x^5+4x^4-12x^3-7x^2+2x+4$, which has exactly one there. The reduced family of Huang and Peng is, for $\gamma \neq 0$ and θ,

$$\begin{aligned} x_{11} &= -y_1 + (1+\gamma)y_2 + y_3 + y_4, & x_{12} &= \gamma(y_3 - y_1), \\ x_{21} &= \theta y_3 - \gamma y_2, & x_{22} &= \gamma(y_2 + y_4), \\ x_{32} &= \gamma\theta y_3, & x_{33} &= \gamma^2 y_4, \end{aligned}$$

$x_{13} = (1+\gamma)(y_1-y_2) + (\theta/\gamma - 1 - \gamma)y_3$ and $x_{14} = x_{23} = x_{24} = x_{34} = 0$.

Definition 17 (The m-free forms). Put $K = m-2$ and let $0 \le \alpha \le 1$, so that $1 + K\alpha(2-\alpha) \ge 1$. With F as above,

$$\begin{aligned} F_K(x;w) = {} & F(x;w) + K\alpha\, x_{11}x_{22} \\ & - \tfrac{wK}{2}(2\alpha x_{11}(x_{11}-x_{12}) - (\alpha x_{11})^2), \end{aligned}$$

$$w_K(\gamma) = \frac{2\gamma^3 + \gamma^2 + 2K\alpha\gamma}{1 + K\alpha(2-\alpha)},$$

$$\theta_K(\gamma) = 2 + 2K\alpha + 2\gamma + 2\gamma^2 - 2(1+K\alpha)w_K(\gamma),$$

for $K > 0$; for $K = 0$ these are $F_0 = F$, $w_0(\gamma) = \gamma^2 + 2\gamma^3$ and $\theta_0(\gamma) = 2 + 2\gamma - 4\gamma^3$. For $m = 3,4,5$ let $(K,\alpha) = (1,1/14), (2,1/25), (3,2/75)$ and let $\theta = \theta_K(\gamma)$ in the reduced family. Then $\gamma F_K(x(0,0,1,0); w_K(\gamma))$ is a polynomial in γ of degree 8, equal to $P_3^*/4873442$, $P_4^*/36300625$ and $P_5^*/2945775625$, where P_m^* has integer coefficients (listed in `recompute.py`, which checks these identities); γ_m^* is its only root in $[0.715, 0.720]$. The graph-theoretic L below is $x_{11} - x_{12} - x_{22}/w$.

3. PROOF OF THE MAIN RESULTS

The two-vertex system.

Lemma 18 (Nonpositivity at γ_{HP}, after Huang and Peng). *Let $\gamma = \gamma_{HP}$, $w = \gamma^2 + 2\gamma^3$ and $\theta = 2+2\gamma-4\gamma^3$. For all real y_1, y_2, y_3, y_4, if the point $x(y)$ of the reduced family satisfies $x_{21} - x_{12} - x_{13} \ge 0$, $x_{21} \ge 0$, $x_{22} \ge 0$, $x_{32} \ge 0$ and $x_{33} \ge 0$, then $F(x(y);w) \le 0$.*

This is their Lemma 3.1, stated and proved in Lean with an explicit sum-of-squares identity for $P_1 = x_{22}(x_{21} - x_{12} - x_{13})/\gamma$, $P_2 = x_{21}x_{33}/\gamma^3$, $P_3 = x_{32}x_{22}/(\gamma^2\theta)$ and $P_4 = x_{32}x_{33}/(\gamma^3\theta)$.

Proof of Theorem 1. The Lean theorem is `hp_primal_barrier_rational`. Put $w^*(\mu) = 2\mu^3 + \mu^2$. If $w < w^*(\mu)$, take $\delta = \min(\mu/4, (w^*(\mu) - w)/20)$ and $x_{11} = 1$, $x_{21} = \delta$, $x_{22} = \mu - 2\delta$, $x_{33} = \mu^2 - \delta$, all other variables 0. Then $d_2 < \mu d_1$, $d_3 < \mu^2 d_1$ and $F = \frac{1}{2}(w^*(\mu) - w) - \delta(1 + 2\mu + 2\mu^2) + \frac{5}{2}\delta^2 > 0$. If $w \ge w^*(\mu)$, put $\theta = 2 + 2\mu - 4\mu^3$ and $d_1 = \max(\theta/\mu, 1+\mu)$, and start from $x_{11} = 1$, $x_{12} = \mu$, $x_{13} = d_1 - 1 - \mu$, $x_{21} = \mu d_1$,

$x_{32} = \mu^2 d_1$. The derivative of F in w there is $\mu - 1/2 > 0$, and F at $w^*(\mu)$ is positive: if $d_1 = \theta/\mu$, then $\mu F = (\mu-1)(2\mu^2 + 2\mu + 1)P(\mu)$, where $P < 0$ on $(\gamma_{\mathrm{HP}}, 1]$; if $d_1 = 1 + \mu$, then $2F = \mu g(\mu)$ with $g(\mu) = 2\mu^4 + 9\mu^3 + 4\mu^2 - 3\mu - 4 > 0$ because then $\mu > 796/1000$, so g is positive. Finally subtract ε from x_{12}, $\frac{3}{2}\varepsilon$ from x_{21} and 3ε from x_{32} for a small rational ε, which makes the four strict inequalities strict and keeps $F > 0$ and $x_{12} + x_{13} + x_{14} \le x_{21}$. □

The remainder of this subsection concerns the bounds for m-free graphs. Let D be an oriented graph with no vertex v with $d^{++}(v) \ge \mu d^+(v)$, and suppose that it is minimal with respect to arcs. Huang and Peng choose a vertex u of minimum out-degree, put $X_1 = N^+(u)$ and $X_2 = N^{++}(u)$, and choose $v \in X_1$ that minimizes $w|N^+(v) \cap X_1| + |N^+(v) \cap X_2|$. Let X_3 be the set of vertices at distance 3 from u. For $i = 1, 2, 3$ and $j = 1, 2, 3$ let X_{ij} be the set of vertices of X_i at distance j from v, and let X_{i4} be the rest of X_i: the vertices of X_i at distance 4 or more from v or not reachable from it, and v itself when $i = 1$. Put $x_{ij} = |X_{ij}|$. Then $|X_i| = x_{i1} + x_{i2} + x_{i3} + x_{i4}$, and X_{31} is empty, because an out-neighbor of v is at distance 1 or 2 from u. Their Lemma 2.2 gives, for every $w \ge 1$, a nonnegative vector with the conditions of Theorem 1: the first three inequalities strict, $x_{12} + x_{13} + x_{14} \le x_{21}$, and $F > 0$, which comes from counting in two ways the arcs with tail in $Y = X_{11} \cup X_{12} \cup X_{21} \cup X_{22}$. We use two facts from the graph.

Lemma 19 (Out-degrees inside X_{11}). *Let D be a finite oriented graph, u and v vertices, $w > 0$, and suppose that $w|N^+(v) \cap X_1| + |N^+(v) \cap X_2| \le w|N^+(y) \cap X_1| + |N^+(y) \cap X_2|$ for every $y \in X_1$, where $X_1 = N^+(u)$ and $X_2 = N^{++}(u)$. Put $X_{11} = X_1 \cap N^+(v)$, $X_{12} = X_1 \cap N^{++}(v)$ and $X_{22} = X_2 \cap N^{++}(v)$. Then every $y \in X_{11}$ satisfies $w(|X_{11}| - |X_{12}|) - |X_{22}| \le w|N^+(y) \cap X_{11}|$.*

Proof. An out-neighbor of y in X_1 is in X_{11} or in X_{12}, and one in X_2 is an out-neighbor of v or in X_{22}, because $v \to y \to z$. So $|N^+(y) \cap X_1| \le |N^+(y) \cap X_{11}| + |X_{12}|$ and $|N^+(y) \cap X_2| \le |N^+(v) \cap X_2| + |X_{22}|$, and the minimality inequality for y gives the claim. The Lean proof is `internal_ outDeg_ concentration_X11`. □

Lemma 20 (A tangent line). *For all real $\alpha \ge 0$, $x_{11} > 0$ and L, $2(\alpha x_{11})L - (\alpha x_{11})^2 \le \max(0, L)^2$.*

Proof. If $L \ge 0$, the difference is $(L - \alpha x_{11})^2$; if $L < 0$, the left side is at most 0. (Lean: `max_sq_ge_tangent`.) □

Proposition 21 (A certificate at γ_3^*). *Let $K = 1$, $\alpha = 1/14$, $\gamma = \gamma_3^*$ (Definition 17), $w = w_K(\gamma)$ and $\theta = \theta_K(\gamma)$. For all real y_1, y_2, y_3, y_4, if $x(y)$ satisfies $x_{21} - x_{12} - x_{13} \ge 0$, $x_{21} \ge 0$, $x_{22} \ge 0$, $x_{32} \ge 0$ and $x_{33} \ge 0$, then $F_K(x(y); w) \le 0$.*

Proposition 22 (A certificate at γ_4^*). *The statement of Proposition 21 with $K = 2$, $\alpha = 1/25$ and $\gamma = \gamma_4^*$.*

Proposition 23 (A certificate at γ_5^*). *The statement of Proposition 21 with $K = 3$, $\alpha = 2/75$ and $\gamma = \gamma_5^*$.*

The three propositions are proved in Lean (`m3_sos_nonpos`, `m4_sos_nonpos`, `m5_sos_nonpos`) by sum-of-squares identities in the ten monomials of degree two, checked by the kernel. With $w = w_K(\gamma)$ the coefficient of y_4^2 in $F_K(x(y); w)$ is zero, and with $\theta = \theta_K(\gamma)$ the coefficient of $y_1 y_3$ is zero; the equation of γ makes the coefficient of y_3^2 zero.

Remark 24 (What the three propositions give). By the reduction in the proof of Theorem 2, a vertex with $d^{++}(v) \ge \gamma_m^* d^+(v)$ exists in every m-free oriented graph for $m = 3, 4, 5$, where $\gamma_3^* = 0.717933\ldots$, $\gamma_4^* = 0.718322\ldots$ and $\gamma_5^* = 0.718379\ldots$. The bounds of Theorem 2 are larger, so these values are used only in Table 1.

Proof of Theorem 2. Let $m \in \{3, 4, 5, 6\}$, $K = m - 2$, $\mu = c_m$, and suppose that some nonempty m-free oriented graph has no vertex with $d^{++}(v) \ge \mu d^+(v)$. Choose one with the fewest arcs. Removing arcs from an m-free graph keeps it m-free, so the proof of Huang and Peng's Lemma 2.1 applies inside the class of m-free graphs, and their Lemma 2.2 gives the system of Theorem 1, strict and weak inequalities as there, and, at (1) below, a stronger inequality than $F > 0$.

The penalty. Let t be the number of nonadjacent pairs in X_{11}. The out-degrees inside X_{11} satisfy, by Lemma 19 with $L = x_{11} - x_{12} - x_{22}/w$, $|N^+(y) \cap X_{11}| \ge \max(0, L)$. The graph on X_{11} is m-free and is obtained from a tournament by deleting t arcs, and Liang and Xu's Lemma 1.7 [to check] gives a vertex with at most $\sqrt{2t/K}$ out-neighbors in X_{11}. Hence $t \ge \frac{K}{2}\max(0, L)^2 \ge \frac{K}{2}(2\alpha x_{11} L - (\alpha x_{11})^2)$ for every $\alpha \ge 0$, by Lemma 20. The count behind $F > 0$ runs over all arcs with tail in Y, those inside X_{11} among them, so each of the t missing pairs inside X_{11} enters both of its bounds: together they raise the lower bound by $(w-1)t$ and lower the upper bound by t, and the conclusion $F > 0$ becomes

$$F(x; w) - wt > 0, \qquad \text{hence} \qquad F_K(x; w) > 0 \tag{1}$$

with F_K as in Definition 17.

The reduction. Huang and Peng's Lemma 2.3 changes a feasible vector in four steps so that the first, second and third inequalities become equalities and $x_{14} = x_{24} = x_{34} = x_{23} = 0$. With G the difference $F_K - F$, as a function of x, the steps act on G as follows. Steps 1 and 2 change neither x_{11}, x_{12} nor x_{22}, so G is unchanged. In step 3, x_{13} and x_{22} grow and x_{12} and x_{23} shrink by the same amount; the derivative of $F + G$ is at least $\mu^2 x_{11} - (w-1)(1 + K\alpha)x_{11} > 0$ because $(w-1)(1 + K\alpha) < \mu^2$. In step 4a, x_{21} shrinks and x_{22} grows, and G grows with derivative $K\alpha x_{11}$; in step 4b, x_{12} grows and x_{13} shrinks, and G grows with derivative $wK\alpha x_{11}$. So the four steps keep the conclusion $F_K > 0$. The reduced vector lies in the family of Definition 16 with $\gamma = \mu$ for every $\theta > 0$, not only for one: given $\theta > 0$, put $y_3 = x_{32}/(\gamma\theta)$, $y_4 = x_{33}/\gamma^2$, $y_1 = y_3 - x_{12}/\gamma$ and $y_2 = x_{22}/\gamma - y_4$. Then x_{12}, x_{22}, x_{32} and x_{33} have the values of the family; the second equality gives $d_1 = \theta y_3/\gamma + y_4$, the first then gives $x_{21} = \theta y_3 - \gamma y_2$, the third gives $x_{11} = -y_1 + (1 + \gamma)y_2 + y_3 + y_4$, and $x_{13} = d_1 - x_{11} - x_{12}$ is the value of the family. So $x = x(y)$, whichever $\theta > 0$ the certificate below chooses.

The certificate. Choose $w = w_K(\mu) + \Delta w$ with $\Delta w > 0$ and θ so that the coefficient of $y_1 y_3$ in $F_K(x(y); w)$ is zero; let $i_4 = x_{21} - x_{12} - x_{13}$ in the family, the linear form $-y_1 + y_2 + \rho y_3$ with $\rho = 1 + \theta - \theta/\mu$. The three products $i_4(y_2 + y_4)$, $(y_2 + y_4)y_3$ and $y_3 y_4$ are nonnegative, being equal to $x_{22}(x_{21} - x_{12} - x_{13})/\gamma$, $x_{22}x_{32}/(\gamma^2\theta)$ and $x_{32}x_{33}/(\gamma^3\theta)$. Write c_{ij} for the coefficient of $y_i y_j$ in $F_K(x(y); w)$, and put $m_3 = c_{14}$, $m_1 = -(c_{23} + \rho m_3)$

and $m_2 = -(c_{34} + \rho m_3 + m_1)$. Then

$$F_K(x(y); w) + m_3\, i_4(y_2 + y_4) + m_1(y_2 + y_4)y_3 + m_2 y_3 y_4 = c_{33} y_3^2 + (y_1, y_2, y_4) M (y_1, y_2, y_4)^T,$$

where c_{33} is the coefficient of y_3^2 and M is the symmetric 3×3 matrix of the remaining terms in $y_1^2, y_1y_2, y_2^2, y_2y_4, y_4^2$, whose $(1,3)$ entry is zero: the terms in y_1y_3, y_1y_4, y_2y_3 and y_3y_4 cancel. For the four choices of $(\mu, \alpha, \Delta w)$ in Table 1, exact rational arithmetic in `cert/sylvester.py` gives $w > 1$, $\theta > 0$, $(w-1)(1+K\alpha) < \mu^2$, $m_1, m_2, m_3 > 0$, $c_{33} \le 0$ and leading principal minors of M with signs $-, +, -$, so M is negative definite and $F_K(x(y); w) \le 0$. This contradicts (1). □

The perturbation $\Delta w > 0$ is needed: for $m = 3$ and the same μ and α with $\Delta w = 0$ the third leading principal minor of M, its determinant, is positive, so M is invertible but not negative definite and the certificate fails, as `cert/sylvester.py` also checks. With $\Delta w = 0$ and the values of α of Definition 17 (which differ from those of Table 1) the family gives the numbers γ_m^* of Propositions 21–23.

The decimals of the irrational roots are the first digits of the twelve-digit brackets computed by `recompute.py`. The Lean library proves the orderings behind the column of γ_m^*: $\max(\lambda_m^{\mathrm{LX}}, \gamma_{\mathrm{HP}}) < \gamma_m^* < \lambda_m^{\mathrm{S}}$ for $m = 3, 4, 5$ and $\gamma_3^* < \gamma_4^* < \gamma_5^* < \lambda_6^{\mathrm{LX}}$ (`theorem12_m3_ordering`, `theorem12_m4_ordering`, `theorem12_ m5_ ordering`, `theorem12_ m6_ crossover`). Only these orderings, the irrational numbers of the table and the signs of the certificate are claimed; the table does not claim that any bound is the best possible.

Converses and circulant blowups.

Lemma 25 (Reciprocity). *In every finite oriented graph, the sum over the vertices of $d^{--}(v)$ equals the sum over the vertices of $d^{++}(v)$.*

Proof. Both sums count the ordered pairs (u, w) with $u \ne w$, $u \not\to w$ and $u \to z \to w$ for some z: the pair contributes to u's second out-neighborhood and to w's second in-neighborhood. (Lean: `sum_ secondInDeg_ eq_ sum_ secondOutDeg`.) □

Lemma 26 (One inequality suffices). *Let D be a finite Eulerian Seymour-tight oriented graph. The converse of D is Seymour-tight if and only if $d^{--}(v) \le d^-(v)$ for every vertex v.*

Proof. By Lemma 25 and $d^{++} = d^+ = d^-$, $\sum_v (d^{--}(v) - d^-(v)) = \sum_v (d^{++}(v) - d^+(v)) = 0$; a sum of nonpositive numbers that is zero has all terms zero, and the converse is Seymour-tight exactly when $d^{--}(v) = d^-(v)$ at every vertex. (Lean: `converse_seymourTight_iff_le`.) □

Proof of Theorem 3. The proof is in Lean (`dense_ regular_ converse_seymourTight`). If $k = 0$ there are no arcs and both graphs are tight. Let $k \ge 1$ and $n = |V|$. For each v the sets $\{v\}$, $N^+(v)$ and $N^{++}(v)$ are disjoint, so $n \ge 2k+1$. If $n = 2k+1$, then $d^{--}(w) \le n-1-d^-(w) = k$ for every w, and Lemma 26 applies. Let $n = 2k+2$. Then each v has exactly one vertex $\pi(v)$ at distance at least 3 from it. Fix w and put $T(w) = V \setminus (\{w\} \cup N^-(w))$, a set of $k+1$ vertices. If $\pi(v) = w$, then $v \in T(w)$ and no out-neighbor of v lies in $\{w\} \cup N^-(w)$, so $N^+(v) \subseteq T(w) \setminus \{v\}$; both sets have k elements, so they are equal. Two different v, v' with $\pi(v) = \pi(v') = w$ would then dominate each other. So π is injective, hence a bijection, and each w has exactly one vertex at distance at least 3 to it. The vertices at distance exactly 2 to w are then $n-1-k-1 = k$ in number, so $d^{--}(w) = k = d^-(w)$. □

Proposition 27 (Trap bound). *Let $k \ge 1$ and let D be an oriented graph on $2k+3$ vertices in which every vertex has out-degree k and in-degree k. For a vertex w let $M(w)$ be the set of vertices from which w is at distance at least 3. Then $|M(w)| \le 3$, and if $|M(w)| = 3$ then $D[M(w)]$ is a directed triangle and every vertex of $M(w)$ dominates every vertex of $T(w)$ outside $M(w)$, where $T(w)$ is the set of vertices other than w that do not dominate w.*

Proof. $|T(w)| = k+2$. A vertex $v \in M(w)$ is in $T(w)$, and all its out-neighbors are in $T(w)$, because an out-neighbor that dominates w would put v at distance at most 2 from w. So v sends at most $k+2-|M(w)|$ arcs to $T(w) \setminus M(w)$ and at least $|M(w)|-2$ arcs inside $M(w)$. Summing over $M(w)$ with $s = |M(w)|$ gives $s(s-2) \le \binom{s}{2}$, hence $s \le 3$; if $|M(w)| = 3$ there are exactly three arcs inside and each vertex has one out-neighbor and one in-neighbor there, a directed triangle, and then each vertex of $M(w)$ has $k-1$ out-neighbors in $T(w) \setminus M(w)$, which has $k-1$ elements. □

Proposition 28 (Nine vertices). *Let D be a 3-regular Seymour-tight oriented graph with 9 vertices. Then the converse of D is Seymour-tight.*

Proof. The proof is in Lean (`regular_ three_ order_ nine_ converse_seymourTight`). By Lemma 25, $\sum_w (d^{--}(w) - 3) = 0$, so if the converse is not tight some w has $d^{--}(w) < 3$, that is $|M(w)| = 8 - 3 - d^{--}(w) \ge 3$. By Proposition 27, $M(w) = \{a, b, c\}$ is a directed triangle and the set $R = T(w) \setminus M(w)$ of two vertices receives all arcs from $\{a, b, c\}$, so each $r \in R$ already has three in-neighbors in $M(w)$ and $w \not\to r$. As $N^+(w) \subseteq T(w)$ has three elements, $N^+(w) = \{a, b, c\}$, and $N^{++}(w) \subseteq R$ has at most two elements, against the tightness of w. □

Proof of Theorem 4. Because $n \ge 2k+1$, the sets $\{i+1, \dots, i+k\}$, $\{i+k+1, \dots, i+2k\}$ and $\{i\}$ are disjoint, and a path of two arcs from i ends in $\{i+2, \dots, i+2k\}$. So a vertex (i, x) has out-degree $\sum_{j=1}^{k} s_{i+j} + d^+_{G_i}(x)$ and second out-degree $\sum_{j=k+1}^{2k} s_{i+j} + d^{++}_{G_i}(x)$, and $\sigma_O(i, x) = \sigma_{G_i}(x) - \Delta_i$ with $\Delta_i = \sum_{j=1}^{k} s_{i+j} - \sum_{j=k+1}^{2k} s_{i+j}$. If the sizes have period $d = \gcd(n, k)$, each of the two sums is the sum over a complete set of residues, so $\Delta_i = 0$, and tightness, the Eulerian property and (applied to the converses) tightness of the converse are the same for O and all G_i. Conversely, $\sum_i \Delta_i = 0$; if every G_i has a Seymour vertex and O is tight, then $\Delta_i \ge 0$ for all i from the tight vertex of G_i, so $\Delta_i = 0$ and every G_i is tight, and $\Delta_i - \Delta_{i+1} = s_{i+1} - 2s_{i+k+1} + s_{i+2k+1} = 0$. Along the cycle $j, j+k, j+2k, \dots$, which runs through the coset $j + d\mathbb{Z}_n$, the second differences of s vanish, so s is an arithmetic progression along a closed cycle, hence constant there: $s_{i+d} = s_i$. For the Eulerian property and the converse, the in-degree of (i, x) is $\sum_{j=1}^{k} s_{i-j} + d^-_{G_i}(x)$, and the converse of O is the blowup of $C_n(-1, \dots, -k) \cong C_n(1, \dots, k)$ by the converses of the G_i, so the same computation applies. □

Table 1. Bounds for m-free oriented graphs. λ_m^{LX} is the root of Liang and Xu's polynomial, γ_m^* the value of Remark 24, c_m the bound of Theorem 2 with its parameters $(\alpha, \Delta w)$, and λ_m^{S} the root of $x^m + x^{m-1} = 1$ of Seacrest.

m	λ_m^{LX}	γ_m^*	c_m	$(\alpha, \Delta w)$	λ_m^{S}
3	0.682327 …	0.717933 …	7203/10000	(13/50, 23/2000)	0.754877 …
4	0.700721 …	0.718322 …	361/500	(7/50, 17/2000)	0.819172 …
5	0.715225 …	0.718379 …	1807/2500	(1/10, 17/2000)	0.856674 …
6	0.727162 …	—	3617/5000	(2/25, 19/2000)	0.881271 …

Proposition 29 (Blowups of D_6). *Let $D_6 = C_6(1,2)$ and let $G_0, \dots, G_5$ be nonempty finite oriented graphs with $|G_{i+3}| = |G_i|$ for all i, such that for each $r \in \{0,1,2\}$ one of G_r and G_{r+3} has a Seymour vertex. Then $D_6[G_0, \dots, G_5]$ is Eulerian and Seymour-tight if and only if all G_i have the same number of vertices and every G_i is Eulerian and Seymour-tight; and if all G_i have the same number of vertices, the converse of $D_6[G_0, \dots, G_5]$ is Seymour-tight if and only if the converse of every G_i is.*

This is proved in Lean (`lexProd_D6_period3_rigidity`). The second statement does not use the hypothesis on Seymour vertices, and the first uses a weaker one than Theorem 4, a Seymour vertex in one fiber of each antipodal pair.

Proposition 30 (Census of converses). *Every Eulerian Seymour-tight oriented graph on one to six vertices (21 up to isomorphism), every labeled one on 7 vertices (7959) and every labeled 3-regular one on 8 vertices (5040) has a Seymour-tight converse.*

Proposition 30 is the exhaustive enumeration of `cert/census.py`; it is a statement about the listed orders only.

Strong Seymour vertices.

Proof of Theorem 5. The tournament has $2k+1$ vertices. Fix v, put $A = N^+(v)$ and $B = N^-(v)$, both of size k. Every $z \in B$ is in $N^{++}(v)$: otherwise z dominates v and all of A and has out-degree at least $k+1$. So $N^{++}(v) = B$. Suppose the largest matching from A to B has $k - \delta$ arcs, $\delta \geq 1$. By the Hall–König theorem some nonempty $S \subseteq A$ has $W = N^+(S) \cap B$ with $|W| = |S| - \delta$. Each vertex of S sends its k arcs inside S, to $A \setminus S$ or to W, so $|S|k \leq \binom{|S|}{2} + |S|(k - |S|) + |S||W|$, which gives $|S| \geq 2\delta + 1$. Each $z \in B \setminus W$ dominates v and every vertex of S, so summing out-degrees over $B \setminus W$, $(k - |W|)k \geq (k - |W|)(1 + |S|) + \binom{k-|W|}{2}$, which gives $k \geq 1 + |S| + \delta \geq 3\delta + 2$. Hence $\delta \leq \lfloor (k-2)/3 \rfloor$ and $k - \delta \geq \lfloor (2k+4)/3 \rfloor$. The Lean proof is `regular_tournament_matching_lower_bound`. □

Corollary 31 (Small regular tournaments). *If $1 \leq k \leq 4$, every vertex of every tournament in which every vertex has out-degree and in-degree k is a strong Seymour vertex.*

(Lean: `regular_tournament_all_strong_of_le_four`; from the bound, $\min(k, \lfloor (2k+4)/3 \rfloor) = k$ for these k.)

Proposition 32 (The eleven-vertex example). *In the tournament R_{11} on 11 vertices, a largest matching of arcs from $N^+(0)$ to $N^{++}(0)$ has exactly 4 arcs.*

Proposition 33 (The other vertices). *Every vertex of R_{11} other than 0 is a strong Seymour vertex.*

Both are Lean theorems (`R11Star_vertex0_exact_matching`, `R11Star_other_vertices_strong`). R_{11} is 5-regular and the bound of Theorem 5 is 4 for $k = 5$, so the first shows that the bound is attained for $k = 5$.

Lemma 34 (Hall criterion for empty blowups). *Let Q be a finite oriented graph and n a vector of positive integers indexed by its vertices. Then $Q[n]$ has a strong Seymour vertex if and only if some vertex i of Q has no Hall violation.*

Lemma 35 (Hall criterion for transitive fibers). *The statement of Lemma 34 with $Q[n]$ replaced by $Q[\mathrm{TT}_n]$.*

Both are Lean theorems (`emptyBlowup_exists_strong_iff`, `ttBlowup_exists_strong_iff`); for the first, a vertex (i, x) of $Q[n]$ is strong exactly when the bipartite graph between the fibers of $N^+(i)$ and those of $N^{++}(i)$ has a matching that covers all of the former, and Hall's theorem, with weights equal on fibers, is the criterion.

Proposition 36 (Quotients with at most five vertices). *For $k = 1, \dots, 5$, every tournament Q on k vertices and every vector n of positive integers, both $Q[n]$ and $Q[\mathrm{TT}_n]$ have a strong Seymour vertex.*

Proposition 37 (Order 36). *For $n = (7,3,11,3,9,3)$, the blowups $Q_{15}[n]$ and $Q_{15}[\mathrm{TT}_n]$ have 36 vertices, 509 and 630 arcs, and no strong Seymour vertex.*

Proposition 38 (Deleting one arc). *Let (a,b) be an arc of Q_{15} and n a vector of 6 positive integers. If (a,b) is not $(0,5)$, both blowups of $Q_{15} - (a,b)$ by n have a strong Seymour vertex. If $(a,b) = (0,5)$ and one of the two blowups has no strong Seymour vertex, then $n_0 + \cdots + n_5 \geq 51$.*

Proposition 39 (Order 51). *Let $n = (8,7,16,3,14,3)$. The blowups $Q_{14}[n]$ and $Q_{14}[\mathrm{TT}_n]$ have no strong Seymour vertex, and $Q_{14}[n]$ has 51 vertices and 985 arcs.*

Proposition 40 (Deleting two arcs). *For any two different arcs of Q_{15} and any vector n of 6 positive integers, both blowups of Q_{15} minus the two arcs by n have a strong Seymour vertex.*

Propositions 36, 37, 38, 39 and 40 and Theorem 6 are Lean theorems. Each works over all sizes at once: for every quotient, the violations of Lemma 34 over all Hall sets give a finite set of systems $Vn \geq 1$, and each system without a solution carries an integer dual vector $y \geq 0$, $y \neq 0$, with $yV \leq 0$, checked in the kernel, while the remaining systems are bounded below by a dual bound. The orbits of all tournaments under relabeling are listed by an orbit-counting argument inside the library. The order 36 in Theorem 6 and the orders 51 in Propositions 38 and 39 are attained by the vectors of the two propositions with witnesses.

Proof of Theorem 7. By Lemmas 34 and 35 the two blowups $Q[n]$ and $Q[\mathrm{TT}_n]$ have a strong Seymour vertex for the same vectors n, so it suffices to treat $Q[n]$. Let Q be an oriented graph on at most 6 vertices with a vector n such that $Q[n]$ has no strong Seymour vertex, and let Q_0 be a terminal strongly connected component of Q, one that no arc leaves.

For $i \in Q_0$ the sets $N^+(i)$ and $N^{++}(i)$ are the same in Q and in Q_0, so every vertex of Q_0 has a Hall violation for the restriction of n, and $Q_0[n|_{Q_0}]$ has no strong Seymour vertex. The program `cert/quotients.py` lists the 4314 strongly connected oriented graphs on one to six vertices up to isomorphism and shows that only Q_{15} and Q_{14} admit such a vector: the graph with one vertex has no Hall violation at all, and for each other graph except Q_{15} and Q_{14}, each choice of one undominated Hall violation per vertex gives a system $Vn \geq 1$ that an integer vector $y \geq 0$, $y \neq 0$, with $yV \leq 0$ rules out, and every such y is checked by the program; for Q_{15} and Q_{14} some choices (twelve and eight) have positive integer solutions. So Q_0 is Q_{15} or Q_{14}, which have 6 vertices, and $Q = Q_0$. Conversely, Propositions 37 and 39 give such vectors for Q_{15} and Q_{14}. □

Theorem 8 follows from Losonczy's theorem, as shown in Section 1; what follows is a direct argument. Its transform is the Dyson transform, which is also among the tools of Losonczy's proof according to Eliahou and Lecouvey [20].

Direct proof of Theorem 8. Let n be odd and S the connection set, so $S \cap (-S) = \emptyset$ and $S \cup (-S) = \mathbb{Z}_n \setminus \{0\}$. Translations are automorphisms, so it suffices to consider the vertex 0, where $N^+(0) = S$. An arc $a \to z$ with $a \in S$ and $z \in -S$ ends in $N^{++}(0)$, since $z \neq 0$ and $z \notin S$. By Hall's theorem it suffices to show that every nonempty $A \subseteq S$ has at least $|A|$ out-neighbors in $-S$.

Let B consist of 0 and of the $t \in S$ such that no $a \in A$ has $a \to -t$. Then $A + B \subseteq S$: $a + 0 = a$, and for $t \in B \setminus \{0\}$ we have $a + t \neq 0$, since $-t \notin S$, and $-(a+t) = -t - a \notin S$, since $a \not\to -t$, so $a + t \in S$. We show $|A| + |B| \leq |S| + 1$. Call a pair (X, Y) of subsets of $\mathbb{Z}_n$ admissible if $0 \in Y \subseteq B$, $X + Y \subseteq A + B$ and $|X| + |Y| = |A| + |B|$; the pair (A, B) is admissible. If (X, Y) is admissible and $x \in X$ has $x + Y \not\subseteq X$, then $X' = X \cup (x + Y)$ and $Y' = Y \cap (X - x)$ give an admissible pair with $|X'| > |X|$: $0 \in Y'$ because $x \in X$; $|X'| = |X| + |Y| - |Y'|$ because $(x + Y) \cap X = x + Y'$; and $X' + Y' \subseteq X + Y$ because $(x + y) + y' = (x + y') + y$ with $x + y' \in X$. Take an admissible pair with $|X|$ largest. Then $X + Y \subseteq X$, so $X + y = X$ for every $y \in Y$. As $0 \in Y$, $X \subseteq X + Y \subseteq A + B \subseteq S$, so $0 \notin X$; and if $y \in X \cap Y$, then $X - y = X$ would contain 0. So X and $Y \setminus \{0\}$ are disjoint subsets of S, and $|A| + |B| = |X| + |Y| \leq |S| + 1$. Finally the elements $-t$ with $t \in S \setminus B$ are out-neighbors of A in $-S$, and there are $|S| - (|B| - 1) \geq |A|$ of them. □

For prime order the finding's proof, which we keep, uses two theorems of additive number theory instead.

Second proof of Theorem 8 for prime order. Let $p = 2k + 1$ be a prime and S the connection set, so $S \cap (-S) = \emptyset$, $S \cup (-S) = \mathbb{Z}_p \setminus \{0\}$ and $|S| = k$. Translations are automorphisms, so it suffices to consider the vertex 0, where $N^+(0) = S$ and $N^{++}(0) \subseteq -S$. We show that every nonempty $A \subseteq S$ satisfies $|(A+S) \cap (-S)| \geq |A|$; by Hall's theorem this gives a matching that covers S. Note that $0 \notin A + S$, since $a + s = 0$ would put $-s$ in S.

If $|A| = 1$, then $a \notin a + S$, so $|(a + S) \cap S| \leq k - 1$ and $|(a + S) \cap (-S)| \geq 1$. If $A = S$, let $y \in -S$; the k elements of $y - S$ are different from 0 and from y, so they cannot all lie in $-S \setminus \{y\}$, which has $k - 1$ elements, and some lies in S; so $y \in S + S$ and $-S \subseteq S + S$. Let $2 \leq |A| \leq k - 1$. The theorem of Cauchy and Davenport ([21], [to check]) gives $|A + S| \geq |A| + k - 1$. If $|A + S| \geq |A| + k$, then $|(A + S) \cap (-S)| = |A + S| - |(A + S) \cap S| \geq |A|$. Otherwise $|A + S| = |A| + k - 1 \leq 2k - 2 = p - 3$, and Vosper's theorem ([21], [to check]) says that A and S are arithmetic progressions with the same difference e. Multiplication by e^{-1} and, if needed, by -1 are isomorphisms of circulant tournaments, after which $S = \{1, \dots, k\}$ (the only progressions of difference 1 with $S \cup (-S) = \mathbb{Z}_p \setminus \{0\}$ are $\{1, \dots, k\}$ and its negative) and $A = \{a, \dots, a + |A| - 1\}$ with $1 \leq a$ and $a + |A| - 1 \leq k$. Then $A + S = \{a + 1, \dots, a + |A| + k - 1\}$, $|(A + S) \cap S| = k - a$ and $|(A + S) \cap (-S)| = |A| + a - 1 \geq |A|$. □

The two places of the second proof that use that p is prime are the two cited theorems.

Proposition 41 (Circulant tournaments of odd order at most 21). *At vertex* 0 *of each of the* 2046 *circulant tournaments of odd order from* 3 *to* 21, *the arcs from* $N^+(0)$ *to* $N^{++}(0)$ *contain a matching covering* $N^+(0)$.

The exhaustive check is `cert/circulants.py`. By the vertex transitivity of circulants the same holds at every vertex; the proposition is stated for vertex 0 and the check is at vertex 0.

The number of Seymour vertices.

Lemma 42 (Zero slack and multiplicity). *Let D be a finite oriented graph, v_0 a vertex with $|N^{++}(v_0)| = |N^+(v_0)|$ and u_0 an out-neighbor of v_0. If D satisfies the weighted second neighborhood property for the weight that is* 3/2 *at u_0 and* 1 *elsewhere, then D has a Seymour vertex other than v_0.*

Proof. The Lean theorem is `zero_ slack_ multiplicity_ lemma`. Since $u_0 \notin N^{++}(v_0)$, $\eta_{u_0}(N^+(v_0)) = d^+(v_0) + \frac{1}{2} > d^{++}(v_0) = \eta_{u_0}(N^{++}(v_0))$. So the vertex v_1 with $\eta_{u_0}(N^+(v_1)) \leq \eta_{u_0}(N^{++}(v_1))$ is not v_0, and $d^{++}(v_1) + \frac{1}{2} \geq \eta_{u_0}(N^{++}(v_1)) \geq \eta_{u_0}(N^+(v_1)) \geq d^+(v_1)$. As these are integers, $d^{++}(v_1) \geq d^+(v_1)$. □

Corollary 43 (A unique Seymour vertex). *Let D be a finite oriented graph that satisfies the weighted second neighborhood property for each of the weights of Lemma 42. If v_0 is the only Seymour vertex of D, then $|N^+(v_0)| = 0$ or $|N^+(v_0)| < |N^{++}(v_0)|$.*

(Lean: `unique_ seymourVertex_ sink_ or_ strict_ slack`.) For a positive rational weight η with common denominator N, the weighted property of D for η follows from the second neighborhood conjecture for the blowup of D whose fiber over v is an independent set of $N\eta(v)$ vertices: a vertex of the fiber over v has the fibers over $N^+(v)$ as its out-neighbors and the fibers over $N^{++}(v)$ as its second out-neighbors. So Lemma 42, Corollary 43 and Proposition 45 below hold for every oriented graph if the SNC preprints are correct; OpenAI also state the weighted form as their Corollary 5.1(i) [to check].

Lemma 44 (Forcing). *Let D be an oriented graph without a sink in which v_0 is the only Seymour vertex. Then every vertex v other than v_0 has out-degree at least* 3, *and if v_0 has out-degree* 1, *with out-neighbor u, then $|N^{++}(v_0)| = |N^+(u)| \geq 3$.*

Proof. A vertex $v \neq v_0$ is not a Seymour vertex. If $d^+(v) = 1$ with $v \to u$, then u has an out-neighbor, which is not v, so $d^{++}(v) \geq 1$ and v is a Seymour vertex; so $d^+(v) \geq 2$ for $v \neq v_0$. Suppose $d^+(v) = 2$ for some $v \neq v_0$, with $N^+(v) = \{a, b\}$. Then $d^{++}(v) \leq 1$, and we may assume $b \not\to a$. An out-neighbor of b is not v, a or b, so it lies in $N^{++}(v)$, and $d^+(b) \leq 1$. As D has no sink and vertices other than v_0 have out-degree at least 2, $b = v_0$, $N^+(v_0) = \{u\}$ and $N^{++}(v) = \{u\}$. The out-neighbors of a lie in $\{v_0, u\}$, and $a \neq v_0$ has at least two, so $N^+(a) = \{v_0, u\}$. The vertex $u \neq v_0$ has $d^+(u) \geq 3$: if $d^+(u) = 2$, the argument just given, applied to u, would give $u \to v_0$, but $v_0 \to u$. Every out-neighbor z of u differs from v_0 (as $v_0 \to u$), from u, and from a (as $a \to u$), so $z \in N^{++}(a)$. Hence $d^{++}(a) \geq d^+(u) \geq 3 > d^+(a)$, and $a \neq v_0$ is a Seymour vertex, a contradiction. So every $v \neq v_0$ has out-degree at least 3. If $N^+(v_0) = \{u\}$, then no out-neighbor of u is v_0 or u, so $N^{++}(v_0) = N^+(u)$, and $d^+(u) \geq 3$. □

Proposition 45 (Dual perturbations). *Let D be an oriented graph without a sink in which v_0 is the only Seymour vertex, put $s = \sigma(v_0)$, and suppose that D satisfies the weighted second neighborhood property for every positive rational weight. Then $s \geq 1$, and there are vertices v_1 and v_2 other than v_0 with*

$$\begin{aligned} &|N^{++}(v_1) \cap N^+(v_0)| - |N^+(v_1) \cap N^+(v_0)| \\ &\qquad \geq \lceil |N^+(v_0)|(-\sigma(v_1))/s \rceil, \\ &|N^+(v_2) \cap N^{++}(v_0)| - |N^{++}(v_2) \cap N^{++}(v_0)| \\ &\qquad \geq \lceil |N^{++}(v_0)|(-\sigma(v_2))/s \rceil. \end{aligned}$$

Proof. If $s = 0$, then $|N^+(v_0)| \geq 1$, and Lemma 42 gives a second Seymour vertex. So $s \geq 1$. Put ε for a small positive rational number. The weight η that is $1 + (s+\varepsilon)/d^+(v_0)$ on $N^+(v_0)$ and 1 elsewhere is positive, and $\eta(N^+(v_0)) = d^+(v_0) + s + \varepsilon > d^{++}(v_0) = \eta(N^{++}(v_0))$, because $N^+(v_0) \cap N^{++}(v_0) = \emptyset$. Hence v_0 is not the vertex that the weighted property gives, and some $v_1 \neq v_0$ has $\eta(N^{++}(v_1)) \geq \eta(N^+(v_1))$, which in unit terms says $d^{++}(v_1) - d^+(v_1) + \frac{s+\varepsilon}{d^+(v_0)}(|N^{++}(v_1) \cap N^+(v_0)| - |N^+(v_1) \cap N^+(v_0)|) \geq 0$. As $\varepsilon \to 0$ and the number of vertices is finite, one v_1 works for a sequence of ε, and the integer $|N^{++}(v_1) \cap N^+(v_0)| - |N^+(v_1) \cap N^+(v_0)|$ is at least $-\sigma(v_1)d^+(v_0)/s$, hence at least its ceiling. The second statement is the same with the weight $1 - (s + \varepsilon)/d^{++}(v_0)$ on $N^{++}(v_0)$, positive because $d^{++}(v_0) = d^+(v_0) + s > s$, so that $\eta(N^+(v_0)) = d^+(v_0) > d^{++}(v_0) - s - \varepsilon = \eta(N^{++}(v_0))$ and a nonnegative combination is again attained at a vertex other than v_0. □

Proof of Theorem 9. On at most 6 vertices the statement is part of the enumeration of Proposition 47 (Section 4); the enumeration uses no weighted form of the conjecture. Let D be a sink-free graph on 7 vertices.

Suppose first that D has no Seymour vertex. A vertex v with $d^+(v) = 1$, $v \to b$, is a Seymour vertex: b has an out-neighbor, which is neither v nor b, so $d^{++}(v) \geq 1$. If $d^+(v) = 2$, label the two out-neighbors a and b so that $b \not\to a$; every out-neighbor of b is then in $N^{++}(v)$, so $d^+(b) \leq d^{++}(v) \leq 1$ and b is a Seymour vertex by the first case. So every vertex has out-degree at least 3, D has at least 21 arcs, and it is a tournament in which every vertex has out-degree and in-degree 3. By Corollary 31 its vertices are Seymour vertices, a contradiction.

Suppose now that v_0 is the only Seymour vertex of D. By Lemma 44 the six vertices other than v_0 have out-degree at least 3, so the number of arcs is at least $18 + d^+(v_0) \geq 19$ and at most 21 minus the number of nonadjacent pairs, so at most two pairs are nonadjacent. Up to relabeling there are eight positions of at most two nonadjacent pairs with v_0 fixed. `cert/multiplicity.py` lists, for each of the eight patterns and with the fixed vertex 0 as v_0, all completions to a graph in which the out-degree of 0 is at least 1 and the out-degrees of the other vertices are at least 3, and checks whether all vertices other than 0 are non-Seymour: there are 35984 completions, none of them satisfies the condition. So D does not exist. □

The completions contain every graph with a unique Seymour vertex v_0 that meets the degree condition of Lemma 44, and so the search is exhaustive for the seven-vertex case. The case in which D has fewer than seven vertices is the census of Section 4.

Proposition 46 (Six vertices). *D_6 is a Pisa graph and Seymour-tight; the 13-arc graph P_6 is a Pisa graph and not Seymour-tight.*

(Lean: `halkiewicz_ pisa_ ne_ sc_ seymourTight_ at_ six`.) In P_6 the slack is 0 at five vertices and -1 at vertex 5. Every strongly connected Seymour-tight graph is a Pisa graph. On six vertices Halkiewicz [17] finds 1050 labeled Pisa graphs, the number of Table 2, of which 270 are Seymour-tight, as the repository of wustep [26] also reports; Proposition 46 gives one of the others with a proof in Lean.

4. COMPUTATIONAL VERIFICATION AND EMPIRICAL PICTURE

All programs in `cert/` use integer and rational arithmetic and no floating point; `recompute.py` reads the Python box of Section 1 from this source, runs it and checks every printed number in the text with the same programs. The counts below are those of the exhaustive programs, and the figure and the tables state them.

Proposition 47 (Census of small graphs). *Every nonempty oriented graph on at most 6 vertices and every nonempty tournament on at most 8 vertices has a strong Seymour vertex and a vertex v with $|N^{++}(v)| \geq |N^-(v)|$, every nonempty sink-free oriented graph on at most 6 vertices has at least two Seymour vertices, and the counts of Table 2 hold.*

The number of oriented graphs up to isomorphism (OEIS A001174) and of tournaments (OEIS A000568) agrees with the counts of the program at every order ([28]; A000568 starts at order 0). The program lists the 22114 isomorphism classes of oriented graphs on one to six vertices by canonical forms, the strongly connected ones, and the counts of the table; the numbers of labeled graphs are $n!/|\mathrm{Aut}|$ summed over the classes. Figure 2 shows the distribution of the largest slack. It also checks that every sink-free graph of the list has at least two Seymour vertices; with the labeled Eulerian tight graphs on seven vertices and the labeled 3-regular tight graphs on eight vertices, these are all the inputs of Proposition 30.

Table 2. Oriented graphs on $1 \le n \le 6$ vertices up to isomorphism: number of classes, of strongly connected classes, of Seymour-tight classes, of Pisa classes (unlabeled / labeled), of strongly connected Seymour-tight classes (unlabeled / labeled), of Eulerian Seymour-tight classes, and the number of classes with maximum slack $0, 1, 2, 3$.

n	classes	str. conn.	tight	Pisa	s.c. tight	Eul. tight	classes by largest slack
1	1	1	1	1 / 1	1 / 1	1	0:1
2	2	0	1	0 / 0	0 / 0	1	0:2
3	7	1	2	1 / 2	1 / 2	2	0:7
4	42	4	4	1 / 6	1 / 6	3	0:35, 1:7
5	582	76	12	2 / 48	2 / 48	5	0:336, 1:202, 2:44
6	21480	4232	58	6 / 1050	3 / 270	9	0:6438, 1:10643, 2:3841, 3:558

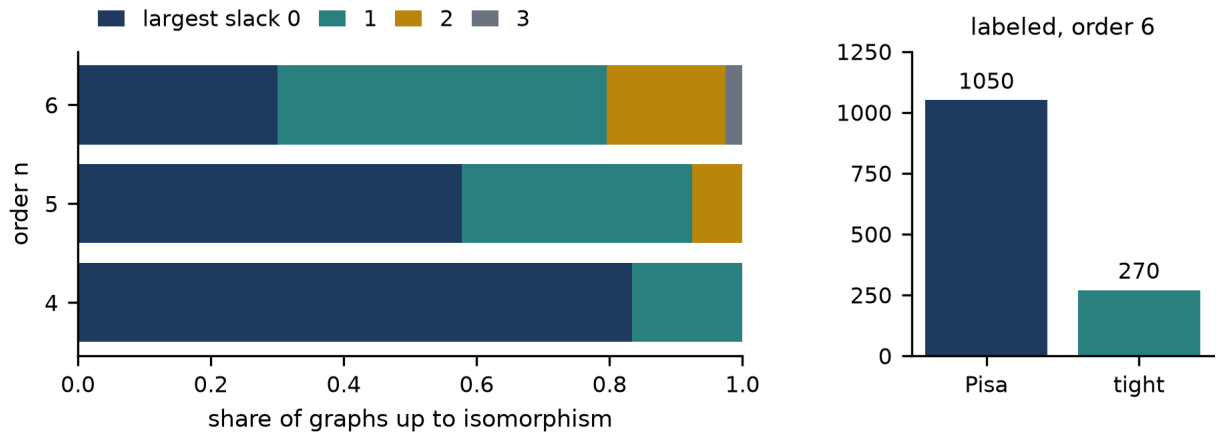


Figure 2. Left: the share of oriented graphs on four, five and six vertices up to isomorphism by their largest slack $\max_v(d^{++}(v) - d^{+}(v))$, from Table 2. Right: numbers of labeled Pisa graphs and of labeled strongly connected Seymour-tight graphs at order six.

Blowup classification. For Theorem 7 the program `cert/quotients.py` runs over the 4314 strongly connected oriented graphs on one to six vertices and closes 53578 cases: in 53577 a choice of Hall violations is ruled out by an integer dual vector, checked on the spot, and in one, the graph with one vertex, that vertex has no Hall violation to choose. Only for Q_{15} (twelve choices) and Q_{14} (eight) does it find positive integer vectors, with totals 42 and 66, which are not the smallest ones. Vertex-level checks of $Q_{15}[(7,3,11,3,9,3)]$ and $Q_{14}[(8,7,16,3,14,3)]$ confirm order 36 and 51 and no strong Seymour vertex.

Circulants. The program `cert/circulants.py` checks vertex 0 of all 2046 circulant tournaments of odd order 3 to 21 (Proposition 41) and the blowups of $C_n(1,\dots,k)$ by empty fibers $E_{s_0},\dots,E_{s_{n-1}}$ for $n \le 8$, $k \le (n-1)/2$ and fiber sizes at most 3: 28296 cases, 48 of them tight, 12 of these not constant in the sizes, and no disagreement with the criterion of Theorem 4. These examples include the blowup of D_6 with fiber sizes $1,2,1,2,1,2$, which has 9 vertices, is 3-regular and Seymour-tight and has a Seymour-tight converse.

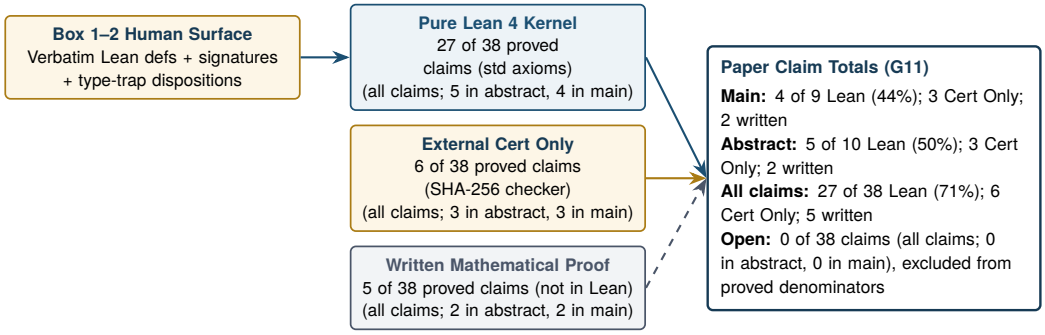
Figures and the two-vertex system. The script `recompute.py` recomputes the roots of the polynomials to twelve digits: γ_{HP} lies between 0.715538861763 and 0.715538861764, γ_3^* begins 0.717933012571, γ_4^* begins 0.718322495443, γ_5^* begins 0.718379069817. The gain of c_m over γ_{HP} divided by the gain of γ_m^* over γ_{HP} is larger than 1.98, 2.32 and 2.55 for $m = 3, 4, 5$, as the same program checks by exact comparisons. Huang and Peng report that a three-vertex version of their system, solved numerically, has no solution even at $\mu = 0.74530$, which is larger than the constants above.

5. FORMAL VERIFICATION & READER'S TRUST BOUNDARY

What You Must Trust to Accept This Paper:

1. **Lean 4 Kernel & Toolchain:** leanprover/lean4:v4.35.0; committed axiom receipt anchored to git commit 6d132f9d40ae (independent gate-box kernel replay pending); standard axioms only (propext, Classical.choice, Quot.sound).
2. **Core Definitions & Fidelity (Boxes 1–2):** Verify that the verbatim Lean definitions in Box 1 match Section 2 and that each Lean signature in Box 2 matches its stated theorem.
3. **External Certificates & Written Proofs:** 5 SHA-256-pinned external artifact(s) plus any claims marked Written or Unsettled in Table 4 below.

Reader's Trust-Flow Architecture (Distinct Evidence Paths to Claim Totals)



Anti-Shortcut Audit Strip — Source-Level Approx (Gate Replay Pending)

WARN: Committed receipt (gate replay pending) | PASS: sorry/admit: 0 (per committed receipt) | PASS: native_decide: 0 (per committed receipt) | PASS: Custom axioms & meta: 0 (per committed receipt) | PASS: Coverage: 222/222 public theorems (source-level approx) | WARN: Fidelity: 27/27 exact (draft; auditor/formalizer relationship unknown) | WARN: Witnesses: pending gate replay | WARN: Human referee: pending

Box 1 — Verbatim Lean 4 Custom Definitions (Human Audit Surface)

SeymourSNC.OrientedGraph (SeymourSNC/Basic.lean:43-49) — Section 2, Definition "Oriented graphs" (def:graph)

```
structure OrientedGraph (V : Type*) where
  /-- Directed adjacency relation `u → v`. -/
  adj : V → V → Prop
  /-- No self-loops. -/
  loopless : ∀ v, ¬ adj v v
  /-- No digons (2-cycles). -/
  asymm : ∀ {u v}, adj u v → ¬ adj v u
```

SeymourSNC.OrientedGraph.outNbr (SeymourSNC/Basic.lean:104-105) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V]`

```
def outNbr (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : Finset V :=
  Finset.univ.filter (fun u => D.adj v u)
```

SeymourSNC.OrientedGraph.inNbr (SeymourSNC/Basic.lean:108-109) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V]`

```
def inNbr (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : Finset V :=
  Finset.univ.filter (fun u => D.adj u v)
```

SeymourSNC.OrientedGraph.outDeg (SeymourSNC/Basic.lean:112-113) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V]`

```
def outDeg (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : ℕ :=
  (D.outNbr v).card
```

SeymourSNC.OrientedGraph.inDeg (SeymourSNC/Basic.lean:116-117) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V]`

```
def inDeg (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : ℕ :=
  (D.inNbr v).card
```

SeymourSNC.OrientedGraph.secondOutNbr (SeymourSNC/Basic.lean:194-195) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def secondOutNbr (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : Finset V :=
  Finset.univ.filter (fun w => w ≠ v ∧ ¬ D.adj v w ∧ ∃ u, D.adj v u ∧ D.adj u w)
```

SeymourSNC.OrientedGraph.secondInNbr (SeymourSNC/Basic.lean:198-199) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def secondInNbr (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : Finset V :=
  Finset.univ.filter (fun w => w ≠ v ∧ ¬ D.adj w v ∧ ∃ u, D.adj w u ∧ D.adj u v)
```

SeymourSNC.OrientedGraph.secondOutDeg (SeymourSNC/Basic.lean:202-203) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def secondOutDeg (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : ℕ :=
  (D.secondOutNbr v).card
```

SeymourSNC.OrientedGraph.secondInDeg (SeymourSNC/Basic.lean:206-207) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def secondInDeg (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : ℕ :=
  (D.secondInNbr v).card
```

SeymourSNC.OrientedGraph.arcCount (SeymourSNC/Basic.lean:120-121) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V]`

```
def arcCount (D : OrientedGraph V) [DecidableRel D.adj] : ℕ :=
  ∑ v : V, D.outDeg v
```

SeymourSNC.OrientedGraph.IsTournament (SeymourSNC/Basic.lean:78-79) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*}`

```
def IsTournament (D : OrientedGraph V) : Prop :=
  ∀ u v : V, u ≠ v → D.adj u v ∨ D.adj v u
```

SeymourSNC.OrientedGraph.IsStronglyConnected (SeymourSNC/Basic.lean:82-83) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*}`

```
def IsStronglyConnected (D : OrientedGraph V) : Prop :=
  ∀ u v : V, Relation.ReflTransGen D.adj u v
```

SeymourSNC.OrientedGraph.IsEulerian (SeymourSNC/Basic.lean:124-125) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V]`

```
def IsEulerian (D : OrientedGraph V) [DecidableRel D.adj] : Prop :=
  ∀ v : V, D.outDeg v = D.inDeg v
```

SeymourSNC.OrientedGraph.IsRegular (SeymourSNC/Basic.lean:128-129) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V]`

```
def IsRegular (D : OrientedGraph V) [DecidableRel D.adj] (k : ℕ) : Prop :=
  ∀ v : V, D.outDeg v = k ∧ D.inDeg v = k
```

SeymourSNC.OrientedGraph.Iso (SeymourSNC/Basic.lean:90-92) — Section 2, Definition "Oriented graphs" (def:graph)

Context binders in scope: `variable {V W : Type*}`

```
structure Iso (D₁ : OrientedGraph V) (D₂ : OrientedGraph W) where
  toEquiv : V ≃ W
  map_adj' : ∀ u v, D₁.adj u v ↔ D₂.adj (toEquiv u) (toEquiv v)
```

SeymourSNC.OrientedGraph.IsSeymourVertex (SeymourSNC/Basic.lean:214-215) — Section 2, Definition "Seymour vertices, tight graphs, Pisa graphs" (def:seymour)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def IsSeymourVertex (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : Prop :=
  D.outDeg v ≤ D.secondOutDeg v
```

SeymourSNC.OrientedGraph.IsSeymourTightVertex (SeymourSNC/Basic.lean:218-219) — Section 2, Definition "Seymour vertices, tight graphs, Pisa graphs" (def:seymour)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def IsSeymourTightVertex (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : Prop :=
  D.secondOutDeg v = D.outDeg v
```

SeymourSNC.OrientedGraph.IsSeymourTight (SeymourSNC/Basic.lean:222-223) — Section 2, Definition "Seymour vertices, tight graphs, Pisa graphs" (def:seymour)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def IsSeymourTight (D : OrientedGraph V) [DecidableRel D.adj] : Prop :=
  ∀ v : V, D.IsSeymourTightVertex v
```

SeymourSNC.OrientedGraph.IsPisaGraph (SeymourSNC/Basic.lean:230-233) — Section 2, Definition "Seymour vertices, tight graphs, Pisa graphs" (def:seymour)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def IsPisaGraph (D : OrientedGraph V) [DecidableRel D.adj] : Prop :=
  D.IsStronglyConnected ∧
    (∀ v : V, D.secondOutDeg v ≤ D.outDeg v) ∧
    (∃ v : V, D.secondOutDeg v = D.outDeg v)
```

SeymourSNC.OrientedGraph.IsStrongSeymourVertex (SeymourSNC/Basic.lean:237-240) — Section 2, Definition "Strong Seymour vertices" (def:strong)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def IsStrongSeymourVertex (D : OrientedGraph V) [DecidableRel D.adj] (v : V) : Prop :=
  ∃ f : V → V,
    Set.InjOn f (D.outNbr v : Set V) ∧
    ∀ u ∈ D.outNbr v, f u ∈ D.secondOutNbr v ∧ D.adj u (f u)
```

SeymourSNC.OrientedGraph.MaxMatchingAtLeast (SeymourSNC/Basic.lean:243-246) — Section 2, Definition "Strong Seymour vertices" (def:strong)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def MaxMatchingAtLeast (D : OrientedGraph V) [DecidableRel D.adj] (v : V) (m : ℕ) :
  ↪ Prop :=
  ∃ S : Finset V, S ⊆ D.outNbr v ∧ S.card = m ∧
    ∃ f : V → V, Set.InjOn f (S : Set V) ∧
      ∀ u ∈ S, f u ∈ D.secondOutNbr v ∧ D.adj u (f u)
```

SeymourSNC.OrientedGraph.MaxMatchingAtMost (SeymourSNC/Basic.lean:249-253) — Section 2, Definition "Strong Seymour vertices" (def:strong)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def MaxMatchingAtMost (D : OrientedGraph V) [DecidableRel D.adj] (v : V) (m : ℕ) :
  ↪ Prop :=
  ∀ S : Finset V, S ⊆ D.outNbr v →
    (∃ f : V → V, Set.InjOn f (S : Set V) ∧
      ∀ u ∈ S, f u ∈ D.secondOutNbr v ∧ D.adj u (f u)) →
    S.card ≤ m
```

SeymourSNC.OrientedGraph.HasExactMaxMatching (SeymourSNC/Basic.lean:256-257) — Section 2, Definition "Strong Seymour vertices" (def:strong)

Context binders in scope: `variable {V W : Type*} ; variable [Fintype V] [DecidableEq V]`

```
def HasExactMaxMatching (D : OrientedGraph V) [DecidableRel D.adj] (v : V) (m : ℕ) :
  ↪ Prop :=
  D.MaxMatchingAtLeast v m ∧ D.MaxMatchingAtMost v m
```

SeymourSNC.HasWeightedSNC (SeymourSNC/StrongSeymourAndZeroSlack.lean:380-381) — Section 2, Definition "Weights" (def:weights)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def HasWeightedSNC (D : OrientedGraph V) [DecidableRel D.adj] (η : V → ℚ) : Prop :=
  ∃ v : V, ∑ u ∈ D.outNbr v, η u ≤ ∑ w ∈ D.secondOutNbr v, η w
```

SeymourSNC.unitPerturbWeight (SeymourSNC/StrongSeymourAndZeroSlack.lean:384-385) — Section 2, Definition "Weights" (def:weights)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def unitPerturbWeight (u₀ : V) (x : V) : ℚ :=
  if x = u₀ then (3 / 2 : ℚ) else 1
```

SeymourSNC.lexProd (SeymourSNC/ConverseAndCirculants.lean:526-543) — Section 2, Definition "Circulants, blowups, Hall violations" (def:blowups)

```
def lexProd {I : Type*} {F : I → Type*} (Q : OrientedGraph I)
    (G : ∀ i : I, OrientedGraph (F i)) : OrientedGraph (Σ i : I, F i) where
  adj a b := Q.adj a.1 b.1 ∨ ∃ h : a.1 = b.1, (G a.1).adj a.2 (h ▸ b.2)
  loopless a := by
    rintro (hQ | ⟨_, hG⟩)
    · exact Q.loopless a.1 hQ
    · exact (G a.1).loopless a.2 hG
  asymm {a b} h1 h2 := by
    rcases h1 with hQ1 | ⟨heq1, hG1⟩
    · rcases h2 with hQ2 | ⟨heq2, _⟩
      · exact Q.asymm hQ1 hQ2
      · rw [heq2] at hQ1; exact Q.loopless a.1 hQ1
    · rcases h2 with hQ2 | ⟨heq2, hG2⟩
      · rw [heq1] at hQ2; exact Q.loopless b.1 hQ2
      · rcases a with ⟨i, x⟩; rcases b with ⟨j, y⟩
        dsimp at heq1 heq2 hG1 hG2
        subst heq1
        exact (G i).asymm hG1 hG2
```

SeymourSNC.D6 (SeymourSNC/ConverseAndCirculants.lean:509-512) — Section 2, Definition "Circulants, blowups, Hall violations" (def:blowups)

```
def D6 : OrientedGraph (Fin 6) where
  adj i j := let d := (j.val + 6 - i.val) % 6; d = 1 ∨ d = 2
  loopless i := by dsimp; omega
  asymm {i j} h1 h2 := by dsimp at h1 h2; omega
```

SeymourSNC.emptyBlowup (SeymourSNC/FarkasEngine.lean:49-52) — Section 2, Definition "Circulants, blowups, Hall violations" (def:blowups)

Context binders in scope: `variable {I J : Type*}`

```
def emptyBlowup (Q : OrientedGraph I) (n : I → ℕ) : OrientedGraph (Σ i : I, Fin (n
  ↪ i)) where
  adj a b := Q.adj a.1 b.1
  loopless a := Q.loopless a.1
  asymm h := Q.asymm h
```

SeymourSNC.ttBlowup (SeymourSNC/FarkasEngine.lean:60-76) — Section 2, Definition "Circulants, blowups, Hall violations" (def:blowups)

Context binders in scope: `variable {I J : Type*}`

```
def ttBlowup (Q : OrientedGraph I) (n : I → ℕ) : OrientedGraph (Σ i : I, Fin (n i))
  ↪ where
  adj a b := Q.adj a.1 b.1 ∨ ∃ h : a.1 = b.1, a.2.val < (h ▸ b.2).val
  loopless a := by
    rintro (hQ | ⟨_, hlt⟩)
    · exact Q.loopless a.1 hQ
    · exact lt_irrefl _ hlt
  asymm {a b} h1 h2 := by
    rcases h1 with hQ1 | ⟨heq1, hlt1⟩
    · rcases h2 with hQ2 | ⟨heq2, _⟩
      · exact Q.asymm hQ1 hQ2
      · rw [heq2] at hQ1; exact Q.loopless a.1 hQ1
    · rcases h2 with hQ2 | ⟨heq2, hlt2⟩
      · rw [heq1] at hQ2; exact Q.loopless b.1 hQ2
      · rcases a with ⟨i, x⟩; rcases b with ⟨j, y⟩
        dsimp at heq1 heq2 hlt1 hlt2
        subst heq1
        exact lt_asymm hlt1 hlt2
```

SeymourSNC.HasBlowupHallViolation (SeymourSNC/FarkasEngine.lean:90-92) — Section 2, Definition "Circulants, blowups, Hall violations" (def:blowups)

Context binders in scope: `variable {I J : Type*} ; variable [Fintype I] [DecidableEq I]`

```
def HasBlowupHallViolation (Q : OrientedGraph I) [DecidableRel Q.adj] (n : I → ℕ) (i
  ↪ : I) : Prop :=
  ∃ S : Finset I, S ⊆ Q.outNbr i ∧ S.Nonempty ∧
    ∑ w ∈ S.biUnion (fun u => Q.outNbr u ∩ Q.secondOutNbr i), n w < ∑ u ∈ S, n u
```

SeymourSNC.TourMask (SeymourSNC/FarkasEngine.lean:867-868) — Section 2, Definition "Named graphs" (def:named)

```
abbrev TourMask (k : ℕ) :=
  { f : Fin k → Fin k → Bool // (∀ i, f i i = false) ∧ (∀ i j, i ≠ j → f i j = !f j
    ↪ i) }
```

SeymourSNC.TourMask.toOrientedGraph (SeymourSNC/FarkasEngine.lean:877-883) — Section 2, Definition "Named graphs" (def:named)

```
def TourMask.toOrientedGraph {k : ℕ} (T : TourMask k) : OrientedGraph (Fin k) where
  adj u v := T.1 u v = true
  loopless v := by simp [T.2.1 v]
  asymm {u v} h1 h2 := by
    have hne : u ≠ v := fun h => by subst h; simp [T.2.1 u] at h1
    have := T.2.2 u v hne
    rw [h1, h2] at this; simp at this
```

SeymourSNC.tourMaskOfOutMasks (SeymourSNC/FarkasEngine.lean:871-874) — Section 2, Definition "Named graphs" (def:named)

```
def tourMaskOfOutMasks {k : ℕ} (om : Fin k → ℕ)
    (hloop : ∀ i : Fin k, (om i).testBit i.val = false)
    (htour : ∀ i j : Fin k, i ≠ j → (om i).testBit j.val = !(om j).testBit i.val) :
      ↪ TourMask k :=
  ⟨fun i j => (om i).testBit j.val, hloop, htour⟩
```

SeymourSNC.dzitsoevOutMask (SeymourSNC/FarkasDzitsoev.lean:46-52) — Section 2, Definition "Named graphs" (def:named)

```
def dzitsoevOutMask : Fin 6 → ℕ
  | ⟨0, _⟩ => 50
  | ⟨1, _⟩ => 56
  | ⟨2, _⟩ => 11
  | ⟨3, _⟩ => 17
  | ⟨4, _⟩ => 36
  | ⟨5, _⟩ => 12
```

SeymourSNC.dzitsoevTourMask (SeymourSNC/FarkasDzitsoev.lean:55-58) — Section 2, Definition "Named graphs" (def:named)

```
def dzitsoevTourMask : TourMask 6 :=
  tourMaskOfOutMasks dzitsoevOutMask
    (by intro i; fin_cases i <;> decide)
    (by intro i j hne; revert hne; fin_cases i <;> fin_cases j <;> decide)
```

SeymourSNC.DzitsoevQ (SeymourSNC/FarkasDzitsoev.lean:61-61) — Section 2, Definition "Named graphs" (def:named)

```
def DzitsoevQ : OrientedGraph (Fin 6) := dzitsoevTourMask.toOrientedGraph
```

SeymourSNC.delOneArc (SeymourSNC/FarkasDzitsoev.lean:71-74) — Section 2, Definition "Named graphs" (def:named)

```
def delOneArc {V : Type*} [DecidableEq V] (Q : OrientedGraph V) (u v : V) :
  ↪ OrientedGraph V where
  adj a b := Q.adj a b ∧ ¬ (a = u ∧ b = v)
  loopless a h := Q.loopless a h.1
  asymm h1 h2 := Q.asymm h1.1 h2.1
```

SeymourSNC.delTwoArcs (SeymourSNC/FarkasDzitsoev.lean:81-85) — Section 2, Definition "Named graphs" (def:named)

```
def delTwoArcs {V : Type*} [DecidableEq V] (Q : OrientedGraph V) (u1 v1 u2 v2 : V) :
    OrientedGraph V where
  adj a b := Q.adj a b ∧ ¬ (a = u1 ∧ b = v1) ∧ ¬ (a = u2 ∧ b = v2)
  loopless a h := Q.loopless a h.1
  asymm h1 h2 := Q.asymm h1.1 h2.1
```

SeymourSNC.dzitsoev36Sizes (SeymourSNC/FarkasDzitsoev.lean:678-684) — Section 2, Definition "Named graphs" (def:named)

```
def dzitsoev36Sizes : Fin 6 → ℕ
  | ⟨0, _⟩ => 7
  | ⟨1, _⟩ => 3
  | ⟨2, _⟩ => 11
  | ⟨3, _⟩ => 3
  | ⟨4, _⟩ => 9
  | ⟨5, _⟩ => 3
```

SeymourSNC.dzitsoevDel05Sizes (SeymourSNC/FarkasDzitsoev.lean:782-788) — Section 2, Definition "Named graphs" (def:named)

```
def dzitsoevDel05Sizes : Fin 6 → ℕ
  | ⟨0, _⟩ => 8
  | ⟨1, _⟩ => 7
  | ⟨2, _⟩ => 16
  | ⟨3, _⟩ => 3
  | ⟨4, _⟩ => 14
  | ⟨5, _⟩ => 3
```

SeymourSNC.r11AdjBool (SeymourSNC/StrongSeymourAndZeroSlack.lean:263-274) — Section 2, Definition "Named graphs" (def:named)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def r11AdjBool (u v : Fin 11) : Bool :=
  let a := u.val
  let b := v.val
  if a = 0 then (1 ≤ b ∧ b ≤ 5)
  else if b = 0 then (6 ≤ a ∧ a ≤ 10)
  else if (a = 1 ∧ b = 2) ∨ (a = 2 ∧ b = 3) ∨ (a = 3 ∧ b = 1) then true
  else if (1 ≤ a ∧ a ≤ 3) ∧ (4 ≤ b ∧ b ≤ 7) then true
  else if (8 ≤ a ∧ a ≤ 10) ∧ (1 ≤ b ∧ b ≤ 3) then true
  else if (a = 8 ∧ b = 9) ∨ (a = 9 ∧ b = 10) ∨ (a = 10 ∧ b = 8) then true
  else if (4 ≤ a ∧ a ≤ 7) ∧ (8 ≤ b ∧ b ≤ 10) then true
  else (a = 4 ∧ b = 5) ∨ (a = 6 ∧ b = 7) ∨ (a = 5 ∧ b = 6) ∨
       (a = 5 ∧ b = 7) ∨ (a = 4 ∧ b = 6) ∨ (a = 7 ∧ b = 4)
```

SeymourSNC.R11Star (SeymourSNC/StrongSeymourAndZeroSlack.lean:278-281) — Section 2, Definition "Named graphs" (def:named)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def R11Star : OrientedGraph (Fin 11) where
  adj u v := r11AdjBool u v = true
  loopless v := by fin_cases v <;> decide
  asymm {u v} h1 h2 := by revert h1 h2; fin_cases u <;> fin_cases v <;> decide
```

SeymourSNC.pisaNonTight6AdjBool (SeymourSNC/StrongSeymourAndZeroSlack.lean:506-511) — Section 2, Definition "Named graphs" (def:named)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def pisaNonTight6AdjBool (u v : Fin 6) : Bool :=
  match u.val, v.val with
  | 0, 4 | 0, 5 | 1, 4 | 1, 5
  | 2, 0 | 2, 1 | 3, 0 | 3, 1
  | 4, 2 | 4, 3 | 5, 2 | 5, 3 | 5, 4 => true
  | _, _ => false
```

SeymourSNC.PisaNonTight6 (SeymourSNC/StrongSeymourAndZeroSlack.lean:515-518) — Section 2, Definition "Named graphs" (def:named)

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
def PisaNonTight6 : OrientedGraph (Fin 6) where
  adj u v := pisaNonTight6AdjBool u v = true
  loopless v := by fin_cases v <;> decide
  asymm {u v} h1 h2 := by revert h1 h2; fin_cases u <;> fin_cases v <;> decide
```

SeymourSNC.Lemma22Vec (SeymourSNC/HuangPengBarrier.lean:121-132) — Section 2, Definition "The two-vertex system" (def:system)

```
structure Lemma22Vec where
  x11 : ℝ
  x12 : ℝ
  x13 : ℝ
  x14 : ℝ
  x21 : ℝ
  x22 : ℝ
  x23 : ℝ
  x24 : ℝ
  x32 : ℝ
  x33 : ℝ
  x34 : ℝ
```

SeymourSNC.Lemma22Vec.d1 (SeymourSNC/HuangPengBarrier.lean:135-135) — Section 2, Definition "The two-vertex system" (def:system)

```
def Lemma22Vec.d1 (x : Lemma22Vec) : ℝ := x.x11 + x.x12 + x.x13 + x.x14
```

SeymourSNC.Lemma22Vec.d2 (SeymourSNC/HuangPengBarrier.lean:138-138) — Section 2, Definition "The two-vertex system" (def:system)

```
def Lemma22Vec.d2 (x : Lemma22Vec) : ℝ := x.x21 + x.x22 + x.x23 + x.x24
```

SeymourSNC.Lemma22Vec.d3 (SeymourSNC/HuangPengBarrier.lean:141-141) — Section 2, Definition "The two-vertex system" (def:system)

```
def Lemma22Vec.d3 (x : Lemma22Vec) : ℝ := x.x32 + x.x33 + x.x34
```

SeymourSNC.Lemma22VecQ (SeymourSNC/HuangPengBarrier.lean:1042-1053) — Section 2, Definition "The two-vertex system" (def:system)

```
structure Lemma22VecQ where
  x11 : ℚ
  x12 : ℚ
  x13 : ℚ
  x14 : ℚ
  x21 : ℚ
  x22 : ℚ
  x23 : ℚ
  x24 : ℚ
  x32 : ℚ
  x33 : ℚ
  x34 : ℚ
```

SeymourSNC.Lemma22VecQ.toReal (SeymourSNC/HuangPengBarrier.lean:1055-1058) — Section 2, Definition "The two-vertex system" (def:system)

```
def Lemma22VecQ.toReal (x : Lemma22VecQ) : Lemma22Vec where
  x11 := x.x11; x12 := x.x12; x13 := x.x13; x14 := x.x14
  x21 := x.x21; x22 := x.x22; x23 := x.x23; x24 := x.x24
  x32 := x.x32; x33 := x.x33; x34 := x.x34
```

SeymourSNC.evalLemma22F (SeymourSNC/HuangPengBarrier.lean:144-151) — Section 2, Definition "The two-vertex system" (def:system)

```
def evalLemma22F (x : Lemma22Vec) (w : ℝ) : ℝ :=
  x.x21 * (x.x32 - x.x11 - x.x13) +
  x.x22 * (x.x32 + x.x23 + x.x33) +
  x.x23 * x.x12 -
  x.x14 * (x.x12 + x.x21 + x.x22) +
  1 / 2 * (x.x21 + x.x22) ^ 2 -
  1 / 2 * (w * x.x11 ^ 2 + x.x12 ^ 2) +
  (w - 1) * x.x11 * x.x12
```

SeymourSNC.IsLemma22Feasible (SeymourSNC/HuangPengBarrier.lean:154-163) — Section 2, Definition "The two-vertex system" (def:system)

```
def IsLemma22Feasible (x : Lemma22Vec) (μ w : ℝ) : Prop :=
  0 < x.x11 ∧
  0 ≤ x.x12 ∧ 0 ≤ x.x13 ∧ 0 ≤ x.x14 ∧
  0 ≤ x.x21 ∧ 0 ≤ x.x22 ∧ 0 ≤ x.x23 ∧ 0 ≤ x.x24 ∧
  0 ≤ x.x32 ∧ 0 ≤ x.x33 ∧ 0 ≤ x.x34 ∧
  x.d2 < μ * x.d1 ∧
  x.d3 < μ ^ 2 * x.d1 ∧
  x.x12 + x.x22 + x.x32 < μ * (x.x11 + x.x21) ∧
  x.x12 + x.x13 + x.x14 ≤ x.x21 ∧
  0 < evalLemma22F x w
```

SeymourSNC.evalMFreeF (SeymourSNC/HuangPengBarrier.lean:171-173) — Section 2, Definition "The m-free forms" (def:mfree)

```
def evalMFreeF (K α : ℝ) (x : Lemma22Vec) (w : ℝ) : ℝ :=
  evalLemma22F x w -
  (w * K / 2 * (2 * α * x.x11 * (x.x11 - x.x12) - (α * x.x11) ^ 2) - K * α * x.x11 *
    ↪ x.x22)
```

SeymourSNC.wParam (SeymourSNC/HuangPengBarrier.lean:188-189) — Section 2, Definition "The m-free forms" (def:mfree)

```
def wParam (K α γ : ℝ) : ℝ :=
  (2 * γ ^ 3 + γ ^ 2 + 2 * K * α * γ) / (1 + K * α * (2 - α))
```

SeymourSNC.thetaParam (SeymourSNC/HuangPengBarrier.lean:192-193) — Section 2, Definition "The m-free forms" (def:mfree)

```
def thetaParam (K α γ : ℝ) : ℝ :=
  2 + 2 * K * α + 2 * γ + 2 * γ ^ 2 - 2 * (1 + K * α) * wParam K α γ
```

SeymourSNC.hpReducedVec (SeymourSNC/HuangPengBarrier.lean:196-207) — Section 2, Definition "The two-vertex system" (def:system)

```
def hpReducedVec (γ gI th y1 y2 y3 y4 : ℝ) : Lemma22Vec where
  x11 := -y1 + (1 + γ) * y2 + y3 + y4
  x12 := -γ * y1 + γ * y3
  x13 := (1 + γ) * y1 - (1 + γ) * y2 + (th * gI - 1 - γ) * y3
  x14 := 0
  x21 := -γ * y2 + th * y3
  x22 := γ * y2 + γ * y4
  x23 := 0
  x24 := 0
  x32 := γ * th * y3
  x33 := γ ^ 2 * y4
  x34 := 0
```

SeymourSNC.P_HP (SeymourSNC/HuangPengBarrier.lean:324-324) — Section 2, Definition "The two-vertex system" (def:system)

```
def P_HP (x : ℝ) : ℝ := 8 * x ^ 5 + 4 * x ^ 4 - 12 * x ^ 3 - 7 * x ^ 2 + 2 * x + 4
```

SeymourSNC.P_HP_existsUnique_root (SeymourSNC/HuangPengBarrier.lean:385-404) — Section 2, Definition "The two-vertex system" (def:system)

```
theorem P_HP_existsUnique_root : ∃! γ : ℝ, γ ∈ Set.Icc (0 : ℝ) 1 ∧ P_HP γ = 0
```

SeymourSNC.gammaHP (SeymourSNC/HuangPengBarrier.lean:407-407) — Section 2, Definition "The two-vertex system" (def:system)

```
noncomputable def gammaHP : ℝ := Classical.choose P_HP_existsUnique_root
```

SeymourSNC.hpInvPoly (SeymourSNC/HuangPengBarrier.lean:480-480) — Section 2, Definition "The two-vertex system" (def:system)

```
def hpInvPoly (γ : ℝ) : ℝ := (-1 / 2 : ℝ) + (7 / 4 : ℝ) * γ + (3 : ℝ) * γ ^ 2 + (-1 :
  ↪ ℝ) * γ ^ 3 + (-2 : ℝ) * γ ^ 4
```

SeymourSNC.P3_star (SeymourSNC/HuangPengBarrier.lean:1293-1293) — Section 2, Definition "The m-free forms" (def:mfree)

```
def P3_star (x : ℝ) : ℝ := (-22378050 : ℝ) + (-24963735 : ℝ) * x + (28639207 : ℝ) * x
  ↪ ^ 2 + (125596310 : ℝ) * x ^ 3 + (42451990 : ℝ) * x ^ 4 + (-113347780 : ℝ) * x ^ 5
  ↪ + (-134773912 : ℝ) * x ^ 6 + (30293760 : ℝ) * x ^ 7 + (69148800 : ℝ) * x ^ 8
```

SeymourSNC.P4_star (SeymourSNC/HuangPengBarrier.lean:1608-1608) — Section 2, Definition "The m-free forms" (def:mfree)

```
def P4_star (x : ℝ) : ℝ := (-169364196 : ℝ) + (-182104902 : ℝ) * x + (216077471 : ℝ)
  ↪ * x ^ 2 + (929837575 : ℝ) * x ^ 3 + (304289900 : ℝ) * x ^ 4 + (-831316250 : ℝ) *
  ↪ x ^ 5 + (-984860000 : ℝ) * x ^ 6 + (217125000 : ℝ) * x ^ 7 + (506250000 : ℝ) * x
  ↪ ^ 8
```

SeymourSNC.P5_star (SeymourSNC/HuangPengBarrier.lean:1923-1923) — Section 2, Definition "The m-free forms" (def:mfree)

```
def P5_star (x : ℝ) : ℝ := (-13743810756 : ℝ) + (-14781214422 : ℝ) * x + (17511669431
  ↪ : ℝ) * x ^ 2 + (75410797375 : ℝ) * x ^ 3 + (24720972650 : ℝ) * x ^ 4 +
  ↪ (-67344533750 : ℝ) * x ^ 5 + (-79832960000 : ℝ) * x ^ 6 + (17546625000 : ℝ) * x ^
  ↪ 7 + (41006250000 : ℝ) * x ^ 8
```

SeymourSNC.P3_star_existsUnique_root (SeymourSNC/HuangPengBarrier.lean:1307-1317) — Section 2, Definition "The m-free forms" (def:mfree)

```
theorem P3_star_existsUnique_root : ∃! γ : ℝ, γ ∈ Set.Icc (715 / 1000 : ℝ) (720 /
  ↪ 1000) ∧ P3_star γ = 0
```

SeymourSNC.P4_star_existsUnique_root (SeymourSNC/HuangPengBarrier.lean:1622-1632) — Section 2, Definition "The m-free forms" (def:mfree)

```
theorem P4_star_existsUnique_root : ∃! γ : ℝ, γ ∈ Set.Icc (715 / 1000 : ℝ) (720 /
  ↪ 1000) ∧ P4_star γ = 0
```

SeymourSNC.P5_star_existsUnique_root (SeymourSNC/HuangPengBarrier.lean:1937-1947) — Section 2, Definition "The m-free forms" (def:mfree)

```
theorem P5_star_existsUnique_root : ∃! γ : ℝ, γ ∈ Set.Icc (715 / 1000 : ℝ) (720 /
  ↪ 1000) ∧ P5_star γ = 0
```

SeymourSNC.gamma3Star (SeymourSNC/HuangPengBarrier.lean:1320-1320) — Section 2, Definition "The m-free forms" (def:mfree)

```
noncomputable def gamma3Star : ℝ := Classical.choose P3_star_existsUnique_root
```

SeymourSNC.gamma4Star (SeymourSNC/HuangPengBarrier.lean:1635-1635) — Section 2, Definition "The m-free forms" (def:mfree)

```
noncomputable def gamma4Star : ℝ := Classical.choose P4_star_existsUnique_root
```

SeymourSNC.gamma5Star (SeymourSNC/HuangPengBarrier.lean:1950-1950) — Section 2, Definition "The m-free forms" (def:mfree)

```
noncomputable def gamma5Star : ℝ := Classical.choose P5_star_existsUnique_root
```

SeymourSNC.m3InvPoly (SeymourSNC/HuangPengBarrier.lean:1363-1363) — Section 2, Definition "The m-free forms" (def:mfree)

```
def m3InvPoly (γ : ℝ) : ℝ := (-7463 / 6690 : ℝ) + (28639207 / 22378050 : ℝ) * γ +
  ↪ (12559631 / 2237805 : ℝ) * γ ^ 2 + (4245199 / 2237805 : ℝ) * γ ^ 3 + (-11334778 /
  ↪ 2237805 : ℝ) * γ ^ 4 + (-67386956 / 11189025 : ℝ) * γ ^ 5 + (1009792 / 745935 :
  ↪ ℝ) * γ ^ 6 + (153664 / 49729 : ℝ) * γ ^ 7
```

SeymourSNC.m4InvPoly (SeymourSNC/HuangPengBarrier.lean:1678-1678) — Section 2, Definition "The m-free forms" (def:mfree)

```
def m4InvPoly (γ : ℝ) : ℝ := (-13993 / 13014 : ℝ) + (216077471 / 169364196 : ℝ) * γ +
  ↪ (929837575 / 169364196 : ℝ) * γ ^ 2 + (76072475 / 42341049 : ℝ) * γ ^ 3 +
  ↪ (-415658125 / 84682098 : ℝ) * γ ^ 4 + (-246215000 / 42341049 : ℝ) * γ ^ 5 +
  ↪ (6031250 / 4704561 : ℝ) * γ ^ 6 + (1562500 / 522729 : ℝ) * γ ^ 7
```

SeymourSNC.m5InvPoly (SeymourSNC/HuangPengBarrier.lean:1993-1993) — Section 2, Definition "The m-free forms" (def:mfree)

```
def m5InvPoly (γ : ℝ) : ℝ := (-126083 / 117234 : ℝ) + (17511669431 / 13743810756 : ℝ)
  ↪ * γ + (75410797375 / 13743810756 : ℝ) * γ ^ 2 + (12360486325 / 6871905378 : ℝ) *
  ↪ γ ^ 3 + (-2590174375 / 528608106 : ℝ) * γ ^ 4 + (-19958240000 / 3435952689 : ℝ) *
  ↪ γ ^ 5 + (54156250 / 42419169 : ℝ) * γ ^ 6 + (14062500 / 4713241 : ℝ) * γ ^ 7
```

Selected Concrete Sanity Examples (checked in Lean; examples only, not global identity proofs): SeymourSNC.DzitsoevQ_arcCount (DzitsoevQ.arcCount = 15).

Box 2 — Verbatim Main Theorem Signatures, Context Binders & Applicability

SeymourSNC.hp_primal_barrier_rational (SeymourSNC/HuangPengBarrier.lean:1064-1219) src:3d3d97a8be2161

```
theorem hp_primal_barrier_rational {μ w : ℚ}
    (hμ_gt : gammaHP < (μ : ℝ)) (hμ_le : μ ≤ 1) (_hw : 1 ≤ w) :
    ∃ x : Lemma22VecQ, IsLemma22Feasible x.toReal (μ : ℝ) (w : ℝ)
```

Assumptions: Rational mu and w with gammaHP < mu (as reals), mu <= 1 and 1 <= w; the witness is a vector of 11 rationals, read as reals in IsLemma22Feasible. The hypothesis 1 <= w is not used by the proof. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.hp_sos_nonpos (SeymourSNC/HuangPengBarrier.lean:709-722) src:d36ea3c426dcbb

```
theorem hp_sos_nonpos (y1 y2 y3 y4 : ℝ)
    (hi4 : 0 ≤ (hpReducedVec gammaHP (hpInvPoly gammaHP) (thetaParam 0 0 gammaHP) y1
      ↪ y2 y3 y4).x21 -
               (hpReducedVec gammaHP (hpInvPoly gammaHP) (thetaParam 0 0 gammaHP) y1
            ↪ y2 y3 y4).x12 -
               (hpReducedVec gammaHP (hpInvPoly gammaHP) (thetaParam 0 0 gammaHP) y1
            ↪ y2 y3 y4).x13)
    (hx21 : 0 ≤ (hpReducedVec gammaHP (hpInvPoly gammaHP) (thetaParam 0 0 gammaHP) y1
      ↪ y2 y3 y4).x21)
    (hx22 : 0 ≤ (hpReducedVec gammaHP (hpInvPoly gammaHP) (thetaParam 0 0 gammaHP) y1
      ↪ y2 y3 y4).x22)
    (hx32 : 0 ≤ (hpReducedVec gammaHP (hpInvPoly gammaHP) (thetaParam 0 0 gammaHP) y1
      ↪ y2 y3 y4).x32)
    (hx33 : 0 ≤ (hpReducedVec gammaHP (hpInvPoly gammaHP) (thetaParam 0 0 gammaHP) y1
      ↪ y2 y3 y4).x33) :
    evalMFreeF 0 0 (hpReducedVec gammaHP (hpInvPoly gammaHP) (thetaParam 0 0 gammaHP)
      ↪ y1 y2 y3 y4) (wParam 0 0 gammaHP) ≤ 0
```

Assumptions: x(y) is hpReducedVec with gI = hpInvPoly, the polynomial that equals 1/gamma at the root (hp_inv_eq), so that it is the reduced family of Section 2; any reals y1, y2, y3, y4 with the five displayed inequalities on x(y). Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.internal_outDeg_concentration_X11 (SeymourSNC/HuangPengBarrier.lean:1227-1277) src:e7f363ab762753

```
theorem internal_outDeg_concentration_X11 {V : Type*} [Fintype V] [DecidableEq V]
    (D : OrientedGraph V) [DecidableRel D.adj] (u v : V)
    {w : ℝ} (hw : 0 < w)
    (hmin : ∀ y ∈ D.outNbr u,
      w * ((D.outNbr v ∩ D.outNbr u).card : ℝ) + ((D.outNbr v ∩ D.secondOutNbr
        ↪ u).card : ℝ) ≤
      w * ((D.outNbr y ∩ D.outNbr u).card : ℝ) + ((D.outNbr y ∩ D.secondOutNbr
        ↪ u).card : ℝ)) :
    let X1 := D.outNbr u
    let X2 := D.secondOutNbr u
    let X11 := X1 ∩ D.outNbr v
    let X12 := X1 ∩ D.secondOutNbr v
    let X22 := X2 ∩ D.secondOutNbr v
    ∀ y ∈ X11,
      w * ((X11.card : ℝ) - (X12.card : ℝ)) - (X22.card : ℝ) ≤
      w * ((D.outNbr y ∩ X11).card : ℝ)
```

Assumptions: D is an oriented graph on a finite type with decidable equality and adjacency; u, v any vertices; w > 0 real; the minimality inequality for every y in N+(u). v need not be in N+(u). Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.max_sq_ge_tangent (SeymourSNC/HuangPengBarrier.lean:1282-1290) src:068f639f5c8c2b

```
theorem max_sq_ge_tangent {α x11 L : ℝ} (hα : 0 ≤ α) (hx : 0 < x11) :
    2 * (α * x11) * L - (α * x11) ^ 2 ≤ (max 0 L) ^ 2
```

Assumptions: Real alpha, x11 and L with 0 <= alpha and 0 < x11. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.m3_sos_nonpos (SeymourSNC/HuangPengBarrier.lean:1592-1605) src:3d3fd92188432d

```
theorem m3_sos_nonpos (y1 y2 y3 y4 : ℝ)
    (hi4 : 0 ≤ (hpReducedVec gamma3Star (m3InvPoly gamma3Star) (thetaParam 1 (1 / 14
      ↪ : ℝ) gamma3Star) y1 y2 y3 y4).x21 -
             (hpReducedVec gamma3Star (m3InvPoly gamma3Star) (thetaParam 1 (1 / 14
           ↪ : ℝ) gamma3Star) y1 y2 y3 y4).x12 -
             (hpReducedVec gamma3Star (m3InvPoly gamma3Star) (thetaParam 1 (1 / 14
           ↪ : ℝ) gamma3Star) y1 y2 y3 y4).x13)
    (hx21 : 0 ≤ (hpReducedVec gamma3Star (m3InvPoly gamma3Star) (thetaParam 1 (1 / 14
      ↪ : ℝ) gamma3Star) y1 y2 y3 y4).x21)
    (hx22 : 0 ≤ (hpReducedVec gamma3Star (m3InvPoly gamma3Star) (thetaParam 1 (1 / 14
      ↪ : ℝ) gamma3Star) y1 y2 y3 y4).x22)
    (hx32 : 0 ≤ (hpReducedVec gamma3Star (m3InvPoly gamma3Star) (thetaParam 1 (1 / 14
      ↪ : ℝ) gamma3Star) y1 y2 y3 y4).x32)
    (hx33 : 0 ≤ (hpReducedVec gamma3Star (m3InvPoly gamma3Star) (thetaParam 1 (1 / 14
      ↪ : ℝ) gamma3Star) y1 y2 y3 y4).x33) :
    evalMFreeF 1 (1 / 14 : ℝ) (hpReducedVec gamma3Star (m3InvPoly gamma3Star)
      ↪ (thetaParam 1 (1 / 14 : ℝ) gamma3Star) y1 y2 y3 y4) (wParam 1 (1 / 14 : ℝ)
      ↪ gamma3Star) ≤ 0
```

Assumptions: x(y) is hpReducedVec with gI = m3InvPoly, the polynomial that equals 1/gamma at the root (m3_inv_eq), so that it is the reduced family of Section 2; gamma3Star is the root of P3_star in [0.715, 0.720]; any reals y1, y2, y3, y4 with the five inequalities. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.m4_sos_nonpos (SeymourSNC/HuangPengBarrier.lean:1907-1920) src:66e59a9f6002fe

```
theorem m4_sos_nonpos (y1 y2 y3 y4 : ℝ)
    (hi4 : 0 ≤ (hpReducedVec gamma4Star (m4InvPoly gamma4Star) (thetaParam 2 (1 / 25
      ↪ : ℝ) gamma4Star) y1 y2 y3 y4).x21 -
           (hpReducedVec gamma4Star (m4InvPoly gamma4Star) (thetaParam 2 (1 / 25
        ↪ : ℝ) gamma4Star) y1 y2 y3 y4).x12 -
           (hpReducedVec gamma4Star (m4InvPoly gamma4Star) (thetaParam 2 (1 / 25
        ↪ : ℝ) gamma4Star) y1 y2 y3 y4).x13)
    (hx21 : 0 ≤ (hpReducedVec gamma4Star (m4InvPoly gamma4Star) (thetaParam 2 (1 / 25
      ↪ : ℝ) gamma4Star) y1 y2 y3 y4).x21)
    (hx22 : 0 ≤ (hpReducedVec gamma4Star (m4InvPoly gamma4Star) (thetaParam 2 (1 / 25
      ↪ : ℝ) gamma4Star) y1 y2 y3 y4).x22)
    (hx32 : 0 ≤ (hpReducedVec gamma4Star (m4InvPoly gamma4Star) (thetaParam 2 (1 / 25
      ↪ : ℝ) gamma4Star) y1 y2 y3 y4).x32)
    (hx33 : 0 ≤ (hpReducedVec gamma4Star (m4InvPoly gamma4Star) (thetaParam 2 (1 / 25
      ↪ : ℝ) gamma4Star) y1 y2 y3 y4).x33) :
    evalMFreeF 2 (1 / 25 : ℝ) (hpReducedVec gamma4Star (m4InvPoly gamma4Star)
      ↪ (thetaParam 2 (1 / 25 : ℝ) gamma4Star) y1 y2 y3 y4) (wParam 2 (1 / 25 : ℝ)
      ↪ gamma4Star) ≤ 0
```

Assumptions: x(y) is hpReducedVec with gI = m4InvPoly, the polynomial that equals 1/gamma at the root (m4_inv_eq), so that it is the reduced family of Section 2; gamma4Star is the root of P4_star in [0.715, 0.720]; any reals y1, y2, y3, y4 with the five inequalities. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.m5_sos_nonpos (SeymourSNC/HuangPengBarrier.lean:2222-2235) src:50cbfe2137e9ad

```
theorem m5_sos_nonpos (y1 y2 y3 y4 : ℝ)
    (hi4 : 0 ≤ (hpReducedVec gamma5Star (m5InvPoly gamma5Star) (thetaParam 3 (2 / 75
      ↪ : ℝ) gamma5Star) y1 y2 y3 y4).x21 -
           (hpReducedVec gamma5Star (m5InvPoly gamma5Star) (thetaParam 3 (2 / 75
        ↪ : ℝ) gamma5Star) y1 y2 y3 y4).x12 -
           (hpReducedVec gamma5Star (m5InvPoly gamma5Star) (thetaParam 3 (2 / 75
        ↪ : ℝ) gamma5Star) y1 y2 y3 y4).x13)
    (hx21 : 0 ≤ (hpReducedVec gamma5Star (m5InvPoly gamma5Star) (thetaParam 3 (2 / 75
      ↪ : ℝ) gamma5Star) y1 y2 y3 y4).x21)
    (hx22 : 0 ≤ (hpReducedVec gamma5Star (m5InvPoly gamma5Star) (thetaParam 3 (2 / 75
      ↪ : ℝ) gamma5Star) y1 y2 y3 y4).x22)
    (hx32 : 0 ≤ (hpReducedVec gamma5Star (m5InvPoly gamma5Star) (thetaParam 3 (2 / 75
      ↪ : ℝ) gamma5Star) y1 y2 y3 y4).x32)
    (hx33 : 0 ≤ (hpReducedVec gamma5Star (m5InvPoly gamma5Star) (thetaParam 3 (2 / 75
      ↪ : ℝ) gamma5Star) y1 y2 y3 y4).x33) :
    evalMFreeF 3 (2 / 75 : ℝ) (hpReducedVec gamma5Star (m5InvPoly gamma5Star)
      ↪ (thetaParam 3 (2 / 75 : ℝ) gamma5Star) y1 y2 y3 y4) (wParam 3 (2 / 75 : ℝ)
      ↪ gamma5Star) ≤ 0
```

Assumptions: x(y) is hpReducedVec with gI = m5InvPoly, the polynomial that equals 1/gamma at the root (m5_inv_eq), so that it is the reduced family of Section 2; gamma5Star is the root of P5_star in [0.715, 0.720]; any reals y1, y2, y3, y4 with the five inequalities. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.OrientedGraph.sum_secondInDeg_eq_sum_secondOutDeg (SeymourSNC/ConverseAndCirculants.lean:51-67) src:f93b1e789c975a

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (D : OrientedGraph V) [DecidableRel D.adj]`

```
theorem sum_secondInDeg_eq_sum_secondOutDeg :
    ∑ v : V, D.secondInDeg v = ∑ v : V, D.secondOutDeg v
```

Assumptions: D is an oriented graph on a finite type with decidable equality and adjacency. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.OrientedGraph.converse_seymourTight_iff_le (SeymourSNC/ConverseAndCirculants.lean:83-104) src:0e0ca52a3aa22d

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (D : OrientedGraph V) [DecidableRel D.adj]`

```
theorem converse_seymourTight_iff_le
    (hE : D.IsEulerian) (hST : D.IsSeymourTight) :
    D.converse.IsSeymourTight ↔ ∀ v : V, D.secondInDeg v ≤ D.inDeg v
```

Assumptions: D as in Lemma recip, Eulerian (out-degree equals in-degree at every vertex) and Seymour-tight. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.OrientedGraph.dense_regular_converse_seymourTight (SeymourSNC/ConverseAndCirculants.lean:319-330) src:12eb9b8ecf391e

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (D : OrientedGraph V) [DecidableRel D.adj]`

```
theorem dense_regular_converse_seymourTight
    {k : ℕ} (hle : Fintype.card V ≤ 2 * k + 2)
    (hreg : D.IsRegular k) (hST : D.IsSeymourTight) :
    D.converse.IsSeymourTight
```

Assumptions: D as in Lemma recip; a natural number k with |V| <= 2k + 2; D is k-regular and Seymour-tight. k = 0 is allowed. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.OrientedGraph.regular_three_order_nine_converse_seymourTight (SeymourSNC/ConverseAndCirculants.lean:334-464) src:ca651f76c9aab8

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V] (D : OrientedGraph V) [DecidableRel D.adj]`

```
theorem regular_three_order_nine_converse_seymourTight
    (hcard : Fintype.card V = 9)
    (hreg : D.IsRegular 3) (hST : D.IsSeymourTight) :
    D.converse.IsSeymourTight
```

Assumptions: D as in Lemma recip with |V| = 9, 3-regular, Seymour-tight. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.lexProd_D6_period3_rigidity (SeymourSNC/ConverseAndCirculants.lean:865-966) src:42ccd8eba884d7

Context binders in scope: `variable {F : Fin 6 → Type*} [∀ i, Fintype (F i)] [∀ i, DecidableEq (F i)] [∀ i, Nonempty (F i)] (G : ∀ i : Fin 6, OrientedGraph (F i)) [∀ i, DecidableRel (G i).adj]`

```
theorem lexProd_D6_period3_rigidity
    (hper : ∀ i : Fin 6, Fintype.card (F (i + 3)) = Fintype.card (F i))
    (hSSNC0 : (∃ x : F 0, (G 0).IsSeymourVertex x) ∨ (∃ x : F 3, (G
      ↪ 3).IsSeymourVertex x))
    (hSSNC1 : (∃ x : F 1, (G 1).IsSeymourVertex x) ∨ (∃ x : F 4, (G
      ↪ 4).IsSeymourVertex x))
    (hSSNC2 : (∃ x : F 2, (G 2).IsSeymourVertex x) ∨ (∃ x : F 5, (G
      ↪ 5).IsSeymourVertex x)) :
    (((lexProd D6 G).IsEulerian ∧ (lexProd D6 G).IsSeymourTight) ↔
      ((∀ i j : Fin 6, Fintype.card (F i) = Fintype.card (F j)) ∧
       (∀ i : Fin 6, (G i).IsEulerian ∧ (G i).IsSeymourTight))) ∧
    ((∀ i j : Fin 6, Fintype.card (F i) = Fintype.card (F j)) →
      ((lexProd D6 G).converse.IsSeymourTight ↔ ∀ i : Fin 6, (G
        ↪ i).converse.IsSeymourTight))
```

Assumptions: Six finite nonempty vertex types F i with decidable equality and oriented graphs G i on them with decidable adjacency; |F (i + 3)| = |F i| for i in Fin 6; for r = 0, 1, 2 a Seymour vertex in G r or in G (r + 3). lexProd is the lexicographic blowup. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.emptyBlowup_exists_strong_iff (SeymourSNC/FarkasEngine.lean:208-215) src:5c6deb8cc45133

Context binders in scope: `variable {I J : Type*} ; variable [Fintype I] [DecidableEq I]`

```
theorem emptyBlowup_exists_strong_iff (Q : OrientedGraph I) [DecidableRel Q.adj]
    (n : I → ℕ) (hn : ∀ j, 1 ≤ n j) :
    (∃ v, (emptyBlowup Q n).IsStrongSeymourVertex v) ↔ ∃ i : I, ¬
      ↪ HasBlowupHallViolation Q n i
```

Assumptions: Q is an oriented graph on a finite type with decidable equality and adjacency; n assigns to each vertex a natural number at least 1. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.ttBlowup_exists_strong_iff (SeymourSNC/FarkasEngine.lean:218-259) src:188b15c055823d

Context binders in scope: `variable {I J : Type*} ; variable [Fintype I] [DecidableEq I]`

```
theorem ttBlowup_exists_strong_iff (Q : OrientedGraph I) [DecidableRel Q.adj]
    (n : I → ℕ) (hn : ∀ j, 1 ≤ n j) :
    (∃ v, (ttBlowup Q n).IsStrongSeymourVertex v) ↔ ∃ i : I, ¬ HasBlowupHallViolation
      ↪ Q n i
```

Assumptions: As in Lemma hall. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.regular_tournament_matching_lower_bound (SeymourSNC/StrongSeymourAndZeroSlack.lean:115-247) src:6bdf415bc81f42

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem regular_tournament_matching_lower_bound
    (T : OrientedGraph V) [DecidableRel T.adj] {k : ℕ} (hk : 1 ≤ k)
    (hT : T.IsTournament) (hreg : T.IsRegular k) (v : V) :
    T.MaxMatchingAtLeast v (min k ((2 * k + 4) / 3))
```

Assumptions: T is an oriented graph on a finite type with decidable equality and decidable adjacency, a tournament, k-regular with 1 <= k; v any vertex. The bound is written min k ((2 k + 4) / 3) with natural division. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.regular_tournament_all_strong_of_le_four (SeymourSNC/StrongSeymourAndZeroSlack.lean:251-258) src:47fe5604877983

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem regular_tournament_all_strong_of_le_four
    (T : OrientedGraph V) [DecidableRel T.adj] {k : ℕ} (hk1 : 1 ≤ k) (hk4 : k ≤ 4)
    (hT : T.IsTournament) (hreg : T.IsRegular k) (v : V) :
    T.IsStrongSeymourVertex v
```

Assumptions: T as in Theorem matching (an oriented graph on a finite type with decidable equality and decidable adjacency, a k-regular tournament), with 1 <= k <= 4; v any vertex. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.R11Star_vertex0_exact_matching (SeymourSNC/StrongSeymourAndZeroSlack.lean:310-328) src:99aa37592e9738

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem R11Star_vertex0_exact_matching : R11Star.HasExactMaxMatching 0 4
```

Assumptions: No hypotheses: R11Star is the tournament on Fin 11 of Section 5; HasExactMaxMatching 0 4 means a matching of 4 arcs exists and none of 5. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.R11Star_other_vertices_strong (SeymourSNC/StrongSeymourAndZeroSlack.lean:358-369) src:abfc8f70261074

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem R11Star_other_vertices_strong :
    ∀ v : Fin 11, v ≠ 0 → R11Star.IsStrongSeymourVertex v
```

Assumptions: Every v in Fin 11 with v different from 0. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.theorem25_k_le_five_impossibility (SeymourSNC/FarkasDzitsoev.lean:421-433) src:fcded6d17d6f6d

```
theorem theorem25_k_le_five_impossibility :
    (∀ (Q : OrientedGraph (Fin 1)) [DecidableRel Q.adj], Q.IsTournament →
      ∀ n : Fin 1 → ℕ, (∀ i, 1 ≤ n i) → (∃ v, (emptyBlowup Q n).IsStrongSeymourVertex
        ↪ v) ∧ (∃ v, (ttBlowup Q n).IsStrongSeymourVertex v)) ∧
    (∀ (Q : OrientedGraph (Fin 2)) [DecidableRel Q.adj], Q.IsTournament →
      ∀ n : Fin 2 → ℕ, (∀ i, 1 ≤ n i) → (∃ v, (emptyBlowup Q n).IsStrongSeymourVertex
        ↪ v) ∧ (∃ v, (ttBlowup Q n).IsStrongSeymourVertex v)) ∧
    (∀ (Q : OrientedGraph (Fin 3)) [DecidableRel Q.adj], Q.IsTournament →
      ∀ n : Fin 3 → ℕ, (∀ i, 1 ≤ n i) → (∃ v, (emptyBlowup Q n).IsStrongSeymourVertex
        ↪ v) ∧ (∃ v, (ttBlowup Q n).IsStrongSeymourVertex v)) ∧
    (∀ (Q : OrientedGraph (Fin 4)) [DecidableRel Q.adj], Q.IsTournament →
      ∀ n : Fin 4 → ℕ, (∀ i, 1 ≤ n i) → (∃ v, (emptyBlowup Q n).IsStrongSeymourVertex
        ↪ v) ∧ (∃ v, (ttBlowup Q n).IsStrongSeymourVertex v)) ∧
    (∀ (Q : OrientedGraph (Fin 5)) [DecidableRel Q.adj], Q.IsTournament →
      ∀ n : Fin 5 → ℕ, (∀ i, 1 ≤ n i) → (∃ v, (emptyBlowup Q n).IsStrongSeymourVertex
        ↪ v) ∧ (∃ v, (ttBlowup Q n).IsStrongSeymourVertex v))
```

Assumptions: For each k in 1..5: any tournament Q on Fin k with decidable adjacency and any n : Fin k -> N with every n i >= 1. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.theorem25_k6_dzitsoev_uniqueness_and_min36 (SeymourSNC/FarkasDzitsoev.lean:639-675) src:58088cb5f4b522

```
theorem theorem25_k6_dzitsoev_uniqueness_and_min36 (Q : OrientedGraph (Fin 6))
  ↪ [DecidableRel Q.adj]
    (hT : Q.IsTournament) (n : Fin 6 → ℕ) (hn : ∀ i, 1 ≤ n i)
    (hNoStrong : (¬ ∃ v, (emptyBlowup Q n).IsStrongSeymourVertex v) ∨
                 (¬ ∃ v, (ttBlowup Q n).IsStrongSeymourVertex v)) :
    Nonempty (Q.Iso DzitsoevQ) ∧ 36 ≤ ∑ i : Fin 6, n i
```

Assumptions: Any tournament Q on Fin 6 with decidable adjacency and n : Fin 6 -> N with every n i >= 1; the hypothesis is the disjunction of the two no-strong-vertex conditions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.dzitsoev_36_witness (SeymourSNC/FarkasDzitsoev.lean:689-708) src:210e7271b74cd5

```
theorem dzitsoev_36_witness :
    (∑ i : Fin 6, dzitsoev36Sizes i = 36) ∧
    (emptyBlowup DzitsoevQ dzitsoev36Sizes).arcCount = 509 ∧
    (ttBlowup DzitsoevQ dzitsoev36Sizes).arcCount = 630 ∧
    (¬ ∃ v, (emptyBlowup DzitsoevQ dzitsoev36Sizes).IsStrongSeymourVertex v) ∧
    (¬ ∃ v, (ttBlowup DzitsoevQ dzitsoev36Sizes).IsStrongSeymourVertex v)
```

Assumptions: No hypotheses: n is dzitsoev36Sizes = (7, 3, 11, 3, 9, 3). Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.theorem25_dzitsoev_del1_rigidity (SeymourSNC/FarkasDzitsoev.lean:759-779) src:082e5a04626fb3

```
theorem theorem25_dzitsoev_del1_rigidity (u v : Fin 6) (harc : DzitsoevQ.adj u v)
    (n : Fin 6 → ℕ) (hn : ∀ i, 1 ≤ n i) :
    ((u ≠ 0 ∨ v ≠ 5) →
      (∃ x, (emptyBlowup (delOneArc DzitsoevQ u v) n).IsStrongSeymourVertex x) ∧
      (∃ x, (ttBlowup (delOneArc DzitsoevQ u v) n).IsStrongSeymourVertex x)) ∧
    ((u = 0 ∧ v = 5) →
      ((¬ ∃ x, (emptyBlowup (delOneArc DzitsoevQ u v) n).IsStrongSeymourVertex x) ∨
       (¬ ∃ x, (ttBlowup (delOneArc DzitsoevQ u v) n).IsStrongSeymourVertex x)) →
      51 ≤ ∑ i : Fin 6, n i)
```

Assumptions: u, v in Fin 6 with DzitsoevQ.adj u v; n : Fin 6 -> N with every n i >= 1. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.dzitsoev_del05_51_witness (SeymourSNC/FarkasDzitsoev.lean:794-811) src:5dfb649c3262f6

```
theorem dzitsoev_del05_51_witness :
    (∑ i : Fin 6, dzitsoevDel05Sizes i = 51) ∧
    (emptyBlowup (delOneArc DzitsoevQ 0 5) dzitsoevDel05Sizes).arcCount = 985 ∧
    (¬ ∃ v, (emptyBlowup (delOneArc DzitsoevQ 0 5)
      ↪ dzitsoevDel05Sizes).IsStrongSeymourVertex v) ∧
    (¬ ∃ v, (ttBlowup (delOneArc DzitsoevQ 0 5)
      ↪ dzitsoevDel05Sizes).IsStrongSeymourVertex v)
```

Assumptions: No hypotheses: n is dzitsoevDel05Sizes = (8, 7, 16, 3, 14, 3). Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.theorem25_dzitsoev_del2_rigidity (SeymourSNC/FarkasDeletions.lean:315-380) src:48b20c6f52b5fd

```
theorem theorem25_dzitsoev_del2_rigidity (u1 v1 u2 v2 : Fin 6)
    (h1 : DzitsoevQ.adj u1 v1) (h2 : DzitsoevQ.adj u2 v2)
    (hne : (u1, v1) ≠ (u2, v2))
    (n : Fin 6 → ℕ) (hn : ∀ i, 1 ≤ n i) :
    (∃ x, (emptyBlowup (delTwoArcs DzitsoevQ u1 v1 u2 v2) n).IsStrongSeymourVertex x)
      ↪ ∧
    (∃ x, (ttBlowup (delTwoArcs DzitsoevQ u1 v1 u2 v2) n).IsStrongSeymourVertex x)
```

Assumptions: Two arcs (u1, v1) and (u2, v2) of DzitsoevQ with (u1, v1) different from (u2, v2); n : Fin 6 -> N with every n i >= 1. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.zero_slack_multiplicity_lemma (SeymourSNC/StrongSeymourAndZeroSlack.lean:401-430) src:7f52377ea2cf31

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem zero_slack_multiplicity_lemma
    (D : OrientedGraph V) [DecidableRel D.adj] {v₀ u₀ : V}
    (hu₀ : u₀ ∈ D.outNbr v₀)
    (hzero : D.secondOutDeg v₀ = D.outDeg v₀)
    (hWSNC : HasWeightedSNC D (unitPerturbWeight u₀)) :
    ∃ v₁ : V, v₁ ≠ v₀ ∧ D.IsSeymourVertex v₁
```

Assumptions: D an oriented graph on a finite type with decidable equality and adjacency; u0 in N+(v0); |N++(v0)| = |N+(v0)|; HasWeightedSNC D (unitPerturbWeight u0). Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.unique_seymourVertex_sink_or_strict_slack (SeymourSNC/StrongSeymourAndZeroSlack.lean:435-451) src:42ebbd889780bb

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem unique_seymourVertex_sink_or_strict_slack
    (D : OrientedGraph V) [DecidableRel D.adj] {v₀ : V}
    (hWSNC : ∀ u : V, HasWeightedSNC D (unitPerturbWeight u))
    (hSey : D.IsSeymourVertex v₀)
    (huniq : ∀ v : V, D.IsSeymourVertex v → v = v₀) :
    D.outDeg v₀ = 0 ∨ D.outDeg v₀ < D.secondOutDeg v₀
```

Assumptions: D as in Lemma zeroslack; HasWeightedSNC D (unitPerturbWeight u) for every vertex u; v0 a Seymour vertex and every Seymour vertex equal to v0. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SeymourSNC.halkiewicz_pisa_ne_sc_seymourTight_at_six (SeymourSNC/StrongSeymourAndZeroSlack.lean:596-600) src:d1424e15f50d69

Context binders in scope: `variable {V : Type*} [Fintype V] [DecidableEq V]`

```
theorem halkiewicz_pisa_ne_sc_seymourTight_at_six :
    (D6.IsPisaGraph ∧ D6.IsSeymourTight) ∧
    (PisaNonTight6.IsPisaGraph ∧ ¬ PisaNonTight6.IsSeymourTight)
```

Assumptions: No hypotheses: D6 and PisaNonTight6 on Fin 6. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

Table 3. Lean 4 Type-Trap Audit Checklist — Source-Level Approximation (Gate Replay Pending).[a]

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
`regular_tournament_matching_lower_bound`	`nat_div`	`/ 3`	intended_convention	Natural-number division rounds down, and floor((2k + 4)/3) = ceil((2k + 2)/3) for every natural k, the bound printed in Theorem matching.
`D6`	`nat_sub`	`- i.val`	intended_convention	In (j.val + 6 - i.val) % 6 the subtraction never truncates because i.val < 6; the value is (j - i) mod 6, as in the definition of C_6(1, 2).
`D6`	`nat_mod`	`% 6`	intended_convention	Reduction modulo 6 of a natural number, the intended residue.
`TourMask`	`nat_div`	`/ /`	intended_convention	Not a division: the scan reads the bar // of the subtype notation as a division sign.
`TourMask`	`nat_div`	`/ (∀ i, f i i = false)`	intended_convention	Not a division: the subtype bar again.
`evalLemma22F`	`nat_div`	`/ 2`	intended_convention	The expression is real-valued; 1 / 2 is the real number one half, as in Huang and Peng.
`evalMFreeF`	`nat_div`	`/ 2`	intended_convention	Real-valued; w * K / 2 is real division.
`unitPerturbWeight`	`nat_div`	`/ 2`	intended_convention	The weight is rational; 3 / 2 is the rational number three halves.
`wParam`	`nat_div`	`/ (1 + K * α * (2 - α))`	intended_convention	Real division, total in Lean (x / 0 = 0). Definition mfree takes K >= 0 and 0 <= alpha <= 1, where 1 + K alpha (2 - alpha) >= 1; it equals 1 at (K, alpha) = (0, 0) and exceeds 1 at (1, 1/14), (2, 1/25), (3, 2/75) and at the pairs of Table mfree.

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
gammaHP	classical_choose	Classical.choose	intended_convention	gammaHP is chosen from a theorem of unique existence (ExistsUnique) of a root in the stated interval, so every choice gives the same real number.
gamma3Star	classical_choose	Classical.choose	intended_convention	gamma3Star is chosen from a theorem of unique existence (ExistsUnique) of a root in the stated interval, so every choice gives the same real number.
gamma4Star	classical_choose	Classical.choose	intended_convention	gamma4Star is chosen from a theorem of unique existence (ExistsUnique) of a root in the stated interval, so every choice gives the same real number.
gamma5Star	classical_choose	Classical.choose	intended_convention	gamma5Star is chosen from a theorem of unique existence (ExistsUnique) of a root in the stated interval, so every choice gives the same real number.
converseDecidableRel	local_instance_or_notation	converseDecidableRel	intended_convention	Decidability instance for the adjacency relation of the converse, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
circ9_123DecidableRel	local_instance_or_notation	circ9_123DecidableRel	intended_convention	Decidability instance for the adjacency relation of C_9(1, 2, 3), found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
c3BlowupE3DecidableRel	local_instance_or_notation	c3BlowupE3DecidableRel	intended_convention	Decidability instance for the adjacency relation of C_3[E_3, E_3, E_3], found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
d6DecidableRel	local_instance_or_notation	d6DecidableRel	intended_convention	Decidability instance for the adjacency relation of D6, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
lexProdDecidableRel	local_instance_or_notation	lexProdDecidableRel	intended_convention	Decidability instance for the adjacency relation of a lexicographic blowup, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
dzitsoevQDecidableRel	local_instance_or_notation	dzitsoevQDecidableRel	intended_convention	Decidability instance for the adjacency relation of Q15, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
delOneArcDecidableRel	local_instance_or_notation	delOneArcDecidableRel	intended_convention	Decidability instance for the adjacency relation of Q minus one arc, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
delTwoArcsDecidableRel	local_instance_or_notation	delTwoArcsDecidableRel	intended_convention	Decidability instance for the adjacency relation of Q minus two arcs, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
emptyBlowupDecidableRel	local_instance_or_notation	emptyBlowupDecidableRel	intended_convention	Decidability instance for the adjacency relation of Q[n], found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
ttBlowupDecidableRel	local_instance_or_notation	ttBlowupDecidableRel	intended_convention	Decidability instance for the adjacency relation of Q[TT_n], found by instance search; it decides the relation of the definition and no statement depends on which instance is used.

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
tourMaskGraphDecidableRel	local_instance_or_notation	tourMaskGraphDecidableRel	intended_convention	Decidability instance for the adjacency relation of a TourMask graph, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
r11StarDecidableRel	local_instance_or_notation	r11StarDecidableRel	intended_convention	Decidability instance for the adjacency relation of R11, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
c3GraphDecidableRel	local_instance_or_notation	c3GraphDecidableRel	intended_convention	Decidability instance for the adjacency relation of C_3, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.
pisaNonTight6DecidableRel	local_instance_or_notation	pisaNonTight6DecidableRel	intended_convention	Decidability instance for the adjacency relation of P6, found by instance search; it decides the relation of the definition and no statement depends on which instance is used.

[a] Covers the specified Lean 4 hazard set (Nat subtraction/division/mod, totalized real/rational division, Real.log/rpow, Finset.range, Classical.choose, local instances); no finite checklist certifies all semantics.

Table 4. Per-Claim Reader's Trust Boundary Ledger (G11 Denominators Derived Mechanically).[a]

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Thm. 1: For all rational numbers mu and w with gamma_HP < mu <= 1 and w >= 1 there are rational numbers x11, x12, x13, x14, x21, x22, x23, x24, x32, x33, x 34, all nonnegative, with x11 > 0, d2 < mu d1, d3 < mu^2 d1, x12 + x22 + x32 < mu (x11 + x21), x12 + x13 + x14 <= x21 and F(x; w) > 0, where d1 = x11 + x12 + x 13 + x14, d2 = x21 + x22 + x23 + x24 and d3 = x32 + x 33 + x34. Here gamma_HP is the root of P_HP in [0, 1].	Lean Kernel	SeymourSNC.hp_primal_barrier_rational	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Lem. 18: Let gamma = gamma_HP, w = gamma^2 + 2 gamma^3 and theta = 2 + 2 gamma - 4 gamma^3. For all real y1, y2, y3, y4, if the point x(y) of the reduced family satisfies x21 - x12 - x13 >= 0, x21 >= 0, x 22 >= 0, x32 >= 0 and x33 >= 0, then F(x(y); w) <= 0.	Lean Kernel	SeymourSNC.hp_sos_nonpos	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Lem. 19: Let D be a finite oriented graph, u and v vertices, w > 0, and suppose that w |N+(v) & X1| + |N+(v) & X2| <= w |N+(y) & X1| + |N+(y) & X2| for every y in X1, where X1 = N+(u) and X2 = N++(u). Put X11 = X1 & N+(v), X12 = X1 & N++(v) and X22 = X2 & N++(v). Then every y in X11 satisfies w (|X11| - |X12|) - |X22| <= w |N+(y) & X11|.	Lean Kernel	SeymourSNC.internal_outDeg_concentration_X11	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Lem. 20: For all real alpha >= 0, x11 > 0 and L, 2 (alpha x11) L - (alpha x 11)^2 <= max(0, L)^2.	Lean Kernel	`SeymourSNC.max_sq_ge_tangent`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 21: Let K = 1, alpha = 1/14, gamma = gamma_3* (the root in [0.715, 0.720] of P_3* = 4873442 gamma F_K(x(0, 0, 1, 0); w_K(gamma))), w = w_K(gamma) and theta = theta_K(gamma). For all real y1, y2, y3, y4, if x(y) satisfies x21 - x12 - x13 >= 0, x21 >= 0, x22 >= 0, x32 >= 0 and x33 >= 0, then F_K(x(y); w) <= 0.	Lean Kernel	`SeymourSNC.m3_sos_nonpos`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 22: The statement of Proposition sos3 with K = 2, alpha = 1/25 and gamma = gamma_4*.	Lean Kernel	`SeymourSNC.m4_sos_nonpos`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 23: The statement of Proposition sos3 with K = 3, alpha = 2/75 and gamma = gamma_5*.	Lean Kernel	`SeymourSNC.m5_sos_nonpos`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Thm. 2: For m = 3, 4, 5, 6, every m-free oriented graph has a vertex v with |N++(v)| >= c_m |N+(v)|, where c_3 = 7203/10000, c_4 = 361/500, c_5 = 1807/2500 and c_6 = 3617/5000.	Cert Only	sylvester.py (d7396b5934f3) [regenerable; replay pending]	External cert only; fidelity: n/a	Written reduction of Section 3 (Huang and Peng, Lemmas 2.1 to 2.3, and Liang and Xu, Lemma 1.7 [to check], as printed there), applied to the exact rational certificate of cert/sylvester.py.
Lem. 25: In every finite oriented graph, the sum over the vertices of |N−(v)| equals the sum over the vertices of |N++(v)|.	Lean Kernel	`SeymourSNC.OrientedGraph.sum_secondInDeg_eq_sum_secondOutDeg`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Lem. 26: Let D be a finite Eulerian Seymour-tight oriented graph. The converse of D is Seymour-tight if and only if |N−(v)| <= |N-(v)| for every vertex v.	Lean Kernel	`SeymourSNC.OrientedGraph.converse_seymourTight_iff_le`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Thm. 3: Let k be a natural number and D a k-regular Seymour-tight oriented graph with at most 2k + 2 vertices. Then the converse of D is Seymour-tight.	Lean Kernel	`SeymourSNC.OrientedGraph.dense_regular_converse_seymourTight`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 28: Let D be a 3-regular Seymour-tight oriented graph with 9 vertices. Then the converse of D is Seymour-tight.	Lean Kernel	`SeymourSNC.OrientedGraph.regular_three_order_nine_converse_seymourTight`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Prop. 27: Let k >= 1 and let D be an oriented graph on 2k + 3 vertices in which every vertex has out-degree k and in-degree k. For a vertex w let M(w) be the set of vertices from which w is at distance at least 3. Then |M(w)| <= 3, and if |M(w)| = 3 then D[M(w)] is a directed triangle and every vertex of M(w) dominates every vertex of T(w) outside M(w), where T(w) is the set of vertices other than w that do not dominate w.	Written	Written proof	Lean: none; fidelity: n/a	Written proof in this paper; not checked in Lean.
Thm. 4: Let k >= 1 and n >= 2k + 1, let G_0, ..., G_(n-1) be nonempty finite oriented graphs with s_i vertices, and let O be the lexicographic blowup of the circulant C_n(1, ..., k) by G_0, ..., G_(n-1). Put d = gcd(n, k). If s_(i+d) = s_i for all i, then O is Seymour-tight if and only if every G_i is Seymour-tight, O is Eulerian if and only if every G_i is Eulerian, and the converse of O is Seymour-tight if and only if the converse of every G_i is. If every G_i has a Seymour vertex and O is Seymour-tight, then s_(i+d) = s_i for all i.	Written	Written proof	Lean: none; fidelity: n/a	Written proof in this paper; not checked in Lean.
Prop. 29: Let D6 = C_6(1, 2) and let G_0, ..., G_5 be nonempty finite oriented graphs with |G_(i+3)| = |G_i| for all i, such that for each r in {0, 1, 2} one of G_r and G_(r+3) has a Seymour vertex. Then D6[G_0, ..., G_5] is Eulerian and Seymour-tight if and only if all G_i have the same number of vertices and every G_i is Eulerian and Seymour-tight; and if all G_i have the same number of vertices, the converse of D6[G_0, ..., G_5] is Seymour-tight if and only if the converse of every G_i is.	Lean Kernel	`SeymourSNC.lexProd_D6_period3_rigidity`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 30: Every Eulerian Seymour-tight oriented graph on one to six vertices (21 up to isomorphism), every labelled one on 7 vertices (7959) and every labelled 3-regular one on 8 vertices (5040) has a Seymour-tight converse.	Cert Only	census.py (bf4d7b04e8f8) [regenerable; replay pending]	External cert only; fidelity: n/a	Exhaustive enumeration of cert/census.py.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Lem. 34: Let Q be a finite oriented graph and n a vector of positive integers indexed by its vertices. Then Q[n] has a strong Seymour vertex if and only if some vertex i of Q has no Hall violation, where a Hall violation at i is a nonempty set S of out-neighbors of i with n(N+(S) & N++(i)) < n(S).	Lean Kernel	`SeymourSNC.emptyBlowup_exists_strong_iff`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Lem. 35: The statement of Lemma hall with Q[n] replaced by Q[TT_n], the blowup whose fibres are transitive tournaments.	Lean Kernel	`SeymourSNC.ttBlowup_exists_strong_iff`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Thm. 5: Let k >= 1, let T be a tournament in which every vertex has out-degree and in-degree k, and let v be a vertex. Then the arcs from N+(v) to N++(v) contain a matching with min(k, floor((2k + 4)/3)) arcs.	Lean Kernel	`SeymourSNC.regular_tournament_matching_lower_bound`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Cor. 31: If 1 <= k <= 4, every vertex of every tournament in which every vertex has out-degree and in-degree k is a strong Seymour vertex.	Lean Kernel	`SeymourSNC.regular_tournament_all_strong_of_le_four`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 32: In the tournament R11 on 11 vertices, a largest matching of arcs from N+(0) to N++(0) has exactly 4 arcs.	Lean Kernel	`SeymourSNC.R11Star_vertex0_exact_matching`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 33: Every vertex of R11 other than 0 is a strong Seymour vertex.	Lean Kernel	`SeymourSNC.R11Star_other_vertices_strong`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 36: For k = 1, ..., 5, every tournament Q on k vertices and every vector n of positive integers, both Q[n] and Q[TT_n] have a strong Seymour vertex.	Lean Kernel	`SeymourSNC.theorem25_k_le_five_impossibility`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Thm. 6: Let Q be a tournament on 6 vertices and n a vector of 6 positive integers. If Q[n] or Q[TT_n] has no strong Seymour vertex, then Q is isomorphic to Q15 and n_0 + ... + n_5 >= 36.	Lean Kernel	`SeymourSNC.theorem25_k6_dzitsoev_uniqueness_and_min36`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 37: For n = (7, 3, 11, 3, 9, 3), the blowups Q15[n] and Q15[TT_n] have 36 vertices, 509 and 630 arcs, and no strong Seymour vertex.	Lean Kernel	`SeymourSNC.dzitsoev_36_witness`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Prop. 38: Let (a, b) be an arc of Q15 and n a vector of 6 positive integers. If (a, b) is not (0, 5), both blowups of Q15 - (a, b) by n have a strong Seymour vertex. If (a, b) = (0, 5) and one of the two blowups has no strong Seymour vertex, then n_0 + ... + n_5 >= 51.	Lean Kernel	`SeymourSNC.theorem25_dzitsoev_del1_rigidity`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 39: For n = (8, 7, 16, 3, 14, 3), the blowups Q14[n] and Q14[TT_n] have no strong Seymour vertex; Q14[n] has 51 vertices and 985 arcs.	Lean Kernel	`SeymourSNC.dzitsoev_del05_51_witness`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 40: For any two different arcs of Q15 and any vector n of 6 positive integers, both blowups of Q15 minus the two arcs by n have a strong Seymour vertex.	Lean Kernel	`SeymourSNC.theorem25_dzitsoev_del2_rigidity`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Thm. 7: Let Q be an oriented graph on one to six vertices. Then Q has a vector n of positive integers for which Q[n] has no strong Seymour vertex exactly when Q is isomorphic to Q15 or to Q14, and the same holds with Q[TT_n] in place of Q[n].	Cert Only	quotients.py (46b982fc3433) [regenerable; replay pending]	External cert only; fidelity: n/a	Written reduction to strongly connected quotients and Lemmas hall and halltt (Lean), applied to the exact classification of cert/quotients.py.
Thm. 8: Every vertex of every circulant tournament is a strong Seymour vertex.	Written	Written proof	Lean: none; fidelity: n/a	Written proof in this paper (self-contained); not checked in Lean. The statement is an immediate corollary of Losonczy (1998) [to check] and of Eliahou and Lecouvey, Theorem 3.2.
Prop. 41: At vertex 0 of each of the 2046 circulant tournaments of odd order from 3 to 21, the arcs from N+(0) to N++(0) contain a matching covering N+(0).	Cert Only	circulants.py (b384cd8420e5) [regenerable; replay pending]	External cert only; fidelity: n/a	Exhaustive check of cert/circulants.py.
Lem. 42: Let D be a finite oriented graph, v0 a vertex with |N++(v0)| = |N+(v0)| and u0 an out-neighbor of v 0. If D satisfies the weighted second neighborhood property for the weight that is 3/2 at u0 and 1 elsewhere, then D has a Seymour vertex other than v0.	Lean Kernel	`SeymourSNC.zero_slack_multiplicity_lemma`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Cor. 43: Let D be a finite oriented graph that satisfies the weighted second neighborhood property for each of the weights of Lemma zeroslack. If v0 is the only Seymour vertex of D, then |N+(v0)| = 0 or |N+(v0)| < |N++(v0)|.	Lean Kernel	`SeymourSNC.unique_seymourVertex_sink_or_strict_slack`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.
Prop. 46: D6 is a Pisa graph and Seymour-tight; the 13-arc graph P6 is a Pisa graph and not Seymour-tight.	Lean Kernel	`SeymourSNC.halkiewicz_pisa_ne_sc_seymourTight_at_six`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 kernel with the axioms propext, Classical.choice and Quot.sound; the definitions of Section 2; the hypotheses as stated.

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Lem. 44: Let D be an oriented graph without a sink in which v0 is the only Seymour vertex. Then every vertex v other than v0 has out-degree at least 3, and if v0 has out-degree 1, with out-neighbor u, then |N++(v0)| = |N+(u)| >= 3.	Written	Written proof	Lean: none; fidelity: n/a	Written proof in this paper; not checked in Lean.
Prop. 45: Let D be an oriented graph without a sink in which v0 is the only Seymour vertex, put s = sigma(v0), and suppose that D satisfies the weighted second neighborhood property for every positive rational weight. Then s >= 1, and there are vertices v1 and v2 other than v0 with |N++(v1) & N+(v0)| - |N+(v1) & N+(v0)| >= ceil(|N+(v0)| (-sigma(v1)) / s) and |N+(v2) & N++(v0)| - |N++(v2) & N++(v0)| >= ceil(|N++(v0)| (-sigma(v2)) / s).	Written	Written proof	Lean: none; fidelity: n/a	Written proof in this paper; not checked in Lean.
Thm. 9: Every oriented graph on at most 7 vertices without a sink has at least two Seymour vertices.	Cert Only	multiplicity.py (a046e0fabc7a) [regenerable; replay pending]; census.py (bf4d7b04e8f8) [regenerable; replay pending]	External cert only; fidelity: n/a	Lemma forcing (written) and the exhaustive searches of cert/multiplicity.py (7 vertices) and cert/census.py (at most 6 vertices).
Prop. 47: Every nonempty oriented graph on at most 6 vertices and every nonempty tournament on at most 8 vertices has a strong Seymour vertex and a vertex v with |N++(v)| >= |N-(v)|, every nonempty sink-free oriented graph on at most 6 vertices has at least two Seymour vertices, and the counts of Table census hold.	Cert Only	census.py (bf4d7b04e8f8) [regenerable; replay pending]	External cert only; fidelity: n/a	Exhaustive enumeration of cert/census.py.

[a] Allowed pure-kernel axioms: `propext`, `Classical.choice`, `Quot.sound`. Actual axioms printed above are read directly from verified outputs. Provenance nonce: 334eea35cf14 | Lean mode: committed_receipt | Closure SHA-256: b8374d7f2e59 | Git anchor: 6d132f9d40ae | Axioms SHA-256: e54e8e8dfd4a | Fidelity SHA-256: ac0f9466e0e8.

6. PRIOR WORK AND OPEN QUESTIONS

What was read. Read in full: Huang and Peng [6] and Bourneuf, Charbit, Coulomb and Thomassé [13]. Read in the parts cited, whose locators are marked [to check]: Liang and Xu [7] (Section 1, Lemmas 1.5 to 1.7 [to check]; their Lemma 1.5 is from Hamburger, Haxell and Kostochka [8], not read), Seacrest [9] (Lemma 4 and Theorem 3 [to check]), Guo, Kang and Zwaneveld [10] (Section 4 and Conjecture 8.2 [to check]), Bai, Li and Park [11] (Remark 3.1 [to check] and the statements), the OpenAI preprint [12] (the statements and Corollary 5.1 [to check]), Cary [18] (Theorem 2 [to check]), Halkiewicz's third version [17] (the definition of Pisa graphs and the tables), Brukhman [14], Sadhukhan and Sen [15] and Mezher and Daamouch [16] (abstracts and main statements), Eliahou and Lecouvey [20] (read in full, as arXiv:math/0611614v1), Losonczy [19] (the statement only, in the zbMATH review and as Theorem 1.2 of Eliahou and Lecouvey; the article was not read) [to check], and the repositories of Gibbons [22], sunnyspot [23], Morten-Ness [24], Labianco [25], wustep [26] and njallskarp [27] (their readme files and the notes named in `references-checked.md`). Not read: Dean and Latka, Fisher, Havet and Thomassé, Kaneko and Locke, Chen, Shen and Yuster, the article of Losonczy, and the book of Nathanson for the theorems of Cauchy–Davenport and Vosper; they are cited [to check].

Relation to these sources. The two-vertex system and Lemma 18 are Huang and Peng's; our solutions in Theorem 1 satisfy the stronger requirement that all eleven variables are nonnegative. Brukhman proves the conjecture for oriented graphs on

at most $2\delta^+ + 2$ vertices and Sadhukhan and Sen for minimum out-degree at most 7; Mezher and Daamouch treat k-anti-transitive $(k-4)$-free graphs. Guo, Kang and Zwaneveld prove the converse statement for $k = 1, 2$, and prove that a blowup of an orientation by Seymour-tight graphs is Seymour-tight when the vector of fiber sizes is in the kernel of a matrix of the orientation, which covers circulants with periodic sizes; Theorem 4 adds the reverse implication and the necessity of periodic sizes when each fiber has a Seymour vertex, which holds for instance when the fibers are tournaments, by Fisher's theorem, or have minimum out-degree at most 6, by Kaneko and Locke, and for all fibers if the SNC preprints are correct. Labianco's repository treats the circulants $C_n(1, 2)$ as Pisa graphs, and the repository of wustep reports the numbers 1050 and 270 at order six. The tournament Q_{15} and its 36-vertex blowup are Dzitsoev's, recorded by Bai, Li and Park; the uniqueness among six-vertex tournaments and the minimum 36 are in the repository of njallskarp. Ness gives a 36-vertex tournament without a strong Seymour vertex, a blowup of a six-vertex tournament with circulant fibers, and sunnyspot gives tournaments of orders 21 and 24 without a strong Seymour vertex and reports that every regular tournament of order 13 has at least 11 strong Seymour vertices. For tournaments, the conclusion of Theorem 9 holds at every order by the theorem of Havet and Thomassé [to check].

Search. The prior-work search (25 November 2026) used Google Scholar through a keyed service, OEIS, arXiv, zbMATH Open and the repositories above; the keyed CORE search failed with a server error. Its log is `sources/searchlog.md`, with the query records in `sources/records/` and the receipt of the log check in `sources/searchcheck.txt`. Statements with the same content were found for Theorem 6 (njallskarp) and Theorem 5 (Gibbons, with unit weights). Theorem 8 was not reached by that search, which did not look at matchings in groups. A second search for it (25 November 2026, log `sources/circulant/searchlog.md`) found Losonczy's theorem and that of Eliahou and Lecouvey, which imply it at once (Section 1); the statement in the wording of tournaments or strong Seymour vertices was not found in the logged searches. The other main results were not found in the searches logged, in the forms stated here. Not searched: OpenAlex and Semantic Scholar (both refused requests with HTTP 429), Stack Exchange in the first search (throttled during the session), Zenodo, DataCite, HAL and OSF notes, MathSciNet, books, paywalled texts (among them the full text of Losonczy's article), the papers citing Losonczy or Eliahou and Lecouvey beyond the records returned, and posts on social media.

Open questions. (1) Can an oriented graph without a sink on 8 or more vertices have only one Seymour vertex? Bourneuf et al. announce that it cannot. (2) Conjecture 8.2 [to check] of Guo, Kang and Zwaneveld for $k \geq 3$, beyond the cases $n \leq 2k + 2$ and $(k, n) = (3, 9)$ proved here. (3) What is the smallest order of a nonempty oriented graph without a strong Seymour vertex? Blowups of quotients on one to six vertices with nonempty independent or transitive tournament fibers need at least 36 vertices, sunnyspot's examples have orders 21 and 24, and that repository states the bounds $14 \leq n \leq 21$ for oriented graphs. (4) Lean proofs of the written proofs and certificates of this paper.

7. DATA AVAILABILITY, REPRODUCIBILITY, AND USE OF AI

The Lean 4 library `SeymourSNC` (Lean `v4.35.0` with Mathlib) with its axiom report and build receipt is in the bundle `findings/seymour-second-neighborhood` of the `mathdisc` repository. The paper directory contains `recompute.py`, the certificate programs in `cert/` with their outputs, the figure scripts in `figures/`, the ledgers `claims.md` and `references-checked.md`, the search log and quotations in `sources/`, and the manifests behind Section 5. Running `python3 recompute.py` executes the Python box of Section 1 from this source and checks, in exact arithmetic, the root brackets, the signs of the certificates, the solutions of Theorem 1 at a grid of rational points and the named graphs; with `-a` it also reruns the five programs of `cert/` and compares their output with the committed files. Its output is `recompute.out`. Section 5 is written by `ourotrix paper trust -p . -g`, and `ourotrix paper check -p . -b -x -B` rebuilds the PDF and runs the checks recorded in `receipt.json`.

This draft carries over the content of an earlier draft in the finding bundle, which stays unchanged. That draft printed a different quadratic F for the two-vertex system (the Lean library and this draft use the one printed by Huang and Peng), relied on a certificate receipt that is not in the bundle, which the programs of `cert/` replace, and counted the seven-vertex search differently. `README.md` lists the changes.

AI research agents built and directed by Tyler B. Hughes generated and checked this work: they found the statements, wrote and audited the Lean library, wrote the certificate programs, and drafted this paper. No human mathematician has yet refereed the proof.

Drafted: November 2026 **Updated:** November 25, 2026

REFERENCES

(1) N. Dean and B. J. Latka, *Squaring the tournament—an open problem*, Congr. Numer. (1995), 73–80. [to check]

(2) D. C. Fisher, *Squaring a tournament: a proof of Dean's conjecture*, J. Graph Theory 23(1) (1996), 43–48. [to check]

(3) F. Havet and S. Thomassé, *Median orders of tournaments: a tool for the second neighborhood problem and Sumner's conjecture*, J. Graph Theory 35(4) (2000), 244–256. [to check]

(4) Y. Kaneko and S. C. Locke, *The minimum degree approach for Paul Seymour's distance 2 conjecture*, Congr. Numer. (2001), 201–206. [to check]

(5) G. Chen, J. Shen and R. Yuster, *Second neighborhood via first neighborhood in digraphs*, Ann. Comb. 7(1) (2003), 15–20. [to check]

(6) H. Huang and F. Peng, *An improved bound on Seymour's second neighborhood conjecture*, arXiv:2412.20234v1 (2024).

(7) H. Liang and J.-M. Xu, *On Seymour's second neighborhood conjecture of m-free digraphs*, Discrete Math. 340(8) (2017), 1944–1949; read as arXiv:1701.00328v1. [to check]

(8) P. Hamburger, P. Haxell and A. Kostochka, *On directed triangles in digraphs*, Electron. J. Combin. 14 (2007), #N19. [to check]

(9) T. Seacrest, *Seymour's second neighborhood conjecture for subsets of vertices*, arXiv:1808.06293v3 (2019). [to check]

(10) K. Guo, R. J. Kang and G. Zwaneveld, *Seymour-tight orientations*, arXiv:2603.29626v1 (2026). [to check]

(11) Y. Bai, B. Li and B. Park, *Towards a strengthening of the second neighborhood conjecture*, arXiv:2607.18047v2 (2026). [to check]

(12) OpenAI, *A proof of Seymour's second-neighborhood conjecture*, Manuscript 173, https://github.com/openai/math (23 September 2026). [to check]

(13) R. Bourneuf, P. Charbit, S. Coulomb and S. Thomassé, *A short proof of Seymour's Second Neighborhood Conjecture*, arXiv:2611.32872v1 (2026).

(14) J. Brukhman, *A dense-case theorem for Seymour's second neighborhood conjecture*, arXiv:2608.11530 (2026). [to check]

(15) A. Sadhukhan and S. Sen, *Locating a Seymour vertex in oriented graphs having minimum outdegree at most 7*, arXiv:2606.30588v2 (2026). [to check]

(16) D. Mezher and M. Daamouch, *A note on the second neighborhood problem for k-anti-transitive and m-free digraphs*, arXiv:2405.17797v1 (2024). [to check]

(17) S. M. S. Halkiewicz, *Split-twin extensions preserving Seymour vertices*, arXiv:2601.21563 (2026); version 3, the one read here, is titled *Extremal instances of Seymour's second neighborhood conjecture*, and the fourth version is withdrawn. [to check]

(18) M. Cary, *Vertices with the second neighborhood property in Eulerian digraphs*, arXiv:1711.01189v4 (2019). [to check]

(19) J. Losonczy, *On matchings in groups*, Adv. Appl. Math. 20(3) (1998), 385–391. [to check]

(20) S. Eliahou and C. Lecouvey, *Matchings in arbitrary groups*, Adv. Appl. Math. 40(2) (2008), 219–224; read as arXiv:math/0611614v1.

(21) M. B. Nathanson, *Additive Number Theory: Inverse Problems and the Geometry of Sumsets*, Graduate Texts in Mathematics 165, Springer, 1996. [to check]

(22) A. B. Gibbons, repository `AustinBGibbons/ssnc`, https://github.com/AustinBGibbons/ssnc (2026). [to check]

(23) sunnyspot114514, repository https://github.com/sunnyspot114514/strong-seymour-order13-verification (2026). [to check]

(24) Morten-Ness, repository `StrongSeymour`, https://github.com/Morten-Ness/StrongSeymour (2026). [to check]

(25) K. Labianco, repository `pisa-counterexamples`, https://github.com/klabianco/pisa-counterexamples (2026). [to check]

(26) wustep, repository `maths`, https://github.com/wustep/maths (2026). [to check]

(27) njallskarp, repository `math_source_code_open`, https://github.com/njallskarp/math_source_code_open (2026). [to check]

(28) OEIS Foundation, *The On-Line Encyclopedia of Integer Sequences*, sequences A001174 and A000568, https://oeis.org (accessed 25 November 2026).