

EXACT FINITE-RANK CERTIFICATES FOR TWO-COMPONENT LOGARITHMIC INVARIANT TWO-JET VANISHING

LEI HOU, PENGCHAO WANG, AND SONG-YAN XIE

ABSTRACT. This guide documents the exact computation used to prove the finite key vanishing statements in the accompanying paper on complements of two plane curves. It records the characteristic-zero reduction, the finite ambient normal form, the chart-transition and divisibility equations, the symmetry blocks, the modular full-rank argument, the roles of the generator and the independent structural checker, and commands sufficient to reproduce every certificate. No floating-point arithmetic enters the proof.

The completed archive contains six exact proof sets, all accepted by the structural checker and supplemented by direct independent reconstruction of representative matrices. The cubic SMT-57 theorem profile has 14 cases, 168 symmetry blocks, and 280,798 columns. The five unequal-degree sets contain respectively 1, 8, 36, 48, and 18 full-rank blocks. For the conic–quintic pair, the extra certified case $(m, t) = (8, 3)$ is precisely the additional input used in the paper to obtain the coefficient $A_{2,5} = 45$. The degree-one cases use valid line pairs with $A = Y$ and have 67,011 and 5,526 certified columns. The archive also records a complete source-bound rerun of all 168 cubic blocks, its exact pre-nomenclature producer, the current label-compatible source and executable, warning-clean rebuild data, and a complete SHA-256 manifest.

CONTENTS

1. Scope, status, and the logical certificate	1
1.1. The target	1
1.2. What a certificate proves	2
1.3. Completed certificate sets	2
2. Explicit SNC test pairs	2
3. Finite ambient normal form	3
4. Mixed-chart equations and exact divisibility	4
4.1. The logarithmic one-jet frame	4
4.2. The triangular Wronskian substitution	5
4.3. The polynomial remainder system	5
5. Symmetry decomposition	6
5.1. Two cubics	6
5.2. The $(2, e)$ pairs	6
5.3. The $(1, e)$ line pairs	6
6. Exact case and size tables	7
6.1. Cubic SMT-57 profile	7
6.2. Unequal-degree certificates	7
7. Uncomputed constant frontier and cluster plan	8
8. Why modular full rank proves characteristic-zero vanishing	8
9. Generator, certificate, and checker	9
9.1. Matrix generator	9
9.2. Independent structural checker	10
9.3. The separate $\mathcal{O}(3)$ identity supplement	10
10. Reproducibility	10
10.1. Recorded workstation	10
10.2. Warning-clean builds	11

10.3.	Internal self-tests	11
10.4.	Definitive proof runs	12
10.5.	Structural aggregation	12
10.6.	SHA-256 verification	13
11.	Frozen snapshot and archive manifest	13
11.1.	Audited source snapshot	13
11.2.	Proof-file hashes	13
12.	Finalization checklist	14
13.	Credit and relation to preceding computations	15
	References	15

1. SCOPE, STATUS, AND THE LOGICAL CERTIFICATE

1.1. **The target.** For an ordered simple-normal-crossing pair

$$D = C_1 + C_2 \subset \mathbb{P}^2, \quad \deg C_i = d_i,$$

the finite computation concerns

$$(1.1) \quad H^0(\mathbb{P}^2, E_{2,m} T_{\mathbb{P}^2}^*(\log D) \otimes \mathcal{O}_{\mathbb{P}^2}(-t)).$$

The six proof-effective degree pairs are

$$(d_1, d_2) \in \{(3, 3), (3, 4), (2, 6), (2, 5), (1, 8), (1, 9)\}.$$

For two cubics the SMT-57 range is

$$(1.2) \quad 1 \leq m \leq 14, \quad t = t_0(m) := \left\lceil \frac{m+1}{5} \right\rceil.$$

Equivalently, the minimal pairs are

t	m
1	1, 2, 3, 4
2	5, 6, 7, 8, 9
3	10, 11, 12, 13, 14.

For the other degree pairs the minimal pairs are

(d_1, d_2)	(m, t)
(3, 4)	(3, 3)
(2, 6)	(3, 3)
(2, 5)	(3, 2), (4, 2), (5, 2), (6, 3), (7, 3), (8, 3)
(1, 8)	(3, 2), (4, 2), (5, 2), (6, 3), (7, 3), (8, 3)
(1, 9)	(3, 3), (4, 3).

Vanishing at the least twist implies vanishing at every larger twist for the same m : multiplication by a nonzero form of degree $t - t_0$ injects a section twisted by $\mathcal{O}(-t)$ into the space twisted by $\mathcal{O}(-t_0)$.

1.2. **What a certificate proves.** For every degree–weight–twist–symmetry block, the program constructs an integral homogeneous system

$$(1.4) \quad M_{\mathbb{Z}} c = 0,$$

whose characteristic-zero kernel is the corresponding subspace of (1.1). The proof chain is

$$\begin{array}{c}
\text{global logarithmic two-jet section} \\
\Downarrow \\
\text{finite coefficient vector } c \text{ in the normal form} \\
\Downarrow \\
M_{\mathbb{Z}}c = 0 \\
\Downarrow \quad \text{rank}_{\mathbb{F}_p}(M_{\mathbb{Z}} \bmod p) = \#\text{columns} \\
c = 0 \text{ over } \mathbb{C}.
\end{array}$$

The last implication is elementary but decisive: a full-column-rank reduction has a maximal minor nonzero modulo p . The same minor is a nonzero integer, hence is nonzero over both $\mathbb{Q}$ and $\mathbb{C}$.

After all blocks vanish at one explicit SNC pair, upper semicontinuity of fiberwise h^0 gives the same vanishing on a nonempty Zariski-open subset of the parameter space of pairs. The finite intersection of these open sets, one for each (m, t) , is again nonempty.

1.3. Completed certificate sets.

pair	cases complete	blocks complete	columns certified	status
(3, 3)	14/14	168/168	280,798	complete
(3, 4)	1/1	1/1	781	complete
(2, 6)	1/1	8/8	1,040	complete
(2, 5)	6/6	36/36	38,286	complete
(1, 8)	6/6	48/48	67,011	complete
(1, 9)	2/2	18/18	5,526	complete

The independent structural checker accepted every displayed set with exit code zero. In particular, the last cubic case $(m, t) = (14, 3)$ contributes twelve full-mode blocks and 76,905 columns; every block has rank equal to its column count and nullity zero.

Remark 1.1 (Profile nomenclature). The mathematical and executable name of the certified range is **smt57**. The certificate filenames and run headers retain **smt38** as a historical exact alias: both names enumerate the same fourteen pairs and the same 168 blocks. The stronger fixed-method frontier is exposed as **smt27**; it has not been computed and supplies no evidence for the accompanying theorem. The intermediate name **smt23** is retained only for reproducibility.

2. EXPLICIT SNC TEST PAIRS

All computations are made at the following integral pairs:

$$(2.1) \quad \begin{array}{c|cc}
(d_1, d_2) & A & B \\
\hline
(3, 3) & X^3 + 2Y^3 + Z^3 & 2X^3 + Y^3 + Z^3 \\
(3, 4) & X^3 + Y^3 + Z^3 & X^4 + Y^4 + Z^4 \\
(2, e), e = 5, 6 & XY + Z^2 & X^e + Y^e + Z^e \\
(1, 8) & Y & X^8 + Y^8 + Z^8 + X^4Y^4 \\
(1, 9) & Y & X^9 + Y^9 + Z^9 + X^3Y^3Z^3.
\end{array}$$

In each row $A + B$ is SNC, and adjoining $L = \{Z = 0\}$ preserves the SNC property.

For the cubic pair, $A = B = 0$ gives $X^3 = Y^3$ and $3X^3 + Z^3 = 0$, so every common point has nonzero coordinates. Proportional gradients would have proportionality factor 1 from the Z -components, but the X -components would then give $3X^2 = 6X^2$, a contradiction. Moreover, $A = B = Z = 0$ has no projective solution, and both cubics meet L transversally.

For the $(3, 4)$ row, proportional gradients at a common point force all nonzero coordinates to have the same value. There are at least two nonzero coordinates, and their equal cubes cannot sum to zero in characteristic zero. On L , a common point would give $(X/Y)^3 = (X/Y)^4 = -1$, also impossible. Smoothness and transversality to L are immediate for the Fermat curves.

Finally, let $A = XY + Z^2$ and $B = X^e + Y^e + Z^e$, $e = 5, 6$. A tangency cannot occur on $Z = 0$. After setting $Z = 1$, one has $XY = -1$. If $\nabla B = \lambda \nabla A$, then the third component gives $\lambda = e/2$, while the first two give

$$2X^{e-1} = Y, \quad 2Y^{e-1} = X.$$

Thus $X^e = Y^e = -1/2$, and consequently

$$(-1)^e = (XY)^e = X^e Y^e = \frac{1}{4},$$

a contradiction. The conic meets L at the two coordinate points, neither of which lies on B , and B meets L transversally.

For the degree-one rows set $A = Y$. For the octic,

$$B_X = 4X^3(2X^4 + Y^4), \quad B_Y = 4Y^3(2Y^4 + X^4), \quad B_Z = 8Z^7.$$

A singular point would have $Z = 0$ and $XY \neq 0$, after which $2X^4 + Y^4 = X^4 + 2Y^4 = 0$; the coefficient determinant is 3, a contradiction. For the nonic, an all-nonzero singular point, with $a = X^3, b = Y^3, c = Z^3$, would satisfy

$$3a^2 + bc = 3b^2 + ac = 3c^2 + ab = 0.$$

Multiplication gives $27 = -1$, again impossible; a zero coordinate is even easier. On $Y = 0$, the intersection equation is $X^e + Z^e = 0$, so both coordinates are nonzero and the intersection is transverse. The same argument applies on L . Finally, $A \cap L = [1 : 0 : 0]$ lies on neither degree- e curve. Thus both degree-one test divisors remain SNC after adjoining L .

The choice $A = Y$ is essential. An earlier experiment with $A = Z$ identified the boundary component with the auxiliary line L , causing both computational charts to miss the entire line. No output from that experiment is included or used as evidence.

3. FINITE AMBIENT NORMAL FORM

Put

$$S = d_1 + d_2, \quad K = \lfloor m/3 \rfloor, \quad n_k = m - 2k, \quad s_k = m - 3k,$$

and

$$(3.1) \quad d_k^+ = (S - 1)m - (2S - 1)k.$$

On the affine chart $Z = 1$, write

$$x = X/Z, \quad y = Y/Z, \quad a = A(x, y, 1), \quad b = B(x, y, 1),$$

and $W_{xy} = x'y'' - x''y'$.

Proposition 3.1 (Ambient presentation used by the program). *Every section contributing to (1.1) has, on $\{ZAB \neq 0\}$, an expression*

$$(3.2) \quad \sum_{k=0}^K \sum_{i+j=s_k} \frac{P_{i,j,k}(x, y)}{a^{n_k} b^{n_k}} (x')^i (y')^j W_{xy}^k,$$

where

$$(3.3) \quad P_{i,j,k} \in \mathbb{C}[x, y], \quad \deg P_{i,j,k} \leq d_k^+.$$

Conversely, a candidate of the form (3.2) represents a global section of the twisted logarithmic bundle exactly when it satisfies the mixed-chart divisibility equations of Section 4.

The exponent n_k is the layerwise logarithmic pole bound. Locally, for a branch $s = 0$, a transverse coordinate v , and $\lambda = s'/s$, one has

$$\lambda v'' - \lambda' v' = \frac{1}{s}(s'v'' - s''v') + \frac{(s')^2 v'}{s^2}.$$

In weighted degree m , the coefficient of Wronskian layer k therefore has pole order at most $m - 2k$. Applying this to both $a = 0$ and $b = 0$ gives the denominators in (3.2). Applying the refined bound to the auxiliary logarithmic component $L = \{Z = 0\}$, using

$$x = z^{-1}, \quad y = uz^{-1}, \quad x' = -z'z^{-2}, \quad y' = (zu' - uz')z^{-2}, \quad W_{xy} = -W_{zu}z^{-3},$$

gives (3.3). The negative twist is not inserted by reducing the degree bound; it is imposed exactly by the divisibility test below.

The number of unsplit ambient variables is

$$(3.4) \quad N_{\text{raw}}^+(m; S) = \sum_{k=0}^{\lfloor m/3 \rfloor} (m - 3k + 1) \binom{(S-1)m - (2S-1)k + 2}{2},$$

with a summand interpreted as zero if the degree bound is negative.

4. MIXED-CHART EQUATIONS AND EXACT DIVISIBILITY

4.1. The logarithmic one-jet frame. On the chart $X = 1$, put

$$u = Y/X, \quad z = Z/X, \quad a_X = A(1, u, z), \quad b_X = B(1, u, z).$$

For diagonal pairs the implementation uses the projectively invariant normalization

$$q = \frac{1}{d_1} d \log a_X - \frac{1}{d_2} d \log b_X.$$

For the mixed conic–Fermat pair it uses the nonzero rational rescaling

$$q = d \log a_X - \frac{2}{e} d \log b_X.$$

This rescaling is chosen so that the transition formulas have integral polynomial numerators. It changes neither the characteristic-zero kernel nor the resulting vanishing statement.

Write

$$(4.1) \quad q = \frac{\Delta u' + E z'}{H}, \quad H = a_X b_X.$$

For the diagonal pairs,

$$\begin{aligned} \Delta &= a_1 u^{d_1-1} b_X - a_X b_1 u^{d_2-1}, \\ E &= a_2 z^{d_1-1} b_X - a_X b_2 z^{d_2-1}, \end{aligned}$$

where $a_X = a_0 + a_1 u^{d_1} + a_2 z^{d_1}$, and similarly for b_X . For $(2, e)$,

$$(4.2) \quad \begin{aligned} a_X &= u + z^2, & b_X &= 1 + u^e + z^e, \\ \Delta &= b_X - 2u^{e-1} a_X, & E &= 2z b_X - 2z^{e-1} a_X. \end{aligned}$$

For the degree-one rows, $a_X = u$, and we use the integral projectively invariant normalization

$$(4.3) \quad q = e d \log a_X - d \log b_X.$$

For $e = 8$,

$$b_X = 1 + u^8 + z^8 + u^4, \quad \Delta = 8 + 8z^8 + 4u^4, \quad E = -8uz^7,$$

whereas for $e = 9$,

$$b_X = 1 + u^9 + z^9 + u^3 z^3, \quad \Delta = 9 + 9z^9 + 6u^3 z^3, \quad E = -9uz^8 - 3u^4 z^2.$$

These are exactly $\Delta = e(a_X)_u b_X - a_X (b_X)_u$ and $E = e(a_X)_z b_X - a_X (b_X)_z$. Multiplying q by a nonzero rational constant changes neither the mixed frame nor the characteristic-zero kernel. Thus

$$(4.4) \quad u' = \frac{H}{\Delta} q - \frac{E}{\Delta} z'.$$

4.2. The triangular Wronskian substitution.

$$A_0 = H/\Delta, \quad B_0 = -E/\Delta, \quad W_X = qz'' - q'z'.$$

Differentiating $u' = A_0q + B_0z'$ and eliminating q' and z'' through W_X gives the exact identity

$$(4.5) \quad \begin{aligned} W_{zu} = & -A_0W_X + A_0(A_0)_uq^2z' \\ & + ((A_0)_uB_0 + (A_0)_z + A_0(B_0)_u)q(z')^2 \\ & + (B_0(B_0)_u + (B_0)_z)(z')^3. \end{aligned}$$

It is triangular in Wronskian degree: a source layer r contributes only to target layers $k \leq r$. The definitive certificates use the complete coupled system, not merely its diagonal layers.

4.3. The polynomial remainder system. After substituting (4.4) and (4.5), collect a candidate in the regular mixed frame:

$$(4.6) \quad \sum_{k=0}^K \sum_{j=0}^{s_k} R_{j,k}(u, z)(z')^{s_k-j} q^j W_X^k.$$

The denominator bounds imply

$$(4.7) \quad \mathcal{N}_{j,k} := z^{n_k} a_X^{n_k} b_X^{n_k} \Delta^{n_k} R_{j,k} \in \mathbb{Q}[u, z].$$

After the fixed rational normalization is cleared, the coefficients lie in $\mathbb{Z}[u, z]$. Regularity along the two components, removal of the auxiliary logarithmic pole at L , and vanishing of order t at L are together equivalent to

$$(4.8) \quad z^{n_k+t} a_X^{n_k} b_X^{n_k} \mid \mathcal{N}_{j,k}.$$

For the first four degree pairs the program computes one remainder in $\mathbb{F}_p[u][z]$, using deterministic long division in z ; the divisor has invertible constant leading z -coefficient in those runs.

For a degree-one pair $a_X = u$, so the required factor is

$$(4.9) \quad z^{n_k+t} u^{n_k} b_X^{n_k}.$$

The factors z, u, b_X, Δ are pairwise coprime. Because (4.9) is not monic in z , the implementation imposes three equivalent families of linear equations: all terms of $\mathcal{N}_{j,k}$ have z -degree at least $n_k + t$, all have u -degree at least n_k , and the remainder modulo the monic polynomial $b_X^{n_k}$ is zero. The three families receive disjoint row labels. For completeness, in degree eight put $v = u^4, w = z^8$ and $\delta_0 = \Delta/4 = 2 + 2w + v$; then $2b_X - \delta_0 = v(2v + 1)$, which immediately excludes a common factor of b_X and Δ . In degree nine, a common factor would divide both b_X and $u(b_X)_u$; after removing u , the only remaining equation is $3u^6 + z^3 = 0$, on which $b_X = 1 - 2u^9 - 27u^{18}$ is not identically zero. The other coprimality conditions are immediate by setting $u = 0$ or $z = 0$. Setting every resulting coefficient to zero produces the columns of $M_{\mathbb{Z}}$ after reduction modulo p . Rows are indexed by

(condition family, target Wronskian layer, q -degree, u -degree, z -degree).

5. SYMMETRY DECOMPOSITION

5.1. Two cubics. For the cubic test pair, $(\mu_3)^2$ acts by

$$[X : Y : Z] \mapsto [\alpha X : \beta Y : Z], \quad \alpha^3 = \beta^3 = 1,$$

and the involution σ exchanges X and Y , hence exchanges A and B . A normal-form monomial

$$x^r y^s (x')^i (y')^j W_{xy}^k$$

has character

$$(4.1) \quad (r + i + k, s + j + k) \pmod{3}.$$

The nine characters form six σ -orbits:

$$00, \quad 11, \quad 22, \quad 01 \leftrightarrow 10, \quad 02 \leftrightarrow 20, \quad 12 \leftrightarrow 21.$$

For each orbit the program constructs the two σ -eigenspaces $\varepsilon = \pm 1$. Since

$$\sigma(W_{xy}^k) = (-1)^k W_{xy}^k,$$

paired monomials have coefficient relation

$$P_{i,j,k}(x, y) = \varepsilon(-1)^k P_{j,i,k}(y, x).$$

This yields exactly twelve blocks per pair (m, t) . Semisimplicity of the finite group action in characteristic zero shows that their direct sum exhausts the candidate space. Thus zero kernel in all twelve blocks is equivalent to zero kernel in the unsplit system.

5.2. The $(2, e)$ pairs. For $e = 5, 6$, the action

$$[X : Y : Z] \mapsto [\zeta X : \zeta^{-1} Y : Z], \quad \zeta^e = 1,$$

preserves $A = XY + Z^2$ and $B = X^e + Y^e + Z^e$. The character of a normal-form monomial is

$$(5.2) \quad r - s + i - j \pmod{e};$$

the Wronskian has character zero. Coordinate exchange maps character c to $-c$ and again multiplies Wronskian layer k by $(-1)^k$. For $e = 5$, the character orbits

$$\{0\}, \quad \{1, 4\}, \quad \{2, 3\}$$

and the two signs give six blocks. For $e = 6$, the orbits

$$\{0\}, \quad \{1, 5\}, \quad \{2, 4\}, \quad \{3\}$$

give eight blocks. The $(3, 4)$ system is small enough to be run as one unsplit block.

5.3. The $(1, e)$ line pairs. For $e = 8, 9$, the same cyclic action preserves the divisor $A + B$, because $A = Y$ is semi-invariant and every monomial of B has character zero. Coordinate exchange does not preserve the distinguished line; hence there is no involution decomposition. Moreover $a = y$ has character -1 , so the factor a^{-n_k} contributes $+n_k$. The correct character is therefore

$$(5.3) \quad r - s + i - j + n_k \pmod{e}, \quad n_k = m - 2k.$$

Thus each $(1, 8)$ case has eight character blocks and each $(1, 9)$ case has nine. The executable verifies both exhaustion of the source monomials and equivariance of every transformed row at $m = 3$. If a cleared row has monomial $u^U z^Z$, q -degree Q , and target layer k , its checked character is

$$-2U - Z + 2(m - 2k) + Q \pmod{e}.$$

This row-level check prevents a formally exhaustive but non-equivariant block split from being mistaken for a proof.

6. EXACT CASE AND SIZE TABLES

6.1. Cubic SMT-57 profile. The following sizes are generated independently by the executable's `--estimate-only` path. The raw column count is (3.4); the largest block is measured after both finite symmetry reductions.

m	$t_0(m)$	raw columns	blocks	largest block
1	1	42	12	6
2	1	198	12	26
3	1	559	12	69
4	1	1,265	12	156
5	2	2,466	12	297
6	2	4,357	12	518
7	2	7,163	12	847
8	2	11,109	12	1,303
9	2	16,511	12	1,924
10	3	23,663	12	2,750
11	3	32,865	12	3,804
12	3	44,570	12	5,141
13	3	59,125	12	6,808
14	3	76,905	12	8,830
total		280,798	168	

All fourteen cases form the prime-5 proof set. A complete prime-7 replay is available for the case $(m, t) = (9, 2)$; this second prime is an implementation check, not an additional logical hypothesis.

6.2. Unequal-degree certificates. Every entry below is a complete full-mode certificate. The displayed time is the largest per-block wall time in that case and is not part of the mathematical proof.

pair	(m, t)	prime	blocks	rows summed	columns	rank	max sec
(3, 4)	(3, 3)	5	1	2,659	781	781	4.002
(2, 6)	(3, 3)	7	8	7,056	1,040	1,040	0.506
(2, 5)	(3, 2)	5	6	5,231	781	781	0.134
(2, 5)	(4, 2)	5	6	12,282	1,781	1,781	1.999
(2, 5)	(5, 2)	5	6	21,569	3,489	3,489	8.149
(2, 5)	(6, 3)	5	6	42,646	6,187	6,187	46.468
(2, 5)	(7, 3)	5	6	70,882	10,199	10,199	242.959
(2, 5)	(8, 3)	5	6	113,331	15,849	15,849	1,170.066
(1, 8)	(3, 2)	5	8	3,623	1,336	1,336	0.009
(1, 8)	(4, 2)	5	8	8,799	3,077	3,077	0.045
(1, 8)	(5, 2)	5	8	17,902	6,066	6,066	0.216
(1, 8)	(6, 3)	5	8	30,627	10,807	10,807	1.005
(1, 8)	(7, 3)	5	8	51,623	17,876	17,876	3.986
(1, 8)	(8, 3)	5	8	81,487	27,849	27,849	14.516
(1, 9)	(3, 3)	5	9	4,527	1,669	1,669	0.020
(1, 9)	(4, 3)	5	9	10,970	3,857	3,857	0.110

The (3, 4) system has also been reproduced at $p = 7$, every (2, 5) case at $p = 7$, and the (2, 6), (1, 8), and (1, 9) systems at $p = 11$. The degree-one totals are 67,011 and 5,526 columns, distributed among 48 and 18 full-rank blocks.

7. UNCOMPUTED CONSTANT FRONTIER AND CLUSTER PLAN

The certified files above prove the constants stated in the manuscript. They do not prove the limiting constants discussed here. The latter require only additional instances of the same map and are included to make the remaining computation quantitatively reproducible.

After twist monotonicity removes redundant values of t , the exact new targets are

$$\begin{aligned} (3, 3) &: (5, 1), (6, 1), (10, 2), (11, 2), (12, 2), (13, 2), (15, 3), \\ &\quad (16, 3), (17, 3), (18, 3), (19, 3), (20, 3); \\ (2, 5) &: (3, 1), (4, 1), (6, 2), (7, 2), (8, 2), (9, 3), (10, 3), (11, 3), (12, 3), (13, 3); \\ (1, 8) &: (3, 1), (4, 1), (6, 2), (7, 2), (8, 2), (9, 2), (10, 3), (11, 3), (12, 3), (13, 3), (14, 3). \end{aligned}$$

Thus the frontier contains 12, 10, and 11 new (m, t) -targets for the three families, respectively. The generator's matrix-free `--estimate-only` path gives:

pair	new targets	raw columns	shards	endpoint raw	largest shard
(3, 3)	12	1,248,727	144	280,686	31,931
(2, 5)	10	288,129	60	84,865	16,973
(1, 8)	11	704,491	88	195,750	24,725
total	33	2,241,347	292	—	—

Here “raw columns” counts coefficient variables before symmetry reduction; the last column is the number B of unknowns in the largest individual finite-field rank problem. The corresponding largest Wronskian layers have 12,257, 8,848, and 12,184 unknowns. A dense $B \times B$ array of 32-bit field entries would occupy about 3.80, 1.07, and 2.28 GiB, before row storage, polynomial generation, fill-in, and elimination workspace. The matrix-free estimator does not construct target rows, so exact row counts, nonzero counts, and peak memory are not yet known. Accordingly, these dense arrays are dimensional baselines and the archive does not assert a minimum hardware configuration for the unrun frontier.

The parallelization is exact and communication-free. A job fixes the degree pair, (m, t) , prime, character orbit, and, when present, involution sign. The options `--orbit`, `--epsilon`, and `--shard-prefix` generate one independently checkable certificate per job; `--list-shards` prints the corresponding commands. Thus an ordinary scheduler may dispatch all 292 primary-prime jobs without sharing an evolving matrix between nodes. The current implementation uses exact modular sparse elimination. Packed sorted rows and blocked dense updates after fill-in are natural node-local optimizations and do not change the proof object. The archive freezes the exact estimate rows and all emitted primary-prime commands in `reports/FRONTIER_ESTIMATES.csv` and `reports/FRONTIER_PRIMARY_PRIME_SHARDS.txt`.

Calibrating only as an engineering estimate against the completed runs gives rough largest-shard times of about 32–39 hours for two cubics, 40–50 hours for conic–quintic, and 1.4 hours for line–octic. These are not measured frontier times and are not mathematical evidence. They support scheduling experiments, but do not replace the missing row/nonzero/fill-in measurements. Full column rank at one good prime proves the characteristic-zero rank claim; selected large shards should nevertheless be repeated at a second prime to detect implementation errors.

8. WHY MODULAR FULL RANK PROVES CHARACTERISTIC-ZERO VANISHING

Lemma 8.1 (Rank lifting). *Let $M \in \text{Mat}_{r \times n}(\mathbb{Z})$. If the reduction $\overline{M}$ has rank n over $\mathbb{F}_p$, then M has rank n over $\mathbb{Q}$.*

Proof. Some $n \times n$ minor of $\overline{M}$ has nonzero determinant. The corresponding determinant of M is an integer not divisible by p , so it is nonzero. Hence $\text{rank}_{\mathbb{Q}} M = n$, and the same minor shows that $\text{rank}_{\mathbb{C}} M = n$. $\square$

The modular calculation is therefore an arithmetic certificate for a fixed integral presentation; it is not an assertion that logarithmic jet bundles have been transported to characteristic p . Rational rescalings used to choose q are first absorbed into the integral presentation over $\mathbb{Q}$. In particular, a proof prime need not be prime to every degree appearing in the geometric notation. What matters is that the program constructs the specified integer matrix and finds a nonzero maximal minor modulo that prime.

Remark 8.2 (A deficient prime is inconclusive). At $p = 5$, the $(2, 6)$ block $c0, -$ has rank 92 for 93 columns. This does not produce a characteristic-zero section; a maximal minor may vanish after reduction even when it is nonzero over $\mathbb{Q}$. The $p = 5$ output is retained as a bad-prime diagnostic only. The proof uses full rank at $p = 7$, independently reproduced at $p = 11$.

Remark 8.3 (The unresolved $(1, 7)$ boundary). The formal Riemann–Roch endpoint for $(1, 7)$ produces 178 cases: $3 \leq m \leq 60$ at twist 1, $61 \leq m \leq 120$ at twist 2, and $121 \leq m \leq 180$ at twist 3. This is not a routine extension of the present laptop certificate: genuine nonvanishing may occur. No $(1, 7)$ result is claimed here. A future large-cluster calculation can test the matrices, but a deficient modular rank would remain inconclusive until an exact characteristic-zero kernel were produced.

9. GENERATOR, CERTIFICATE, AND CHECKER

9.1. Matrix generator. The source

`two_components_finite_vanishing_03.cpp`

is a self-contained C++17 program with no external algebra library. Its principal components have the following mathematical roles:

component	role
Poly	exact sparse polynomials in $\mathbb{F}_p[u, z]$
RationalDelta, RationalRing	exact coefficients in $\mathbb{F}_p[u, z, \Delta^{-1}]$
JetPoly	coefficients in the mixed frame (z', q, W_X)
CurveConfig, Context	test pair, chart data, powers, divisors, and exact first- and second-jet substitutions
generate_basis	cubic ambient basis and the twelve symmetry blocks
generate_cyclic_basis	cyclic/involution basis for $(2, e)$
generate_cyclic_character_basis	shifted cyclic character basis for the degree-one pairs
add_atom_image	denominator clearing and remainder equations (4.8), including the separate degree-one factor tests
sparse_column_rank	deterministic modular column elimination
write_case_json	machine-readable proof metadata and fingerprints
self_test	dimension, symmetry, division, low-weight, and regression checks

In full mode, every column includes the complete triangular coupling between source and target Wronskian layers. Columns are reduced against normalized sparse pivots ordered by the encoded row tuple. The current output schema records for each block:

- the degree pair, m, t , modulus, character block, and, when applicable, involution sign;
- the number of nonzero rows encountered, columns, rank, and nullity;
- the product of unnormalized pivots modulo p ;
- a deterministic input-matrix fingerprint and normalized-echelon fingerprint;
- elapsed wall time, and optionally every pivot-row tuple.

The two fingerprints are deterministic 64-bit FNV-1a audit values. They are useful for detecting an unintended change in monomial order, chart substitution, or elimination. They are not cryptographic hashes and do not replace either the rank computation or the SHA-256 archive manifest.

9.2. Independent structural checker. The separate source

`verify_finite_vanishing_certificates.cpp`

shares no polynomial or elimination code with the generator. It:

- (1) constructs the exact expected list of degree–case–orbit–sign keys;
- (2) rejects degree mismatches and unexpected blocks;
- (3) rejects missing blocks, non-full proof modes, nonpositive column counts, nonzero nullity, or rank smaller than the number of columns;
- (4) requires deterministic fingerprints; every proof-effective cubic record in this release has separate matrix and echelon fingerprints;
- (5) rejects conflicting duplicate certificates at the same prime.
- (6) for degree-one runs, requires the audited program version, the exact $A = Y$ test pair, denominator-character shift n_k , and the `separate-z-a-b` divisibility identifier.

This checker is intentionally an aggregation and structural-audit layer. It does not reconstruct the algebraic matrix. Independent matrix reconstruction is obtained by rebuilding and rerunning the generator from source; the checker then establishes that the resulting collection is complete and internally consistent.

The proof-effective cubic file was regenerated in one complete run from the exact pre-nomenclature source and executable preserved under `provenance/producer_snapshot/`. All 168 blocks use the current schema and carry distinct `matrix_hash` and `echelon_hash` fields together with their pivot rows. The older mixed-schema cubic file is retained only under `provenance/legacy/` as historical run metadata and is not an input to the proof or to the final coverage check. The structural checker remains a coverage and consistency checker; source-bound rank evidence comes from the complete generator rerun, the frozen sources, and the independent reconstruction tests documented in this archive.

9.3. The separate $\mathcal{O}(3)$ identity supplement. The source `03_identity_checker.cpp` is logically separate from the finite-rank calculation. Its tangency part is a deterministic exact-rational regression test: it evaluates the value, gradient, and Hessian identities for one fixed polynomial, point, and tensor package at each of the degrees 2, 3, 5. The universal tangency identities are proved algebraically in the manuscript; this finite sample is not presented as a symbolic proof of them. By contrast, the residual-curvature part checks the commutator identities on a basis of the six-dimensional space of pairs of symmetric 2×2 matrices, so linearity does give the corresponding symbolic identities for arbitrary parameters. The checker’s expected terminal line is

```
03 IDENTITY CHECK PASSED: 3 tangency identities at degrees 2,3,5;
3 curvature commutators on the symbolic six-parameter basis.
```

10. REPRODUCIBILITY

10.1. Recorded workstation. The local runs were made on:

- Microsoft Surface Pro, 13-inch, 12th Edition;
- Snapdragon X2 Elite at 4.03 GHz, 12 physical/logical cores;
- 33,823,641,600 bytes installed memory (approximately 31.5 GiB);
- Windows 11 Home, ARM64, version/build 10.0.28000/28000;
- LLVM-MinGW clang 22.1.8, C++17;
- static *x86_64*-Windows executables run under Windows emulation.

The proof-effective cubic collection was generated in one complete run from the following pre-nomenclature frozen producer:

```
executable SHA-256:
8CE4AE756DAE16D5FAF3E47C54D43DC657B81B9F14B9AAC824EC661DCDFB688A
source SHA-256:
F1E397FFA22FF4A6B4E86D729601AF4B9489B40839C4FA6258A46C840F9BAE9F
```

Both exact producer files are included under `provenance/producer_snapshot/`. It contains all fourteen cases and all 168 current-schema blocks. The current archive rebuild changes only the public profile nomenclature from `smt38` to `smt57`; the bases, matrices, and elimination are

unchanged. The run interval, output hashes, and precise scope of the recorded provenance are given in `reports/CURRENT_SOURCE_CUBIC_RUN_PROVENANCE.md`.

For historical comparison, an earlier (14,3) run had an observed file interval of 1:44:04, a largest block of 8,830 columns, and a largest recorded block time of 2423.014 seconds. Monitoring sampled at most 20,753,010,688 bytes (19.33 GiB) of working set and 22,957,387,776 bytes (21.38 GiB) of paged memory. Its preserved binary and the old mixed-schema logs remain under `provenance/` solely for audit history; neither is used by the present proof file. In the later complete-profile rerun the largest block time was 2919.931 seconds. Times and memory assist reproduction planning and are not proof data.

The new line-component computations are much smaller. The largest (1,8) block has 3,543 columns. At $p = 5$, the 48 recorded block times sum to 139.592 seconds and the largest block time is 14.516 seconds; parallel wall time is lower than this sum. The two (1,9) cases are smaller still. Thus these key-vanishing calculations are genuine laptop computations on the recorded 32-GiB machine.

10.2. Warning-clean builds. The following commands are run from the archive root with LLVM-MinGW `clang++.exe` on PATH. The directory `rebuild` is a disposable output directory; the frozen files in `bin` remain intact.

```
New-Item -ItemType Directory -Force .\rebuild | Out-Null
```

```
clang++.exe -std=c++17 -O3 -DNDEBUG -pthread -static -Wall -Wextra
-Wpedantic -Wconversion -Wshadow -Wsign-conversion
.\source\two_components_finite_vanishing_03.cpp
-o .\rebuild\two_components_finite_vanishing_03.exe
```

```
clang++.exe -std=c++17 -O2 -static -Wall -Wextra -Wpedantic
-Wconversion -Wshadow -Wsign-conversion
.\source\verify_finite_vanishing_certificates.cpp
-o .\rebuild\verify_finite_vanishing_certificates.exe
```

```
clang++.exe -std=c++17 -O2 -static -Wall -Wextra -Wpedantic
-Wconversion -Wshadow -Wsign-conversion
.\source\03_identity_checker.cpp
-o .\rebuild\03_identity_checker.exe
```

In actual PowerShell use, join each wrapped compiler command into one line or append PowerShell continuation characters. The archived binaries were rebuilt directly from the staged sources; every compiler returned exit code zero and emitted no warning. The three self-test primes, the identity check, and all six structural checks passed. Details are in `reports/BUILD_AND_TOOLCHAIN.txt`.

10.3. Internal self-tests.

```
.\bin\two_components_finite_vanishing_03.exe --self-test --prime 5
.\bin\two_components_finite_vanishing_03.exe --self-test --prime 7
.\bin\two_components_finite_vanishing_03.exe --self-test --prime 11
.\bin\03_identity_checker.exe
```

The finite-rank self-test verifies selected raw counts, exhaustion of the cubic and cyclic symmetry decompositions, exact polynomial remainder, low-weight end-to-end vanishings, unsplit basis counts, the two degree-one mixed-frame fingerprints, transformed-row character equivariance, and a fixed prime-5 full-matrix/echelon regression fingerprint. Passing the self-test is necessary for a clean reproduction but does not replace the full proof runs.

10.4. Definitive proof runs. Two cubics, all fourteen cases: the historical alias in this command reproduces the archived run header exactly; replacing it by `smt57` gives the same matrices and ranks.

```
.\bin\two_components_finite_vanishing_03.exe --degrees 3 3 --profile smt38
--all --prime 5 --jobs 4 --mode full --full-certificate
--certificate certificates\proof\cubic_smt38_p5.jsonl
--report certificates\proof\cubic_smt38_p5.txt
```

Unequal-degree pairs:

```
.\bin\two_components_finite_vanishing_03.exe --degrees 3 4 --all --prime 5
--mode full --certificate certificates\proof\pair_34_p5.jsonl
--report certificates\proof\pair_34_p5.txt
```

```
.\bin\two_components_finite_vanishing_03.exe --degrees 2 6 --all --prime 7
--jobs 8 --mode full --certificate certificates\proof\pair_26_p7.jsonl
--report certificates\proof\pair_26_p7.txt
```

```
.\bin\two_components_finite_vanishing_03.exe --degrees 2 5 --all --prime 5
--jobs 6 --mode full --certificate certificates\proof\pair_25_p5.jsonl
--report certificates\proof\pair_25_p5.txt
```

```
.\bin\two_components_finite_vanishing_03.exe --degrees 1 8 --all --prime 5
--jobs 6 --mode full --full-certificate
--certificate certificates\proof\pair_18_p5.jsonl
--report certificates\proof\pair_18_p5.txt
```

```
.\bin\two_components_finite_vanishing_03.exe --degrees 1 9 --all --prime 5
--jobs 6 --mode full --full-certificate
--certificate certificates\proof\pair_19_p5.jsonl
--report certificates\proof\pair_19_p5.txt
```

The release keeps the original five conic–quintic cases and the added $(8, 3)$ case in separate files, so the provenance of the new computation remains visible. To reproduce only the added case, use

```
.\bin\two_components_finite_vanishing_03.exe --degrees 2 5 --m 8 --t 3
--prime 5 --jobs 6 --mode full
--certificate certificates\proof\pair_25_m8_t3_p5.jsonl
--report certificates\proof\pair_25_m8_t3_p5.txt
```

For cluster execution, first enumerate the exact jobs:

```
.\bin\two_components_finite_vanishing_03.exe --degrees 3 3 --profile smt57
--all --prime 5 --mode full --list-shards
```

Each emitted command fixes (m, t) , character orbit, sign, and prime. To produce deterministically named JSONL and text files, add the option

`--shard-prefix.`

A failed or interrupted shard can then be rerun without disturbing completed blocks.

10.5. Structural aggregation.

```
.\bin\verify_finite_vanishing_certificates.exe --degrees 3 3
--profile smt57 certificates\proof\cubic_smt38_p5.jsonl
```

```
.\bin\verify_finite_vanishing_certificates.exe --degrees 3 4
certificates\proof\pair_34_p5.jsonl
```

```
.\bin\verify_finite_vanishing_certificates.exe --degrees 2 6
certificates\proof\pair_26_p7.jsonl
```

```
.\bin\verify_finite_vanishing_certificates.exe --degrees 2 5
certificates\proof\pair_25_p5.jsonl
certificates\proof\pair_25_m8_t3_p5.jsonl
certificates\replays\pair_25_p7.jsonl
certificates\replays\pair_25_m8_t3_p7.jsonl
```

```
.\bin\verify_finite_vanishing_certificates.exe --degrees 1 8
certificates\proof\pair_18_p5.jsonl
certificates\replays\pair_18_p11.jsonl
```

```
.\bin\verify_finite_vanishing_certificates.exe --degrees 1 9
certificates\proof\pair_19_p5.jsonl
certificates\replays\pair_19_p11.jsonl
```

The recorded first line is

```
CERTIFICATE SET VERIFIED: 168 required full-rank block(s),
deterministic fingerprints present.
```

The other expected block counts are 1, 8, 36, 48, and 18, respectively. The checker accepts multiple JSONL files, so the same command works with sharded certificates. It deliberately rejects exploratory `diagonal-triangular` records; public proof certificates use `--mode full`.

10.6. SHA-256 verification.

```
Get-ChildItem -Recurse -File |
  Get-FileHash -Algorithm SHA256 |
  Sort-Object Path
```

The resulting list must agree byte-for-byte with the final manifest. A same-prime rerun should also reproduce the deterministic matrix and echelon fingerprints. Across different primes, row counts and fingerprints may differ because an integer coefficient can become zero modulo one prime. Column counts and full ranks must agree.

11. FROZEN SNAPSHOT AND ARCHIVE MANIFEST

11.1. Audited source snapshot. The following hashes identify the frozen source and executable snapshot in the release archive:

file	SHA-256
two_components_finite_vanishing_03.cpp	CC7A1825ADCC2226C882F78F3F239DD3 C026EF25C9D32E0F1F8CC2BAF27766D0
two_components_finite_vanishing_03.exe	23CD4C43235207E0733BDECDB3B1A484 48D510AA08FAEF637333F9477859D03F
verify_finite_vanishing_certificates.cpp	629BBBF94E27A5BB5B318D8DCA07A6A0 E946C936C22003E4ABE6CBDDD94E5295
verify_finite_vanishing_certificates.exe	763DD899CC5542DD2601597902297DEF FC1E52ABBAC3374CB0F8988398E98AD
03_identity_checker.cpp	F63B15D9F86839474FB18E7B1F4C1685 95AE9D6CC0E56C141AD9681DC445B648
03_identity_checker.exe	F2A95CB0F2E12BB1C2352BAC6BABA3A4 C7EB95ACC841BEB86B750C23C8262F2

11.2. Proof-file hashes. The principal proof files have the following final hashes. The complete release-relative list, including text reports, replay data, documentation, and this guide, is `SHA256SUMS.txt`.

file	SHA-256
cubic_smt38_p5.jsonl	BC69BF57240A6E49C68DF8BDCB146CD25D2C4F9DFEFCF200 FF6D9BC71C303085
cubic_smt38_p5.txt	5D407B869DDC56DEFD2EFA56FEBFAE1FBAC6E8DBD7CE3B1E 6DF9B1F8D8B55546
pair_34_p5.jsonl	51A2E753D8702ABF43792C2A7AE30D2AE02FAC943F514AC0 DFA33DC98B149756
pair_34_p5.txt	6AEE27564BEDD5DD0DC19B1D684DC2D7B921F1872B5CC862 EEE0B570AFD62D1A
pair_26_p7.jsonl	7536A3EF5D3D70E4FC48014BE7C5D86AEA12D3C2CA667DF4 E4C216C06CC8EFC2
pair_26_p7.txt	B6A8A6A5BC794637DA4F1347254392DB233C44243051AC34 FCAB689FE33898A9
pair_25_p5.jsonl	EC29B7135F077F3CE8C50D586C166361901FAF967E5CBFF7 4E4488308B6B7E09
pair_25_p5.txt	F5D0A98B16C04DF5BA6A57A3A9345D7CF1670EDBBEFCBAC14 C8F751C59603DC95
pair_25_m8_t3_p5.jsonl	D344FC9E55B990D9A7E03C1FA70832AC4503AE12E08F8EC9 E8ACD34EF08A7C0F
pair_25_m8_t3_p5.txt	1CD6730AC5D0E383305524DB7231706D52AA1C4762D410A6 830ABFF0EA01F336
pair_18_p5.jsonl	4CDB406132A608FE24E5F3DC685E4052F05E908DA4E04147 B76714E1A4EE0300
pair_18_p5.txt	88C11BD0AD6814E860ADDB4DD6D4BD89725D770ACE427548 AE6767D231D395A2
pair_19_p5.jsonl	37066F31909D44D2DA39DF846F6E82D62DFA329295ED77B6 6FFF079820BBCCE8
pair_19_p5.txt	540F13A0ABDC9B15D173631429789FEABBE5258EA22A6CDB 208C8D16077040CA

The manifest lists every payload file except itself: a manifest cannot contain its own stable checksum. Its SHA-256 is therefore recorded in the external release audit. If a ZIP is made after review, its independent hash must likewise be recorded without altering this staged tree.

12. FINALIZATION CHECKLIST

Before upload, the release process checked all of the following:

- ✓ the complete source-bound prime-5 cubic run from the preserved pre-nomenclature producer finished with exit code zero, and all 168 modern-schema blocks have rank equal to columns, nullity zero, separate matrix/echelon fingerprints, and pivot rows;
- ✓ the cubic certificate set contains exactly the expected fourteen cases and no obsolete (9, 3) record in place of the required (9, 2) record;
- ✓ the independent structural checker printed the expected accepted count for all six degree pairs;
- ✓ the conic–quintic proof and replay each contain six full-rank blocks for the added case (8, 3), increasing that certificate set to (36) blocks and (38, 286) columns;
- ✓ the proof primes are 5 for (3, 3), (3, 4), (2, 5), (1, 8), (1, 9) and 7 for (2, 6); the deficient (2, 6), $p = 5$ record is labelled diagnostic;
- ✓ the degree-one proof files identify the audited $A = Y$ pairs, and the staged tree contains no obsolete $A = Z$ certificate;
- ✓ warning-clean release builds and the three self-test primes passed;
- ✓ the exact $\mathcal{O}(3)$ identity checker printed its expected success line;
- ✓ every staged payload appears in the SHA-256 manifest, and every manifest hash has been independently rechecked;

- ✓ a clean staging copy has exactly the manifested file set, and the six structural checks and three self-tests replay there with exit code zero.

13. CREDIT AND RELATION TO PRECEDING COMPUTATIONS

The one-component normal form and finite key-vanishing philosophy developed by Hou–Huynh–Merker–Xie [1] provide an important starting point. The symmetry-block and modular-C++ organization follows the strategy developed by Cui–Hou–Liu–Xie [2]. The present certificate is not a replay of either calculation: two residue directions lead to a coupled overlap system, unequal degrees require projectively balanced logarithmic forms, and the $(2, e)$ models require a different cyclic symmetry. The generator, divisibility presentation, block lists, and certificates documented here belong to the two-component argument.

REFERENCES

- [1] L. Hou, D. T. Huynh, J. Merker, and S.-Y. Xie, *Vanishing of invariant 2-jet differentials and improved hyperbolicity degree bounds in dimension two*, arXiv:2603.14881, 2026.
- [2] T. Cui, L. Hou, J. Liu, and S.-Y. Xie, *A symmetry method for key vanishing lemmas and optimal 2-jet thresholds for hyperbolicity*, arXiv:2606.19155, 2026.