

# An Explicit Nullstellensatz Certificate for the Seventh Hadamard Ideal

Alper Ferudun\*

19 October 2026

## Abstract

The nonexistence of a Hadamard matrix of order 28 with one circulant core is classical. A related AIM workshop challenge asks for an explicit polynomial combination of the 41 generators of the seventh Hadamard ideal that equals one. We give such a combination over the rationals, in factored finite-sum form, with total degree at most 29. Compression from length 27 to length 9 leaves 36 compressed rows and 78732 Boolean lifts; each lift violates an original correlation equation. Boolean projectors and ordered polynomial differences convert this finite computation into explicit multipliers for the original generators. The certificate uses 340840 leaf identities and is accompanied by integer enumeration, exact symbolic checks and a second bit-mask replay. The contribution is the reproducible certificate, not a new nonexistence theorem or a claim of optimal certificate degree.

## 1 The specific certificate problem

The AIM Computational Algebraic Statistics workshop lists a challenge of Kotsireas: establish inconsistency of a 41-equation system in 27 variables and explicitly express one as a polynomial combination of its generators [1]. The challenge follows, but is separate from, Onn’s question on cell spectra of contingency tables. It concerns the seventh Hadamard ideal of Kotsireas, Koukouvinos and Seberry [2]. Their Section 4.3.2 records a known nonexistence theorem and an exhaustive search. Neither that theorem nor the use of finite enumeration is claimed as new here. The AIM page also records Sturmfels’s suggestion to assemble certificates for pieces of the system.

We state the target ideal explicitly. Indices on original variables are taken modulo 27, and

$$q_i = a_i^2 - 1, \quad L = \sum_{i=0}^{26} a_i - 1, \quad s_d = 1 + \sum_{i=0}^{26} a_i a_{i+d} \quad (1 \leq d \leq 13). \quad (1)$$

Thus

$$H_7 = \langle L, q_0, \dots, q_{26}, s_1, \dots, s_{13} \rangle \subseteq \mathbb{Q}[a_0, \dots, a_{26}].$$

This is Definition 1 of [2], with indices shifted by one. The positive row-sum normalization is part of that definition. The three legacy data-file links in the workshop statement were not retrievable with valid HTTPS certificate checking; our statement uses the published definition, without claiming a byte-for-byte comparison with those files.

**Theorem 1.** *The rational polynomials  $A_L, A_d, Q_i$  defined in Section 4 satisfy*

$$1 = A_L L + \sum_{d=1}^{13} A_d s_d + \sum_{i=0}^{26} Q_i q_i. \quad (2)$$

*Every summand has degree at most 29. The specified representation uses 262108 compressed leaf identities and 78732 full-sign leaf identities. In particular,  $H_7 = (1)$  over  $\mathbb{Q}$  and over  $\mathbb{C}$ .*

---

\*Mercury Software GmbH. alper@mercurycodelab.com; <https://github.com/AlperTheKing>.

All coefficient formulas below are finite sums with explicit indexing sets. Every division is by a fixed nonzero integer, or is a divided difference explicitly interpreted as a polynomial. No unspecified ideal-membership computation is part of the definition.

## 2 Compression with explicit generator relations

For  $0 \leq r < 9$ , put  $B_r = a_r + a_{r+9} + a_{r+18}$ . In nine formal variables  $X$ , with indices modulo 9, define

$$\begin{aligned} g_0(X) &= \sum_r X_r - 1, & g_1(X) &= \sum_r X_r^2 - 25, \\ g_{d+1}(X) &= 3 + \sum_r X_r X_{r+d} & (1 \leq d \leq 4). \end{aligned}$$

Direct polynomial expansion gives

$$\begin{aligned} g_0(B) &= L, & g_1(B) &= \sum_i q_i + 2s_9, \\ g_2(B) &= s_1 + s_8 + s_{10}, & g_3(B) &= s_2 + s_7 + s_{11}, \\ g_4(B) &= s_3 + s_6 + s_{12}, & g_5(B) &= s_4 + s_5 + s_{13}. \end{aligned} \tag{3}$$

For the energy identity,  $\sum_r B_r^2 = \sum_i a_i^2 + 2 \sum_i a_i a_{i+9}$ . For the correlation identities, expand the products blockwise: the compressed shift  $d$  is the sum of the original shifts  $d, d+9, d+18$ . The original correlations at opposite shifts agree. These are literal identities in the polynomial ring; no equation from (1) has been imposed to establish them.

Let  $T = \{-3, -1, 1, 3\}$  and define the finite set

$$\mathcal{C} = \{b \in T^9 : g_j(b) = 0 \text{ for every } 0 \leq j \leq 5\}.$$

For  $b \notin \mathcal{C}$ , let  $j(b)$  be the least  $j$  with  $g_j(b) \neq 0$ . For  $b \in \mathcal{C}$ , put

$$\mathcal{S}_b = \{\sigma \in \{-1, 1\}^{27} : \sigma_r + \sigma_{r+9} + \sigma_{r+18} = b_r \ (0 \leq r < 9)\}, \quad \mathcal{S} = \bigcup_{b \in \mathcal{C}} \mathcal{S}_b.$$

**Lemma 2.** *The set  $\mathcal{C}$  consists of the nine cyclic rotations of each of*

$$\begin{aligned} &(-3, -1, 1, 1, -1, 1, 3, -1, 1), \\ &(-3, -1, 1, 3, -1, 1, 1, -1, 1), \\ &(-3, 1, -1, 1, 1, -1, 3, 1, -1), \\ &(-3, 1, -1, 3, 1, -1, 1, 1, -1). \end{aligned} \tag{4}$$

*These 36 rows are distinct. Each  $\mathcal{S}_b$  has  $3^7 = 2187$  elements. Every  $\sigma \in \mathcal{S}$  has  $s_d(\sigma) \neq 0$  for some  $1 \leq d \leq 6$ .*

*Finite verification.* Appendix A gives a complete integer-only check. It first visits all  $4^9$  rows, tests all six compressed equations, and compares the survivors with the rotations of (4). A surviving row has one entry 3, one entry  $-3$  and seven entries of absolute value one. The saturated blocks have one sign triple each; every other block has three. The second loop therefore visits every one of the  $36 \cdot 3^7 = 78732$  lifts, without a symmetry reduction, and tests the original equations. It raises an error if a lift survives. The first violated equation has the following histogram:

| $d$        | 1     | 2     | 3    | 4    | 5   | 6    |
|------------|-------|-------|------|------|-----|------|
| $\#\sigma$ | 50220 | 18252 | 7236 | 2052 | 324 | 648. |

The supplied second implementation reconstructs the compressed rows by choosing the two saturated positions. For each lift it uses a 27-bit mask  $m$  of its negative entries and the exact formula

$$s_d(\sigma) = 28 - 2 \text{ wt}(m \text{ xor } \text{rot}_d(m)).$$

It reproduces the entire first-failure histogram, including residual values. These are exhaustive finite verifications, not random tests.  $\square$

For the remainder define  $d(\sigma)$  as the least shift in  $\{1, \dots, 13\}$  with  $s_{d(\sigma)}(\sigma) \neq 0$ . The lemma ensures that this index exists and is at most six. It also proves that all scalar divisors used below are nonzero. The finite verification is part of the certificate, not a search whose failure to find a point is interpreted as a proof.

### 3 Boolean projectors and polynomial differences

For  $x = (x_0, x_1, x_2)$  and  $\varepsilon \in \{-1, 1\}^3$ , set

$$e_\varepsilon(x) = \prod_{k=0}^2 \frac{1 + \varepsilon_k x_k}{2}.$$

For  $t \in T$  define

$$\begin{aligned} P_t(x) &= \sum_{\varepsilon_0 + \varepsilon_1 + \varepsilon_2 = t} e_\varepsilon(x), \\ R_{t,k}(x) &= \sum_{\varepsilon_0 + \varepsilon_1 + \varepsilon_2 = t} \frac{\varepsilon_k}{2} \prod_{h \neq k} \frac{1 + \varepsilon_h x_h}{2}. \end{aligned} \tag{5}$$

**Lemma 3.** *As polynomial identities over  $\mathbb{Q}$ ,*

$$\sum_{t \in T} P_t(x) = 1, \quad (x_0 + x_1 + x_2 - t)P_t(x) = \sum_{k=0}^2 (x_k^2 - 1)R_{t,k}(x). \tag{6}$$

*Proof.* The first equality is the product of the three identities  $(1 + x_k)/2 + (1 - x_k)/2 = 1$ . For the second, in each term of (5) use  $\sum_k \varepsilon_k = t$  and

$$(x_k - \varepsilon_k) \frac{1 + \varepsilon_k x_k}{2} = \frac{\varepsilon_k}{2} (x_k^2 - 1).$$

$\square$

Write  $x_r = (a_r, a_{r+9}, a_{r+18})$  and set

$$P_b = \prod_{r=0}^8 P_{b_r}(x_r), \quad E_\sigma = \prod_{i=0}^{26} \frac{1 + \sigma_i a_i}{2}.$$

Then the following partition holds in the polynomial ring:

$$1 = \sum_{b \in T^9} P_b, \quad P_b = \sum_{\sigma \in S_b} E_\sigma \quad (b \in \mathcal{C}). \tag{7}$$

The second identity is simply the blockwise expansion of the product.

For a polynomial  $f$  in  $n$  variables and a constant vector  $z$ , define the ordered divided difference

$$D_i f(X; z) = \frac{f(z_0, \dots, z_{i-1}, X_i, \dots, X_{n-1}) - f(z_0, \dots, z_i, X_{i+1}, \dots, X_{n-1})}{X_i - z_i}. \tag{8}$$

The numerator is divisible by  $X_i - z_i$  in the polynomial ring; (8) denotes this quotient polynomial, including at  $X_i = z_i$ . Telescoping yields

$$f(X) - f(z) = \sum_{i=0}^{n-1} (X_i - z_i) D_i f(X; z). \tag{9}$$

For our quadratic polynomials,  $D_i f$  has degree at most one. For complete computational specificity: a monomial  $X_i X_h$  with  $i < h$  contributes  $X_h$  to  $D_i$  and  $z_i$  to  $D_h$ ;  $X_i^2$  contributes  $X_i + z_i$  to  $D_i$ , and  $X_i$  contributes one.

## 4 The explicit multipliers

For  $b \in T^9 \setminus \mathcal{C}$  put  $j = j(b)$ ,  $v_b = g_j(b)$ , and

$$\begin{aligned} W_b &= \frac{P_b}{v_b}, \\ V_{b,r+9k} &= \frac{D_r g_j(B; b)}{v_b} R_{b_r, k}(x_r) \prod_{u \neq r} P_{b_u}(x_u) \quad (0 \leq r < 9, 0 \leq k < 3). \end{aligned} \tag{10}$$

Multiply (9) by  $P_b$  and apply (6) in each block. Rearranging gives

$$P_b = W_b g_{j(b)}(B) - \sum_{i=0}^{26} V_{b,i} q_i. \tag{11}$$

For  $\sigma \in \mathcal{S}$  put  $d = d(\sigma)$ ,  $w_\sigma = s_d(\sigma)$ , and define

$$\begin{aligned} W_\sigma &= \frac{E_\sigma}{w_\sigma}, \\ V_{\sigma,i} &= \frac{D_i s_d(a; \sigma)}{w_\sigma} \frac{\sigma_i}{2} \prod_{h \neq i} \frac{1 + \sigma_h a_h}{2}. \end{aligned} \tag{12}$$

Now (9) and the one-variable identity in Lemma 3 give

$$E_\sigma = W_\sigma s_d(\sigma) - \sum_{i=0}^{26} V_{\sigma,i} q_i. \tag{13}$$

Let  $\alpha_{j,d}$  be the integer coefficient of  $s_d$  in (3). Explicitly,  $\alpha_{1,9} = 2$ ; for  $j = 2, 3, 4, 5$  the nonzero entries are one on, respectively,

$$\{1, 8, 10\}, \quad \{2, 7, 11\}, \quad \{3, 6, 12\}, \quad \{4, 5, 13\}.$$

All other entries are zero. The 41 multipliers in Theorem 1 are

$$\begin{aligned} A_L &= \sum_{\substack{b \notin \mathcal{C} \\ j(b)=0}} W_b, \\ A_d &= \sum_{b \notin \mathcal{C}} \alpha_{j(b),d} W_b + \sum_{\substack{\sigma \in \mathcal{S} \\ d(\sigma)=d}} W_\sigma \quad (1 \leq d \leq 13), \\ Q_i &= \sum_{\substack{b \notin \mathcal{C} \\ j(b)=1}} W_b - \sum_{b \notin \mathcal{C}} V_{b,i} - \sum_{\sigma \in \mathcal{S}} V_{\sigma,i} \quad (0 \leq i < 27). \end{aligned} \tag{14}$$

Here every sum over  $b$  is restricted to  $T^9$ . Equations (5), (8), (10), (12) and (14), together with the finite indexing sets, explicitly specify the multipliers. Expanding their many monomials is not required to interpret the identity.

*Proof of Theorem 1.* Sum (11) over  $b \notin \mathcal{C}$  and (13) over  $\sigma \in \mathcal{S}$ . The left sides sum to one by (7). Substituting (3) and collecting the coefficients of the original generators yields exactly (14), proving (2).

The degrees of  $P_b, E_\sigma$  are at most 27. For a coarse correction, the degrees of  $R_{b_r, k}$ , the eight other block projectors and  $D_r g_j$  are at most 2, 24, 1, respectively. After multiplication by  $q_i$  the degree is at most 29. A fine correction has degree at most  $1 + 26 + 2 = 29$ . The  $W$  terms times their equations also have degree at most 29. Finally  $4^9 - 36 = 262108$ , which, with the 78732 fine leaves, gives the stated count.  $\square$

## 5 Verification and interpretation

The accompanying source package includes three standard-library Python programs. The enumeration program records the compressed survivors, the integer divisor histograms and a deterministic leaf-stream digest. The symbolic program checks 32 polynomial identities over the rationals: the block projectors, their first moments, the six compressed relations, and the generic telescoping identities. The bit-mask program supplies a separate implementation of the finite lift test. Normal and optimized Python runs agree; no correctness condition relies on an `assert` statement that optimization could remove.

The proof combines the finite lemma with the algebraic identities, rather than treating numerical substitution at sample points as a polynomial-identity proof. This is self-audited computer-assisted work, not a proof-assistant formalization or external independent review. The local two-program comparison should not be interpreted as such a review. AI assistance was used in developing and checking the certificate, implementing the verifiers and preparing the manuscript; responsibility for the submitted text rests with the author.

Boolean interpolation, polynomial telescoping and sequence compression are classical. The certificate is provided as an explicit reproducible answer for the stated ideal, without a claim of absolute priority, minimal degree, minimum size, or a new general Nullstellensatz method. It does not settle the other workshop questions concatenated with this challenge in the dataset record.

## A A complete finite check

The following executable check uses only the Python standard library. The supplementary programs also record full residual histograms and perform the symbolic and bit-mask checks discussed above.

```
from itertools import product
from collections import Counter
T = (-3, -1, 1, 3)
def g(b):
    return (sum(b)-1, sum(x*x for x in b)-25) + tuple(
        3+sum(b[r]*b[(r+d)%9] for r in range(9))
        for d in range(1,5))
C = [b for b in product(T, repeat=9) if not any(g(b))]
seeds = [
    (-3,-1,1,1,-1,1,3,-1,1),
    (-3,-1,1,3,-1,1,1,-1,1),
    (-3,1,-1,1,1,-1,3,1,-1),
    (-3,1,-1,3,1,-1,1,1,-1)]
rotations = {b[r:]+b[:r] for b in seeds for r in range(9)}
if len(C) != 36 or set(C) != rotations:
    raise ValueError("compressed enumeration")
triples = {t: [s for s in product((-1,1), repeat=3)
               if sum(s) == t] for t in T}
hist = Counter()
for b in C:
    for blocks in product(*(triples[t] for t in b)):
        a = [0]*27
        for r, block in enumerate(blocks):
            for k in range(3):
                a[r+9*k] = block[k]
        for d in range(1,14):
            if 1+sum(a[i]*a[(i+d)%27] for i in range(27)):
                hist[d] += 1
                break
    else:
        raise ValueError("surviving common zero")
```

```

expected = {1:50220,2:18252,3:7236,4:2052,5:324,6:648}
if dict(hist) != expected or sum(hist.values()) != 78732:
    raise ValueError("lift histogram")
print("PASS", len(C), sum(hist.values()))

```

## References

- [1] American Institute of Mathematics, *Computational Algebraic Statistics: Open Problems*, Kotsireas challenge following Question 38, <https://aimath.org/WWN/compalgstat/articles/html/2a/>. Workshop PDF: <https://aimath.org/WWN/compalgstat/compalgstat.pdf>.
- [2] I. S. Kotsireas, C. Koukouvinos and J. Seberry, Hadamard ideals and Hadamard matrices with circulant core, *J. Combin. Math. Combin. Comput.* **57** (2006), 47–63. Author-hosted version: <https://documents.uow.edu.au/~jennie/WEB/WEB06/kks1cc.pdf>.