THE OBDD SPLIT EXPONENT $s_\star(n)$ OF A RESTRICTED INTEGER MULTIPLICATION: EXACT VALUES, OPTIMAL VARIABLE ORDERINGS, AND THE FAILURE OF THE OBVIOUS CLOSED FORM

KOREA SUPERINTELLIGENCE LABS

Abstract. Qin studies the middle bit of the product of two n-bit numbers restricted to multipliers of Hamming weight two, and shows that the width of an ordered binary decision diagram for that function is $\Theta(2^{s_\star(n)})$, where $s_\star(n) = \min_\pi \max_{(L,R)} \max_{a<b} |\mathrm{Split}(L, R; a, b)|$ is a purely combinatorial minimax — over orderings π of the x-variables, over the partitions (L, R) induced by π, and over pairs of bit positions. That paper proves $n/8 \le s_\star(n) \le n/2$, evaluates $s_\star(n)$ at no n at all, and leaves both the exact values and a combinatorial description of a minimising ordering as explicit open questions. We determine $s_\star(n)$ exactly for every $2 \le n \le 26$, exhibit for each such n one ordering certified optimal, and read off the first seven first-occurrence thresholds: $\min\{n : s_\star(n) = K\} = 2, 5, 8, 13, 17, 22, 26$ for $K = 1, \ldots, 7$. What makes this possible is an observation absent from the source: a partition with respect to π is a prefix of π, so $s_\star(n)$ is a minimum-bottleneck maximal chain in the Boolean lattice — 2^n sets rather than $n!$ orderings — together with a level-and-depth blocking certificate that refutes every ordering at once. The values kill one closed form and suggest another. The formula $\lfloor\sqrt{2n}\rfloor - 1$ agrees with $s_\star$ at every n from 2 to 16, and its first-occurrence thresholds $2, 5, 8, 13$ are the opening terms of an OEIS sequence; it is false at $n = 17$. The formula $\lfloor(2n+11)/9\rfloor$ agrees at every n in $[2, 26]$ except $n = 4$; it is *linear*, and its slope $2/9$ lies strictly inside the source's own interval $[1/8, 1/2]$. It is a fit, and is stated as one — but a fit that has since made a falsifiable prediction and survived it: an unverified search, reported separately from everything proved, gives $s_\star(n)$ up to $n = 36$ and confirms the predicted threshold $\min\{n : s_\star(n) = 9\} = 35$, so that the formula agrees at 34 of the 35 values $2 \le n \le 36$. Every *theorem* about an individual n rests on exhaustive computation and is machine-checked in Lean 4; the values past $n = 26$ are measurements, and are labelled as such wherever they appear.

1. Introduction

1.1. **Ordered binary decision diagrams and the middle bit of a product.** An ordered binary decision diagram (OBDD) for a Boolean function of N variables fixes one ordering of those variables and branches on each of them at most once, in that order; Bryant [2] introduced the model and proved that the reduced OBDD for a fixed ordering is a canonical form. Its *width* is the largest number of nodes that carry a single variable, so an OBDD of width w has size at most Nw; and both the size and the width depend on the ordering, sometimes catastrophically, which is why the minimising ordering is the object of interest. (Definition 2.1 of [1] measures the width at a variable instead by the number of edges that start at a node before it and end at it or later, and that is the convention its width theorems use. Nothing below depends on either convention: no decision diagram is built anywhere in this paper.) A numbered statement attributed to [1] anywhere below — "Definition 2.1 of [1]", "its Theorem 3.3" — always carries the number of the v1 e-print and never one of this paper's own; the bibliography entry lists them.

Integer multiplication is the classical hard function for the model. Write $\mathrm{Mul}^n_{n-1}(x, y)$ for the middle bit — bit $n-1$ — of the product of the n-bit numbers with bit vectors x and y. Bryant [3] proved that every OBDD for Mul^n_{n-1} has $\Omega(1.09^n)$ vertices. What the proof supplies is the equivalent exponential form that later work, [1] included, quotes instead: for every balanced partition of the variables there is a fooling set of size at least $2^{n/8}$, and $2^{1/8} > 1.09$. Woelfel [4] sharpened the lower bound to $2^{\lfloor n/2\rfloor}/61 - 4$ and gave the first non-trivial upper bound $\frac{7}{3} \cdot 2^{4n/3}$. The gap between $2^{n/2}$ and $2^{4n/3}$ is still open, and

work since then (see [5] and the references there) has largely attacked neighbouring bits rather than closed it.

1.2. **A restricted multiplication, and its split exponent.** Qin [1] isolates a restriction of the middle bit for which the width *can* be described exactly. Let SMul^n_{n-1} agree with Mul^n_{n-1} when the multiplier y has Hamming weight exactly 2 and be 0 otherwise (Definition 3.2 of [1]). Setting $y_{n-a-1} = y_{n-b-1} = 1$ for a pair $a < b$ in $\{0, \dots, n-1\}$ and all other y_i to 0 leaves a function of x alone, and equation (3) of [1] evaluates it: with $h = b - a$ and $H = \{h, \dots, b-1\}$, the value is $x_a \oplus x_b \oplus x_m$ if $x_i = x_{i-h}$ for some $i \in H$, where m is the largest such i, and $x_a \oplus x_b$ otherwise. So whether an OBDD can carry the computation across a cut of the variable order depends on how many residue classes modulo h straddle the cut, and that count is what [1] names.

Definition 1.1 ([1, Definition 3.3])**.** Let $0 \le a < b \le n-1$, put $h = b - a$ and $k(j,p) = b - p - jh$. For a partition (L, R) of the variables $x_0, \dots, x_{n-1}$,

$$\mathrm{Split}(L,R;a,b) = \Big\{\, p \in \{0,\dots,h-1\} \;:\; \exists\, j \in \big\{0,\dots,\lfloor (b-p)/h \rfloor\big\} \text{ such that } x_{k(j,p)} \text{ and } x_{k(j+1,p)} \text{ lie on opposite sides of } (L,R) \,\Big\}.$$

A *partition with respect to an ordering* π is, in the terminology of [1] (the paragraph following its Definition 2.1), obtained from a variable index i by taking L to be the set of variables that come before x_i under π and R the rest. The quantity the present paper is about is then

$$s_\star(n) \;=\; \min_{\pi} \max_{(L,R)} \max_{a<b} \big|\mathrm{Split}(L,R;a,b)\big|, \tag{1}$$

π ranging over all orderings of the x-variables, (L,R) over all partitions with respect to π, and $a < b$ over $\{0,\dots,n-1\}$; this is equation (4) of [1]. The point of (1) is that it is the answer: the abstract of [1] states that "the width of such OBDDs is $\Theta(2^{s_\star(n)})$", and its two width theorems give the sandwich explicitly. Its Theorem 3.3 says that under every variable ordering some x-variable has width at least $2^{s_\star(n)}$, and its Theorem 3.5 that under one particular ordering every variable has width at most $8n \cdot 2^{s_\star(n)}$. (The introduction of [1] also announces the size bounds $2 \cdot 2^{s_\star(n)} - 1 \le |\mathcal{B}(\mathrm{SMul}^n_{n-1})| \le n \cdot 2^{s_\star(n)}$. Those are what its Corollary 2.3 delivers from a width sandwich with both constants equal to 1; its Theorem 3.3 does give the lower constant 1, but its Theorem 3.5 gives $8n$ rather than a constant, and its Corollary 2.3 then yields $8n^2 \cdot 2^{s_\star(n)}$. We quote the width statement, which is the one its theorems establish; nothing below uses either.)

What [1] then proves about $s_\star$ itself is one two-sided estimate,

$$\frac{n}{8} \;\le\; s_\star(n) \;\le\; \frac{n}{2}, \tag{2}$$

its Lemma 4.3, whose lower half it takes, through its own Lemma 4.2, from the fooling-set lemma of [3]; and it evaluates $s_\star(n)$ at no n whatsoever. Two questions are left explicitly open there. Of the minimising ordering, written π_{wopt}, the paper says, in its §3.1 just after the equation reproduced above as (1),

> "we cannot give a combinatorial characterization to π_{wopt} in this paper, and this is an important question left open"

and its conclusion, §5, says

> "we established only asymptotic bounds on $s_\star(n)$ rather than an exact closed-form formulation, leaving its precise determination for future research".

Both quotations are verbatim from the v1 e-print. It is the second of these that the present paper attacks, and the first that it feeds with data.

1.3. **Results.** Everything below concerns (1) as a combinatorial quantity; no OBDD is constructed anywhere. Write $[n] = \{0, 1, \dots, n-1\}$.

Theorem 1.2 (Theorem 4.1)**.** *For every n with $2 \le n \le 22$, $s_\star(n)$ is the entry of Table 1. In particular $s_\star$ first attains the value 5 at $n = 17$ and the value 6 at $n = 22$.*

[1] evaluates $s_\star$ at no n, and we have found no evaluation of it anywhere else either; §1.4 says exactly what that search was.

Theorem 1.3 (Theorem 5.1)**.** *For every n with $2 \le n \le 22$ the ordering listed in Table 3 attains the minimum in* (1)*; that is, its value equals $s_\star(n)$.*

Table 3 prints, besides the orderings, the induced value profile of each: the sequence of the n numbers $\max_{a<b} |\mathrm{Split}|$ over the n partitions with respect to that ordering. No pattern is claimed and none is conjectured; the table is data for whoever attacks the characterisation question of [1].

The engine behind both is a reduction, absent from [1], which replaces the minimisation over $n!$ orderings by a search over 2^n sets, together with a certificate that refutes all orderings at once.

Theorem 1.4 (Theorem 3.2)**.** *The partitions with respect to π are exactly the n proper prefixes of π, and $|\mathrm{Split}(L, R; a, b)|$ depends on the partition only through the set L; hence $s_\star(n)$ is a minimum-bottleneck maximal chain in the Boolean lattice on $[n]$. Moreover, for all K, m, d with $m + d < n$, if every m-element subset of $[n]$ is d-blocked at threshold K in the sense of Definition 3.1, then $K \le s_\star(n)$.*

At $n = 22$ this is the difference between $22! = 1.12 \cdot 10^{21}$ orderings and $2^{22} = 4\,194\,304$ sets. Finally, the values decide the two closed forms that the data suggest.

Theorem 1.5 (Theorem 6.1)**.** $s_\star(n) = \lfloor\sqrt{2n}\rfloor - 1$ *for every n with $2 \le n \le 16$, and this is* false *at $n = 17$, where $s_\star(17) = 5$ and the formula gives* 4*. Nor is $s_\star(n) = \lceil n/4 \rceil$: that already fails at $n = 8$.*

The refutation matters because the fit is seductive: the first-occurrence thresholds $\min\{n : s_\star(n) = K\}$ for $K = 1, 2, 3, 4$ are $2, 5, 8, 13$, which are $\lceil m^2/2 \rceil$ for $m = 2, 3, 4, 5$ — the sequence A000982 of the OEIS [6] — and that fit forces $\lfloor\sqrt{2n}\rfloor - 1$ on $2 \le n \le 16$. Stopping the computation at $n = 16$ would have produced a confident false conjecture with an apparent confirmation from a published sequence. The thresholds in fact continue $17, 22$ and not $18, 25$.

Theorem 1.6 (Theorem 6.2)**.** $s_\star(n) = \lfloor(2n + 11)/9\rfloor$ *for every n with $2 \le n \le 22$ except $n = 4$, where $s_\star(4) = 1$ and the formula gives* 2*. Equivalently, the first-occurrence thresholds satisfy $\min\{n : s_\star(n) = K\} = \lceil(9K - 11)/2\rceil$ for $3 \le K \le 6$.*

This is a *fit*, and §6 says exactly how good a fit: eight first-occurrence thresholds known, seven of them reproduced, one exceptional n, and a proved range of twenty-one values. It is recorded because it is linear. If $s_\star(n) \sim \frac{2}{9}n$, then the width of SMul^n_{n-1} is $2^{(2/9+o(1))n}$, and $2/9 = 0.2222\ldots$ sits strictly inside the interval $[\frac{1}{8}, \frac{1}{2}]$ that (2) allows.

The range of Theorems 1.2 and 1.3 has since been pushed four values further, which is what brings the derived object of the source's question — the first-occurrence thresholds — within reach.

Theorem 1.7 (Theorems 4.5 and 5.2)**.** $s_\star(23) = s_\star(24) = s_\star(25) = 6$ *and* $s_\star(26) = 7$*; and for each of these four n the ordering listed in Table 4 attains the minimum in* (1)*.*

With Theorem 1.2 this determines $s_\star$ on the whole of $2 \le n \le 26$, and the closed form of Theorem 1.6 extends with it. Writing $T(K) = \min\{n \ge 2 : s_\star(n) = K\}$ for the first-occurrence thresholds:

Theorem 1.8 (Theorems 6.6 and 6.7)**.** $s_\star(n) = \lfloor(2n+11)/9\rfloor$ *for every n with $2 \le n \le 26$ except $n = 4$, where $s_\star(4) = 1$ and the formula gives* 2*. Consequently $T(K)$ exists and equals $2, 5, 8, 13, 17, 22, 26$ for $K = 1, \ldots, 7$, in the strong sense that $s_\star(T(K)) = K$ while $s_\star(n) < K$ for every n with $2 \le n < T(K)$.*

$T(7) = 26$ is the first threshold outside the range of Theorem 1.2. Theorem 1.8 is as close as this paper comes to the "exact closed-form formulation" that [1] asks for: an exact formula, on a finite range, with one exceptional argument.

Beyond $n = 26$ the same search, run outside the proof assistant and reported as measurement throughout, now reaches $n = 36$: $s_\star(n) = 8$ for $n = 32, 33, 34$ and $s_\star(n) = 9$ for $n = 35, 36$, so that $T(8) = 31$ and $T(9) = 35$ (Remark 6.8). The value $T(9) = 35$ is the prediction the fit of Theorem 1.6 had made and that had not yet been tested: it was recorded, before the computation, as the first untested consequence of the formula, and it is correct. This does not turn the fit into a theorem and we do not present it as one; what it changes is the weight of the evidence, and §6 says exactly how much.

1.4. **What is not claimed.** Five limits, stated here rather than left to be inferred.

First, no asymptotic statement is proved. The values below are a finite table. Nothing here improves either half of (2), and nothing here proves that $s_\star$ is monotone in n, or linear, or anything else outside the range that has been computed. Theorems 1.6 and 6.6 state one and the same fit to a finite table, and are stated as a fit; a proof, in either direction, would be a genuinely new theorem about SMul. The confirmed prediction $T(9) = 35$ is evidence and not a proof: a different law agreeing with $s_\star$ at every $n \le 36$ and parting from it later is ruled out by nothing here.

Second, the OBDD side is untouched. This paper works with (1) as a combinatorial minimax. The theorems of [1] that connect it to the width of an OBDD for SMul^n_{n-1} are quoted, never used and never re-proved; no decision diagram appears in the formal development, and no claim is made about Mul on the unrestricted domain, where the bounds of [3, 4] remain the state of the art and do not transfer. Remark 7.2 records, as arithmetic on the source's own statements and as nothing more, what its width and size bounds say once the constants in them are substituted; no error is asserted there and nothing here depends on either bound.

Third, the orderings are optima, not canonical optima. Tables 3 and 4 list, at each n, one ordering returned by a minimum-bottleneck search; at most n there are many, and the two tables were produced by two different searches. They settle no part of the characterisation question of [1].

Fourth, every value rests on exhaustive computation, and the values past $n = 26$ are not proved at all. Each entry of Table 1 and each value of Theorem 4.5 is an equality proved in two halves: an upper bound by evaluating one explicit ordering, and a lower bound by a certificate that examines all 2^n subsets of $[n]$. These are finite checks, and §8 says exactly how large each is. Only Theorem 1.4 and Proposition 3.3 are free of them. The values for $27 \le n \le 36$, and the thresholds $T(8)$ and $T(9)$ that depend on them, are *measurements*: they come from a program that reproduces every proved value exactly, but they carry no machine-checked proof, and every statement of this paper that uses them is labelled where it appears.

Fifth, on provenance. The abstract page of [1] was fetched live on 3 September 2026: the preprint is still at v1, single-authored, 11 pages, primary category cs.CC with a cs.DS cross-list. The quantity (1) is introduced there and, as far as we can determine, nowhere else: the preprint has no citing work in either of the two citation indices we queried, no accompanying software artifact could be found, no sequence in the OEIS *is* $s_\star$ or its threshold sequence — neither on the range of Table 1 nor on the extended lists of Table 5, and the numerical coincidences that a search does turn up are all recorded in Remark 6.5 — and a full-text scan of a large local preprint corpus for OBDD work on integer multiplication found nothing that could contain it. The searches are recorded in §8. We therefore report the values below as *not found* elsewhere, which is a weaker and more honest claim than new; what is certain is that [1] itself computes none of them.

1.5. **Changes from version 1.** This is a revision of an earlier version of the same paper. Every statement of that version stands, with its number and its wording; nothing is retracted. What is new:

(1) the values $s_\star(23), \dots, s_\star(26)$ and four optimal orderings (Theorems 4.5 and 5.2), with the negative controls that go with them (Proposition 4.6);
(2) the whole proved range in one closed form (Theorem 6.6) and the first seven first-occurrence thresholds (Theorem 6.7), of which $T(7) = 26$ is the first outside the earlier range;
(3) a second implementation of the certificate predicate, provably the same function as the first (Proposition 3.3), which is what makes $n = 26$ practical to check;
(4) the measured range extended from $n \le 31$ to $n \le 36$, giving $T(9) = 35$ (Remark 6.8) — the prediction that the earlier version's Remark 6.4 named as the first untested consequence of the fit;
(5) a measured price for the next step and the reason it is an algorithmic and not a budgetary wall (Remark 7.3), and the arithmetic behind one parenthesis of §1 written out in full (Remark 7.2).

Two estimates made in the earlier version, or alongside it, were wrong and are corrected here rather than quietly dropped. *(a)* The cost of deciding the $n = 25$ certificate by interpretation was estimated at about an hour; measured, it is about nine minutes (544 seconds of processor time), so the estimate was

pessimistic by a factor of six or seven. What actually needs the compiled evaluator of Proposition 3.3 is $n = 26$, whose certificate is seven times the size of the $n = 25$ one. *(b)* Remark 6.3 attributes the cost of a threshold-crossing n in the unverified search to the failing pass, "because a threshold crossing must exhaust the failing pass before the succeeding one". That is the wrong half, and Remark 6.9 gives the measurement and the consequence: the expensive half is the *succeeding* pass, and it is avoidable.

2. The objects

Fix $n \geq 2$ and write $[n] = \{0, 1, \ldots, n-1\}$ for the index set of the variables $x_0, \ldots, x_{n-1}$. (At $n = 1$ there is no pair $a < b$, so $s_\star(1) = 0$ trivially and nothing below is about it.)

Definition 2.1. An *ordering* is a bijection $\pi\colon [n] \to [n]$, read as: x_i occupies position $\pi(i)$. Its *prefixes* are the sets $P_t(\pi) = \{i \in [n] : \pi(i) < t\}$ for $t = 0, 1, \ldots, n$.

A partition of the variables is recorded by the set $L \subseteq [n]$ of indices on the left; R is its complement in $[n]$, and indices outside $[n]$ belong to neither side. Restated in this notation, Definition 1.1 reads: for $0 \leq a < b \leq n-1$ and $h = b - a$,

$$\text{(3)}\quad \mathrm{Split}(L; a, b) = \big\{\, p \in \{0, \ldots, h-1\} \;:\; \exists j \geq 0,\ k(j{+}1, p) \geq 0 \text{ and } \big[\, k(j,p) \in L \not\Leftrightarrow k(j{+}1,p) \in L \,\big] \,\big\},$$

with $k(j, p) = b - p - jh$ as before. Two points of transcription, both recorded because a reader comparing with [1] will meet them.

Remark 2.2 (transcription). (i) The display of Definition 3.3 is typographically broken in the v1 e-print of [1]: the brace opened before $p \in \{0, \ldots, h-1\}$ is never closed after $\lfloor (b-p)/h \rfloor$, and a pair of stray dollar signs, one on either side of a `\text` box, interrupt the display's math mode. It is repaired above in the only way that parses. (ii) The stated range $j \in \{0, \ldots, \lfloor (b-p)/h \rfloor\}$ contains one value of j, namely the largest, for which $k(j+1, p) = b - p - (j+1)h < 0$; there is no variable $x_{k(j+1,p)}$, so that value of j can never witness a split. Formula (3) therefore enumerates $0 \leq j < \lfloor (b-p)/h \rfloor$, which is equivalent. We note that the prose following Definition 3.3 of [1] — the indices $k(j,p)$ and $k(j+1,p)$ are "nothing but i and $i-h$ for some $i \in H$", with $H = \{h, \ldots, b-1\}$ — suggests a second, strictly smaller reading in which the pair (x_b, x_a) is also dropped; that reading is excluded by the paper's own proof of its Claim 2 (Appendix A), which treats the case in which the correcting index equals b and opens "Since $m' = b$, x_a and x_b must be in the different partition". Both readings were computed for $2 \leq n \leq 12$; the strict one gives $s_\star(n-1)$ there, a shift of one in n, so nothing structural depends on the choice.

Definition 2.3. For $L \subseteq [n]$ put

$$g_n(L) = \max_{0 \leq a < b \leq n-1} \big|\mathrm{Split}(L; a, b)\big|, \qquad \hat{g}_n(L) = \max_{1 \leq h \leq n-1} \big|\mathrm{Split}(L; n-1-h,\, n-1)\big|,$$

so that $\hat{g}_n \leq g_n$ by construction: $\hat{g}_n$ inspects only the $n-1$ pairs with $b = n-1$, one per difference h. For an ordering π put $\mathrm{val}_n(\pi) = \max_{0 \leq t \leq n-1} g_n\big(P_t(\pi)\big)$.

Lemma 2.4. *The partitions with respect to π, in the sense of* [1]*, are exactly the sets $P_0(\pi)$, $P_1(\pi)$, …, $P_{n-1}(\pi)$; consequently $s_\star(n) = \min_\pi \mathrm{val}_n(\pi)$, the minimum being over all $n!$ orderings.*

Proof. The partition with respect to π determined by the index i has left part $\{j : \pi(j) < \pi(i)\} = P_{\pi(i)}(\pi)$, and $\pi(i)$ runs over $0, \ldots, n-1$ as i does. Substituting into (1) gives the display. □

This is the identification made once, at the point where (1) is written down in the notation of Definition 2.3; every statement below is about $\min_\pi \mathrm{val}_n(\pi)$.

Remark 2.5 (what Split counts; outside the formal development). The following reformulation is elementary, is not part of the machine-checked development, and is recorded because it is the quickest way to see what (3) measures. For fixed p the indices $k(0,p), k(1,p), \ldots$ run through $b{-}p,\ b{-}p{-}h,\ b{-}p{-}2h, \ldots$ down to the least non-negative one — that is, through the residue class $C(p) = \{i \in [0,b] : i \equiv b - p \pmod h\}$ — and as p runs over $\{0, \ldots, h-1\}$ these classes partition $\{0, \ldots, b\}$. A finite chain whose

members are two-coloured contains two consecutive members of different colours if and only if it contains members of both colours, so

$$|\mathrm{Split}(L;a,b)| \;=\; \#\big\{\text{residue classes mod } h \text{ inside } [0,b] \text{ meeting both } L \text{ and } [n]\setminus L\big\}.$$

Three consequences follow at once, none of them used below. First $|\mathrm{Split}| \le \min(h,|L|,|R|) \le \lfloor n/2\rfloor$, since the classes are disjoint and each bichromatic one consumes an index from each side — which is the upper half of (2), with a sharper minimum. Second, enlarging b only adds elements to classes, so the count is monotone in b and the outer maximum in (1) is attained at $b = n-1$; that is, $\hat g_n = g_n$, of which only the inequality $\hat g_n \le g_n$ is used below. This equality was checked exhaustively over all 2^n subsets for every $n \le 24$, with no mismatch. Third, writing $\tau = \pi$ for the position map, each class C spans a time interval $[\min\tau(C), \max\tau(C))$ and is bichromatic at the prefix of length t exactly when t lies in that interval; so $\mathrm{val}_n(\pi) \le K$ says that for every h the h intervals have depth at most K everywhere, simultaneously in h. We record the last as a possible handle on the characterisation question of [1].

3. From orderings to chains in the Boolean lattice

The minimisation in (1) is over $n!$ orderings. By Lemma 2.4 and Definition 2.3 it is really over the maximal chains of the Boolean lattice on $[n]$: an ordering contributes only its chain of prefixes $\emptyset = P_0 \subset P_1 \subset \cdots \subset P_n = [n]$, every maximal chain arises from an ordering, and val_n depends on the chain alone. Thus

$$s_\star(n) \;=\; \min_{\text{maximal chains}} \; \max_{1\le t\le n-1} g_n(L_t), \tag{4}$$

a minimum-bottleneck maximal chain, over 2^n sets rather than $n!$ orderings. For an upper bound one chain suffices. A lower bound has to exclude all of them, and the device used here is local: it looks at one level of the lattice and at a bounded number of levels above it.

Definition 3.1. Fix n and a threshold K. A set $L \subseteq [n]$ is 0-*blocked* if $\hat g_n(L) \ge K$, and $(d+1)$-*blocked* if either $\hat g_n(L) \ge K$ or $L \cup \{x\}$ is d-blocked for every $x \in [n]\setminus L$. The *level certificate* $\mathrm{cert}(n,K,m,d)$ asserts that every $L \subseteq [n]$ with $|L| = m$ is d-blocked.

Theorem 3.2. *Let $n \ge 1$ and let K, m, d satisfy $m + d < n$. If $\mathrm{cert}(n,K,m,d)$ holds, then $K \le s_\star(n)$.*

Proof. Let π be any ordering and $P_0 \subset \cdots \subset P_n$ its chain of prefixes. Then $|P_m| = m$, so P_m is d-blocked. We claim that for every d and every m with $m + d < n$, if P_m is d-blocked then $\hat g_n(P_{m+j}) \ge K$ for some $j \le d$; this is an induction on d. At $d = 0$ it is the definition. For $d+1$: if $\hat g_n(P_m) \ge K$ take $j = 0$; otherwise $m < n$, so the index x occupying position m under π lies in $[n]\setminus P_m$ and $P_{m+1} = P_m \cup \{x\}$ is d-blocked, and the inductive hypothesis applies at $m+1$, which still satisfies $(m+1)+d < n$. Now $m + j \le m + d < n$, so P_{m+j} is one of the n prefixes appearing in $\mathrm{val}_n(\pi)$, and $\mathrm{val}_n(\pi) \ge g_n(P_{m+j}) \ge \hat g_n(P_{m+j}) \ge K$. As π was arbitrary, Lemma 2.4 gives $K \le s_\star(n)$. □

The hypothesis $m + d < n$ is not cosmetic. At the top of the lattice, $L = [n]$, the universal quantifier in Definition 3.1 is empty, so $[n]$ is d-blocked for every $d \ge 1$ and every K, however large; Proposition 4.4(iii) exhibits this at a threshold that is false. Without $m + d < n$ the certificate would prove anything.

Note also that Definition 3.1 uses the cheap $\hat g_n$, which examines $n-1$ pairs (a,b) instead of $\binom{n}{2}$. Since $\hat g_n \le g_n$ by construction, a lower bound obtained this way is a lower bound for the true quantity; the equality $\hat g_n = g_n$ of Remark 2.5 is a convenience, not a dependency.

The lower bounds of §4 beyond $n = 22$ are decided by evaluating cert through a second implementation of Definition 3.1, written so that it can be compiled to machine code rather than interpreted: the maximum defining $\hat g_n$ is computed by tail recursion rather than as a supremum over a finite index set, a partition is extended by a bit shift rather than by a power of two, and the test over all subsets of $[n]$ is one loop rather than a fold over a predicate. Write cert' for it.

Proposition 3.3. *$\mathrm{cert}'(n,K,m,d) = \mathrm{cert}(n,K,m,d)$ for all n, K, m, d, and likewise at every intermediate stage: the two implementations of* Split*, of $\hat g_n$, of d-blockedness and of the test over subsets agree*

pointwise. The proofs are inductions — on the range of h, on the depth d, and on the number of subsets still to test — and evaluate nothing.

So a lower bound obtained by evaluating cert′ is a lower bound obtained by evaluating the cert that Theorem 3.2 consumes, and nothing new is assumed: the same finite object is decided by a second program. What is bought is a measured factor of about 20 in evaluation time. The figures are processor time above a start-up baseline of under two seconds: at $n = 23$ the certificate costs 9.7 seconds through the compiled route against 182 to 195 seconds interpreted, over three runs whose spread is machine load and in which the compiled figure reproduced to the tenth of a second; at $n = 25$ it is 25 against 544. That factor is what decides whether $n = 26$ can be checked at all: its certificate is the largest in this paper, 3 631 560 evaluations of $\hat{g}_n$, seven times the $n = 25$ count, and it takes about two minutes this way against an extrapolated forty minutes by interpretation.

4. The exact values

Theorem 4.1. *For $2 \le n \le 22$ the value of $s_\star(n)$ is*

$$s_\star(n) = 1, 1, 1, 2, 2, 2, 3, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, 6 \qquad (n = 2, 3, \ldots, 22),$$

as tabulated in Table 1. In particular $s_\star$ first attains 5 at $n = 17$ and 6 at $n = 22$.

n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
$s_\star(n)$	1	1	1	2	2	2	3	3	3	3	3	4	4	4	4	5	5	5	5	5	6

Table 1. $s_\star(n)$ for $2 \le n \le 22$ (Theorem 4.1). Every entry is an equality proved in both directions; see §8 for the size of each check.

Proof sketch, and what was computed. Each entry is proved twice. *Upper bound.* The ordering listed at that n in Table 3 is a permutation of $[n]$, and evaluating val_n at it — the maximum of g_n over its n prefixes, each of which is a maximum over the $\binom{n}{2}$ pairs (a, b) of a scan of at most h residues — returns the tabulated value. By Lemma 2.4 this bounds $s_\star(n)$ above. *Lower bound.* Theorem 3.2 is applied at the parameters (K, m, d) of Table 2, which satisfy $m+d < n$; the certificate $\mathrm{cert}(n, K, m, d)$ is decided by walking all 2^n subsets of $[n]$, testing each for cardinality m and, when it has cardinality m, running the bounded depth-d recursion of Definition 3.1. Both halves are exhaustive finite computations, evaluated by compiled reduction inside the proof assistant; §8 gives their cost. At $n = 22$, the largest case, the walk covers all $2^{22} = 4\,194\,304$ subsets and the split count is evaluated 335 790 times.

The two ingredients that are *searched for* rather than verified — the witness ordering and the parameters (m, d) — were found outside the proof assistant, by a minimum-bottleneck dynamic program over the Boolean lattice (a frontier breadth-first search) and by a sweep over (m, d) minimising the number of evaluations of $\hat{g}_n$. Neither search is trusted: what is verified is the object each returns.

Before any of this, the transcription of Definition 1.1 was cross-checked against three independent implementations — a direct search over all $n!$ orderings using a literal transcription of the definition, for $2 \le n \le 10$; the lattice search itself; and a third reimplementation in another language used on the witness orderings and on sampled values of g_n — all of which agree with each other and with the formal definitions on every value in Table 1 and on the sampled cells of Proposition 4.4(v). □

Remark 4.2 (when depth is necessary; outside the formal development). A certificate of depth $d = 0$ is the plain argument "every partition at one particular level is already bad". Over the range of Table 2 this is impossible exactly at $n \in \{5, 8, 9, 13, 17, 22\}$ — at those n every level m carries an m-set with $\hat{g}_n < K$, and the best level falls exactly one short, $\max_m \min_{|L|=m} \hat{g}_n(L) = K - 1$ — so positive depth is genuinely necessary there; at the other n a positive depth was chosen because it was cheaper, not because $d = 0$ fails. This is an observation about the search, not part of the verified development.

The next two propositions are not results; they are the checks that the table means what it says.

n	2	3	4	5	6	7	8	9	10	11	12
K	1	1	1	2	2	2	3	3	3	3	3
m	1	1	1	2	2	2	3	5	3	3	3
d	0	0	0	1	0	0	2	1	1	1	1
evaluations	–	–	–	–	–	–	170	142	148	213	238

n	13	14	15	16	17	18	19	20	21	22
K	4	4	4	4	5	5	5	5	5	6
m	6	4	4	4	6	6	6	5	5	8
d	1	2	2	1	5	3	2	2	2	4
evaluations	1842	2231	2437	2564	25008	25124	31204	35032	38817	335790

TABLE 2. The certificate parameters used in the proof of Theorem 4.1, and the number of evaluations of $\hat{g}_n$ each costs (counted by the external prototype; a dash means the case is small enough that no count was recorded). All satisfy $m + d < n$.

Proposition 4.3 (the source's own bounds, as a control)**.** *For every n with $2 \le n \le 22$ the bounds* (2) *of* [1] *hold: $n \le 8s_\star(n)$ and $2s_\star(n) \le n$. Moreover the upper bound is never tight for $n \ge 3$: $2s_\star(n) < n$ for every n with $3 \le n \le 22$.*

The first half is a control on the transcription: a table contradicting (2) would mean that Definition 1.1 had been copied wrongly. It holds with room — at $n = 22$ it reads $2.75 \le 6 \le 11$. The second half is an immediate corollary of the table.

Proposition 4.4 (negative controls)**.**

(i) (not too large) $s_\star(20) \ge 6$ *is false.*

(ii) (not too small) $s_\star(20) \le 4$ *is false.*

(iii) (the certificate is a real check, and $m + d < n$ is load-bearing) $\mathrm{cert}(20, 6, 5, 2)$ *is false — at the very parameters that certify the true threshold $K = 5$ — while the full set* $[20]$ *is* 1*-blocked at the same false threshold $K = 6$, vacuously, because it has no proper extension. So the hypothesis $m + d < n$ of Theorem 3.2 cannot be dropped.*

(iv) (the ordering hypothesis is load-bearing) val_{20} *evaluated at the empty list is* 0*, so a minimisation that forgot to require a genuine permutation would return* 0 *at every n.*

(v) (data) $g_8(L) = 1, 2, 3, 4, 3$ *at* $L = \{0\}$, $\{0, 1\}$, $\{0, 2, 4, 5, 7\}$, $\{4, 5, 6, 7\}$, $\{1, 3, 5, 7\}$ *respectively, and* $g_8(\emptyset) = g_8([8]) = 0$*; and* $|\mathrm{Split}(\{4, 5, 6, 7\}; 3, 7)| = 4$ *while* $|\mathrm{Split}(\{4, 5, 6, 7\}; 0, 1)| = 0$.

(vi) (the minimisation carries the content) *The natural ordering $x_0, x_1, \ldots, x_{n-1}$ has* $\mathrm{val}_n = n/2$ *at $n = 16$ and at $n = 20$ — exactly the upper bound of* (2)*, and twice $s_\star(n)$.*

Item (vi) is the informative one. The bound $s_\star(n) \le n/2$ of [1] is attained by the *worst* ordering, not by the best: an interval prefix $\{0, \ldots, t-1\}$ makes every residue class modulo h bichromatic for every $h \le n - t$, so its value is $\min(t, n - t)$ and the maximum over t is $\lfloor n/2 \rfloor$. All of the content of $s_\star$ is therefore in the minimisation over orderings, which is what makes the search of §3 necessary.

The same two halves reach four values further, once the certificate is evaluated through the second implementation of Proposition 3.3.

Theorem 4.5. *$s_\star(23) = s_\star(24) = s_\star(25) = 6$ and $s_\star(26) = 7$. Together with Theorem 4.1 this determines $s_\star(n)$ for every n with $2 \le n \le 26$; in particular $s_\star$ first attains* 7 *at $n = 26$.*

Proof sketch, and what was computed. Exactly as for Theorem 4.1. *Upper bound.* Evaluating val_n at the ordering listed for that n in Table 4 — a permutation of $[n]$ — returns $6, 6, 6, 7$ respectively. *Lower bound.* Theorem 3.2 is applied at the parameters $(K, m, d) = (6, 7, 4)$, $(6, 7, 3)$, $(6, 7, 2)$ and $(7, 9, 6)$ for $n = 23, 24, 25, 26$, all of which satisfy $m + d < n$; deciding $\mathrm{cert}(n, K, m, d)$ walks all 2^n subsets of $[n]$ and evaluates $\hat{g}_n$ altogether 355 251, 417 602, 525 292 and 3 631 560 times. As before the parameters were found outside the proof assistant, by a sweep minimising that count, and are not trusted; what is verified is the certificate they name.

Each of the four lower bounds also has a second proof, outside the proof assistant and by a different algorithm. A breadth-first search through the live set $\{L \subseteq [n] : \hat{g}_n(L) < K\}$, started at $\emptyset$, exhausts it

without ever reaching $[n]$ at $(n, K) = (23, 6)$, $(24, 6)$, $(25, 6)$ and $(26, 7)$; since the prefixes of an ordering of value $< K$ would form such a path, this is again $s_\star(n) \geq K$. It shares no code, and no data structure, with the certificate. The upper bounds likewise have a second, independent check: each of the four orderings was re-evaluated against Definition 1.1 taken literally, in a third implementation and with all $\binom{n}{2}$ pairs (a, b) rather than the $n - 1$ pairs with $b = n - 1$. □

Proposition 4.6 (negative controls for the extended range)**.** (i) (the certificate is a real check) $\mathrm{cert}'(25, 7, 7, 2)$ *and* $\mathrm{cert}'(26, 8, 9, 6)$ *are both false — each at the very* (m, d) *that certifies the true threshold at that* n*, and each therefore equal, by Proposition 3.3, to a false value of* cert*.*

(ii) (not too large, not too small) $s_\star(25) \geq 7$ *is false, and* $s_\star(25) \leq 5$ *is false.*

(iii) (the two implementations agree where they overlap) $\mathrm{cert}'(22, 6, 8, 4)$ *is true: the certificate of Theorem 4.1 at its largest* n*, re-decided by the compiled implementation.*

(iv) (the minimisation carries the content) *the natural ordering* $x_0, x_1, \ldots, x_{23}$ *has* $\mathrm{val}_{24} = 12 = 24/2$*, the upper bound of* (2) *and twice* $s_\star(24)$*.*

(v) (the source's own bounds) (2) *holds at every* n *with* $2 \leq n \leq 26$*:* $n \leq 8s_\star(n)$ *and* $2s_\star(n) \leq n$*.*

(vi) (monotonicity, as far as it is proved) $s_\star(22) \leq s_\star(23) \leq s_\star(24) \leq s_\star(25) \leq s_\star(26)$*.*

(vii) (a refuted formula agreeing again) $s_\star(25) = \lfloor\sqrt{50}\rfloor - 1 = 6$*.*

Two of these are informative rather than routine. Item (i) is what makes the compiled implementation a check and not a tautology: the same program, at the same level and depth, returns false one threshold higher, so a certificate that returned true for every input would have been caught. Item (vii) is a caution about how to read Theorem 6.1: the formula refuted there at $n = 17$ is correct again at $n = 25$, and a formula that fails once is refuted for good — the agreement at $n = 25$ is a coincidence of a step function crossing another, not a rehabilitation.

5. Optimal orderings

Theorem 5.1. *For each* n *with* $2 \leq n \leq 22$*, the ordering* π *listed in Table 3 satisfies* $\mathrm{val}_n(\pi) = s_\star(n)$*, and is therefore optimal in* (1)*.*

Proof sketch. Evaluating val_n at the listed permutation — an exhaustive computation over its n prefixes and, for each, over all pairs $a < b$ — returns the value in Table 1. Combined with Theorem 4.1, which gives the matching lower bound, this is optimality. The orderings themselves were produced by the minimum-bottleneck dynamic program of §4; the program reconstructs one optimal chain, and at most n there are others. □

Two things are visible in Table 3 and worth recording, both as descriptions of these particular optima rather than as claims about all of them. Every profile starts at 0 and ends at 1, and every one except $n = 2$ — where there is no room — attains its maximum $s_\star(n)$ strictly inside; and the profiles are unimodal at every listed n except $n = 14$ and $n = 20$, where the profile dips by one before climbing again — so an optimal chain need not increase the split count monotonically up to its bottleneck. And the large-index variables tend to come first, but not uniformly: at $n = 12$, for instance, the ordering interleaves. We repeat that no pattern is claimed; Table 3 is offered as the first data on the question [1] raises.

Table 1 and Table 3 extend together.

Theorem 5.2. *For each* n *with* $23 \leq n \leq 26$*, the ordering* π *listed in Table 4 satisfies* $\mathrm{val}_n(\pi) = s_\star(n)$*, and is therefore optimal in* (1)*.*

Proof sketch. As for Theorem 5.1: evaluating val_n at the listed permutation returns the value of Theorem 4.5, whose lower half is the matching bound. These four orderings come from a different search from the one behind Table 3 — a depth-first descent through the lattice with the children of a set ordered by increasing $\hat{g}_n$ and the sets it has failed on remembered — and each was re-checked against Definition 1.1 taken literally, as described in the proof of Theorem 4.5. □

Two features of Table 4 are worth recording, again as descriptions of these four optima and not as claims about all of them. Each of the four begins with an identity prefix — $0, 1, 2, 3, 4$ at $n = 23, 24, 25$

n	optimal ordering (t-th entry is the variable in position t)	prefix values $g_n(P_t)$, $t = 0, \ldots, n-1$
2	1 0	0 1
3	2 1 0	0 1 1
4	3 1 2 0	0 1 1 1
5	4 2 3 1 0	0 1 1 2 1
6	5 4 2 3 1 0	0 1 2 2 2 1
7	6 5 2 4 3 1 0	0 1 2 2 2 2 1
8	7 6 3 4 5 2 1 0	0 1 2 2 2 3 2 1
9	8 7 6 3 5 4 2 1 0	0 1 2 3 3 3 3 2 1
10	9 8 7 3 6 4 5 2 1 0	0 1 2 3 3 3 3 3 2 1
11	9 7 4 3 10 8 6 5 2 1 0	0 1 2 3 3 3 3 3 3 2 1
12	11 9 5 4 1 10 8 3 7 6 2 0	0 1 2 3 3 3 3 3 3 3 2 1
13	12 11 10 9 4 8 5 7 6 3 2 1 0	0 1 2 3 4 4 4 4 4 4 3 2 1
14	13 11 9 5 4 12 10 8 6 7 3 2 1 0	0 1 2 3 4 3 4 4 4 4 4 3 2 1
15	13 11 6 5 14 10 4 9 12 8 7 3 2 1 0	0 1 2 3 3 4 4 4 4 4 4 4 3 2 1
16	14 13 11 5 4 10 3 9 12 15 7 8 6 2 1 0	0 1 2 3 4 4 4 4 4 4 4 4 4 3 2 1
17	15 13 11 6 5 16 14 12 10 7 9 8 4 3 2 1 0	0 1 2 3 4 4 4 5 5 5 5 5 5 4 3 2 1
18	17 15 13 7 6 12 16 5 11 14 10 8 9 4 3 2 1 0	0 1 2 3 4 4 4 5 5 5 5 5 5 5 4 3 2 1
19	17 15 13 6 5 12 16 4 11 14 18 8 10 9 7 3 2 1 0	0 1 2 3 4 4 4 5 5 5 5 5 5 5 5 4 3 2 1
20	18 17 15 13 6 5 14 7 11 4 19 16 12 9 10 8 3 2 1 0	0 1 2 3 4 5 4 5 5 5 5 5 5 5 5 5 4 3 2 1
21	18 17 14 13 7 5 6 4 11 15 19 8 20 16 12 10 9 3 2 1 0	0 1 2 3 4 5 5 5 5 5 5 5 5 5 5 5 5 4 3 2 1
22	21 19 17 16 9 8 15 20 7 14 6 13 18 12 10 11 5 4 3 2 1 0	0 1 2 3 4 5 5 5 6 6 6 6 6 6 6 6 6 5 4 3 2 1

TABLE 3. One optimal ordering per n (Theorem 5.1), with the induced value profile. The maximum of each profile is $s_\star(n)$.

n	optimal ordering (t-th entry is the variable in position t)	prefix values $g_n(P_t)$, $t = 0, \ldots, n-1$
23	0 1 2 3 4 9 11 12 10 14 18 13 5 21 6 16 7 8 15 17 19 20 22	0 1 2 3 4 5 6 6 6 6 6 6 6 6 6 6 6 5 6 5 4 3 2 1
24	0 1 2 3 4 9 12 11 14 10 22 18 13 16 20 5 6 15 7 8 17 19 21 23	0 1 2 3 4 5 6 6 6 6 6 6 6 6 6 6 6 5 5 5 4 3 2 1
25	0 1 2 3 4 11 12 24 18 13 20 14 10 15 5 22 6 9 16 7 8 17 19 21 23	0 1 2 3 4 5 6 6 6 6 6 6 6 6 6 6 6 6 5 5 5 4 3 2 1
26	0 1 2 3 4 5 9 13 12 14 20 15 11 25 6 16 10 18 22 7 8 17 19 21 23 24	0 1 2 3 4 5 6 7 7 7 7 7 7 7 7 7 7 7 6 5 6 5 4 3 2 1

TABLE 4. One optimal ordering per n for $23 \le n \le 26$ (Theorem 5.2), with the induced value profile. The maximum of each profile is $s_\star(n)$.

and $0, 1, \ldots, 5$ at $n = 26$ — which no ordering of Table 3 does; part of that is an artifact of the new search's tie-breaking, but it does exhibit an optimum with such a prefix at each of these four n, which the earlier table did not. And the profiles at $n = 23$ and $n = 26$ dip by one after the maximum and climb back, so the non-monotone behaviour first seen at $n = 14$ and $n = 20$ is not rare. The descent found these four optima quickly — 57, 41, 89 and 51 nodes — where at $n = 34$ the same search needed 50 365; whatever the characterisation of π_{wopt} is, greedy descent is not uniformly close to it.

6. Two closed forms, one dead and one alive

Write $T(K) = \min\{n : s_\star(n) = K\}$ for the first-occurrence thresholds. From Table 1,

$$T(1), \ldots, T(6) \;=\; 2,\ 5,\ 8,\ 13,\ 17,\ 22. \tag{5}$$

6.1. **The formula that the first fifteen values force, and refute.** The first four entries of (5) are $\lceil m^2/2 \rceil$ for $m = 2, 3, 4, 5$, which is the sequence A000982 of the OEIS [6], $a(m) = \lceil m^2/2 \rceil$. A threshold sequence $T(K) = \lceil (K+1)^2/2 \rceil$ is exactly equivalent to the closed form $s_\star(n) = \lfloor \sqrt{2n} \rfloor - 1$. That closed form is correct on the whole range that suggests it, and then fails.

Theorem 6.1. *$s_\star(n) = \lfloor \sqrt{2n} \rfloor - 1$ for every n with $2 \le n \le 16$. It is false at $n = 17$: $s_\star(17) = 5$ while $\lfloor \sqrt{34} \rfloor - 1 = 4$. Nor is $s_\star(n) = \lceil n/4 \rceil$, which fails already at $n = 8$, where $s_\star(8) = 3$ and the formula gives 2.*

Proof. Both parts are immediate from Table 1, evaluated case by case at each n in the stated range; the underlying values are the exhaustive computations of Theorem 4.1. □

The interest of Theorem 6.1 is methodological. The fit is not idle numerology: it matches every one of the fifteen values available below $n = 17$, and its threshold sequence is a published integer sequence, so a computation stopped at $n = 16$ would have produced a confident false conjecture with an apparent independent confirmation. What actually happens is that $T(5) = 17$ and not 18, and thereafter $T(6) = 22$ and not 25.

6.2. **The law that fits the whole range.** From $K = 3$ onwards the thresholds (5) step $5, 4, 5$, which is $T(K) = \lceil (9K - 11)/2 \rceil$, equivalently $s_\star(n) = \lfloor (2n + 11)/9 \rfloor$.

Theorem 6.2. *$s_\star(n) = \lfloor (2n + 11)/9 \rfloor$ for every n with $2 \le n \le 22$ with the single exception $n = 4$, where $s_\star(4) = 1$ and the formula gives 2. Equivalently, $T(K) = \lceil (9K - 11)/2 \rceil$ for $3 \le K \le 6$.*

Proof. Case by case from Table 1, in three parts: $n \in \{2, 3\}$; the exception $n = 4$; and $5 \le n \le 22$. As above, the values themselves are the exhaustive computations of Theorem 4.1. □

Remark 6.3 (the measured continuation; outside the formal development). The same search, run outside the proof assistant, returns

$$s_\star(n) = 6, 6, 6, 7, 7, 7, 7, 7, 8 \qquad (n = 23, \ldots, 31),$$

so that $T(7) = 26$ and $T(8) = 31$, and $\lfloor (2n + 11)/9 \rfloor$ continues to agree at every one of these n as well. These nine values are *measurements*: they are produced by the same program that reproduces every value of Table 1 exactly, but they have no machine-checked proof, being beyond the budget of the compiled checks used above. The last is expensive because a threshold crossing must exhaust the failing pass before the succeeding one: $n = 30$ cost 106 seconds and 0.4 GB, $n = 31$ cost 1421 seconds and 0.36 GB, with about $3.5 \cdot 10^8$ evaluations of the split count and a peak frontier of 12.8 million sets.

Remark 6.4 (on the strength of the fit). Stated plainly: Theorem 6.2 is a two-parameter formula weighed against the eight first-occurrence thresholds now known — six proved above, two measured — of which it reproduces seven; the one it misses is $T(2) = 5$, where the formula would say 4, and that is the exceptional argument $n = 4$ seen from the other side. On the proved range $2 \le n \le 22$ it is right at twenty of the twenty-one values. It is not a theorem about $s_\star$ as a function on $\mathbb{N}$, and we offer no proof, no heuristic and no asymptotic argument for it. Two things nevertheless recommend recording it. It is *linear*, so it is a statement of a different kind from the sub-linear $\lfloor \sqrt{2n} \rfloor - 1$ that the initial segment suggested, and the slope $2/9$ sits strictly inside the interval $[\frac{1}{8}, \frac{1}{2}]$ permitted by (2); if it holds, the width of SMul^n_{n-1} is $2^{(2/9+o(1))n}$ and the sandwich of [1] becomes an exact exponent. And it is falsifiable at a price: the first untested prediction is $T(9) = 35$, which the search reaches in roughly an hour and a half of computation.

Remark 6.5 (the sequence databases; outside the formal development). Three numerical coincidences were met while checking that $s_\star$ is not a known sequence, and all three are recorded so that no reader has to rediscover them. (a) The partial threshold list $2, 5, 8, 13, 18, 25$ — that is, the continuation that $\lfloor\sqrt{2n}\rfloor - 1$ would force — is A000982 [6]; the true continuation $2, 5, 8, 13, 17, 22, 26, 31$ matches no sequence in the database. (b) The values of $s_\star$ for $2 \le n \le 20$ occur, as a run of terms, inside A011752, "repeat n m times, where m is the number of letters in the English word for the number"; the run ends there, because $s_\star$ takes the value 5 five times and A011752 four. (c) The values from $n = 13$ on occur inside A289133 [7], whose definition — the number of odd multiples of 9 in $]2(m-1)^2, 2m^2[$ — has nothing to do with decision diagrams, and which is simply the arithmetic function of Theorem 6.2 under another description: $\mathrm{A289133}(m) = \lfloor(2m+3)/9\rfloor$, so $\mathrm{A289133}(n+4)$ is that formula. So (c) matches the formula and not $s_\star$: it breaks at $n = 4$, exactly where Theorem 6.2 does. All three searches were run on 3 September 2026; none of them is prior computation of $s_\star$.

6.3. **The proved range in closed form, and the thresholds it gives.** With Theorem 4.5 the law of Theorem 6.2 covers four more values, and the whole proved range can be stated as one formula.

Theorem 6.6. *For every n with $2 \le n \le 26$,*

$$s_\star(n) = \begin{cases} 1, & n = 4, \\ \lfloor (2n+11)/9 \rfloor, & \textit{otherwise.} \end{cases}$$

Proof. Case by case in three parts: $n \in \{2, 3\}$; the exception $n = 4$; and $5 \le n \le 26$. The values themselves are the exhaustive computations of Theorems 4.1 and 4.5, collected in Table 5. □

The thresholds follow, and they are what the closing question of [1] — "an exact closed-form formulation" of $s_\star$ — is really about: a formula for $s_\star$ and a formula for the n at which it steps are the same information.

Theorem 6.7. *For $K = 1, \dots, 7$ the first-occurrence threshold $T(K) = \min\{n \ge 2 : s_\star(n) = K\}$ is*

$$T(1), \dots, T(7) \;=\; 2,\ 5,\ 8,\ 13,\ 17,\ 22,\ 26,$$

in the sense that $s_\star(T(K)) = K$ and $s_\star(n) < K$ for every n with $2 \le n < T(K)$.

Proof. The seven equalities $s_\star(T(K)) = K$ are seven entries of Table 1 and Theorem 4.5. For the strict inequalities, Theorem 6.6 gives the arithmetic step: if $2 \le n \le 26$ and $2n + 11 < 9K$ then $s_\star(n) < K$, the exceptional argument $n = 4$ included, since there $s_\star$ is smaller than the formula rather than larger. For $3 \le K \le 7$ the hypothesis $2n + 11 < 9K$ holds for every $n < T(K)$, because the value claimed for $T(K)$ is $\lceil (9K - 11)/2 \rceil$ at those K. The case $K = 1$ is vacuous, and the case $K = 2$ is exactly where the arithmetic step is unavailable — $T(2) = 5$ is the exception $n = 4$ seen from the other side — so $n = 2, 3, 4$ are checked individually. □

$T(7) = 26$ is the first threshold outside the range of Theorem 4.1; the earlier six lay inside it but had not been stated as thresholds. Beyond $K = 7$ the thresholds are measured, not proved.

n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$s_\star(n)$	1	1	1	2	2	2	3	3	3	3	3	4	4	4	4	5	5	5	5

n	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
$s_\star(n)$	5	6	6	6	6	7	7	7	7	7	8	8	8	8	9	9

TABLE 5. $s_\star(n)$ as far as it is known. Everything to the left of the second rule is proved (Theorems 4.1 and 4.5); the ten values $27 \le n \le 36$ to the right of it are measurements with no machine-checked proof (Remarks 6.3 and 6.8). The first-occurrence thresholds are $2, 5, 8, 13, 17, 22, 26$ (proved, Theorem 6.7) and then 31 and 35 (measured).

Remark 6.8 (the measured continuation, to $n = 36$; outside the formal development). The search of Remark 6.3, re-implemented over 64-bit partition masks (the earlier program held a partition in a 32-bit word and so stopped at $n = 32$) and run one (n, K) pass per invocation, returns

$$s_\star(32) = s_\star(33) = s_\star(34) = 8, \qquad s_\star(35) = s_\star(36) = 9,$$

so that $T(9) = 35$: every $n \le 34$ has $s_\star(n) \le 8$ — by Theorems 4.1 and 4.5 for $n \le 26$, by Remark 6.3 for $27 \le n \le 31$ and by the display for $n = 32, 33, 34$ — while $s_\star(35) = 9$. These five values are measurements in exactly the sense of Remark 6.3: no machine-checked proof, and no theorem of this paper uses them. Before the program was used beyond $n = 31$ it was made to reproduce, by both halves, the proved values $s_\star(21) = 5$ and $s_\star(22) = 6$ and the measured $s_\star(30) = 7$, together with $s_\star(31) \ge 8$; and its split count was compared against a literal transcription of Definition 1.1 on 300 000 sampled partitions, with no mismatch. Each value is one exhaustive pass over the live set $\{L : \hat{g}_n(L) < K\}$ for the lower half and one directed depth-first descent returning an explicit ordering for the upper half, and every one of those orderings was re-checked in a third implementation against Definition 1.1 with all $\binom{n}{2}$ pairs. The last two passes are the expensive ones: $n = 35$ took 1 558 seconds and 2.72 GB, with $4.18 \cdot 10^8$ evaluations of the split count and a peak frontier of 23.5 million sets, and $n = 36$ took 1 912 seconds and 4.25 GB ($4.75 \cdot 10^8$ evaluations, peak frontier 30.3 million).

Remark 6.9 (a correction to Remark 6.3). Remark 6.3 explains the cost of $n = 31$ — 1 421 seconds against 106 at $n = 30$ — by saying that a threshold crossing must exhaust the failing pass before the succeeding one. The numbers stand; the explanation is the wrong way round. Measured directly, the exhaustive pass at $(n, K) = (31, 8)$, which is the failing one, costs 100 seconds; nearly all of the 1 421 was spent in the *succeeding* pass at $K = 9$, which walks the much larger live set at threshold $s_\star(n) + 1$ until it stumbles on a chain that reaches $[n]$. That half is avoidable, and removing it is what made Remark 6.8 affordable: a depth-first descent that orders the children of a set by increasing $\hat{g}_n$ and remembers its failures produces an ordering of value 8 at $n = 31$ after 63 nodes, against the $3.5 \cdot 10^8$ split evaluations the undirected search spent. Upper bounds are therefore effectively free, and only the exhaustive pass costs anything — which is what Remark 7.3 extrapolates. The split was confirmed by running every pass of $n = 31$ under the one program: $K = 5, 6, 7, 8$ cost 0.5, 3.0, 18.3 and 119.6 seconds — the last of these is the 100 above, on a busier machine — and $K = 9$ cost 1 410.5, nine tenths of the whole. That last pass alone accounts for the $3.5 \cdot 10^8$ split evaluations and the peak frontier of 12.8 million that Remark 6.3 reports for $n = 31$, and reaches them exactly: $3.536 \cdot 10^8$ and 12 790 425.

Remark 6.10 (the fit, weighed again). Remark 6.4 weighed $\lfloor (2n + 11)/9 \rfloor$ against the eight thresholds then known, of which it reproduced seven, and named $T(9) = 35$ as the first untested prediction. The prediction has been tested and is correct. The formula now agrees with $s_\star$ at 34 of the 35 values $2 \le n \le 36$ — proved at 24 of the 25 values $2 \le n \le 26$ by Theorem 6.6, measured at the other ten — and reproduces eight of the nine thresholds now known, missing only $T(2) = 5$, which is the exceptional argument $n = 4$ seen from the other side. Four of the five new values, $n = 32, 33, 34$ and 36, are further agreements outside the range that fixed the two parameters. None of this is a proof, and the change should be stated exactly: before, the formula was a two-parameter law fitted to the data it was fitted to; now it has made one falsifiable prediction and has not been falsified. What is at stake is unchanged. If $s_\star(n) \sim \frac{2}{9}n$ then the width of SMul^n_{n-1} is $2^{(2/9+o(1))n}$ and the sandwich (2) of [1] collapses to an exact exponent; and no argument here bears on any n outside Table 5.

7. What is finite, what is not, and what is left

Each n is finite, and expensive. For fixed n, (1) is a minimum over $n!$ orderings of a maximum over $n\binom{n}{2}$ cells, and (4) reduces this to a bottleneck shortest-path problem on the Boolean lattice: 2^n vertices, $n2^{n-1}$ edges. So each value is decidable, and the cost doubles with each n. The certificates of Table 2 are cheaper than the full lattice — they inspect one level and a bounded neighbourhood above it — but they still walk all 2^n subsets to find that level.

The function is not finite. Nothing here bounds $s_\star$ outside the computed range, and no argument here is uniform in n. The following elementary bound, recorded outside the formal development, is the

only statement we have that is not confined to a finite range, and it is weaker than what [1] already quotes from [3].

Remark 7.1 (a counting bound; outside the formal development). Let $L \subseteq [n]$ with $|L| = m$ and suppose $g_n(L) \le K$. For $1 \le h \le n-1$ let $D(h)$ be the number of i with $0 \le i < n-h$ and $i, i+h$ on opposite sides. Summing over h counts each bichromatic pair once, at h equal to its difference, so $\sum_{h=1}^{n-1} D(h) = m(n-m)$. On the other hand a monochromatic residue class contributes nothing to $D(h)$, a class of size s contributes at most $s-1 \le \lceil n/h \rceil - 1$, and by Remark 2.5 at most K classes are bichromatic; also $D(h) \le n-h$ trivially. Hence for every m,

$$m(n-m) \;\le\; \sum_{h=1}^{n-1} \min\Big(n-h,\; K\big(\lceil n/h \rceil - 1\big)\Big), \qquad K = s_\star(n), \quad 0 \le m \le n-1,$$

because an ordering attaining the minimum in (1) has a prefix of every size $m \le n-1$, and each of those prefixes satisfies $g_n \le s_\star(n)$. The right-hand side is $O\big(Kn \log(n/K)\big)$, so taking $m = \lfloor n/2 \rfloor$ gives $s_\star(n) = \Omega(n/\log n)$; in particular no formula of order $\sqrt{n}$ can hold for all n, and this argument alone refutes $\lfloor \sqrt{2n} \rfloor - 1$ at $n = 239$, where it forces $s_\star(239) \ge 21 > 20$. The computation refutes it at $n = 17$, which is why the bound is recorded as background rather than as a result. It is weaker than the $n/8$ that [1] quotes from [3], and it never contradicts a computed value. Its only merits are that it is self-contained and that it is about the Split of Definition 1.1 rather than about the fooling-set count of [3], which is a different combinatorial object: the latter counts *pairs* at one fixed offset, the former counts residue classes. Nothing above depends on this remark.

One more statement about [1] itself, recorded because a reader who wants the width of SMul^n_{n-1} rather than the exponent $s_\star(n)$ will need the constants.

Remark 7.2 (the source's width and size constants; arithmetic on its own statements). Corollary 2.3 of [1] assumes a function $S_f(n)$ and constants $c_1, c_2 > 0$ such that under every ordering some variable has width at least $c_1 2^{S_f(n)}$, and under some ordering every variable has width at most $c_2 2^{S_f(n)}$; from these it concludes $2c_1 2^{S_f(n)} - 1 \le |\mathcal{B}(f)| \le c_2 n 2^{S_f(n)}$. For SMul^n_{n-1} its Theorem 3.3 supplies the lower constant $c_1 = 1$, while its Theorem 3.5 supplies $8n$ in the place of c_2. Substituting the two into the corollary's own conclusion gives

$$2 \cdot 2^{s_\star(n)} - 1 \;\le\; \big|\mathcal{B}(\mathrm{SMul}^n_{n-1})\big| \;\le\; 8n^2 \cdot 2^{s_\star(n)},$$

a factor $8n$ above the bound $n \cdot 2^{s_\star(n)}$ displayed in the introduction of [1], and, at the level of widths, $2^{s_\star(n)} \le \mathrm{Width} \le 8n \cdot 2^{s_\star(n)}$, that is $\mathrm{Width} = 2^{s_\star(n) + O(\log n)}$, where a $\Theta(2^{s_\star(n)})$ statement needs the upper factor to be independent of n. This is arithmetic and not a correction: which value of c_2 the author intends, and whether the $8n$ of Theorem 3.5 is tight, are questions for the author, and the three readings are in any case indistinguishable at the level this paper works at — if $s_\star(n) \sim \frac{2}{9}n$ then $2^{s_\star(n)}$, $n \cdot 2^{s_\star(n)}$ and $8n^2 \cdot 2^{s_\star(n)}$ are all $2^{(2/9 + o(1))n}$. Nothing in this paper depends on either bound; every result here is about $s_\star$ itself.

Remark 7.3 (the frontier of the method; outside the formal development). The next prediction of Theorem 6.2 is $T(10) = 40$, and the search used for Remark 6.8 does not reach it, for two measured reasons. *Time.* At fixed K the exhaustive pass grows by a factor between 1.16 and 1.24 per unit of n (135, 167 and 194 seconds at $n = 32, 33, 34$ with $K = 8$; 1 558 and 1 912 at $n = 35, 36$ with $K = 9$). The factor per unit of K at fixed n was measured directly, by ten further passes: at $n = 31$ the thresholds $K = 5, 6, 7, 8$ cost 0.5, 3.0, 18.3 and 119.6 seconds, at $n = 32$ the thresholds $K = 6, 7, 8$ cost 3.7, 21.4 and 138.5, and at $n = 34$ they cost 5.4, 32.3 and 205.6 — seven ratios, all between 5.8 and 6.5. (Those ten ran on a busier machine, which is why $(32, 8)$ and $(34, 8)$ come out a few per cent above the 135 and 194 just quoted; only the ratios are used.) That puts the single pass $(40, 10)$ at roughly $2.5 \cdot 10^4$ seconds, six to seven hours, before the values at $n = 37, 38, 39$ that would be needed beside it. *Memory.* The set of visited partitions is a bitmap of 2^n bits: 4.3 GB of address space at $n = 35$, 8.6 at $n = 36$, 17.2 at $n = 37$. Nor is the obvious improvement enough — a bitmap per level indexed by colexicographic rank, with two levels live, is about 1.1 GB at $n = 35$ but $\binom{40}{20}$ bits, about 17 GB again, at $n = 40$. So $n \le 36$

is the frontier of this method, and $T(10)$ waits on a different representation of the live set rather than on a larger machine.

What is left. Three concrete questions. (1) Is $s_\star$ monotone in n? It is on the whole computed range, and five of the links are now proved (Proposition 4.6(vi)), but the definition depends on n through both the index set and the range of the pairs (a, b), so this is not immediate. (2) Does $T(10) = 40$, as Theorem 6.2 predicts? The previous prediction of the same formula, $T(9) = 35$, was tested and held; this one is out of reach of the present search, and Remark 7.3 says what would have to change before it is not. (3) The question of [1]: a combinatorial characterisation of the minimising ordering. Tables 3 and 4 are the data on it, and the depth reformulation at the end of Remark 2.5 — for every h, the h residue-class intervals must have depth at most K everywhere, simultaneously in h — is the form in which we would attack it.

8. Verification

Every statement above asserted as a theorem or proposition — Theorems 3.2, 4.1, 4.5, 5.1, 5.2, 6.1, 6.2, 6.6 and 6.7, and Propositions 3.3, 4.3, 4.4 and 4.6 — is checked, in the form stated, by the Lean 4 proof assistant [8] over the Mathlib library [9]. Lemma 2.4 is the one exception, and deliberately so: it is the step at which (1) is transcribed into the notation of Definition 2.3, and in the formal development val_n is *defined* by its right-hand side, so the lemma belongs to the statement of what is verified rather than to the list of verified statements. The development is ten modules and about 1 230 lines: the definitions of Split, g_n, $\hat{g}_n$, orderings, val_n and $s_\star$ transcribed from [1] as in §2; the chain lemmas and the soundness of the level certificate; the twenty-five values with their witness orderings and certificate parameters; the second implementation of the certificate predicate and its identification with the first; and the controls, the closed-form table and the thresholds. It contains no `sorry`.

Two levels of trust are involved and they should be distinguished. The structural half — Theorem 3.2, the inequality $\hat{g}_n \le g_n$, the lemmas about prefixes and their encodings, and the whole of Proposition 3.3, including the form of Theorem 3.2 that takes its hypothesis in terms of the second implementation — depends only on `propext`, `Classical.choice` and `Quot.sound`, that is, on the kernel alone. The individual values, the optimal orderings, and every statement quantified over a range of n — Theorems 4.1, 4.5, 5.1, 5.2, 6.1, 6.2, 6.6, 6.7 and Propositions 4.3, 4.4, 4.6 — are exhaustive finite computations and are decided by compiled evaluation, so they additionally rest on the proof assistant's compiled-evaluation axiom; there is no other external input, no solver, no certificate file and no floating-point arithmetic anywhere. The bookkeeping is worth stating exactly, since the range statements inherit it: each of the twenty-five values $s_\star(n)$, $2 \le n \le 26$, carries two such axioms, one for its certificate and one for its witness ordering; each of the four optimality statements of Theorem 5.2 carries three, its own together with the two of the corresponding value; Theorems 6.6 and 6.7 and part (v) of Proposition 4.6, which quantify over the whole range $2 \le n \le 26$, each carry the union of all fifty; and parts (i) and (iii) of Proposition 4.6, being plain Boolean identities, carry `propext` and their own single axiom alone.

Total elaboration of the six modules that cover $n \le 22$ is about nine minutes on one core, the single largest check being the $n = 22$ certificate at three and a half minutes. The four modules that carry $23 \le n \le 26$ cost, with the compiled implementation of Proposition 3.3 loaded, three seconds to build that implementation, five for the identities of Proposition 3.3 themselves, 173 for the four values and their orderings, and 17 for the controls, the closed-form table and the thresholds. The searches that produced the witnesses — the lattice dynamic program, the depth-first descent behind Table 4, the sweep for the cheapest (m, d), the brute-force check over all $n!$ orderings for $n \le 10$, the exhaustive comparison of $\hat{g}_n$ with g_n over all 2^n subsets for $n \le 24$, and the 64-bit reachability search of Remarks 6.8 and 6.9 — are separate programs and are not trusted: only their outputs enter the theorems above, and each output is re-verified inside the proof assistant. The one place where an untrusted program's output is reported without such re-verification is the measured range $27 \le n \le 36$, which enters no theorem and is labelled a measurement wherever it appears. The material explicitly outside the formal development is labelled where it stands: Remarks 2.5, 4.2, 6.3, 6.5, 7.1, 6.8, 6.9, 7.2 and 7.3.

On the literature searches behind the provenance claim of §1.4: the abstract page of [1] was fetched on 3 September 2026 and shows one version only; two citation indices report zero citing works for it; a

repository search on five phrasings, and an inspection of the author's own public code, found no software artifact associated with the paper; the Mathlib library contains no development of binary decision diagrams at all; the OEIS queries are those of Remark 6.5, re-run the same day on the extended lists, where the values for $2 \le n \le 36$ and the threshold list $2, 5, 8, 13, 17, 22, 26, 31, 35$ each return nothing and the tail from $n = 20$ upwards returns only the coincidence A289133 of Remark 6.5(c) together with four short-run accidents; and a paper-scoped full-text scan of a local corpus of recent preprints for OBDD work co-occurring with integer multiplication returned nothing — with the honest caveat that the corpus is mathematics-heavy and holds few cs.CC-primary preprints, so it is the absence of a signal rather than evidence of absence.

The source of the development is currently held in a private repository: repository reference to be added at submission.

References

[1] T. Qin, *Upper and lower bounds on the OBDD-width of a special integer multiplication*, arXiv:2608.30664v1 [cs.CC], 31 August 2026 (cross-listed cs.DS; 11 pages, 1 figure). Title, author, category and version confirmed from the arXiv abstract page, fetched 3 September 2026, which lists v1 as the only version. All quotations above, Definition 1.1, and equation (1) are from the LaTeX e-print of that version. Every statement of [1] is cited above by the number it carries when that e-print is compiled as its `00README.json` prescribes (`pdflatex`, TeX Live 2025): Definition 2.1 (level and width) and the paragraph after it (partition with respect to an ordering); Definition 3.2 (SMul) and equation (3) (its value under a weight-two multiplier); Definition 3.3 (Split) and equation (4) ($s_\star$); Theorems 3.3 and 3.5 (the width sandwich) and Corollary 2.3 (width to size); Lemma 4.1 (Bryant's bound, as quoted there), Lemma 4.2 and Lemma 4.3 ($n/8 \le s_\star \le n/2$); and Claim 2, proved in Appendix A. These numbers are pinned to v1 and could move in a later version.

[2] R. E. Bryant, *Graph-based algorithms for Boolean function manipulation*, IEEE Trans. Comput. **C-35** (1986), no. 8, 677–691, doi:10.1109/TC.1986.1676819. Bibliographic data verified at Crossref, 3 September 2026; cited for the model.

[3] R. E. Bryant, *On the complexity of VLSI implementations and graph representations of Boolean functions with application to integer multiplication*, IEEE Trans. Comput. **40** (1991), no. 2, 205–213, doi:10.1109/12.73590. Bibliographic data verified at Crossref, 3 September 2026. Read here in the author's self-archived retypeset (`cs.cmu.edu/~bryant/pubdir/ieeetc91.pdf`, dated 24 July 1998) rather than in the journal typesetting, so its internal numbering is that of the retypeset: its Theorem 4 is the $\Omega(1.09^n)$ bound and its Lemma 3, "For any balanced partitioning (L, R), the Boolean function $Mult^n_{n-1}$ has a VLSI fooling set such that $|\mathcal{A}_{\mathrm{VLSI}}(L, R)| \ge 2^{n/8}$", is the fooling-set bound from which [1] takes the lower half of (2). The Split_p of that proof counts *pairs* at one fixed offset and is a different object from the Split of Definition 1.1.

[4] P. Woelfel, *Bounds on the OBDD-size of integer multiplication via universal hashing*, J. Comput. System Sci. **71** (2005), no. 4, 520–534, doi:10.1016/j.jcss.2005.05.004. Bibliographic data verified at Crossref, dblp and zbMATH, 3 September 2026. The two bounds quoted in §1 are its Theorem 3, "The OBDD-size of $MUL_{n-1,n}$ is at least $2^{\lfloor n/2 \rfloor}/61 - 4$", and its Theorem 4, "The OBDD-size for $MUL_{n-1,n}$ is at most $7/3 \cdot 2^{4n/3}$", read in the author's copy of the accepted manuscript (Elsevier preprint dated 25 January 2005), not in the journal typesetting. Concerns the unrestricted middle bit, so its bounds do not transfer to $s_\star$.

[5] B. Bollig and J. Klump, *New results on the most significant bit of integer multiplication*, Theory Comput. Syst. **48** (2011), no. 1, 170–188, doi:10.1007/s00224-009-9238-y; conference version in ISAAC 2008, 883–894. Crossref and dblp agree on volume, issue, pages and the print year 2011 (Crossref records the online version as 17 September 2009); verified 3 September 2026. Cited for context only; not consulted here.

[6] OEIS Foundation Inc., *The On-Line Encyclopedia of Integer Sequences*, entry A000982, $a(n) = \lceil n^2/2 \rceil$ (`oeis.org/A000982`), consulted 3 September 2026.

[7] OEIS Foundation Inc., *The On-Line Encyclopedia of Integer Sequences*, entry A289133, the number of odd integers divisible by 9 in $]2(n-1)^2, 2n^2[$ (`oeis.org/A289133`), consulted 3 September 2026.

[8] L. de Moura and S. Ullrich, *The Lean 4 theorem prover and programming language*, in: Automated Deduction — CADE 28, Lecture Notes in Computer Science **12699**, Springer, 2021, 625–635, doi:10.1007/978-3-030-79876-5_37.

[9] The mathlib Community, *The Lean mathematical library*, in: Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020), ACM, 2020, 367–381, doi:10.1145/3372885.3373824.