The cheapest F-type function costs n + 2 gates at n = 4, 5, 6

Luiz Antonio Busnello (Independent Researcher, ORCID 0009-0007-5911-8141)

15 October 2026 · doi:10.5281/zenodo.23393106

Abstract

A Boolean function is *F-type* if every size-optimal And-Inverter Graph for it contains at least two gates of fan-out at least two, while its minimum formula is exactly one gate larger than its minimum circuit. Let `μ(n)` be the minimum circuit size of an F-type function with exactly `n` essential variables. An elementary edge count gives `μ(n) ≥ n + 1`, and a lift construction gives `μ(n) ≤ n + 2` for every `n ≥ 4`. We record that `μ(4) = 6`, `μ(5) = 7` and `μ(6) = 8`, so the upper bound is attained at the first three arities, and we state `μ(n) = n + 2` as a conjecture. At `n = 6` no census of six-variable functions is needed: the equality case of the edge count fixes the shape of any circuit that could meet the lower bound, and the candidates reduce to 321 NPN classes. An exhaustive enumeration that uses no solver settles all of them, and a SAT computation run class by class on the same list agrees on every class. We have no proof for general `n`, and we say what one would have to use.

1. Model and definitions

We use the conventions of [K], the source of the gap question, without change. A *circuit* is a directed acyclic graph whose sources are the variables `x_1, …, x_n` and the constant `1`; every internal node is a two-input AND gate, and every edge and the output may be inverted at no cost. The *size* is the number of AND gates. The output is a designated wire, which may be a source, so a literal has size `0`. `opt(f)` is the minimum size of a circuit computing `f`. A *formula* is a circuit in which every non-output gate has fan-out exactly `1` (variables may be read any number of times), and `tree(f)` is the minimum size of a formula for `f`; `gap(f) = tree(f) − opt(f) ≥ 0`.

The *fan-out* of a gate is the number of *distinct* gates that read it; being the output does not count. A gate of fan-out at least two is *shared*.

Definition. Let `f` depend on all `n` of its variables. `f` is *forced-multi* if every circuit of size `opt(f)` computing `f` has at least two shared gates, and *F-type* if it is forced-multi and `gap(f) = 1`. Define

`μ(n) = min { opt(f) : f is F−type with exactly n essential variables }.`

All of `opt`, `tree`, `gap`, forcedness and the number of essential variables are invariant under the NPN group (input permutation, input negation, output negation), so `μ(n)` can be computed on NPN classes.

Lemma 1 below, the lift construction, and the values at `n = 4` and `n = 5`, each with its certificates, were made public earlier in [U]. The value at `n = 6`, the equality case (Lemma 2) and the conjecture are not in [U]. We repeat the earlier material so that the note can be read on its own.

2. The two bounds

Lemma 1 (lower bound). If `f` is non-constant, forced-multi and has `n` essential variables, then `opt(f) ≥ n + 1`.

Proof. Take a circuit `C` of size `K = opt(f)`. Optimality excludes a dead gate, a gate whose two operands are the same wire, and a read of an inessential variable (fixing it to a constant and propagating would give a smaller circuit); every essential variable is read. Count operand slots: `C` has `2K` of them, and each is filled by a variable or by a gate. Each of the `n` variables fills at least one, each of the `K − 1` non-output gates fills at least one, and each shared gate fills at least one more. With `R` shared gates,

$2K \geq n + (K - 1) + R, \quad \text{that is} \quad K \geq n - 1 + R,$

and $R \geq 2$ gives $K \geq n + 1$. ■

The same count with $R = 1$ gives the companion bound: $\mathtt{gap}(f) \geq 1$ implies $\mathtt{opt}(f) \geq n$. Lemma 1 is tight for forced-multi functions in general: at $n = 4$ the class of `0x0356` is forced-multi with $\mathtt{opt} = 5 = n + 1$, but its gap is 2. So the lemma alone cannot give $\mu(n) \geq n + 2$, and any proof of that must use $\mathtt{gap} = 1$.

Upper bound. If z is F-type and x is a fresh variable, then $z \wedge x$ is F-type with opt and tree both one larger (Lift Lemma, with a complete proof in [U]). Starting from `0x03de` at $n = 4$, which is F-type with $\mathtt{opt} = 6$ and $\mathtt{tree} = 7$, the tower $z_{n+1} = z_n \wedge x_{n+1}$ gives an F-type function with all n variables essential and $\mathtt{opt}(z_n) = n + 2$ for every $n \geq 4$. Hence $\mu(n) \leq n + 2$ for $n \geq 4$.

So $n + 1 \leq \mu(n) \leq n + 2$ for every $n \geq 4$, and the question at each arity is whether the single level $\mathtt{opt} = n + 1$ contains an F-type function.

3. The equality case

Lemma 2 (shape at the lower bound). Let f be forced-multi with n essential variables and $\mathtt{opt}(f) = n + 1$, and let C be an optimal circuit with $R \geq 2$ shared gates. Then $R = 2$; every essential variable is an operand of exactly one gate, exactly once; the two shared gates have fan-out exactly 2; every other non-output gate has fan-out exactly 1; the output gate has fan-out `0`.

Proof. With $K = n + 1$ the inequality of Lemma 1 reads $R \leq 2$, so $R = 2$, and then every inequality used in the count is an equality. ■

Lemma 2 gives no contradiction with $\mathtt{gap} = 1$ by itself. What it does is describe every possible counterexample completely: an F-type function with n essential variables and $\mathtt{opt} = n + 1$ would have an optimal circuit that is read-once in the variables and has exactly two shared gates of fan-out two. The enumeration of §5 checks Lemma 2 rather than assuming it: over every circuit of $n + 1$ gates that reads all n variables, for $n = 4, 5, 6$, no circuit with $R \geq 2$ falls outside this shape.

4. The three measured points

n	$\mu(n)$	level $\mathtt{opt} = n + 1$ empty because	level $\mathtt{opt} = n + 2$ populated by
4	6	the only forced-multi class at $\mathtt{opt} = 5$ is `0x0356`, with $\mathtt{gap} = 2$	`0x03de` ($\mathtt{opt}\ 6$, $\mathtt{tree}\ 7$)
5	7	exhaustive enumeration of all 62,400 candidates at $\mathtt{opt} = 6$, $\mathtt{gap} = 1$, no solver	six classes, two lifts and four non-lifts
6	8	enumeration over 321 NPN classes, two routes (§5)	the lift of `0x03de`

$n \leq 3$. There is no forced-multi function, hence no F-type function, with at most three essential variables. The enumeration program of §5 at $N = 2$ ($K = 2, 3$) and $N = 3$ ($K = 3, \ldots, 6$) finds no surviving class at any level, and these levels cover every possible opt: the largest AIG size is 3 with two essential variables and 6 with three. With one or none there is no gate.

$n = 4$. Lemma 1 gives $\mathtt{opt} \geq 5$. The six F-type classes at $n = 4$ have $\mathtt{opt} \in \{6, 7, 7, 7, 8, 8\}$; the only forced-multi class at $\mathtt{opt} = 5$ is `0x0356`, with $\mathtt{tree} = 7$. The minimum needs only $\mathtt{opt} \geq 6$ for each of the six, and that bound is certified by DRAT: unsatisfiable exact-synthesis instances at $k = 1, \ldots, 5$ for each class, 30 proofs

in all, re-verified with drat-trim [DT] on 12 October 2026. The list of six is closed by exhaustive enumeration and the four-variable census. The enumeration program used for `n = 6` (§5) reproduces the levels `K = 5, 6, 7` of the four-variable atlas class by class.

`n = 5`. Lemma 1 gives `opt ≥ 6`. The level `opt = 6` is empty of F-type functions by exhaustion without a solver: all 62,400 five-essential functions with `gap = 1` and `opt = 6` were enumerated, and for each one the minimum, over all its optimal circuits, of the number of shared gates is `1`. The level `opt = 7` contains 23,040 forced-multi functions among 453,600 candidates. They form six NPN classes: the AND- and OR-lifts of `0x03de`, which the Lift Lemma already places there, and four that are not lifts (`0x0001aac9`, `0x0003dcde`, `0x0007a8a7`, `0x000ff3fd`). Each of the four carries a complete DRAT chain with no symmetry-breaking argument: unsatisfiability at `k = 1, …, 6`, a re-simulated model at `k = 7`, `tree = 8` by an unsatisfiable formula instance at `k = 7` and a model at `8`, and forcedness by an unsatisfiable at-most-one-shared instance [U].

`n = 6`. Lemma 1 gives `opt ≥ 7`, the lift gives an F-type function with `opt = 8`, and §5 shows that the level `opt = 7` is empty. Hence `μ(6) = 8`.

5. The computation at n = 6

A census of six-variable functions is out of reach. The sandwich of §2 makes it unnecessary, because only the level `opt = 7` is open. An F-type function at that level would have an optimal circuit of seven gates with `R ≥ 2`, and it would read all six variables. So the candidates are exactly the functions computed by seven-gate circuits that read all six variables and have at least two shared gates.

Route 1: direct enumeration, no solver. Every seven-gate circuit that reads all six variables, with no dead gate and no redundant gate (constant, or equal up to negation to an earlier wire), was enumerated: 374,145,216 labelled circuits with polarities, variables labelled in order of first appearance. Of these, 7,801,856 have `R = 2`, all of the shape of Lemma 2, and none has `R ≥ 3`. Their functions are 104,608 distinct truth tables, which fall into 321 NPN classes. The canonical form is exact, over the full group of 92,160 transformations; the same code recovers the 222 classes at `n = 4`, and in the run at `n = 6` it is checked for invariance under random transformations. The 321 classes split as follows:

status	classes
fewer than six essential variables (outside the question)	30
`opt ≤ 6` (a circuit of 5 or 6 gates is in the NPN orbit)	146
not forced-multi (a 7-gate circuit with `R ≤ 1` is in the orbit)	78
forced-multi with `opt = 7`	67

All 67 survivors have no formula of size `≤ 8`, so `tree ≥ 9` and `gap ≥ 2`. None is F-type.

The exclusions are made by membership in the NPN orbits of the candidates (3,886,080 functions), so no step relies on choosing representatives. The same binary, run on the known levels, reproduces the atlas at `n = 4` (`K = 5, 6, 7`), the empty level `opt = 6` at `n = 5`, and the six classes at `opt = 7` at `n = 5`, with the same representatives. A first version of the program confused "reads all `n` variables" with "depends on all `n`". Route 2 caught it on a class with five essential variables, and the fix changed only classes outside the question, not the 67 survivors or the controls.

Route 2: SAT, class by class. Each of the 321 classes was decided again with the exact-synthesis encoder used for the four- and five-variable certificates, without using the enumeration's verdicts: circuit instances at `k = 5, 6`, an at-most-one-shared instance at `k = 7`, and formula instances at `k = 7, 8`. Every satisfying assignment was decoded and its circuit re-simulated. All 321 classes received the same status as in route 1, with no timeout. The 67 survivors have unsatisfiable circuit instances at `k = 5, 6`, an unsatisfiable at-most-one-shared instance at `k = 7`, and unsatisfiable formula instances at `k = 7, 8`. The slowest class took 740 s of wall time for all its instances together. For the 67 survivors these five unsatisfiable instances were solved again with proof logging,

and drat-trim verified all 335 proofs. The unsatisfiable answers for the other 254 classes carry no certificate. Route 1, which is exhaustive and uses no solver, does not depend on any of them. Route 2 takes its list of classes from route 1, so the two routes are independent in how they classify a class, and the completeness of the list rests on route 1 alone.

6. The conjecture

Conjecture. For every $n \geq 4$, $\mu(n) = n + 2$.

By §2 the conjecture is equivalent to the absence of F-type functions at the level $\mathtt{opt} = n + 1$. Since a forced-multi function at that level has $\mathtt{tree} \geq \mathtt{opt} + 1 = n + 2$, it is also equivalent to the statement:

> every forced-multi function with n essential variables and $\mathtt{opt} = n + 1$ has $\mathtt{tree} \geq n + 3$.

It holds at $n = 4, 5, 6$. The level $\mathtt{opt} = n + 1$ is not empty of forced-multi functions at those arities: it contains one, seven and 67 NPN classes respectively, all with $\mathtt{gap} \geq 2$. That is the whole of the evidence: three arities, the third being the first beyond a census. We do not claim the conjecture for any $n \geq 7$.

7. What would settle it

A proof has to use $\mathtt{gap} = 1$, because Lemma 1 is attained at $n = 4, 5, 6$ by forced-multi functions of larger gap. Lemma 2 says where to look. It is enough to show that a function computed by a circuit of $n + 1$ gates, read-once in the variables, with exactly two shared gates of fan-out two, has a smaller circuit, or has a circuit of $n + 1$ gates with at most one shared gate, or has no formula of $n + 2$ gates. Closing an explicitly defined subclass of these shapes (for instance a fixed relative position of the two shared gates) would be a partial result.

A refutation is an explicit function with $n \geq 7$ essential variables and an explicit circuit of $n + 1$ gates, together with certificates that $\mathtt{tree} = n + 2$ and that every circuit of $n + 1$ gates has two shared gates. Function and circuit are checkable by simulation. The two certificates are not, and would need unsatisfiability proofs or an argument.

The next point is $n = 7$, level $\mathtt{opt} = 8$. The present enumeration stores truth tables in 64 bits and accepts $n \leq 6$ only, so this point needs new code. We have not measured its cost.

8. What the argument does not use

- Lemma 1 uses no solver, no symmetry breaking and no hypothesis on $\mathtt{gap}$; it holds for every n.
- The upper bound uses no computation beyond the base case $\mathtt{0x03de}$ at $n = 4$.
- Route 1 at $n = 6$ uses no SAT solver, no DRAT, no gate-ordering symmetry break, and no canonical representatives in the exclusions (it works with full NPN orbits). It does not use Lemma 2 as a pruning rule; it checks it.
- The $n = 5$ level $\mathtt{opt} = 6$ is excluded by enumeration without a solver. The four non-lift witnesses at $\mathtt{opt} = 7$ rest on DRAT chains without any ordering argument.
- No step uses a characterisation of F-type functions. None is known to us, and two natural kinds of criteria fail already at $n = 4$. An invertible linear change of the inputs maps the F-type class $\mathtt{0x03de}$ to the class of $\mathtt{0x037e}$ ($\mathtt{opt}\ 8$, $\mathtt{tree}\ 9$, not forced-multi), so no criterion invariant under such changes separates F-type functions. The F-type class $\mathtt{0x07bc}$ and the class $\mathtt{0x07b4}$ ($\mathtt{opt}\ 7$, $\mathtt{tree}\ 8$, not forced-multi) have the same multiset of NPN classes of restrictions fixing one, two and three variables, so no criterion that depends only on that profile separates them either.
- No step assumes that an optimal formula is an optimal circuit plus one gate; $\mathtt{gap} = 1$ does not say that.

Data availability

The certificates and reproduction scripts for $n = 4$ and $n = 5$, and the proof of the Lift Lemma, are public in [U]. The enumeration program of §5, its outputs for every (N, K) used here, the encoders and per-class results of route 2, and the record of the 335 DRAT proofs for the 67 survivors, with the SHA-256 of each formula and a script that rebuilds them, are in the data record doi:10.5281/zenodo.23395917.

Use of generative AI

This work used AI models extensively, under the author's direction: Anthropic's Claude for the analysis, code, encodings and drafting, and models of other families for read-only review. The author is responsible for all claims. Correctness rests on the exhaustive enumeration of §5 and the DRAT proofs checked by `drat-trim`, not on the models.

References

- [K] K. Krinkin. The Unit Gap: How Sharing Works in Boolean Circuits. arXiv:2603.08033, version 2, 2026 (withdrawn in version 4). The source of the gap question and of the conventions of §1.
- [U] L. A. Busnello. Machine-checkable artefacts for the unit-gap and forced-reconvergence results, directory `theorem7/`: proof of the Lift Lemma, DRAT certificates and reproduction scripts for `n = 4` and `n = 5`. https://github.com/totobusnello/unit-gap-artefacts, commit `88c1807`, 13 October 2026.
- [DT] N. Wetzler, M. J. H. Heule and W. A. Hunt Jr. DRAT-trim: Efficient checking and trimming using expressive clausal proofs. In *Theory and Applications of Satisfiability Testing (SAT 2014)*, LNCS 8561, pp. 422–429, Springer, 2014.