SHALLOWER RELU NETWORK REPRESENTATIONS VIA EXACT LINEAR ALGEBRA

Anonymous authors
Paper under double-blind review

ABSTRACT

We study the number of hidden layers required by ReLU networks to represent piecewise-linear functions exactly, focusing on the maximum function. This problem has recently received significant attention in both the ML and TCS literature. We prove that $\max_n(x) = \max\{x_1, \dots, x_n\}$ is exactly representable with two hidden layers for every $n \leq 12$. Previously, the largest known arity was 5 (Bakaev et al., 2026, STOC '26). We obtain our constructions through an exact computer-assisted search within a space of candidate solutions. After a symmetry reduction, we obtain a finite system of linear equations over $\mathbb{Q}$ such that any solution yields a valid representation of the maximum function.

The resulting constructions have a structured first hidden layer, which enables recursive substitution into networks with more hidden layers. This yields an exact ReLU representation of $\max_n$ with at most $\lceil \log_6(n/2) \rceil + 1$ hidden layers. Consequently, every continuous piecewise-linear function on $\mathbb{R}^d$ admits an exact representation with at most $\lceil \log_6((d+1)/2) \rceil + 1$ hidden layers. In particular, two hidden layers suffice for $d \leq 11$. These results improve upon Bakaev et al. (2026, STOC '26), who proved analogous logarithmic bounds with base three.

1 INTRODUCTION

We study the minimum number of hidden layers required to represent continuous piecewise-linear functions exactly by ReLU networks of unrestricted width. A scalar-output ReLU network with ℓ hidden layers consists of affine maps $T_i \colon \mathbb{R}^{n_{i-1}} \to \mathbb{R}^{n_i}$, for $i = 1, \dots, \ell + 1$, with $n_{\ell+1} = 1$, and computes

$$f = T_{\ell+1} \circ \mathrm{ReLU} \circ T_\ell \circ \cdots \circ \mathrm{ReLU} \circ T_1,$$

where $\mathrm{ReLU}(x) \coloneqq \max\{0, x\}$ is applied coordinate-wise after each of the first ℓ affine maps. For a given function f, we ask for the smallest possible ℓ when the hidden-layer widths are unrestricted.

Every ReLU network computes a continuous piecewise-linear (CPWL) function: the input space can be partitioned into finitely many convex polyhedral regions such that the network is affine on each region. Conversely, every CPWL function admits an exact ReLU representation (Arora et al., 2018). With unrestricted width, one-hidden-layer networks can approximate continuous functions arbitrarily well on compact sets (Leshno et al., 1993), and approximation of $\max_n$ is studied by Safran et al. (2024). Here we ask how many layers are needed for *exact* equality on the entire domain.

The depth question for general CPWL functions reduces to maximum functions (Hertrich et al., 2023). Define

$$\max_n(x) \coloneqq \max\{x_1, \dots, x_n\},$$

and let CPWL_d denote the vector space of CPWL functions from $\mathbb{R}^d$ to $\mathbb{R}$. Wang & Sun (2005) showed that every $f \in \mathrm{CPWL}_d$ can be written as

$$f(x) = \sum_{i=1}^{m} \sigma_i \max_{d+1}(A_i(x)), \tag{1}$$

for some $m \in \mathbb{N}$, where $\sigma_i \in \{\pm 1\}$ and each $A_i \colon \mathbb{R}^d \to \mathbb{R}^{d+1}$ is affine. Hence, if $\max_{d+1}$ can be computed with ℓ hidden layers, then every function in CPWL_d can also be computed with at most ℓ hidden layers, since the summands in (1) can be computed in parallel with ℓ hidden layers.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Conversely, if $\max_{d+1}$ requires at least ℓ hidden layers, then $x \mapsto \max\{0, x_1, \ldots, x_d\}$, which lies in CPWL_d, also requires at least ℓ hidden layers, as shown by Hertrich et al. (2023). Therefore, upper and lower bounds on the number of hidden layers required to represent $\max_{d+1}$ directly imply corresponding bounds for functions in CPWL_d.

1.1 Bounds on the number of hidden layers for exact maximum representations

The elementary identity

$$\max\{x_1, x_2\} = \mathrm{ReLU}(x_1) - \mathrm{ReLU}(-x_1) + \mathrm{ReLU}(x_2 - x_1) \tag{2}$$

gives a one-hidden-layer representation of $\max_2$. Recursively applying this construction in a balanced binary tree gives a representation of $\max_n$ with $\lceil \log_2 n \rceil$ hidden layers. Together with Wang–Sun (1), this gives a $\lceil \log_2(d+1) \rceil$-hidden-layer representation of every function in CPWL_d (Arora et al., 2018, ICLR '18).

Despite substantial prior work, no CPWL function is known to require more than two hidden layers for exact representation when width is unrestricted. Hertrich et al. (2023, NeurIPS '21 / SIDMA '23) connected the problem to polyhedral geometry and conjectured that the balanced binary-tree construction $\lceil \log_2 n \rceil$ is optimal for $\max_n$. Haase et al. (2023, ICLR '23) proved the conjectured lower bound for networks with integral weights. For networks whose parameters are decimal fractions, that is, rationals of the form $a/10^k$, Averkov et al. (2025, ICLR '25) showed that representing $\max\{0, x_1, \ldots, x_n\}$ requires at least $\lceil \log_3(n+1) \rceil$ hidden layers. Other lower bounds assume compatibility of the breakpoint set with the braid arrangement (Grillo et al., 2025, NeurIPS '25), focus on monotone or input-convex networks (Bakaev et al., 2025; Valerdi, 2026), or impose a width bound when the number of hidden layers is fixed (Safran, 2026, COLT '26).

Without additional assumptions on the weights or breakpoints, however, the binary-tree construction is not optimal. Bakaev et al. (2026, STOC '26) disproved the conjecture of Hertrich et al. (2023) by constructing a two-hidden-layer representation of $\max_5$, and then bootstrapped this construction to show that $\lceil \log_3(n-2) \rceil + 1$ hidden layers suffice for $\max_n$. Through the Wang–Sun decomposition (1), this yields $\lceil \log_3(d-1) \rceil + 1$ hidden layers for every function in CPWL_d. Thus, before this work, the best known unrestricted upper bound scaled logarithmically with base 3, matching the lower bound established under the more restrictive assumption of decimal-fraction weights.

1.2 Main results

Our first result is an exact two-hidden-layer representation of $\max_{12}$ with rational weights, improving the largest previously known arity from 5 to 12 (Bakaev et al., 2026). Duplicating input coordinates extends the construction to every smaller arity. Together with the known lower bound (Mukherjee & Basu, 2017, Proposition 2.2), this determines the minimum number of hidden layers for every $3 \le n \le 12$.

Theorem 1.1. *For every integer $3 \le n \le 12$, two hidden layers are necessary and sufficient to represent $\max_n$ exactly by a ReLU network. Moreover, a representation exists with rational weights.*

The solutions have a structured first hidden layer, which conceptually computes only pairwise maxima. This structure enables an efficient recursive construction. As in the construction of Bakaev et al. (2026), which recursively reuses lower-arity maximum representations, we substitute such representations into the first hidden layer. In our construction, each additional hidden layer increases the represented arity by a factor of six, see Section 4. This improves the logarithmic base from 3 by Bakaev et al. (2026) to 6.

Corollary 1.2. *For every integer $n \ge 1$, the function $\max_n$ has an exact ReLU representation whose number of hidden layers is at most*

$$\left\lceil \log_6\left(\frac{n}{2}\right) \right\rceil + 1.$$

Table 1 compares the arities obtained at a small number of hidden layers by the binary tree construction (Arora et al., 2018), the construction of Bakaev et al. (2026), and our construction. Equivalently, writing the number of hidden layers as $c \log n + O(1)$, our construction reduces the leading coefficient by a factor of $\log(6)/\log(3) \approx 1.631$.

Hidden layers	Arora et al. (2018)	Bakaev et al. (2026)	This paper
2	$\max_4$	$\max_5$	$\max_{12}$
3	$\max_8$	$\max_{11}$	$\max_{72}$
4	$\max_{16}$	$\max_{29}$	$\max_{432}$

Table 1: Maximum arity produced at each number of hidden layers by the three constructions.

Setting $n = d + 1$ in the maximum-function bound and applying the Wang–Sun decomposition (1) gives the corresponding upper bound for all functions in CPWL_d.

Corollary 1.3. *For $d \geq 1$, every function in CPWL_d is exactly representable by a ReLU network with at most*

$$\left\lceil \log_6 \left(\frac{d+1}{2} \right) \right\rceil + 1$$

hidden layers. In particular, every function in CPWL_d with $d \leq 11$ admits an exact representation with two hidden layers.

Conceptual contribution. While Bakaev et al. (2026) disproved the conjecture of Hertrich et al. (2023) by pushing two-hidden-layer representability from $\max_4$ to $\max_5$ using geometric arguments, our conceptual contribution is that this can be pushed much further, to $\max_{12}$, by selecting a suitable candidate space, applying symmetry reductions, and performing exact-arithmetic linear algebra computations.

Furthermore, the upper bound of Bakaev et al. (2026) almost matches the lower bound of Averkov et al. (2025) for decimal fractions, both scaling logarithmically with base three. This leaves open whether general rational weights have an advantage over decimal fractions. The rationality of our construction implies that this is indeed the case, improving the upper bound from base three to base six, and separating those two weight classes already for two hidden layers: for $10 \leq n \leq 12$, rational weights suffice to represent $\max_n$ with two hidden layers, whereas decimal-fraction weights do not, see Corollary 3.3.

Finally, at this point, it seems unlikely to us that our bounds are optimal, with computational limits being the main obstacle for scaling our approach. The computational evidence provided in this paper therefore supports the following conjecture, which is the complementary extremal case compared to the now disproven conjecture by Hertrich et al. (2023).

Conjecture 1.4. *Every function in CPWL_d in any dimension d can be exactly represented by a ReLU network with two hidden layers.*

As discussed above, to prove this, it would suffice to prove it for the $\max$ function. It is of course conceivable that additional algorithmic insights will make it possible to compute $\max_n$ with some $n > 12$ in two hidden layers, and every such improvement will consequently lead to improvements in Corollary 1.2 and Corollary 1.3. However, computational results of this type would only increase the base of the logarithm further. A genuine advancement at this point would be a sub-logarithmic upper bound or a lower bound of more than two hidden layers, both of which seem out of reach with current techniques and probably require nontrivial new theoretical insights.

1.3 Idea of the construction

Our computational search was inspired by the polyhedral interpretation of the $\max_5$ construction of Bakaev et al. (2026). Using the duality between convex, positively homogeneous CPWL functions and polytopes via support functions, the construction derives a formal Minkowski difference from a subdivision of the simplex, and hence a ReLU representation.

With no bound on network width, a representation of $\max_n$ may involve arbitrarily many parameters, making a direct computational search difficult. To cast it into a finite-dimensional problem, we fix a finite family of candidate polytopes as generators for formal Minkowski differences. In Appendix A, we show that, once the candidate family is fixed, finding the coefficients in such a formal Minkowski difference is equivalent to solving a finite system of linear equations. Note that we make a specific choice of candidate polytopes for which a solution is not guaranteed even if there may be a solution

to the actual problem. We focused on highly structured families of polytopes to minimize the search space, as described below. For the actual search, we work directly with the equivalent piecewise-linear functions and use rank-2 maxout networks. A rank-2 maxout unit (Goodfellow et al., 2013) has the form $x \mapsto \max\{L_1(x), L_2(x)\}$, where L_1 and L_2 are affine. A rank-2 maxout network is a neural network in which every hidden unit is a rank-2 maxout unit. The following observation shows that passing to this representation does not affect the number of hidden layers. It follows immediately by applying (2) to each maxout unit.

Lemma 1.5. *Every rank-2 maxout network can be converted into a ReLU network with the same number of hidden layers. The conversion increases the width of each hidden layer by at most a factor of three. Further, if the maxout network has rational weights, so does the resulting ReLU network.*

The restriction to a finite candidate family is specific to our search and may exclude valid representations. The next reduction does not have this limitation. Since $\max_n$ is symmetric in its n arguments, we may restrict attention to permutation-invariant solutions. We average each candidate over coordinate permutations, so that both the target and the resulting candidates are S_n-invariant, where S_n denotes the symmetric group on n elements. Every vector in $\mathbb{R}^n$ can be permuted into the cone of sorted vectors

$$\mathcal{C}_n \coloneqq \{x \in \mathbb{R}^n \mid x_1 \le \cdots \le x_n\}.$$

Since both sides of the desired equality are permutation-invariant, it therefore suffices to establish equality on this cone. There, $\max_n(x) = x_n$, and the pairwise maxima $\max\{x_i, x_j\}$ in the first hidden layer of our construction simplify to $x_{\max\{i,j\}}$. Expanding the remaining maxima into linear terms and nonlinear ReLU terms reduces the desired identity to a finite system of linear equations over $\mathbb{Q}$, where the nonlinear terms must cancel, and the remaining linear term must equal x_n. We solve these systems using exact arithmetic, obtain rational rank-2 maxout representations, and convert them to ReLU representations with the same number of hidden layers using Lemma 1.5. The resulting identities are checked independently by an exact-arithmetic verifier.

2 Symmetry reduction to the sorted cone

Permutation symmetry allows us to reduce checking whether two symmetric functions agree on $\mathbb{R}^n$ to checking whether they agree on the cone of sorted vectors $\mathcal{C}_n$. Let the symmetric group S_n act on $\mathbb{R}^n$ by permuting coordinates. Explicitly, for $x \in \mathbb{R}^n$ and $\sigma \in S_n$, define $(\sigma x)_i \coloneqq x_{\sigma^{-1}(i)}$ for all $i \in \{1, \dots, n\}$. For any function $f\colon \mathbb{R}^n \to \mathbb{R}$, define its *symmetrization* by

$$f^{\mathrm{sym}}(x) \coloneqq \frac{1}{n!} \sum_{\sigma \in S_n} f(\sigma x).$$

We call $f\colon \mathbb{R}^n \to \mathbb{R}$ *symmetric* if $f(\sigma x) = f(x)$ for every $x \in \mathbb{R}^n$ and $\sigma \in S_n$. The symmetrization operator is linear and maps any function to a symmetric function. In particular, f is symmetric if and only if $f = f^{\mathrm{sym}}$. In invariant theory, this symmetrization is called the *averaging* (or *Reynolds*) operator associated with the action of S_n (Derksen & Kemper, 2015).

Proposition 2.1. *Let $g\colon \mathbb{R}^n \to \mathbb{R}$ be symmetric, and suppose that $g = \sum_{i=1}^m c_i f_i$ for some functions $f_i\colon \mathbb{R}^n \to \mathbb{R}$ and coefficients $c_i \in \mathbb{R}$. Then*

$$g = \sum_{i=1}^m c_i f_i^{\mathrm{sym}}.$$

Moreover, if $f, h\colon \mathbb{R}^n \to \mathbb{R}$ are symmetric, then $f = h$ on $\mathbb{R}^n$ if and only if $f = h$ on $\mathcal{C}_n$.

The proof is elementary and is deferred to Appendix B.1.

Since $\max_n(x) = x_n$ for every $x \in \mathcal{C}_n$, a symmetric function equals $\max_n$ if and only if it equals x_n on $\mathcal{C}_n$. The cone $\mathcal{C}_n$ is the closure of a chamber of the braid arrangement $\{x_i = x_j \mid 1 \le i < j \le n\}$. This braid arrangement has been used to study exact representations of the maximum function by ReLU networks (Grillo et al., 2025).

3 Search for two-hidden-layer identities

We now describe our setup for the computational search. We consider a restricted family of two-hidden-layer rank-2 maxout blocks in which the first hidden layer computes only pairwise maxima.

After symmetrizing these blocks and restricting to the sorted cone $\mathcal{C}_n$, the condition that a rational linear combination equals $\max_n$ reduces to a finite system of linear equations over $\mathbb{Q}$.

3.1 THE TWO-HIDDEN-LAYER ANSATZ

Let $\mathcal{E}_n \coloneqq \{(i,j) \mid 1 \le i \le j \le n\}$ denote the unordered pairs of indices with repetition, including the diagonal pairs, and define

$$m_{ij}(x) \coloneqq \max\{x_i, x_j\} \quad \text{for } x \in \mathbb{R}^n.$$

For $k \in \mathbb{N}$, let $\mathcal{M}_{n,k}$ denote the set of k-element multisets over $\mathcal{E}_n$. For $A, B \in \mathcal{M}_{n,k}$, define

$$\Phi_{A,B}(x) \coloneqq \max\left\{ \sum_{(i,j)\in A} m_{ij}(x), \sum_{(i,j)\in B} m_{ij}(x) \right\},$$

where, since both A and B have cardinality k, each sum consists of exactly k terms. Observe that $\Phi_{A,B}$ can be computed by a two-hidden-layer rank-2 maxout network, where the first hidden layer computes pairwise maxima, and the second computes the maximum of two sums of k first-layer outputs. We refer to those networks as *candidate blocks*, as each is a potential building block of the final network, but only a subset of them may ultimately be selected. By Lemma 1.5, every such candidate block has a two-hidden-layer ReLU realization with rational weights. The parameter k controls the complexity of the candidate blocks and the size of the resulting search space. To impose permutation symmetry,

$$F_{A,B}(x) \coloneqq \sum_{\sigma \in S_n} \Phi_{A,B}(\sigma x) = n!\, \Phi^{\mathrm{sym}}_{A,B}(x). \tag{3}$$

The factor $1/n!$ from the symmetrization operator is omitted because it only rescales the coefficients of a representation. We consider the vector space spanned by the finitely many $F_{A,B}$

$$V_{n,k} \coloneqq \operatorname{span}\{F_{A,B} \mid A, B \in \mathcal{M}_{n,k}\}.$$

Equivalently, $V_{n,k}$ is the class of functions represented by the fixed two-hidden-layer architecture whose second-hidden-layer outputs are the symmetrized candidate features $F_{A,B}$, when only the weights of the final affine output layer are allowed to vary. Our goal is therefore to find a certificate that $\max_n \in V_{n,k}$. The spaces are nested, $V_{n,k} \subseteq V_{n,k+1}$, as shown in Lemma D.3. Moreover, $\max_n \notin V_{n,k}$ for every $k < \lfloor (n-1)/2 \rfloor$, which we prove in Appendix D.1 using tools from polyhedral geometry.

3.2 REDUCTION TO A FINITE SYSTEM OF LINEAR EQUATIONS

By Proposition 2.1, it suffices to establish the desired identity on the sorted cone $\mathcal{C}_n$. For $x \in \mathcal{C}_n$, the coordinates are ordered, so the maximum of two coordinates is the one with the larger index. Hence

$$m_{ij}(\sigma x) = x_{\max\{\sigma^{-1}(i), \sigma^{-1}(j)\}}.$$

Accordingly, for $A \in \mathcal{M}_{n,k}$ and $\sigma \in S_n$ define the linear form

$$\ell_{\sigma,A}(x) \coloneqq \sum_{(i,j)\in A} x_{\max\{\sigma^{-1}(i), \sigma^{-1}(j)\}}.$$

Then, for $x \in \mathcal{C}_n$,

$$F_{A,B}(x) = \sum_{\sigma \in S_n} \max\{\ell_{\sigma,A}(x), \ell_{\sigma,B}(x)\}.$$

For each σ, let $a_{\sigma,A} \in \mathbb{N}^n$ be the coefficient vector of $\ell_{\sigma,A}$, so that $\ell_{\sigma,A}(x) = a_{\sigma,A}^\top x$. Define $d_{\sigma,A,B} \coloneqq a_{\sigma,B} - a_{\sigma,A} \in \mathbb{Z}^n$. Using $\max\{u,v\} = u + \mathrm{ReLU}(v-u)$ and grouping permutations with the same difference vectors gives

$$F_{A,B}(x) = L_{A,B}(x) + \sum_{d \in D_{A,B}} c_{A,B,d}\, \mathrm{ReLU}(d^\top x),$$

where $D_{A,B} \coloneqq \{d_{\sigma,A,B} \mid \sigma \in S_n\}$, $L_{A,B}$ is linear and $c_{A,B,d}$ is the multiplicity of d. We search for rational coefficients $\lambda_{A,B}$ satisfying

$$\sum_{A,B\in\mathcal{M}_{n,k}} \lambda_{A,B} c_{A,B,d} = 0 \qquad \text{for every } d \in \bigcup_{A,B\in\mathcal{M}_{n,k}} D_{A,B}, \tag{4}$$

$$\sum_{A,B\in\mathcal{M}_{n,k}} \lambda_{A,B} L_{A,B}(x) = x_n. \tag{5}$$

The equations in (4) cancel the nonlinear ReLU terms, while (5) requires the remaining linear part to equal $x_n = \max_n(x)$ on $\mathcal{C}_n$. Thus the search reduces to a finite system of linear equations over $\mathbb{Q}$. Any rational solution $(\lambda_{A,B})$ gives an exact identity

$$\max_n = \sum_{A,B\in\mathcal{M}_{n,k}} \lambda_{A,B} F_{A,B}$$

on $\mathcal{C}_n$, and hence on all of $\mathbb{R}^n$ by permutation symmetry. Since the blocks $F_{A,B}$ have rank-2 maxout realizations with rational weights and the coefficients $\lambda_{A,B}$ are rational, the resulting network also has rational weights. Applying Lemma 1.5 gives a ReLU network with the same number of hidden layers and rational weights.

Remark 3.1. Before assembling the system, we simplify the ReLU terms using elementary identities valid on $\mathcal{C}_n$. If $d^\top x$ has fixed sign on $\mathcal{C}_n$, then $\mathrm{ReLU}(d^\top x)$ is either identically zero or linear on the cone and can therefore be absorbed into the linear part. A criterion for deciding the case is given in Appendix D.3. Opposite directions are related by

$$\mathrm{ReLU}(-d^\top x) = \mathrm{ReLU}(d^\top x) - d^\top x,$$

and positive homogeneity gives

$$\mathrm{ReLU}(q\, d^\top x) = q\ \mathrm{ReLU}(d^\top x), \qquad q > 0.$$

We use these identities to identify equivalent nonlinear terms and reduce the number of constraints.

3.3 Enumeration modulo symmetries

The candidate pairs (A, B) have two immediate symmetries. First, $\Phi_{A,B} = \Phi_{B,A}$, and hence $F_{A,B} = F_{B,A}$. Second, simultaneous relabeling of all indices only permutes the summands in (3). Thus

$$F_{A,B} = F_{B,A} = F_{\tau A,\tau B} = F_{\tau B,\tau A} \qquad \text{for every } \tau \in S_n,$$

where τA is obtained by applying τ to every index and sorting the entries of each resulting pair. Let $C_2 = \{0, 1\} \cong \mathbb{Z}/2\mathbb{Z}$ act by exchanging the two multisets, and define the action of $S_n \times C_2$ by

$$(\tau, \varepsilon) \cdot (A, B) \coloneqq \begin{cases} (\tau A, \tau B), & \varepsilon = 0, \\ (\tau B, \tau A), & \varepsilon = 1. \end{cases}$$

Since $F_{A,B}$ is constant on orbits of this action, it suffices to retain one representative from each orbit when forming the system of linear equations.

Equivalently, A and B may be viewed as the two edge colors of a multigraph on $\{1, \ldots, n\}$, with loops and edge multiplicities allowed. Permuting the coordinates relabels the vertices, while exchanging A and B exchanges the two colors. Orbit enumeration therefore reduces to a colored-multigraph isomorphism problem with interchangeable colors. We enumerate these equivalence classes using *nauty* (McKay & Piperno, 2014).

3.4 Computational results

Theorem 1.1. *For every integer $3 \le n \le 12$, two hidden layers are necessary and sufficient to represent $\max_n$ exactly by a ReLU network. Moreover, a representation exists with rational weights.*

Proof. The cases $n = 3, 4$ follow from balanced binary-tree rank-2 maxout representations. For each $5 \le n \le 12$, we use $k = \lfloor (n-1)/2 \rfloor$, construct the system described above, and apply the reductions from Remark 3.1. For $n \in \{11, 12\}$, we also restrict the pairs (A, B) as described in

Remark 3.2. We then solve the resulting systems over $\mathbb{Q}$. The coefficients $\lambda_{A,B}$ of these solutions satisfy (4) and (5), and hence yield

$$\max_n = \sum_{A,B} \lambda_{A,B} F_{A,B}$$

on $\mathcal{C}_n$. Since both sides are permutation-invariant, Proposition 2.1 extends the identity to all of $\mathbb{R}^n$.

Each $F_{A,B}$ is realized by evaluating two-hidden-layer rank-2 maxout blocks in parallel and combining their outputs linearly. All coefficients appearing inside the blocks are integral, and the coefficients $\lambda_{A,B}$ are rational, so the resulting maxout network has rational weights. By Lemma 1.5, it converts to a ReLU network with the same number of hidden layers and rational weights.

Finally, Mukherjee & Basu (2017, Proposition 2.2) proves that $\max\{0, x_1, x_2\}$ has no one-hidden-layer ReLU representation. Restricting $\max_n$ to inputs of the form $(x_1, x_2, 0, \ldots, 0)$ transfers this lower bound to $\max_n$ for every $n \geq 3$, proving optimality. □

Table 2 summarizes the search for $n \leq 12$; the support size $K_{\text{cert}} := |\{(A, B) \mid \lambda_{A,B} \neq 0\}|$ is the number of nonzero coefficients in the reported certificate.

Remark 3.2. For $n \in \{5, \ldots, 12\}$, the systems remain solvable after restricting the ansatz further. In particular, it suffices to exclude diagonal pairs (i, i) and to take A and B to be sets of cardinality k rather than multisets, so that no pair occurs with multiplicity. Moreover, we can restrict to pairs (A, B) such that $A \cap B = \varnothing$ and such that, when A and B are viewed separately as edge sets of graphs on $\{1, \ldots, n\}$, both graphs are forests, i.e., acyclic.

Combining the rational-weight constructions with the previously established impossibility of representing $\max_n$ using decimal-fraction weights yields a separation between the two weight classes.

Corollary 3.3. *For $10 \leq n \leq 12$, $\max_n$ has a two-hidden-layer ReLU representation with rational weights, but none with decimal-fraction weights.*

Proof. The rational representation follows from Theorem 1.1. If a two-hidden-layer network with decimal-fraction weights represented $\max_n$, then setting one input identically to zero would yield a two-hidden-layer representation of $\max\{0, x_1, \ldots, x_{n-1}\}$ with decimal-fraction weights. For $10 \leq n \leq 12$, this contradicts the lower bound $\lceil \log_3 n \rceil = 3$ of Averkov et al. (2025). □

For $n = 6$ and $k = 2$, Figure 1 lists the four pairs of multisets that give the certificate

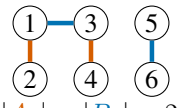
$$\max_6 = \frac{1}{720} F_{A_1,B_1} + \frac{1}{360} F_{A_2,B_2} - \frac{1}{1440} F_{A_3,B_3} - \frac{1}{360} F_{A_4,B_4}.$$

Here $F_{A,B}$ is the symmetrized function defined in Section 3.1. The expansions verifying this identity and a construction for $n = 5$, are given in Appendix C. The certificates for $n = 7, \ldots, 12$ are substantially larger and are therefore not reproduced in the paper. They are provided with the anonymized supplementary material. We also provide a verifier that checks the certificates directly, separating verification of the resulting identities from the code used to discover them.

n	k	$\lvert\mathcal{M}_{n,k}\rvert$	N_{rep}	$\lvert D\rvert$	Support size
5	2	120	131	15	3
6	2	231	144	35	4
7	3	4060	4469	630	57
8	3	8436	4716	1428	69
9	4	194 580	210 540	20 685	337
10	4	424 270	216 428	47 157	402
11	5	12 103 014	12 179 657	657 822	39 042
12	5	27 285 336	12 337 637	1 504 932	16 536

Table 2: Computational statistics for $5 \leq n \leq 12$. Here $|\mathcal{M}_{n,k}|$ is the number of individual size-k multisets, N_{rep} is the number of orbit representatives of pairs (A, B), and $|D|$ is the number of nonlinear direction vectors remaining after the reductions in Remark 3.1.

i	A_i	B_i
1	$\{\{(1,2),(1,2)\}\}$	$\{\{(3,4),(3,4)\}\}$
2	$\{\{(1,2),(1,3)\}\}$	$\{\{(1,4),(5,6)\}\}$
3	$\{\{(1,2),(1,3)\}\}$	$\{\{(4,5),(4,6)\}\}$
4	$\{\{(1,2),(3,4)\}\}$	$\{\{(1,3),(5,6)\}\}$



$|A_4| = |B_4| = 2$

$$F_{A_4,B_4}(x_1,x_2,x_3,x_4,x_5,x_6) = 72x_3 + 272x_4 + 472x_5 + 624x_6 + 8\,\mathrm{ReLU}(x_2 - x_3 - x_4 + x_6) + 16\,\mathrm{ReLU}(x_2 - x_3 - x_5 + x_6) + 24\,\mathrm{ReLU}(x_2 - x_4 - x_5 + x_6) + 48\,\mathrm{ReLU}(x_3 - x_4 - x_5 + x_6)$$

Figure 1: The four multiset pairs A_i, B_i (*left*), a bicolored-graph depiction of A_4, B_4 (*center*), and the decomposition of F_{A_4,B_4} into a linear part and nonlinear ReLU parts on $\mathcal{C}_6$ (*right*).

The setup is designed to make the search for two-hidden-layer representations computationally feasible. It does not optimize the width of the resulting construction, which can be factorial in n. In Appendix E, we discuss this width in more detail and compare it with the known quadratic lower bound (Safran, 2026).

4 Structured substitution and the base-six hidden-layer bound

To obtain representations of $\max_n$ for larger n, any D-hidden-layer representation of $\max_l$ can be composed in a tree-like fashion: applying copies of the representation to groups of at most l inputs increases the arity by a factor of l at the cost of D additional hidden layers.

For the specific two-hidden-layer representation constructed in this paper, the additional structure of its first hidden layer allows a more efficient composition. Exploiting this structure, each step increases the arity by a factor of $r := \lfloor l/2 \rfloor$ while requiring only one additional hidden layer.

Theorem 4.1. *Let $\mathcal{N}_l$ be a two-hidden-layer rank-2 maxout network representing $\max_l$, whose first hidden layer consists only of pairwise maxima $\max\{x_i, x_j\}$, and let $r := \lfloor l/2 \rfloor \geq 2$. Then, for every integer $s \geq 0$, the function $\max_n$ for $n = lr^s$ has an exact rank-2 maxout representation with $2 + s$ hidden layers whose first hidden layer again consists only of pairwise maxima. Consequently, for every $n \geq 1$, $\max_n$ has an exact ReLU representation with at most*

$$2 + \max\left\{0, \left\lceil \log_r \frac{n}{l} \right\rceil\right\}$$

hidden layers. If $\mathcal{N}_l$ has rational weights, then so do the resulting representations.

Proof sketch. We illustrate the substitution for the case $l = 12$, which yields the base-six recursion used below. Partition 72 inputs into 12 disjoint blocks $S_1, \ldots, S_{12}$ of size 6, and set

$$y_p := \max_{i \in S_p} x_i.$$

Evaluating $\mathcal{N}_{12}$ at $(y_1, \ldots, y_{12})$ would compute $\max_{72}$. By construction, every unit in the first hidden layer of $\mathcal{N}_{12}$ is a pairwise maximum. After the substitution $x_p \mapsto y_p$, such a unit becomes

$$\max\{y_p, y_q\} = \max_{i \in S_p \cup S_q} x_i,$$

a maximum of 12 original coordinates, which can therefore itself be computed by a copy of $\mathcal{N}_{12}$. Replacing each pairwise maximum in the first layer by a separate copy of the two-hidden-layer network $\mathcal{N}_{12}$ and combining affine maps gives a representation of $\max_{72}$ with three hidden layers; see Figure 2 for an illustration. Moreover, the new first hidden layer again consists only of pairwise maxima, so the substitution can be iterated. The general inductive proof is given in Appendix B.3. □

Corollary 1.2. *For every integer $n \geq 1$, the function $\max_n$ has an exact ReLU representation whose number of hidden layers is at most*

$$\left\lceil \log_6\left(\frac{n}{2}\right) \right\rceil + 1.$$

Proof sketch. The cases $n \in \{1, 2\}$ are immediate. For $n \geq 3$ apply Theorem 4.1 with $l = 12$ and $r = 6$. Simplifying the inequality gives the claimed upper bound. See Appendix B.2 for details. □

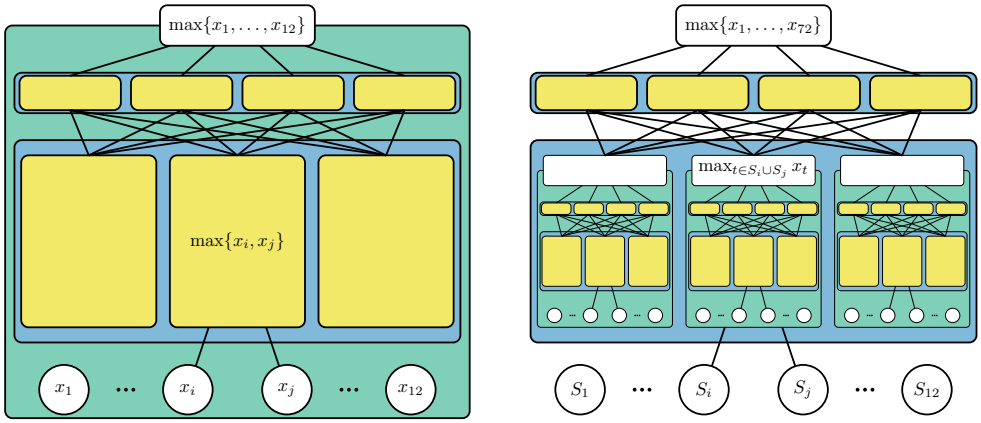


Figure 2: Substitution for $12 \to 72$. Each first-layer comparison $\max\{y_p, y_q\}$ of two block maxima is a maximum over at most 12 original inputs and can therefore be replaced by a copy of $\mathcal{N}_{12}$. This increases the number of hidden layers by one while multiplying the arity by six.

Corollary 1.3. *For $d \geq 1$, every function in CPWL_d is exactly representable by a ReLU network with at most*

$$\left\lceil \log_6 \left(\frac{d+1}{2} \right) \right\rceil + 1$$

hidden layers. In particular, every function in CPWL_d with $d \leq 11$ admits an exact representation with two hidden layers.

Proof. Follows from combining the Wang–Sun decomposition (1) with Corollary 1.2. □

5 Conclusion and Open Problems

We proved that $\max_n$ admits an exact two-hidden-layer ReLU representation for every $n \leq 12$. As a consequence, every CPWL function on $\mathbb{R}^d$ admits such a representation for $d \leq 11$. Previously, the construction for $\max_5$ might plausibly have been interpreted as a small-arity exception. The constructions found here show that this phenomenon extends considerably further. Whether two hidden layers suffice for $\max_n$ for arbitrary n, and hence for CPWL functions in arbitrary dimension, remains open.

The computer-assisted part of our proof searches within a restricted class of representations. We choose a finite pool of candidates, exploit permutation symmetry, and reduce the resulting identities to a system of linear equations on the sorted cone. These restrictions make the problem computationally accessible, but they leave out many possible two-hidden-layer representations. Consequently, failure to find a solution within this ansatz would not imply that $\max_n$ has no two-hidden-layer representation. Our computation finds solutions in this restricted ansatz up to $n = 12$. The case $n = 13$ remains unresolved because the corresponding computation exceeds the resources available to us.

It would be useful to understand whether the certificates for different values of n arise from a common construction. A construction valid for arbitrary n would remove the need for separate computations at each arity. At present, we do not know how to derive such a construction from the certificates found by our search. Our approach relies on symmetrization to make the problem computationally feasible. However, symmetrization also produces very large second hidden layers and may obscure the structure of the identities. Sparse or nonsymmetric solutions could reduce this width and reveal patterns hidden by full symmetrization.

We conjecture that two hidden layers suffice for $\max_n$ for every n. A weaker goal is to reduce the number of hidden layers to $o(\log n)$. The recursive substitution used here cannot yield such a bound, as providing identities for larger fixed arities can only improve the constant in the resulting $\Theta(\log n)$ bound. A sublogarithmic bound would require a substantially stronger substitution principle or some other theoretical insight.

Reproducibility Statement. All computer-assisted claims in this work can be verified using the artifacts provided in the supplementary material. For $5 \leq n \leq 12$, we provide the certificates used in Section 3 and two verifiers that check the corresponding identities using exact arithmetic. The Python verifier enumerates permutations directly and is practical for $n \leq 8$. The Rust verifier uses the subset dynamic program described in Appendix F.3 and can verify the certificates through $n = 12$. Both verifiers are separate from the search code, so verification requires only the certificates and a verifier. The supplementary material gives the software requirements and instructions for running them.

AI USE STATEMENT

We used large language models (LLMs) as auxiliary tools during the research and preparation of this work. LLMs were used to assist with writing, editing, and optimizing code, primarily to enable rapid exploration of ideas. We also experimented with using LLMs to identify patterns or derive further insights from the solutions found through our computational search; these attempts did not yield any insights or results incorporated into this work. Finally, LLMs were used to proofread the manuscript and provide suggestions for improving its presentation and readability.

The research conception, mathematical claims, and conclusions reported in this paper were developed by the authors and not generated by LLMs. The authors reviewed the AI-assisted material and take full responsibility for the final content of the paper.

REFERENCES

Raman Arora, Amitabh Basu, Poorya Mianjy, and Anirbit Mukherjee. Understanding deep neural networks with rectified linear units. In *International Conference on Learning Representations*, 2018.

Gennadiy Averkov, Christopher Hojny, and Maximilian Merkert. On the expressiveness of rational ReLU neural networks with bounded depth. In *International Conference on Learning Representations*, 2025.

Egor Bakaev, Florestan Brunck, Christoph Hertrich, Daniel Reichman, and Amir Yehudayoff. On the depth of monotone ReLU neural networks and ICNNs. arXiv preprint arXiv:2505.06169, 2025.

Egor Bakaev, Florestan Brunck, Christoph Hertrich, Jack Stade, and Amir Yehudayoff. Better neural network expressivity: Subdividing the simplex. In *Proceedings of the 58th Annual ACM Symposium on Theory of Computing*, Salt Lake City, UT, USA, 2026. ACM. doi: 10.1145/3798129.3800768.

Harm Derksen and Gregor Kemper. *Computational Invariant Theory*. Encyclopaedia of Mathematical Sciences. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015. doi: 10.1007/978-3-662-48422-7.

Ian Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28. PMLR, 2013.

Moritz Grillo, Christoph Hertrich, and Georg Loho. Depth-bounds for neural networks via the braid arrangement. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025.

Christian Alexander Haase, Christoph Hertrich, and Georg Loho. Lower bounds on the depth of integral ReLU neural networks via lattice polytopes. In *International Conference on Learning Representations*, 2023.

Christoph Hertrich, Amitabh Basu, Marco Di Summa, and Martin Skutella. Towards lower bounds on the depth of ReLU neural networks. *SIAM Journal on Discrete Mathematics*, 37(2), 2023. doi: 10.1137/22M1489332.

Christoph Koutschan, Bernhard Moser, Anton Ponomarchuk, and Josef Schicho. Representing piecewise linear functions by functions with small arity. *Applicable Algebra in Engineering, Communication and Computing*, 36(4), 2025. doi: 10.1007/s00200-023-00627-1.

Moshe Leshno, Vladimir Ya. Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6), 1993. doi: 10.1016/S0893-6080(05)80131-5.

Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism, II. *Journal of Symbolic Computation*, 60, 2014. doi: 10.1016/j.jsc.2013.09.003.

Anirbit Mukherjee and Amitabh Basu. Lower bounds over Boolean inputs for deep neural networks with ReLU gates. Technical Report TR17-190, Electronic Colloquium on Computational Complexity, Rehovot, Israel, 2017.

Itay Safran. A depth hierarchy for computing the maximum in ReLU networks via extremal graph theory. In *Conference on Learning Theory*, volume 336. PMLR, 2026.

Itay Safran, Daniel Reichman, and Paul Valiant. How many neurons does it take to approximate the maximum? In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 2024. doi: 10.1137/1.9781611977912.113.

Juan L. Valerdi. On minimal depth in neural networks. arXiv preprint arXiv:2402.15315, 2026.

S. Wang and X. Sun. Generalization of hinging hyperplanes. *IEEE Transactions on Information Theory*, 51(12), 2005. doi: 10.1109/TIT.2005.859246.

A SIGNED MINKOWSKI IDENTITIES VIA LINEAR ALGEBRA

This section gives the polyhedral origin of our search method. The construction of Bakaev et al. (2026) subdivides the standard simplex and turns the cells into a signed Minkowski identity for $\max_5$. We first used exact linear algebra to recover that identity from the cells and their intersections. The same observation then became our search principle: once a family of candidate polytopes is fixed, the unknown signed coefficients satisfy a finite system of linear equations. The main implementation works with the equivalent support-function equations on the sorted cone.

A *polytope* is the convex hull of finitely many points. The support function of a polytope $P \subset \mathbb{R}^n$ is

$$h_P(u) := \max_{p \in P} u^\top p.$$

Thus, if

$$f(x) = \max_{i \in \{1,\dots,m\}} a_i^\top x,$$

then $f = h_P$ for the Newton polytope $P := \operatorname{conv}\{a_1, \dots, a_m\}$. Conversely, the support function of every polytope is a convex, positively homogeneous CPWL function. Under this correspondence, addition of support functions corresponds to Minkowski addition, pointwise maxima correspond to convex hulls, and multiplication by a positive scalar corresponds to dilation. Signed linear combinations of support functions are therefore naturally interpreted as formal Minkowski differences.

In particular, $\max_n$ is the support function of the standard simplex $\operatorname{conv}\{e_1, \dots, e_n\}$, and $\max\{x_i, x_j\}$ is the support function of the segment $[e_i, e_j]$. Hence, the sums of pairwise maxima and the maxima of such sums used in the main text correspond, respectively, to Minkowski sums and convex hulls.

Bakaev et al. (2026) use a structured subdivision of the simplex to obtain a weighted Minkowski difference of polytopes whose support functions have two-hidden-layer representations. Their signed sum is the support function of the simplex and hence equals the maximum function.

Our search begins directly with candidate polytopes $Q_1, \dots, Q_r$ whose support functions have two-hidden-layer representations. We seek coefficients $c_i \in \mathbb{R}$ such that the formal Minkowski identity

$$P = \sum_{i=1}^{r} c_i Q_i$$

holds. Some coefficients may be negative. By definition, the identity means that

$$P + \sum_{c_i < 0} (-c_i) Q_i = \sum_{c_i > 0} c_i Q_i$$

as an equality of ordinary Minkowski sums. Equivalently, the support functions satisfy

$$h_P = \sum_{i=1}^{r} c_i h_{Q_i}.$$

Since the support functions are piecewise-linear, we may choose a complete polyhedral fan on whose cones they are all linear. On each cone, their equality is determined by their values on any finite conic generating set. This reduces the desired identity to a finite system of linear equations.

A *complete polyhedral fan* in $\mathbb{R}^n$ is a finite collection of polyhedral cones with apex at the origin, closed under taking faces, such that the intersection of any two cones is a face of each and the cones cover $\mathbb{R}^n$.

Proposition A.1. *Let Σ be a complete polyhedral fan, and let $P, Q_1, \dots, Q_r \subset \mathbb{R}^n$ be polytopes whose support functions are linear on every cone of Σ. For each $\tau \in \Sigma$, choose a finite conic generating set $U_\tau \subset \tau$, so that*

$$\tau = \operatorname{cone}(U_\tau).$$

Set

$$U := \bigcup_{\tau \in \Sigma} U_\tau,$$

and define

$$A_{u,i} \coloneqq h_{Q_i}(u), \qquad b_u \coloneqq h_P(u) \qquad (u \in U,\ i \in \{1,\dots,r\}).$$

Then a vector $c = (c_1, \dots, c_r)$ *satisfies*

$$h_P = \sum_{i=1}^{r} c_i h_{Q_i}$$

if and only if $Ac = b$.

Proof. Let

$$f \coloneqq h_P - \sum_{i=1}^{r} c_i h_{Q_i}.$$

The restriction of f to each cone $\tau \in \Sigma$ is linear. If $Ac = b$, then f vanishes on U_τ. Every point of τ is a nonnegative linear combination of elements of U_τ, so f vanishes on all of τ. Since the cones of Σ cover $\mathbb{R}^n$, we obtain $f \equiv 0$. The converse follows by evaluating the functional identity at the points of U. □

Proposition A.1 therefore gives a direct search procedure. Let $P \coloneqq \mathrm{conv}\{e_1, \dots, e_n\}$ be the standard simplex, choose candidate polytopes $Q_1, \dots, Q_r$ whose support functions have two-hidden-layer representations, and choose a common fan on which all relevant support functions are linear. The proposition reduces the signed Minkowski identity to a system of linear equations over $\mathbb{R}$. A solution gives a two-hidden-layer representation of $\max_n$. An inconsistent system rules out only the chosen candidate family.

Remark A.2. To recover the $\max_5$ construction of Bakaev et al. (2026), take its cells and their nonempty intersections as the candidate summands Q_i. Evaluating their support functions on generators of a common refinement of their normal fans produces $Ac = b$. A solution recovers the signed coefficients directly from the pieces of the subdivision.

B Deferred Proofs

B.1 Proof of Proposition 2.1

Proof of Proposition 2.1. By linearity of symmetrization and symmetry of g,

$$g = g^{\mathrm{sym}} = \left(\sum_{i=1}^{m} c_i f_i\right)^{\mathrm{sym}} = \sum_{i=1}^{m} c_i f_i^{\mathrm{sym}},$$

which proves the first claim. For the second claim, only the reverse implication requires proof. Suppose that $f = h$ on $\mathcal{C}_n$, and let $x \in \mathbb{R}^n$. Choose $\sigma \in S_n$ such that $\sigma x \in \mathcal{C}_n$. By symmetry,

$$f(x) = f(\sigma x) = h(\sigma x) = h(x).$$

Hence $f = h$ on $\mathbb{R}^n$. □

B.2 Proof of Corollary 1.2

Proof of Corollary 1.2. The cases $n = 11, 12$ are immediate. Assume $n \geq 3$. Apply Theorem 4.1 to the two-hidden-layer rank-2 maxout representation of $\max_{12}$ from Theorem 1.1. Here $l = 12$ and $r = 6$, so for every $n \geq 3$ the theorem gives

$$2 + \max\left\{0, \left\lceil \log_6\left(\frac{n}{12}\right)\right\rceil\right\} = 2 + \left\lceil \log_6\left(\frac{n}{12}\right)\right\rceil = \left\lceil \log_6\left(\frac{n}{2}\right)\right\rceil + 1.$$

The first equality holds because $n \geq 3$ implies $\lceil \log_6(n/12)\rceil \geq 0$. □

B.3 Proof of the substitution theorem

Proof of Theorem 4.1. We induct on s, maintaining the structure of the first hidden layer. For $s = 0$, the required representation is $\mathcal{N}_l$.

Suppose the claim holds for some $s \geq 0$, and set $M := lr^s$. Partition rM input coordinates into M disjoint blocks $S_1, \ldots, S_M$, each of size r, and write

$$y_p := \max_{i \in S_p} x_i.$$

Conceptually, evaluating the representation of $\max_M$ at $(y_1, \ldots, y_M)$ gives

$$\max_{1 \leq p \leq M} y_p = \max_{1 \leq i \leq rM} x_i.$$

By the induction hypothesis, every unit in the first hidden layer of this representation is a pairwise maximum. After the substitution $x_p \mapsto y_p$, such a unit becomes

$$\max\{y_p, y_q\} = \max_{i \in S_p \cup S_q} x_i.$$

This is a maximum of at most $2r \leq l$ original input coordinates. Hence it can be computed by a copy of $\mathcal{N}_l$, with repeated-coordinate padding if fewer than l distinct inputs occur.

In the network $\mathcal{N}_M$ representing $\max_M$, replace every first-layer unit in parallel by a copy of $\mathcal{N}_l$. The output affine maps of these copies can be absorbed into the affine map feeding the next maxout layer. Thus one hidden layer is replaced by two, increasing the total number of hidden layers from $2 + s$ to $3 + s$. The resulting network represents

$$\max_{rM} = \max_{lr^{s+1}}.$$

Moreover, its first hidden layer consists of the pairwise-max units from the inserted copies of $\mathcal{N}_l$, so the induction invariant is preserved.

For arbitrary $n \geq 1$, set

$$s := \max\Big\{0, \Big\lceil \log_r \frac{n}{l} \Big\rceil\Big\}.$$

Then $lr^s \geq n$. Feeding the n input coordinates into the lr^s inputs and repeating coordinates as necessary gives a representation of $\max_n$ with the same number of hidden layers. Finally, Lemma 1.5 converts the resulting rank-2 maxout network into a ReLU network without increasing the number of hidden layers. If the weights of $\mathcal{N}_l$ are rational, all operations above preserve rationality. □

C Explicit certificates for $n = 5$ and $n = 6$

We give explicit solutions for $n = 5$ and $n = 6$. For each row in the tables below, set $F_i := F_{A_i,B_i}$. We provide coefficients c_i such that $\sum_i c_i F_i = \max_n$, expanding the blocks on $\mathcal{C}_n$ after the canonical reductions of Remark 3.1.

C.1 Certificate for $n = 5$

i	A_i	B_i
1	$\{\{(1,1),(1,1)\}\}$	$\{\{(2,3),(2,3)\}\}$
2	$\{\{(1,1),(2,3)\}\}$	$\{\{(2,4),(3,5)\}\}$
3	$\{\{(1,2),(1,3)\}\}$	$\{\{(1,4),(1,5)\}\}$

Table 3: The three orbit representatives used in the certificate for $\max_5$.

Setting $h := \mathrm{ReLU}(x_2 - x_3 - x_4 + x_5)$, the blocks expand on $\mathcal{C}_5$ as

$$\begin{aligned} F_1 &= 24x_3 + 72x_4 + 144x_5, \\ F_2 &= 48x_3 + 84x_4 + 108x_5 + 12h, \\ F_3 &= 48x_3 + 64x_4 + 128x_5 + 16h. \end{aligned}$$

The nonlinear terms cancel, and on $\mathcal{C}_5$ (and therefore on all of $\mathbb{R}^5$), we get

$$\frac{1}{120}F_1 - \frac{1}{60}F_2 + \frac{1}{80}F_3 = x_5 = \max_5.$$

C.2 THE CASE $n = 6$

i	A_i	B_i
1	$\{\{(1,2),(1,2)\}\}$	$\{\{(3,4),(3,4)\}\}$
2	$\{\{(1,2),(1,3)\}\}$	$\{\{(1,4),(5,6)\}\}$
3	$\{\{(1,2),(1,3)\}\}$	$\{\{(4,5),(4,6)\}\}$
4	$\{\{(1,2),(3,4)\}\}$	$\{\{(1,3),(5,6)\}\}$

Table 4: The four orbit representatives used in the certificate for $\max_6$.

Set

$$\begin{aligned}
h_1 &= \mathrm{ReLU}(x_4 - 2x_5 + x_6),\\
h_2 &= \mathrm{ReLU}(x_3 - x_4 - x_5 + x_6),\\
h_3 &= \mathrm{ReLU}(x_3 - 2x_5 + x_6),\\
h_4 &= \mathrm{ReLU}(x_2 - x_3 - x_4 + x_6),\\
h_5 &= \mathrm{ReLU}(x_2 - x_3 - x_5 + x_6),\\
h_6 &= \mathrm{ReLU}(x_2 - x_4 - x_5 + x_6),\\
h_7 &= \mathrm{ReLU}(x_2 - 2x_5 + x_6).
\end{aligned}$$

On $\mathcal{C}_6$, each F_i is the sum of a linear term and a linear combination of the h_i. The coefficients are:

	x_3	x_4	x_5	x_6	h_1	h_2	h_3	h_4	h_5	h_6	h_7
F_1	0	96	384	960	0	0	0	0	0	0	0
F_2	72	272	400	696	12	64	8	8	16	32	4
F_3	0	192	480	768	48	64	32	0	0	32	16
F_4	72	272	472	624	0	48	0	8	16	24	0

Again the nonlinear terms cancel, and on $\mathcal{C}_6$ (and therefore on all of $\mathbb{R}^6$), we get

$$\frac{1}{720}F_1 + \frac{1}{360}F_2 - \frac{1}{1440}F_3 - \frac{1}{360}F_4 = x_6 = \max_6 .$$

D STRUCTURAL PROPERTIES OF THE ANSATZ

D.1 A NECESSARY LOWER BOUND ON k

The parameter k controls the expressivity of the ansatz, as shown in Lemma D.3. At the same time, the number of pairs of multisets $(A, B) \in \mathcal{M}_{n,k} \times \mathcal{M}_{n,k}$ grows rapidly with k, even after accounting for symmetries. It is therefore natural to ask for the smallest value of k that is not ruled out by theoretical obstructions. This question also has practical significance, since k directly governs the size of the system of linear equations we have to solve.

Proposition D.1. *Let $S \subset \mathbb{R}^N$ be a d-dimensional simplex. There do not exist polytopes $P_1, \dots, P_m \subset \mathbb{R}^N$ with $\dim P_r < d$ and coefficients $c_r \in \mathbb{R}$ such that*

$$h_S = \sum_{r=1}^{m} c_r h_{P_r}.$$

Proof. Let $L \coloneqq \mathrm{span}(S - S)$ and let $\pi\colon \mathbb{R}^N \to L$ be the orthogonal projection. Restricting the assumed identity to L gives

$$h_{\pi(S)} = \sum_{r=1}^{m} c_r h_{\pi(P_r)},$$

since $h_{\pi(P)}(u) = h_P(u)$ for every $u \in L$. The polytope $\pi(S)$ is a d-dimensional simplex in L, whereas

$$\dim \pi(P_r) \le \dim P_r < d.$$

After identifying L with $\mathbb{R}^d$, separate the positive and negative coefficients in the support-function identity. Additivity of support functions under Minkowski sums and positive dilations gives

$$h_{\pi(S)+\sum_{c_r<0}(-c_r)\pi(P_r)} = h_{\sum_{c_r>0} c_r\pi(P_r)}.$$

Since a compact convex set is determined by its support function, it follows that

$$\pi(S) + \sum_{c_r<0}(-c_r)\pi(P_r) = \sum_{c_r>0} c_r\pi(P_r).$$

Every polytope appearing in the two sums has dimension strictly less than d, and hence has zero volume in L. Thus $\pi(S)$ is a zero summand, contradicting Corollary 5.2 of Koutschan et al. (2025). □

Corollary D.2. *The systems of linear equations assembled in Section 3.2 can have a solution only if*

$$k \geq k_{\min} \coloneqq \left\lfloor \frac{n-1}{2} \right\rfloor.$$

Thus, $k_{\min}$ is the smallest value of k not ruled out by Proposition D.1.

Proof. For each (i,j), $m_{ij} = h_{[e_i,e_j]}$. Hence

$$\sum_{(i,j)\in A} m_{ij} = h_{Z_A}, \qquad \text{where} \qquad Z_A \coloneqq \sum_{(i,j)\in A} [e_i, e_j],$$

and $\dim Z_A \leq k$. Therefore

$$\Phi_{A,B} = \max\{h_{Z_A}, h_{Z_B}\} = h_{\mathrm{conv}(Z_A \cup Z_B)}.$$

Since Z_A and Z_B both have dimension at most k, their convex hull has dimension at most $2k+1$. Consequently, after expanding the symmetrized blocks into their summands, any solution of the exact system would express

$$\max(x_1, \ldots, x_n) = h_{\mathrm{conv}\{e_1,\ldots,e_n\}}$$

as a signed linear combination of support functions of polytopes of dimension at most $2k+1$. The polytope $\mathrm{conv}\{e_1, \ldots, e_n\}$ is an $(n-1)$-dimensional simplex. Hence, if $2k+1 < n-1$, this representation contradicts Proposition D.1. We must therefore have

$$2k + 1 \geq n - 1,$$

which is equivalent to $k \geq \lfloor (n-1)/2 \rfloor$. □

D.2 Nestedness of the search spaces

The parameter k does control the complexity of our ansatz. In the following, we prove that increasing k does not decrease the expressive power of the ansatz, i.e., if there is a solution for some k_0 then also for all $k > k_0$.

Lemma D.3. *Fix n. The spaces $V_{n,k}$ defined in Section 3.1 are nested: $V_{n,k} \subseteq V_{n,k+1}$.*

Proof. It suffices to show that every generator of $V_{n,k}$ belongs to $V_{n,k+1}$. Fix $A, B \in \mathcal{M}_{n,k}$ and choose any index pair $p = (i,j) \in \mathcal{E}_n$. Let $\uplus$ denote multiset union, i.e., addition of multiplicities, and define

$$A^+ \coloneqq A \uplus \{p\}, \qquad B^+ \coloneqq B \uplus \{p\},$$

so that $A^+, B^+ \in \mathcal{M}_{n,k+1}$. Adding the same term $m_p \coloneqq \max(x_i, x_j)$ to both multisets gives

$$\Phi_{A^+,B^+} = \Phi_{A,B} + m_p.$$

Symmetrizing both sides yields, for all $x \in \mathbb{R}^n$,

$$F_{A^+,B^+}(x) = F_{A,B}(x) + \sum_{\sigma\in S_n} m_p(\sigma x).$$

Now let C be the multiset consisting of $k+1$ copies of p. Since $C \in \mathcal{M}_{n,k+1}$,

$$F_{C,C}(x) = (k+1) \sum_{\sigma \in S_n} m_p(\sigma x).$$

Therefore,

$$F_{A,B} = F_{A^+,B^+} - \frac{1}{k+1} F_{C,C}.$$

Both terms on the right-hand side belong to $V_{n,k+1}$, and hence $F_{A,B} \in V_{n,k+1}$. Because $F_{A,B}$ was an arbitrary generator of $V_{n,k}$, it follows that $V_{n,k} \subseteq V_{n,k+1}$. □

D.3 Testing for linearity on the sorted cone

We give a computational criterion for determining whether a term $\mathrm{ReLU}(d^\top x)$ is linear on the sorted cone $\mathcal{C}_n$, and hence can be absorbed into the linear part. Every direction d arising from the expansion of some $\Phi_{A,B}$ is of the form $d = b - a$, where $a, b \in \mathbb{Z}^n_{\geq 0}$ are coefficient vectors with $a^\top \mathbf{1} = b^\top \mathbf{1} = k$. Hence $d^\top \mathbf{1} = 0$ and the following lemma gives the desired criterion.

Lemma D.4. *Let $d \in \mathbb{R}^n$ satisfy $d^\top \mathbf{1} = 0$, and let*

$$S_j \coloneqq \sum_{i=1}^{j} d_i.$$

Then $\mathrm{ReLU}(d^\top x)$ is linear on $\mathcal{C}_n$ if and only if $S_1, \ldots, S_{n-1}$ are all nonnegative or nonpositive.

Proof. Since $S_n = d^\top \mathbf{1} = 0$ and $d_j = S_j - S_{j-1}$ with $S_0 = 0$, summation by parts gives

$$d^\top x = \sum_{j=1}^{n} (S_j - S_{j-1}) x_j = \sum_{j=1}^{n-1} S_j (x_j - x_{j+1}).$$

For $x \in \mathcal{C}_n$, each difference $x_j - x_{j+1}$ is nonpositive and can be chosen independently. Hence $d^\top x$ has a fixed sign on $\mathcal{C}_n$ if and only if the S_j are either all nonnegative or nonpositive. □

E Width of the construction

Symmetrization makes the search tractable at the cost of a large second-layer width. Suppose the certificate has K_{cert} nonzero coefficients. Expanding each $F_{A,B}$ over permutations yields at most $n!$ blocks $\Phi_{\sigma A, \sigma B}$, so the second hidden layer contains at most

$$Q \leq K_{\mathrm{cert}} n!$$

rank-2 maxout units. The first hidden layer computes only pairwise maxima and therefore contains at most

$$\binom{n+1}{2} = \mathcal{O}(n^2)$$

maxout units. Hence the total number of maxout units is at most

$$\binom{n+1}{2} + Q.$$

For the corresponding ReLU network, consider the identity

$$\max\{x_i, x_j\} = \mathrm{ReLU}(x_j - x_i) + \mathrm{ReLU}(x_i) - \mathrm{ReLU}(-x_i) \qquad (i < j).$$

The terms $\mathrm{ReLU}(x_i)$ and $\mathrm{ReLU}(-x_i)$ can be shared across all pairs and also represent the diagonal terms $\max\{x_i, x_i\} = x_i$. Thus the first hidden layer needs at most one ReLU unit per pair and two per input coordinate, giving width at most

$$\binom{n}{2} + 2n = \mathcal{O}(n^2).$$

Each of the Q second-layer maxout units requires at most three ReLU units by Lemma 1.5, so the second hidden layer has width at most

$$3Q \le 3K_{\mathrm{cert}} n!.$$

Thus the direct symmetrized construction has quadratic first-layer width and, in the worst case, factorial second-layer width. This bound is not optimized and remains far above the quadratic lower bounds for two-hidden-layer representations established by Safran (2026).

F Efficient computation of the symmetrized functions

Constructing the system of linear equations in Section 3.2 requires, for each candidate pair $A, B \in \mathcal{M}_{n,k}$, a decomposition of the symmetrized function $F_{A,B}$ into a linear part and ReLU terms. Computing this decomposition by directly summing over all $n! = |S_n|$ permutations quickly becomes intractable; for instance, $n = 12$ already yields $12! = 479\,001\,600$ terms.

F.1 Strategy

Recall from Section 3.2 that on $\mathcal{C}_n$, the summand indexed by σ is the maximum of the two linear forms $\ell_{\sigma,A}$ and $\ell_{\sigma,B}$. Let $d_\sigma \in \mathbb{Z}^n$ be the coefficient vector of their difference, so that $\ell_{\sigma,B}(x) - \ell_{\sigma,A}(x) = d_\sigma^\top x$. Using the decomposition of maxima into a linear part and a ReLU term, we have

$$\max\{\ell_{\sigma,A}(x), \ell_{\sigma,B}(x)\} = \ell_{\sigma,A}(x) + \mathrm{ReLU}(d_\sigma^\top x).$$

The sum of the linear terms has a closed form (see Appendix F.5). For the ReLU terms, it suffices to count the multiplicity of each vector d_σ using a dynamic program, instead of iterating through all $\sigma \in S_n$.

To compute these multiplicities, regard a permutation as an ordering of the labels $\{1, \dots, n\}$ and construct it one label at a time. We track how the coordinates of d_σ change as each successive label is placed and count the orderings that yield the same difference prefix.

F.2 Difference vectors

Fix $A, B \in \mathcal{M}_{n,k}$, and write $V \coloneqq \{1, \dots, n\}$. For $C \in \{A, B\}$ and $u, v \in V$, let $\mu_C(u, v)$ denote the multiplicity in C of the pair $(\min\{u, v\}, \max\{u, v\})$. For $S \subseteq V$ and $v \notin S$, define

$$\alpha_C(S, v) \coloneqq \mu_C(v, v) + \sum_{u \in S} \mu_C(u, v), \qquad \Delta_{A,B}(S, v) \coloneqq \alpha_B(S, v) - \alpha_A(S, v). \tag{6}$$

The quantity $\alpha_C(S, v)$ counts the pairs in C whose last label is placed at this step: the pairs joining v to an element of S, together with the copies of (v, v). The difference $\Delta_{A,B}(S, v)$ is the corresponding contribution in this step to d_σ. A given $\sigma \in S_n$ defines such a chain of sets S:

$$S_r^\sigma \coloneqq \{\sigma(1), \dots, \sigma(r)\}, \qquad S_0^\sigma \coloneqq \varnothing,$$

and define $d_\sigma \in \mathbb{Z}^n$ by

$$\ell_{\sigma,B}(x) - \ell_{\sigma,A}(x) = d_\sigma^\top x. \tag{7}$$

Lemma F.1. *For every $\sigma \in S_n$ and $r \in \{1, \dots, n\}$,*

$$(d_\sigma)_r = \Delta_{A,B}(S_{r-1}^\sigma, \sigma(r)). \tag{8}$$

Equivalently,

$$d_\sigma = \big(\Delta_{A,B}(S_0^\sigma, \sigma(1)), \dots, \Delta_{A,B}(S_{n-1}^\sigma, \sigma(n))\big). \tag{9}$$

Proof. In $\ell_{\sigma,C}$, the coefficient of x_r counts the pairs of C whose last label in the order $\sigma(1), \dots, \sigma(n)$ is $\sigma(r)$, with a diagonal pair counted when its label occurs. These are exactly the pairs counted by $\alpha_C(S_{r-1}^\sigma, \sigma(r))$. Applying this to A and B and subtracting proves the claim. □

Adding $v \notin S_r$ to form $S_{r+1} \coloneqq S_r \cup \{v\}$ contributes $\Delta_{A,B}(S_r, v)$ to the next coordinate of d_σ by Lemma F.1. This contribution depends only on S_r and v, independently of the order in which the elements of S_r were placed. Hence orderings with the same set S_r and the same difference prefix can be grouped by their multiplicity.

F.3 SUBSET DYNAMIC PROGRAM

For $S \subseteq V$ with $|S| = r$ and $q \in \mathbb{Z}^r$, let $M_S(q)$ be the number of orderings of S that produce the difference prefix q. For a tuple q and an integer z, write $q \parallel z$ for the tuple obtained by appending z to q. Initially, all entries are zero except

$$M_{\varnothing}(()) = 1. \tag{10}$$

For nonempty $T \subseteq V$ and $q' \in \mathbb{Z}^{|T|}$, the recurrence is

$$M_T(q') = \sum_{v \in T} \sum_{\substack{q \in \mathbb{Z}^{|T|-1}: \\ q' = q \parallel \Delta_{A,B}(T\setminus\{v\}, v)}} M_{T\setminus\{v\}}(q). \tag{11}$$

Subsets are processed in increasing order of cardinality. In the implementation, M_S is a dictionary containing only prefixes with positive multiplicity. For each stored prefix q and each $v \notin S$, its multiplicity is added to the entry indexed by $q \parallel \Delta_{A,B}(S, v)$ in the dictionary for $S \cup \{v\}$.

Proposition F.2 (Correctness of the subset dynamic program)**.** *For every $d \in \mathbb{Z}^n$,*

$$M_V(d) = \#\{\sigma \in S_n \mid d_\sigma = d\}. \tag{12}$$

Consequently,

$$\sum_{\sigma \in S_n} \mathrm{ReLU}(d_\sigma^\top x) = \sum_{d \in \mathbb{Z}^n} M_V(d)\, \mathrm{ReLU}(d^\top x), \tag{13}$$

where only finitely many terms on the right are nonzero.

Proof. Every ordering of $S \cup \{v\}$ that ends in v is obtained uniquely by appending v to an ordering of S. By Lemma F.1, the last entry is $\Delta_{A,B}(S, v)$. Induction on $|S|$ therefore gives the stated interpretation of $M_S(q)$. At $S = V$, these orderings are the elements of S_n, which gives (12). Grouping equal vectors gives (13). □

Before applying the simplifications from Remark 3.1, we may therefore take

$$D_{A,B} = \{d \in \mathbb{Z}^n \mid M_V(d) > 0\}, \qquad c_{A,B,d} = M_V(d).$$

The fixed-sign, opposite-direction, and positive-scaling reductions from Remark 3.1 can then be applied directly to these grouped terms.

F.4 STATE COMPLEXITY

The size of the dynamic program is controlled by the number of possible difference prefixes, rather than by the number of permutations. For a prefix of length r, the contribution from each of A and B is a nonnegative integer vector with total mass at most k, since each of the k pairs contributes exactly once, namely when its later endpoint is placed. Hence every stored $q \in \mathbb{Z}^r$ satisfies $\|q\|_1 \le 2k$. To count the possible prefixes, choose their j nonzero coordinates, their signs, and their positive absolute values. This gives

$$N(r, m) \coloneqq \#\{q \in \mathbb{Z}^r \mid \|q\|_1 \le m\} = \sum_{j=0}^{\min\{r,m\}} 2^j \binom{r}{j}\binom{m}{j}.$$

Thus a subset of cardinality r has at most $N(r, 2k)$ states. Since there are $\binom{n}{r}$ such subsets and $n - r$ possible next labels, the number of transition updates is at most

$$\begin{aligned} T &\le \sum_{r=0}^{n-1} \binom{n}{r}(n-r)N(r, 2k) \\ &= n2^{n-1}\binom{n+2k-1}{2k}. \end{aligned} \tag{14}$$

The equality follows by substituting the formula for $N(r, 2k)$, exchanging the two sums, and applying the binomial theorem and Vandermonde's identity. For the choice $k = \lfloor (n-1)/2 \rfloor$ used in Section 3, we have $2k \le n - 1$, and hence

$$T \le n2^{3n-3} = \mathcal{O}(n8^n) = o(n!). \tag{15}$$

Thus the number of updates is singly exponential rather than factorial in n. At the terminal state, the number of distinct nonlinear terms satisfies $|D_{A,B}| \le N(n, 2k)$. Since $2k \le n$,

$$|D_{A,B}| \le N(n,n) \le \left(\sum_{j=0}^{n} 2^{j/2} \binom{n}{j} \right)^2 = \left(1+\sqrt{2}\right)^{2n} = \left(3+2\sqrt{2}\right)^n. \tag{16}$$

Thus the $n!$ summands produce at most exponentially many difference vectors before the reductions in Remark 3.1.

F.5 Linear contribution

It remains to compute the linear part of $F_{A,B}$. This can be expressed as a simple sum by a combinatorial argument. Let k_A^{diag} and k_A^{off} denote the numbers of diagonal and non-diagonal pairs in A, respectively, counted with multiplicity, so that $k_A^{\mathrm{diag}} + k_A^{\mathrm{off}} = k$.

Lemma F.3 (Symmetrized linear part)**.** *If $n \ge 2$, then*

$$L_{A,B}(x) = \sum_{r=1}^{n} \left(k_A^{\mathrm{diag}} (n-1)! + 2k_A^{\mathrm{off}} (r-1)(n-2)! \right) x_r. \tag{17}$$

If $n = 1$, then $L_{A,B}(x) = kx_1$.

Proof. Fix a position r. A fixed diagonal pair contributes to x_r in $(n-1)!$ permutations. For a fixed non-diagonal pair, choose which endpoint appears at position r, choose one of the $r-1$ earlier positions for the other endpoint, and permute the remaining labels. Thus it contributes to x_r in $2(r-1)(n-2)!$ permutations. Summing over the pairs in A gives (17). The case $n = 1$ is immediate. □