Structural Information Loss in Loop Flattening

Exact ambiguity counts and reconstruction-bit bounds

J. R. Landers

Abstract

Loop flattening can preserve every executed operation while erasing the visible geometry of the source iteration space. We formalize this loss for perfectly rectangular loop nests. If a depth-d nest with ordered bounds $(n_1, \ldots, n_d)$ is flattened to a single count $N = \prod_i n_i$, its compatible source structures are precisely the ordered factorizations of N into d factors at least two. We derive an exact formula for this ambiguity from the prime exponents of N, and show that the minimum fixed-length metadata required for exact reconstruction is $\lceil \log_2 K_d(N) \rceil$ bits. For prime powers $N = p^m$, the ambiguity reduces to $\binom{m-1}{d-1}$, yielding a binary-entropy-shaped depth profile whose maximum occurs at intermediate nesting depth. We also package the depth profile into a structural ambiguity polynomial and obtain a Möbius alternating-sum identity. The combinatorial ingredients are classical; the contribution is their interpretation as exact, transformation-specific measures of erased program structure.

1 Motivation

Consider the source program

```
for i in range(a):
    for j in range(b):
        work(i, j)
```

and a flattened implementation

```
for k in range(N):
    work_flat(k)        # N = a*b
```

The programs may perform the same work and produce the same outputs, yet the second representation does not by itself identify the original shape (a, b). When $N = 12$, the ordered shapes $(2, 6)$, $(3, 4)$, $(4, 3)$, and $(6, 2)$ are all compatible with the flattened count. The computation is preserved; source geometry is not.

This distinction matters whenever later stages attempt to recover, explain, debug, reroll, or parallelize optimized code. Compiler infrastructures contain explicit loop-flattening transformations, while decompilation and loop-rerolling research address the inverse problem of recovering higher-level repetition from lower-level representations [2, 5, 3]. Existing quantitative information-flow work measures uncertainty about hidden values from observable program behavior [6, 1]. Here the hidden variable is instead the source-level loop structure.

Contributions. This note makes four tightly scoped observations.

- It defines the *depth-resolved structural ambiguity* $K_d(N)$ of a flattened rectangular loop nest.
- It derives $K_d(N)$ exactly from the prime factorization of N.
- It converts that count into a sharp worst-case reconstruction cost of $\lceil \log_2 K_d(N) \rceil$ bits.
- It identifies a prime-power depth law, an entropy-shaped maximum at mid-depth, and an arithmetic signature $\sum_d (-1)^d K_d(N) = \mu(N)$.

2 Model and structural ambiguity

A perfectly rectangular, ordered loop nest of depth d is represented by

$$M = (n_1, \ldots, n_d) \in \mathbb{N}_{\geq 2}^d.$$

Its idealized flattening retains only

$$F(M) = \prod_{i=1}^{d} n_i = N.$$

The model intentionally suppresses index expressions, memory traces, debug metadata, and compiler conventions. Those can be added later as observations; the present question is what can be reconstructed from (N, d) alone.

Definition 1 (Depth-resolved ambiguity class)**.** For $N > 1$ and $d \geq 1$, define

$$\mathcal{M}_{N,d} = \left\{ (n_1, \ldots, n_d) \in \mathbb{N}_{\geq 2}^d : \prod_{i=1}^{d} n_i = N \right\}, \qquad K_d(N) = |\mathcal{M}_{N,d}|.$$

The integer $K_d(N)$ is the *depth-d structural ambiguity* induced by flattening.

Ordered dimensions are essential: $(2, 6)$ and $(6, 2)$ induce different coordinate maps even though they have the same product. The maximum possible depth is

$$\Omega(N) = \sum_{j=1}^{r} \alpha_j$$

when $N = \prod_{j=1}^{r} p_j^{\alpha_j}$; every loop bound must consume at least one prime factor.
If compatible shapes are equiprobable, the residual structural uncertainty is exactly

$$H(M \mid N, d) = \log_2 K_d(N).$$

For a nonuniform prior it is instead the posterior entropy over $\mathcal{M}_{N,d}$. The counting quantity $K_d(N)$ is prior-free and controls worst-case identifiability.

3 Exact ambiguity counting

Write the prime factorization of the flattened count as

$$N = \prod_{j=1}^{r} p_j^{\alpha_j}.$$

Every source dimension can be written

$$n_i = \prod_{j=1}^{r} p_j^{e_{ji}},$$

where each row of the exponent matrix (e_{ji}) sums to α_j. A zero column corresponds to the forbidden factor $n_i = 1$.

Theorem 2 (Exact depth ambiguity)**.** *For $N = \prod_{j=1}^{r} p_j^{\alpha_j} > 1$,*

$$\boxed{K_d(N) = \sum_{q=1}^{d} (-1)^{d-q} \binom{d}{q} \prod_{j=1}^{r} \binom{\alpha_j + q - 1}{q - 1}.}$$

Proof. First permit dimensions of size one. For a fixed prime p_j, distributing its α_j exponent units across q ordered dimensions is a weak-composition problem, giving

$$\binom{\alpha_j + q - 1}{q - 1}$$

possibilities. Distributions for distinct primes are independent, so the number of $r \times q$ exponent matrices, with zero columns allowed, is the product of these binomial coefficients.
To enforce $n_i \geq 2$, exclude exponent matrices containing zero columns. If exactly $d - q$ designated columns are empty, the exponents are distributed over the remaining q columns. Inclusion–exclusion over the d columns therefore gives

$$K_d(N) = \sum_{q=1}^{d} (-1)^{d-q} \binom{d}{d-q} \prod_{j=1}^{r} \binom{\alpha_j + q - 1}{q - 1},$$

and $\binom{d}{d-q} = \binom{d}{q}$. □

The formula is an inclusion–exclusion representation of the ordered factorization function with exactly d factors, a classical object in multiplicative partition theory [7]. Its role here is operational: it counts the source shapes erased by the transformation.

Table 1: Depth profiles for representative flattened counts.

N	factorization	K_1	K_2	K_3	K_4	K_5
12	$2^2 \cdot 3$	1	4	3	0	0
30	$2 \cdot 3 \cdot 5$	1	6	6	0	0
36	$2^2 \cdot 3^2$	1	7	12	6	0
72	$2^3 \cdot 3^2$	1	10	27	28	10
144	$2^4 \cdot 3^2$	1	13	48	76	55

4 Exact reconstruction cost

Suppose an auxiliary metadata string Z is attached to the flattened representation and must identify the original member of $\mathcal{M}_{N,d}$.

Theorem 3 (Sharp fixed-length recovery bound)**.** *The minimum number of fixed-length metadata bits required to recover every source shape in $\mathcal{M}_{N,d}$ is*

$$\boxed{B_{\min}(N,d) = \lceil \log_2 K_d(N) \rceil .}$$

Consequently, exact recovery under a budget of B bits is impossible whenever $K_d(N) > 2^B$.

Proof. A B-bit string has at most 2^B values. Exact reconstruction requires an injective map from $\mathcal{M}_{N,d}$ into those values, so $2^B \geq K_d(N)$. Conversely, enumerate the $K_d(N)$ compatible shapes and encode their indices using $\lceil \log_2 K_d(N) \rceil$ bits. □

Define the real-valued structural deficit

$$\Delta_B(N,d) = \log_2 K_d(N) - B.$$

The sign of Δ_B gives a clean point-of-no-return criterion: if $\Delta_B > 0$, no metadata scheme of capacity B can distinguish all compatible sources. If the source shape is random with posterior $P(M \mid N, d)$, the corresponding average description-length lower bound is $H(M \mid N, d)$ by source coding [4]. The fixed-length theorem is stronger in the worst case and requires no prior.

Minimal structure to restore. For depth two, one inner dimension b determines the other as $a = N/b$. Information-theoretically, however, the optimal metadata is not the integer b itself but its index among the $K_2(N)$ admissible divisors. That index needs exactly $\lceil \log_2 K_2(N) \rceil$ bits.

5 Prime powers and the depth law

The cleanest depth dependence occurs when $N = p^m$.

Corollary 4 (Prime-power depth law)**.** *For a prime p and $1 \leq d \leq m$,*

$$\boxed{K_d(p^m) = \binom{m-1}{d-1}}, \qquad \boxed{B_{\min}(p^m, d) = \left\lceil \log_2 \binom{m-1}{d-1} \right\rceil}.$$

Proof. Every factor is $n_i = p^{a_i}$ for some $a_i \geq 1$. The product constraint is equivalent to the composition $a_1 + \cdots + a_d = m$. Choosing $d-1$ separators among $m-1$ gaps gives $\binom{m-1}{d-1}$ ordered compositions. □

This law is nonmonotone in depth. There is one possible shape at $d = 1$, namely (p^m), and one at $d = m$, namely $(p, \ldots, p)$. Intermediate depths allow many allocations of exponent mass and are therefore harder to reconstruct.

Let

$$q = \frac{d-1}{m-1}.$$

Stirling's approximation gives

$$\log_2 \binom{m-1}{d-1} = (m-1)h_2(q) - \frac{1}{2}\log_2\big(2\pi(m-1)q(1-q)\big) + o(1),$$

where $h_2(q) = -q \log_2 q - (1-q)\log_2(1-q)$ is binary entropy. Its derivatives are

$$h_2'(q) = \log_2 \frac{1-q}{q}, \qquad h_2''(q) = -\frac{1}{\ln 2}\left(\frac{1}{q} + \frac{1}{1-q}\right) < 0.$$

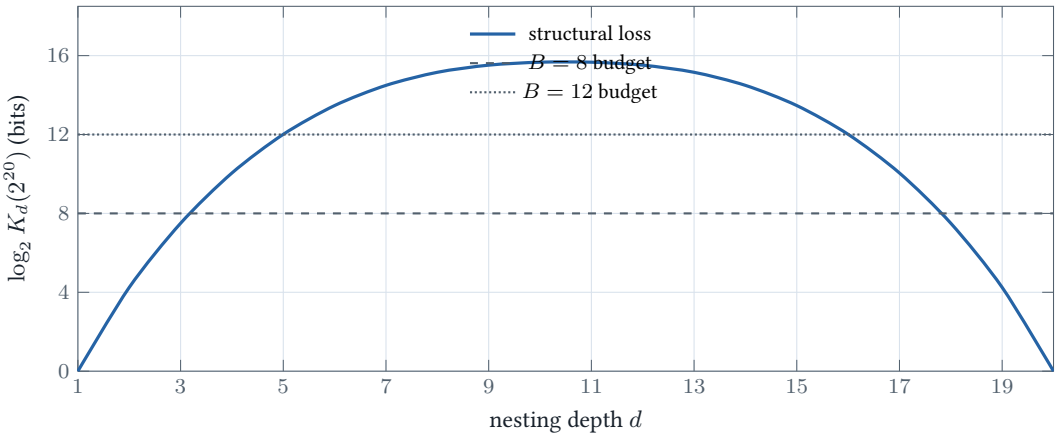

Figure 1: Prime-power structural loss is symmetric and peaks at intermediate depth. A budget line is exceeded precisely where exact recovery is impossible.

Thus the unique continuous maximum occurs at $q = 1/2$, giving the discrete maximizers nearest

$$\boxed{d^\star = \frac{m+1}{2}.}$$

We call this the *mid-depth ambiguity principle*: for a fixed prime-power workload, intermediate-depth source nests are the least identifiable after flattening.

If depth is also erased, summing over all possible depths yields

$$\sum_{d=1}^{m} \binom{m-1}{d-1} = 2^{m-1}.$$

Hence a flattened p^m workload with unknown depth loses exactly $m-1$ fixed-length bits of boundary-pattern information.

6 The ambiguity polynomial

The whole depth profile can be collected into one program invariant.

Definition 5 (Structural ambiguity polynomial)**.**

$$\mathcal{A}_N(z) = \sum_{d=1}^{\Omega(N)} K_d(N) z^d.$$

Let $N = \prod_{j=1}^{r} p_j^{\alpha_j}$ and introduce formal variables $\mathbf{x} = (x_1, \ldots, x_r)$. A valid loop dimension corresponds to a nonzero exponent vector, whose multivariate generating function is

$$G(\mathbf{x}) = \prod_{j=1}^{r} \frac{1}{1-x_j} - 1.$$

Therefore

$$K_d(N) = [x_1^{\alpha_1} \cdots x_r^{\alpha_r}] G(\mathbf{x})^d,$$

and summing the geometric series in d gives

$$\boxed{\mathcal{A}_N(z) = [x_1^{\alpha_1} \cdots x_r^{\alpha_r}] \frac{zG(\mathbf{x})}{1 - zG(\mathbf{x})}.}$$

Corollary 6 (Möbius depth signature)**.** *For every $N > 1$,*

$$\boxed{\mathcal{A}_N(-1) = \sum_{d=1}^{\Omega(N)} (-1)^d K_d(N) = \mu(N),}$$

where μ is the arithmetic Möbius function.

Proof. At $z = -1$,

$$\frac{-G}{1+G} = \prod_{j=1}^{r}(1 - x_j) - 1.$$

For a nonconstant coefficient, the final -1 contributes nothing. The coefficient of $x_1^{\alpha_1} \cdots x_r^{\alpha_r}$ in $\prod_j (1 - x_j)$ is zero if any $\alpha_j > 1$, and is $(-1)^r$ when all exponents equal one. This is exactly $\mu(N)$. □

The identity says that alternating cancellation across all possible source depths detects whether the flattened count is squarefree. We do not claim the number-theoretic identity itself as new; it is a natural consequence of ordered-factorization convolution. Its value here is that it gives the ambiguity polynomial a compact arithmetic signature.

7 Prototype calculation and implications

The exact formula is easy to evaluate once N is factored. The following direct implementation mirrors theorem 2.

```
def ambiguity(exponents, depth):
    total = 0
    for q in range(1, depth + 1):
        weak = prod(comb(a + q - 1, q - 1)
                    for a in exponents)
        total += (-1)**(depth-q) * comb(depth, q) * weak
    return total

def recovery_bits(exponents, depth):
    K = ambiguity(exponents, depth)
    return 0 if K <= 1 else ceil(log2(K))
```

The profile is sensitive not merely to the magnitude of N but to its prime-exponent geometry. For example, $N = 2^{12} = 4096$ has profile

$$1, 11, 55, 165, 330, 462, 462, 330, 165, 55, 11, 1,$$

whereas a squarefree number with twelve prime factors follows the ordered set-partition counts $d!\, S(12, d)$ and has a markedly different shape. Thus total work alone does not smoothly predict reconstructability: arithmetic structure and source depth jointly determine the loss.

The framework also separates three questions that are often conflated:

- **Semantic preservation:** does the transformation preserve observable behavior?
- **Structural identifiability:** how many source structures remain compatible?
- **Algorithmic recovery:** how hard is it to find the correct source given additional traces or metadata?

This paper addresses the second exactly for a deliberately minimal model. The bit bound is a capacity result, not a runtime-complexity claim.

8 Related work, limitations, and directions

Ordered factorizations. The coefficients $K_d(N)$ are ordered factorizations with a prescribed number of factors, studied as arithmetic functions and through generating series [7]. Our exact formula is a direct inclusion–exclusion form; the new perspective is to treat the coefficient as transformation-induced source ambiguity.

Loop transformations and reconstruction. LLVM implements a pass that flattens nested loop pairs into one loop [2]. In the reverse direction, loop rolling and rerolling reconstruct repeated structure from low-level code or hardware descriptions [3, 5]. Those systems infer structure from rich syntactic and semantic evidence. Our model asks what is possible when only the total iteration count and depth survive.

Quantitative information flow. QIF uses entropy, vulnerability, and channel models to quantify what observations reveal about hidden values [6, 1]. We use the same basic language of residual uncertainty but assign the hidden variable to source structure rather than secret input.

Limitations. We assume rectangular nests, constant integer bounds, known ordering, no early exits, and an observation model that suppresses index arithmetic and memory behavior. Real optimized programs may retain enough evidence to make the ambiguity class much smaller. Conversely, syntactically distinct programs may be considered equivalent by a richer source model. The present result is therefore a baseline: it isolates information erased by the product map itself.

Promising extensions. The closest next step is partial boundary retention: quantify $H(M \mid N, Z)$ when Z reveals selected loop boundaries or dimensions, then optimize which markers a compiler should preserve per metadata bit. Other promising

directions are transformation composition, where structural uncertainty should be monotone under further deterministic erasure; irregular domains with dependent bounds; and empirical comparison of the theoretical ambiguity with rerolling or decompilation success. More speculative directions include a time-indexed recovery horizon over compiler passes and connecting erased iteration-space geometry to uncertainty about available parallelism.

A dynamic recovery horizon. Let $X_0, X_1, \ldots, X_T$ be successively transformed representations of a source structure M, with $M \to X_t \to X_{t+1}$ a Markov chain. Define the residual structural loss at stage t by

$$L_t = H(M \mid X_t).$$

Data processing gives $I(M; X_{t+1}) \leq I(M; X_t)$ and therefore $L_{t+1} \geq L_t$: deterministic post-processing cannot recreate distinctions already absent from the current representation. Under a future metadata budget of B bits, define

$$\tau_B = \min\{t : L_t > B\}.$$

At and after τ_B, no auxiliary tag of capacity B can guarantee exact source recovery, since exact recovery from (X_t, Z) would require $H(M \mid X_t) \leq H(Z) \leq B$. This is a standard information-theoretic consequence rather than a new inequality, but transformation-specific estimates of L_t or τ_B could connect the static counting theory to real compiler pipelines.

Claim boundary. The paper does not present the ordered-factorization formulas, source-coding bound, or Möbius function as new mathematics in isolation. The proposed contribution is the synthesis: a semantics-preserving transformation induces a finite source ambiguity class; its depth profile is exactly computable for loop flattening; and its logarithm has an operational meaning as the smallest worst-case reconstruction tag. A stronger full paper would need either a broader class of transformations, a compiler implementation that selects minimal structural metadata, or evidence that $K_d(N)$ predicts practical reconstruction difficulty.

9 Conclusion

Loop flattening can preserve every execution while erasing a precisely countable amount of source geometry. For rectangular nests, that geometry is an ordered factorization class. Its cardinality gives an exact ambiguity measure; its logarithm gives the minimum reconstruction information; and its dependence on depth exposes a sharp mid-depth maximum for prime-power workloads. The result is intentionally small, but it provides a concrete model in which structural preservation can be discussed quantitatively rather than informally.

References

[1] M. S. Alvim, K. Chatzikokolakis, A. McIver, C. Morgan, C. Palamidessi, and G. Smith. *The Science of Quantitative Information Flow.* Springer, 2020.

[2] LLVM Project. *LoopFlatten.cpp: flatten pairs of nested loops into a single loop.* LLVM source documentation, accessed 2026.

[3] R. C. O. Rocha, P. A. M. Guimarães, and F. M. Q. Pereira. Loop rolling for code size reduction. In *CGO*, 2022.

[4] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.

[5] Z. D. Sisco et al. Loop rerolling for hardware decompilation. *Proceedings of the ACM on Programming Languages*, 7(PLDI), 2023.

[6] G. Smith. On the foundations of quantitative information flow. In *FoSSaCS*, 2009.

[7] J. Sprittulla. Ordered factorizations with k factors. arXiv:1610.04826, 2016.