Dimensional Expansion Transform (DET): A Simple Inversion Formula and Conditioning Analysis with an Engineering Use Case

Geby Jaff
UC Berkeley

Gemini DeepThink

GPT5 Pro

October 5, 2025

Abstract

We formalize a simple "dimensional expansion" viewpoint for polynomials: derivatives step down a dimension and anti-derivatives step up, while integer factors encode expansion multiplicities. The resulting linear operator on coefficients is the *Dimensional Expansion Transform* (DET). We solve the inverse problem (recovering coefficients from expansion counts) in two exact ways: (i) a standard $O(N^2)$ back-substitution and (ii) a closed-form $O(N)$ *adjacent-differences* formula

$$a_s = \Psi_s - (s+1)\Psi_{s+1} \quad (\Psi_{N+1} \equiv 0).$$

We then give a precise conditioning analysis: the forward matrix has

$$\kappa_\infty(U) = (N+1)\sum_{k=0}^{N} k!,$$

which grows on the order of $(N+1)!$ and is *not* uniformly well-conditioned. We also show that a factorial rescaling reduces the forward map to a first-sum operator with condition number growing only linearly in N. A worked cubic example and practical guidance are included. The mathematics is undergraduate-level (factorials, triangular systems, and basic norm bounds).

1 Introduction

Let $f(x) = \sum_{k=0}^{N} a_k x^k$ be a real polynomial. Interpret x^k as a k-dimensional power object at side length x. Heuristically, differentiation reduces degree by one and brings down a factor k (the number of $(k-1)$-dimensional *expansions* needed to sweep out x^k from x^{k-1}). Integrating increases degree by one and normalizes by $(k+1)$.

We make this viewpoint algebraic by defining *expansion counts* at unit side length $x = 1$ and assembling them into a linear transform on coefficients. The resulting operator is triangular and thus exactly invertible.

2 Dimensional Expansion Transform (DET)

Let $N \in \mathbb{N}$ be fixed and write

$$f(x) = \sum_{k=0}^{N} a_k x^k.$$

Definition 1 (Expansion counts at unit side length)**.** For $s \in \{0, 1, \ldots, N\}$ define

$$\Psi_s(f) \;:=\; \sum_{k=s}^{N} a_k \,\frac{k!}{s!}. \tag{1}$$

We refer to Ψ_s as the *s-dimensional expansion count* at $x = 1$, and write $\Psi(f) = (\Psi_0, \ldots, \Psi_N)^\top$.

Proposition 2 (Upper-triangular form)**.** *Let* $\mathbf{a} = (a_0, \ldots, a_N)^\top$ *and* $\boldsymbol{\Psi} = (\Psi_0, \ldots, \Psi_N)^\top$*. Then*

$$\boldsymbol{\Psi} = U\,\mathbf{a}, \qquad U_{s,k} = \begin{cases} \dfrac{k!}{s!}, & k \geq s, \\ 0, & k < s, \end{cases}$$

so U is upper triangular with $U_{s,s} = 1$.

Proposition 3 (Linearity and scaling)**.** *Ψ is linear: $\Psi(f+g) = \Psi(f)+\Psi(g)$ and $\Psi(\lambda f) = \lambda\Psi(f)$. If $f(\alpha x) = \sum_k a_k \alpha^k x^k$, then $\Psi_s(f(\alpha\cdot)) = \sum_{k \geq s} a_k \alpha^k \frac{k!}{s!}$.*

Proposition 4 (Derivative and anti-derivative at unit length)**.** *For a monomial $f(x) = a\,x^n$ we have*

$$\Psi_{n-1}(f) = a\,\frac{n!}{(n-1)!} = a\,n = f'(1).$$

If $g(x) = \int f(x)\,dx = \frac{a}{n+1} x^{n+1}$, then for $0 \leq s \leq n$,

$$\Psi_s(g) = \frac{a}{n+1}\,\frac{(n+1)!}{s!} = a\,\frac{n!}{s!} = \Psi_s(f),$$

while the only new count is $\Psi_{n+1}(g) = \frac{a}{n+1}$. By linearity, these statements extend termwise to polynomials.

3 Inverse problem: recover coefficients from counts

Given $\Psi_0, \ldots, \Psi_N$, recover $a_0, \ldots, a_N$.

Define $b_s := s!\,\Psi_s = \sum_{k=s}^{N} a_k\,k!$.

3.1 Two exact inversion formulas

(i) Back-substitution ($O(N^2)$). From $a_N = \frac{b_N}{N!}$ and

$$b_k = a_k k! + \sum_{j=k+1}^{N} a_j j!,$$

we obtain

$$a_k = \frac{1}{k!}\Big(b_k - \sum_{j=k+1}^{N} a_j j!\Big), \qquad k = N-1, \ldots, 0. \tag{2}$$

(ii) Adjacent differences ($O(N)$). Note $b_s = a_s s! + b_{s+1}$ (with $b_{N+1} := 0$), hence

$$a_s = \frac{b_s - b_{s+1}}{s!} = \Psi_s - (s{+}1)\Psi_{s+1}, \qquad \Psi_{N+1} := 0.$$

We record this as a theorem.

Theorem 5 (Linear-time inversion)**.** *The coefficients are recovered by the adjacent-differences rule*

$$\boxed{a_s = \Psi_s - (s{+}1)\Psi_{s+1}}, \qquad s = 0, 1, \ldots, N, \quad \Psi_{N+1} := 0. \tag{3}$$

Theorem 6 (Explicit inverse matrix)**.** *The inverse U^{-1} is upper bidiagonal:*

$$(U^{-1})_{k,k} = 1, \qquad (U^{-1})_{k,k+1} = -(k{+}1) \text{ for } k < N, \qquad \text{all other entries } 0.$$

Equivalently, $a_k = \Psi_k - (k{+}1)\Psi_{k+1}$.

Proof. We verify $UU^{-1} = I$. For $t > s$, we compute the matrix product $(UU^{-1})_{s,t}$. Since U^{-1} is bidiagonal, the only non-zero terms in the sum $\sum_j U_{s,j}(U^{-1})_{j,t}$ are for $j = t$ and $j = t-1$. Note that $t - 1 \geq s$ since $t > s$.

$$(UU^{-1})_{s,t} = U_{s,t}(U^{-1})_{t,t} + U_{s,t-1}(U^{-1})_{t-1,t} = \frac{t!}{s!} \cdot 1 + \frac{(t-1)!}{s!} \cdot (-(t)) = \frac{t! - t!}{s!} = 0.$$

For the diagonal entries $(UU^{-1})_{s,s} = U_{s,s}(U^{-1})_{s,s} = 1 \cdot 1 = 1$. Thus $UU^{-1} = I$ and reading the kth row of $\mathbf{a} = U^{-1}\boldsymbol{\Psi}$ gives (3). □

3.2 Worked example (cubic)

Suppose $\Psi_3 = 2,\ \Psi_2 = 5,\ \Psi_1 = 15,\ \Psi_0 = 18$. Then from (3)

$a_3 = \Psi_3 = 2, \quad a_2 = \Psi_2 - 3\Psi_3 = 5 - 6 = -1, \quad a_1 = \Psi_1 - 2\Psi_2 = 15 - 10 = 5, \quad a_0 = \Psi_0 - \Psi_1 = 18 - 15 = 3.$

Hence $f(x) = 2x^3 - x^2 + 5x + 3$. This matches the back-substitution solution.

4 Conditioning and numerical stability

We analyze conditioning with the ∞-norm $\|\cdot\|_\infty$ (maximum absolute row sum).

Proposition 7 (Exact ∞-norms)**.**

$$\|U\|_\infty = \sum_{k=0}^{N} k!, \qquad \|U^{-1}\|_\infty = N + 1.$$

Proof. For U (Proposition 2), the row sums are $\sum_{k=s}^{N} k!/s!$. This is maximized when $s = 0$, giving $\|U\|_\infty = \sum_{k=0}^{N} k!$.

For U^{-1} (Theorem 6), we examine the absolute row sums. For $0 \leq k \leq N-1$, the row sum is $|(U^{-1})_{k,k}| + |(U^{-1})_{k,k+1}| = |1| + |-(k+1)| = k+2$. For $k = N$, the row sum is $|(U^{-1})_{N,N}| = |1| = 1$.

The maximum absolute row sum is $\|U^{-1}\|_\infty = \max(\max_{0 \leq k \leq N-1}(k+2), 1)$. If $N = 0$, the maximum is 1. If $N \geq 1$, the maximum is attained at $k = N-1$, giving $(N-1) + 2 = N + 1$. In both cases, $\|U^{-1}\|_\infty = N + 1$. □

Theorem 8 (Condition number)**.** *The ∞-norm condition number is*

$$\boxed{\kappa_\infty(U) = \|U\|_\infty \|U^{-1}\|_\infty = (N{+}1) \sum_{k=0}^{N} k!}.$$

As $N \to \infty$, $\kappa_\infty(U) \sim (N{+}1)!$. Thus the forward map is not *uniformly well-conditioned.*

Remark 9 (Componentwise sensitivity)**.** Since $a_s = \Psi_s - (s{+}1)\Psi_{s+1}$,

$$|\,\Delta a_s\,| \;\leq\; |\Delta\Psi_s| + (s{+}1)|\Delta\Psi_{s+1}|,$$

which gives a transparent, per-coefficient error bound linear in s. This complements the global norm bound involving $\kappa_\infty(U)$.

4.1 Numerically friendly factorial scaling

Define $c_k := a_k\,k!$ and $b_s := s!\Psi_s$. Then

$$b_s = \sum_{k=s}^{N} c_k.$$

Let T be the $(N+1) \times (N+1)$ matrix with $T_{s,k} = 1$ for $k \geq s$ and 0 otherwise (upper-triangular ones). Then $b = Tc$. T^{-1} is the first-difference operator:

$$(T^{-1})_{k,k} = 1, \qquad (T^{-1})_{k,k+1} = -1 \text{ for } k < N.$$

One checks $\|T\|_\infty = N+1$. For T^{-1}, the absolute row sums are $|1|+|-1| = 2$ for $0 \leq k \leq N-1$, and $|1| = 1$ for $k = N$.

If $N = 0$, $\|T^{-1}\|_\infty = 1$ and $\kappa_\infty(T) = 1$. If $N \geq 1$, $\|T^{-1}\|_\infty = 2$ and

$$\kappa_\infty(T) = 2(N+1).$$

Thus, in the factorially scaled variables (b, c), the forward map has only linear growth in N, isolating the large factorials into easily handled rescalings.

5 Engineering use case: reverse-engineering a scaling law

In layer-by-layer manufacturing (e.g. 3D printing), tools may report counts of s-dimensional sweeps at unit scale. The DET framework gives a minimal, explainable model:

1. Measure $\Psi_0, \ldots, \Psi_N$ at $x = 1$ (counts of s-dimensional expansions).
2. Recover a_s by (3) (or by (2)).
3. Use $f(x) = \sum_{k=0}^{N} a_k x^k$ for prediction, $f'(x)$ for marginal change, and $\int_0^x f(t)\,dt$ for cumulative usage.

When N is unknown, fit several degrees and validate on held-out data. For noisy measurements or more than $N+1$ observations, solve a small least-squares problem in (b, c) with $b = Tc$.

6 Numerical check

For selected N, the exact values

$$\kappa_\infty(U) = (N+1) \sum_{k=0}^{N} k!$$

are (rounded ratios for display):

N	$\kappa_\infty(U)$	$\kappa_\infty(U)/(N+1)!$
3	40	1.6667
10	44,417,054	1.11274
15	22,425,642,181,024	1.07183
20	53,787,877,376,348,226,594	1.05279

These confirm factorial-order growth with a ratio $\to 1$ as N increases.

7 Practical guidance

- **Prefer the $O(N)$ formula.** Use $a_s = \Psi_s - (s+1)\Psi_{s+1}$.
- **Rescale for regression.** If you fit with noise, work with $b_s = s!\Psi_s$ and $c_k = a_k k!$ so the forward map is $b = Tc$ with conditioning linear in N.
- **Watch for cancellation.** If $\Psi_s \approx (s+1)\Psi_{s+1}$, then a_s is small and sensitive by necessity (the model is nearly indistinguishable in that component).

8 Conclusion

We defined the Dimensional Expansion Transform (DET), gave two exact inversions (including a linear-time adjacent-differences rule), and established precise conditioning bounds. The framework is compact (factorials, sums, first differences), explainable, and useful for reverse-engineering scaling laws from expansion counts in manufacturing contexts.

Appendix: Minimal reproducibility script (optional)

```
# Python 3; verifies UU^{-1} = I, computes κ_∞(U),
# and demonstrates the O(N) inversion.
import math
def kappa_inf(N):
S = sum(math.factorial(k) for k in range(N+1))
return (N+1)*S
def U_inv_apply(Psi):  # Psi[0..N], Psi[N+1]=0
N=len(Psi)-1
a=[0]*(N+1)
for s in range(N+1):
nextPsi = Psi[s+1] if s+1<=N else 0
a[s]=Psi[s]-(s+1)*nextPsi
return a
for N in (3,10,15,20):
print(N, kappa_inf(N))
```