Procrustes Computational Cost

Stephan to Marc and Sebastian

January 13, 2026

Abstract

A summary of the computational cost of our current Procrustes solution.

1 Summary Cost

Assume that the spatial dimension is M. We start with K_0 DFT bins and end after N iterations with K_N bins, with $K_n = 2K_{n-1}$ at the nth iteration, all of which are assumed to be powers of two. For Algorithm 1, we require:

step	cost
DFT	$M^2\,\{K_N \log_2 K_N - (K_N - K_0)\}$
SVD	$\mathcal{O}\{M^3\}K_N$
associations	$6M^3(K_N - K_0)$
per allpass	$\mathcal{O}\{\frac{8}{7}\{(2K_N)^3 - (2K_0)^3\}\}$
IDFT	$(M + M^2)\,\{2K_N \log_2 2K_N - K_0 \log_2 K_0 - 2(K_0 - 2K_N)\}$
overall*	$\mathcal{O}\{6M^3\}K_N + \mathcal{O}\{M^2 2K_N \log_2 2K_N\} + \mu\mathcal{O}\{\frac{8}{7}(2K_N)^3\}$

* The value $\mu \in \mathbb{Z}$ represents the number of singular values that require an allpass design, with $0 \leq \mu \leq M$.

Since typically the analytic SVD yields factors that are Laurent series, we have $K_N \gg M$, and the cost is — in the absence of any allpass filter design — dominated by the SVD calculations, the bin associations, and the reconstruction. If an allpass filter has to be designed, it usually is the most expensive aspect of Algorithm 1.

2 Iteration for DFT Size

For the individual cost items, we have:

1. DFT calculation: at the nth iteration, we use the previously calculated K_{n-1} bins, and calculate additional K_{n-1} bins after modulation of the time domain coefficients with an FFT size of K_{n-1}:

$$C_{\mathrm{FFT},n} = K_{n-1} \log_2 K_{n-1} + K_{n-1} = K_0 2^{n-1}\{1 + \log_2(K_0 2^{n-1})\} = 2^{n-1}K_0(n + \log_2 K_0) \quad (1)$$

$$= 2^{n-1}K_0 \log_2 K_0 \;+\; n2^{n-1}K_0 \quad (2)$$

After N iterations:

$$C_{\text{FFT}} = K_0 \log_2 K_0 + \sum_{n=1}^{N} C_{\text{FFT},n} \tag{3}$$
$$= K_0 \log_2 K_0 + \sum_{n=1}^{N} K_{n-1} \log_2 K_{n-1} + (K_N - K_0) \tag{4}$$
$$= K_0 \log_2 K_0 + \sum_{n=10}^{N} 2^{n-1} K_0 \log_2 2^{n-1} K_0 + (K_N - K_0) \tag{5}$$
$$= \{1 + \sum_{n=1}^{N} 2^{n-1}\} K_0 \log_2 K_0 + \sum_{n=0}^{N-1} n 2^n K_0 + (K_N - K_0) \tag{6}$$
$$= 2^N K_0 \log_2 K_0 + \{2 + (N-2)2^N\} K_0 + (K_N - K_0) \tag{7}$$

which exploits $\sum_{n=0}^{N} n2^n = 2 + (N-1)2^{N+1}$ (can be shown by induction). Further,

$$C_{\text{FFT}} = 2^N K_0 (\log_2 K_0 + N) + 2(1 - 2^N) K_0 + (K_N - K_0) \tag{8}$$
$$=2^N K_0 (\log_2 (2^N K_0) - 2(K_N - K_0) + (K_N - K_0) \tag{9}$$
$$= K_N \log_2 K_N - (K_N - K_0) \tag{10}$$

which is hence slightly cheaper than a K_N-point FFT. This is somewhat surprising, as the cost of an FFT partition in this way seems cheaper than a standard K_N-pont DFT, but it generalises if the initial and final sizes $K_N - K_0$ are identical.

Thus, overall, we need $M^2 C_{\text{FFT}}$ multiply-accumulate operations for the DFT of $\mathbf{A}[n]$.

2. Bin-wise SVD evaluation. We only need to evaluate an SVD in every bin once, hence

$$C_{\text{SVD}} = \mathcal{O}\{M^3\} K_N \tag{11}$$

3. evaluate $M \times M$ matrices to determine associations and sign changes: eqns (52), (54), (55), (58), and (59). At the nth iteration: multiplying out three matrix products of $M \times M$ matrices across every frequency bin.

$$C_{\text{assoc.},n} = 3M^3 K_n \tag{12}$$

In total, using the geometric series:

$$C_{\text{assoc.}} = 3M^3 \sum_{n=1}^{N} K_n = M^3 K_N \sum_{n=0}^{N-1} 2^{-n} \tag{13}$$
$$= 3M^3 K_N \frac{1 - (\frac{1}{2})^N}{1 - \frac{1}{2}} = 6M^3 (K_N - K_0) \tag{14}$$

4. is an allpass is to be designed, we may require anything between 0 and M such designs. If the nth iteration, assume that $L_\Omega = K_n$. Then for the inversion in (45), we have

$$C_{\text{allpass},n} = \mathcal{O}\{(2K_n)^3\} = \mathcal{O}\{8K_n\} \; . \tag{15}$$

Overall:

$$C_{\text{allpass}} = \mathcal{O}\{(2K_0)^3 + \ldots + (2K_N)^3\} \tag{16}$$
$$= \mathcal{O}\{(2 \cdot 2^{-N} K_N)^3 + \ldots + (2K_N)^3\} \tag{17}$$
$$= \mathcal{O}\{(2K_N)^3 \frac{1 - (\frac{1}{8})^{N+1}}{1 - \frac{1}{8}}\} \tag{18}$$
$$= \mathcal{O}\{\frac{8}{7}\{(2K_N)^3 - (2K_0)^3\}\} \tag{19}$$

5. reconstruction of singular values, and of singular vectors; per iteration (an extra small overhead is required if there bins that have been shifted — this does not occur very often, and hence is discounted here).

$$C_{\mathrm{recon.},n} = (M + M^2) K_n \log_2 K_n \ . \tag{20}$$

Overall:

$$\begin{aligned}
C_{\mathrm{recon.}} &= (M + M^2) \sum_{n=1}^{N} K_n \log_2 K_n && (21)\\
&= (M + M^2) \sum_{n=1}^{N} 2^n K_0 (\log_2 K_0 + n) && (22)\\
&= (M + M^2) \left\{ (2^{N+1} - 1) K_0 \log_2 K_0 + [2 + (N-1) 2^{N+1}] K_0 \right\} && (23)\\
&= (M + M^2) \left\{ 2^{N+1} K_0 \log_2 K_0 - K_0 \log_2 K_0 + 2K_0 + (N + 1 - 2) 2^{N+1} K_0 \right\} && (24)\\
&= (M + M^2) \left\{ 2^{N+1} K_0 [\log_2 K_0 + (N+1)] - K_0 \log_2 K_0 + 2K_0 - 2 \cdot 2^{N+1} K_0 \right\} && (25)\\
&= (M + M^2) \{ 2K_N \log_2 2K_N - K_0 \log_2 K_0 - 2(K_0 - 2K_N) \} && (26)
\end{aligned}$$

References