Pattern-Based Matrix Analysis Under Permutations

Andrei Ermakov
42 NL
ermakovhse@gmail.com

Abstract

Low-rank approximation captures one important notion of matrix simplicity: a matrix is simple if it is well described by a low-dimensional linear model. This paper studies a complementary notion. We consider matrices that become structurally simple only after permuting rows and columns, so that their mass concentrates in a prescribed geometric pattern such as a band, block-diagonal, arrowhead, or related form. We introduce a pattern distance that measures how close a matrix is to a given pattern after optimal reordering, giving a common framework for structural problems such as latent ordering, community detection, and bandwidth minimization. The problem is NP-hard in general; we derive an alternating algorithm based on Linear Assignment subproblems. We further introduce a pattern-plus-low-rank decomposition that combines permutation-structured fitting with a low-rank residual, whose permutation step is pattern fitting on the residual; its gains across pattern families persist at a matched parameter budget and are substantially reduced under a density-matched random-mask control. We evaluate the framework on synthetic recovery tasks, real correlation and graph adjacency matrices, and transformer attention heads, showing that a single universal algorithm recovers qualitatively different forms of matrix structure.

1 Introduction

A basic question in matrix analysis is what structure a matrix has. The standard answer is low-rank: the SVD gives the best rank-k approximation and reveals latent linear factors. But many matrices are simple in another sense: after reordering rows and columns they show an interpretable geometric form, blocks, bands, chains or hierarchies. Seriation [Atkins et al., 1998, Fogel et al., 2013], community detection [Newman, 2006, Holland et al., 1983] and bandwidth minimization [Cuthill and McKee, 1969, George, 1973] are usually studied separately yet share this intuition. Goldengorin and Krushinsky [2017] make it precise: many combinatorial problems (LAP, TSP, linear ordering, matching, cut) reduce to finding a permutation that aligns a matrix with a geometric *pattern*, and fixing one permutation reduces the remaining search to a Linear Assignment Problem.

This paper turns that combinatorial idea into a matrix-analysis tool. A binary *pattern mask* M specifies the target geometry; the *pattern distance* $d_M(A)$ is the off-pattern mass after the best row/column permutations. This viewpoint, which we call *permutation-structural simplicity*, complements low-rank approximation: a matrix may have clear geometric organization after reordering even when no low-rank model fits, and vice versa; the target geometry, the discrepancy measure and the search over permutations become independent ingredients of one formalism.

1.1 Contributions

1. **Pattern-based framework.** We define the *pattern distance* d_M, the off-pattern Frobenius mass after optimal reordering, and use it to formalize permutation-structural simplicity.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

2. **Alternating LAP algorithm.** An alternating procedure built on the auxiliary-matrix construction of Goldengorin and Krushinsky [2017] performs exact one-sided updates of d_M in $O(n^3)$, applicable to any binary mask without changes to the solver.
3. **Pattern-plus-low-rank (PatLR) decomposition.** A hybrid model combining permutation-structured fitting with a low-rank residual, with exact closed-form updates for both parts and a monotone alternating algorithm.
4. **Adaptive pattern selection.** When the pattern is unknown, a simulated-annealing search jointly optimizes the pattern and the permutation against a density-normalized score.
5. **Experimental validation.** On synthetic and real data the same algorithm recovers qualitatively different structures, scored by the metrics of latent ordering, consecutive ones and block recovery against the methods developed for them; on the attention heads of three transformers the same score gives labels with a shared depth profile.

1.2 Related Work

Structure after reordering. Several literatures study matrices that become simple once rows and columns are permuted, each around one geometry. Seriation seeks a Robinson order by spectral [Robinson, 1951, Atkins et al., 1998] or convex [Fogel et al., 2013] methods, with the consecutive-ones property [Booth and Lueker, 1976, Kendall, 1971] and circular seriation [Recanati et al., 2018] as its interval and cyclic forms. Community detection seeks block-diagonal structure, by modularity and spectral methods [Girvan and Newman, 2002, von Luxburg, 2007] or under the stochastic block model, where belief propagation and its spectral linearizations are optimal near the detectability threshold [Holland et al., 1983, Decelle et al., 2011, Krzakala et al., 2013, Saade et al., 2014]. Bandwidth and profile reduction seek near-diagonal sparsity [Cuthill and McKee, 1969, Gibbs et al., 1976, George, 1973]; nestedness [Brualdi and Sanderson, 1999], bordered block-diagonal forms [Aykanat et al., 2004], chordal sparsity [Vandenberghe and Andersen, 2015], core–periphery structure [Rombach et al., 2014] and biclustering [Hartigan, 1972] are further such geometries (Table 2). Robust PCA separates a low-rank part from a sparse one [Candès et al., 2011] with no permutation; GoDec [Zhou and Tao, 2011] alternates low-rank and sparse approximation, whereas PatLR restricts the support of the second component to a geometric mask under unknown permutations.

A common optimization core. Goldengorin and Krushinsky [2017] introduced patterns as a description of the input matrices of combinatorial optimization problems. For a fixed pattern the alignment is a Koopmans–Beckmann quadratic assignment problem [Koopmans and Beckmann, 1957, Lawler, 1963, Çela, 1998], with two permutations a bilinear assignment problem [Ćustić et al., 2017], for which alternating LAP heuristics are studied by Sokol et al. [2020]. Generic solvers are the FAQ heuristic [Vogelstein et al., 2015] and Birkhoff-polytope relaxations [Fogel et al., 2013, Cuturi, 2013, Mena et al., 2018]. What the single-geometry literatures lack, and this paper adds, is the pattern as the interface, a joint pattern-plus-low-rank model, and the application to attention heads.

2 Theory of Patterns

A *pattern* is a target structure (a band, block-diagonal, arrowhead, etc.), encoded by a binary *mask* $M \in \{0,1\}^{n\times n}$ marking the positions it allows; we use "pattern" for the structural hypothesis and "mask" for the matrix M. A permutation $\pi \in \mathcal{S}_n$ maps an original index to its new position, $(P_\pi)_{ij} = \mathbf{1}\{i = \pi(j)\}$, so the (π,σ)-reordered matrix $A^{\pi,\sigma} := P_\pi A P_\sigma^\top$ has entries $A_{\pi^{-1}(i),\sigma^{-1}(j)}$; $\Pi_M(X) := M \odot X$ is the masking operator. For fixed row and column permutations the *off-pattern mass* is

$$d_M(A;\pi,\sigma) \;:=\; \left\|(I-\Pi_M)(A^{\pi,\sigma})\right\|_F^2 \;=\; \sum_{(i,j):\, M_{ij}=0} \left(A^{\pi,\sigma}_{ij}\right)^2. \tag{1}$$

Definition 1 (Pattern distance)**.** *The pattern distance of $A \in \mathbb{R}^{n\times n}$ from the mask $M \in \{0,1\}^{n\times n}$ is the smallest off-pattern mass over row and column permutations,*

$$d_M(A) \;:=\; \min_{\pi,\sigma\in\mathcal{S}_n} d_M(A;\pi,\sigma). \tag{2}$$

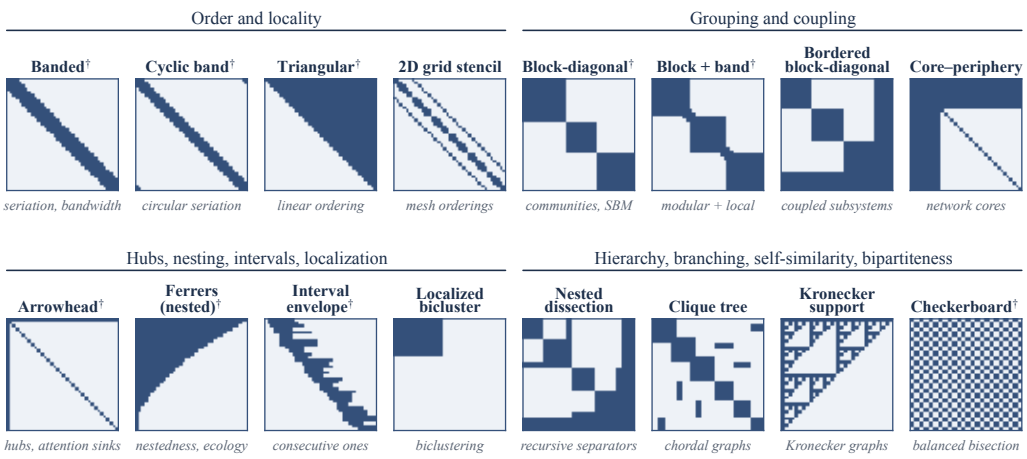

Figure 1: Gallery of patterns, one representative per structural hypothesis (each cell is the binary mask M, blue $= 1$), grouped by the kind of structure they encode: order, grouping, nesting, separability, branching, self-similarity. Daggers mark the families used in our experiments; Table 2 gives the classical problem behind each pattern. Masks are shown up to the permutations the framework optimizes over (an anti-diagonal band equals a band under column reversal). Any binary mask is admissible: M encodes the structure hypothesized for an application or is selected from a vocabulary.

For $A \neq 0$ the *explained pattern variance* $\nu_M(A) := 1 - d_M(A)/\|A\|_F^2 \in [0,1]$ is the fraction of matrix energy placed inside the pattern after optimal reordering. For symmetric matrices and for the attention matrices we also minimize over a common permutation, $d_M^{\mathrm{com}}(A) := \min_\pi d_M(A;\pi,\pi)$ (the case $\sigma = \pi$), writing $d_M(A;\pi) := d_M(A;\pi,\pi)$ at a fixed π; the restriction shrinks the feasible set, so $d_M^{\mathrm{com}}(A) \geq d_M(A)$. The mask density is $\rho_M := |M|_1/n^2$ ($|M|_1$ the number of ones). $d_M(A) = 0$ iff A fits inside M after some reordering, and under independent uniform row and column permutations a fraction ρ_M of the energy lies inside the mask in expectation (Appendix B.2).

Example (two notions of matrix simplicity). Pattern simplicity and low rank are independent: neither bounds the other. Let $A = PS_0P^\top$ with S_0 k-block-diagonal, constant blocks of size $b = n/k$ and entries μ. The k-block mask fits A exactly, $d_M(A) = 0$, while A has k equal singular values μb, so its best rank-r approximation has squared Frobenius error $(k-r)\,b^2\mu^2$ for every $r < k$: what the mask describes by its support, low rank describes only at rank k. Conversely a generic rank-one $uu^\top$, the simplest matrix for the SVD, fits no mask that leaves an entry out, $d_M(uu^\top) > 0$. A matrix can thus be simple in one sense and not in the other, which is why Section 4 fits both parts jointly instead of choosing one.

Hardness.

Theorem 1 (NP-hardness of pattern distance)**.** *Computing $d_M(A)$ exactly is NP-hard for the 2-block-diagonal mask, for independent (π,σ) as well as under $\sigma = \pi$.*

Proof sketch. Reduction from MINIMUM BISECTION [Garey and Johnson, 1979] through the adjacency matrix of the graph with a heavy diagonal, which forces the row and column bisections to coincide; the band of width 1 under $\sigma = \pi$ is NP-hard too, as a maximum-weight Hamiltonian path (Appendix B.5). Hardness is mask-specific: the arrowhead under $\sigma = \pi$ and the diagonal and anti-diagonal masks with two permutations are solved exactly (Appendix C.3).

Permutation ambiguities. An exact ordering need not be unique. If a noiseless signal is nonzero at every position allowed by the mask and zero elsewhere, all zero-error orderings differ only by symmetries of that mask (Appendix B.6). Under a common row/column permutation these include the reversal of a banded order and arbitrary reorderings within blocks and among equal-sized blocks; for an arrowhead mask with $n \geq 3$ the hub is identifiable, but the remaining indices have no identifiable order.

Algorithm 1 Alternating LAP

Require: $A \in \mathbb{R}^{n\times n}$, mask M, iterations T, restarts R

for $r = 1, \dots, R$ **do**
  $\pi \leftarrow$ Fiedler ordering if r=1, else random permutation; $\sigma \leftarrow \pi$
  for $t = 1, \dots, T$ **do**
    $Q_{pc} \leftarrow \sum_i (1-M_{ip})\,(P_\pi A)^2_{ic}$; $\sigma \leftarrow \mathrm{LAP}(Q)$
    $Q^{\mathrm{row}}_{pi} \leftarrow \sum_j (1-M_{pj})\,(AP_\sigma^\top)^2_{ij}$; $\pi \leftarrow \mathrm{LAP}(Q^{\mathrm{row}})$
  end for
  Update (π^*, σ^*) if $F(\pi, \sigma)$ improved (Proposition 1)
end for
if a common permutation is required: $\hat\pi \leftarrow$ best of π^*, σ^* and the start under $d_M(A; \pi, \pi)$; EXCHANGE: swap two positions while a swap lowers $d_M(A; \hat\pi, \hat\pi)$
return best (π^*, σ^*), or $\hat\pi$

3 Alternating LAP

Computing $d_M(A)$ is a search over two permutations at once: since permutations preserve $\|\cdot\|_F$, $d_M(A) = \|A\|_F^2 - \max_{\pi,\sigma} \sum_{i,j} M_{ij}\, A^2_{\pi^{-1}(i),\sigma^{-1}(j)}$, a quadratic assignment objective (NP-hard, Theorem 1). Building on the alternating LAP schemes of Goldengorin and Krushinsky [2017] and Sokol et al. [2020], we derive the assignment updates for the pattern-distance objective.

Lemma 1 (Column-separability). *Fix π_r and let $B = P_{\pi_r} A$. With $Q_{p,c} := \sum_i (1 - M_{i,p})\, B^2_{i,c}$,*

$$\min_{\sigma\in\mathcal{S}_n} \left\|(I - \Pi_M)(BP_\sigma^\top)\right\|_F^2 \;=\; \min_{\sigma\in\mathcal{S}_n} \sum_c Q_{\sigma(c),c}, \tag{3}$$

a Linear Assignment Problem whose solution $\mathrm{LAP}(Q)$ assigns each original column c its position $\sigma(c)$. The row case is symmetric (Appendix B).

Here Q is the auxiliary cost matrix of Goldengorin and Krushinsky [2017]: $Q_{p,c}$ is the off-pattern cost of placing original column c at position p under the fixed row order; since these costs do not depend on where the other columns go, choosing the column permutation is exactly a linear assignment problem. Alternating the two one-sided problems gives Algorithm 1: from a spectral start (the Fiedler order [Barnard et al., 1995] for bands, spectral clustering [von Luxburg, 2007] for blocks), fix the rows and solve a column LAP, fix the columns and solve a row LAP, and repeat; random restarts guard against poor local minima. A step costs $O(n^3)$ (auxiliary matrix and Hungarian solver), a run $O(RTn^3)$ for R restarts of T iterations. Below, $\textsc{AltLAP}(A, M, T;\, (\pi_0, \sigma_0))$ denotes T iterations of Algorithm 1 from (π_0, σ_0) (spectral start if the pair is omitted).

Proposition 1 (Monotone convergence). *Each LAP sub-step is a global minimizer of the one-sided objective (Lemma 1), so the iterates satisfy $F(\pi^{(t+1)}, \sigma^{(t+1)}) \le F(\pi^{(t)}, \sigma^{(t)})$ with $F(\pi, \sigma) = \|(I - \Pi_M)(P_\pi A P_\sigma^\top)\|_F^2$, and the F-values stabilize after finitely many steps; if ties in the LAP are broken in favour of the current permutation, the iterates themselves stop at a coordinate-wise local minimum. For a common permutation the exchange phase strictly decreases $d_M(A; \pi, \pi)$ and stops at an order that no pairwise exchange improves.*

Common permutation. For $\sigma = \pi$ the two-sided iterate is rounded to one permutation and finished by an exchange descent on $d_M(A; \pi, \pi)$: the objective change of an exchange has a closed form, one matrix product scores all n^2 exchanges, and improving ones are applied until none remains (Appendix D.1). The LAP steps move whole rows and columns and stall on the ties of a $\{0, 1\}$ cost matrix; the exchanges move one pair at a time and complete the descent (Appendix A.2).

Variants. The lemma uses only that the cost is a sum of per-entry weights, so Algorithm 1 runs on any separable score $\sum_{ij} (1 - M_{ij})\, W_{\pi^{-1}(i),\sigma^{-1}(j)}$: $W = A \odot A$ is the pattern distance; $W = A$ (ALTLAP-lin) is the likelihood criterion for a mean shift rather than an energy contrast (Section 5.2).

Algorithm 2 PATLR: Safeguarded Pattern + Low-Rank Decomposition

Require: $A \in \mathbb{R}^{n\times n}$, mask M, rank k, outer iterations T, LAP iterations T_{LAP}

$(\pi,\sigma) \leftarrow \mathrm{ALTLAP}(A, M, T_{\mathrm{LAP}})$; $\quad X \leftarrow 0$; $\quad S \leftarrow \Pi_M(P_\pi A P_\sigma^\top)$

for $t = 1,\dots,T$ **do**

  $B \leftarrow P_\pi(A-X)P_\sigma^\top$; $\quad S \leftarrow \Pi_M(B)$

  $X \leftarrow \mathrm{SVD}_k(A - P_\pi^\top S P_\sigma)$

  $(\tilde\pi,\tilde\sigma) \leftarrow \mathrm{ALTLAP}(A-X,\ M,\ T_{\mathrm{LAP}};\ (\pi,\sigma))$

  if $d_M(A-X;\tilde\pi,\tilde\sigma) < d_M(A-X;\pi,\sigma)$ **then** $(\pi,\sigma) \leftarrow (\tilde\pi,\tilde\sigma)$

  $S \leftarrow \Pi_M\big(P_\pi(A-X)P_\sigma^\top\big)$ $\quad$ {re-synchronize S with the accepted (π,σ)}

end for

return $(\pi,\sigma),\ S,\ X$

4 Pattern + Low-Rank Decomposition

Formulation. The **PatLR decomposition** approximates $P_\pi A P_\sigma^\top$ by a pattern-supported component S ($S_{ij} = 0$ when $M_{ij} = 0$) and a low-rank component $UV^\top$ of rank $\le k$. Writing $X = P_\pi^\top UV^\top P_\sigma$ for the low-rank part in the original coordinates, the objective is

$$\inf_{\pi,\sigma,S,X} \big\|P_\pi(A-X)P_\sigma^T - S\big\|_F^2 \quad \text{s.t.} \quad S\odot(1-M)=0,\ \ \mathrm{rank}(X)\le k. \tag{4}$$

For fixed (π,σ,X) the optimal S is the mask projection of the permuted residual, and substituting it back gives

$$\min_{S:\,S\odot(1-M)=0} \big\|P_\pi(A-X)P_\sigma^T - S\big\|_F^2 \ = \ d_M(A-X;\pi,\sigma), \tag{5}$$

so PatLR looks for a low-rank component after whose removal the residual fits the pattern. Algorithm 2 alternates the two closed-form updates, S by mask projection and X by the rank-k SVD of A minus the pattern part, with a *safeguarded* permutation update: $\mathrm{ALTLAP}(A-X, M, T_{\mathrm{LAP}};\ (\pi,\sigma))$ on the residual from the current pair is accepted only if it lowers $d_M(A-X;\cdot)$, and S is then re-synchronized. For a common permutation the update takes the common-permutation output of Algorithm 1, sets $\sigma = \pi$ and compares $d_M(A-X;\pi,\pi)$; Proposition 2 is unchanged. The infimum need not be attained and the split between S and X need not be unique: PatLR is a structured approximation model, scored by its reconstruction. Since $\pi = \sigma = \mathrm{id}$ with $S = 0$ is feasible, the infimum never exceeds the rank-k SVD error, and the example of Section 2 shows that the gap can be large.

Proposition 2 (Monotone convergence of PATLR). *For $F(\pi,\sigma,S,X) = \|P_\pi(A-X)P_\sigma^\top - S\|_F^2$ subject to $S\odot(1-M)=0$ and $\mathrm{rank}(X)\le k$, Algorithm 2 yields a non-increasing sequence of F-values, which therefore converges (Appendix C.1). Because X is continuous we do not claim finite termination or convergence of the iterates (S,X) themselves; the algorithm stops at a tolerance ε on the decrease of F or a capped iteration count.*

4.1 Adaptive Pattern Selection

Pattern M is rarely known a priori. We select it jointly with the permutations from a fixed vocabulary $\mathcal{V} = \{M_1,\dots,M_K\}$ by the *density-normalized score* (for $A \neq 0$, $\rho_M < 1$)

$$\tilde s_M(A;\pi,\sigma) := \frac{d_M(A;\pi,\sigma)}{\|A\|_F^2\cdot(1-\rho_M)^{3/2}}, \qquad \tilde s_M(A) := \min_{\pi,\sigma} \tilde s_M(A;\pi,\sigma). \tag{6}$$

The denominator is a reference against independent uniform row and column permutations, which place a fraction ρ_M of the energy inside the mask in expectation (Section 2); its superlinear exponent penalizes dense masks beyond this coverage advantage. Under a common permutation the random-permutation reference also depends on the split of the energy between the diagonal and the off-diagonal entries (Appendix B.2); the normalization is used as stated in both settings. Without noise a mask and every mask containing it score 0 alike, so the identified geometry need not be unique (Appendix C.2). The default vocabulary has nine masks, blocks $k \in \{2,3,4,6\}$, bands $w \in \{1,2,4\}$, arrowhead and block+band (k=3, w=2); any binary mask can be added.

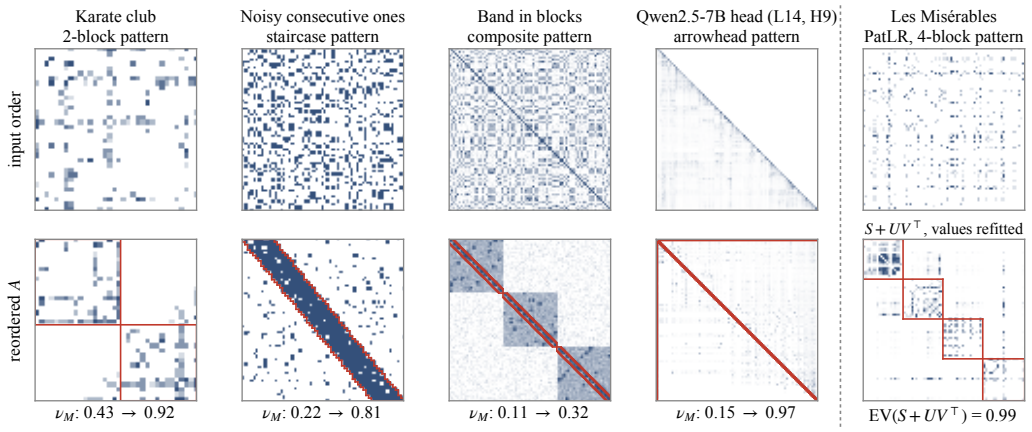

Figure 2: Outputs of the framework on one input of each kind (not a comparison). Top: the input in its given order; bottom: the order found by Alternating LAP for the named pattern (mask outlined in red), with the in-mask energy fraction $1 - d_M(A;\pi)/\|A\|_F^2$ before and after. The consecutive-ones column uses two permutations; the attention head is at $n{=}128$ on a square-root colour scale; the last column is the PatLR reconstruction $S + UV^\top$ (rank 3) in the found order, with its explained variance.

Adaptive PatLR. Pattern and permutation are searched jointly by simulated annealing over $\mathcal{V}$ with Metropolis acceptance on $\tilde{s}_M$ (Algorithm 3, Appendix D.2). The current (π, σ) is a warm start for *any* proposed pattern: good permutations for similar patterns are correlated, so a candidate $M_{j'}$ is evaluated with $\ell{=}3$ LAP steps instead of a full AltLAP run (exchange phase and PatLR only on the selected mask); with $N{=}80$ proposals the total work is $O(N(1{+}\ell)n^3)$ independently of K, in practice the cost of one PatLR call (Appendix E.8).

5 Experiments

Together, the experiments test whether one solver recovers different geometric structures, whether these structures complement low-rank approximation, and whether they can be selected and interpreted on real data. Unless stated otherwise we use a common row/column permutation ($\sigma{=}\pi$). We report our objective where the planted order is known and, where a task has its own literature, its metric alongside the methods developed for it. Comparisons are paired over 20 seeds with 95% bootstrap intervals; Figure 2 illustrates the outputs; setups, baselines and full grids are in Appendix E.

5.1 Ordering Structures

Banded recovery (our objective). We first ask whether the generic solver optimizes the pattern distance as well as a general QAP solver and classical ordering heuristics do. On randomly permuted banded targets in a Gaussian and a sparse-graph regime, Table 1 reports the error reduction (ER, the fraction of the input's off-pattern mass removed by the found order) of Alternating LAP next to reverse Cuthill–McKee [Cuthill and McKee, 1969] (RCM), Fiedler ordering plus greedy 2-SUM swaps (F+greedy) and QAP-FAQ [Vogelstein et al., 2015]. Alternating LAP gives the largest reduction on the Gaussian inputs at every n and on the sparse graphs at $w{=}5$; FAQ is ahead by 0.04–0.06 on the sparse $w{=}2$ graphs, where the ties of a $\{0,1\}$ cost matrix stall the LAP steps and the exchange phase carries the fit (Appendix E.1). RCM and GPS reduce the bandwidth, a criterion that saturates on noisy inputs, rather than the off-band mass (Appendix E.2).

Latent ordering (Figure 3a). A lower pattern distance need not mean a more accurate order, so the remaining tasks are scored by their own metrics. Planted Robinson matrices, scrambled and noised, scored by Kendall $|\tau|$ against the planted order, next to spectral seriation [Atkins et al., 1998] and the convex relaxation of Fogel et al. [2013]. Over 24 configurations one Alternating LAP run from the spectral order, with the band width set to the generator's correlation length, improves on spectral seriation in 18 cells and matches or exceeds the convex relaxation in 19, the gain growing with noise

Table 1: Error reduction (ER; higher = better) on banded patterns, 20 seeds. RCM (via KNN sparsification on dense inputs) and F+greedy (Fiedler order plus greedy 2-SUM swaps) optimize other criteria; FAQ (three starts) and AltLAP (five restarts) both minimize d_M.

	Gauss w=2				**Gauss** w=5				**Sparse** w=2				**Sparse** w=5			
	other		d_M		other		d_M		other		d_M		other		d_M	
n	RCM	F+gr.	FAQ	**AltLAP**	RCM	F+gr.	FAQ	**AltLAP**	RCM	F+gr.	FAQ	**AltLAP**	RCM	F+gr.	FAQ	**AltLAP**
30	0.03	0.84	0.60	**0.87**	0.07	**0.98**	0.84	**0.98**	0.23	0.45	**0.73**	0.67	0.41	0.78	0.88	**0.91**
100	0.05	0.37	0.51	**0.69**	0.09	0.72	0.66	**0.87**	0.06	0.09	**0.49**	0.45	0.16	0.29	0.66	**0.69**
200	0.03	0.14	0.46	**0.61**	0.08	0.33	0.61	**0.75**	0.02	0.03	**0.36**	0.30	0.09	0.10	0.52	**0.53**
500	0.03	0.03	0.40	**0.52**	0.06	0.08	0.57	**0.61**	0.00	0.00	**0.20**	0.15	0.03	0.02	**0.33**	**0.34**

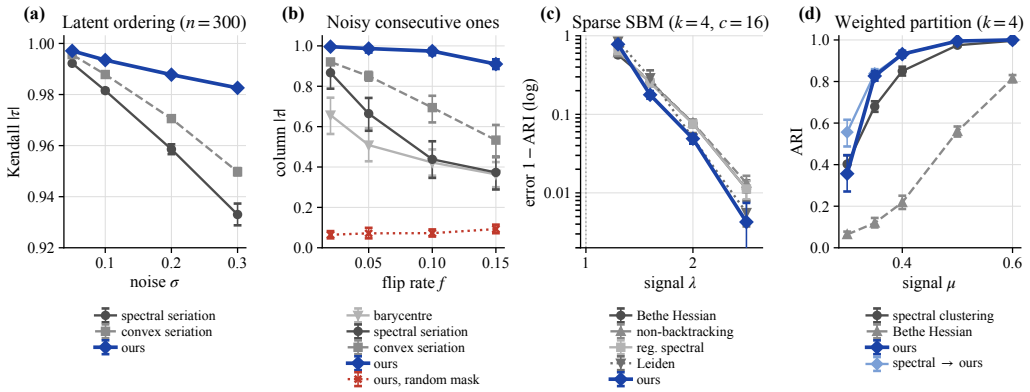

Figure 3: Four classical tasks scored by their own metrics (means, 95% bootstrap intervals, 20 seeds). "Ours" is Alternating LAP on the mask of the task: (a) band of width 25 from the spectral order; latent ordering, n=300, correlation length 25 (y-axis from 0.92). (b) staircase of interval length 12 from random restarts, with the same run on a random mask as control; noisy consecutive ones, 60×60. (c) equal-size block mask; sparse SBM, k=4, mean degree 16, drawn as the error $1 - \mathrm{ARI}$ on a log axis (lower is better; dotted: detectability threshold). (d) block mask on the linear score, from random restarts and from the spectral partition; weighted planted partition, k=4.

and with n; the exceptions are the cells at ℓ=25, n=100, where every method is near τ=1, and one cell at ℓ=6, n=100 where the relaxation leads by 0.04 (Appendix Table 12).

Consecutive ones (Figure 3b). The consecutive-ones property (every row an interval after a column order [Booth and Lueker, 1976, Kendall, 1971]) is the staircase band in a rectangular matrix, fitted with independent row and column permutations (Appendix B.1). On planted instances with flip noise, Alternating LAP from random restarts recovers the column order with a higher τ and fewer edits than convex seriation on every seed at every noise level, while a random or a block mask of the same density recovers almost nothing: the gain is the geometry of the mask.

Circular seriation and the sensitivity of these fits to the mask width and to the generator are in Appendix E.6, as are nested matrices, a boundary case: where the marginals already determine the order, the general optimizer does not improve over the specialized one.

5.2 Block Structures

We turn from order to grouping, where the block-mask objective is the likelihood criterion of the planted models; on the weighted matrices the solver runs on the linear score (AltLAP-lin, Section 3).

Sparse and weighted block models (Figure 3c,d). On the sparse SBM at the detectability scale (λ=1 the threshold [Decelle et al., 2011]) the block-mask objective is the maximum-likelihood criterion [Abbe, 2018], and two regimes appear: below $\lambda \approx 1.5$ a better criterion value does not give a better partition (Alternating LAP is below the Bethe Hessian in ARI, above it when started from its partition); above this regime Alternating LAP is competitive with the specialized methods, with the highest ARI in most cells, and at $c \geq 16$ from λ=1.6 on it matches or exceeds the spectral baselines

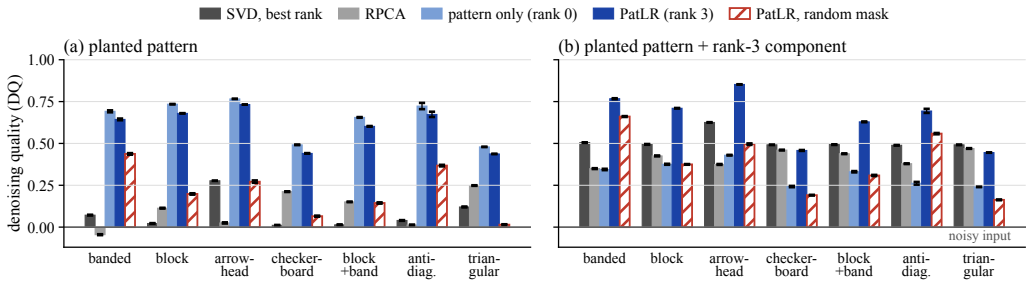

Figure 4: Denoising quality at SNR 1 (n=200, 20 seeds, 95% bootstrap intervals; the noisy input has DQ 0). (a) Planted pattern. (b) Planted pattern plus a rank-3 component of equal energy. SVD at the rank that maximizes DQ; RPCA; the pattern alone (PatLR at rank 0); PatLR at rank 3; and PatLR with a random mask of the same density as control.

(Appendix Table 18). On a weighted planted partition the linear score is the likelihood criterion and AltLAP-lin is above spectral clustering from μ=0.35 on (Appendix Table 19). The spectral baselines receive the true k; our mask also assumes equal block sizes, and under unequal planted sizes, sizes estimated from a spectral partition recover most of the gap (Appendix Table 20).

Composite target: a band inside blocks. The band *inside* the blocks (Figure 2, third column) tests the joint refinement of communities and their internal order: two structural hypotheses combined in one mask and refined by one criterion. On 16 planted configurations we compare the natural pipeline, spectral clustering then seriation inside each cluster, with the same pipeline followed by one Alternating LAP pass on the composite mask, with block boundaries re-estimated from the order. The pass raises the error reduction on the target in every configuration, from a median 45% of the planted order's value to 94%, keeps the blocks exact and improves the within-block order where the band is wide enough to determine it (w=3); similar ER gains hold with convex seriation inside the clusters. From a cold start the composite mask alone recovers the blocks only in part: the pipeline supplies the start and the mask the target (Appendix Table 23).

Labelled networks. On four networks with published communities no single panel method is best; PatLR with rank 1 recovers the karate-club split exactly (Appendix Table 21).

5.3 Pattern + Low-Rank Decomposition

The experiments above test whether permutations reveal structure; we now ask whether representing it explicitly improves reconstruction beyond low rank. We measure **denoising quality** (DQ) $= 1 - \|\hat{A} - A_{\text{true}}\|_F^2 / \|A_{\text{true}}\|_F^2$ of reconstructions $\hat{A}$ from $A_{\text{true}} + E$ (E symmetric Gaussian), against SVD and Robust PCA, on planted matrices whose family is known (PatLR receives its mask).

Planted structure at SNR 1 (Figure 4). Seven pattern families at n=200, each planted alone and together with a rank-3 component of equal energy, scrambled, with noise at SNR $= \|A_{\text{true}}\|_F^2 / \|E\|_F^2 = 1$, so that the noisy input has DQ 0. On the pattern alone the mask removes most of the noise and the low-rank estimates little; on pattern plus low rank no single component suffices and PatLR reconstructs the combined signal more accurately than either estimate alone on the masks of density ≤ 0.25, while on the two half-dense masks every estimate that copies the in-mask entries also copies their noise and PatLR ends 0.03–0.04 below SVD (bound in Appendix E.4). Two controls: a random mask of the same density, solver and rank stays 0.2–0.5 DQ below the planted mask (0.1–0.4 with the rank-3 component) but above the SVD level on the band and anti-diagonal masks, so part of the gain is mass that a free permutation packs into $|M|$ entries and the rest is the geometry; and at matched nominal parameter budgets PatLR has a higher mean DQ than SVD and spectral co-clustering on all seven families (Appendix Table 10).

Real data. Where no truth is planted, Figure 2 shows the decomposition of Les Misérables [Knuth, 1993] under a four-block template; Appendix Figure 8 follows it step by step on this network, on the karate club [Zachary, 1977] and on three feature-correlation matrices: the pattern part takes the community blocks, the band and the feature groups, and the low-rank part the hubs and the general correlation factor.

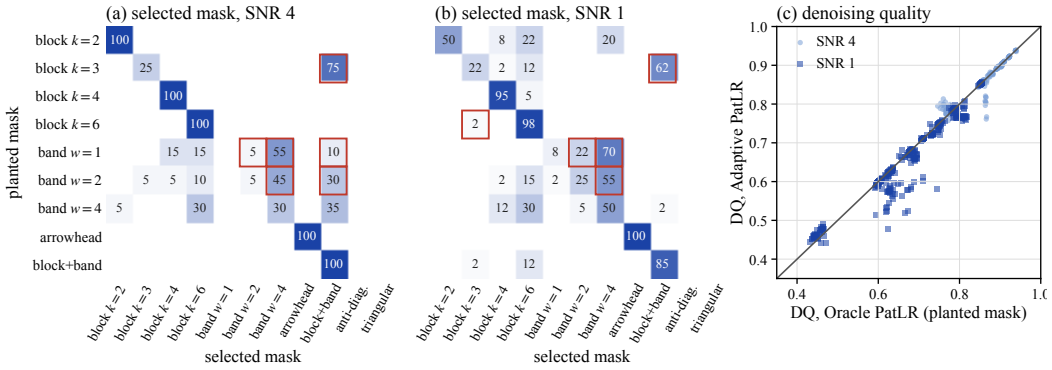

Figure 5: Adaptive pattern selection (n=200, 540 instances). (a, b) Mask selected by Adaptive PatLR (columns; the checkerboard is counted with block k=2) for each planted mask (rows) at SNR 4 and SNR 1, in % of instances over both generators; red frames: the selected mask contains the planted one (block+band covers block k=3 up to two rows). (c) DQ of Adaptive against Oracle PatLR per instance. Per-family numbers: Appendix Table 11.

5.4 Adaptive Pattern Selection

The experiments above give PatLR its mask; Algorithm 3 selects it jointly with the permutation, and the experiment asks how much reconstruction quality this costs and whether the planted geometry is identified. We plant each of the nine masks of the default vocabulary (n=200, with and without a rank-3 component, SNR 4 and 1; 540 instances) and compare Adaptive PatLR, run over these nine masks plus three gallery distractors, with Oracle PatLR (the planted mask), brute force over the same vocabulary and SVD at its best rank (Figure 5; Appendix E.5.1). *Quality* (Figure 5c): Adaptive PatLR reaches the oracle's denoising quality on essentially every instance (median ratio 1.00, 5th percentile 0.92), where SVD at its best rank averages less than half of the oracle's value at SNR 1. *Identification* (Figure 5a,b) varies by family: arrowhead, block+band and the block templates $k \geq 4$ are recovered reliably, whereas thin bands and block k=3 often select a broader or alternative mask (a wider band, a block template or block+band) within 0.07 of the oracle's DQ; good reconstruction therefore does not imply exact recovery of the geometry. The anti-diagonal and triangular distractors are never selected. Brute force over the vocabulary with the LAP phase gives the same reconstruction (within 0.01) and similar selections at a cost growing with $|\mathcal{V}|$; the search costs one PatLR call.

5.5 Attention Heads

We use attention matrices to test whether pattern labels are stable across prompts and context lengths, whether the reordering adds information beyond the native token order, in which the literature describes head types, and whether the layerwise profile of labels is shared across three models.

Setup. We score every head of GPT-2, Qwen2.5-0.5B and **Qwen2.5-7B** against a vocabulary of nine geometric masks (bands, blocks, arrowhead, block+band, diagonal), applied to the causal attention matrices without requiring the reordered matrix to stay causal, with the density-normalized score (6), on 20 WikiText-2 passages at $n \in \{128, 512\}$ tokens; the native token order is a candidate for every mask, and the label of a (head, prompt) pair is the mask with the lowest score (Appendix E.7).

Labels and layer organization. Arrowhead, the geometry of the attention sink [Xiao et al., 2024, Zhai et al., 2023], is the label of 69% / 67% / 80% of the (head, prompt) pairs of GPT-2 / Qwen2.5-0.5B / Qwen2.5-7B at n=512; block masks take most of the rest, thin bands and the diagonal a few percent. The label is a fit, not the best of a poor set: the selected mask's score is about an order of magnitude below its random-permutation level. By layer (Figure 6a), the first three to four layers of every model carry block labels and the later layers arrowhead; on Qwen2.5-0.5B the last two layers return to block-k2. Most block-labelled pairs are wide local heads in the native token order, labelled by the block mask containing their window (Appendix Figure 13).

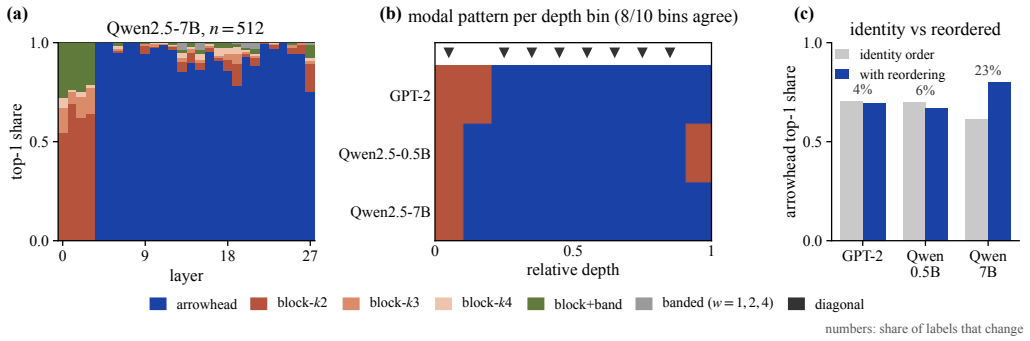

Figure 6: Attention heads at n=512, unified scoring. (a) Qwen2.5-7B: share of the top-1 label per layer over 20 prompts. (b) Modal label per relative-depth bin for the three models; markers show the bins where all three agree. (c) Arrowhead share under the native token order and with reordering, with the share of (head, prompt) labels that change (Appendix E.7.4).

Are the permutations needed? Scoring the same pairs in their native token order (Figure 6c) changes a few percent of the labels on GPT-2 and Qwen2.5-0.5B; on Qwen2.5-7B the arrowhead share rises from 0.61 to 0.80 and 23% of the labels change, because on a quarter of the passages the sink is not the first token, the native-order arrowhead share is below 2% there, and the reordering restores it (Appendix E.7.4): it buys a label independent of where the sink sits.

Stability. The mean agreement of a head's label with its modal label over the 20 prompts is 0.96–0.99 for arrowhead heads and 0.64–0.74 for the others. Between n=128 and n=512 the modal label of a head is unchanged for more than nine heads in ten. On a shared 10-bin relative-depth grid the per-layer modal label agrees across the three models in $8/10$ bins (Figure 6b): blocks in the first tenth and arrowhead thereafter (Appendix E.7.3).

6 Discussion and Limitations

Limitations. The dense implementation costs $O(n^3)$ per LAP step and per exchange round, which makes large-matrix and long-context analyses costly (Appendix E.8). The permutation search is heuristic and does not guarantee global optimality (Section 3). Adaptive selection operates within a supplied mask vocabulary: a selected label is the best fit among the candidates, not an identification of the generating geometry (Section 5.4). Our theoretical guarantees concern optimization, while recovery under noise is evaluated empirically. The attention study covers three decoder-only models at 128 and 512 tokens; other architectures and longer contexts remain future work.

7 Conclusion

We introduced *permutation-structural simplicity* as an axis of matrix structure complementary to low rank: the pattern distance relates a dozen classical reordering problems to one objective (Table 2), Alternating LAP is one solver for any binary mask, PatLR adds a low-rank residual, Adaptive PatLR selects the pattern from a vocabulary, and on attention heads the same score gives stable labels with one depth profile across three models.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers and the area chair for suggestions that led to the specialized baselines, the capacity controls, and the hardness statement. This work received no third-party funding. The authors declare no competing interests.

References

Emmanuel Abbe. Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research*, 18(177):1–86, 2018. URL http://jmlr.org/papers/v18/16-480.html.

Stefan Aeberhard and M. Forina. Wine. UCI Machine Learning Repository, 1992.

E. Alpaydin and C. Kaynak. Optical recognition of handwritten digits. UCI Machine Learning Repository, 1998.

Jonathan E. Atkins, Erik G. Boman, and Bruce Hendrickson. A spectral algorithm for seriation and the consecutive ones problem. *SIAM Journal on Computing*, 28(1):297–310, 1998. doi: 10.1137/S0097539795285771.

Cevdet Aykanat, Ali Pinar, and Ümit V. Çatalyürek. Permuting sparse rectangular matrices into block-diagonal form. *SIAM Journal on Scientific Computing*, 25(6):1860–1879, 2004.

Stephen T. Barnard, Alex Pothen, and Horst Simon. A spectral algorithm for envelope reduction of sparse matrices. *Numerical Linear Algebra with Applications*, 2(4):317–334, 1995. doi: 10.1002/nla.1680020402.

Annabell Berger and Berit Schreck. The isomorphic version of Brualdi's and Sanderson's nestedness. *Algorithms*, 10(3):74, 2017.

Dimitri P. Bertsekas. The auction algorithm: A distributed relaxation method for the assignment problem. *Annals of Operations Research*, 14(1):105–123, 1988. doi: 10.1007/BF02186476.

Kellogg S. Booth and George S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *Journal of Computer and System Sciences*, 13(3):335–379, 1976.

Richard A. Brualdi and James G. Sanderson. Nested species subsets, gaps, and discrepancy. *Oecologia*, 119(2):256–264, 1999.

Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):1–37, 2011. doi: 10.1145/1970392.1970395.

Eranda Çela. *The Quadratic Assignment Problem: Theory and Algorithms*. Kluwer Academic Publishers, 1998.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? An analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286. Association for Computational Linguistics, 2019. doi: 10.18653/v1/W19-4828.

Ante Ćustić, Vladyslav Sokol, Abraham P. Punnen, and Binay Bhattacharya. The bilinear assignment problem: complexity and polynomially solvable special cases. *Mathematical Programming*, 166:185–205, 2017.

E. Cuthill and J. McKee. Reducing the bandwidth of sparse symmetric matrices. In *Proceedings of the 24th ACM National Conference*, pages 157–172, 1969. doi: 10.1145/800195.805928.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Allison Davis, Burleigh B. Gardner, and Mary R. Gardner. *Deep South: A Social Anthropological Study of Caste and Class*. University of Chicago Press, Chicago, 1941.

Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84:066106, 2011.

Inderjit S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 269–274, 2001.

Fajwel Fogel, Rodolphe Jenatton, Francis Bach, and Alexandre d'Aspremont. Convex relaxations for permutation problems. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Linton C. Freeman. Finding social groups: A meta-analysis of the Southern Women data. In Ronald Breiger, Kathleen Carley, and Philippa Pattison, editors, *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers*, pages 39–97. National Academies Press, Washington, DC, 2003. doi: 10.17226/10735.

Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

Alan George. Nested dissection of a regular finite element mesh. *SIAM Journal on Numerical Analysis*, 10(2):345–363, 1973. doi: 10.1137/0710032.

Norman E. Gibbs, William G. Poole, and Paul K. Stockmeyer. An algorithm for reducing the bandwidth and profile of a sparse matrix. *SIAM Journal on Numerical Analysis*, 13(2):236–250, 1976.

Michelle Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002. doi: 10.1073/pnas.122653799.

Boris Goldengorin and Dmitry Krushinsky. Linear assignment problems in combinatorial optimization. In Sergiy Butenko, Panos M. Pardalos, and Volodymyr Shylo, editors, *Optimization Methods and Applications: In Honor of Ivan V. Sergienko's 80th Birthday*, volume 130 of *Springer Optimization and Its Applications*, pages 183–216. Springer, Cham, 2017. doi: 10.1007/978-3-319-68640-0_9.

Bruce Hajek, Yihong Wu, and Jiaming Xu. Submatrix localization via message passing. *Journal of Machine Learning Research*, 18(186):1–52, 2018.

John A. Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, 67(337):123–129, 1972.

Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: first steps. *Social Networks*, 5(2):109–137, 1983. doi: 10.1016/0378-8733(83)90021-7.

David G. Kendall. Seriation from abundance matrices. In F. R. Hodson, D. G. Kendall, and P. Tăutu, editors, *Mathematics in the Archaeological and Historical Sciences*, pages 215–252. Edinburgh University Press, 1971.

Donald E. Knuth. *The Stanford GraphBase: A Platform for Combinatorial Computing*. ACM Press / Addison-Wesley, New York, 1993.

Tjalling C. Koopmans and Martin Beckmann. Assignment problems and the location of economic activities. *Econometrica*, 25(1):53–76, 1957. doi: 10.2307/1907742.

Florent Krzakala, Cristopher Moore, Elchanan Mossel, Joe Neeman, Allan Sly, Lenka Zdeborová, and Pan Zhang. Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, 110(52):20935–20940, 2013.

Eugene L. Lawler. The quadratic assignment problem. *Management Science*, 9(4):586–599, 1963.

Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. Kronecker graphs: An approach to modeling networks. *Journal of Machine Learning Research*, 11:985–1042, 2010.

Ross M. McConnell. A certifying algorithm for the consecutive-ones property. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 768–777, 2004.

Gonzalo Mena, David Belanger, Scott Linderman, and Jasper Snoek. Learning latent permutations with Gumbel-Sinkhorn networks. In *International Conference on Learning Representations*, 2018.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006. doi: 10.1073/pnas.0601602103.

Jeff Pool and Chong Yu. Channel permutations for N:M sparsity. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Alex Pothen and Chin-Ju Fan. Computing the block triangular form of a sparse matrix. *ACM Transactions on Mathematical Software*, 16(4):303–324, 1990.

Tai Qin and Karl Rohe. Regularized spectral clustering under the degree-corrected stochastic blockmodel. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

Antoine Recanati, Thomas Kerdreux, and Alexandre d'Aspremont. Reconstructing latent orderings by spectral clustering. *arXiv preprint arXiv:1807.07122*, 2018.

W. S. Robinson. A method for chronologically ordering archaeological deposits. *American Antiquity*, 16(4):293–301, 1951. doi: 10.2307/276978.

M. Puck Rombach, Mason A. Porter, James H. Fowler, and Peter J. Mucha. Core-periphery structure in networks. *SIAM Journal on Applied Mathematics*, 74(1):167–190, 2014.

Alaa Saade, Florent Krzakala, and Lenka Zdeborová. Spectral clustering of graphs with the Bethe Hessian. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

James B. Saxe. Dynamic-programming algorithms for recognizing small-bandwidth graphs in polynomial time. *SIAM Journal on Algebraic and Discrete Methods*, 1(4):363–369, 1980.

Vladyslav Sokol, Ante Ćustić, Abraham P. Punnen, and Binay Bhattacharya. Bilinear assignment problem: Large neighborhoods and experimental analysis of algorithms. *INFORMS Journal on Computing*, 32(3):730–746, 2020. doi: 10.1287/ijoc.2019.0893.

Jared Tanner, Andrew Thompson, and Simon Vary. Matrix rigidity and the ill-posedness of robust PCA and matrix completion. *SIAM Journal on Mathematics of Data Science*, 1(3):537–554, 2019.

Vincent A. Traag, Ludo Waltman, and Nees Jan van Eck. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports*, 9:5233, 2019.

Lieven Vandenberghe and Martin S. Andersen. Chordal graphs and semidefinite optimization. *Foundations and Trends in Optimization*, 1(4):241–433, 2015.

Joshua T. Vogelstein, John M. Conroy, Vince Lyzinski, Louis J. Podrazik, Steven G. Kratzer, Eric T. Harley, Donniell E. Fishkind, R. Jacob Vogelstein, and Carey E. Priebe. Fast approximate quadratic programming for graph matching. *PLOS ONE*, 10(4):e0121002, 2015.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1580.

Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007. doi: 10.1007/s11222-007-9033-z.

William Wolberg, Olvi Mangasarian, Nick Street, and W. Street. Breast cancer Wisconsin (diagnostic). UCI Machine Learning Repository, 1993.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations (ICLR)*, 2024.

Wayne W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977. doi: 10.1086/jar.33.4.3629752.

Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yizhe Zhang, Jiatao Gu, and Joshua M. Susskind. Stabilizing transformer training by preventing attention entropy collapse. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *PMLR*, pages 40770–40803, 2023.

Tianyi Zhou and Dacheng Tao. GoDec: Randomized low-rank & sparse matrix decomposition in noisy case. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pages 33–40, 2011.

A Pattern Dictionary, Worked Example and Notation

A.1 Pattern dictionary

Table 2 lists the pattern families of Figure 1 together with the classical problems they encode; families used in our experiments are marked with a dagger, the others are included to delineate the scope of the formulation and are not analysed here. Table 3 relates the masks to the established problems and to the specialized methods used as baselines.

Table 2: Pattern dictionary: mask families, the structure they encode and the classical problem behind each (†: used in our experiments). The last column records only what this paper establishes about computing d_M (Theorem 1 and exact special cases); dashes: not analysed here.

Pattern M	Shape	Classical problem	d_M in this paper
Order and locality			
Diagonal†	δ_{ij}	Linear assignment (LAP)	one LAP§§
Banded (w)†	$\lvert i-j\rvert \le w$	Seriation, bandwidth min. [Atkins et al., 1998, Cuthill and McKee, 1969]	NP-hard at $w{=}1$, $\sigma{=}\pi$ (App. B.5)
Anti-diagonal†	$i+j=n+1$	Reverse ordering (diagonal under column reversal)	one LAP§§
Cyclic band†	$\min(\lvert i-j\rvert, n-\lvert i-j\rvert) \le w$	Circular seriation [Recanati et al., 2018]	–
Upper triangular†	$i \le j$	Linear ordering / feedback arc set	–¶
2D grid stencil	$\lVert u-v\rVert_1 \le 1$ on a grid	Mesh / PDE orderings [George, 1973]	–
Grouping and coupling			
Block-diagonal (k)†	k diagonal blocks	Community detection, SBM [Holland et al., 1983]	NP-hard, Thm. 1
Block + band†	blocks ∪ band	Modular networks with local structure	–
Band-in-block†	blocks ∩ band	Composite target (Sec. 5.2)	–
Bordered block-diagonal	blocks + coupling border	Decomposition for parallel LA / LP [Aykanat et al., 2004]	–
Core–periphery	$J_{cc}, J_{cp}, J_{pc}, 0_{pp}$	Core–periphery structure [Rombach et al., 2014]	–
Block-triangular	k blocks, upper part	Dulmage–Mendelsohn decomposition [Pothen and Fan, 1990]	–††
Hubs, nesting, intervals, localization			
Arrowhead†	row/col 0 + diagonal	Hub / sink structure	$O(n^2)$ under $\sigma{=}\pi$ (App. C.3)
Ferrers (nested)†	$\mathbf{1}[j \le r_i]$, r non-incr.	Nestedness [Brualdi and Sanderson, 1999, Berger and Schreck, 2017]	–♮
Interval envelope†	$\mathbf{1}[\ell_i \le j \le r_i]$	Consecutive ones, interval graphs [Booth and Lueker, 1976, McConnell, 2004]	–♮
Localized bicluster	$k \times \ell$ rectangle	Submatrix localization, biclustering [Hartigan, 1972, Hajek et al., 2018]	–
Hierarchy, branching, self-similarity, parity			
Nested dissection	recursive separators	Fill-reducing orderings [George, 1973]	–
Clique tree	overlapping cliques on a tree	Chordal sparsity, GGMs [Vandenberghe and Andersen, 2015]	–
Kronecker support	$K^{\otimes q}$	Kronecker graphs [Leskovec et al., 2010]	–
Checkerboard†	$i+j$ even	Balanced bisection (the 2-block mask under the even/odd relabeling)	as 2 blocks‡
Local + strided	band ∪ periodic columns	Sparse attention layouts [Child et al., 2019]	–

§§With two permutations any permutation-matrix mask is a single LAP (Lemma 1); under $\sigma = \pi$ the diagonal objective is constant and the anti-diagonal is a maximum-weight matching. The anti-diagonal *band* of Section 5.3 is the banded mask under column reversal. ¶Under $\sigma = \pi$ the off-pattern mass is the weight of the backward arcs, the linear-ordering objective (constant for symmetric A). ††Dulmage–Mendelsohn computes the finest block-triangular form in polynomial time; fitting a prescribed block profile with given sizes is a different problem. ♮Exact recognition of the noiseless structure is polynomial (Berger and Schreck, 2017; Booth and Lueker, 1976); our experiments concern the noisy fixed-profile versions (Appendix E.6.2). ‡The same-parity mask (ones where $i+j$ is even, including the diagonal) is the 2-block mask under the even/odd relabeling, so d_M coincides with the 2-block case; the opposite-parity mask (zero diagonal) would correspond to maximum bisection instead and is not used.

Table 3: Correspondence between pattern masks and established problems, with the specialized methods used as baselines in this paper.

Mask / setting	Classical problem	Specialized methods compared	Where
Fixed M, one permutation	Koopmans–Beckmann QAP	QAP-FAQ (as alternative solver)	§5.1
Fixed M, two permutations	Bilinear assignment, graph matching	Birkhoff relaxations (discussed)	§3
Banded, exact support	Bandwidth / profile reduction	RCM, Gibbs–Poole–Stockmeyer	§5.1
Banded / Robinson, noisy	Seriation, latent ordering	Spectral and convex seriation	§5.1
Staircase band (rectangular)	Consecutive-ones, interval seriation	Barycentre, spectral, convex seriation	§5.1
Cyclic band	Circular seriation	Circular spectral embedding	App. E.6.2
Block-diagonal, $\{0,1\}$	Community detection, SBM	Bethe Hessian, non-backtracking, regularized spectral, Leiden	§5.2
Block-diagonal, weighted	Planted partition (likelihood)	Spectral clustering	§5.2
Band inside blocks	Composite (no dedicated method)	Cluster-then-order pipeline	§5.2
Any M + low rank (PatLR)	Structured plus low-rank approximation	SVD at matched nominal budget, RPCA, spectral co-clustering	§5.3

A.2 A 6×6 example

Take $n = 6$, the band mask $M_{ij} = 1$ iff $|i - j| \le 1$, and the reference matrix $B_{ij} = \max(0,\, 5 - 2|i - j|)$. Scramble it by $p = (0, 2, 4, 1, 3, 5)$ (zero-based), i.e. $A_{ij} = B_{p_i, p_j}$:

B (reference)
$$\begin{pmatrix} 5 & 3 & 1 & 0 & 0 & 0 \\ 3 & 5 & 3 & 1 & 0 & 0 \\ 1 & 3 & 5 & 3 & 1 & 0 \\ 0 & 1 & 3 & 5 & 3 & 1 \\ 0 & 0 & 1 & 3 & 5 & 3 \\ 0 & 0 & 0 & 1 & 3 & 5 \end{pmatrix}$$

A (scrambled)
$$\begin{pmatrix} 5 & 1 & 0 & 3 & 0 & 0 \\ 1 & 5 & 1 & 3 & 3 & 0 \\ 0 & 1 & 5 & 0 & 3 & 3 \\ 3 & 3 & 0 & 5 & 1 & 0 \\ 0 & 3 & 3 & 1 & 5 & 1 \\ 0 & 0 & 3 & 0 & 1 & 5 \end{pmatrix}$$

B is concentrated near the diagonal but not exactly supported on the mask: its eight entries at $|i-j| = 2$ have magnitude one, so the optimum is $d_M(A) = d_M(B) = 8$, while the objective at the identity ordering is $\|(1 - M) \odot A\|_F^2 = 90$. *One step.* With rows fixed, placing column c at position q costs $Q_{q,c} = \sum_i (1 - M_{iq})\, A_{ic}^2$, the off-pattern mass this placement would create (Lemma 1):

M (band, $w = 1$)
$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Q (rows = positions q, columns = c)
$$\begin{pmatrix} 9 & 19 & 43 & 26 & 36 & 35 \\ 9 & 18 & 18 & 26 & 27 & 26 \\ 25 & 10 & 18 & 10 & 26 & 26 \\ 26 & 26 & 10 & 18 & 10 & 25 \\ 26 & 27 & 26 & 18 & 18 & 9 \\ 35 & 36 & 26 & 43 & 19 & 9 \end{pmatrix}$$

An optimal assignment on Q orders the columns $(0, 1, 3, 2, 4, 5)$ at cost 74, exactly the off-pattern mass after the move; the row step is symmetric. *Run.* The successive exact one-sided minimizations give $90 \to 74 \to 40 \to 8$, and both final orders equal $(0, 3, 1, 4, 2, 5) = p^{-1}$, so the reordered matrix is exactly B. Each step is the global optimum of its subproblem, so the objective never increases (Proposition 1); this does not imply global convergence in general, since other initializations or tied assignments can end at coordinatewise local minima, which is what the spectral initialization and the restarts address.

A.3 Notation and terminology

Table 4: Symbols and terms used throughout; standard notions (LAP, QAP, Fiedler vector, ARI, Kendall τ) are used in their usual sense.

Symbol / term	Meaning
$A^{\pi,\sigma} = P_\pi A P_\sigma^\top$	rows reordered by π, columns by σ; $(A^{\pi,\sigma})_{ij} = A_{\pi^{-1}(i),\sigma^{-1}(j)}$, a permutation mapping an original index to its position
$\Pi_M(X) = M \odot X$	masking operator; $(I - \Pi_M)(X)$ is the off-pattern part
$d_M(A; \pi, \sigma)$	off-pattern squared Frobenius mass at fixed permutations (1)
$d_M(A)$	its minimum over independent row and column permutations (2)
$d_M^{\mathrm{com}}(A)$, $d_M(A; \pi)$	minimum under a common permutation ($\sigma = \pi$); the value at a fixed π
$\nu_M(A)$	explained pattern variance $1 - d_M(A)/\|A\|_F^2$
$\rho_M = \|M\|_1/n^2$	mask density ($\|M\|_1$ the number of ones)
$\tilde{s}_M(A)$	density-normalized score $d_M(A)/\big(\|A\|_F^2(1-\rho_M)^{3/2}\big)$, equation (6)
S, $X = P_\pi^\top U V^\top P_\sigma$	PatLR pattern part (reordered frame) and low-rank part (original frame), Section 4
2-SUM	$\sum_{ij} \|A_{ij}\|(\pi(i) - \pi(j))^2$, the seriation objective whose spectral relaxation is the Fiedler order
ER	error reduction $1 - d_M(A; \hat{\pi})/d_M(A; \mathrm{id})$ of the found order over the input order
BE	band energy, the in-mask energy fraction after reordering
DQ	denoising quality $1 - \|\hat{A} - A_{\mathrm{true}}\|_F^2/\|A_{\mathrm{true}}\|_F^2$, against the clean matrix
EV	explained variance $1 - \|A - \hat{A}\|_F^2/\|A\|_F^2$, against the observed matrix
block+band (k, w)	*union* of k equal diagonal blocks and the band $\|i - j\| \le w$ (vocabulary, Table 10, attention)
band-in-block	*intersection*, a band inside each block: the composite target of Section 5.2 and Appendix E.6.4
anti-diagonal (w)	anti-diagonal band $\|i + j - (n-1)\| \le w$ (w=10 at n=200); Table 2 lists w=0
ALTLAP-lin	Alternating LAP on the separable score with $W = A$, the linear criterion (Section 3)

B Deferred Proofs

B.1 Basic identities and scope of the formulation

We use the conventions $A^{\pi,\sigma} := P_\pi A P_\sigma^\top$, $(A^{\pi,\sigma})_{ij} = A_{\pi^{-1}(i),\sigma^{-1}(j)}$; $\Pi_M(X) := M \odot X$ is the masking operator and $(I - \Pi_M)(X) = (1 - M) \odot X$ extracts the off-pattern part. For rectangular $A, M \in \mathbb{R}^{m \times n}$ the same definitions apply with $\pi \in S_m$, $\sigma \in S_n$ and $\rho_M = |M|_1/(mn)$; Lemma 1 and Algorithm 1 are unchanged, the two LAPs having sizes n and m. The restriction $\sigma = \pi$ gives the common-permutation distance $d_M^{\mathrm{com}}(A) = \min_\pi d_M(A; \pi)$ with $d_M(A; \pi) := \|(I - \Pi_M)(A^\pi)\|_F^2$; it is natural for symmetric $A = A^\top$, $M = M^\top$ and is also imposed on the asymmetric attention matrices (Section 5.5). Because $\|A^{\pi,\sigma}\|_F = \|A\|_F$, the partition of entries by M yields the in-pattern reformulation $d_M(A) = \|A\|_F^2 - \max_{\pi,\sigma} \|\Pi_M(A^{\pi,\sigma})\|_F^2$ (Section 2), which we use freely in the proofs below.

The boundary of the formulation. Masks whose objective does not depend on the permutation (the full mask; under $\sigma = \pi$ the diagonal and, for symmetric A, the triangle) are collected in Appendix C.3. Some classical notions of structure after reordering are not fixed binary supports and therefore fall outside Definition 1 as stated: Monge and Robinson matrices are ordering constraints on the *values* of A (a Robinson matrix is banded only approximately, which is why Section 5.1 fits a band of the correlation length to it); a nested hierarchy of blocks coincides, as a binary mask, with its coarsest level, so the hierarchy is visible in a support only through separators (nested dissection); and N:M sparsity [Pool and Yu, 2021] is a *family* of masks whose allowed positions differ between rows and groups, fitted by alternating the permutation with a per-group selection of the support rather than with one mask.

B.2 Basic properties of d_M

(i) Nonnegativity follows from the Frobenius norm being nonnegative. Equality $d_M(A) = 0$ iff some (π, σ) places every off-pattern entry of $A^{\pi,\sigma}$ at zero, i.e. $A^{\pi,\sigma}$ is M-supported.

(ii) Upper bound. By the in-pattern reformulation of (2), $d_M(A) = \|A\|_F^2 - \max_{\pi,\sigma} \|\Pi_M(A^{\pi,\sigma})\|_F^2 \le \|A\|_F^2$. For the tighter bound, average over $(\pi,\sigma) \in \mathcal{S}_n \times \mathcal{S}_n$: each entry A_{kl}^2 appears at each position (i,j) with equal probability, so $\mathbb{E}_{\pi,\sigma}[\|\Pi_M(A^{\pi,\sigma})\|_F^2] = \rho_M \|A\|_F^2$. Hence $\max_{\pi,\sigma} \|\Pi_M(A^{\pi,\sigma})\|_F^2 \ge \rho_M \|A\|_F^2$, giving $d_M(A) \le (1-\rho_M)\|A\|_F^2$. The averaging argument uses independent π and σ. Under the restriction $\sigma = \pi$ a common permutation never moves a diagonal entry off the diagonal, so with $D_A = \sum_i A_{ii}^2$, $O_A = \sum_{i \ne j} A_{ij}^2$ and m_{diag}, m_{off} the numbers of diagonal and off-diagonal positions of M, $\mathbb{E}_\pi \|\Pi_M(A^{\pi,\pi})\|_F^2 = \frac{m_{\text{diag}}}{n} D_A + \frac{m_{\text{off}}}{n(n-1)} O_A$, and the bound $(1-\rho_M)\|A\|_F^2$ can fail: for $A = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$ and $M = I_2$, $\min_\pi d_M(A; \pi, \pi) = 2 > (1-\rho_M)\|A\|_F^2 = 1$. The same remark applies to the expectation $\rho_M \|A\|_F^2$ used to motivate the density normalization in Section 4.1. More generally the common-permutation distance is a restriction of the feasible set: it is never below $d_M(A)$ and can exceed it even for symmetric A and M (the same example has $d_M(A) = 0$ with two permutations and 2 under one), so statements proved for independent permutations transfer to the case $\sigma = \pi$ only where this is said.

(iii) Permutation invariance. $d_M(P_\alpha A P_\beta^T) = \min_{\pi,\sigma} \|(I - \Pi_M)(P_\pi P_\alpha A P_\beta^T P_\sigma^T)\|_F^2 = \min_{\pi,\sigma} \|(I - \Pi_M)(P_{\pi\alpha}\, A\, P_{\sigma\beta}^T)\|_F^2 = \min_{\pi',\sigma'} d_M(A; \pi', \sigma') = d_M(A)$, reparametrising $\pi' = \pi\alpha$, $\sigma' = \sigma\beta$. □

B.3 Proof of Lemma 1 (Column-separability)

Expand the off-pattern mass with rows fixed at π_r and columns at σ:

$$\begin{aligned} &\sum_{i,j} (1 - M_{ij})\, (P_{\pi_r} A P_\sigma^T)_{ij}^2 \\ &= \sum_{i,j} (1 - M_{ij})\, A_{\pi_r^{-1}(i),\, \sigma^{-1}(j)}^2 = \sum_{i,j} (1 - M_{ij})\, B_{i,\, \sigma^{-1}(j)}^2. \end{aligned} \tag{7}$$

Substituting $p = j$ (target position) and $c = \sigma^{-1}(j)$ (original column placed at position j):

$$(7) = \sum_{p=1}^n \sum_{i=1}^n (1 - M_{ip})\, B_{i,\, \sigma^{-1}(p)}^2 = \sum_{c=1}^n \sum_{i=1}^n (1 - M_{i,\sigma(c)})\, B_{i,c}^2 = \sum_{c=1}^n Q_{\sigma(c),\, c}\,, \tag{8}$$

using $Q_{p,c} = \sum_i (1 - M_{ip}) B_{ic}^2$. The expression $\sum_c Q_{\sigma(c),c}$ over bijections $\sigma\colon [n] \to [n]$ is exactly a Linear Assignment Problem. The row case follows identically with rows and columns exchanged. □

B.4 Proof of Proposition 1 (Monotone convergence)

Define $F(\pi_r, \sigma) := \left\|(I - \Pi_M)(P_{\pi_r} A P_\sigma^T)\right\|_F^2$. At each half-step, one permutation is fixed and the other is set to the *global* minimizer of the resulting LAP (Lemma 1):

$$\sigma^{(t+1)} = \arg\min_\sigma F(\pi_r^{(t)}, \sigma) \quad \Rightarrow \quad F(\pi_r^{(t)}, \sigma^{(t+1)}) \le F(\pi_r^{(t)}, \sigma^{(t)}).$$

Similarly, $\pi_r^{(t+1)} = \arg\min_{\pi_r} F(\pi_r, \sigma^{(t+1)})$ gives $F(\pi_r^{(t+1)}, \sigma^{(t+1)}) \le F(\pi_r^{(t)}, \sigma^{(t+1)})$. Each full iteration reduces (or maintains) F. Since F takes values in the finite set $\{F(\pi,\sigma) : \pi, \sigma \in \mathcal{S}_n\}$, the sequence of values is eventually constant. Stabilization of the values alone does not stop the iterates: when the LAP has several optimal assignments the permutations can keep moving on a plateau of constant F. If ties are broken by keeping the current permutation whenever it already attains the LAP optimum, every change of a permutation strictly decreases F, so only finitely many changes occur and the final pair is a coordinate-wise local minimum (neither one-sided LAP can improve it). □

B.5 Proof of Theorem 1 (NP-hardness)

Throughout, $G = (V, E)$ is a simple graph with $|V| = n$ and $|E| = m$, and A_G its $\{0,1\}$ adjacency matrix. Restricting the feasible set to $\sigma = \pi$ changes the problem, so hardness of the symmetric problem does not by itself imply hardness of the two-permutation problem; the reduction treats both cases. For a pair (π, σ) the entry A_{uv} is placed at position $(\pi(u), \sigma(v))$.

We reduce from MINIMUM BISECTION [Garey and Johnson, 1979]: given G with n even, find $|S| = n/2$ minimizing $\mathrm{cut}(S, \bar{S})$. Let M be the 2-block mask with blocks of size $n/2$ and set

$$A \;=\; A_G + (m+1)\, I.$$

For (π, σ) let S be the set of vertices that π places in the first $n/2$ row positions and T the set that σ places in the first $n/2$ column positions. An entry is off-pattern iff its row and column positions lie in different blocks, so

$$d_M(A; \pi, \sigma) \;=\; (m+1)^2\, |S \triangle T| \;+ \sum_{u \in S,\, v \notin T} (A_G)_{uv} \;+ \sum_{u \notin S,\, v \in T} (A_G)_{uv}.$$

The two edge sums together are at most $2m$. If $S \neq T$ then, because $|S| = |T|$, $|S \triangle T| \geq 2$ and $d_M(A; \pi, \sigma) \geq 2(m+1)^2 > 2m$, whereas any pair with $S = T$ costs exactly $2\,\mathrm{cut}(S, \bar{S}) \leq 2m$. Hence every minimizer has $S = T$ and $d_M(A) = 2 \min_{|S|=n/2} \mathrm{cut}(S, \bar{S})$. The same identity holds under $\sigma = \pi$ (where $S = T$ automatically), which is the original symmetric reduction. □

The band of width 1 under $\sigma = \pi$. Let A be symmetric and M the banded mask with $w = 1$. Under $\sigma = \pi$ the in-band mass is

$$\sum_{|i-j| \leq 1} A^2_{\pi^{-1}(i), \pi^{-1}(j)} \;=\; \sum_u A^2_{uu} \;+\; 2 \sum_{i=1}^{n-1} A^2_{\pi^{-1}(i), \pi^{-1}(i+1)},$$

so by the in-pattern reformulation (Section 2) minimizing d_M is equivalent to maximizing $\sum_i A^2_{\pi^{-1}(i), \pi^{-1}(i+1)}$ over orderings, i.e. finding a maximum-weight Hamiltonian path in the complete graph with edge weights A^2_{uv}. For $A = A_G$ this sum equals $n - 1$ iff G has a Hamiltonian path, so even $\{0, 1\}$ weights suffice and computing the value of d_M is NP-hard [Garey and Johnson, 1979]. Deciding whether $d_M = 0$ (bandwidth 1) is polynomial [Saxe, 1980]; the hardness concerns the value of d_M when it is positive, and we do not claim it for independent permutations. □

B.6 Permutation ambiguities: automorphisms of a mask

In words: a planted instance has zero error at the planted permutations; any other zero-error pair must place the whole support of the signal inside the mask; and when the signal fills the mask, such a pair can only be a symmetry of the mask. Formally, let $\mathrm{Aut}(M) := \{(\alpha, \beta) \in \mathcal{S}_n^2 : M_{\alpha(i), \beta(j)} = M_{ij}\ \forall i, j\}$ be the joint automorphism group of M (the pairs of row and column permutations that map the mask onto itself), and $\mathrm{Aut}^{\mathrm{sym}}(M) := \{\sigma \in \mathcal{S}_n : M_{\sigma(i), \sigma(j)} = M_{ij}\}$ its single-permutation version for $M = M^\top$.

Proposition 3 (Zero-error permutations)**.** *Let S_0 be M-supported and $A = P_{\pi^*}^\top S_0\, P_{\sigma^*}$. Then $d_M(A) = 0$, attained at (π^*, σ^*); every pair with $d_M(A; \hat{\pi}, \hat{\sigma}) = 0$ has relative permutations $(\alpha, \beta) = (\hat{\pi} \circ (\pi^*)^{-1}, \hat{\sigma} \circ (\sigma^*)^{-1})$, the found pair expressed relative to the planted one, for which $P_\alpha S_0 P_\beta^\top$ is M-supported; and if $\mathrm{supp}(S_0) = \mathrm{supp}(M)$, then $(\alpha, \beta) \in \mathrm{Aut}(M)$. Under $\sigma = \pi$ the same holds for instances planted by one permutation, $A = P_{\pi^*}^\top S_0 P_{\pi^*}$, with $\mathrm{Aut}^{\mathrm{sym}}(M)$ in place of $\mathrm{Aut}(M)$ (an instance planted by two different permutations need not have $d_M(A; \pi) = 0$ for any π).*

Proof. $d_M(A; \pi^*, \sigma^*) = \|(I - \Pi_M)(S_0)\|_F^2 = 0$ by the support condition. If $d_M(A; \hat{\pi}, \hat{\sigma}) = 0$ then $P_{\hat{\pi}} A P_{\hat{\sigma}}^\top = P_\alpha S_0 P_\beta^\top$ is M-supported. If moreover $\mathrm{supp}(S_0) = \mathrm{supp}(M)$, then $\mathrm{supp}(P_\alpha S_0 P_\beta^\top) = \{(\alpha(i), \beta(j)) : M_{ij} = 1\} \subseteq \mathrm{supp}(M)$, and since permutations preserve cardinality the containment is an equality, i.e. $M_{\alpha(i), \beta(j)} = M_{ij}$ for all i, j. The symmetric case is the same argument with $\alpha = \beta$. Note that $\mathrm{Aut}(M)$ is the joint row × column action, in general strictly larger than the product of the row and column stabilizers: for $M = I_n$ it is the diagonal copy of $\mathcal{S}_n$, while the only purely row-preserving permutation is the identity. □

Proposition 4 (Automorphism groups of symmetric patterns)**.** *In the symmetric case ($\sigma = \pi$, $M = M^\top$) the ambiguities of the three main families are the following.*

(i) ***Banded mask*** *($M_{ij} = \mathbf{1}[|i-j| \le w]$, $w \ge 1$, $n > 2w+1$): the only symmetries are the identity and the reversal $i \mapsto n+1-i$, so an order is recoverable up to global reversal;* $\mathrm{Aut}^{\mathrm{sym}}(M) = \{\mathrm{id}, \mathrm{rev}\}$.

(ii) ***k-block mask*** *(equal blocks of size $b = n/k$): the symmetries are arbitrary permutations of the k blocks combined with arbitrary permutations within each block, so the partition into blocks is recoverable and the order within a block is not;* $\mathrm{Aut}^{\mathrm{sym}}(M) \cong (S_b)^k \rtimes S_k$, *the wreath product $S_b \wr S_k$.*

(iii) ***Arrowhead mask*** *($n \ge 3$): every permutation fixing the hub is a symmetry and no other is, so only the hub is identifiable;* $\mathrm{Aut}^{\mathrm{sym}}(M) \cong S_{n-1}$.

These ambiguities fix what recovery can mean, and the experiments score accordingly: an order by $|\tau|$, i.e. up to reversal (Section 5.1), a partition by ARI (Section 5.2), and an arrowhead fit as the identification of a hub.

Proof. For (i): in the simple graph on $\{1, \dots, n\}$ defined by the off-diagonal support of M ($i \sim j$ iff $1 \le |i-j| \le w$; the diagonal is preserved by every permutation), the vertices of minimum degree w are exactly 1 and n (for $w \ge 1$ and $n > 2w+1$ every interior vertex has degree $\ge w+1$), so an automorphism fixes or swaps them; composing with rev we may assume $\sigma(1) = 1$. The neighbours $2, \dots, w+1$ of vertex 1 have the distinct degrees $w+1, \dots, 2w$, so each is fixed, and for $j = 2, \dots, n-w$ the closed neighbourhood of the fixed vertex j contains exactly one vertex not yet fixed, $j+w$, which is therefore fixed as well. Hence $\sigma = \mathrm{id}$. (At $w = 0$ the mask is I_n and every permutation is an automorphism.) For (ii): $M_{ij} = 1$ iff $\lceil i/b \rceil = \lceil j/b \rceil$, so the off-diagonal support is a disjoint union of k cliques of size b; an automorphism maps cliques to cliques, hence permutes the blocks and acts by an arbitrary permutation within each. For (iii): any σ with $\sigma(1) = 1$ preserves M, and for $n \ge 3$ the hub is the only vertex adjacent to all others, so every automorphism fixes it (at $n = 2$ the arrowhead is the full mask). □

C Further Properties of the Framework

C.1 PatLR updates and proof of monotone convergence

The infimum in (4) need not be attained: the set of sums $S + UV^\top$ is not closed, as for low-rank plus sparse models [Tanner et al., 2019]; the algorithm uses only the conditional minimizers in S and X, which exist, and a safeguarded proposal in the permutations. Nor is the split unique: for an in-mask position (i, j) and $H = e_i e_j^\top$, every pair $S = tH$, $X = (1-t)H$ reconstructs $A = H$ exactly, so without further constraints the components are a decomposition of the fit, not identified parts, and the experiments score the reconstruction $S + UV^\top$. For fixed permutations, both inner updates are exact. Writing

$$F(\pi, \sigma, S, X) := \left\| P_\pi (A - X) P_\sigma^T - S \right\|_F^2$$

with $S \odot (1 - M) = 0$ and $\mathrm{rank}(X) \le k$:

(i) the pattern step is the projection $S^* = \Pi_M(P_\pi(A - X)P_\sigma^T)$;

(ii) the low-rank step is the rank-k truncation $X^* = \mathrm{SVD}_k(A - P_\pi^T S P_\sigma)$ by Eckart–Young (using $\left\| P_\pi(A - X)P_\sigma^\top - S \right\|_F^2 = \left\| (A - X) - P_\pi^\top S P_\sigma \right\|_F^2$).

Useful identity. For any (π, σ, X) the optimal S given the rest is $S^* = \Pi_M(P_\pi(A - X)P_\sigma^\top)$, and substituting back yields equation (5) of Section 4, restated here for the proof:

$$\min_{S:\, S \odot (1-M) = 0} F(\pi, \sigma, S, X) = \left\| (I - \Pi_M)(P_\pi (A - X) P_\sigma^\top) \right\|_F^2 = d_M(A - X; \pi, \sigma).$$

Proof of Proposition 2. Index iterations by t and write the state at the start of iteration t as $(\pi_t, \sigma_t, S_t, X_t)$. The body of the loop performs three updates.

(1) *Pattern step.* $S_{t+1} = \Pi_M(P_{\pi_t}(A - X_t)P_{\sigma_t}^\top)$ is the exact minimizer of F over M-supported S, so $F(\pi_t, \sigma_t, S_{t+1}, X_t) \le F(\pi_t, \sigma_t, S_t, X_t)$.

(2) *Low-rank step.* $X_{t+1} = \mathrm{SVD}_k(A - P_{\pi_t}^\top S_{t+1} P_{\sigma_t})$ is the global minimizer over rank-k X (Eckart–Young), so $F(\pi_t, \sigma_t, S_{t+1}, X_{t+1}) \le F(\pi_t, \sigma_t, S_{t+1}, X_t)$.

(3) *Safeguarded permutation step, followed by re-synchronization of S.* A candidate $(\tilde\pi, \tilde\sigma)$ is produced by ALTLAP on $A - X_{t+1}$ and accepted only if $d_M(A - X_{t+1}; \tilde\pi, \tilde\sigma) < d_M(A - X_{t+1}; \pi_t, \sigma_t)$. By (5) this compares the reduced objectives $\min_S F(\tilde\pi, \tilde\sigma, S, X_{t+1})$ and $\min_S F(\pi_t, \sigma_t, S, X_{t+1})$. The algorithm then recomputes $S'_{t+1} = \Pi_M(P_{\pi_{t+1}}(A - X_{t+1})P_{\sigma_{t+1}}^\top)$ for the accepted pair, which is the minimizer in (5), so

$$\begin{aligned} F(\pi_{t+1}, \sigma_{t+1}, S'_{t+1}, X_{t+1}) &= \min_S F(\pi_{t+1}, \sigma_{t+1}, S, X_{t+1}) \\ &\le \min_S F(\pi_t, \sigma_t, S, X_{t+1}) \le F(\pi_t, \sigma_t, S_{t+1}, X_{t+1}). \end{aligned}$$

If the proposal is rejected, $(\pi_{t+1}, \sigma_{t+1}) = (\pi_t, \sigma_t)$ and the same chain holds with equality. The recomputation matters: without it the stored S_{t+1} belongs to the old pair and $F(\pi_{t+1}, \sigma_{t+1}, S_{t+1}, X_{t+1})$ can exceed the previous value even though the reduced objective decreased.

Combining the three inequalities, $F(\pi_{t+1}, \sigma_{t+1}, S'_{t+1}, X_{t+1}) \le F(\pi_t, \sigma_t, S_t, X_t)$, where the left-hand side is exactly the state the loop carries into the next iteration and returns at termination. The sequence of F-values is therefore non-increasing and bounded below by 0, hence convergent. □

Unlike Proposition 1, this does not give finite termination: the permutations range over the finite set $\mathcal{S}_n \times \mathcal{S}_n$, but X is continuous, and for fixed (π, σ) the alternating S/X updates are alternating projections that converge in value without necessarily reaching a fixed point in finitely many steps. The algorithm stops when the decrease of F falls below a tolerance ε or after a capped number of iterations.

Coordinates of X. The low-rank factor X is kept in the original (unpermuted) coordinates across iterations: when (π, σ) changes, X does not need to be relabeled, whereas a copy stored in the permuted frame would become inconsistent with the new (π, σ) and break monotonicity.

C.2 Identifiability limits of the pattern score

The score $\tilde s_M(A; \pi, \sigma)$ and its minimised form $\tilde s_M(A)$ are those of equation (6). Let $A = P_{\pi^*}^\top S_0 P_{\sigma^*}$ with $\mathrm{supp}(S_0) = \mathrm{supp}(M^*)$. Then $d_{M^*}(A) = 0$ (the planted pair places every entry of S_0 inside M^*), so $\tilde s_{M^*}(A) = 0$. For a candidate M', $P_{\hat\pi} A P_{\hat\sigma}^\top = P_\alpha S_0 P_\beta^\top$ with $\alpha = \hat\pi(\pi^*)^{-1}$, $\beta = \hat\sigma(\sigma^*)^{-1}$, so $d_{M'}(A; \hat\pi, \hat\sigma) = 0$ exactly when $P_\alpha M^* P_\beta^\top \le M'$ entrywise. Hence every mask that contains a row/column-permuted copy of M^* also scores exactly 0, and every mask that does not scores strictly more, for any strictly positive density penalty $g(\rho_M)$ in place of $(1 - \rho_M)^{3/2}$; a sufficient condition for the latter is $|M'|_1 < |M^*|_1$. Among the zero-score masks the score does not distinguish at all: 0 divided by any penalty is 0, so the planted mask and its containers tie, and preferring the sparsest of them is a separate convention, not a property of the score. The remark concerns exact support only: a zero residual cannot distinguish a mask from its containers, whatever the density factor.

Role of the exponent. Under noise the containers acquire positive scores and the density factor becomes active. The exponent $3/2$ is a heuristic over-correction of coverage, not a derived constant: under independent uniform row and column permutations $\mathbb{E}[d_M(A; \pi, \sigma)] = (1 - \rho_M)\|A\|_F^2$ (Appendix B.2), so the normalized null level is $(1 - \rho_M)^{-1/2}$ and denser masks receive a larger null score, whereas the exponent 1 would equalize the null level across densities. Whether this is enough once the permutations are optimized is an empirical question, measured in Appendix E.5.1.

C.3 Special masks: permutation-invariant objectives and exact solutions

Table 5 summarizes the cases; the derivations follow.

Table 5: Masks for which the objective is permutation-invariant or exactly solvable.

Mask, setting	Objective
full mask $M = \mathbf{1}$	identically 0
diagonal, $\sigma = \pi$	constant $\sum_{a \neq b} A_{ab}^2$ (permutation-invariant)
upper triangle, $\sigma = \pi$, symmetric A	constant $\frac{1}{2} \sum_{a \neq b} A_{ab}^2$
upper triangle, $\sigma = \pi$, asymmetric A	weight of the backward arcs (linear ordering)
diagonal or anti-diagonal, independent (π, σ)	one LAP (Lemma 1)
arrowhead, $\sigma = \pi$	exact: choice of the hub, $O(n^2)$

Objectives that do not depend on the permutation. For $M = \mathbf{1}$, $d_M(A; \pi, \sigma) = 0$ for every A, π, σ. Under a common permutation ($\sigma = \pi$) and for any A, the diagonal mask gives $d_M(A; \pi) = \sum_{i \neq j} A_{\pi^{-1}(i), \pi^{-1}(j)}^2 = \sum_{a \neq b} A_{ab}^2$, since the pairs $(\pi^{-1}(i), \pi^{-1}(j))$ with $i \neq j$ range over all ordered pairs $a \neq b$. Under $\sigma = \pi$ and for *symmetric* A, the upper-triangular mask gives $d_M(A; \pi) = \sum_{i > j} A_{\pi^{-1}(i), \pi^{-1}(j)}^2 = \frac{1}{2} \sum_{a \neq b} A_{ab}^2$, because $A_{ab}^2 = A_{ba}^2$ and the pairs $i > j$ cover every unordered pair once; for an asymmetric A the same quantity is the weight of the backward arcs of the ordering, the linear-ordering objective. With independent permutations none of these masks is trivial: the diagonal and the anti-diagonal are single LAPs (Lemma 1), and the triangle is not constant.

Exact solution for the arrowhead mask under $\sigma = \pi$. The diagonal is invariant under a common permutation, so the only choice is which index becomes the hub h:

$$\max_{\pi} \|\Pi_M(A^{\pi,\pi})\|_F^2 \;=\; \sum_i A_{ii}^2 + \max_h \sum_{j \neq h} \big(A_{hj}^2 + A_{jh}^2\big),$$

computable in $O(n^2)$. The attention pipeline of Appendix E.7 uses this value as the arrowhead score and as a check on the solver.

D Algorithmic Details

D.1 Initialization and the exchange phase

Initialization. For banded patterns, the Fiedler vector of the graph Laplacian $L = D - |A|$, $D = \mathrm{diag}(|A|\mathbf{1})$, provides the start: ordering vertices by their spectral embedding aligns with bandwidth-minimal orderings [Barnard et al., 1995]. For block patterns we use spectral clustering [von Luxburg, 2007]; for the anti-diagonal band the Fiedler order of the folded path is unfolded from both ends and the two starts are kept as candidates; random restarts mitigate the residual local-minimum issue.

Common permutation. For $\sigma = \pi$ the two-sided iterate of a restart is rounded to one permutation, $\hat{\pi} \in \arg\min_{\pi \in \{\pi_0, \pi^*, \sigma^*\}} d_M(A; \pi)$ with π_0 the start (rounding alone can lose ground, so the start is kept as a candidate), and the best $\hat{\pi}$ over restarts is finished by the exchange descent. With C the permuted weight matrix ($C = B \odot B$ for the pattern distance, $C = B$ for the linear score, $B = P_{\hat{\pi}} A P_{\hat{\pi}}^\top$) and $D = 1 - M$, exchanging positions p and q changes the objective $\sum_{ij} D_{ij} C_{ij}$ by

$$\begin{aligned}\Delta_{pq} \;=\; &\sum_{j \neq p,q} \big[(D_{pj} - D_{qj})(C_{qj} - C_{pj}) + (D_{jp} - D_{jq})(C_{jq} - C_{jp})\big] \\ &+ (D_{pq} - D_{qp})(C_{qp} - C_{pq}) + (D_{pp} - D_{qq})(C_{qq} - C_{pp}),\end{aligned}$$

valid for asymmetric A as well; the matrix (Δ_{pq}) is two products $DC^\top$ and $D^\top C$ plus entrywise terms, $O(n^3)$ per evaluation. A round evaluates Δ, then applies the improving exchanges in order of gain on disjoint positions, at most $n/32$ per round, each re-checked exactly in $O(n)$ under the current order before it is applied; rounds continue while an exchange improves. The output is never worse than the start and no exchange of two positions improves it (Proposition 1). With several starts the LAP phase runs from each start and the exchange phase from the best rounded permutation; on the sparse graphs of Appendix E.6.3, where the LAP steps stall early, the two phases are interleaved per start (three rounds). The exchange phase costs about 1 s at n=500 and 7 s at n=1000 on one core (Appendix E.8).

Algorithm 3 ADAPTIVE PATLR

Require: A, rank k, vocabulary $\mathcal{V}=\{M_1,\dots,M_K\}$, budget N, cooling γ, proposal steps ℓ

$(\pi,\sigma) \leftarrow \textsc{FiedlerInit}(A)$; $j \leftarrow \arg\min_{j'} \tilde{s}_{M_{j'}}(A,\pi,\sigma)$; $T \leftarrow T_0 > 0$
$(j^*,\pi^*,\sigma^*) \leftarrow (j,\pi,\sigma)$ {incumbent: best visited state}
for $t = 1,\dots,N$ **do**
  $(\pi,\sigma) \leftarrow \textsc{AltLAP}(A, M_j, 1; (\pi,\sigma))$
  if $\tilde{s}_{M_j}(A,\pi,\sigma) < \tilde{s}_{M_{j^*}}(A,\pi^*,\sigma^*)$ **then** $(j^*,\pi^*,\sigma^*) \leftarrow (j,\pi,\sigma)$ {update incumbent}
  with prob. $p(T)=\max(0.15, T/T_0)$:
    $j' \sim \mathrm{Uniform}[K]$; $(\pi',\sigma') \leftarrow \textsc{AltLAP}(A, M_{j'}, \ell; (\pi,\sigma))$
    $\Delta \leftarrow \tilde{s}_{M_{j'}}(A,\pi',\sigma') - \tilde{s}_{M_j}(A,\pi,\sigma)$
    if $\Delta < 0$ **or** $\mathrm{rand}() < e^{-\Delta/T}$ **then** $(j,\pi,\sigma) \leftarrow (j',\pi',\sigma')$
  if $\tilde{s}_{M_j}(A,\pi,\sigma) < \tilde{s}_{M_{j^*}}(A,\pi^*,\sigma^*)$ **then** $(j^*,\pi^*,\sigma^*) \leftarrow (j,\pi,\sigma)$ {update incumbent}
  $T \leftarrow \gamma T$
end for
return $\textsc{PatLR}(A, M_{j^*}, k)$ started from (π^*,σ^*) {Alg. 2; a fresh ALTLAP start on M_{j^*} is also tried and the better kept}

D.2 Adaptive PatLR: joint pattern–permutation search

Settings not stated in Section 4.1: the default vocabulary is blocks $k \in \{2,3,4,6\}$, bands $w \in \{1,2,4\}$, arrowhead and block+band (k=3, w=2), nine masks with fixed parameters (the attention study uses the nine-mask vocabulary of Section 5.5); the exploration probability is $p(T) = \max(0.15,\, T/T_0)$, T_0 is calibrated from the score spread at the Fiedler initialization (floored at a small positive constant), γ=0.95, and a proposal is evaluated with ℓ=3 LAP steps from the current pair (a permutation pair optimized for block k=6 is already a reasonable start for block k=4 or block+band). The ALTLAP calls inside the search are the LAP phase of Algorithm 1, without the exchange phase: the search compares many masks under a small budget, and finishing every proposal at a local minimum would turn each cheap evaluation into a full fit. The exchange phase, a fresh three-restart fit and PATLR are applied to the selected mask only. The incumbent updates make the output the best visited state; whether the finite search visits the planted mask is the empirical question of Section 5.4 and Appendix E.5.1.

Out-of-vocabulary structure. The selector never rejects: it returns the best-scoring visited state, so a structure outside the vocabulary (unequal or nested blocks, a hybrid shape) is approximated by the closest available mask; in our planted experiments the nearest single-family mask is selected rather than a noise-level answer. Two practical remedies follow. A best score close to the random-permutation level indicates that no mask in the vocabulary explains the matrix. That level is not a universal constant: for independent random permutations and no optimization, $\mathbb{E}[\tilde{s}_M] = (1-\rho_M)/(1-\rho_M)^{3/2} = (1-\rho_M)^{-1/2}$, under $\sigma = \pi$ it depends on the split of the mass between the diagonal and the off-diagonal (Appendix B), and optimizing over permutations lowers it further, so a threshold on it is a heuristic calibrated per mask rather than a derived rejection rule; $A = 0$ and $\rho_M = 1$ are excluded by convention. The vocabulary is extended by adding a mask, at a cost linear in $|\mathcal{V}|$. Estimating mask parameters from the data instead of enumerating fixed masks (e.g. block boundaries by change-point segmentation of the reordered matrix, alternated with the permutation step), and calibrating each mask's score against its own random-permutation null rather than a density correction, are left for future work.

E Experimental Details and Additional Results

For the experiments summarized in Section 5 this appendix gives the generators, the information available to each method, the baseline settings, the full grids and the paired intervals; it also reports the experiments that appear only here (the compute-budget ablation, the circular and nested ordering geometries, block-size estimation, the long-context control and wall-clock scaling), each with the question it answers. Unless stated otherwise, comparisons are paired on identical instances over 20 seeds with 95% bootstrap intervals of the paired difference.

E.1 Ablation: Initialization, Restarts, and Compute Budget

Figure 7 runs the solvers of Table 1 on the same instances with increasing budgets and records the running best error reduction against cumulative wall-clock: Alternating LAP from the Fiedler order plus random restarts (the protocol of Table 1, five restarts there) and from random restarts only, up to 20 restarts, and FAQ with 1 to 10 starts. Three things are visible. The first run accounts for almost all of the gain over the Fiedler order alone; additional restarts add at most 0.04 over 20 restarts, and random starts reach the values of the Fiedler start on all but one panel (Gauss w=5, n=200, where the Fiedler start stays 0.05 ahead), so on these planted instances the descent, not the spectral start, does the work. On the sparse $\{0,1\}$ graphs the restart budget adds about 0.01: the LAP steps stall on the many ties of a sparse cost matrix (the same stalling appears without any noise on the w=1 band, the Hamiltonian-path case of Appendix B.5) and the exchange phase supplies the remaining descent (next paragraph). FAQ with one start is weak there and owes its final value to its randomized starts; on the Gaussian inputs Alternating LAP is above FAQ at every budget.

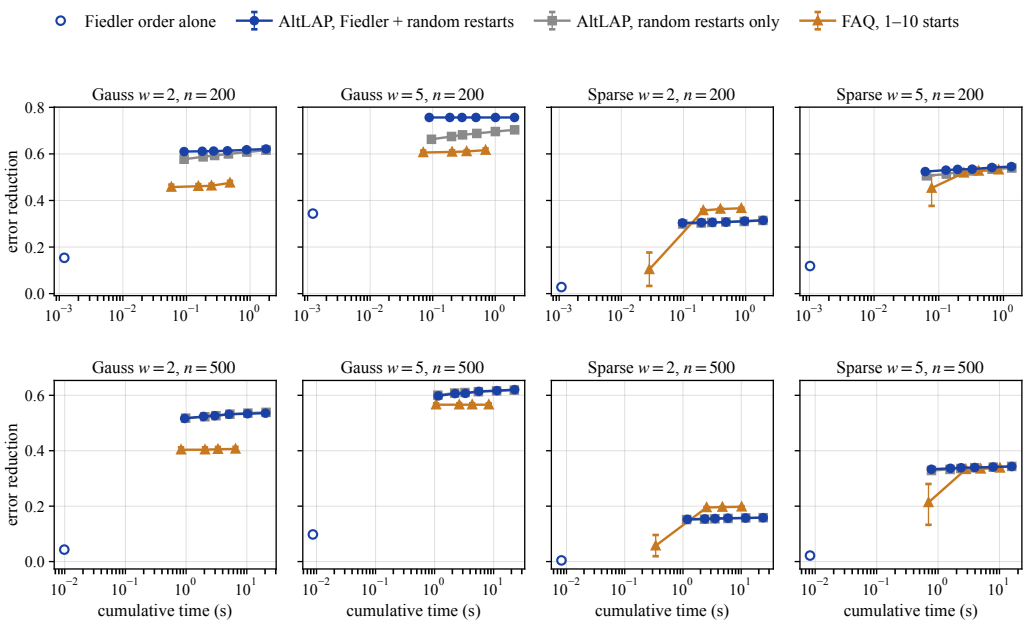

Figure 7: Error reduction against cumulative wall-clock on the instances of Table 1 (20 seeds, 95% bootstrap intervals; single CPU core). Markers: AltLAP after $1, 2, 3, 5, 10, 20$ restarts (Fiedler start plus random, or random only), FAQ after $1, 3, 5, 10$ starts; the open circle is the Fiedler order without any LAP step.

Without the exchange phase. Running Algorithm 1 with the exchange phase disabled, on the instances and budget of Table 1, separates the two phases. The LAP phase alone reaches ER $0.61/0.50/0.41$ (Gauss w=2), $0.78/0.65/0.55$ (Gauss w=5), $0.15/0.06/0.01$ (sparse w=2) and $0.63/0.48/0.29$ (sparse w=5) at n=100/200/500; the exchange phase raises these to $0.69/0.61/0.52$, $0.87/0.75/0.61$, $0.45/0.30/0.15$ and $0.69/0.53/0.34$, the values of Table 1. On the Gaussian inputs it adds 0.06–0.11 and on the sparse graphs it supplies most of the reduction, since the LAP steps stall on the ties of a $\{0,1\}$ cost matrix, the same stalling as on the sparse SBM graphs of Section 5.2. At n=500 the phase takes about a third of the run time on the Gaussian instances and about a tenth on the sparse graphs (runtimes in Appendix E.2).

E.2 Banded Baselines from the Seriation, QAP, and Bandwidth Literatures

Table 6 extends Table 1 with all methods and both of our metrics (ER / BE), and Table 7 scores the same orderings by the bandwidth family's own criteria. Methods: RCM [Cuthill and McKee, 1969] and Gibbs–Poole–Stockmeyer (GPS) [Gibbs et al., 1976] on the $\{0,1\}$ support (the graph itself in the sparse regime; the KNN-20 sparsification in the Gaussian regime); F+greedy (Fiedler

order plus greedy adjacent swaps on 2-SUM); the plain spectral 2-SUM order of Atkins et al. [1998] (the Fiedler ordering itself); convex seriation [Fogel et al., 2013] (Frank–Wolfe over the permutahedron with the concave vertex-pulling term, penalty weight and start chosen by the 2-SUM value); QAP-FAQ with three starts; FAQ followed by AltLAP (keeping the better of the two); AltLAP with one restart (equal-budget variant) and with five. Our objective and FAQ's are the same QAP, but the two heuristics solve different inner problems: FAQ relaxes the single-permutation problem, whereas AltLAP descends on the two-sided objective, rounds to one permutation and finishes with the exchange descent; the rounding can lose ground, which is why the start is kept as a candidate. The two are closely related: at a permutation P_π the FAQ gradient $\nabla_P \operatorname{tr}(M^\top P W P^\top) = 2MP_\pi W$ equals $-Q$ of Lemma 1 up to a constant per column, so the first FAQ direction-finding step from a permutation is the LAP step of Algorithm 1; FAQ then takes a fractional step inside the Birkhoff polytope, while AltLAP moves to the vertex and alternates sides. Warm-starting AltLAP from the FAQ order (the FAQ+AltLAP column of Table 6) adds 0.02–0.12 on the Gaussian inputs, where the LAP steps rarely move the FAQ order and the gain is the exchange phase, and at most 0.01 on the sparse graphs.

Table 6: Banded recovery, ER / BE (higher = better), 20 seeds. Bold marks the best ER per row (ties within 0.005).

Regime	n	RCM	GPS	F+greedy	Spectral	Convex	FAQ	FAQ+AltLAP	AltLAP-r1	AltLAP
Gauss w=2	30	0.03/0.39	0.05/0.41	0.84/0.90	0.76/0.85	0.85/0.91	0.60/0.75	0.68/0.80	**0.87/0.92**	**0.87/0.92**
	100	0.05/0.34	0.05/0.34	0.37/0.56	0.32/0.52	0.36/0.56	0.51/0.66	0.62/0.73	**0.69/0.78**	**0.69/0.78**
	200	0.03/0.30	0.04/0.31	0.14/0.38	0.15/0.39	0.14/0.38	0.46/0.61	0.58/0.70	**0.61/0.72**	**0.61/0.72**
	500	0.03/0.27	0.03/0.27	0.03/0.27	0.04/0.28	0.04/0.28	0.40/0.55	**0.52/0.64**	0.52/0.64	**0.52/0.64**
Gauss w=5	30	0.07/0.54	0.09/0.55	**0.98/0.99**	0.97/0.98	**0.98/0.99**	0.84/0.92	0.86/0.93	**0.98/0.99**	**0.98/0.99**
	100	0.09/0.41	0.09/0.41	0.72/0.82	0.61/0.75	0.73/0.82	0.66/0.78	0.72/0.82	**0.87/0.91**	**0.87/0.91**
	200	0.08/0.35	0.08/0.36	0.33/0.53	0.34/0.54	0.34/0.54	0.61/0.73	0.66/0.76	**0.76/0.83**	0.75/0.82
	500	0.06/0.30	0.05/0.29	0.08/0.32	0.10/0.33	0.09/0.32	0.57/0.68	0.60/0.70	0.60/0.70	**0.61/0.71**
Sparse w=2	30	0.23/0.33	0.24/0.33	0.45/0.51	0.39/0.47	0.46/0.52	0.73/0.76	**0.74/0.77**	0.67/0.71	0.67/0.71
	100	0.06/0.10	0.06/0.10	0.09/0.13	0.09/0.13	0.09/0.13	**0.49/0.51**	**0.50/0.52**	0.44/0.46	0.45/0.47
	200	0.02/0.05	0.02/0.04	0.03/0.05	0.03/0.05	0.03/0.05	**0.36/0.37**	**0.36/0.37**	0.30/0.32	0.30/0.31
	500	0.00/0.01	0.00/0.01	0.00/0.01	0.00/0.01	0.01/0.01	**0.20/0.20**	**0.20/0.20**	0.15/0.16	0.15/0.16
Sparse w=5	30	0.41/0.59	0.04/0.34	0.78/0.85	0.76/0.83	0.78/0.85	0.88/0.92	0.89/0.93	**0.91/0.94**	**0.91/0.94**
	100	0.16/0.24	0.15/0.23	0.29/0.36	0.29/0.36	0.30/0.37	0.66/0.69	0.66/0.69	0.68/0.71	**0.69/0.72**
	200	0.09/0.14	0.08/0.13	0.10/0.14	0.12/0.16	0.10/0.15	0.52/0.54	0.52/0.55	0.53/0.55	**0.53/0.56**
	500	0.03/0.05	0.02/0.04	0.02/0.04	0.02/0.04	0.02/0.04	**0.33/0.35**	**0.34/0.35**	**0.33/0.35**	**0.34/0.35**

Two metric families, one support. Bandwidth $\max\{|i-j| : S_{ij} = 1\}$ and profile $\sum_i (i - \min\{j \le i : S_{ij} = 1\})$ need a discrete support, so every ordering is scored on the same $\{0, 1\}$ support that RCM and GPS are given. GPS is the better bandwidth reducer, as expected, and neither bandwidth method reduces off-band mass. Conversely our methods barely reduce bandwidth: a few small entries far from the diagonal cost almost no mass but fix $\max |i-j|$. On noisy inputs the classical criterion saturates (at n=500 in the Gaussian regime every ordering, including the identity, has a bandwidth close to the maximum), because the signal falls below the noise at $|i-j| \approx 7$ and the support beyond is noise, so the criterion is set by the farthest noise entries while the pattern distance measures the mass concentration.

Runtimes at n=500 (one EPYC core): RCM, GPS and the spectral order 0.01–0.02 s; convex seriation 0.45–0.55 s; F+greedy 0.7 s; FAQ (three starts) 2.0–2.8 s; AltLAP with one restart 0.8–1.2 s and with five 1.7–4.1 s (the exchange phase runs once, from the best restart).

E.3 PatLR Decomposition Visualizations

Figure 8 follows the decomposition on one synthetic instance of the experiment of Figure 4, where the planted parts are known and can be compared with the recovered ones, and, without added noise, on the two real networks of Section 5.3 and on three feature-correlation matrices from the UCI repository [Wine, Breast Cancer Wisconsin (Diagnostic) and Optical Recognition of Handwritten Digits; Aeberhard and Forina, 1992, Wolberg et al., 1993, Alpaydin and Kaynak, 1998], each with the mask of the default vocabulary that fits it best at rank 1. The first column shows each matrix in its native order (node numbering of the graph, feature order of the dataset), the second the random order given to the solver. The row norm of the low-rank part follows the row mass on Les Misérables,

Table 7: The same orderings scored by the bandwidth family: bandwidth (lower = better; identity order and six methods) and profile for RCM, GPS and AltLAP, means over 20 seeds on the shared $\{0, 1\}$ support.

			Bandwidth						Profile		
Regime	n	identity	RCM	GPS	F+greedy	Convex	FAQ	AltLAP	RCM	GPS	AltLAP
Gauss w=2	30	29	**26**	**26**	28	29	29	29	**416**	**417**	422
	100	97	**82**	83	96	95	97	96	**4251**	4515	4466
	200	196	162	**159**	192	192	197	195	**16809**	17712	17592
	500	494	405	**399**	489	490	496	492	**104377**	109910	110834
Gauss w=5	30	29	**26**	**26**	28	29	29	29	**416**	**417**	423
	100	97	**82**	83	96	95	97	97	**4251**	4515	4476
	200	196	162	**159**	192	192	196	194	**16809**	17712	17598
	500	494	405	**399**	489	490	493	494	**104377**	109910	110756
Sparse w=2	30	26	10	**8**	10	10	25	16	135	142	**120**
	100	94	47	**45**	54	52	92	87	**2298**	2634	2363
	200	192	117	**111**	139	133	190	192	**11442**	12589	13083
	500	492	365	**358**	455	441	491	491	**93432**	101738	103868
Sparse w=5	30	29	29	**24**	29	29	29	29	430	**407**	428
	100	96	58	**55**	70	71	93	91	2823	3165	**2483**
	200	195	131	**128**	164	159	192	189	**13100**	14894	13319
	500	493	**392**	393	456	460	493	490	**98342**	106402	103963

Wine and Breast Cancer; on the karate club the rank-1 part concentrates on the two highest-degree nodes, the two faction leaders.

E.4 PatLR: Full Denoising Grid, Ablations and Capacity Control

Setup of Figure 4, remaining details. The pattern part is $S_0 = 2\,M \odot R + 0.15\,(1 - M) \odot R$ for one symmetric Gaussian matrix R, with the diagonal set to $|R_{ii}| + 0.3$ (the generator of the capacity control below); S_0 is symmetric for the six symmetric masks, while for the upper-triangular mask it is asymmetric (upper triangle at scale 2, lower at 0.15), so under the common permutation its objective is the linear-ordering problem of Table 2, not the constant it would be for a symmetric matrix. The low-rank part is $L_0 = UU^\top$ with $U \in \mathbb{R}^{n\times 3}$ Gaussian, scaled to a fraction f of the clean energy (f=0.5 in Figure 4); the noise is scaled per instance to $\mathrm{SNR} = \|A\|_F^2/\|E\|_F^2$, so the noisy input has DQ $1 - 1/\mathrm{SNR}$. RPCA is principal component pursuit with its low-rank part as the estimate; SVD-dof is the SVD at the rank with the parameter count of PatLR (rule below); PatLR runs from the Fiedler order and from its unfolding (one random restart each) and keeps the lower objective; PatLR-rnd uses a random mask of the same density; the oracle column fixes the permutation to the planted one and alternates S and X only.

Table 8 gives the full grid. Three regularities hold across it. An estimate that copies the in-mask entries, i.e. $\hat{C} = M \odot (C + E) + (1 - M) \odot X$ in the coordinates of the mask (pattern alone with $X = 0$, PatLR and the oracle with the rank-3 X), has error $\|M \odot E\|_F^2 + \|(1 - M) \odot (X - C)\|_F^2$ exactly, the two terms having disjoint supports, so its DQ is at most $1 - \|M \odot E\|_F^2/\|C\|_F^2$. For a mask placed independently of the noise the in-mask noise energy is $\rho_M\|E\|_F^2$ in expectation, with a relative fluctuation of about $1/\sqrt{}$ at n=200, which gives the value $1 - \rho_M/\mathrm{SNR}$; a permutation chosen from the data tends to place more noise inside the mask, not less. The off-mask term is the clean energy outside the mask for the pattern-only estimate ($X = 0$) and $\|(1 - M) \odot (X - C)\|_F^2$ for PatLR, which is smaller when C has low-rank structure outside the mask and larger when X fits noise. On the pattern-only families the off-mask clean values are i.i.d. and carry no low-rank structure, so on the five families whose order the solver recovers the rank-3 oracle lands below the pattern-only value by 0.01 at SNR 4, 0.05 at SNR 1 and 0.07–0.11 at SNR 0.5 (on the band and the anti-diagonal the oracle's planted order puts it above the pattern-only estimate, whose order is the solver's). The gap between PatLR and the oracle is concentrated on the banded and anti-diagonal families, i.e. in the permutation search on sparse masks, and grows as the SNR falls. The random-mask control PatLR-rnd lies 0.2–0.5 below PatLR on the planted pattern and 0.1–0.4 below it with the rank-3 component at SNR 1; it stays above the SVD level where random positions can absorb signal mass under a free permutation (the band, anti-diagonal, block and block+band families on the planted pattern, the band and anti-diagonal with the rank-3 component) and within 0.05 of it or below elsewhere. With 80% of the clean energy in the low-rank part, SVD-best is the better estimate

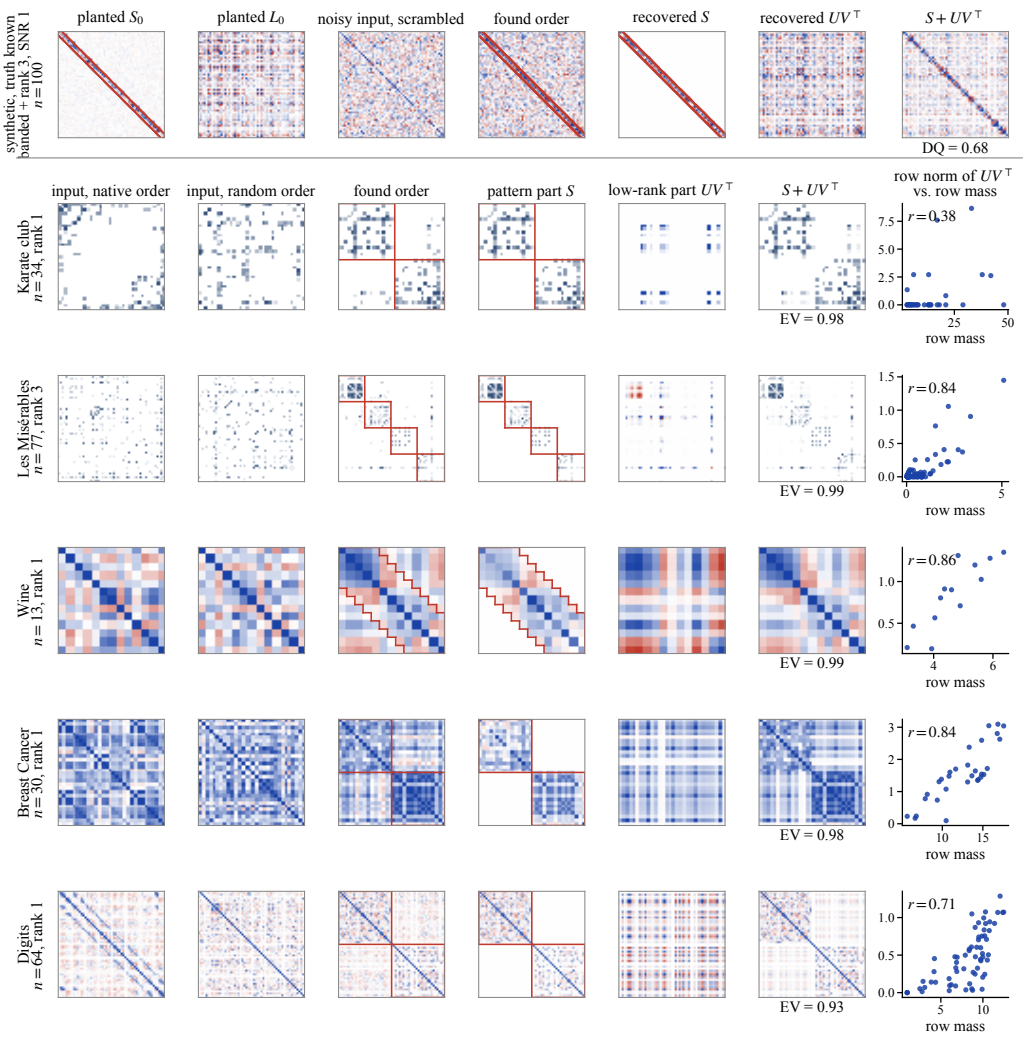

Figure 8: PatLR step by step. Top row: a synthetic instance with known truth (banded $w{=}3$ pattern plus a rank-3 component of equal energy, $n{=}100$, scrambled, SNR 1): the planted S_0 and L_0, the noisy input, the input in the found order, the recovered S and $UV^\top$ and their sum with its DQ. Remaining rows: real matrices without added noise, karate club (weighted adjacency, $\mathrm{block}(18, 16)$ from the two factions, rank 1); Les Misérables (co-occurrence weights scaled to $[0, 1]$, $\mathrm{block}(19, 19, 19, 20)$ template, rank 3); and the signed feature-correlation matrices of Wine (band $w{=}4$), Breast Cancer ($\mathrm{block}(15, 15)$) and Digits (pixel correlations, $\mathrm{block}(32, 32)$), rank 1; left to right: the input in its native order; the input in the random order given to the solver; the input in the order found by PatLR, mask outlined; the pattern part S; the low-rank part $UV^\top$; their sum with its explained variance; the row norm of $UV^\top$ against the row mass $\sum_j |A_{ij}|$ (weighted degree for the graphs, total absolute correlation for the features). Graphs on a white-to-blue scale, signed matrices and $UV^\top$ on a diverging scale (blue positive); r is the correlation in the last column.

on the dense masks, and with 20% PatLR is the best estimate on every family: the mask pays for the in-mask noise it copies only while the pattern carries enough of the signal.

Ablation of the permutation. Table 9 fixes the permutation instead of searching it, on the planted-pattern matrices at $\sigma{=}0.3$ (scrambled). With the permutation fixed to the input order the effect differs between families: block, checkerboard, block+band, anti-diagonal and triangular fall below the SVD level, whereas the masks that contain the diagonal keep what a symmetric scrambling leaves in place; the band stays above SVD on the diagonal entries alone, and the arrowhead reaches

Table 8: Denoising quality on the seven families, all settings of the experiment of Figure 4 (n=200; 20 seeds unless stated; best of SVD-dof, SVD-best, RPCA, pattern only, PatLR and PatLR-rnd per row in bold; "oracle" is PatLR at the planted permutation).

Pattern	ρ_M	input	SVD-dof	SVD-best	RPCA	pattern only	**PatLR**	PatLR-rnd	oracle
planted pattern, SNR 4.0									
Banded	0.035	0.75	0.12	0.75	0.10	0.76	**0.76**	0.55	0.88
Block	0.250	0.75	0.34	0.75	0.21	**0.92**	0.91	0.54	0.91
Arrowhead	0.015	0.75	0.34	0.75	0.19	**0.78**	**0.78**	0.36	0.78
Checkerboard	0.500	0.75	0.53	0.75	0.34	**0.87**	0.86	0.60	0.86
Block+band	0.327	0.75	0.41	0.75	0.25	**0.91**	0.89	0.55	0.89
Anti-diagonal	0.102	0.75	0.21	0.75	0.12	**0.86**	**0.86**	0.55	0.91
Triangular	0.502	0.75	0.66	0.75	0.41	**0.87**	0.86	0.53	0.86
planted pattern, SNR 1.0									
Banded	0.035	0.00	0.03	0.07	-0.05	**0.69**	0.64	0.44	0.81
Block	0.250	-0.00	0.01	0.02	0.11	**0.73**	0.68	0.20	0.68
Arrowhead	0.015	0.00	0.27	0.28	0.03	**0.77**	0.73	0.27	0.72
Checkerboard	0.500	-0.00	-0.00	0.01	0.21	**0.49**	0.44	0.07	0.44
Block+band	0.327	-0.00	0.01	0.01	0.15	**0.66**	0.60	0.14	0.60
Anti-diagonal	0.102	0.00	0.03	0.04	0.01	**0.72**	0.67	0.37	0.80
Triangular	0.502	-0.00	0.11	0.12	0.25	**0.48**	0.44	0.01	0.44
planted pattern, SNR 0.5									
Banded	0.035	-1.00	-0.10	-0.02	-0.28	**0.59**	0.49	0.28	0.71
Block	0.250	-1.00	-0.42	-0.02	-0.11	**0.48**	0.37	-0.28	0.37
Arrowhead	0.015	-1.00	0.19	0.25	-0.22	**0.75**	0.67	0.14	0.65
Checkerboard	0.500	-1.00	-0.72	-0.02	-0.04	**-0.01**	-0.12	-0.63	-0.12
Block+band	0.327	-1.00	-0.53	-0.02	-0.08	**0.32**	0.21	-0.40	0.22
Anti-diagonal	0.102	-1.00	-0.20	-0.01	-0.21	**0.50**	0.36	0.08	0.64
Triangular	0.502	-1.00	-0.63	**-0.01**	**-0.01**	-0.04	-0.13	-0.68	-0.11
pattern + rank-3 component (equal energy), SNR 4.0									
Banded	0.035	0.75	0.55	0.75	0.54	0.42	**0.86**	0.76	0.94
Block	0.250	0.75	0.62	0.75	0.60	0.57	**0.92**	0.71	0.92
Arrowhead	0.015	0.75	0.67	0.76	0.58	0.44	**0.89**	0.58	0.89
Checkerboard	0.500	0.75	0.68	0.75	0.65	0.63	**0.86**	0.71	0.85
Block+band	0.327	0.75	0.64	0.75	0.62	0.59	**0.90**	0.70	0.90
Anti-diagonal	0.102	0.75	0.57	0.75	0.55	0.43	**0.88**	0.75	0.94
Triangular	0.502	0.75	0.74	0.76	0.67	0.63	**0.86**	0.67	0.86
pattern + rank-3 component (equal energy), SNR 1.0									
Banded	0.035	0.00	0.47	0.51	0.35	0.34	**0.77**	0.66	0.88
Block	0.250	-0.00	0.30	0.49	0.43	0.38	**0.71**	0.37	0.71
Arrowhead	0.015	0.00	0.63	0.63	0.37	0.43	**0.85**	0.50	0.85
Checkerboard	0.500	0.00	0.15	**0.49**	0.46	0.24	0.46	0.19	0.45
Block+band	0.327	-0.00	0.24	0.49	0.44	0.33	**0.63**	0.31	0.63
Anti-diagonal	0.102	-0.00	0.41	0.49	0.38	0.26	**0.69**	0.56	0.84
Triangular	0.502	0.00	0.19	**0.49**	0.47	0.24	0.45	0.16	0.46
pattern + rank-3 component (equal energy), SNR 0.5									
Banded	0.035	-1.00	0.37	0.47	0.08	0.22	**0.61**	0.50	0.81
Block	0.250	-1.00	-0.12	**0.46**	0.15	-0.09	0.36	-0.08	0.42
Arrowhead	0.015	-1.00	0.56	0.56	0.09	0.41	**0.80**	0.38	0.80
Checkerboard	0.500	-1.00	-0.57	**0.46**	0.17	-0.39	-0.22	-0.50	-0.08
Block+band	0.327	-1.00	-0.28	**0.46**	0.16	-0.19	0.18	-0.22	0.27
Anti-diagonal	0.102	-1.00	0.19	**0.45**	0.11	-0.04	0.35	0.27	0.71
Triangular	0.502	-1.00	-0.54	**0.46**	0.18	-0.28	-0.12	-0.52	-0.08
*pattern + rank-3 component carrying 20% of the clean energy, SNR 1 (*10 *seeds)*									
Banded	0.035	0.00	0.21	0.22	0.11	0.55	**0.71**	0.53	0.85
Block	0.250	-0.00	0.13	0.20	0.24	0.59	**0.70**	0.27	0.70
Arrowhead	0.015	0.00	0.42	0.42	0.16	0.64	**0.70**	0.29	0.79
Checkerboard	0.500	-0.00	0.06	0.20	0.31	0.40	**0.46**	0.11	0.45
Block+band	0.327	0.00	0.10	0.20	0.26	0.53	**0.62**	0.21	0.63
Anti-diagonal	0.102	0.00	0.18	0.20	0.16	0.55	**0.67**	0.45	0.83
Triangular	0.502	-0.00	0.14	0.21	0.33	0.38	**0.45**	0.06	0.46
*pattern + rank-3 component carrying 80% of the clean energy, SNR 1 (*10 *seeds)*									
Banded	0.035	0.00	0.73	0.78	0.57	0.14	**0.81**	0.77	0.91
Block	0.250	-0.00	0.47	**0.78**	0.60	0.16	0.57	0.48	0.72
Arrowhead	0.015	0.00	0.81	0.81	0.57	0.21	**0.91**	0.73	0.91
Checkerboard	0.500	-0.00	0.23	**0.78**	0.60	0.12	0.35	0.27	0.43
Block+band	0.327	-0.00	0.38	**0.78**	0.60	0.15	0.49	0.41	0.64
Anti-diagonal	0.102	0.00	0.63	**0.78**	0.58	0.07	0.67	0.66	0.86
Triangular	0.502	-0.00	0.24	**0.78**	0.60	0.10	0.43	0.26	0.46

the searched value because its hub row and column form a rank-2 matrix that the low-rank term absorbs at any order. With the permutation fixed to the planted order PatLR is high on every family. The joint solver attains the planted-order value on five families and stays below it on banded and anti-diagonal, the two sparse masks whose order is hardest to recover.

Table 9: PatLR with the permutation searched, fixed to the input order, or fixed to the planted order (DQ, n=200, σ=0.3, scrambled inputs, 20 seeds).

Pattern	SVD-dof	pattern only	PatLR, input order	**PatLR**	PatLR, planted order
Banded	0.10	0.73	0.29	0.73	0.86
Block	0.41	0.96	0.31	0.96	0.96
Arrowhead	0.28	0.48	0.74	0.74	0.73
Checkerboard	0.68	0.97	0.54	0.97	0.97
Block+band	0.50	0.96	0.38	0.97	0.96
Anti-diagonal	0.21	0.81	0.14	0.83	0.92
Triangular	0.79	0.97	0.55	0.97	0.97

Capacity control at a matched parameter count. Table 10 compares the models at an equal nominal parameter count on the planted-pattern matrices at σ=0.3 in their planted order (the scrambled version changes no entry by more than 0.01). PatLR with mask M and residual rank r has $\mathrm{df} = |M| + r(2n - r)$ free parameters (the in-mask entries plus a rank-r factorization). SVD-dof receives the rank $r' = \mathrm{round}\big(n - \sqrt{\max(n^2 - \mathrm{df}, 0)}\big)$ clipped to $[1, n]$, which has the same count; SVD-3 is the fixed-rank estimate. The blockmodel baseline is spectral co-clustering [Dhillon, 2001]: k-means on the leading singular vectors of rows and of columns, then block means. "BM" chooses $k \in \{2, 3, 4, 5, 6, 8\}$ by the best DQ (an oracle choice, generous to the baseline); "BM-dof" uses $k = \mathrm{round}(\sqrt{\mathrm{df}})$, the same nominal budget as PatLR. PatLR-rnd replaces the planted mask by a random mask of the same density, redrawn per seed, with an otherwise identical pipeline. At σ=0.3 the noise carries up to half of the energy of these matrices (column "input"), so the table measures which model class retains a full-rank in-mask structure at a given parameter count rather than noise removal; the i.i.d. in-mask values are a model mismatch for the blockmodel, which is why it sits at the SVD-dof level at matched budget. The hierarchical pattern of Table 2 is omitted: as a binary mask a two-level hierarchy coincides with its coarsest level.

Table 10: Capacity control, DQ at n=200, σ=0.3, 20 seeds: ρ_M is the mask density, r' the matched SVD rank, k' the matched number of co-clusters, "input" the DQ of the noisy matrix itself.

Pattern	ρ_M	r'	k'	input	SVD-3	SVD-dof	RPCA	BM	BM-dof	**PatLR**	PatLR-rnd
Banded	0.035	7	51	0.57	0.05	0.10	0.07	0.04	0.27	**0.73**	0.52
Block	0.250	30	106	0.92	0.06	0.41	0.18	0.04	0.42	**0.96**	0.61
Arrowhead	0.015	5	42	0.13	0.28	0.28	0.05	0.27	0.43	**0.74**	0.29
Checkerboard	0.500	63	146	0.96	0.06	0.68	0.31	0.04	0.62	**0.97**	0.73
Block+band	0.327	40	120	0.93	0.06	0.51	0.22	0.04	0.42	**0.97**	0.64
Anti-diagonal	0.102	14	73	0.79	0.06	0.21	0.13	0.04	0.26	**0.83**	0.57
Triangular	0.502	63	146	0.96	0.07	0.79	0.40	0.02	0.81	**0.97**	0.67

E.5 Adaptive PatLR: Additional Analysis

E.5.1 Selection, Identification and Cost on Planted Matrices

Setup. Each of the nine masks of the default vocabulary is planted at n=200 with the generator of Figure 4 (i.i.d. Gaussian values of scale 2 on the mask and 0.15 off it), either alone (*pattern only*) or plus a rank-3 PSD component of equal energy (*pattern + rank-3*); the matrix is scrambled by a random permutation and symmetric Gaussian noise is added at SNR 4 and SNR 1 (10 and 20 seeds; 540 instances). The selector's vocabulary has 12 masks: the nine planted families and three gallery masks (parity checkerboard, anti-diagonal band of width 10, upper triangular). The parity checkerboard is a row/column-permuted copy of the two-block mask (reorder even and odd indices), so the density-normalized score cannot and should not separate the two; it is counted with block k=2 throughout. Methods: Oracle PatLR (rank 3, planted mask; 12 iterations, 3 restarts); Adaptive PatLR (Algorithm 3, N=80, ℓ=3, rank 3); brute force (every mask fitted by the LAP phase of

Alternating LAP, 15 iterations from 3 restarts, i.e. the search's LAP-only evaluation at a full budget per mask; the mask with the lowest density-normalized score kept, then PatLR with it); SVD at the rank with the highest DQ. DQ is measured against the clean matrix in its planted order. Times are single-core seconds on an Apple M3 (six instances at SNR 1).

Results (Table 11, Figure 5). Across the 540 instances the ratio $\mathrm{DQ}_{\mathrm{adapt}}/\mathrm{DQ}_{\mathrm{oracle}}$ has median 1.00, 5th percentile 0.92 and minimum 0.76 (a band w=4 instance at SNR 1, planted alone, for which block+band was selected); it exceeds 1 in a third of the instances. SVD at its best rank averages DQ 0.29 against the oracle's 0.68 at SNR 1 and 0.75 (the noisy input itself) against 0.87 at SNR 4. Brute force has median ratio 1.00 and minimum 0.74; its DQ is within 0.01 of the search's in every (family, generator, SNR) cell, and its selection frequencies agree with the search's within 10% in 29 of the 36 cells, the larger differences being between masks of equal DQ (block k=3 against block+band; on block k=2 with the rank-3 component at SNR 1, block k=6 against band w=4). It takes 3.7 s against 2.9 s for the search and 2.5 s for the oracle, and its cost grows with $|\mathcal{V}|$ where the search evaluates about N candidates whatever the vocabulary. Fitting every mask with the full solver (exchange phase included) before comparing scores is not better: the wide band then attains the lowest normalized score more often, and on block k=3, arrowhead and block+band with the rank-3 component at SNR 1 the selected mask has a lower DQ (0.53, 0.79, 0.53 against 0.61, 0.85, 0.62), the exact minimizer of the heuristic score not being the best-reconstructing mask (Appendix C.2). On six real matrices (the feature correlations of Iris, Wine, Breast Cancer and Digits, the karate club and Les Misérables), where no planted mask exists, the adaptively selected mask matches the best fixed mask of the vocabulary in explained variance within 0.025 in 24 of 29 (matrix, rank) cells, the exceptions being a sparser choice at rank $\leq$ 3 on Digits and rank $\leq$ 2 on Les Misérables. Identification (Figure 5a,b and the last columns of Table 11) splits by the geometry of the vocabulary. Arrowhead, block+band and the block templates k=4 and k=6 are identified in nearly every instance, although block k=4 fits inside block k=2 and block k=6 inside block k=3: under noise the denser container's smaller residual does not pay back its density penalty. The few misses go to a neighbouring block size and all occur at SNR 1. Block k=2 is identified whenever the pattern is planted alone and at SNR 4 with the rank-3 component; at SNR 1 with the rank-3 component the selector prefers a sparser mask (block k=6 or band w=4) at the oracle's DQ, the low-rank term absorbing the two dense blocks so that the density penalty of the half-dense mask is no longer paid back. The nested families go to a larger mask in most instances (narrow bands to the wide band; block k=3 to block+band, whose blocks cover those of block k=3 up to two rows), within 0.04 of the oracle's DQ. The remaining non-exact choices are the thin bands without a low-rank part: their order is recovered only partially at this budget, the planted mask then copies a misaligned band, and the selector prefers wider masks (block k=6, block+band), which tolerate the misalignment; at SNR 4 they reach a higher DQ than the planted mask and at SNR 1 a DQ 0.02–0.07 below it. With the rank-3 component the wide band is identified in most instances.

E.5.2 Sensitivity to the Residual Rank

Figure 9 shows DQ against the residual rank r on the setup of Figure 4 (n=200, the planted pattern alone and with a rank-3 component of equal energy, SNR 4 and 1, 20 seeds) for PatLR with the planted mask, SVD at the same rank and Adaptive PatLR. With a rank-3 component in the data, DQ peaks at the true rank for PatLR and declines slowly beyond it, while SVD needs a higher rank to reach a lower plateau. With the pattern alone every unit of rank costs DQ at SNR 1 (the low-rank term fits noise) and little at SNR 4. Adaptive PatLR follows PatLR within 0.05 at every rank; the largest differences are on the band planted alone, where it selects a wider mask (Appendix E.5.1) and lies 0.04 above PatLR at SNR 4 and 0.04–0.05 below it at SNR 1. The rank is a budget to be chosen, not a free improvement.

E.6 Ordering and Block-Structure Benchmarks: Full Results

This appendix gives the full grids behind Figure 3 and Sections 5.1–5.2, and two ordering geometries reported only here, circular seriation and nestedness (Appendix E.6.2). Counting the cells of each grid by the sign of the paired difference to the named reference (positive / interval containing zero / negative): latent ordering 18/2/4 against spectral and 9/10/5 against convex seriation (24 cells); consecutive ones 4/0/0; circular seriation 2/2/0; sparse SBM 8/1/3 against the Bethe Hessian (12 cells, the negative ones at λ=1.3); weighted partition 4/1/0 against spectral clustering; band-in-

Table 11: Adaptive pattern selection on planted matrices (n=200). DQ of Oracle PatLR (planted mask), Adaptive PatLR and SVD at its best rank, and the share of instances (%) in which Adaptive PatLR selects the planted mask (second number where present: a mask containing the planted one, a wider band for the bands and block k=3 for block k=6, or block+band for block k=3, whose blocks it covers up to two rows). At SNR 4 the best SVD rank is the full rank, i.e. the noisy input itself (DQ $= 1 - 1/\mathrm{SNR} = 0.75$). 10 seeds at SNR 4, 20 at SNR 1.

	SNR 4				SNR 1			
planted mask	Oracle	Adaptive	SVD	exact / cont.	Oracle	Adaptive	SVD	exact / cont.
pattern only								
block k=2	0.86	0.86	0.75	100	0.44	0.46	0.01	100
block k=3	0.89	0.89	0.75	40 / 60	0.60	0.60	0.01	45 / 55
block k=4	0.91	0.91	0.75	100	0.68	0.68	0.02	100
block k=6	0.92	0.92	0.75	100	0.75	0.74	0.03	95 / 5
band w=1	0.77	0.78	0.75	0 / 40	0.69	0.67	0.12	5 / 95
band w=2	0.77	0.79	0.75	0 / 60	0.67	0.63	0.09	10 / 55
band w=4	0.76	0.79	0.75	0	0.63	0.56	0.05	10
arrowhead	0.78	0.78	0.75	100	0.73	0.72	0.28	100
block+band	0.89	0.89	0.75	100	0.60	0.60	0.01	100
pattern + rank-3 component								
block k=2	0.86	0.80	0.75	100	0.46	0.47	0.49	0
block k=3	0.90	0.90	0.75	10 / 90	0.63	0.61	0.49	0 / 70
block k=4	0.92	0.92	0.75	100	0.71	0.70	0.49	90
block k=6	0.94	0.94	0.75	100	0.79	0.79	0.50	100
band w=1	0.88	0.88	0.75	0 / 100	0.81	0.77	0.51	10 / 90
band w=2	0.87	0.88	0.75	10 / 90	0.79	0.78	0.51	40 / 55
band w=4	0.86	0.85	0.75	60	0.75	0.74	0.50	90
arrowhead	0.89	0.89	0.76	100	0.85	0.85	0.63	100
block+band	0.90	0.90	0.75	100	0.63	0.62	0.49	70

block $16/0/0$ for the refining pass against the pipeline. In the tables, "seeds $+/-$" counts the seeds on which the paired difference is positive / negative.

E.6.1 Latent-order recovery

Setup. $B_{ij} = \exp(-(i-j)^2/2\ell^2)$ is a Robinson matrix with correlation length ℓ; it is scrambled by a uniformly random permutation p and symmetric Gaussian noise of standard deviation σ is added. The target is the banded mask of width $w = \ell$. The metric is $|\tau|$, the absolute Kendall correlation between the recovered and the planted latent positions (absolute because reversing an order preserves banded structure). Baselines: spectral seriation (Fiedler order of the Laplacian of $|A|$ [Atkins et al., 1998]); convex seriation [Fogel et al., 2013], implemented as Frank–Wolfe on the 2-SUM objective over the permutahedron with the concave vertex-pulling term $-\mu \operatorname{var}(y)$, the penalty weight $\mu \in \{0.1, 0.3, 0.5, 1, 2\}$ and the start (spectral or two random) selected by the 2-SUM value itself, never by τ (on noiseless inputs it recovers the planted order exactly); QAP-FAQ with three starts; and a single run of Alternating LAP initialized from the spectral order (AltLAP-r1; five restarts change no conclusion). Grid: $\ell \in \{6, 12, 25\}$, $n \in \{100, 300\}$, $\sigma \in \{0.05, 0.1, 0.2, 0.3\}$.

Results. Table 12 lists the 24 cells behind the counts of Section 5.1. The four negative cells are $\ell = w = 25$ at $n = 100$, where the band covers 44.5% of the matrix; the two ties are $\ell = 6$, $n = 300$ at $\sigma \geq 0.2$, where the correlation length is short relative to n, the global order is weakly identified and all methods are poor (τ between 0.25 and 0.34). Runtimes at $n = 300$ (one EPYC core): spectral 0.002 s, convex ≈ 0.25 s, FAQ 0.6–1.1 s, AltLAP-r1 0.1–0.2 s, of which the exchange phase is the larger part.

Mask width. The band width given to the solver is the correlation length of the generator. Table 13 repeats AltLAP-r1 with the width halved and doubled: a factor-two error in either direction changes τ by less than 0.01 in most cells and by up to 0.06 (doubled width at ℓ=12, n=100, σ=0.3), where the lead over convex seriation is lost; a four-fold overestimate (w=25 at ℓ=6, n=100) falls below the spectral order. Choosing the width by the density-normalized score of Section 4.1 is not an

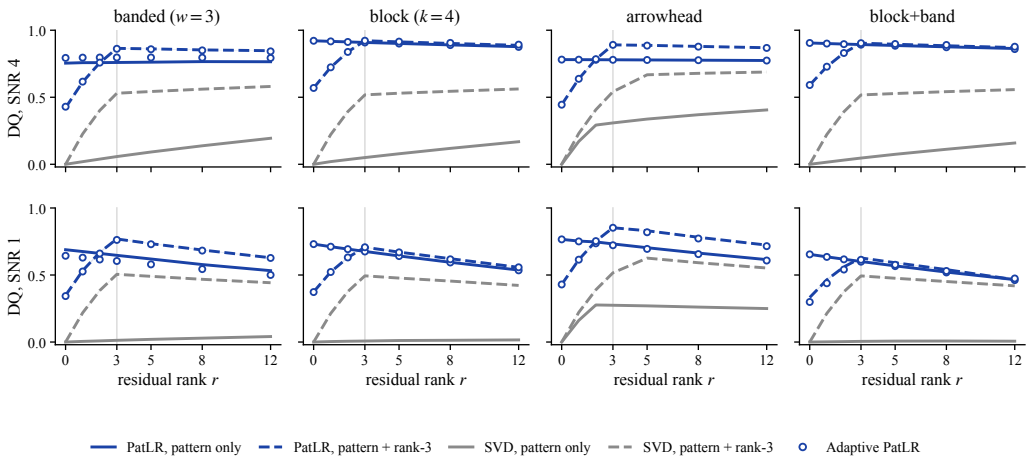

Figure 9: Denoising quality against the residual rank r on the setup of Figure 4 (20 seeds, 95% intervals shaded): PatLR with the planted mask and SVD at the same rank, for the planted pattern alone (solid) and with a rank-3 component (dashed); markers are Adaptive PatLR. The vertical line marks the planted rank 3.

Table 12: Latent-order recovery, all 24 cells: $|\tau|$ against the planted order (mean over 20 seeds; best of spectral, convex and ours in bold) and paired differences of AltLAP-r1 against spectral and convex seriation. FAQ optimizes the same pattern objective from its default starts.

ℓ	n, σ	Spectral	Convex	FAQ	**AltLAP-r1**	Δ vs spectral [CI], seeds +/−	Δ vs convex [CI], seeds +/−
6	100, 0.05	0.981	0.992	0.269	**1.000**	+0.019 [+0.017, +0.021], 20/0	+0.008 [+0.007, +0.008], 20/0
6	100, 0.1	0.943	0.975	0.357	**0.986**	+0.043 [+0.036, +0.051], 20/0	+0.011 [−0.001, +0.021], 17/3
6	100, 0.2	0.824	**0.905**	0.293	0.861	+0.037 [+0.020, +0.053], 17/3	−0.044 [−0.084, −0.003], 9/11
6	100, 0.3	0.673	0.700	0.225	**0.722**	+0.049 [+0.029, +0.072], 18/2	+0.022 [−0.007, +0.047], 15/5
6	300, 0.05	0.811	**0.861**	0.159	0.828	+0.017 [+0.011, +0.022], 18/2	−0.033 [−0.083, +0.015], 8/12
6	300, 0.1	0.543	0.549	0.165	**0.554**	+0.011 [+0.001, +0.020], 14/6	+0.004 [−0.072, +0.078], 8/12
6	300, 0.2	0.336	0.334	0.117	**0.337**	+0.002 [−0.006, +0.009], 9/11	+0.003 [−0.088, +0.080], 10/10
6	300, 0.3	0.250	**0.284**	0.109	0.256	+0.006 [−0.006, +0.017], 12/8	−0.029 [−0.097, +0.033], 10/10
12	100, 0.05	0.995	**0.997**	0.634	**0.997**	+0.002 [+0.002, +0.003], 17/1	−0.000 [−0.001, +0.001], 9/7
12	100, 0.1	0.983	0.988	0.592	**0.991**	+0.008 [+0.007, +0.009], 20/0	+0.003 [+0.002, +0.004], 18/2
12	100, 0.2	0.954	0.966	0.662	**0.981**	+0.027 [+0.023, +0.030], 20/0	+0.015 [+0.013, +0.017], 20/0
12	100, 0.3	0.921	0.939	0.658	**0.972**	+0.051 [+0.045, +0.058], 20/0	+0.033 [+0.030, +0.036], 20/0
12	300, 0.05	0.980	0.990	0.241	**0.999**	+0.019 [+0.018, +0.020], 20/0	+0.010 [+0.009, +0.010], 20/0
12	300, 0.1	0.943	0.966	0.326	**0.977**	+0.034 [+0.026, +0.041], 19/1	+0.011 [−0.001, +0.020], 16/4
12	300, 0.2	0.839	0.861	0.264	**0.889**	+0.050 [+0.032, +0.065], 18/2	+0.028 [−0.010, +0.052], 17/3
12	300, 0.3	0.709	**0.787**	0.169	0.750	+0.041 [+0.018, +0.062], 17/3	−0.037 [−0.133, +0.026], 14/6
25	100, 0.05	**0.999**	**0.999**	0.990	0.991	−0.008 [−0.008, −0.007], 0/20	−0.008 [−0.008, −0.007], 0/20
25	100, 0.1	**0.993**	**0.993**	0.979	0.980	−0.012 [−0.013, −0.011], 0/20	−0.012 [−0.013, −0.011], 0/20
25	100, 0.2	**0.978**	**0.978**	0.960	0.962	−0.016 [−0.018, −0.013], 0/20	−0.016 [−0.018, −0.014], 0/20
25	100, 0.3	0.960	**0.961**	0.942	0.945	−0.015 [−0.018, −0.012], 0/20	−0.016 [−0.018, −0.013], 0/20
25	300, 0.05	0.992	0.996	0.497	**0.997**	+0.005 [+0.004, +0.005], 20/0	+0.001 [+0.001, +0.002], 20/0
25	300, 0.1	0.982	0.988	0.560	**0.993**	+0.012 [+0.011, +0.013], 20/0	+0.006 [+0.005, +0.006], 20/0
25	300, 0.2	0.959	0.971	0.516	**0.988**	+0.029 [+0.027, +0.031], 20/0	+0.017 [+0.017, +0.018], 20/0
25	300, 0.3	0.933	0.950	0.492	**0.983**	+0.050 [+0.045, +0.054], 20/0	+0.033 [+0.031, +0.034], 20/0

option on these smooth kernels: the score never selects a width below the correlation length and picks a wider candidate whenever one is available, so w is a user input here.

E.6.2 Consecutive ones, circular seriation, and nestedness

Consecutive ones (fixed-width staircase recovery). The consecutive-ones property (Section 5.1) is decided in linear time by PQ-trees [Booth and Lueker, 1976], but recognition is brittle (one corrupted entry destroys the property), so the applied literature, from physical mapping to archaeological seriation [Kendall, 1971], orders the matrix with heuristics: barycentre reordering, spectral [Atkins et al., 1998] and convex [Fogel et al., 2013] seriation. Our target is the staircase band $M_{ij} = \mathbf{1}[s_i \le j < s_i + w]$ with non-decreasing starts s_i, fitted with independent row and column

Table 13: Latent-order recovery with the band width halved and doubled ($|\tau|$, mean over 20 seeds; the paper's width $w{=}\ell$ in bold). At $n{=}100$ the doubled width 50 covers the matrix and is omitted.

ℓ	n, σ	Spectral	Convex	$w{=}\ell/2$	$w{=}\ell$	$w{=}2\ell$
6	100, 0.1	0.943	0.975	0.981	**0.986**	0.983
6	100, 0.3	0.673	0.700	0.710	**0.722**	0.726
6	300, 0.1	0.543	0.549	0.547	**0.554**	0.553
6	300, 0.3	0.250	0.284	0.255	**0.256**	0.261
12	100, 0.1	0.983	0.988	0.989	**0.991**	0.972
12	100, 0.3	0.921	0.939	0.967	**0.972**	0.913
12	300, 0.1	0.943	0.966	0.975	**0.977**	0.971
12	300, 0.3	0.709	0.787	0.748	**0.750**	0.759
25	100, 0.1	0.993	0.993	0.977	**0.980**	–
25	100, 0.3	0.960	0.961	0.939	**0.945**	–
25	300, 0.1	0.982	0.988	0.992	**0.993**	0.986
25	300, 0.3	0.933	0.950	0.978	**0.983**	0.958

permutations; for a matrix whose rows all carry w ones, A has the C1P iff $d_M(A) = 0$ for some mask of this family (ordering rows by the start of their interval makes the starts non-decreasing). General C1P with varying interval lengths is a larger class and would require a search over the start and length profiles; what we test is the noisy recovery of the fixed-width model, with the evenly spaced profile s_i and its sensitivity below. Instances: planted 60×60 staircases with $w = 12$ and independent flip noise at rate f, 20 seeds. Our solver runs from random restarts only and keeps the candidate with the best mask fit, which uses no labels. Metrics: $|\tau|$ of the recovered row and column orders against the planted ones, and *edits*, the minimum number of entries that must be changed to make every row contiguous under the returned ordering.

Table 14: Noisy consecutive ones (60×60, $w = 12$, 20 seeds): row $|\tau|$ / column $|\tau|$ / edits. Controls use the identical protocol and restart budget with a random or a block mask of the same density.

Method	$f{=}0.02$	$f{=}0.05$	$f{=}0.10$	$f{=}0.15$
Sort by marginals	0.05 / 0.04 / 582	0.05 / 0.05 / 638	0.06 / 0.06 / 734	0.07 / 0.06 / 837
Barycentre reordering	0.73 / 0.66 / 295	0.61 / 0.51 / 448	0.53 / 0.42 / 608	0.46 / 0.36 / 732
Spectral seriation	0.90 / 0.87 / 203	0.84 / 0.66 / 447	0.67 / 0.44 / 659	0.49 / 0.37 / 790
Convex seriation	0.92 / 0.92 / 163	0.85 / 0.85 / 312	0.71 / 0.69 / 550	0.50 / 0.53 / 718
Spectral + 2-SUM local search	0.92 / 0.88 / 181	0.85 / 0.83 / 332	0.71 / 0.61 / 570	0.49 / 0.53 / 721
Staircase mask, AltLAP (cold)	**0.99 / 1.00 / 66**	**0.99 / 0.99 / 164**	**0.98 / 0.97 / 335**	**0.95 / 0.91 / 502**
Spectral, then AltLAP	0.99 / 1.00 / 67	0.98 / 0.97 / 168	0.83 / 0.78 / 368	0.56 / 0.53 / 550
Control: random mask (cold)	0.06 / 0.06 / 617	0.07 / 0.07 / 672	0.05 / 0.07 / 765	0.08 / 0.09 / 859
Control: block mask (cold)	0.28 / 0.25 / 322	0.31 / 0.29 / 402	0.34 / 0.31 / 537	0.24 / 0.21 / 666

Paired against convex seriation, the column τ of the staircase fit is higher by $+0.075$ $[+0.063, +0.088]$, $+0.136$ $[+0.118, +0.157]$, $+0.280$ $[+0.220, +0.358]$ and $+0.377$ $[+0.300, +0.469]$ at $f = 0.02$, 0.05, 0.10, 0.15, and the edit count lower by 97 $[89, 106]$, 148 $[137, 160]$, 215 $[194, 238]$ and 216 $[202, 232]$, on 20 of 20 seeds at every level. An insertion local search on the 2-SUM criterion needs 114–201 more edits than the staircase fit on 20 of 20 seeds, so the gain is not produced by refinement as such. Row totals are constant by construction and column totals vary only near the boundary, which is why sorting by marginals carries almost no information. Runtimes: 0.03 s spectral, 0.18 s convex, 0.3–0.5 s local search, 0.5–1 s ours.

Table 15 repeats the comparison with the shape of the matrix, the start profile and the interval lengths varied. The fixed staircase keeps the higher τ and the lower edit count for rectangular matrices, for an irregular start profile and for interval lengths jittered by up to ± 8, with the smallest margins under the irregular profile and the largest jitter. Re-estimating the mask from the data after the cold fit, one interval per row (the interval-envelope pattern of Figure 1), lowers the edit count slightly and restores the row order under the irregular profile.

Circular seriation. A latent order can be cyclic rather than linear: similarities between points on a closed curve are banded modulo n, and spectral seriation needs a dedicated circular embedding for this case [Recanati et al., 2018]. The question is whether the same solver covers the cyclic

Table 15: Noisy consecutive ones under changes of the generator (20 seeds): column $|\tau|$ / edits for convex seriation, the fixed staircase mask and the mask re-estimated as one interval per row; "noise" is the number of entries flipped during corruption.

	f=0.05				f=0.15			
Setting	noise	convex	fixed mask	re-estimated	noise	convex	fixed mask	re-estimated
60×60, w=12 (as Table 14)	179	0.86 / 317	1.00 / 166	1.00 / 165	540	0.51 / 716	0.92 / 496	0.93 / 494
40×80, w=16	161	0.80 / 328	0.98 / 155	0.98 / 153	483	0.49 / 669	0.77 / 461	0.77 / 451
80×40, w=8	161	0.88 / 247	1.00 / 145	1.00 / 145	483	0.61 / 574	0.97 / 430	0.97 / 430
irregular start profile	179	0.71 / 334	0.85 / 188	0.85 / 184	540	0.51 / 695	0.80 / 510	0.80 / 507
interval lengths 12 ± 4	179	0.86 / 304	0.96 / 189	0.97 / 182	540	0.53 / 704	0.86 / 515	0.85 / 510
interval lengths 12 ± 8	179	0.87 / 291	0.91 / 205	0.93 / 193	540	0.65 / 664	0.78 / 527	0.80 / 519

geometry by changing only the mask. For $n = 120$ points on a circle with a decaying kernel and Gaussian noise, the target is the cyclic band mask of width 10 and the reference is the spectral circular embedding; the metric is circular Kendall τ up to rotation and reflection (20 seeds). Ours is the better of 128 cold restarts and a coarse-to-fine run, chosen by the objective. In Table 16 both methods recover the order at low noise (the difference is within ± 0.01); at noise 0.5 and 0.7 the cyclic band is ahead ($+0.035$ and $+0.103$, on 13 and 18 of 20 seeds), and at noise 1.0 neither recovers the order and the interval contains zero. The linear spectral order is far below throughout, so the cyclic mask, not a generic ordering, carries the result; the gain over the circular embedding is concentrated at moderate noise.

Table 16: Circular seriation (n=120, cyclic band w=10): circular Kendall τ, mean over 20 seeds, and the paired difference of ours against the circular spectral embedding.

Noise	Spectral (linear)	Circ. spectral	Circ. spectral → AltLAP	**AltLAP, cyclic (cold)**	Δ vs circ. spectral [CI]	seeds +/−
0.3	0.372	0.963	**0.965**	0.958	-0.005 [-0.015, $+0.006$]	6/13
0.5	0.393	0.903	0.934	**0.938**	$+0.035$ [$+0.014$, $+0.055$]	13/7
0.7	0.332	0.782	0.879	**0.885**	$+0.103$ [$+0.065$, $+0.138$]	18/2
1.0	0.203	0.431	**0.471**	0.469	$+0.039$ [-0.072, $+0.142$]	11/9

Nestedness: a boundary case. In a nested (Ferrers) matrix the ones of each row contain those of every row with a smaller degree, the structure of species-by-site and similar ecological matrices [Brualdi and Sanderson, 1999]. The question here is whether a general reordering objective adds anything where a one-pass specialized method exists: sorting rows and columns by their marginals recovers a noiseless nested matrix exactly and, although under flip noise it need not minimize our objective, it is the reference. For planted 60×80 Ferrers matrices with flip rate f, the two-sided AltLAP uses the nested mask with its profile estimated from the row degrees (no oracle information); Table 17 gives the column $|\tau|$ and the flips to a nested matrix (20 seeds). The mask fit reproduces the degree order at $f \leq 0.02$ and falls below it from $f = 0.05$ on, by 0.06 in τ at $f = 0.15$ with a similar flip count; warm-starting from the degree order does not help, and a larger restart budget does not change the values. Where degree sorting already recovers the structure, the mask objective adds nothing; this is the limitation recorded in Section 6.

Table 17: Nestedness (60×80 planted Ferrers matrices, flip rate f, 20 seeds): column $|\tau|$ / flips to a nested matrix under the returned order (lower is better).

f	Degree sort	Spectral	AltLAP, nested mask	Degree sort → AltLAP
0.0	0.994 / 0	0.586 / 1277	0.994 / 0	0.994 / 0
0.02	0.967 / 134	0.573 / 1382	0.965 / 129	0.963 / 131
0.05	0.942 / 301	0.544 / 1658	0.932 / 299	0.925 / 301
0.1	0.906 / 574	0.548 / 1712	0.881 / 586	0.862 / 587
0.15	0.876 / 819	0.507 / 1931	0.817 / 843	0.806 / 842

E.6.3 Block models against the state of the art

Sparse binary SBM at the detectability scale. Symmetric SBM with $k = 4$ equal blocks, $n = 500$, $c_{\text{in}} = c + (k-1)\lambda\sqrt{c}$ and $c_{\text{out}} = c - \lambda\sqrt{c}$, so $\lambda = 1$ is the asymptotic (Kesten–Stigum) detectability threshold [Decelle et al., 2011]; mean degree $c \in \{8, 16, 32\}$, 20 seeds. Panel: Bethe Hessian

$H(r) = (r^2 - 1)I - rA + D$ with $r = \sqrt{c}$ [Saade et al., 2014]; the non-backtracking operator in Ihara–Bass form [Krzakala et al., 2013]; regularized spectral clustering [Qin and Rohe, 2013]; Leiden [Traag et al., 2019], which estimates k itself; the block mask fitted by Alternating LAP from eight random starts, the LAP and exchange phases interleaved per start (three rounds) and the best start kept by the criterion ("ours"); and the same run started from the Bethe-Hessian partition. All methods except Leiden receive the true k; our mask also encodes equal block sizes, which the planted model satisfies and the spectral methods do not use. On sparse $\{0, 1\}$ graphs the LAP phase alone stalls early (ARI 0.56 at $\lambda = 1.6$, $c = 16$, with the exchange phase disabled, 32 starts); with the exchange phase the fit reaches 0.82.

Table 18: Sparse SBM, $k = 4$, $n = 500$, mean ARI over 20 seeds. "Ours" is Alternating LAP from random starts; "swap only" is a Kernighan–Lin-style exchange search alone from random balanced partitions (same criterion and block sizes); the paired difference is against the Bethe Hessian, with the best panel method per row in bold.

λ	c	Bethe H.	Non-backtr.	Reg. sp.	Leiden	Swap only	**Ours**	BH $\to$ ours	Δ vs BH [CI]	seeds +/−
1.3	8	0.42	0.38	0.41	0.13	0.05	0.15	**0.47**	-0.272 [-0.296, -0.248]	0/20
1.3	16	0.43	0.39	0.35	0.15	0.08	0.22	**0.51**	-0.208 [-0.249, -0.167]	1/19
1.3	32	0.47	0.39	0.40	0.13	0.11	0.35	**0.57**	-0.116 [-0.168, -0.059]	2/18
1.6	8	0.73	0.72	0.73	0.61	0.13	0.70	**0.78**	-0.034 [-0.083, $+0.009$]	11/9
1.6	16	0.74	0.74	0.75	0.71	0.32	**0.82**	**0.82**	$+0.080$ [$+0.069$, $+0.093$]	20/0
1.6	32	0.78	0.78	0.78	0.83	0.58	**0.86**	**0.86**	$+0.079$ [$+0.063$, $+0.095$]	18/2
2.0	8	0.92	0.91	0.92	0.94	0.79	0.94	**0.95**	$+0.022$ [$+0.013$, $+0.030$]	18/2
2.0	16	0.92	0.92	0.92	**0.95**	0.94	**0.95**	**0.95**	$+0.028$ [$+0.021$, $+0.036$]	20/0
2.0	32	0.94	0.94	0.94	**0.97**	0.96	**0.97**	**0.97**	$+0.027$ [$+0.019$, $+0.034$]	19/1
2.5	8	0.99	0.98	0.99	**0.99**	0.98	**1.00**	**1.00**	$+0.009$ [$+0.006$, $+0.012$]	18/2
2.5	16	0.99	0.99	0.99	**0.99**	**0.99**	**1.00**	**1.00**	$+0.007$ [$+0.004$, $+0.010$]	17/1
2.5	32	0.99	0.99	0.99	**1.00**	**1.00**	**1.00**	**1.00**	$+0.009$ [$+0.006$, $+0.013$]	18/0

With equal block sizes the number of within-block pairs is fixed, so maximizing the within-block edge count, which is our criterion on a $\{0, 1\}$ matrix, is the maximum-likelihood objective for the planted partition at any n and any $p > q$ (for $p < q$ the sign reverses; for $k = 2$ it is minimum bisection; Abbe, 2018). The crossover of Section 5.2 moves with the mean degree (Table 18). Below it the criterion values are 0.33 (ours) against 0.24 (Bethe-Hessian partition) in error reduction at $c = 16$, $\lambda = 1.3$, with the ARIs in the opposite order; the gap between the MAP and the Bayes-optimal (belief-propagation) estimator near the threshold is the reason belief propagation and its spectral linearizations were introduced [Decelle et al., 2011, Krzakala et al., 2013, Saade et al., 2014]. We do not establish inconsistency of the criterion at this n, nor that the search reaches its global optimum. Refining the Bethe-Hessian partition with our criterion costs 0.3–0.5 s on top of its 0.04 s.

Weighted planted partition. $A = \mu \cdot \mathbf{1}[\text{same block}] + E$ with $\mu > 0$ and E symmetric with i.i.d. $\mathcal{N}(0, \sigma^2)$ upper-triangular entries, $\sigma = 1$, $n = 300$, $k = 4$ equal blocks, 20 seeds. This is the regime of the block experiments in the paper, where minimizing the *signed* off-mask mass is the maximum-likelihood criterion for the partition. Spectral clustering on the weighted matrix is the natural baseline; the graph methods are defined for graphs and receive the matrix sparsified at its strongest 10% of entries. The squared loss rewards large negative off-mask entries as if they were structure, which is why it fails on this generator, whose blocks differ in mean but not in variance (the ablation column); the linear loss, which on a $\{0, 1\}$ matrix coincides with the squared one, is the likelihood here. A Kernighan–Lin-style swap search on the same linear criterion tracks the cold run from μ=0.4 on and lies 0.02–0.03 below it at the two weakest signals, so on these dense matrices the criterion, not the particular optimizer, carries the result.

At the weakest signal the cold run is within the interval of spectral clustering and the spectral-started run is the entry that pays: $+0.154$ [$+0.118$, $+0.188$] and $+0.159$ [$+0.135$, $+0.182$] over spectral clustering at $\mu = 0.30$ and 0.35.

Block sizes. The block mask fixes the block sizes, which above are the planted (equal) ones. Table 20 plants blocks of sizes 1:1:2:2 as well and gives the solver the equal-size mask, the true sizes, or sizes estimated as the cluster sizes of the size-free spectral method in the same panel (regularized spectral clustering on the graphs, spectral clustering on the weighted matrices; no labels), next to the size-free baselines, the Kernighan–Lin-style swap search on the same criterion with the true sizes (KL) and Alternating LAP started from the Bethe-Hessian partition. With equal planted blocks the

Table 19: Weighted planted partition ($n = 300$, $k = 4$, $\sigma = 1$), mean ARI over 20 seeds; paired difference of ours (AltLAP-lin, cold start) against spectral clustering. "Squared" is AltLAP on the squared loss of Definition 1 (the ablation row); "KL (lin.)" is a Kernighan–Lin-style swap search on the same linear criterion from random balanced starts, an alternative optimizer of our objective.

μ	Spectral	Reg. sp.	Bethe H.	Non-b.	Leiden	Squared	KL (lin.)	**AltLAP-lin**	Spec. $\to$ ours	Δ vs spectral [CI]	seeds +/−
0.30	0.403	0.028	0.064	0.052	0.033	0.003	0.324	0.356	**0.556**	-0.046 [-0.111, $+0.024$]	7/13
0.35	0.680	0.073	0.119	0.102	0.054	0.004	0.807	0.827	**0.838**	$+0.147$ [$+0.126$, $+0.169$]	20/0
0.40	0.850	0.154	0.218	0.179	0.092	0.006	0.927	0.932	**0.932**	$+0.081$ [$+0.058$, $+0.105$]	19/1
0.50	0.975	0.482	0.560	0.445	0.354	0.018	**0.994**	**0.995**	**0.995**	$+0.020$ [$+0.014$, $+0.027$]	18/1
0.60	0.996	0.821	0.814	0.813	0.866	0.050	**1.000**	**1.000**	**1.000**	$+0.004$ [$+0.002$, $+0.008$]	7/0

estimated sizes cost almost nothing. With unequal blocks the equal-size mask is misspecified and falls below the size-free methods, the true sizes give the best result in the panel, and the estimated sizes recover most of the gap, above the partition they were taken from. The swap search from random starts needs the dense signal of the weighted matrices and stays low on the sparse graphs, so the LAP steps, not the exchanges alone, do the work there.

Table 20: Block recovery with the block sizes in the mask equal, true, or estimated from a spectral partition (ARI). Top: sparse SBM, n=500, k=4, c=16, 20 seeds, ours = Alternating LAP. Bottom: weighted planted partition, n=300, k=4, σ=1, 10 seeds, 16 restarts, ours = AltLAP-lin.

							Ours, mask sizes		
Planted sizes	λ	Bethe H.	Reg. spectral	Leiden	KL (true)	BH $\to$ ours (true)	equal	true	estimated
equal	1.6	0.74	0.75	0.74	0.32	0.79	0.82	0.82	0.80
equal	2.0	0.92	0.92	0.95	0.94	0.95	0.95	0.95	0.94
1:1:2:2	1.6	0.59	0.54	0.52	0.66	0.52	0.49	0.84	0.65
1:1:2:2	2.0	0.89	0.85	0.93	0.92	0.68	0.59	0.96	0.89
Planted sizes	μ	Spectral	KL-lin (true)	Squared loss (true)			equal	true	estimated
equal	0.4	0.84	0.93	0.00			0.93	0.93	0.89
equal	0.6	0.99	1.00	0.03			1.00	1.00	0.99
1:1:2:2	0.4	0.62	0.94	0.01			0.55	0.94	0.79
1:1:2:2	0.6	0.98	1.00	0.07			0.63	1.00	0.98

Real labelled networks. The same panel on standard networks with published communities (Table 21; equal-size masks, or "+sizes": block sizes re-estimated from the order by change-point segmentation, three rounds, no oracle information). Reference partitions: the two factions of Zachary's karate club [Zachary, 1977]; the two groups of nine women of the Davis Southern Women data [Davis et al., 1941, Freeman, 2003], with the bipartite women–events graph used as a symmetric adjacency on all 32 nodes and ARI computed on the 18 women; the three classes of the political-books network (compiled by V. Krebs; Newman, 2006) and the twelve conferences of the football network [Girvan and Newman, 2002]. All methods receive k; PatLR uses rank 1; results are means over 10 seeds (standard deviations ≤ 0.03, zero for the deterministic methods). Les Misérables has no published partition (the $\mathrm{block}(19, 19, 19, 20)$ mask of Section 5.3 is an equal split, not a ground truth) and is not scored here. Two details behind Section 5.2: on Karate the low-rank term of PatLR carries the degree profile, which the plain block fit has to place inside the blocks; on polbooks ($49/43/13$) the equal-size mask improves the error reduction while the ARI does not move, a balanced partition being a different problem from community detection.

E.6.4 Composite target: a band inside blocks

Planted matrices with $k \in \{3, 5\}$ blocks (equal or unequal sizes), a band of width $w \in \{1, 3\}$ inside each block, noise $\sigma \in \{0.1, 0.3\}$, $n = 150$, 20 seeds: 16 configurations. The target mask is the band *inside* the blocks (block AND band); the union mask would be blind to the within-block order. Metrics: error reduction on the target mask, ARI of the block assignment read off the order, and within-block Kendall τ. Methods: spectral clustering (blocks only), global Fiedler ordering, a two-stage pipeline (cluster, then spectral order within each cluster), QAP-FAQ and AltLAP on the composite mask, AltLAP with a change-point estimate of the block boundaries (for unequal sizes), and AltLAP warm-started from the two-stage pipeline.

Table 21: Community recovery on labelled networks (ARI). "+sizes": block sizes re-estimated from the order (change-point segmentation) instead of equal sizes; "BH →" initializes from the Bethe-Hessian partition. PatLR uses rank 1.

Method	Karate (k=2)	Davis (k=2)	Polbooks (k=3)	Football (k=12)
Spectral clustering	0.882	0.000	0.465	0.813
Regularized spectral	0.882	0.000	0.624	0.869
Bethe Hessian	0.882	**1.000**	**0.659**	0.897
Non-backtracking	0.882	**1.000**	0.405	**0.906**
Leiden	0.465	0.405	0.621	0.807
AltLAP, equal blocks	0.772	0.777	0.455	0.739
AltLAP + sizes	0.772	**1.000**	0.473	0.733
BH → AltLAP	0.772	0.777	0.479	0.763
BH → AltLAP + sizes	0.772	0.777	0.479	0.793
PatLR, equal blocks	**1.000**	0.777	0.492	0.743
PatLR + sizes	**1.000**	0.777	0.476	0.775
BH → PatLR	0.772	**1.000**	0.477	0.752
BH → PatLR + sizes	0.772	**1.000**	0.481	0.764

Table 22: Composite band-in-block target, averages over the 16 configurations and 20 seeds; per-configuration values in Table 23.

Method	target ER	block ARI	within-block τ
Spectral clustering	0.057	**1.000**	0.09
Fiedler ordering	0.050	0.839	0.10
Two-stage (cluster, then order)	0.115	**1.000**	0.45
QAP-FAQ (composite mask)	0.137	0.181	0.18
AltLAP (composite mask)	0.171	0.781	0.20
AltLAP + boundary estimation	0.173	0.877	0.20
Two-stage, then AltLAP	**0.190**	**1.000**	**0.47**
Two-stage, convex seriation inside	0.124	**1.000**	0.46
Two-stage (convex), then AltLAP	**0.193**	**1.000**	**0.48**

Table 22 averages over the configurations and Table 23 gives each one with its paired interval; the summary is in Section 5.2. Details: the refining pass re-estimates the block boundaries from the order by change-point segmentation, so the block sizes are not given; it raises the target error reduction by $+17\%$ to $+257\%$ relative (median $+96\%$), from a median 45% of the planted order's value for the pipeline to 94% for the refined run (90–99% at w=3, σ=0.1), and by $+11\%$ to $+227\%$ when the pipeline uses convex seriation inside the clusters; the within-block τ gain at w=3 is $+0.02$ to $+0.09$ with the interval above zero in 7 of 8 configurations, and at w=1 it is within ± 0.02, a band of width one determining little of the order. From a cold start the composite mask alone gives block ARI 0.78, and 0.88 with the boundary estimate (e.g. $0.45 \to 0.89$ on $k = 3$, unequal, $w = 3$, $\sigma = 0.3$); for the within-block order the hard mask truncates the smooth decay, and the pipeline's seriation is the better starting point, which is why the two are combined. FAQ from its default starts does not recover the blocks (Table 22).

E.7 Attention Head Analysis: Full Details

This appendix gives the pipeline behind Section 5.5, the per-model results, and the identity-order control. All numbers come from one run of the unified scoring described below.

E.7.1 Score, Vocabulary and a Synthetic Check

The score $\tilde{s}_M(A)$ of (6) on a causal attention matrix (rows queries, columns keys) is the minimum mass outside the mask after a common reordering of queries and keys, normalized by $\|A\|_F^2(1-\rho_M)^{3/2}$ (equation (6)). The strict upper triangle is identically zero and contributes nothing to d_M for any mask and permutation, so no symmetrization is needed. The masks are the plain geometric masks of Section 2, not their intersection with the lower triangle (the arrowhead is the full hub row, hub column and diagonal, which is what the closed form of Appendix C.3 assumes); a

Table 23: Composite target per configuration (20 seeds): the two-stage pipeline (spectral clustering, spectral seriation inside each cluster) and the same pipeline followed by one AltLAP pass on the band-in-block mask; "oracle" is the error reduction of the planted order on that mask. Paired differences with 95% bootstrap intervals; block ARI is 1.00 for both in every configuration. The last two columns repeat the ER for the pipeline with convex seriation inside the clusters .

			target ER					within-block τ			ER, convex inside	
k	w	σ	oracle	pipel.	+Alt	rel.	Δ [CI]	pipel.	+Alt	Δ [CI]	pipel.	+Alt
3 eq.	1	0.1	0.130	0.043	0.104	+142%	$+0.061$ [$+0.060, +0.062$]	0.31	0.32	$+0.007$ [$-0.009, +0.023$]	0.045	0.105
3 eq.	1	0.3	0.086	0.025	0.084	+239%	$+0.059$ [$+0.058, +0.061$]	0.17	0.16	-0.009 [$-0.029, +0.010$]	0.026	0.084
3 eq.	3	0.1	0.305	0.196	0.288	+47%	$+0.093$ [$+0.088, +0.097$]	0.71	0.76	$+0.054$ [$+0.038, +0.071$]	0.215	0.290
3 eq.	3	0.3	0.214	0.100	0.200	+100%	$+0.100$ [$+0.096, +0.103$]	0.41	0.43	$+0.024$ [$-0.001, +0.050$]	0.100	0.200
3 uneq.	1	0.1	0.116	0.036	0.091	+154%	$+0.055$ [$+0.053, +0.057$]	0.29	0.30	$+0.003$ [$-0.010, +0.015$]	0.038	0.093
3 uneq.	1	0.3	0.079	0.022	0.077	+257%	$+0.055$ [$+0.054, +0.057$]	0.17	0.15	-0.018 [$-0.039, +0.004$]	0.024	0.077
3 uneq.	3	0.1	0.276	0.163	0.248	+53%	$+0.085$ [$+0.082, +0.089$]	0.63	0.68	$+0.055$ [$+0.041, +0.071$]	0.178	0.257
3 uneq.	3	0.3	0.199	0.083	0.179	+115%	$+0.096$ [$+0.093, +0.098$]	0.40	0.44	$+0.030$ [$+0.013, +0.050$]	0.090	0.185
5 eq.	1	0.1	0.193	0.090	0.168	+87%	$+0.078$ [$+0.076, +0.080$]	0.44	0.46	$+0.019$ [$+0.009, +0.028$]	0.096	0.166
5 eq.	1	0.3	0.110	0.043	0.108	+153%	$+0.065$ [$+0.063, +0.068$]	0.24	0.22	-0.021 [$-0.037, -0.004$]	0.046	0.108
5 eq.	3	0.1	0.415	0.353	0.413	+17%	$+0.060$ [$+0.055, +0.065$]	0.87	0.96	$+0.089$ [$+0.080, +0.099$]	0.372	0.414
5 eq.	3	0.3	0.263	0.173	0.260	+50%	$+0.087$ [$+0.084, +0.090$]	0.56	0.63	$+0.065$ [$+0.048, +0.082$]	0.183	0.262
5 uneq.	1	0.1	0.166	0.072	0.138	+93%	$+0.067$ [$+0.065, +0.068$]	0.40	0.41	$+0.013$ [$+0.003, +0.023$]	0.076	0.140
5 uneq.	1	0.3	0.100	0.036	0.097	+168%	$+0.061$ [$+0.060, +0.062$]	0.21	0.21	-0.006 [$-0.026, +0.014$]	0.040	0.099
5 uneq.	3	0.1	0.369	0.272	0.347	+28%	$+0.076$ [$+0.072, +0.080$]	0.80	0.86	$+0.062$ [$+0.050, +0.071$]	0.303	0.361
5 uneq.	3	0.3	0.243	0.142	0.230	+62%	$+0.088$ [$+0.084, +0.091$]	0.51	0.55	$+0.046$ [$+0.017, +0.074$]	0.151	0.238

common permutation does not preserve the causal triangle itself, and the structural labels describe the reordered matrix. The nine masks of Section 5.5 were fixed ahead of time to span the regimes of the standard taxonomy of attention [Clark et al., 2019, Voita et al., 2019, Xiao et al., 2024]: local routing (bands of width 1, 2, 4), segment- or topic-level grouping (blocks, $k = 2, 3, 4$), sink-like global routing of every query to a few distinguished keys such as BOS (arrowhead), modular-local heads (three blocks plus the band $|i - j| \leq 2$) and pure self-loops (diagonal). The vocabulary is a controlled basis set, not an exhaustive one. At n=512 the block+band mask (blocks of 170 plus a remainder block of 2, plus the band) differs from block-k3 (blocks of 170, 170, 172) in 0.3% of the entries, so near-ties between the two labels are expected and the two are read together below.

Synthetic check. Four head types with a constructed ground truth are scored with the full pipeline at n=128 (20 instances each, scrambled by a random permutation; Table 24, Figure 10): a local head (Gaussian kernel of width 1.5 tokens plus noise, so that the band $|i - j| \leq 4$ holds 99% of the kernel mass), a topic head (three equal blocks plus noise), a global head (every query attends the first key, plus a diagonal and noise) and a self head (a dominant diagonal plus noise). The global and topic heads are identified in every instance, the topic head either as block-k3 or as the block+band mask that contains its blocks; the local head is identified in most instances, with block masks selected otherwise at a small margin (the noise carries two thirds of the energy of these instances). The self head is a boundary case of the vocabulary: every mask that contains the diagonal fits it, the optimized thin bands pack slightly more noise than the diagonal alone, and the label is a thin band in every instance, by a small margin. A pure self-loop head is therefore reported by this pipeline as a thin band, and the band labels of real heads should be read as covering it.

Table 24: Synthetic attention heads scored with the paper's pipeline (9-mask vocabulary, density-normalized score, identity candidate, n=128, 20 instances per type): instances in which the true mask is the label, the labels selected, and the median relative margin of the selected mask over the runner-up.

Head type	True mask	True mask selected	Labels selected	Margin
Local	band w=4	15/20	band w=4 (15), block+band (3), block k=3 (1), block k=4 (1)	0.03
Topic	block k=3	12/20	block k=3 (12), block+band (8)	0.56
Global	arrowhead	20/20	arrowhead (20)	0.60
Self	diagonal	0/20	band w=4 (20)	0.02

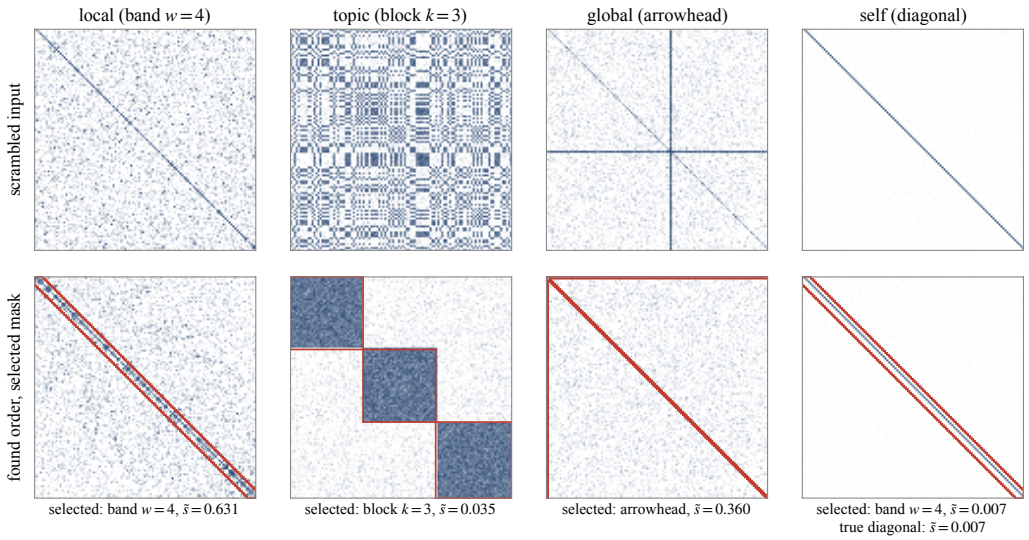

Figure 10: Synthetic attention heads, one instance per type: the scrambled input (top) and the same matrix in the order returned for the selected mask, mask outlined (bottom), with the density-normalized scores of the selected and, where different, of the true mask.

E.7.2 Pipeline

Models and prompts. GPT-2 [117M parameters, 12 layers $\times$ 12 heads; Radford et al., 2019], Qwen2.5-0.5B (24×14) and Qwen2.5-7B (28×28) [Qwen Team, 2024], loaded from the Hugging Face hub (`gpt2`, `Qwen/Qwen2.5-0.5B`, `Qwen/Qwen2.5-7B`). The prompts are the first 20 passages of at least $5n$ characters formed from consecutive non-heading lines of the WikiText-2 validation split [Merity et al., 2017] (its first two to five articles), tokenized by each model's tokenizer and truncated to $n \in \{128, 512\}$ tokens (two passages fall a few tokens short of 512); the passages differ between the two lengths.

Extraction. One forward pass per (model, prompt, length) on a single GPU (one H100; about two hours for the full grid of models, prompts and lengths) returns the $L\times H\times n\times n$ tensor of post-softmax attention weights (fp16 for the Qwen models, fp32 for GPT-2; non-finite rows, which occur in $\leq 0.1\%$ of the rows of Qwen2.5-7B, are zeroed).

Scoring. For every (layer, head, prompt) and every mask of the vocabulary we compute (i) the optimized score, Alternating LAP from a spectral start with 8 iterations and 2 restarts at n=512 (12 and 3 at n=128). On a causal A the symmetric eigensolver of Appendix D.1 reads the lower triangle, so the operator it diagonalizes is $D - W$ with $W = A + A^\top - \operatorname{diag} A$ and $D = \operatorname{diag}(A\mathbf{1})$, the identity for softmax rows; the start is therefore the order by the second-largest eigenvector of the symmetrized attention matrix W, not the Fiedler vector of its graph Laplacian (whose degree matrix would be $\operatorname{diag}(W\mathbf{1})$). The objective itself is not symmetrized. (ii) The identity score, under the native token order, and (iii) the mean score under five random permutations (the null level). The score of a mask is the lower of (i) and (ii); for the arrowhead mask the closed-form optimum of Appendix C.3 is taken as well (it is attained to within 1% by the solver for 99.6–99.8% of the pairs). The label of a pair is the mask with the lowest score; its margin is the relative gap to the runner-up. Scoring one 512-token head takes about 12 core-seconds (Appendix E.8).

Intervals. All intervals are 95% bootstrap intervals over prompts (the 20 prompts resampled with replacement, the statistic recomputed on all heads), i.e. they treat the prompt, not the (head, prompt) pair, as the unit of replication.

E.7.3 Per-Model Results

Table 25 gives the per-model numbers behind Section 5.5 at both lengths; Figure 11 the per-layer label shares and Figure 12 the per-head modal label, its agreement over prompts and its margin.

The depth profile is the same in all six (model, length) cells: block labels in the first three to four layers, arrowhead after them, with Qwen2.5-0.5B returning to block-$k2$ in its last two layers at both lengths. Within a model the label shares move by a few points between n=128 and n=512, and the modal label of a head is the same at both lengths for 91–95% of the heads (0.70–0.77 among the heads whose modal label is not arrowhead at both lengths, so the block-type labels are less stable than the arrowhead label and the layer profile). On the 10-bin relative-depth grid the three models agree in $8/10$ bins at both lengths; pairwise agreement is 0.9 for GPT-2 vs. Qwen2.5-7B and for Qwen2.5-0.5B vs. Qwen2.5-7B and 0.8 for GPT-2 vs. Qwen2.5-0.5B. The last column of Table 25 is the accuracy of a multinomial logistic regression that predicts the label of a pair from four scalar descriptors of the head on that prompt (row entropy, mean maximum weight, sparsity, mean distance from the diagonal), trained on 19 prompts and tested on the held-out one; it is well above the majority label and well below one, so these descriptors partially predict the labels under the tested logistic-regression model. About three quarters of the pairs carry a confident label, a margin over the runner-up above 20% (column "confident"). Figure 13 shows representative heads before and after reordering: the block-labelled heads of the first layers are wide local heads whose mass already lies in the diagonal blocks in the native order (the native order is within 1% of the optimum for 74–84% of the block-labelled pairs at n=512, 57–71% at n=128), and on the Qwen2.5-7B passages whose sink is not the first token the reordering moves the sink key to the front.

Table 25: Per-model attention results (unified scoring). Label shares in % of (head, prompt) pairs: arrowhead with its 95% interval over prompts, block-$k2$, the other block masks and block+band together, bands and diagonal together. Agreement: mean per-head agreement with the head's modal label over the 20 prompts, for all heads, for the heads whose modal label is arrowhead, and for the others. Confident: share of pairs whose margin over the runner-up exceeds 20%. Null: median ratio of the selected mask's score to its score under random permutations. Scalars: leave-one-prompt-out accuracy of predicting the label from four scalar descriptors / majority baseline.

		label share (%)				agreement over prompts					
Model	n	arrowhead [CI]	block-$k2$	other blocks	bands	all	arrowhead	other	confident	null	scalars / maj.
GPT-2	512	69.3 [69.0, 69.7]	14.8	12.9	3.0	0.91	0.99	0.74	0.77	0.07	0.87 / 0.69
	128	72.2 [71.7, 72.6]	12.8	10.8	4.1	0.90	0.98	0.68	0.78	0.08	0.86 / 0.72
Qwen2.5-0.5B	512	67.1 [66.5, 67.6]	15.9	15.4	1.6	0.86	0.96	0.64	0.71	0.12	0.81 / 0.67
	128	69.1 [68.3, 69.8]	16.3	12.9	1.7	0.86	0.96	0.62	0.74	0.12	0.81 / 0.69
Qwen2.5-7B	512	79.9 [79.4, 80.3]	11.4	8.2	0.5	0.92	0.98	0.67	0.83	0.09	0.89 / 0.80
	128	81.3 [80.8, 81.8]	10.2	7.4	1.0	0.91	0.97	0.63	0.85	0.09	0.89 / 0.81

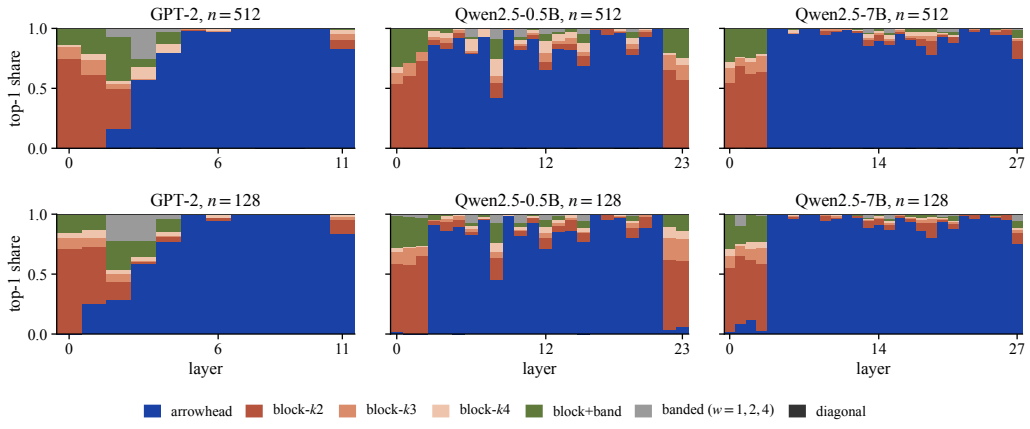

Figure 11: Share of the top-1 label per layer for the three models (columns) at n=512 (top) and n=128 (bottom), 20 prompts.

E.7.4 Identity-Order Control and Long-Context Control

Are the permutations needed? Table 26 scores every (head, prompt) pair at n=512 under three rules: the identity order only, the optimized order only (spectral start, no identity candidate), and the

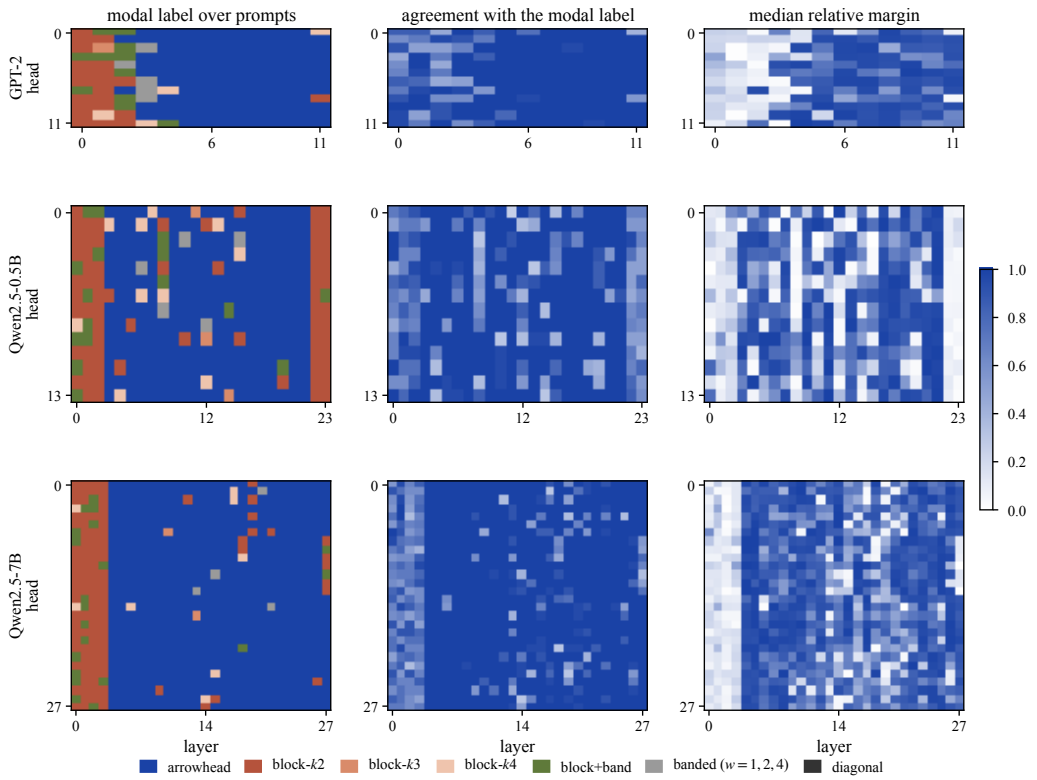

Figure 12: Per-head view at n=512, one row per model: the modal label over the 20 prompts (left), the share of prompts carrying that label (middle) and the median relative margin of the selected mask over the runner-up (right).

unified rule that keeps the identity as a candidate for every mask. The optimized score of a mask can be worse than its identity score, because the solver is started from the spectral start and not from the identity, and on causal attention the native order already is the band; comparing identity labels with optimized-only labels therefore counts label changes caused by competing masks getting worse. Under the unified rule only a few percent of the labels of GPT-2 and Qwen2.5-0.5B change, against about a fifth under the optimized-only rule. On Qwen2.5-7B the change is concentrated on particular passages: on 15 of the 20 passages at n=512 (14 at n=128) the native-order arrowhead share is 0.82 and 3–5% of the labels change, as on the smaller models; on the remaining passages, all of which begin with the token “The”, the sink is not the first token, the native-order arrowhead share is below 0.02, and 83–85% of the labels change. With reordering the arrowhead share is 0.77–0.83 on every passage of every model. This is also why the interval over prompts of the Qwen2.5-7B label-change rate in Table 26 is wide. The last column checks the solver against the closed-form arrowhead optimum (Appendix C.3).

Table 26: Identity-order control at n=512, 20 prompts. Arrowhead share of the (head, prompt) labels under the identity order, under the optimized order only, and under the unified rule (identity kept as a candidate); share of labels that differ from the identity labels under the unified rule (95% interval over prompts) and under the optimized-only rule; share of pairs on which the solver’s arrowhead score is within 1% of the closed-form optimum.

	arrowhead share			labels changed vs identity		
Model	identity	optimized	unified	unified [CI]	optimized	solver = exact
GPT-2	0.70	0.70	0.69	3.5% [3, 4]	19%	99.9%
Qwen2.5-0.5B	0.70	0.68	0.67	6.5% [6, 7]	24%	99.8%
Qwen2.5-7B	0.61	0.80	0.80	23.4% [8, 39]	34%	99.6%

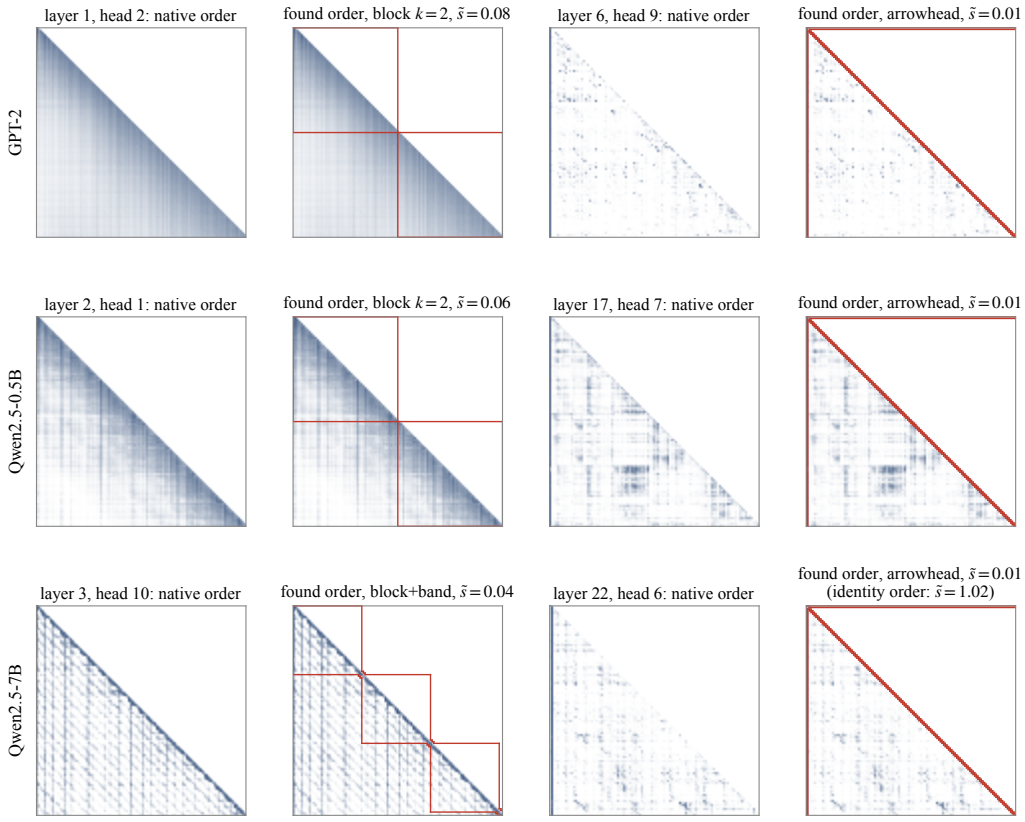

Figure 13: Representative heads of each model at n=128 (one passage per model), logarithmic colour scale: a block-labelled head (left pair) and an arrowhead head (right pair), each in the native token order and in the order found for the selected mask, mask outlined, with the density-normalized score. The block-labelled heads are wide local heads, fitted by the native order; the Qwen2.5-7B arrowhead head has its sink at a key other than the first token (identity score 1.02), which the reordering brings to the front.

Position subsampling does not transfer. A tempting shortcut for long contexts is to score a random subset of 128 of the 512 positions. The label obtained this way agrees with the full-length label for only 0.40 / 0.41 / 0.41 of (head, prompt) pairs (GPT-2 / Qwen2.5-0.5B / Qwen2.5-7B; 5 prompts, two draws each, unified scoring at the sub-length budget), and the per-head modal label over these draws agrees with the full-length one for about half of the heads, against 0.91–0.95 agreement of the per-head modal label between the contiguous n=128 and n=512 contexts. Non-contiguous subsampling destroys the band and block geometry; long contexts should be analysed on contiguous windows or with the scaling routes of Appendix E.8.

E.8 Wall-Clock Scaling of Alternating LAP

Planted banded instances in the Gaussian regime of Section 5.1 (w=5), one Fiedler-initialized run of the LAP phase with the iteration cap of 10 used throughout the paper, 3 seeds at every n; the exchange phase is timed separately below. All times are single-thread wall-clock seconds (BLAS and the assignment solver pinned to one core) on an Apple M3 with 18 GB of memory; peak resident memory is the maximum over the three runs. Two noise series are reported: the per-entry noise fixed at σ_n=0.05 (the regime of Table 1), under which the band holds a $1/n$ fraction of the energy and the error reduction of the planted order itself falls from 0.80 to 0.20; and the noise scaled as $\sigma_n \propto n^{-1/2}$, which keeps the planted order's error reduction at 0.80 for every n. Table 27 gives the measurements.

Table 27: Wall-clock scaling of the LAP phase of one Alternating LAP run (single thread, Apple M3). Times in seconds; "LAPs" is the part of an iteration spent in the two assignment solves; ER is the error reduction of the solver and, in parentheses, of the planted order. FAQ: one run of QAP-FAQ on the same instances and objective. Means over 3 seeds (FAQ at $n \geq 4000$ and the fixed-SNR series at n=8000: one seed).

	time						ER, σ_n=0.05		ER, $\sigma_n \propto n^{-1/2}$	
n	Fiedler	per iter.	LAPs	total	FAQ	memory	AltLAP	FAQ	AltLAP	time
500	0.0	0.04	0.03	0.4	0.7	0.2 GB	0.51 (0.80)	0.57	0.51 (0.80)	1.0
1000	0.1	0.14	0.10	1.5	4.7	0.4 GB	0.44 (0.67)	0.48	0.52 (0.80)	1.6
2000	1.0	0.73	0.49	8.3	32.5	0.6 GB	0.34 (0.50)	0.38	0.55 (0.80)	7.5
4000	10.9	5.4	3.1	65	249	1.3 GB	0.24 (0.34)	0.25	0.59 (0.80)	46
8000	105	37	18	479	1936	3.2 GB	0.15 (0.20)	0.16	0.63 (0.80)	334

Cost. An iteration of the LAP phase is two $O(n^3)$ cost-matrix products and two assignment solves; the solves take three quarters of it at small n and half at n=8000, and the iteration time decreases as the iterates converge (first and last iteration at n=8000: 78 s and 22 s), so the total grows by 4–8$\times$ per doubling of n. The dense Fiedler start (a full eigendecomposition) becomes a fifth of the total at the largest size. Memory is a handful of dense $n \times n$ doubles. FAQ on the same objective costs 2–4$\times$ more and reaches a similar error reduction from one start.

Exchange phase. On the same instances, started from the output of the LAP phase above, the exchange descent of Appendix D.1 takes 0.8–1.1 s at n=500 (20–28 rounds), 6.4–7.6 s at n=1000 (32–39 rounds) and 24–48 s at n=2000 (25–48 rounds), and raises the error reduction by 0.07–0.08, 0.06–0.07 and 0.04–0.05. A round is dominated by the gain evaluation, two $n \times n$ products: 0.4 s at n=2000, 2.7 s at n=4000 and 20 s at n=8000 on this machine, so at the observed 25–50 rounds the phase would take a few minutes at n=4000 and of the order of 10–20 minutes at n=8000, comparable with the LAP phase; it was not run at these two sizes. On the same machine PatLR (rank 3, 12 iterations, 3 restarts) takes 3.0 s at n=200, 22 s at n=500 and 134 s at n=1000; Adaptive PatLR (N=80, ℓ=3, default vocabulary) 2.9 s at n=200 and 31 s at n=500, i.e. 1–1.4$\times$ the cost of one PatLR call. The benchmark panels and the attention scoring run as single-threaded processes on a 24-core AMD EPYC; scoring one 512 $\times$ 512 attention head against the 9-mask vocabulary (8 iterations, 2 restarts per mask, plus the identity and five random permutations) takes about 12 core-seconds there, so the full grid of Section 5.5 (1,264 heads $\times$ 20 prompts at n=512, and again at n=128) is about 85 core-hours (22 processes).

Scaling routes. On one M3 core a solve on a planted banded instance thus takes about two seconds at n=500 and, for the LAP phase alone, minutes at n=8000 (Table 27), and a vocabulary scan of a single 8000-token matrix would take hours per head. The cost matrix is dense (a banded mask only makes it $O(n^2)$ to assemble), so the assignment solve is the bottleneck; sparse assignment with restricted candidates, coarsen-then-refine, and GPU Sinkhorn or auction solvers [Cuturi, 2013, Bertsekas, 1988] are left for future work.

Quality against n. The iteration cap is not binding: with the cap raised to 30 the solver stops by its own convergence test after 9–12 iterations at $n \leq 2000$ and the error reduction changes by less than 0.001. Under fixed per-entry noise the error reduction falls with n because the attainable reduction falls: the solver reaches a stable two thirds to three quarters of the planted order's value at every n. With the matrix SNR held fixed the error reduction does not degrade with size; it rises slowly with n. Under fixed per-entry noise the Fiedler start alone reduces the error by 0.1 at n=500 and by less than 0.04 at $n \geq 1000$ (the Laplacian of $|A|$ is dominated by the dense noise), so the reduction is produced by the LAP iterations.

NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: Claims match Sections 2–5 and the appendices.
   Guidelines:
   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.
2. **Limitations**
   Question: Does the paper discuss the limitations of the work performed by the authors?
   Answer: [Yes]
   Justification: Section 6: cost, local optima, vocabulary scope, proxy objective, theory scope.
   Guidelines:
   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.
3. **Theory assumptions and proofs**
   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?
   Answer: [Yes]
   Justification: Full proofs in Appendices B–C; assumptions stated with each result.
   Guidelines:
   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.

- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**
   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?
   Answer: [Yes]
   Justification: Setups, hyperparameters and seed counts per experiment (typically 20, exceptions stated) in Section 5 and Appendix E.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**
   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?
   Answer: [No]
   Justification: No code release; Algorithms 1–3 fully specified; all data public.
   Guidelines:
   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.

- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**
   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?
   Answer: [Yes]
   Justification: Section 5 and Appendix E.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.
7. **Experiment statistical significance**
   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?
   Answer: [Yes]
   Justification: 95% bootstrap intervals: over seeds (typically 20) for the benchmarks, over the 20 prompts for attention.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.
8. **Experiments compute resources**
   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?
   Answer: [Yes]
   Justification: Attention extraction: one H100, about 2 hours. Benchmark panels and attention scoring: AMD EPYC, single-threaded processes, about 85 core-hours. Scaling study and PatLR timings: Apple M3 (Appendix E.8).
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Conforms.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Foundational matrix analysis; no direct societal impact.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: No such assets released.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?
Answer: [Yes]
Justification: All assets public and cited; UCI CC BY 4.0, GPT-2 MIT, Qwen2.5 Apache 2.0, WikiText-2 CC BY-SA 3.0.
Guidelines:
- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**
Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?
Answer: [N/A]
Justification: None released.
Guidelines:
- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**
Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?
Answer: [N/A]
Justification: None.
Guidelines:
- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**
Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?
Answer: [N/A]
Justification: Not applicable.
Guidelines:
- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are study subjects only.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.