Algorithms as Geodesics: Flattening Entropy and Partial Curvature Removal in Sorting

Jonathan R. Landers
Independent researcher
ORCID: 0000-0003-1872-6179

Abstract

A computational model induces a geometry on state space: its admissible primitive operations determine which motions are available and what they cost, and runtime is the length of a path. In this picture, a classical complexity lower bound acts as a curvature obstruction — a statement that no globally flattening coordinate chart exists inside the model. We develop this viewpoint for sorting. Counting sort is shown to be a literal flattening theorem: in histogram coordinates the relevant state space is globally flat, its Levi–Civita connection vanishes, its Riemann tensor vanishes identically, and the essential computation is geodesic transport to the endpoint histogram. Comparison sorting is shown to saturate a matching quantitative obstruction, which we call the *flattening entropy*: for n distinct keys, $\mathfrak{F}_{\mathrm{cmp}}(n) = \log(n!) + O(1)$. The main contribution is a theorem that bridges these two extremes. We prove that every ordered bucketization removes an explicit amount of sorting curvature,

$$\mathcal{G}(x;\beta) \;=\; \log \frac{n!}{\prod_{j=1}^{B} n_j!},$$

leaving a residual comparison problem whose flattening entropy is $\sum_{j=1}^{B} \log(n_j!)$. Radix sorting is recovered as iteration of this principle, with each pass shedding $n \log B$ nats of residual entropy in the balanced regime. Together, these results turn the geometric narrative into a usable quantitative tool: a way to measure, in units of entropy, how much a preprocessing step trivializes the sorting geometry before local comparisons are required.

1 Computation as motion in an admissible geometry

A sorting algorithm moves through configurations until it reaches the sorted state, and its runtime is the cost of that motion. The viewpoint of this note is that the admissible primitive operations of a computational model define a metric on the space of configurations, so that algorithmic complexity is a path length and an optimal algorithm is a geodesic. In a smooth surrogate, one writes a line element

$$ds^2 \;=\; g^{(\mathcal{A})}_{\mu\nu}(x)\, dx^\mu\, dx^\nu,$$

or more generally a Finsler cost functional, where the subscript $\mathcal{A}$ records which primitive operations the model admits. This stance is deliberately close in spirit to geometric formulations of information and complexity in information geometry and geodesic approaches to computation (Amari and Nagaoka, 2000; Ay et al., 2017; Dowling et al., 2006; Nielsen, 2020). Classical lower bounds then admit a geometric reading: no algorithm inside $\mathcal{A}$ can be faster than the diameter of the admissible geometry, and that diameter is bounded below by the failure of any coordinate chart internal to $\mathcal{A}$ to globally flatten the space. This is the computational analogue of curvature in general relativity: the impossibility of trivializing the geometry by a single change of coordinates.

The interest of this picture is not that it replaces classical complexity theory but that it suggests a quantitative invariant distinct from operation counting. If global flattening is an achievable computational primitive in some models and not in others, then the *amount* of coordinate information required to flatten should be measurable, and should separate the two regimes.

We make that measurement for sorting. Three results follow in order of increasing novelty. First, counting sort is exhibited as a literal flattening theorem in an honest Riemannian sense. Second, comparison sorting is shown to saturate a matching entropy obstruction, the flattening entropy, at $\log(n!) + O(1)$, in direct dialogue with the classical decision-tree and information-theoretic lower-bound literature (Knuth, 1998; Fredman, 1976). Third — and this is the genuinely new piece of the framework — every ordered bucketization is shown to remove an explicit quantity of sorting curvature, leaving a residual comparison problem whose size is a precise entropy. Radix sorting is recovered as the iterate of this principle. The framework thereby interpolates continuously between pure comparison sorting and full flattening, measuring hybrid algorithms by how much geometric trivialization they buy before resorting to local comparisons.

2 Related work and context

Information geometry is the nearest mathematical neighbor: metrics, affine connections, duality, and flat coordinates are extracted there from informational structure (Amari and Nagaoka, 2000; Ay et al., 2017; Nielsen, 2020). Recent GSI work continues that structural line through contrast bi-forms encoding metric and connection data (Ciaglia et al., 2025) and Jensen–Shannon divergences on Gaussian measures (Minh and Nielsen, 2025). A second thread treats geodesic distance itself as a complexity surrogate or structural invariant, from quantum circuit geometry (Dowling et al., 2006) and discrete curvature on metric spaces (Ollivier, 2009) to recent GSI work on reductive homogeneous spaces (Duits et al., 2025). Classical sorting theory supplies the combinatorial side: decision-tree lower bounds (Knuth, 1998; Fredman, 1976), entropy for multiplicities and partial information (Munro and Spira, 1976; Kahn and Kim, 1995; Haeupler et al., 2025), and adaptive sorting via monotone runs (Munro and Wild, 2018). The present contribution is to reinterpret counting, bucketing, and radix passes as explicit flattenings that remove measurable residual sorting entropy.

Figure 1 previews the theorem-level entropy decomposition developed below, while Figures 2 and 3 quantify its occupancy-simplex interpolation and its radix iteration.

3 Counting sort is a flattening

Counting sort and its radix relatives exploit bounded-key structure to bypass the comparison barrier familiar from the decision-tree model (see Knuth, 1998). The flat-chart interpretation below formalizes that advantage geometrically, now in an elementary histogram model rather than a continuous geodesic space (compare Duits et al., 2025).

Assume keys lie in a bounded ordered range $[k] = \{1, \dots, k\}$. The input space is $X_{n,k} = [k]^n$, and the histogram map

$$\Phi : X_{n,k} \to \mathbb{N}^k, \qquad \Phi(x)_r = \#\{\, i : x_i = r \,\},$$

records only the multiset of keys. Because sorting depends on nothing else, it factors through Φ,

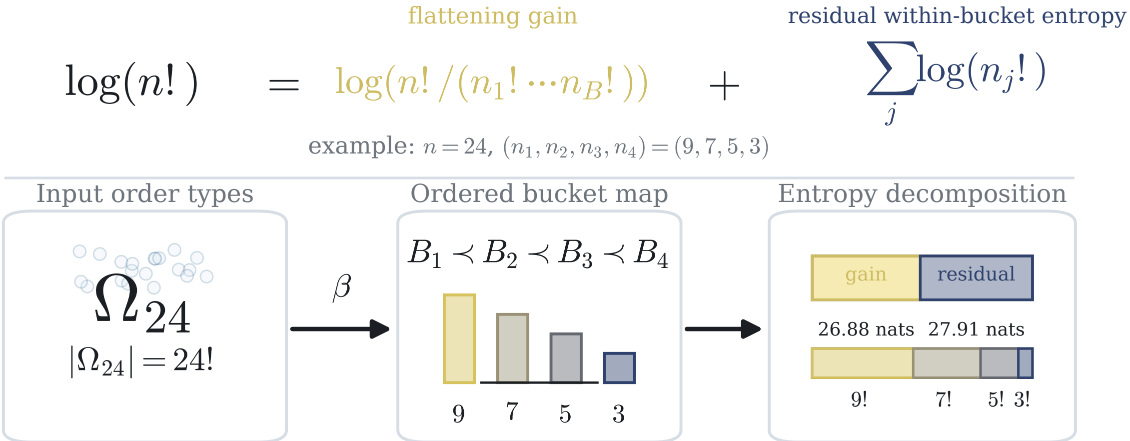

Figure 1: Ordered bucketization removes a multinomial block of sorting entropy exactly. In the example $n = 24$ with occupancies $(n_1, n_2, n_3, n_4) = (9, 7, 5, 3)$, the ordered bucket map resolves the cross-bucket order outright, contributing the flattening gain $\log\big(24!/(9!\,7!\,5!\,3!)\big)$, while the residual within-bucket comparison entropy is $\log(9!\,7!\,5!\,3!) = \sum_j \log(n_j!)$. The bar lengths are proportional to the exact entropies in nats.

and the relevant state variable is the endpoint histogram

$$c \;=\; \Phi(x) \;\in\; M_{n,k} \;:=\; \Big\{\, c \in \mathbb{R}^k_{\geq 0} : \sum_{r=1}^{k} c_r = n \,\Big\}.$$

Proposition 3.1 (Counting sort as flattening)**.** *Equip $M_{n,k}$ with the metric induced from $\mathbb{R}^k$. Because $M_{n,k}$ is an affine simplex in the hyperplane $\sum_{r=1}^k c_r = n$, it is globally flat: in affine coordinates the metric coefficients are constant, the Levi-Civita connection and Riemann tensor vanish, and for each input $x \in [k]^n$ the essential counting-sort trajectory is the straight geodesic $c(t) = t\,\Phi(x)$ from the accumulation origin to $\Phi(x)$.*

Proof. Any affine coordinates on the hyperplane inherit a constant metric from $\mathbb{R}^k$, so curvature vanishes and geodesics are straight segments. □

The exact unit-cost geometry is additive rather than quadratic: replacing the Euclidean metric by the flat translation-invariant Finsler norm $F(c, \xi) = \|\xi\|_1$ gives length one to each primitive increment $c \mapsto c + e_r$ and total length exactly n to the counting trajectory. Either way, bounded-key sorting factors through a flat space, echoing the flat-coordinate and geometric-potential themes of information geometry (Amari and Nagaoka, 2000; Ay et al., 2017; Nielsen, 2020; Ciaglia et al., 2025).

4 Flattening entropy

Proposition 3.1 explains counting sort by the existence of a global flat chart. To measure how far other models lie from flatness, introduce an invariant recording the minimum coordinate information needed for global flattening. This repackages the classical information-theoretic view of comparison sorting (Fredman, 1976; Knuth, 1998) as a geometric obstruction, closer in spirit to recent GSI work that makes geometry legible through explicit potentials and divergences (Ciaglia et al., 2025; Minh and Nielsen, 2025).

Let Ω_n denote the set of order types of n distinct keys, so $|\Omega_n| = n!$.

Definition 4.1. A *flattening chart* for a sorting model on Ω_n is a pair (Z_n, Ψ_n) in which Z_n is flat, $\Psi_n : \Omega_n \to Z_n$ is injective, the sorted output is recoverable from $\Psi_n(\omega)$, and the remaining computation is geodesic motion in Z_n once $\Psi_n(\omega)$ is known.

Definition 4.2. The *flattening entropy* is $\mathfrak{F}(n) := \inf_{(Z_n,\Psi_n)} \log |Z_n|$, where the infimum runs over all flattening charts. Logarithms are natural.

Proposition 4.3 (Comparison flattening entropy)**.** *For comparison sorting on n distinct keys,*

$$\mathfrak{F}_{\text{cmp}}(n) \;=\; \log(n!) + O(1).$$

Proof. Any flattening chart must distinguish all $n!$ order types, so $|Z_n| \geq n!$ and $\mathfrak{F}_{\text{cmp}}(n) \geq \log(n!)$. For the reverse inequality, take Z_n to be the set of permutations itself, or any binary encoding of size $2^{\lceil \log_2(n!) \rceil}$, and let Ψ_n send each input to its permutation type. The sorted output is then determined. □

This is the usual information-theoretic lower bound in geometric language: an optimal flattening chart has the cardinality of the sorting decision tree's leaf set, so the minimum depth is $\lceil \log_2 n! \rceil$ (Knuth, 1998; Fredman, 1976). Complexity gaps between models become coordinate-entropy gaps. The following corollary makes this explicit.

Corollary 4.4 (Runtime lower bound from flattening entropy)**.** *If each primitive operation of a model contributes at most b nats of flattening information, then any algorithm in the model must take at least $\mathfrak{F}(n)/b$ steps in the worst case. For binary comparisons, $b = \log 2$, and*

$$T_{\text{cmp}}(n) \;\geq\; \log_2(n!) \;=\; n \log_2 n - (\log_2 e)\, n + O(\log n).$$

Mergesort matches this up to constants, so the comparison model sits at entropy distance $\Theta(n \log n)$ from flatness.

5 Partial flattening by ordered bucketing

The comparison model carries full permutation entropy $\log(n!)$; the bounded-key model is flat. The intermediate question is whether one can flatten part of the geometry and then fall back to comparisons on the remainder. Ordered bucketization does exactly this. The residual-entropy formula below is closest in spirit to results on multiplicities and partial information (Munro and Spira, 1976; Kahn and Kim, 1995; Haeupler et al., 2025), though the structure here is an ordered bucket map rather than multiplicity counts or a general partial order.

Let $\beta : \mathcal{K} \to \{1, \ldots, B\}$ be an *ordered bucket map* on the key space, meaning

$$\beta(x) < \beta(y) \quad \Longrightarrow \quad x < y.$$

Bucket labels thus carry correct global order information, but nothing about internal order within a bucket. For an input $x_1, \ldots, x_n$, let

$$n_j \;:=\; \#\{\, i : \beta(x_i) = j \,\}, \qquad j = 1, \ldots, B,$$

denote the bucket occupancies.

Theorem 5.1 (Partial flattening by ordered bucketing)**.** *Let β be an ordered bucket map with bucket occupancies $n_1, \ldots, n_B$. Then:*

1. *the residual sorting entropy after bucketization is*

$$\mathfrak{F}_{\mathrm{res}}(x; \beta) = \sum_{j=1}^{B} \log(n_j!);$$

2. *the flattening gain achieved by bucketization is*

$$\mathcal{G}(x; \beta) = \log(n!) - \sum_{j=1}^{B} \log(n_j!) = \log \frac{n!}{\prod_{j=1}^{B} n_j!};$$

3. *any comparison-based refinement after bucketization requires at least*

$$\sum_{j=1}^{B} \log_2(n_j!)$$

comparisons in the worst case, and mergesort within buckets achieves

$$T(x; \beta) = O(n + B) + \sum_{j=1}^{B} O(n_j \log n_j).$$

Proof. The ordering condition makes cross-bucket order deterministic, leaving $\prod_j n_j!$ internal order types; taking logs gives the residual entropy, and subtracting from $\log(n!)$ gives the gain. The comparison lower bound follows by Proposition 4.3 applied independently in each bucket, and the upper bound by linear-time bucketing followed by in-bucket mergesort. □

Equivalently,

$$\log(n!) = \log \frac{n!}{\prod_j n_j!} + \sum_j \log(n_j!),$$

exactly decomposing total sorting entropy into a *flattened* part and a *residual* part. The multinomial term is what bucketization resolves; the remainder is what comparison must still distinguish. Figure 1 visualizes this split in a single ordered-bucketization example.

6 Interpolation between comparison and flattening

At $B = 1$ there is a single bucket containing all n elements, so $\mathcal{G} = 0$ and no flattening has occurred. At the opposite extreme, if every bucket contains at most one element, then $\log(n_j!) = 0$ for all j and the residual entropy vanishes, recovering counting sort. Between these endpoints, a balanced bucketization with $n_j \approx n/B$ gives, by Stirling,

$$\sum_{j=1}^{B} \log(n_j!) \approx B \cdot \frac{n}{B} \log \frac{n}{B} = n \log \frac{n}{B},$$

up to linear terms, so the hybrid runtime is

$$T(x; \beta) \approx n + n \log \frac{n}{B}.$$

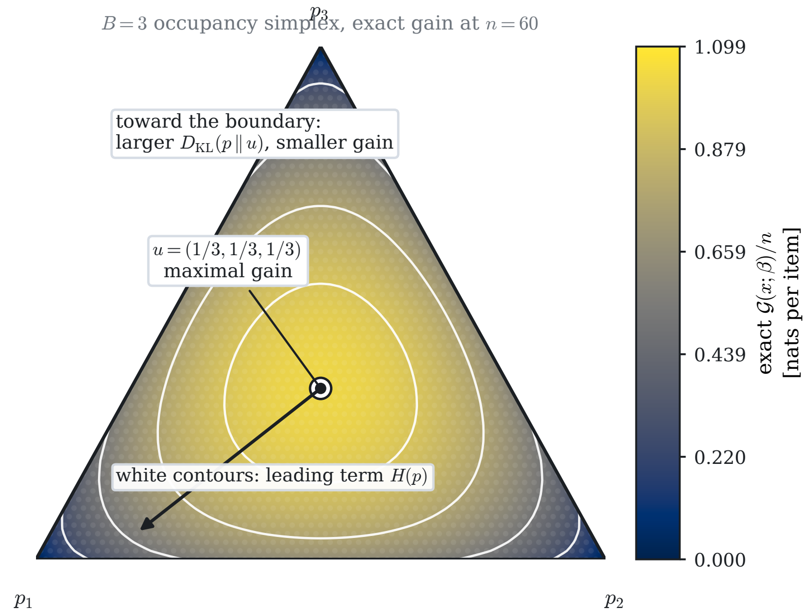

Figure 2: Exact finite-n flattening gain on the bucket-occupancy simplex for $B = 3$ and $n = 60$. Each lattice point corresponds to occupancies (n_1, n_2, n_3) with $n_1+n_2+n_3 = n$, colored by the exact normalized gain $\mathcal{G}(x;\beta)/n$ in nats per item; the white contours show the leading term $H(p)$. The balanced distribution $u = (1/3, 1/3, 1/3)$ maximizes gain, and motion toward the boundary increases the KL shortfall $D_{\mathrm{KL}}(p\,\|\,u)$ from the balanced optimum.

Parameterizing $B = n^\alpha$ for $\alpha \in [0, 1]$,

$$T(x;\beta) \;\approx\; n(1-\alpha)\log n + n.$$

Thus $\alpha = 0$ gives comparison sorting, $\alpha = 1$ gives counting sort, and intermediate α interpolates continuously between them; the exponent is a *partial flattening degree*. Writing $p_j := n_j/n$, the occupancy vector $p = (p_1, \ldots, p_B)$ lies on the probability simplex Δ^{B-1}, the canonical dually-flat manifold of information geometry (Amari and Nagaoka, 2000; Nielsen, 2020; Ciaglia et al., 2025). Stirling's formula gives

$$\mathcal{G}(x;\beta) \;=\; nH(p) + O(\log n) \;=\; n\big(\log B - D_{\mathrm{KL}}(p\,\|\,u)\big) + O(\log n),$$

where $H(p) = -\sum_j p_j \log p_j$ and $u = (1/B, \ldots, 1/B)$. So balanced buckets maximize gain, and the shortfall from the balanced optimum is, to leading order, n times the KL divergence from uniform. This is complementary to adaptive sorting (Munro and Wild, 2018): there the exploitable structure is existing local order, whereas here it is order-preserving coarse information supplied before residual comparisons begin.

Figure 2 evaluates the exact finite-n gain on Δ^2 in the case $B = 3$: each lattice point corresponds to occupancies $n_1 + n_2 + n_3 = n$, the balanced point u maximizes gain, and the decay toward the boundary follows the KL shortfall from uniform that appears in the leading-order formula.

7 Radix sorting as iterated partial flattening

Radix procedures are the repeated-bucketing case. If keys are strings over an alphabet of size B, then after examining the first t symbols the input is partitioned into ordered prefix buckets

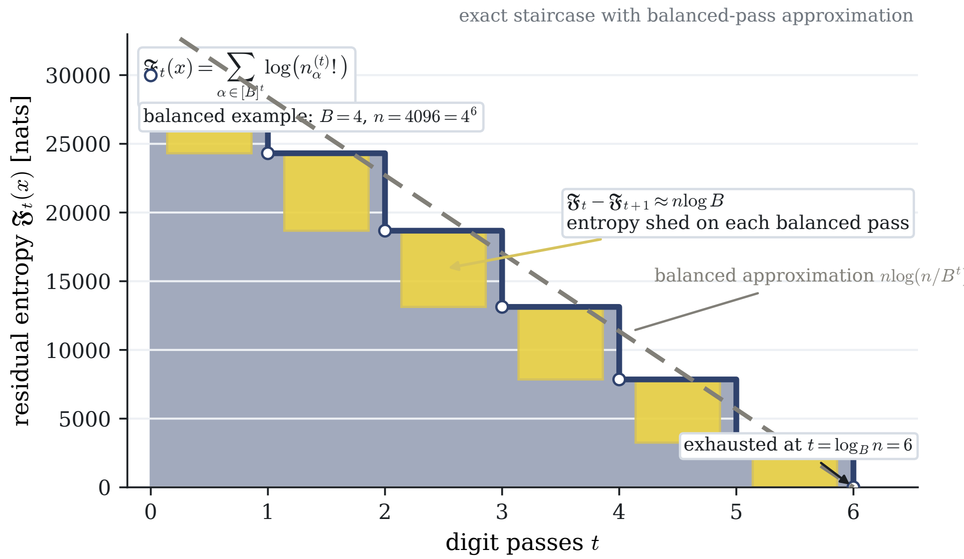

Figure 3: Residual entropy across balanced radix passes for $B = 4$ and $n = 4096$. The exact residual after pass t is $\mathfrak{F}_t(x) = \sum_{\alpha\in[4]^t} \log(n_\alpha^{(t)}!)$; in the balanced regime this is $4^t \log((4096/4^t)!)$. Each decrement $\mathfrak{F}_t(x) - \mathfrak{F}_{t+1}(x)$ is close to $n \log 4$, so each pass sheds one bucketization layer of entropy until the residual is exhausted at $t = \log_4 n = 6$.

$\alpha \in [B]^t$ with occupancies $n_\alpha^{(t)}$, so Theorem 5.1 gives immediately

$$\mathfrak{F}_t(x) \;=\; \sum_{\alpha\in[B]^t} \log\big(n_\alpha^{(t)}!\big).$$

In the balanced regime $n_\alpha^{(t)} \approx n/B^t$, this becomes $\mathfrak{F}_t(x) \approx n \log \frac{n}{B^t}$, so $\mathfrak{F}_t(x) - \mathfrak{F}_{t+1}(x) \approx n \log B = nH(u)$, where u is uniform on B symbols. Each pass therefore removes one uniform-simplex worth of entropy, and the usual stopping criterion $t \sim \log_B n$ can be read as the number of entropy-shedding steps needed to exhaust the flattening budget, as quantified in Figure 3.

Physical interpretation. The invariant $\mathfrak{F}(n)$ is purely informational, though the exact decomposition $\log(n!) = \log \frac{n!}{\prod_j n_j!} + \sum_j \log(n_j!)$ invites a Landauer-style reading (Landauer, 1961), with free-energy cost of order $k_B T$ times the flattened entropy. That interpretation is optional: the theorems above are entropy statements about sorting models and do not rely on thermodynamic assumptions.

8 Closing perspective

Three statements emerge. Counting sort is a literal flattening: the histogram simplex is globally flat, and the algorithm is straight-line motion to the endpoint histogram, placing an elementary sorting procedure in the same geometric register in which flat coordinates, potentials, and geodesic distance are central objects (Amari and Nagaoka, 2000; Ay et al., 2017; Ciaglia et al., 2025; Duits et al., 2025). Comparison sorting saturates a matching quantitative obstruction: the flattening entropy $\mathfrak{F}_{\mathrm{cmp}}(n) = \log(n!) + O(1)$ is the minimum coordinate information required to trivialize the sorting geometry, and the classical $n \log_2 n$ lower bound is its runtime translation (Knuth, 1998; Fredman, 1976). Ordered bucketization then removes an explicit amount of that obstruction:

the flattening gain $\log(n!/\prod_j n_j!)$ and residual $\sum_j \log(n_j!)$ are exact entropies, and radix sorting iterates this decomposition in direct contact with entropy-based views of multiplicities and partial information (Munro and Spira, 1976; Kahn and Kim, 1995; Haeupler et al., 2025). The framework therefore places algorithms on a continuous scale of geometric trivialization and suggests a meeting point between classical sorting theory and the geometric program in which informational structure is expressed through coordinates, potentials, divergences, and geodesic distance (Amari and Nagaoka, 2000; Nielsen, 2020; Ciaglia et al., 2025; Minh and Nielsen, 2025; Duits et al., 2025). Whether the same lens extends usefully to selection, merging, sorting under partial orders, and related structured tasks where a quotient by a symmetry group admits a natural flattening chart is a question the framework makes it possible to ask.

References

Shun-ichi Amari and Hiroshi Nagaoka. *Methods of Information Geometry*. Translations of Mathematical Monographs, vol. 191, American Mathematical Society and Oxford University Press, 2000. doi:10.1090/mmono/191.

Nihat Ay, Jürgen Jost, Hông Vân Lê, and Lorenz Schwachhöfer. *Information Geometry*. Ergebnisse der Mathematik und ihrer Grenzgebiete. 3. Folge / A Series of Modern Surveys in Mathematics, vol. 64, Springer, 2017. doi:10.1007/978-3-319-56478-4.

Florio M. Ciaglia, Giuseppe Marmo, Marco Pacelli, Luca Schiavone, and Alessandro Zampini. Bi-forms approach to potential functions in information geometry. In *Geometric Science of Information: 7th International Conference, GSI 2025, Saint-Malo, France, October 29–31, 2025, Proceedings, Part I*, volume 16033 of *Lecture Notes in Computer Science*, pages 73–82. Springer, 2025. doi:10.1007/978-3-032-03918-7_8.

Mark R. Dowling, Michael A. Nielsen, Mile Gu, and Andrew C. Doherty. Quantum computation as geometry. *Science*, 311(5764):1133–1135, 2006. doi:10.1126/science.1121541.

Remco Duits, G. Bellaard, and A. B. Tumpach. Analysis and computation of geodesic distances on reductive homogeneous spaces. In *Geometric Science of Information: 7th International Conference, GSI 2025, Saint-Malo, France, October 29–31, 2025, Proceedings, Part I*, volume 16033 of *Lecture Notes in Computer Science*, pages 13–23. Springer, 2025. doi:10.1007/978-3-032-03918-7_2.

Michael L. Fredman. How good is the information theory bound in sorting? *Theoretical Computer Science*, 1(4):355–361, 1976. doi:10.1016/0304-3975(76)90078-5.

Bernhard Haeupler, Richard Hladík, John Iacono, Václav Rozhoň, Robert E. Tarjan, and Jakub Tětek. Fast and simple sorting using partial information. In *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3953–3973, 2025. doi:10.1137/1.9781611978322.134.

Jeff Kahn and Jeong Han Kim. Entropy and sorting. *Journal of Computer and System Sciences*, 51(3):390–399, 1995. doi:10.1006/jcss.1995.1077.

Donald E. Knuth. *The Art of Computer Programming, Volume 3: Sorting and Searching*. 2nd edition, Addison-Wesley, 1998.

Rolf Landauer. Irreversibility and heat generation in the computing process. *IBM Journal of Research and Development*, 5(3):183–191, 1961. doi:10.1147/rd.53.0183.

Hà Quang Minh and Frank Nielsen. Geometric Jensen–Shannon divergence between Gaussian measures on Hilbert space. In *Geometric Science of Information: 7th International Conference, GSI 2025, Saint-Malo, France, October 29–31, 2025, Proceedings, Part II*, volume 16034 of *Lecture Notes in Computer Science*, pages 69–79. Springer, 2025. doi:10.1007/978-3-032-03921-7_8.

J. Ian Munro and Philip M. Spira. Sorting and searching in multisets. *SIAM Journal on Computing*, 5(1):1–8, 1976. doi:10.1137/0205001.

J. Ian Munro and Sebastian Wild. Nearly-optimal mergesorts: Fast, practical sorting methods that optimally adapt to existing runs. In *26th Annual European Symposium on Algorithms (ESA 2018)*, volume 112 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 63:1–63:16, 2018. doi:10.4230/LIPIcs.ESA.2018.63.

Frank Nielsen. An elementary introduction to information geometry. *Entropy*, 22(10):1100, 2020. doi:10.3390/e22101100.

Yann Ollivier. Ricci curvature of Markov chains on metric spaces. *Journal of Functional Analysis*, 256(3):810–864, 2009. doi:10.1016/j.jfa.2008.11.001.