Supplementary Material for:

EXaCTz: Guaranteed Extremum Graph and Contour Tree Preservation for Distributed- and GPU-Parallel Lossy Compression

This appendix provides supplementary material to support the main findings of this paper. Section A details the evaluation datasets. Section B presents additional performance evaluation using TTHRESH and SPERR as base compressors. Section C provides empirical analyses on parameter N. Section D analyzes the topological distortion introduced by different base compressors. Section E compares the overall compression ratio with TopoA. Section F reports the precision of topological features. Section G analyzes error distributions before and after correction. Section H compares edit maps with and without integral path computation. Section I conducts weak scaling using downscaled real datasets.

A DETAILS OF THE DATASETS

This section outlines the datasets used in our evaluation.

The **QMCPACK** dataset is from performance tests of a continuum quantum Monte Carlo simulation [5] (accessed via the SDRBench [9], https://sdrbench.github.io/).

The **Adenine Thymine** is a molecular simulation dataset from the Topology ToolKit (TTK) Tutorial Data [8] (https://topology-tool-kit.github.io/tutorials.html), which models the electron density of a DNA base pair. It restricts the scalar field to a 2D planar slice within a 3D domain, reflecting common practices in quantum chemistry to reduce complexity while retaining essential structural information.

The **Turbulent Vortex** dataset is a standard benchmark widely used in topological data analysis and originates from a fluid turbulence simulation and features intricate, interlocking vortex structures (accessed via the SDRBench [9], https://sdrbench.github.io/).

The **NYX** dataset [3] consists of 3D spatial fields generated by the cosmological hydrodynamics code NYX [1]. Each output is a three-dimensional spatial field used for post-analysis and includes variables such as dark matter density and temperature. For our evaluation, we use this dataset at three different dimensions: 512^3, 1024^3, and 2048^3 (accessed via the SDRBench [9], https://sdrbench.github.io/).

The **Turbulent Combustion Simulation** dataset captures a turbulent non-premixed flame with detailed chemistry, generated via high-fidelity direct numerical simulation (DNS) [4]. For our evaluation, we evaluate this dataset in three dimensions: 256^3, 560^3, and 4096^3.

B ADDITIONAL PERFORMANCE EVALUATION ON TTHRESH AND SPERR

We extend our performance evaluation by using TTHRESH [2] and SPERR [6] as base compressors across single- and multi-GPU environments, as shown in Table 1. In the multi-GPU setting, the base compressors operate independently on each local data block. Note that TTHRESH does not support strict pointwise error bounding; we tune its parameters so that the resulting maximum pointwise error is close to the target bound of 10^{-4}.

Across all datasets, EXaCTz consistently completes correction within the same order of magnitude as the base compression time, and often substantially faster, and the edit ratio remains consistently low (typically below 8%). For example, on the 2048^3 NYX dataset, EXaCTz requires 18.43s for correction compared to 437s for the initial SPERR compression, while modifying only 2.86% of the data. For smaller datasets, the correction time is negligible (e.g., 0.03–0.27s for QMCPack and AT), while for larger datasets, the correction cost scales moderately with data size.

Comparing compressors, correcting TTHRESH generally requires more iterations and longer runtime than SPERR. This is because TTHRESH's global tensor decomposition introduces more spatially distributed perturbations, leading to widespread violations that require additional correction iterations.

Table 1: Performance evaluation of EXaCTz on NVIDIA A100 GPUs using SPERR and TTHRESH as base compressors (relative error bound $\xi = 10^{-4}$). SPERR and TTHRESH are CPU-based compressors, and the reported EXaCTz time refers only to the correction stage. #GPUs denotes the minimum number of GPUs required to fit each dataset.

Dataset	Size (GB)	#GPUs	Stage 1 Compression-only			Stage 2 EXaCTz Correction-only			
			Type	Time (s)	CR	Time (s)	Edit %	Iter.	OCR
QMCPack	0.004	1	SPERR	0.30	89.16	0.05	0.84	51	51.06
			TTHRESH	0.22	88.04	0.03	0.77	34	71.11
AT	0.006	1	SPERR	0.54	73.78	0.01	0.27	13	69.21
			TTHRESH	0.27	153.61	0.04	0.01	56	65.47
Vortex	0.015	1	SPERR	1.12	20.41	1.30	1.30	52	17.84
			TTHRESH	0.94	7.36	0.06	0.99	133	7.02
Turbulence	0.125	1	SPERR	10.21	9.22	0.18	4.87	36	6.99
			TTHRESH	6.53	4.85	0.28	4.22	159	4.19
NYX (512^3)	1.000	1	SPERR	11.34	11.92	0.53	2.75	56	10.20
			TTHRESH	47.02	6.98	0.71	1.63	117	6.41
Combustion	1.310	1	SPERR	71.19	145.11	0.45	0.87	79	104.63
			TTHRESH	46.87	64.19	1.36	0.24	287	60.37
NYX (1024^3)	8.00	2	SPERR	441	12.01	5.46	2.65	152	10.42
			TTHRESH	137.64	8.11	8.74	2.89	453	7.34
NYX (2048^3)	64.00	16	SPERR	437	12.45	18.43	2.86	185	10.81
			TTHRESH	137.35	12.09	32.36	7.71	372	9.79
Turbulence (4096^3)	512.00	128	SPERR	447	48.44	15.66	1.76	447	34.74
			TTHRESH	133.26	26.58	46.72	1.69	568	16.44

C EMPIRICAL EVALUATION ON PARAMETER N

We evaluate the impact of the parameter N, which limits the maximum number of edits allowed per vertex, on both computational overhead and overall compression ratio (OCR), as shown in Figure 1.

Across all datasets and compressors, N introduces a trade-off between correction cost and overall compression ratio. When N is small, the algorithm restricts the number of edits per vertex, resulting in lower computational overhead but reduced CR. Conversely, increasing N allows for a higher number of finer edits to be applied to each vertex within each iteration. While this provides improved OCR, it also leads to a higher overall correction time.

A consistent trend emerges despite variations across datasets and compressors: the improvement in CR gradually saturates as N increases. In most cases, OCR shows only marginal gains when $N > 5$, whereas the computational overhead continues to increase. While the optimal value of N may vary slightly across datasets and compressors, $N = 5$ performs well across all evaluated cases. Based on this observation, we set $N = 5$ in our experiments as a practical balance between correction cost and compression ratio.

D TOPOLOGICAL DISTORTION BY BASE COMPRESSORS

To evaluate how different base compressors affect extremum graphs and contour trees, we tested them under the same relative error bound of 10^{-4}, as shown in Table 2. The evaluated compressors include ZFP (transform-based), TTHRESH (tensor-based), SPERR (block-based), MGARD, and cuSZp (cuZFP is excluded because it does not support fixed error bounds). In contrast, after the iterative correction by EXaCTz, both EG and CT are perfectly preserved across all datasets and base compressors, achieving a recall of 1.00.

Across all compressors, CP recall remains relatively high compared to EG and CT, indicating that critical point detection is more robust under error-bounded compression. In contrast, EG and CT exhibit significantly larger distortion, especially on complex datasets such as Combustion and NYX. For example, on the Combustion dataset, CT recall drops to as low as 0.25 for cuSZp and 0.29 for MGARD. Similar trends are observed on NYX, where EG and CT recalls decrease sharply for most compressors.

Different compressors also exhibit different distortion patterns. ZFP and MGARD tend to better preserve EG structures in some cases

Table 2: Topology preservation comparison under relative error bound 10^{-4}. ZFP, TTHRESH, SPERR, MGARD, and cuSZp represent the baselines before correction. All metrics are reported in terms of recall.

Dataset	Before Correction															After Correction		
	ZFP			TTHRESH			SPERR			cuSZp			MGARD			EXaCTz		
	CP	EG	CT	CP	EG	CT	CP	EG	CT	CP	EG	CT	CP	EG	CT	CP	EG	CT
QMCPack	0.93	0.84	0.84	0.91	0.86	0.76	0.80	0.72	0.64	0.27	0.48	0.40	0.91	0.80	0.82	**1.00**	**1.00**	**1.00**
AT	0.98	0.94	0.91	0.75	0.68	0.61	0.89	0.88	0.84	0.68	0.64	0.50	0.90	0.94	0.92	**1.00**	**1.00**	**1.00**
Vortex	0.99	0.99	0.95	0.99	0.88	0.93	0.98	0.94	0.82	0.95	0.89	0.74	0.99	0.97	0.89	**1.00**	**1.00**	**1.00**
Turbulence	0.91	0.91	0.83	0.99	0.99	0.83	0.98	0.97	0.77	0.97	0.95	0.73	0.99	0.99	0.82	**1.00**	**1.00**	**1.00**
NYX	0.99	0.98	0.77	0.97	0.96	0.72	0.90	0.82	0.57	0.80	0.69	0.44	0.85	0.75	0.51	**1.00**	**1.00**	**1.00**
Combustion	0.77	0.32	0.20	0.96	0.84	0.61	0.87	0.54	0.35	0.52	0.27	0.25	0.82	0.37	0.29	**1.00**	**1.00**	**1.00**

Table 3: Topology preservation under a relative error bound of 10^{-4}. SZ3 reports results before correction, while TopoA, pMSz, and EXaCTz report results after correction. All metrics are reported in terms of precision. MS denotes Morse–Smale segmentation-label precision. N/A indicates results that were not evaluated or failed to complete within a one-hour time limit.

Dataset	Before Correction				After Correction											
	SZ3				TopoA				pMSz				EXaCTz			
	CP	EG	CT	MS	CP	EG	CT	MS	CP	EG	CT	MS	CP	EG	CT	MS
QMCPack	0.66	0.52	0.50	0.87	0.98	0.90	**1.00**	0.98	**1.00**	0.69	0.42	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
AT	0.83	0.85	0.74	0.95	**1.00**	0.96	**1.00**	**1.00**	**1.00**	0.93	0.22	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
Vortex	0.96	0.88	0.78	0.90	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.83	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
Turbulence	0.98	0.96	0.76	0.91	N/A	N/A	N/A	N/A	**1.00**	**1.00**	0.77	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
NYX (512^3)	0.84	0.76	0.51	0.75	N/A	N/A	N/A	N/A	**1.00**	0.99	0.63	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
Combustion	0.76	0.22	0.14	0.39	0.99	0.98	**1.00**	0.96	**1.00**	0.95	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

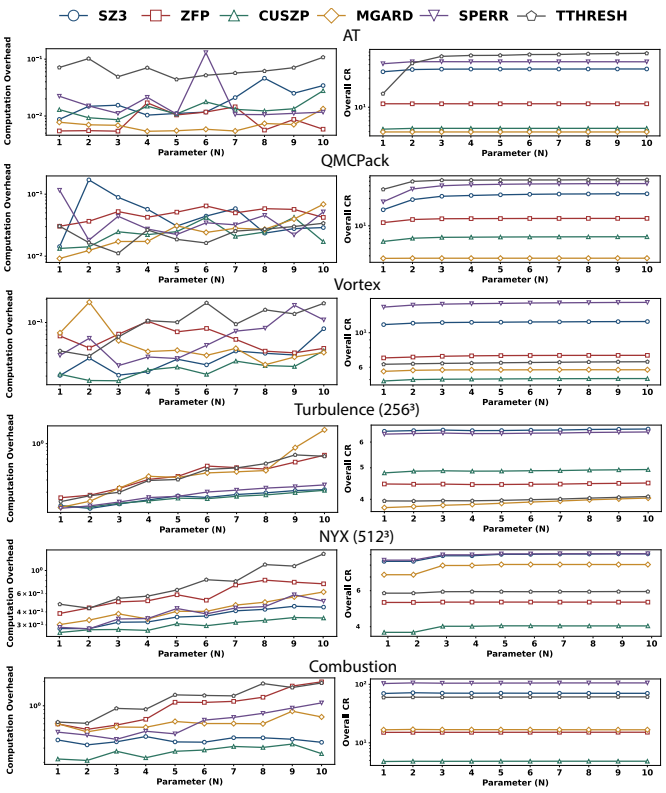

Figure 1: Trade-off between compression ratio (CR) and computation time with respect to parameter N across different datasets and compressors.

(e.g., ZFP achieves 0.99 EG recall on the Vortex dataset), but still distort CT due to global structural changes. SZ3/cuSZp often preserve CP reasonably well but can severely distort EG and CT, particularly on datasets with complex structures. Notably, cuSZp shows strong degradation in CT (e.g., 0.25 on Combustion), indicating that high-throughput GPU compressors may introduce nontrivial topological inconsistencies. SPERR and TTHRESH exhibit intermediate behavior, with moderate degradation across EG and CT.

E Additional Comparison with TopoA

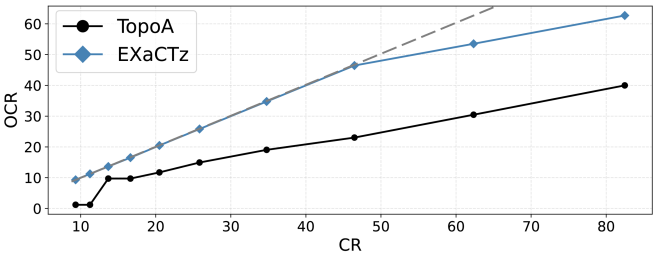

Figure 2: OCR vs. CR for EXaCTz and TopoA on the AT dataset using SZ3 as the base compressor across varying error bounds. The dashed line indicates OCR = CR.

Figure 2 compares the overall compression ratio (OCR) against the base compression ratio (CR) for EXaCTz and TopoA on the AT dataset using SZ3 as the base compressor across varying error bounds. The gap between CR and OCR reflects the storage overhead introduced by topology correction. As the error bound increases (higher CR), both methods require more storage overhead to preserve topology. EXaCTz consistently achieves higher OCR than TopoA across all compression ratios, indicating lower correction overhead. The dashed line (OCR = CR) represents the ideal case with zero correction overhead. EXaCTz closely tracks this line up to CR ≈ 45, after which the gap widens as larger error bounds introduce more topological distortions requiring additional corrections.

F Precision of Topological Features

Table 3 reports the precision of critical points (CP), extremum graphs (EG), contour trees (CT), and Morse–Smale segmentations (MS) under a relative error bound of 10^{-4}. Precision is defined as the fraction of topological features in the corrected data that are also present in the original data.

EXaCTz achieves 1.00 precision across all metrics and datasets. Note that EXaCTz also achieves 1.00 MS precision because our EG constraints preserve the steepest ascending and descending neighbors of every vertex, which preserves all integral lines and thus the induced MS segmentation. TopoA achieves 1.0 CT precision but lower EG precision (e.g., 0.90 on QMCPack and 0.96 on AT and Combustion). pMSz shows lower precision across most metrics, particularly for EG (e.g., 0.42 on QMCPack) and CT (e.g., 0.22 on QMCPack), as it targets Morse–Smale segmentation preservation and does not explicitly enforce extremum graph or contour tree consistency. Before correction, SZ3 exhibits low precision across all metrics, confirming that lossy compression introduces distorted topological features.

G Error Distribution Analysis

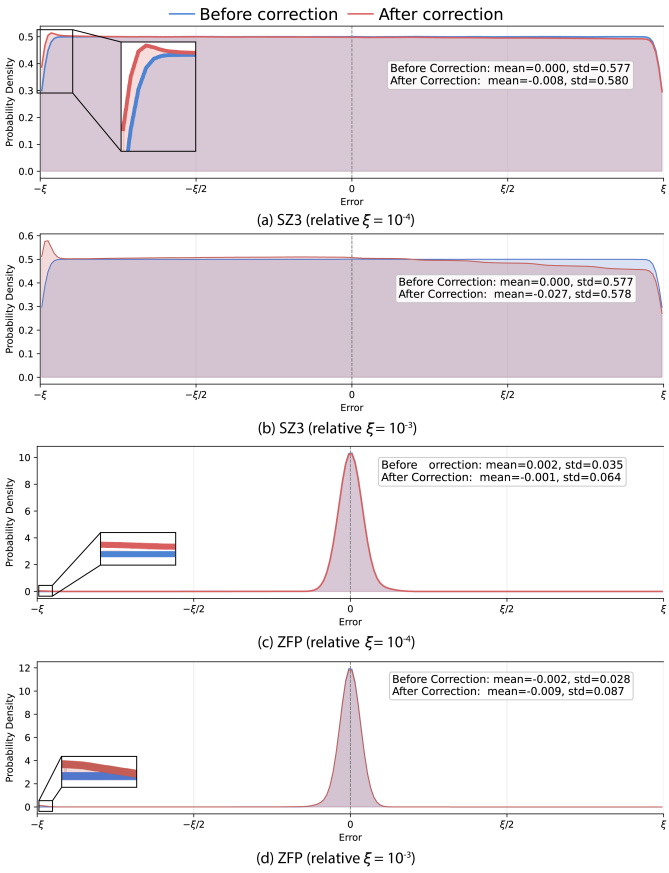

Figure 3: Error distributions before and after EXaCTz correction on the NYX (512^3) dataset with (a, b) SZ3 and (c, d) ZFP as base compressors (relative error bound $\xi = 10^{-4}$ and $\xi = 10^{-3}$).

We analyze the error distributions before and after EXaCTz correction to quantify the effect of monotonic decreasing edits on the error distribution. Figure 3 shows the results on the NYX (512^3) dataset using SZ3 and ZFP as base compressors under two relative error bounds ($\xi = 10^{-4}$ and $\xi = 10^{-3}$).

The before-correction error distributions are consistent with known compressor characteristics [7]: near-uniform for SZ3 and approximately normal for ZFP. After correction, both compressors show a slight negative shift in mean (e.g., from 0.000 to -0.008 for SZ3 and from 0.002 to -0.001 for ZFP at $\xi = 10^{-4}$). The bias increases with larger error bounds (e.g., SZ3 mean shifts to -0.027 at $\xi = 10^{-3}$), as more topological distortions require more edits. For SZ3, as shown in Figure 3 (a) and (b), the after-correction error distribution accumulates near $-\xi$, indicating that more vertices are driven to their lower bound $f_i - \xi$.

H Edit Map Comparison

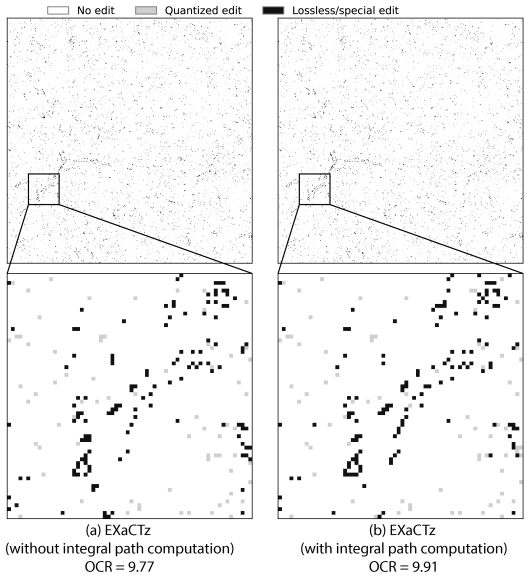

Figure 4: Edit maps of EXaCTz (a) without and (b) with integral path computation on a 2D NYX slice (SZ3, $\xi = 10^{-4}$).

We compare the edit maps of EXaCTz with and without integral path computation to visualize the impact on the edit pattern and storage overhead. Figure 4 visualizes the edit maps on a 2D NYX slice (SZ3, $\xi = 10^{-4}$). While some edits appear in only one of the two maps, this is because the two formulations enforce different constraint sets: the original event constraints target violations along integral paths, whereas the reformulated constraints enforce global critical point ordering. As a result, the spatial distribution of edits differs, but the total number of edits is nearly identical (4,902,506 vs. 4,904,024, a difference of 1,518), with the reformulated version producing slightly more edits overall, consistent with its over-approximation of the original constraints. Avoiding integral path computation introduces a slight OCR overhead (9.77 vs. 9.91) but reduces the correction time from 0.492s to 0.254s (1.94×).

I Weak Scaling on Downscaled Real Data

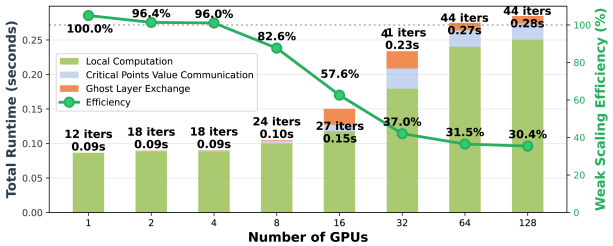

Figure 5: Weak scaling on downscaled real Turbulence data from 1 to 128 GPUs (SZ3, $\xi = 10^{-4}$). Stacked bars show the time breakdown; the line shows weak scaling efficiency.

To complement the upsampled weak scaling evaluation, we conducted weak scaling using downscaled versions of the real Turbulence (4096^3) dataset. As shown in Figure 5, the iteration count varies across scales (12 at 1 GPU vs. 44 at 128 GPUs), as larger global problem sizes introduce more topological distortions. The increased local computation from additional iterations dominates the runtime, reducing weak scaling efficiency from 100% at 1 GPU to 30.4% at 128 GPUs.

REFERENCES

[1] A. S. Almgren, J. B. Bell, M. J. Lijewski, Z. Lukić, and E. Van Andel. Nyx: A massively parallel amr code for computational cosmology. *The Astrophysical Journal*, 765(1):39, 2013. doi: 10.1088/0004-637x/765/1/39 1

[2] R. Ballester-Ripoll, P. Lindstrom, and R. Pajarola. TTHRESH: Tensor compression for multidimensional visual data. *IEEE Transactions on Visualization and Computer Graphics*, 26(9):2891–2903, 2020. doi: 10.1109/TVCG.2019.2904063 1

[3] C. Biwer. Nyx cosmological simulation dataset, 2019. doi: 10.21227/zh6w-kt72 1

[4] J. H. Chen. Terascale direct numerical simulations of turbulent combustion using S3D. *Computational Science & Discovery*, 2(1):015001, 2009. doi: 10.1088/1749-4699/2/1/015001 1

[5] P. R. C. Kent, A. Annaberdiyev, A. Benali, M. C. Bennett, E. J. Landinez Borda, P. Doak et al. QMCPACK: Advances in the development, efficiency, and application of auxiliary field and real-space variational and diffusion quantum monte carlo. *The Journal of Chemical Physics*, 152(17):174105, 05 2020. doi: 10.1063/5.0004860 1

[6] S. Li, P. Lindstrom, and J. Clyne. Lossy scientific data compression with SPERR. In *Proceedings of 2023 IEEE International Parallel and Distributed Processing Symposium*, pp. 1007–1017, 2023. doi: 10.1109/IPDPS54959.2023.00104 1

[7] P. Lindstrom. Error distributions of lossy floating-point compressors. In *Proceedings of the American Statistical Association*, pp. 2574–2589, 2017. 3

[8] J. Tierny, G. Favelier, J. A. Levine, C. Gueunet, and M. Michaux. The topology toolkit. *IEEE Transactions on Visualization and Computer Graphics*, 24(01):832–842, Jan. 2018. doi: 10.1109/TVCG.2017.2743938 1

[9] K. Zhao, S. Di, X. Liang, S. Li, D. Tao, J. Bessac et al. SDRBench: Scientific data reduction benchmark for lossy compressors. In *2020 IEEE International Conference on Big Data (Big Data)*, pp. 2716–2724, 2020. doi: 10.1109/BigData50022.2020.9378449 1