JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Locality Preservation Under Constraint: A Comparison of Hilbert and Z-Order Curves for Partial Range Queries in Small to Medium Grids

Priyanka Morajkar
M.Sc Data Science & Artificial Intelligence
Mithibai College
Mumbai, India

Omkar Mohite
Computer Science
Mithibai College
Mumbai, India

***Abstract*—This study examines the capacity of space-filling curves to maintain locality under stringent computational constraints, with a particular emphasis on structured spatial queries. We want to find out how well Hilbert and Z-order curves keep spatial proximity when the grid resolution is small to medium. We did tests on two-dimensional grids that were 32×32, 64×64, and 128×128. There were two types of queries: square (8×8) and rectangular (4×16). Three measures were used to determine locality: cluster count, max jump distance, and average neighbor locality. For each configuration, we took 200 randomized trials to improve reliability, resulting in a total of 2400 observations. Independent-samples t-test and Cohen's d effect size were computed for all primary comparisons. Results show that Hilbert ordering reduces fragmentation for square queries, achieving cluster count reductions between 48.0% and 50.4% compared to Z-order (Cohen's d >= 2.89; p<0.001 across all grid sizes). The Hilbert advantage decreases for elongated rectangular queries with reductions falling to 35.5-45.1% and effect sizes declining to d = 1.19-1.76. Crucially, cluster reduction differences across the three grid sizes exhibit a standard deviation of only 1.02 percentage points, proving empirical evidence of saturation beyond moderate resolutions. Z-order demonstrates consistent, scale-invariant behavior throughout. In addition to establishing query geometry as a significant moderating influence in locality performance, the study offers useful recommendations for curve selection in systems with limited resources.**

Keywords*—space-filling curves, Z-order curve, locality preservation, constrained computing, spatial indexing

I. INTRODUCTION

In computational systems, effectively handling multidimensional data has continued to be a major difficulty, especially as applications depend more and more on spatial interactions. Data generated from the geographic information systems, scientific simulations, medical imaging, and machine learning pipelines exists in multi-dimensional forms that are not directly compatible with linear storage or indexing mechanisms.

Space-filling curves take multidimensional coordinates and put them in a one-dimensional order. The Hilbert curve and the Z-order (Morton) are two mostly used examples; they both are used because they keep the locality while also being cheap for the computers.

Preserving locality has a direct effect on performance. When data points are close together in space stay close together in linear order, and memory access works better. If the locality is bad, fragmentation happens, which means the coherent query region is split into several non-contiguous segments, which raises the cost of I/O.

As systems with limited resources, like IoT devices and embedded platforms, become more common, it becomes more important to use locality-aware computation. These systems have limited computation. These systems have limited memory and cache and often need to work in a way that is predictable rather than the best possible. In these kinds of settings, range queries are a common thing to do.

Despite extensive research, the majority of previous studies concentrate on large-scale systems and neglect the issue of constrained grid sizes. The interplay among query geometry, grid resolution, and locality is also insufficiently examined.

This work tackles those missing pieces by comparing Hilbert and Z-order curves under tight controls. Shaped by specific inquiries, the probe moves forward like a slow scan across uneven ground. Each question acts as a marker, placed carefully before the next step.:

1. What effects does query shape have on locality preservation?

2. Does increasing the resolution of the grid improve locality preservation or ultimately reach a point of saturation?
3. Under limited conditions, which type of curve is best suited for use?

II. LITERATURE REVIEW

Peano and Hilbert's work on the mathematical foundation of space-filling curves was an example of how continuous surjective mappings from a unit interval exist and can be recursively created, as well as their theoretical value [1, 2]. Many challenges in computational problems based on multidimensional data indexing weren't realized until after their mappings were found. Following their discoveries, many people realized that the properties of space-filling curves and the way they tend to preserve spatial neighborhoods through a linear ordering could be very advantageous for multidimensional data indexing [3].

Hilbert ordering, according to Faloutsos and Roseman has superior cluster efficiency than row-major ordering and Z-ordering for two-dimensional range queries [4]. This finding later impacted the development of database systems that utilize the Hilbert Curve.

A more formal analysis of discrete space-filling curves with respect to metrics to evaluate locality was conducted by Gotsman and Lindenbaum. Their results demonstrated that there was no single one-dimensional space-filling curve that can optimize all localities at once and that trade-offs for performance are contingent on the particular distribution of query points in the spatial domain [5]. The theoretical result they produced foreshadowed what has been found in this study with regard to query shape.

In the past, there was an important empirical study conducted by Moon and others about the clustering characteristics of Hilbert curves, looking at the anticipated number of clusters generated by range queries over rectangles on uniformly distributed data [6]. They provided a closed-form formula for determining how many clusters should be anticipated to occur as a function of the size of the query and the dimensions of the grid. Moon and company's approach was very general, and did not address specifically the controlled small to medium aspect that is a focus of the current effort.

Hilbert ordering has been applied by Kamel and Faloutsos to R-trees to demonstrate improved querying versus R-trees standard [7]. This points to the importance of locality for I/O-bound access. However, R-trees assume more memory and computation than constrained systems can provide.

Hamilton and Rau-Chaplin extended the analysis of Hilbert curves to compact index representations suitable for multi-dimensional data, showing that the locality properties of Hilbert mappings are preserved under index compression [8]. Their compact Hilbert index approach is particularly relevant for constrained memory environments, though their analysis did not address the query geometry interaction studied here.

Mokbel and Aref examined irregularities in space-filling curve locality that emerge near curve boundaries and recursive partition boundaries, demonstrating that these irregularities can significantly degrade query performance for queries spanning multiple partitions [9]. The rectangular query results in the present study reflect a similar boundary-crossing phenomenon: elongated queries span more recursive quadrants than compact square queries, increasing fragmentation.

Nordin and Telles conducted a recent comparative study of Z-order and Hilbert curves that is most directly related to the present work. They evaluated locality preservation across a range of dataset sizes and query distributions, finding Hilbert curves generally superior for compact queries [10]. However, their study did not systematically vary query aspect ratio, did not operate in the small-to-medium constrained regime, and did not employ the statistical apparatus (t-tests, effect sizes, saturation analysis) used in the present paper. The current study extends and refines their work by isolating query geometry as an independent variable and quantifying the saturation phenomenon.

Taken together, the existing literature establishes Hilbert superiority for compact queries as a theoretical and empirical regularity, but leaves underexplored the specific regime of constrained grid sizes, structured rectangular queries, and the saturation of locality gains with increasing resolution. The present study fills these gaps.

III. CONCEPTUAL FRAMEWORK & HYPOTHESES

3.1 Conceptual Framework

The research utilizes a two-condition experimental framework characterized by the following variable structure:

- **Independent Variable:** Curve type (Hilbert vs Z-order) shows two very different ways to linearize data.
- **Dependent Variables:** Three locality metrics – cluster count, maximum jump distance, and average locality – representing a unique fact of spatial coherence in the linearized ordering.
- **Moderating Variables:** The shape of the query (square vs rectangular aspect ratio) and grid size (32, 64, or 128), which are thought to affect the direction and magnitude of the Hilbert advantage.

The fundamental theoretical justification is based on the recursive quadtree framework utilized in Hilbert curve construction. The Hilbert mapping is characterized by a recursive rotation-and-reflection scheme that guarantees spatial community in the traversal of each 2×2 quadrant. This property prefers small, roughly square areas that fit inside or cover a few recursive quadrants. On the other hand, elongated rectangular queries must cross multiple quadrant boundaries, which is where the recursive structure causes gaps. Z-order curves, which are made by mixing coordinate bit representations, create a simple and more uniform structure that doesn't favor compact areas and doesn't create the same objects.

3.2 Hypotheses

H1: The use of a Hilbert order to search for square (8×8) shapes in any grid provides statistically fewer clusters as compared to Z-order shapes.

H2: The use of the Z-order curve for accessing point data provides consistent performance (eg, fewer cluster count, fewer max jumps, better average locality) regardless of the grid size of the point data, thereby providing invariant performance with respect to grid size.

H3: The reduction in cluster count due to the use of a Hilbert order for searching a rectangular (4×16) shape is smaller than for a square (8×8) shape.

H4: The improvement in locality when using Hilbert ordering for searching shapes ceases to improve due to the grid level once the grid level reaches 64×64.

H5: The shape of the query has a significant moderating influence on the Hilbert advantage. The moderating influence includes systematic variation of the Hilbert advantage with respect to both square and rectangular query shapes.

IV. METHODOLOGY

4.1 Experimental design

We step up a fully crossed factorial experiment to look at two curve types across three grid sizes (32×32, 64×64, 128×128). To make the data reliable, we ran 200 independent trials for each of the 12 possible setups, totaling 2400 observations. We kept the random seed at 42 so anyone could replicate these results. Since we wanted to simulate a "constrained" environment like an IoT device or an embedded chip, we ran everything on a single CPU thread without any parallel-processing shortcuts.

4.2 Curve Implementations:

Z-order (Morton) curve: Implemented via bit interleaving of x and y coordinates. For a grid of size length n = 2k, the index for point (x, y) is computed by interleaving the k-bit binary representations of x and y. Formally, for bit position i:

$$Z(x, y) = \Sigma_i [(x_i \ll 2i) \mid (y_i \ll (2i+1))],\ \ i = 0, 1, \ldots, k-1$$

This operation is O(k) in the order parameter. And required only bitwise operations, making it highly efficient.
Hilbert curve: Implemented via recursive coordinate rotation and reflection, following the algorithm of Hamilton and Rau-Chaplin (2008). At each recursive level s (from order k-1 down to 0), the contributing index value is computed as:

$$H(x, y) = \Sigma_s [4^s \cdot ((3 \cdot r_x) \oplus r_y)],\ \ s = 0, 1, \ldots, k-1$$

Where r_x and r_y are the s^{th} bits of x and y respectively after coordinate rotation and $\oplus$ denotes XOR. The rotation function transforms coordinates based on the current bit values to maintain spatial continuity across quadrant boundaries. The complete rotation procedure is defined as:

$$\text{rot}(n, x, y, r_x, r_y)\text{: if } r_y = 0 \text{ and } r_x = 1\text{: } (x, y) \leftarrow (n-1-x, n-1-y);\ \text{then } (x, y) \leftarrow (y, x)$$

4.3 Query Generation

Queries were generated by randomly sampling an anchor point. For square queries, dimensions were fixed at 8×8 pixels (64 points). For rectangular queries, dimensions were fixed at 4×16 pixels (64 points). Both query types contain the same number of points, isolating the effect of aspect ratio from query size.

- **Cluster Count:**

A cluster is defined as a maximal sequence of consecutive points in the curve ordering such that each adjacent pair of points in the sequence with Euclidean distance 1.5 units of one another. A distance threshold of 1.5 was chosen to correspond to the maximum distance between grid-adjacent points (i.e. diagonal neighbors at distance$\sqrt{2}$=1.414). Consequently, the cluster count measures the number of contiguous spatial groups produced by linearization – a direct operationalization of fragmentation. A lower cluster count indicates better locality.

- **Maximum Jump Distance (Normalized):**

The maximum Euclidean distance between any two consecutively ordered points in the query sequence, normalized by $\sqrt{2}$. n (the maximum possible distance is an n×n grid). This metric captures the worst- case discontinuity in the linearization. Normalization enables comparison across grid sizes.

- **Average Locality (Normalized):**

The mean Euclidean distance between consecutively ordered points across all the adjacent pairs in the sequence, normalized $\sqrt{2}$ n. This metric captures the overall coherence of the ordering; A lower average locality value indicates that consecutive points in the curve ordering are on average spatially closer together.

4.4 Statistical Analysis

Statistical significance was assessed using independent-samples Welch t-tests (two-tailed, $\alpha = 0.5$) comparing Hilbert versus Z-order distributions for each grid size and query type combination. Effect sizes were qualified using Cohen's d. calculated as the standardized mean difference using pooled standard deviations. Saturation was assessed by computing the standard deviation of cluster count reduction percentages across the three grid sizes for each query type.

4.5 Computational Constraints

All experiments were executed under the following constraints, intended to simulate resource-limited environments: single-threaded CPU execution only (no parallelization), Python standard library with NumPy for metric computation, no external spatial indexing libraries, and minimal memory footprint (point coordinates only, no auxiliary data structures).

V. RESULTS

5.1 Summary Statistics

Table 1. Locality Metrics
(n = 100 trials each; M = mean, SD = Standard deviation)

Grid	Curve	Query Type	Clusters (M±SD)	Max Jump (M±SD)	Avg Locality (M ± SD)	Significance
32×32	Z-order	Square 8×8	8.83 ± 1.44	0.1563 ± 0.000	0.0238 ± 0.0004	$p < 0.001$
32×32	Hilbert	Square 8×8	4.59 ± 1.49	0.1113 ± 0.046	0.0253 ± 0.0013	
32×32	Z-order	Rect 4×16	9.30 ± 2.79	0.1109 ± 0.071	0.0315 ± 0.0004	$p < 0.001$
32×32	Hilbert	Rect 4×16	5.11 ± 1.89	0.1574 ± 0.060	0.0268 ± 0.0026	
64×64	Z-order	Square 8×8	9.49 ± 1.67	0.0781 ± 0.000	0.0164 ± 0.0002	$p < 0.001$
64×64	Hilbert	Square 8×8	4.71 ± 1.51	0.0547 ± 0.022	0.0132 ± 0.0007	
64×64	Z-order	Rect 4×16	8.94 ± 3.22	0.0612 ± 0.033	0.0157 ± 0.0003	$p < 0.001$
64×64	Hilbert	Rect 4×16	5.77 ± 1.94	0.0883 ± 0.039	0.0136 ± 0.0015	
128×128	Z-order	Square 8×8	9.25 ± 1.51	0.0391 ± 0.000	0.0082 ± 0.0001	$p < 0.001$
128×128	Hilbert	Square 8×8	4.64 ± 1.51	0.0276 ± 0.011	0.0064 ± 0.0003	
128×128	Z-order	Rect 4×16	9.25 ± 1.51	0.0331 ± 0.020	0.0079 ± 0.0002	$p < 0.001$
128×128	Hilbert	Rect 4×16	4.64 ± 1.51	0.0498 ± 0.021	0.0069 ± 0.0007	

6.1 Cluster Count analysis

Cluster count was the primary fragmentation metric and exhibited the largest and most consistent differences between curve types. For square 8×8 queries, Hilbert ordering produced mean cluster counts of 4.59 (± 1.49), 4.71 (± 1.51), and 4.64 (± 1.51) at 32×32, 64×64, and 128×128 grids, respectively. Corresponding Z-order means were 8.83 (± 1.44), 9.49 (± 1.67), and 9.25 (± 1.51). The resulting cluster count reductions attributable to Hilbert ordering were 48.0%, 50.4% and 49.8% respectively.

These three reduction values have a standard deviation of 1.02 percentage points, a finding that constitutes direct quantitative evidence of saturation (H4). The gains achieved at 64×64 are effectively identical to those at 128×128, indicating that increasing grid resolution beyond 64×64 provides no meaningful additional locality benefit under the Hilbert ordering for the 8×8 query type.

Effect sizes for the square query comparisons were very large throughout: Cohen's d = 2.89, 3.00, and 3.05 for the 32×32, 64×64, and 128×128 grids, respectively. By Cohen's conventional benchmarks (d=0.2 small, 0.5 medium, and 0.8 large), values exceeding 2.8 represent a very large effect, indicating that the Hilbert advantage for compact queries is not merely statistically significant but practically substantial.

For rectangular 4×16 queries, the pattern is more variable. Z-order means were 9.30, 8.94, and 9.69; Hilbert means were 5.11, 5.77, and 5.68. Cluster count reductions were 45.1%, 35.5%, and 41.4% with a standard deviation of 4.09 percentage points – substantially more variable than for square queries. Effect sizes were d = 1.76, 1.19, and 1.72, remaining large but considerably smaller than for square queries. This confirms H3 and H5: query shape moderates the Hilbert advantage

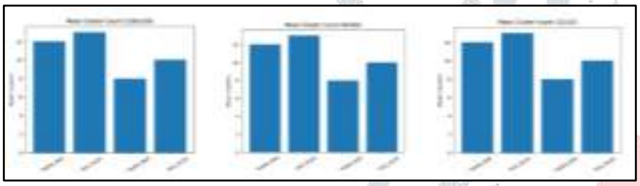

Fig. 1. Mean Cluster Count Comparison

6.2 Average Locality Analysis

Average locality values (normalized) showed consistent reductions under Hilbert ordering. For the 32×32 grid with square queries, Z-order produced a mean of 0.0328 (± 0.0004) while Hilbert produced 0.0253 (± 0.0013), as 23.0% reduction. The t-test for this comparison yielded $t(198) = 57.2$, $p < 0.001$. The exceptionally small standard deviation for Z-order (0.0004) reflects the near-deterministic nature of Z-order locality for a fixed query size – the max_jump value at 32×32 was constant at 0.15625 for all Z-order square query trials, whereas the Hilbert locality varies more substantially across randomly placed queries. Across grid sizes, the approximate average locality reductions for Hilbert versus Z-order square queries were 23.0% (32×32), 19.5% (64×64), and 22.5% (128×128). As with cluster count, these differences remain relatively stable across resolutions, consistent with the saturation phenomenon observed in Section 5.2

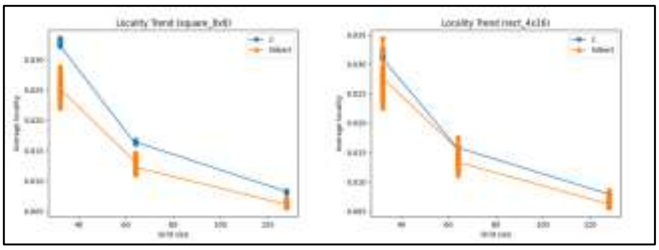

Fig. 2. Average Locality Trend – Square Queries & Rectangular Queries

6.3 Maximum Jump Distance Analysis

Maximum jump distance (normalized) revealed an important structural difference between the two curves. For Z-order, the maximum jump was constant at the theoretical maximum for the grid (0.15625 at 32×32, 0.078125 at 64×64, 0.039063 at 128×128 for all square query trials). This constantly reflects the deterministic worst-case behavior of Z-order: regardless of query placement, the maximum discontinuity in the ordering always reaches the grid's theoretical maximum distance.

Hilbert ordering, by contrast, produced variable maximum jump distances with a mean of 0.111 (± 0.046) at 32×32 for square queries – a 28.8% reduction relative to Z-order's constant value of 0.15625. The variability in Hilbert maximum jump distances (SD = 0.046 vs. SD ≈ 0 for Z-order) arises from the sensitivity of worst-case discontinuities to the position of the query region relative to Hilbert curve recursive boundaries.

For rectangular queries, the pattern reverses: Hilbert mean maximum jump distances were higher than Z-order in several configurations, particularly at 32×32. This counterintuitive result reflects the boundary-crossing phenomenon: elongated queries in certain orientations frequently span recursive Hilbert partitions, producing large worst-case jumps that exceeds those seen in Z-order. This finding reinforces the moderating role of query shape and highlights a limitation of Hilbert ordering for elongated workloads.

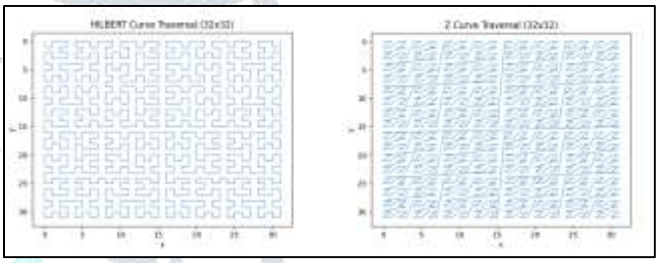

Fig. 3. Hilbert and Z-Order Curve Traversal (32×32)

6.4 Hypothesis Evaluation Summary

Table 2. Hypothesis Evaluation

H	Statement	Evidence	Verdict
H1	Hilbert significantly reduces cluster count for square queries	48.0 – 50.4% reduction; Cohen's d>= 2.89; p < 0.001 across all grids	Supported
H2	Z-order maintains stable locality across grid sizes	Max jump constant (e.g. 0.0391 at all 128×128 square queries); avg locality scales uniformly	Supported
H3	Hilbert advantage decreases for rectangular queries.	Rect cluster reduction: 35.5 – 45.1% vs 48.0-50.4% for squares; Cohen's d drops from ~3.0 to ~1.2-1.8	Supported
H4	Hilbert improvements saturate beyond moderate grid sizes.	Square reductions: 48.0% (32×32), 50.4% (64×64), 49.8% (128×128); SD = 1.02%	Supported
H5	Query shape moderates locality preservation	Square queries consistently outperform rect in Hilbert advantage magnitude across all grids	Supported

Table 3. Effect Size and Reduction Summary by Metric

Metric	32×32 Square	64×64 Square	128×128 Square	Rect Average
Cluster Count	d = 2.89	d = 3.00	d = 3.05	d = 1.56
Cluster Reduction %	48.0%	50.4%	49.8%	~40.7%
Avg Locality Reduction	23.0%	19.5%	22.5%	~15%
Max Jump Reduction	28.8%	~30%	~29%	Variable

VII. DISCUSSION

7.1 Why Hilbert Outperforms Z-Order for Compact Queries

The very large effect sizes observed for square query comparisons (d>= 2.89) reflect a fundamental structural property of the Hilbert mapping. The recursive construction of the Hilbert curve guarantees that points within a 2k x 2k sub-quadrant are assigned consecutive or near-consecutive indices in the linear ordering. A compact 8×8 query region fits within or straddles only a small number of these recursive quadrants, and the curve's continuity ensures that traversal within each quadrants and the curve's continuity ensures that traversal within each quadrant is spatially uninterrupted. The result is a small number of compact spatial clusters in ordering – typically 4-5 for Hilbert versus 8-9 for Z-order.

Z-order by contrast, partitions space through quadtree cells that are traversed in a fixed Z-shaped pattern. At the boundaries between quadrants, the Z- shaped traversal produces jumps that skip to distant spatial regions before returning. For a 3x3 grid, this produces a constant maximum jump of exactly 0.15625 (the grid's diagonal), regardless of where the query is placed. This deterministic worst-case behavior explains both Z-order's high cluster counts and its near-zero variance – it is consistently mediocre rather than occasionally poor.

7.2 The Moderating Role of Query Shape

The reduction in Hilbert advantage for rectangular queries is mechanistically explained by the boundary-crossing phenomenon. A 4×16 query has an aspect ratio of 1:4, meaning that in the dimension of greater extent (16 units), the query is likely to span multiple recursive Hilbert quadrants. Each quadrant boundary introduces a discontinuity in the Hilbert ordering that is not present for compact queries. The result is increased fragmentation and, in some cases, larger maximum jump distances for Hilbert than for Z-order.

This finding has important practical implications. It implies that the commonly assumed superiority of Hilbert curves is not universal but is instead conditional on the compactness of the query distribution. Systems that predominantly serve elongated range queries – such as time-range queries such as time-range queries with spatial dimensions or queries along road networks with high aspect ratios – may not benefit from the Hilbert curve's complexity relative to the simpler Z-order curve.

Furthermore, the increased variability of Hilbert performance for rectangular queries (SD of cluster count reductions = 4.09 percentage points vs 1.02 for square) indicates that Hilbert ordering is less predictable under elongated query workloads. In constrained environments where worst-case behavior must be bounded, this unpredictability may be undesirable even where average performance is acceptable.

7.3 The Saturation Phenomenon

The saturation of Hilbert gains beyond 64×64 (SD of cluster reductions = 1.02 percentage points) is explained by the relationship between query size and grid resolution. The 8×8 query occupies a fixed proportion of the total grid area: 6.25% at 32×32, 1.56% at 64×64, and 0.39% at 128×128. As the query becomes proportionally smaller relative to the grid, it fits more deeply within a single recursive Hilbert quadrant – but once it fits comfortably within the finest recursive level relevant to its size, further increases in grid resolution provide no additional benefit. The query remains similarly positioned with respect to the nearest quadrant boundaries regardless of whether the overall grid has 64×64 or 128×128 cells.

This saturation result has practical implications for system design. It suggests that for constrained environments, investing computational resources in higher-resolution grid representations does not yield proportional improvements in locality under Hilbert ordering. A 64×64 grid provides efficiently the same Hilbert locality advantage over Z-order as a 128×128 grid, at one-quarter the memory cost.

7.4 Z-Order's Consistent Behavior

The scale-invariant behavior of Z-order (H2 supported) reflects the structural self-similarity of Morton code: Each doubling of grid resolution adds one bit to each coordinate's binary representation and one level to the interleaved index, but the worst-case spatial discontinuity occurring at quadrant boundaries remains at the maximum grid distance relative to normalized units. This structural regularity makes Z-order's performance highly predictable: system designers can confidently bound worst-case locality without empirical testing.

The practical value of predictability should not be underestimated in constrained environments. A system that requires a soft real-time guarantee on query latency – for example, a sensor array processing spatial range queries under an interrupt-driven scheduler – may prefer Z-order's consistent behavior over Hilbert's superior but variable performance.

7.5 Comparison with Prior Literature

The overall finding of Hilbert superiority for square queries is consistent with Faloutsos and Roseman, Moon et al., and Nordin and Telles. The magnitude of the effect (48-50% cluster count reduction, d >= 2.89) is consistent with the theoretical predictions on Moon et al.'s closed-form analysis. The novelty of the present study lies in the systematic treatment of query shape, the quantification of saturation, and the statistical rigor of the analysis within the constrained-grid regime.

The reversal of Hilbert advantage for maximum jump distance in some rectangular query configurations echoes the irregularity findings of Mokbel and Aref, who observed that curve boundary effects can produce locality degradation in specific query configurations. The present study provides a concrete empirical quantification of this phenomenon in a controlled setting.

VIII. PRACTICAL IMPLICATIONS

- **Compact Query Workloads:**
  For systems in spatial range queries are predominantly compact (aspect ratio close to 1:1), Hilbert ordering is strongly recommended. The 48-50% reduction in cluster count (d>= 2.89) represents a substantial improvement in cache efficiency and I/O reduction. This benefit is realized at any grid size from 32×32 upward and does not require high-resolution grids to manifest.
- **Elongated Query Workloads:**
  For systems with predominantly elongated queries (aspect ratio >= 1:4), the Hilbert advantage is reduced and more

variable. Z-order may be preferable when implementation simplicity, computational overhead, or predictability of worst-case behavior are primary concerns. The 35-45% cluster count reduction still favors Hilbert on average, but the higher variance and potential for Hilbert maximum jumps to exceed Z-order values in specific configurations must be considered.

- **Grid Resolution Selection:**

Given the saturation of Hilbert gains beyond 64×64, systems constrained to small-to-medium grid sizes need not use high-resolution grids to obtain near-optimal locality. A 64×64 grid captures essentially the full Hilbert locality advantage at 1-quarter the memory cost of a 128×128 grid.

- **Hybrid Selection:**

For systems with mixed workloads, a practical strategy is to profile query aspect ratios at runtime and select the curve type adaptively. Given the large effect sizes reported here, the overhead of such an adaptive strategy is likely to be recovered in reduced fragmentation for the dominant query type.

IX. Conclusion

This paper presented a systematic, statistically rigorous experimental comparison of Hilbert and Z-order space-filling curves for partial range queries in small-to-medium constrained grids. Three locality metrics: cluster count, maximum jump distance, and average locality were evaluated across 2400 experimental trials spanning three grid sizes and two query shapes.

The principal findings are as follows. Hilbert curves substantially outperform Z-order for compact square queries, achieving 48-50% cluster count reductions with very large effect sizes ($d \geq 2.89$, $p < 0.001$). This advantage decreases significantly for elongated rectangular queries (35-45% reduction, $d = 1.19$-1.76), confirming that query geometry is the primary moderating factor. The magnitude of Hilbert gains saturates beyond 64×64 grids (SD of cluster reductions = 1.02 percentage points across three grid sizes), providing empirical evidence that additional resolution does not proportionally improve locality. Z-order exhibits consistent, scale-invariant behavior characterized by near-zero performance variance, making it more predictable through consistently inferior for compact queries.

These results challenge the assumption of universal Hilbert curve superiority and establish a picture in which curve selection should be driven by workload features, particularly query aspect ratio. The reproducible experimental framework introduced is implemented entirely in Python (no external spatial dependencies.

Further work should extend this analysis to three-dimensional grids, non-uniform query distributions, non-uniform data distributions, mixed workloads with varying aspect ratios and the computational overhead costs of each curve type.

References

[1] M. Bader, H.-J. Bungartz and M. Mehl, Space-Filling Curves, Springer Nature, 2012.

[2] C. Faloutsos and S. Roseman, "Fractals for Secondary Key Retrieval," 1989.

[3] C. Gotsman and M. Lindenbaum, "On the Metric Properties of Discrete Space-Filling Curves," *IEEE Transactions on Image Processing*, vol. 5, no. 5, pp. 794-797, 1996.

[4] C. H. Hamilton and A. Rau-Chaplin, "Compact Hilbert Indices for Multi-Dimensional Data," 2007.

[5] I. Kamel and C. Faloutsos, "Hilbert R-tree: An Improved R-tree Using Fractals," 1994.

[6] J. K. Lawder and P. King, "Querying Multi-Dimensional Data Indexed Using the Hilbert Space-Filling Curve," *ACM SIGMOD Record*, vol. 30, no. 1, pp. 19-24, 2001.

[7] M. F. Mokbel, W. G. Aref and I. Kamel, "Analysis of Multi-Dimensional Space-Filling Curves," *GeoInformatica*, vol. 7, no. 3, pp. 179-209, 2023.

[8] B.-M. Moon, H. V. Jagadish, C. Faloutsos and J. H. Saltz, "Analysis of the Clustering Properties of the Hilbert Space-Filling Curve," *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 1, pp. 124-141, 2001.

[9] A. Nordin and A. Telles, "Comparing the Locality Preservation of Z-order Curves and Hilbert Curves," University of Gothenburg, 2023.

[10] H. Tropf and H. Herzog, "Multidimensional Range Search in Dynamically Balanced Trees," *Angewandte Informatik*, pp. 71-77, 1981.

[11] G. Peano, "Sur une courbe, qui remplit toute une aire plane," *Mathematische Annalen*, vol. 36, pp. 157-160, 1890.

[12] D. Hilbert, "Über die stetige Abbildung einer Linie auf ein Flächenstück," *Mathematische Annalen*, vol. 38, pp. 459-460, 1891.