LOCALITY AND CLUSTERING PERFORMANCES OF SPACE-FILLING CURVES

By

HUNG-CHI SU

Bachelor of Chemical Engineering
National Cheng-Kung University
Tainan, Taiwan
1986

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1999

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
December, 2003

LOCALITY AND CLUSTERING PERFORMANCES OF

SPACE-FILLING CURVES

Thesis Approved:

Thesis Advisor

Dean of the Graduate College

ACKNOWLEDGEMENTS

This dissertation would not have been completed without excellent guidance and assistance from a number of people. First and foremost, my deepest gratitude and appreciation go to such a dedicated and respectable teacher, and the chair of my dissertation committee, Dr. H. K. Dai. Without his excellent guidance and encouragement, I would have never been sustained this valuable journey. Likewise, many thanks are indebted to other committee members, Dr. George E. Hedrick, Dr. John P. Chandler and Dr. Kaladi S. Babu for their incisive comments and suggestions throughout the dissertation process.

Words cannot express just enough my appreciations to my family whose love, support and belief always inspire strength and courage in me, especially in those tough times.

Finally, I would like to thank the faculty and staff of Computer Science Department for their help for the past years, with which made my Ph. D. possible.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Mapping schemes between discrete multi-dimensional and discrete linear spaces are essential to applications that involve multi-dimensional data because the linearization techniques of multi-dimensional arrays or grids are needed. Typically, sample applications include multi-dimensional indexing methods [Ore86, ARR+97, BRWW97, GG98, AE99, BKK99, LK00], multimedia databases [SR00, BBK01, ACE+02], geographic information systems [AM90], image processing [LZ84, VG91], similarity search [BBB+97, LLL01], data structures and algorithms [BP82, ARR+97], parallel computation [KOR95, Zum01], etc.

A space-filling curve is a mapping scheme to number the points in a discrete multi-dimensional space by the integers starting from one to the total number of points in the space. As a result, for every point in a discrete multi-dimensional space, viewed equivalently as a grid, there exists a unique point corresponded in linear space (in the range between one and the total number of points in the multi-dimensional space), and vice versa. Thus, a space-filling curve provides a linear traversal in a multi-dimensional space by visiting every point exactly once so that it imposes a linear order for all the points. Peano (1890) constructed the first space-filling curve, and many other space-filling curves have been proposed thereafter by Hilbert, Moore, Lebesgue, Sierpiński, Pólya, etc. For a comprehensive historical development of classical space-filling curves, see [Sag94]. Figure 1.1 lists some examples of space-filling curves in a 4×4 space.

As for a multi-dimensional applications, the mapping schemes (space-filling curves) serve as a pre-processing step by linearizing the multi-dimensional data, then existing linear data structures and algorithms can be adopted with a few modifications

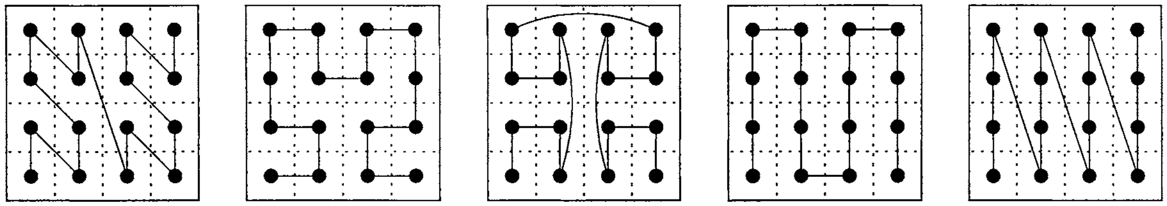


Figure 1.1: Examples of space-filling curves in 4×4 space.

to process these linearized data. For the sake of applicability, a space-filling curve is mostly desired to maintain locality. The reason is that, for multi-dimensional applications, most of the operations are on neighboring points (e.g., range queries in multi-dimensional databases). However, the locality will be deteriorated as the multi-dimensional data are mapped into linear space: nearby data may be mapped to far-away locations in the linear space and/or the far-away data may be mapped to nearby locations in the linear space. Consequently, the locality that has been lost under these circumstances hurts the performances of these multi-dimensional applications. Insomuch as getting better performance for these applications, we should adopt "good" space-filling curves that greatly maintain locality. The purpose of this research is to investigate locality and clustering performances of space-filling curves and also evaluate their applicability so as to help choosing more appropriate ones in practice.

For general applicability of space-filling curves, there are two different categories of measures. One is the locality preservation that reflects proximity between the points in multi-dimensional space; that is, close-by points in multi-dimensional space are mapped to close-by indices in linear space, or vice versa. The other one is clustering performance that measures the distribution of continuous runs of points (clusters) over identically shaped subspaces in a multi-dimensional space, and this category of measures can be characterized by the average number of clusters and the average inter-cluster distance within a subspace.

Among the space-filling curve families, z-order and Hilbert space-filling curves probably are the most popular ones but have different traits. Basically, z-order space-filling curve family is relatively simple and easy to implement by interleaving the bits of all the coordinates. However, Hilbert space-filling curve family is not so straightforward even though it has a very simple structure in a 2×2 space. Numerous empirical and analytical studies of various low-dimensional space-filling curves have been reported (see [AM90, Jag90, Jag97, MJFS01] for details) but there is still some work for us to investigate furthermore in this area such as locality preservation and inter-clustering performances. The objectives of this research are to:

1. investigate the locality preservations and clustering performances for different space-filling curves, especially on the most popular ones, z-order and Hilbert space-filling curve families,

2. derive the closed-form formulas for the measures to quantify the qualities of z-order and Hilbert space-filling curves, and

3. compare z-order curve family with Hilbert curve family by the derived closed-form of formulas from objective 2.

This dissertation is organized in the following manner. Chapter II presents the preliminaries about space-filling curves, including their definitions and constructions. It also contains a brief overview of the work that has been already conducted in this area. Chapters III and IV study two measures of locality preservation, including the derivations of closed-form formulas for Hilbert and z-order space-filling curve families. Chapters V and VI focus on the clustering performances to develop measures by mean number of clusters and also by mean inter-cluster distance within a subspace. Chapter VII is the conclusion.

CHAPTER II

PRELIMINARIES AND LITERATURE REVIEW

2.1 Definition of Space-Filling Curves and Their Features

For a positive integer n, denote $[n] = \{1, 2, \ldots, n\}$. An m-dimensional (discrete) space-filling curve of length n^m is a bijective mapping $C : [n^m] \rightarrow [n]^m$, where m is a positive integer, thus providing a linear indexing/traversal or total ordering of the grid points in the m-dimensional grid space $[n]^m$. (For convenience, we call a point in the m-dimensional space a grid point.) The m-dimensional grid space is said to be of order k if its side length $n = 2^k$; a space-filling curve is of order k if its codomain is of order k. An m-dimensional space-filling curve of order k is denoted by C_k^m (i.e., $C_k^m : [(2^k)^m] \rightarrow [2^k]^m$), and a family of m-dimensional curves of successive orders is denoted by $\mathcal{C}$. The generation of a sequence of m-dimensional space-filling curves of successive orders usually follows a recursive framework (on the dimensionality and order), which results in a few classical families, such as Gray-coded space-filling curves, Hilbert space-filling curves, and z-order space-filling curves (see, for examples, [AN00, MJFS01]).

2.1.1 Self-Similar Space-Filling Curves

In the 2-dimensional space, a space-filling curve of order k is said to be recursive if the space-filling curve can be divided into four equal-sized quadrants with the same structure (via rotation and/or reflection) [ARR$^+$97].

Alber and Niedermeier [AN98, AN00] formalize the idea and extend it for higher-dimensional spaces by defining the class of self-similar curves. The "self-similar" simply means that a space-filling curve C_k^m can be generated by putting together 2^m space-filling curves C_{k-1}^m along a particular curve with some suitable permutations on corners of C_{k-1}^m applied, recursively. Here, the suitable permutations (denoted by μ) on corners of C_{k-1}^m are the permutations that preserve the neighborhood-relations $n(i,j)$ on the indices i,j of the corners in C_{k-1}^m: $n(i,j) = n(\mu(i),\mu(j))$. For 2-dimensional space-filling curve, permutations are the operations of rotations and/or reflections. The space-filling curve C_1^m is called the "generator" for C_k^m. Alber and Niedermeier consider all the permutations μ to explore the construction of Hilbert space-filling curves and show that there is one Hilbert space-filling curve modulo symmetry in the 2-dimensional space and 1536 different structures in the 3-dimensional space.

In general, self-similar space-filling curves (recursive space-filling curves) have many advantages over non-self-similar curves. For instance, in the m-dimensional space with side length n, self-similar space-filling curves use compact recursive representation (formulas) to index all the data so that time complexity to determine the data is $O(\log(n^m))$; on the other hand, non-self-similar ones need to store mapping tables for all of the data so that the space complexity is $O(n^m)$ and time complexity of worst case to look up the table sequentially for data is also $O(n^m)$. Therefore, we will focus on the self-similar space-filling curves in our research and from now on, space-filling curves or curves mentioned in the following content indicate self-similar space-filling curves.

2.1.2 Constructing Space-Filling Curves

The coordinate system used in our work for m-dimensional space with side length n is a Cartesian coordinate system with the axes numbered $1, 2, \ldots, m$, and the indices on each axis are numbered $1, 2, \ldots, n$. Figure 2.1 demonstrates the coordinate system in the 2-dimensional space: the vertical direction is axis-1 and the horizontal direction is axis-2. When work in 2-dimensional space, we also call axis-1 x-coordinate, axis-2 y-coordinate, clockwise rotation (+)-rotation, and counterclockwise rotation (−)-rotation.

For a space-filling curve C, the lowest- and highest-indexed grid points are called

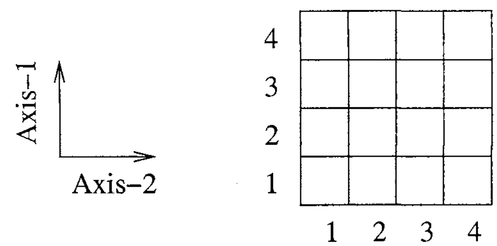


Figure 2.1: The coordinate system for the 2-dimensional space.

the entry and exit grid points, and denoted by $\partial_1(C)$ and $\partial_2(C)$, respectively. Since we are only interested in the self-similar space-filling curves, the constructions are therefore based on the recursive frameworks. Now, we focus on the self-similar space-filling curve C_k^m that is generated by interconnecting 2^m identical C_{k-1}^m-structured subcurves (via suitable permutations) along a C_1^m-structured curve. We denote the 2^m C_{k-1}^m-structured subcurves of C_k^m by $Q_\alpha(C_k^m)$, where $\alpha \in [2^m]$, and a subcurve is numbered by following the linear order along the C_1^m-structured curve (see Figures 2.3 and 2.6). We call the edge connecting $\partial_2(Q_\alpha(C_k^m))$ and $\partial_1(Q_{\alpha+1}(C_k^m))$ a connecting edge where $\alpha \in [2^m - 1]$.

2.1.2.1 Constructing z-Order Curves. A z-order curve is also known as a bit-interleaving curve because it traverses the grid points along the interleaved bits of their coordinates. An m-dimensional z-order curve of order k is denoted by Z_k^m. To index the grid points for Z_k^m, the steps are: (1) to subtract each coordinate by 1 for each grid point, (2) to interleave the binary bits resulted from Step (1) for each grid point, and (3) to add 1 to the results from Step (2) for each grid point. Figure 2.2 illustrates the steps for building z-order curves: (a) the grid point coordinates on axis-1 and -2, (b) the binary code of each grid point (after each coordinate subtracted by 1), (c) the indices of each grid point (after the binary code added by 1), (d) the z-order traversal sequence. By following the steps above, we can construct z-order curves of any order. Figure 2.2(e) and (f) are the results for curves of order 2 and 3, respectively.

Instead of the idea by interleaving bits of coordinates to construct a z-order curve,

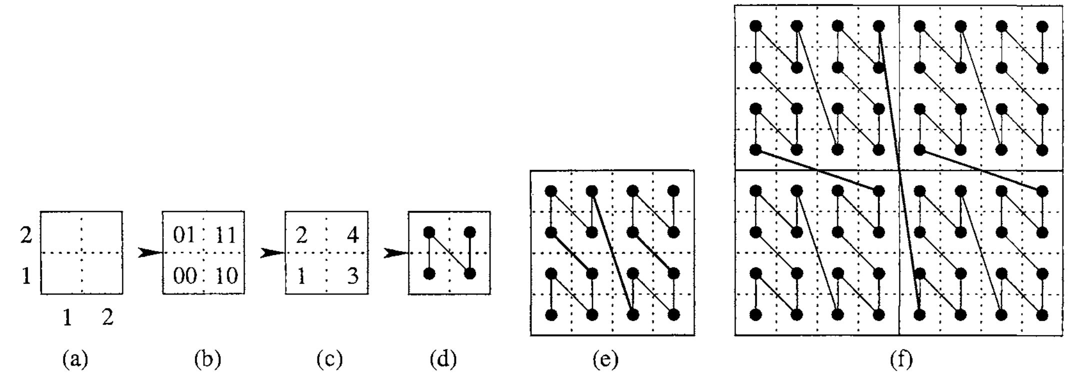


Figure 2.2: Construction of z-order curves in the 2-dimensional space. (a) Space of order 1, (b) interleaving codes (after subtracted by 1 on each coordinate), (c) indices for the grid points, (d) z-order curve of order 1 by connecting the grid points according to the indexing, (e) and (f) z-order curves of order 2 and 3, respectively.

we can construct the curve through recursion. The first step is to create a z-order curve of order 1 as shown in Figures 2.2(a), (b), (c) and (d). The induction step is that we can build $Z_{k'}^m$ by merging 2^m $Z_{k'-1}^m$-structured subcurves along the Z_1^m-structured curve for $k' = 2, 3, \ldots, k$. Note that there are no reflection or rotation operations applied to the construction of z-order curves. Figure 2.3 represents the construction of a 2-dimensional z-order curve from 2^2 curves of lower order (the decomposition into 2^2 curves of lower order, reversely). Based on the the recursive mechanism, we can generate a z-order curve of higher order.

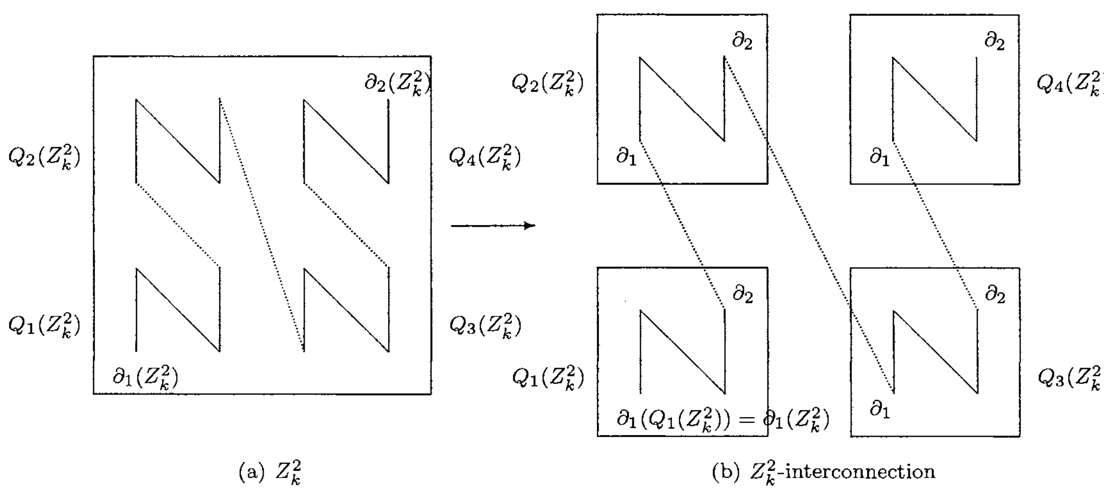


Figure 2.3: Generation of Z_k^2 in (a) from a Z_1^2-interconnection of four Z_{k-1}^2-structured subcurves in (b).

Construction of a higher-dimensional z-order curve Z_k^m from lower-dimensional Z_k^{m-1} is by the following steps:

1. Construct a z-order curve Z_1^{m-1}.

2. Add the m-th dimension to current coordinate system that is of $m-1$ dimensions,

3. Duplicate the curve Z_1^{m-1} along the m-th dimension, and let the exit grid point of this duplicated one inherit the connection link from the exit grid point in the original Z_k^{m-1} if there is one,

4. Add an edge to connect the exit grid of the duplicating curve in Step 3 to the entry grid point of the duplicated one (this step complete the construction of Z_1^m),

5. Follow the construction steps for z-order curve of higher-order to build Z_k^m from Z_1^m.

Note that there are no reflection or rotation operations in the construction. Figure 2.4 demonstrates the constructing steps. Basically, z-order curve traverses all the grid points by lexicographical order on their interleaved binary coded coordinates. We call a z-order curve canonical if it has the structure and orientation as the one generated from above steps. For example, Figure 2.2, 2.3 and 2.4(c).

2.1.2.2 Constructing Hilbert Curves. An m-dimensional Hilbert curve of order k, denoted H_k^m, is more complicated than a z-order curve. First, we define the orientation for a Hilbert curve H_k^m: the direction of the entry grid point to the exit grid point. (This direction is parallel to one of the axes in the coordinate system.) We call an H_k^m α-oriented if the direction of its entry grid point to exit grid point is parallel to the axis-α, and also call an H_k^m α^+-oriented (respectively, α^--oriented) if it is α-oriented and its entry grid point has less (respectively, greater) coordinate value on axis-α than its exit grid point does.

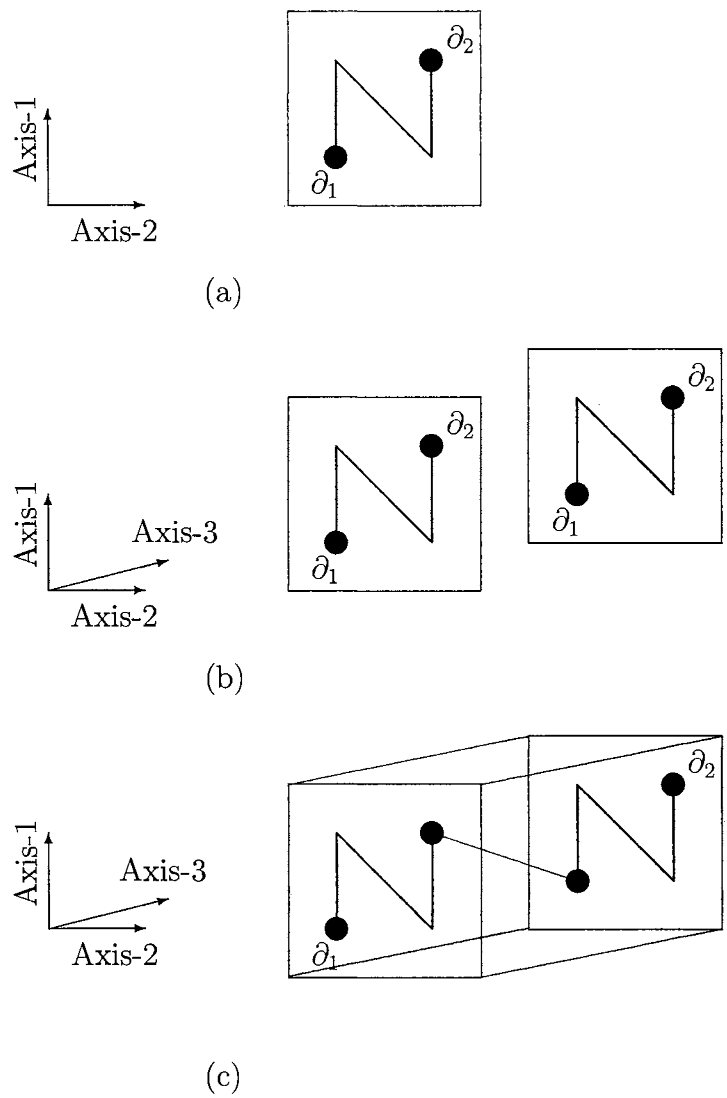
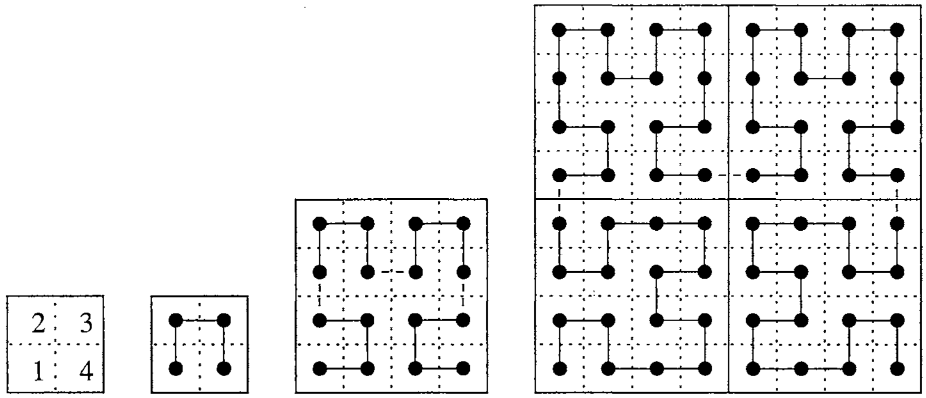


Figure 2.4: Construction of higher dimensional z-order curve. (a) A 2-dimensional coordinate and a Z_1^2; (b) adding a new dimension (axis) and then duplicating the Z_1^2; (c) connecting the exit and entry grid points of the duplicating and duplicated Z_1^2, respectively.



(a) (b) (c) (d)

Figure 2.5: Construction of Hilbert curves in 2-dimensional space. (a) The indices for H_1^2; (b) the traversal line for H_1^2; (c) H_2^2; (d) H_3^2.

Now, we focus on the construction of 2-dimensional Hilbert curve of higher order H_k^2 from H_1^2. The construction is still built through a recursive mechanism. The first step is to have a 2×2 space and have the indices 1, 2, 3, and 4 as seen in Figure 2.5(a). The generator of 2-dimensional Hilbert curve H_1^2 is shown in Figure 2.5(b). The induction step constructs a Hilbert curve $H_{k'}^2$ by merging four $H_{k'-1}^2$-structured subcurves along the H_1^2-structured curve for $k' = 2, 3, \ldots, k$ with (1) the first $H_{k'-1}^2$-structured subcurve left-right reflected and then $(+\frac{\pi}{2})$-rotated, (2) the fourth $H_{k'-1}^2$-structured subcurve left-right reflected and then $(-\frac{\pi}{2})$-rotated, and (3) the second and third ones remaining unchanged (see Figures 2.5(c) and (d)). Figure 2.6 shows the generation of H_k^2 from a H_1^2-connection of four H_{k-1}^2-structured subcurves.

For dimensionality greater than 2, there are many different structures of Hilbert curves as discussed in [AN98, AN00]. Moon, Jagadish, Faloutsos, and Saltz [MJFS01] suggest a construction of a higher-dimensional Hilbert curve H_k^m from a lower-dimensional one H_k^{m-1} through a recursive procedure:

1. Construct a Hilbert curve H_1^{m-1}.

2. Add the m-th dimension to current coordinate system that is of $m-1$ dimensions,

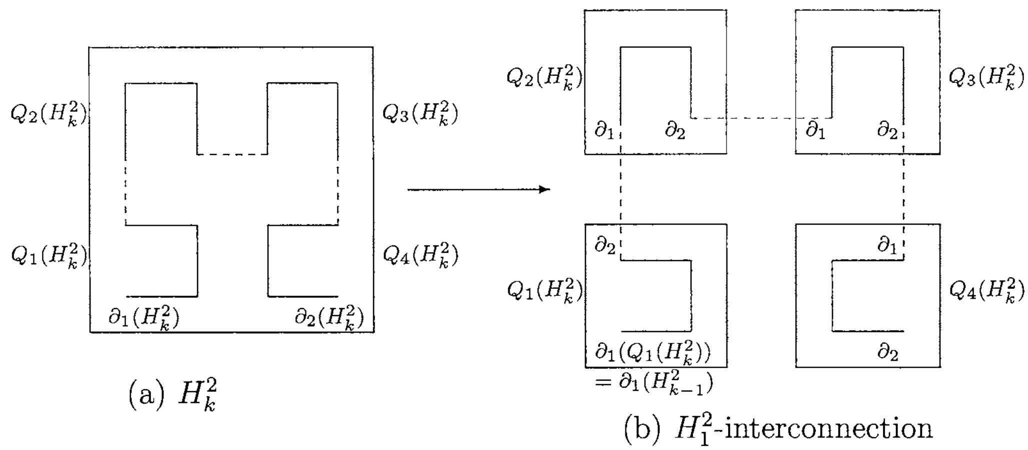


Figure 2.6: Generation of H_k^2 in (a) from a H_1^2-connection of four H_{k-1}^2-structured subcurves in (b).

3. Duplicate the curve H_1^{m-1} along the m-th dimension, and let the exit grid point of this duplicated one inherit the connection link from the exit grid point in the original H_k^{m-1} if there is one,

4. Add an edge to connect the exit grid point of the duplicating curve in step 3 to the entry grid point of the duplicated one , and change the orientation of these two grid point points to be m^+(others remain the same orientation) (this step complete the construction of H_1^m),

5. Construct first half of Hilbert curve H_k^m by putting together 2^{m-1} H_{k-1}^m along H_1^m-structured curve via suitable permutations on the 2^m H_{k-2}^m-structured grid points of H_{k-1}^m to maintain the orientations of the grid points (subcurves) within H_k^m, then duplicate the first half 2^m H_{k-1}^m via reflection along axis-m to build the rest of H_k^m.

Note that step 5 of the construction for higher dimensionality implies that an m-dimensional Hilbert curve H_k^m is symmetrical along the axis-m. Figures 2.7 and 2.8 demonstrate the constructions of H_1^3 and H_1^4, respectively (the dashed lines denote the new added dimension). We call an Hilbert curve canonical if it has the structure and orientation as the one generated from the above steps. For example, Figure 2.5, 2.6 and 2.7. Numerous algorithms in programming codes for constructing higher-dimensional Hilbert curves and generating visual images in lower dimensionality are proposed and published in literature (see [But69, LS94, LS97, BC98, Max98, LK01, Ras01] for details).

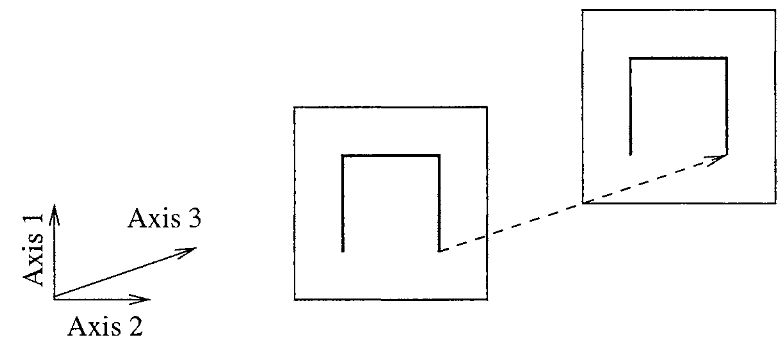


Figure 2.7: Construction of 3-dimensional Hilbert curve (H_1^3) from two 2-dimensional Hilbert curves (H_1^2).

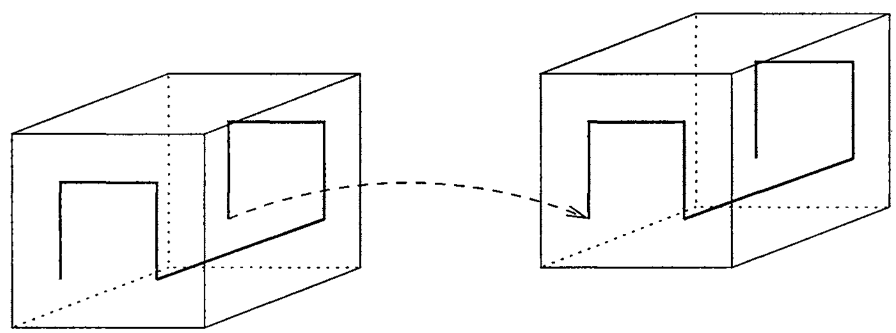


Figure 2.8: Construction of 4-dimensional Hilbert curve (H_1^4) from two 3-dimensional Hilbert curves (H_1^3).

2.2 Locality Preservation and Related Work

The locality preservation of a space-filling curve family is important for the efficiency of many indexing schemes, data structures, and algorithms in its applications such as spatial correlation in multi-dimensional indexing, compression in image processing, and communication optimization in mesh-connected parallel computing. To analyze locality, we need to precisely define its measures so that these measures could be used in practice; good bounds (lower and upper) on the locality measure can translate into good bounds on the declustering (locality loss) in one space in the presence of locality in the other space.

Denote by d_p the p-normed metric (Manhattan ($p = 1$), Euclidean ($p = 2$), and maximum metric ($p = \infty$)). A space-filling curve C is called continuous if $d_2(C(i), C(i+1)) = 1$ (the Euclidean metric is 1) for all $i \in [n^m - 1]$. Thus the traversal of a continuous space-filling curve C covering all grid points of $[n]^m$ is Hamiltonian with unit (Euclidean) steps. (Hilbert curve is a continuous space-filling curve while z-order curve is not.)

Some locality measures have been proposed and analyzed for space-filling curves in the literature. Pérez, Kamata, and Kawaguchi [PKK92] employ an average locality measure to quantify the proximity preservation of close-by points in the m-dimensional space $[n]^m$:

$$L_{\text{PKK}}(C) = \sum_{i,j \in [n^m] | i<j} \frac{|i-j|}{d_2(C(i), C(j))} \text{ for } C \in \mathcal{C},$$

and provide a hierarchical construction for a 2-dimensional $\mathcal{C}$ with good but suboptimal locality with respect to this measure.

Mitchison and Durbin [MD86] use a more restrictive locality measure parameterized by q:

$$L_{\text{MD},q}(C) = \sum_{i,j \in [n^m] | i<j \text{ and } d_2(C(i),C(j))=1} |i-j|^q \text{ for } C \in \mathcal{C}$$

to study optimal 2-dimensional mappings for $q \in [0, 1]$. For the case $q = 1$, the optimal mapping with respect to $L_{\text{MD},1}$ is very different from that in [PKK92]. For the case $q < 1$, they prove a lower bound for arbitrary 2-dimensional curve C:

$$L_{\text{MD},q}(C) \geq \frac{1}{1+2q} n^{1+2q} + O(n^{2q}),$$

and provide an explicit construction for 2-dimensional $\mathcal{C}$ with good but suboptimal locality. They conjecture that the space-filling curves with optimal locality (with respect to $L_{\text{MD},q}$ with $q < 1$) must exhibit a "fractal" character.

Voorhies [Voo91] defines a heuristic locality measure, tailored to computer graphics applications, and the corresponding empirical results indicate that the Hilbert space-filling curve family outperforms other curve families.

For measuring the proximity preservation of close-by points in the indexing space $[n^m]$, Gotsman and Lindenbaum [GL96] develop the following measures:

$$\begin{aligned} L_{\text{GL,min}}(C) &= \min_{i,j\in[n^m]|i<j} \frac{d_2(C(i),C(j))^m}{|i-j|}, \text{ and} \\ L_{\text{GL,max}}(C) &= \max_{i,j\in[n^m]|i<j} \frac{d_2(C(i),C(j))^m}{|i-j|}, \text{ for } C \in \mathcal{C}. \end{aligned}$$

They demonstrate that for arbitrary m-dimensional curve C,

$$\begin{aligned} L_{\text{GL,min}}(C) &= O(n^{1-m}), \text{ and} \\ L_{\text{GL,max}}(C) &> (2^m-1)(1-\frac{1}{n})^m. \end{aligned}$$

For the m-dimensional Hilbert curve family $\{H_k^m \mid k = 1, 2, \ldots\}$, they prove that:

$$L_{\text{GL,max}}(H_k^m) \le 2^m(m+3)^{\frac{m}{2}}.$$

For the 2-dimensional Hilbert curve family, they obtain tight bounds:

$$6(1-O(2^{-k})) \le L_{\text{GL,max}}(H_k^2) \le 6\frac{2}{3}.$$

Alber and Niedermeier [Alb97, AN00] generalize $L_{\text{GL,max}}$ to $L_{\text{AN},p}$ by employing the p-normed metric d_p (Manhattan distance ($p = 1$), Euclidean ($p = 2$), and maximum ($p = \infty$)), in place of the Euclidean distance d_2. They improve and extend the above tight bounds for the 2-dimensional Hilbert curve family to:

$$\begin{aligned} & & L_{\text{AN},1}(H_k^2) \le 9\frac{3}{5}, \\ 6(1-O(2^{-k})) &\le & L_{\text{AN},2}(H_k^2) \le 6\frac{1}{2}, \text{ and} \\ 6(1-O(2^{-k})) &\le & L_{\text{AN},\infty}(H_k^2) \le 6\frac{2}{5}. \end{aligned}$$

2.3 Clustering Performance and Related Work

Studies of clustering and inter-clustering performances for space-filling curves are primarily motivated by the applicability of multi-dimensional space-filling indexing methods [FR89, Jag90, ARR+97]. Basically, these methods are based upon an m-dimensional space-filling curve that maps an m-dimensional grid (data, processor-mesh, etc.) space onto a 1-dimensional linear space (external storage structure).

However, if the average number of external fetch/seek operations, which is related to the clustering statistics, can be minimized, the space-filling index structure can therefore support more efficient query processing such as range queries. Asano, Ranjan, Roos, Welzl, and Widmayer [ARR+97] study the optimization of range queries over space-filling index structures by minimizing the number of seek operations but not the number of block accesses. The reason is due to a tradeoff existing between seek time to proper block (cluster) and latency/transfer time for unnecessary blocks (inter-cluster gap), so good bounds on the inter-clustering statistics can translate into good bounds on the average tolerance of unnecessary block transfers.

Jagadish [Jag97] and Moon, Jagadish, Faloutsos, and Saltz [MJFS01] consider the number of disk accesses for a range query, mostly because the number of clusters within a range query is desired to be as minimum as possible. They focus on Hilbert curve and use the mean number of clusters of grid points within a subspace as the measure of the clustering performance. Jagadish [Jag97] derives exact formulas for the mean numbers of clusters over all rectangular 2×2 and 3×3 subgrids of an H_k^2-structural grid space. Moon, Jagadish, Faloutsos, and Saltz [MJFS01] extend the work in [Jag97] to obtain the exact formula for the mean number of clusters over all rectangular $2^q \times 2^q$ subgrids of an H_k^2-structural grid space. Moreover, they also prove that, in a sufficiently large m-dimensional H_k^m-structural grid space, the mean number of clusters over all rectilinear polyhedral queries with surface area $S_{m,k}$ approaches $\frac{1}{2} \cdot \frac{S_{m,k}}{m}$ as k approaches ∞.

In summary, the empirical and analytical studies of clustering performances of various low-dimensional space-filling curves, the Hilbert curve family is relatively superior to other curve families in this respect.

2.4 Other Measures

In addition to the locality preservation and clustering performance, there are some other measures for space-filling curves. Mokbel and Aref [MA01] propose a quantitative measure for the quality of space-filling curves by the number of "irregularities" in each dimension. For an m-dimensional space-filling curve $C : [n^m] \rightarrow [n]^m$, an "irregularity" occurs in dimension z when two points $i, j \in [n^m]$, $i < j$ and $Z(C(i)) > Z(C(j))$ where $Z(C(i))$ (or $Z(C(j))$) denotes the coordinate of point i (or point j) in the dimension z. An irregularity occurs in l-th dimension if the space-filling curve C, as a function, reverses the inequality $i < j$ in the z-coordinate. Figure 2.9 illustrates all possible scenarios for irregularity in the 2-dimensional space. The quantitative measure $I(z, n, m)$ is defined as the number of irregularities in dimension z for a space-filling curve in an m-dimensional space with side length n:

$$I(z,n,m) = \sum_{j=2}^{n^m} \sum_{i=1}^{j-1} f_{i,j} \qquad \text{where } f_{i,j} = \begin{cases} 1 & \text{if } Z(C(i)) > Z(C(j)), \\ 0 & \text{otherwise.} \end{cases}$$

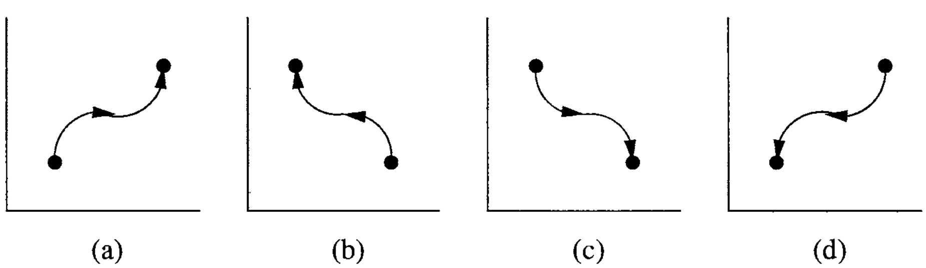


Figure 2.9: Irregularity in the 2-dimensional space: arrows indicate the traversal order of space-filling curves. (a) No irregularity; (b) irregularity in x only; (c) irregularity in y only; (d) irregularity in x and y.

For quantifying partition in parallel computing, Hungershöfer and Wierum [Wie01, HW02] introduce a quality coefficient $\frac{S_C(i,j)}{4\sqrt{V_C(i,j)}}$, for two grid points $i, j \in [n^2]$, where $V_C(i,j)$ is the size (the number of points with indices between i and j), and $S_C(i,j)$

is the surface (perimeter for the 2-dimensional space) that is induced by the two grid points i and j along the space-filling curve C. They use this coefficient to quantify the quality of the induced partition by two grid points i and j in an uniform grid space of size $n \times n$. Then, they extend this coefficient to quantify a space-filling curve by the following two formulations:

$$L_{\mathrm{HW,max}}(C) = \max_{i,j \in [n^2] | i<j} \frac{S_C(i,j)}{4\sqrt{V_C(i,j)}}, \text{ and}$$

$$L_{\mathrm{HW,avg}}(C) = \frac{2}{n^2(n^2-1)} \sum_{j=2}^{n^2} \sum_{i=1}^{j-1} \frac{S_C(i,j)}{4\sqrt{V_C(i,j)}}.$$

By these measures, they obtain tight upper bounds for z-order curve Z_k^2 and Hilbert curve H_k^2. From the results, they conclude that z-order curve performs better than Hilbert curve in this respect.

Based on the types of connecting edges among the grid points in a space-filling curves, Mokbel, Aref, and Kamel [MAK02] define a vector V of five parameters that are the percentages of different types of edges: jump, contiguity, reverse, forward, and still. They suggest to select an appropriate space-filling curves for an application via the knowledge of this vector V.

CHAPTER III

A NEW MEASURE FOR LOCALITY PRESERVATION

On the measures of locality preservation for space-filling curves, our work includes:

1. proposing a new locality measure for space-filling curves by considering the mean absolute index-difference for two grid points at a common distance, and
2. closing the gaps between the upper and lower bounds for the p-normed metric measures $L_{\text{AN},p}(H_k^2)$.

We present the two studies in this and next chapters, respectively.

The new locality measure is similar to $L_{\text{MD},1}$ conditional on a 1-normed distance of δ between points in $[n]^m$:

$$L_\delta(C) = \sum_{i,j\in[n^m] | i<j \text{ and } d_1(C(i),C(j))=\delta} |i-j|.$$

The locality statistics $L_\delta(C)$ cumulates all index-differences of point-pairs (distances traversed in the sequential index space) at a common 1-normed distance δ (local operation in the C-structural grid space). Note that for the three statistics: $L_\delta(C)$, $L_{[\delta]}(C) = \sum_{i=1}^{\delta} L_i(C)$, and the mean absolute index-difference over all point-pairs at a common 1-normed distance δ, the knowledge of one statistics for all δ yields the other two.

The study of locality measures arises in practical contexts. In coding theory, for a bijection C that encodes the integer range $\{0, 1, \ldots, 2^m - 1\}$ into an m-bit binary code $\{0, 1\}^m$, the locality measure $L_\delta(C)$ is proportional to a selection criterion for C that minimizes the mean absolute change resulting from an error of exactly δ bits (see [Har64]). A biological application of locality/spatial measures is their use in modeling a visual nervous system that seeks maximum spatial continuity in mapping visual stimuli onto a 2-dimensional cortex of nerve cells (see [HW79]).

Based on the new measure L_δ, we derive exact formulas for the Hilbert curve family $\{H_k^m \mid k = 1, 2, \ldots\}$ and z-order curve family $\{Z_k^m \mid k = 1, 2, \ldots\}$ for $m = 2$ and arbitrary δ that is an integral power of 2, and $m = 3$ and $\delta = 1$. The exact results yield a constant asymptotic ratio $\lim_{k\to\infty} \frac{L_\delta(H_k^m)}{L_\delta(Z_k^m)} > 1$ for the considered values of m, k, and δ, which suggests the superiority of z-order curve family in low dimensions with respect to L_δ. We verify all the exact formulas (intermediate and final) with computer programs on $m = 2, 3$ and over wide ranges of values for k and δ.

3.1 Locality Measures of 2-Dimensional Space-Filling Curve Families

Hilbert and z-order curve are probably the most popular space-filling curves. One of the salient characteristics of this type of curves is their "self-similarity" (see Chapter II). Let C_k^2 denotes H_k^2 or Z_k^2. Thus, for these two space-filling curves, the self-similar structural property guides us to decompose C_k^2 into four identical C_{k-1}^2-subcurves (via reflection and rotation), which are amalgamated together by an C_1^2-curve (see Figures 2.3, 2.6).

3.1.1 Approach

Our approach to derive the exact formula for $L_\delta(C_k^2)$ is described as follows. The space-filling curve C_k^2 denotes H_k^2 or Z_k^2. The recursive decomposition (in k) of C_k^2 gives that:

$$\begin{aligned} L_\delta(C_k^2) &= \sum_{i,j\in[2^{2k}] \mid i<j \text{ and } d_1(C_k^2(i), C_k^2(j))=\delta} |i-j| \\ &= 4L_\delta(C_{k-1}^2) + \sum_{\alpha,\beta\in\{1,2,3,4\} \mid \alpha<\beta} \Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2)), \end{aligned}$$

where $\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2))$ denotes the cumulative contribution of $|i-j|$ from the two subcurves $Q_\alpha(C_k^2)$ and $Q_\beta(C_k^2)$, that is, for all $i, j \in [2^{2k}]$ such that $i < j$, $d_1(C_k^2(i), C_k^2(j)) = \delta$, and i and j appear in (the index ranges of) $Q_\alpha(C_k^2)$ and $Q_\beta(C_k^2)$, respectively.

We partition the summation $\sum_{\alpha,\beta\in\{1,2,3,4\}|\alpha<\beta}\Delta_\delta(Q_\alpha(C_k^2),Q_\beta(C_k^2))$ according to the two cases: contiguous subcurves ($(\alpha,\beta)\in\{(1,2),(2,3),(3,4),(1,4)\}$ for H_k^2, $(\alpha,\beta)\in\{(1,2),(1,3),(2,4),(3,4)\}$ for Z_k^2) with four similar subcases, and diagonal subcurves ($(\alpha,\beta)\in\{(1,3),(2,4)\}$ for H_k^2, $(\alpha,\beta)\in\{(1,4),(2,3)\}$ for Z_k^2) with two similar subcases. A common thread to the computations in the six subcases is to express $\Delta_\delta(Q_\alpha(C_k^2),Q_\beta(C_k^2))$ as summations of index-cumulations of neighboring geometrical structures in $Q_\alpha(C_k^2)$ and $Q_\beta(C_k^2)$.

3.1.1.1 Geometrical structures (rows, columns, and diagonals). The following denotations illustrate the geometrical structures (rows, columns, and diagonals) involved in the computations.

With respect to the canonical orientation of C_k^2 (shown in Figure 2.6(a) for H_k^2, Figure 2.3(a) for Z_k^2), we cover the 2-dimensional k-order grid with:

1. 2^k rows $(R_{k,1},R_{k,2},\ldots,R_{k,2^k})$, indexed from the bottom,
2. 2^k columns $(C_{k,1},C_{k,2},\ldots,C_{k,2^k})$, indexed from the left,
3. $2^{k+1}-1$ main diagonals $(D_{k,1},D_{k,2},\ldots,D_{k,2^k}=D'_{k,2^k},D'_{k,2^k-1},\ldots,D'_{k,1})$, indexed from the lower-right corner, and
4. $2^{k+1}-1$ auxiliary diagonals $(A_{k,1},A_{k,2},\ldots,A_{k,2^k}=A'_{k,2^k},A'_{k,2^k-1},\ldots,A'_{k,1})$, indexed from the lower-left corner.

For $\alpha\in[2^k]$ and a grid point $p\in[2^k]^2$, we denote:

1. $\hbar_k(v,v')=|(C_k^2)^{-1}(v)-(C_k^2)^{-1}(v')|$, the index-difference between two grid points $v,v'\in[2^k]^2$.
2. $\Delta(X_{k,\alpha},p)=\sum_{v\in X_{k,\alpha}}\hbar_k(v,p)$, where the symbol X denotes R, C, D, D', A, or A' (for example, $\Delta(R_{k,\alpha},p)=\sum_{v\in R_{k,\alpha}}\hbar_k(v,p))$. That is, $\Delta(X_{k,\alpha},p)$ cumulates all index-differences of all grid points in the structure $X_{k,\alpha}$ with respect to p; when $p=\partial_1(C_k^2))$, $\Delta(X_{k,\alpha},p)$ is the index-cumulation of all grid points in $X_{k,\alpha}$.

3. $\mathcal{X}_{k,\alpha} = \sum_{\beta=1}^{\alpha} \sum_{v \in X_{k,\beta}} (\alpha+1-\beta)\hbar_k(v, \partial_1(C_k^2))$, where the symbol-pair $(\mathcal{X}, X)$ denotes $(\mathcal{R}, R)$, $(\mathcal{C}, C)$, $(\mathcal{D}, D)$, $(\mathcal{D}', D')$, $(\mathcal{A}, A)$, or $(\mathcal{A}', A')$ (for example, $\mathcal{R}_{k,\alpha} = \sum_{\beta=1}^{\alpha} \sum_{v \in R_{k,\beta}} (\alpha + 1 - \beta)\hbar_k(v, \partial_1(C_k^2)))$;

   $\mathcal{N}_{k,\alpha} = \sum_{\beta=1}^{\alpha} \sum_{v \in X_{k,\beta}} (\alpha+1-\beta)$, when X denotes D, D', A, or A' (independent of the choice); that is, $\mathcal{N}_{k,\alpha}$ cumulates the number of index (grid point v) references in the summation of $\mathcal{X}_{k,\alpha}$ (that is, $\mathcal{D}_{k,\alpha}$, $\mathcal{D}'_{k,\alpha}$, $\mathcal{A}_{k,\alpha}$, or $\mathcal{A}'_{k,\alpha}$).

   Note that when $\alpha = 0$, all cumulations degenerate to 0, that is, $\mathcal{X}_{k,0} = \mathcal{X}'_{k,0} = \mathcal{N}_{k,0} = 0$.

4. $\overline{\mathcal{X}}_{k,\alpha} = \sum_{\beta=1}^{\alpha} \sum_{v \in X_{k,\beta}} \hbar_k(v, \partial_1(C_k^2)) = \sum_{\beta=1}^{\alpha} \Delta(X_{k,\beta}, \partial_1(C_k^2))$, where the symbol-pair $(\overline{\mathcal{X}}, X)$ denotes $(\overline{\mathcal{R}}, R)$, $(\overline{\mathcal{C}}, C)$, $(\overline{\mathcal{D}}, D)$, $(\overline{\mathcal{D}'}, D')$, $(\overline{\mathcal{A}}, A)$, or $(\overline{\mathcal{A}'}, A')$;

   $\overline{\mathcal{N}}_{k,\alpha} = \sum_{\beta=1}^{\alpha} \sum_{v \in X_{k,\beta}} 1$, when the symbol X denotes D, D', A, or A' (independent of the choice); that is, $\overline{\mathcal{N}}_{k,\alpha}$ cumulates the number of index (grid point v) references in the summation of $\overline{\mathcal{X}}_{k,\alpha}$ (that is, $\overline{\mathcal{D}}_{k,\alpha}$, $\overline{\mathcal{D}'}_{k,\alpha}$, $\overline{\mathcal{A}}_{k,\alpha}$, or $\overline{\mathcal{A}'}_{k,\alpha}$).

   Note that $\mathcal{X}_{k,\alpha} = \mathcal{X}_{k,\alpha-1} + \overline{\mathcal{X}}_{k,\alpha}$ for $\alpha \in \{2, 3, \ldots, 2^k\}$, where the symbol $\mathcal{X}$ denotes $\mathcal{R}$, $\mathcal{C}$, $\mathcal{D}$, $\mathcal{D}'$, $\mathcal{A}$, or $\mathcal{A}'$.

5. $\overline{\mathcal{W}}_k = \sum_{v \in C_k^2} \hbar_k(v, \partial_1(C_k^2))$, which cumulates the indices of all grid points of C_k^2 relative to $\partial_1(C_k^2)$. (For a grid point v, the membership "$v \in C_k^2$" abbreviates "$v \in C_k^2([2^{2k}])$".)

Figure 3.1 illustrates the organization of a 2-dimensional grid into the row, column, main-diagonal, auxiliary-diagonal structures and the coverages for $\mathcal{D}_{k,\alpha}$, $\mathcal{D}'_{k,\alpha}$, $\mathcal{A}_{k,\alpha}$, and $\mathcal{A}'_{k,\alpha}$.

For every grid point p in a canonical C_k^2, we have $\hbar_k(p, \partial_1(C_k^2)) + \hbar_k(p, \partial_2(C_k^2)) = 2^{2k}-1$, and there exists a unique grid point p' such that $\hbar_k(p, \partial_1(C_k^2)) = \hbar_k(p', \partial_2(C_k^2))$. We say that (p, p') is a mirror pair and p' is the mirror point of p. Clearly, the mirror relation is reflexive: (p, p') is a mirror pair if and only if (p', p) is also a mirror pair; moreover,

$$\hbar_k(p, \partial_1(C_k^2)) + \hbar_k(p', \partial_1(C_k^2)) = \hbar_k(p, \partial_2(C_k^2)) + \hbar_k(p', \partial_2(C_k^2)) = 2^{2k} - 1.$$

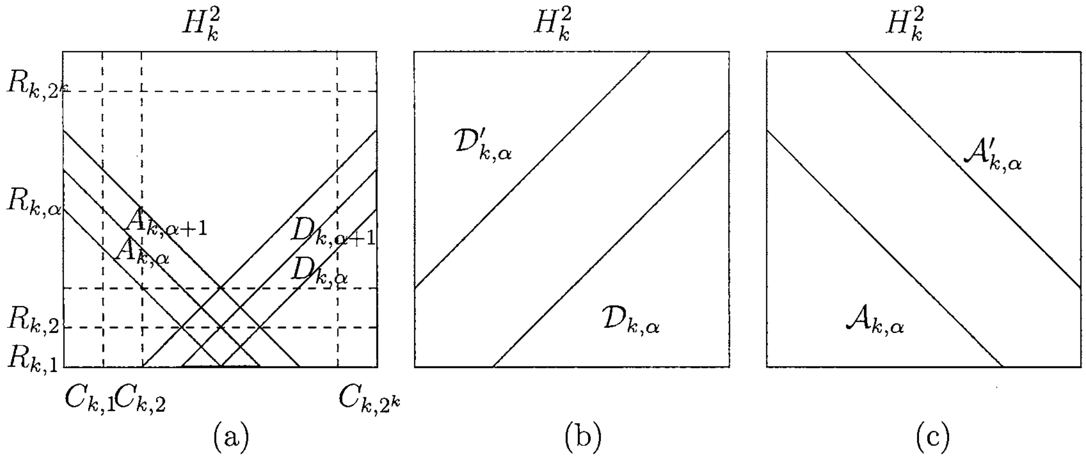


Figure 3.1: (a) Organize a 2-dimensional grid $[2^k]^2$ into the row, column, main-diagonal, and auxiliary-diagonal structures; (b) coverages of $\mathcal{D}_{k,\alpha}$ and $\mathcal{D}'_{k,\alpha}$; (c) coverages of $\mathcal{A}_{k,\alpha}$ and $\mathcal{A}'_{k,\alpha}$.

We extend the notations to identify all C_l^2-subcurves of a structured C_k^2 for all $l \in [k]$ inductively on the order. Let $Q_\alpha(C_k^2)$ denote the α-th C_{k-1}^2-subcurve (along the amalgamating C_1^2-curve) for all $\alpha \in [2^2]$. Then for the α-th C_{l-1}^2-subcurve, $Q_\alpha(C_l^2)$, of C_l^2, where $2 < l \leq k$ and $\alpha \in [2^2]$, let $Q_\beta(Q_\alpha(C_l^2))$ denote the β-th C_{l-2}^2-subcurve of $Q_\alpha(C_l^2)$ for all $\beta \in [2^2]$. We write $Q_\alpha^{q+1}(C_l^2)$ for $Q_\alpha(Q_\alpha^q(C_l^2))$ for all $l \in [k]$ and all positive integers $q < l$. For the two extreme cases: (when $q = 0$) $Q_\alpha^0(C_l^2)$ denotes C_l^2, and (when $q = l$) $Q_\alpha^l(C_l^2)$ identifies the α-th grid point in the C_1^2-subcurve $Q_\alpha^{l-1}(C_l^2)$. Interpreting Q as a "quadrant/subcurve selector", we denote by $Q_{\{\alpha,\beta\}}(C_l^2)$ the α-th or β-th C_{l-1}^2-subcurve of a canonical C_l^2 for all $\alpha, \beta \in [2^2]$. We write $Q_{\{\alpha,\beta\}}^{q+1}(C_l^2)$ for an iteration $Q_{\{\alpha,\beta\}}(Q_{\{\alpha,\beta\}}^q(C_l^2))$ for all $l \in [k]$ and all positive integers $q < l$. Note that the selections in $\{\alpha, \beta\}$ for the iteration are not necessarily identical.

3.1.1.2 The computation of $\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2))$ for $\alpha, \beta \in \{1,2,3,4\}$ with $\alpha < \beta$. Let δ be an arbitrary positive integral power of 2 with $1 \leq \delta < 2^k$. First we derive $\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2))$ for the case of contiguous subcurves. Without loss of generality, we assume that $\alpha < \beta$ and $Q_\alpha(C_k^2)$ appears to the immediate left-side of $Q_\beta(C_k^2)$ for our discussion (see Figures 3.2 and 3.3).

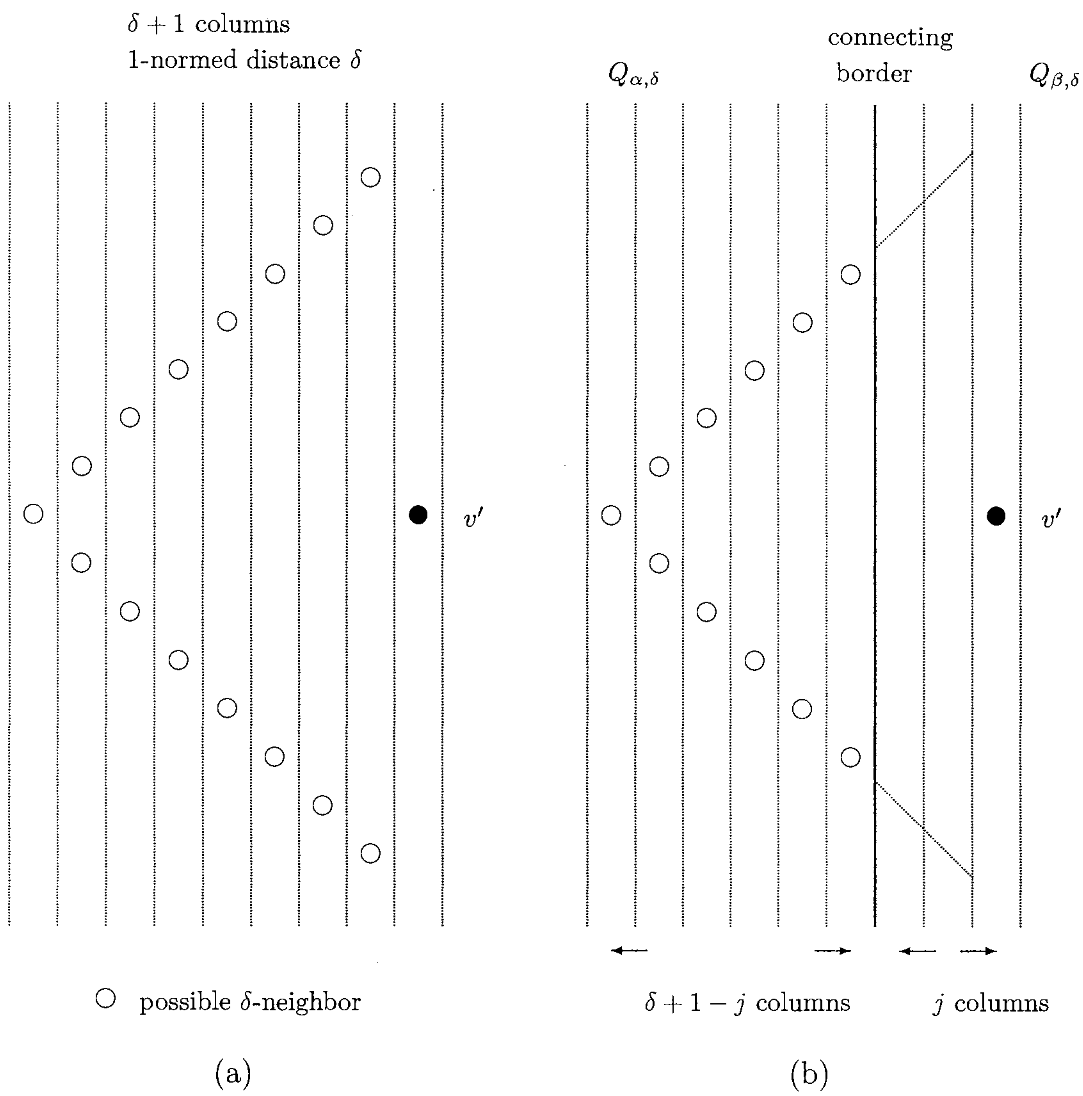


Figure 3.2: (a) $N_\delta(v')$ in the absence of the connecting border (with cardinality of $2\delta - 1$); (b) $N_\delta(v')$ in the presence of the connecting border (with cardinality of $2(\delta + 1 - j) - 1$).

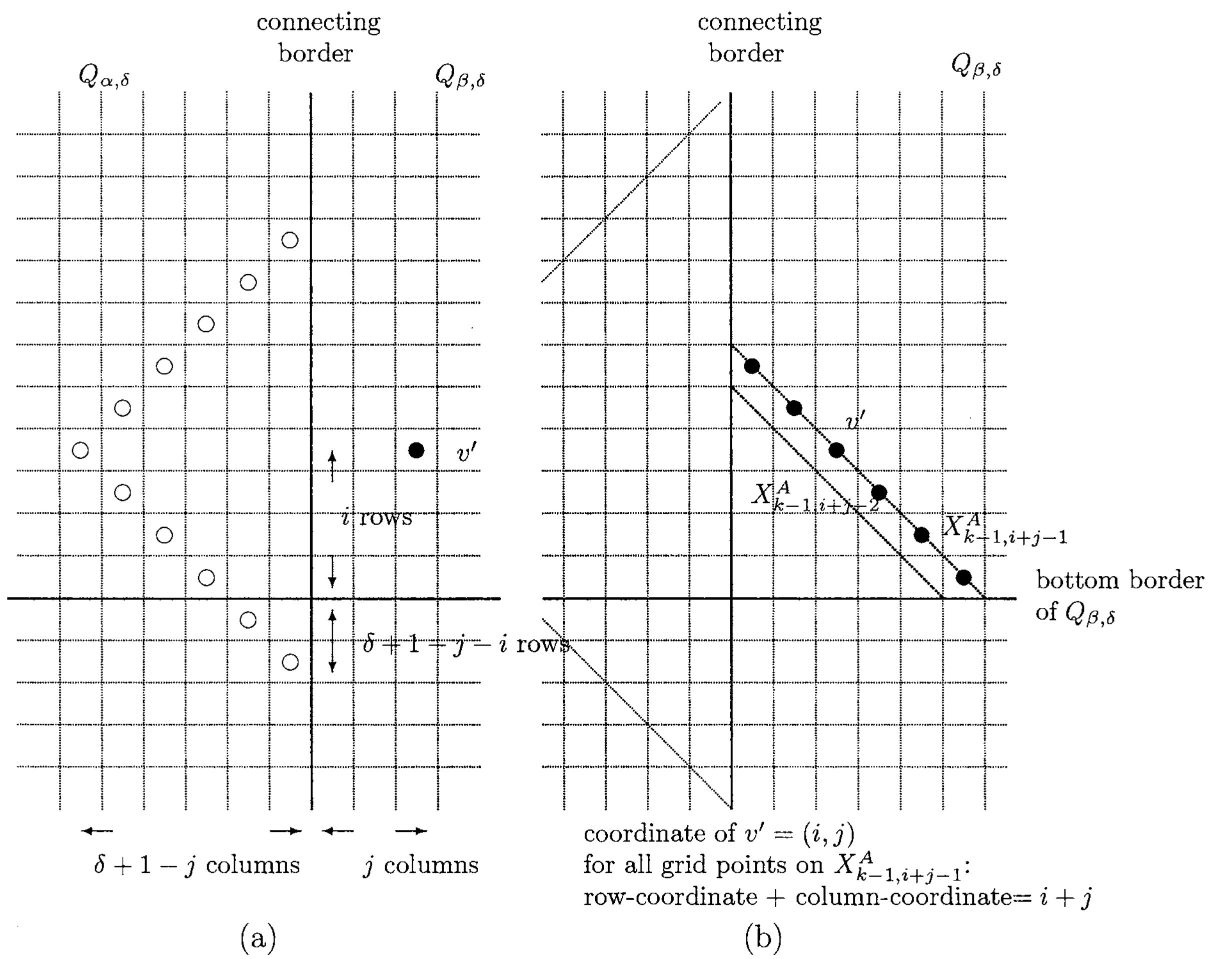


Figure 3.3: (a) $N_\delta(v')$ in the presence of both connecting and bottom borders (with cardinality $(2(\delta+1-j)-1)-(\delta+1-j-i) = \delta-j+i$, provided that $\delta+1-j-i \geq 0$); (b) following the auxiliary diagonals $X^A_{k-1,1}, X^A_{k-1,2}, \ldots, X^A_{k-1,\delta}$ to enumerate all grid points v' with row and column coordinates (i,j) in the bottom boundary region of $Q_{\beta,\delta}$ such that $\delta+1-j-i \geq 0$.

Each of the two quadrants $Q_\alpha(C_k^2)$ and $Q_\beta(C_k^2)$ has its own bottom and top borders to delimit its 2^{k-1} rows, and the border between the two quadrants is called the connecting border. Note that

$$\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2)) = \sum_{(v,v')\in Q_\alpha(C_k^2)\times Q_\beta(C_k^2)|d_1(v,v')=\delta} \hbar_k(v,v');$$

so for a pair $(v,v') \in Q_\alpha(C_k^2) \times Q_\beta(C_k^2)$ with contribution in $\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2))$, the grid points v and v' must lie within the first δ columns in $Q_\alpha(C_k^2)$ and $Q_\beta(C_k^2)$, respectively, from the connecting border. We index the rows and columns of a quadrant locally around the connecting border as: rows $r_1^\gamma, r_2^\gamma, \ldots, r_{2^{k-1}}^\gamma$ (indexed from the bottom in each quadrant), columns $c_1^\gamma, c_2^\gamma, \ldots, c_\delta^\gamma$ (indexed from the connecting border in each quadrant), where γ denotes α or β (indicating the residing quadrant $Q_\alpha(C_k^2)$ or $Q_\beta(C_k^2)$ for the rows and columns. Denote by $Q_{\alpha,\delta}$ ($Q_{\beta,\delta}$) the $2^{k-1} \times \delta$ subgrid of $Q_\alpha(C_k^2)$ (respectively, $Q_\beta(C_k^2)$) consisting of these 2^{k-1} rows and δ columns.

The key idea in deriving $\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2))$ is to express $\hbar_k(v,v')$, where $(v,v') \in Q_{\alpha,\delta} \times Q_{\beta,\delta}$, as

$$\hbar_k(v, \partial_2(Q_\alpha(C_k^2))) + \hbar_k(\partial_2(Q_\alpha(C_k^2)), \partial_1(Q_\beta(C_k^2))) + \hbar_k(v', \partial_1(Q_\beta(C_k^2))),$$

and write:

$$\begin{aligned}
&\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2)) \\
= &\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} (\hbar_k(v, \partial_2(Q_\alpha(C_k^2))) + \hbar_k(\partial_2(Q_\alpha(C_k^2)), \partial_1(Q_\beta(C_k^2))) \\
&\qquad + \hbar_k(v', \partial_1(Q_\beta(C_k^2)))) \\
= &\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(v, \partial_2(Q_\alpha(C_k^2))) \\
&+ \sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(\partial_2(Q_\alpha(C_k^2)), \partial_1(Q_\beta(C_k^2))) \\
&+ \sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(v', \partial_1(Q_\beta(C_k^2)))
\end{aligned} \tag{3.1}$$

Each of these three cumulations of index-differences requires some knowledge of the combinatorial structure of all the pairs $(v,v') \in Q_{\alpha,\delta} \times Q_{\beta,\delta}$ with $d_1(v,v') = \delta$. We say that (v, v') is a δ-pair and v is a δ-neighbor of v' (and vice versa). Denote by $N_\delta(v')$ the δ-neighborhood in $Q_{\alpha,\delta}$ of $v' \in Q_{\beta,\delta}$.

By symmetry, it suffices to consider for an arbitrary grid point $v' \in Q_{\beta,\delta}$, and determine $|N_\delta(v')|$ — which depends on the location of v' within $Q_{\beta,\delta}$. We partition $Q_{\beta,\delta}$ into three regions based on the row and column indices as follows:

1. The middle boundary region consisting of the middle rows r_i^β for $i = \delta+1, \delta+2, \ldots, 2^{k-1}-\delta$, that is, $(\cup_{i=\delta+1}^{2^{k-1}-\delta} r_i^\beta) \cap (\cup_{j=1}^{\delta} c_j^\beta)$,
2. The bottom boundary region consisting of the bottom rows r_i^β for $i = 1, 2, \ldots, \delta$, that is, $(\cup_{i=1}^{\delta} r_i^\beta) \cap (\cup_{j=1}^{\delta} c_j^\beta)$, and
3. The top boundary region consisting of the top rows r_i^β for $i = 2^{k-1}-\delta+1, 2^{k-1}-\delta+2, \ldots, 2^{k-1}$, that is $(\cup_{i=2^{k-1}-\delta+1}^{2^{k-1}} r_i^\beta) \cap (\cup_{j=1}^{\delta} c_j^\beta)$.

For an arbitrary grid point $v' \in r_i^\beta \cap c_j^\beta$ in the middle boundary region of $Q_{\beta,\delta}$, where $i \in \{\delta+1, \delta+2, \ldots, 2^{k-1}-\delta\}$ and $j \in [\delta]$, the columns of $Q_{\alpha,\delta}$ in which all δ-neighbors v of v' reside are $c_1^\beta, c_2^\beta, \ldots, c_{\delta+1-j}^\beta$ of $Q_{\alpha,\delta}$ with the distribution: one δ-neighbor in $c_{\delta+1-j}^\beta$ and two in each of the remaining columns (see Figure 3.2(a) and (b)). Therefore, $N_\delta(v') = 2(\delta+1-j)-1$.

For an arbitrary grid point $v' \in r_i^\beta \cap c_j^\beta$ in the bottom boundary region of $Q_{\beta,\delta}$, where $i \in [\delta]$ and $j \in [\delta]$, in the presence of both connecting and bottom borders, $|N_\delta(v')| = (2(\delta+1-j)-1)-(\delta+1-j-i) = \delta-j+i$, provided that $\delta+1-j-i \geq 0$. The second term $(\delta+1-j)-i$ gives the number of grid points v (δ-neighbors of v' in the absence of the bottom border) eliminated by the bottom border (see Figure 3.3(a)). Therefore,

$$|N_\delta(v')| = \begin{cases} \delta - j + i & \text{if } \delta+1-j-i \geq 0 \\ 2(\delta+1-j)-1 & \text{otherwise.} \end{cases}$$

Note that the condition $\delta+1-j-i \geq 0$ corresponds to the lower-left half of the $C^2_{\log\delta}$-subcurve at the lower-left corner of $Q_{\beta,\delta}$. With denotations for main and auxiliary diagonals similar to those in Section 3.1.1, we cover the lower-left half of the $C^2_{\log\delta}$-subcurve with δ auxiliary diagonals $X^A_{k-1,1}, X^A_{k-1,2}, \ldots, X^A_{k-1,\delta}$, indexed from the lower-left corner of the $C^2_{\log\delta}$-subcurve (see Figure 3.3(b)). Observe that for a grid point v' in $Q_{\beta,\delta}$ with row and column coordinates (i,j), $v' \in X^A_{k-1,\eta}$ for some $\eta \in [\delta]$ if and only if $i+j = \eta+1$; in this case, $|N_\delta(v')| = (2(\delta+1-j)-1)-(\delta-\eta)$.

For an arbitrary grid point $v' \in r^\beta_i \cap c^\beta_j$ in the top boundary region of $Q_{\beta,\delta}$, where $i \in \{2^{k-1}-\delta+1, 2^{k-1}-\delta+2, \ldots, 2^{k-1}\}$ and $j \in [\delta]$, that is, $(\cup^{2^{k-1}}_{i=2^{k-1}-\delta+1} r^\beta_i) \cap (\cup^\delta_{j=1} c^\beta_j)$, we proceed as in the case of bottom boundary region: cover the upper-left half of the $C^2_{\log\delta}$-subcurve at the upper-left cover of $Q_{\beta,\delta}$ with δ main diagonals $X^{D'}_{k-1,1}, X^{D'}_{k-1,2}, \ldots, X^{D'}_{k-1,\delta}$, indexed from the upper-left corner of the $C^2_{\log\delta}$-subcurve, and obtain that

$$|N_\delta(v')| = \begin{cases} (2(\delta+1-j)-1)-(\delta-\eta) & \text{if } v' \in X^{D'}_{k-1,\eta} \text{ for some } \eta \in [\delta] \\ 2(\delta+1-j)-1 & \text{otherwise.} \end{cases}$$

Now, the analysis above allows the three cumulations of index-differences in $\Delta_\delta(Q_\alpha(C^2_k), Q_\beta(C^2_k))$ to be expanded as follows. In order to compute $\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(v', \partial_1(Q_\beta(C^2_k)))$, we consider for all possible $v' \in Q_{\beta,\delta}$, which is partitioned into the middle, bottom, and top boundary regions, and determine the number of index (grid point v') references in the summation — which is $|N_\delta(v')|$. Denote by $v'_{i,j}$ a grid point $v' \in Q_{\beta,\delta}$ with row and column coordinates (i,j). Then,

$$\begin{aligned}
&\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(v', \partial_1(Q_\beta(C^2_k))) \\
= &\sum_{j=1}^{\delta}\sum_{i=\delta+1}^{2^{k-1}-\delta} (2(\delta+1-j)-1)\hbar_k(v'_{i,j}, \partial_1(Q_\beta(C^2_k))) \quad \text{(middle boundary region)} \\
&+\sum_{j=1}^{\delta}\sum_{i=1}^{\delta}(2(\delta+1-j)-1)\hbar_k(v'_{i,j}, \partial_1(Q_\beta(C^2_k))) \\
&-\sum_{\eta=1}^{\delta}\sum_{u\in X^A_{k-1,\eta}} (\delta-\eta)\hbar_k(u, \partial_1(Q_\beta(C^2_k))) \quad \text{(bottom boundary region)} \\
&+\sum_{j=1}^{\delta}\sum_{i=2^{k-1}-\delta+1}^{2^{k-1}} (2(\delta+1-j)-1)\hbar_k(v'_{i,j}, \partial_1(Q_\beta(C^2_k)))
\end{aligned}$$

$$-\sum_{\eta=1}^{\delta}\sum_{u\in X^{D'}_{k-1,\eta}}(\delta-\eta)\hbar_k(u,\partial_1(Q_\beta(C^2_k)))\qquad\text{(top boundary region)}$$

$$= \sum_{j=1}^{\delta}\sum_{i=1}^{2^{k-1}}(2(\delta+1-j)-1)\hbar_k(v'_{i,j},\partial_1(Q_\beta(C^2_k)))$$

$$-\sum_{\eta=1}^{\delta}\sum_{u\in X^{D'}_{k-1,\eta}}(\delta-\eta)\hbar_k(u,\partial_1(Q_\beta(C^2_k)))-\sum_{\eta=1}^{\delta}\sum_{u\in X^{A}_{k-1,\eta}}(\delta-\eta)\hbar_k(u,\partial_1(Q_\beta(C^2_k)))$$

$$= \sum_{j=1}^{\delta}\sum_{u\in c^\beta_j}(2(\delta+1-j)-1)\hbar_k(u,\partial_1(Q_\beta(C^2_k)))$$

$$-\sum_{\eta=1}^{\delta}\sum_{u\in X^{D'}_{k-1,\eta}}(\delta-\eta)\hbar_k(u,\partial_1(Q_\beta(C^2_k)))$$

$$-\sum_{\eta=1}^{\delta}\sum_{u\in X^{A}_{k-1,\eta}}(\delta-\eta)\hbar_k(u,\partial_1(Q_\beta(C^2_k))). \qquad (3.2)$$

Similarly, the symmetry between $Q_{\alpha,\delta}$ and $Q_{\beta,\delta}$ yields an expansion for $\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta}\hbar_k(v,\partial_2(Q_\alpha(C^2_k)))$. Denote by $Y^D_{k-1,1},Y^D_{k-1,2},\ldots,Y^D_{k-1,\delta}$ the main diagonals covering the lower-right half of the $C^2_{\log\delta}$-subcurve at the lower-right corner of $Q_\alpha(C^2_k)$, indexed from the lower-right corner of the $C^2_{\log\delta}$-subcurve, and by $Y^{A'}_{k-1,1},Y^{A'}_{k-1,2},\ldots,Y^{A'}_{k-1,\delta}$ the auxiliary diagonals covering the upper-right half of the $C^2_{\log\delta}$-subcurve at the upper-right corner of $Q_\alpha(C^2_k)$, indexed from the upper-right corner of the $C^2_{\log\delta}$-subcurve. Then,

$$\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta}\hbar_k(v,\partial_2(Q_\alpha(C^2_k)))$$

$$= \sum_{j=1}^{\delta}\sum_{u\in c^\alpha_j}(2(\delta+1-j)-1)\hbar_k(u,\partial_2(Q_\alpha(C^2_k)))$$

$$-\sum_{\eta=1}^{\delta}\sum_{u\in Y^{D}_{k-1,\eta}}(\delta-\eta)\hbar_k(u,\partial_2(Q_\alpha(C^2_k)))$$

$$-\sum_{\eta=1}^{\delta}\sum_{u\in Y^{A'}_{k-1,\eta}}(\delta-\eta)\hbar_k(u,\partial_2(Q_\alpha(C^2_k))). \qquad (3.3)$$

For $\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta}\hbar_k(\partial_2(Q_\alpha(C^2_k)),\partial_1(Q_\beta(C^2_k)))$, we note that it is independent of (v, v') in the summation. It suffices to compute $\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} 1$, similar to the previous two expansions:

$$\begin{aligned}
&\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} 1 \\
=\ & \sum_{j=1}^{\delta}\sum_{u\in c_j^{\beta}}(2(\delta+1-j)-1) - \sum_{\eta=1}^{\delta}\sum_{u\in X_{k-1,\eta}^{D'}}(\delta-\eta) - \sum_{\eta=1}^{\delta}\sum_{u\in X_{k-1,\eta}^{A}}(\delta-\eta) \\
=\ & 2^{k-1}\sum_{j=1}^{\delta}(2\delta+1-2j) - \mathcal{N}_{k-1,\delta-1} - \mathcal{N}_{k-1,\delta-1} \\
=\ & 2^{k-1}\delta + 2^{k-1}(\delta^2-\delta) - 2\mathcal{N}_{k-1,\delta-1} = 2^{k-1}\delta^2 - 2\mathcal{N}_{k-1,\delta-1}.
\end{aligned}$$

This gives that:

$$\begin{aligned}
&\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(\partial_2(Q_\alpha(C_k^2)), \partial_1(Q_\beta(C_k^2))) \\
=\ & (2^{k-1}\delta^2 - 2\mathcal{N}_{k-1,\delta-1})\hbar_k(\partial_2(Q_\alpha(C_k^2)), \partial_1(Q_\beta(C_k^2))). \qquad (3.4)
\end{aligned}$$

For the case of diagonal subcurves, we consider the computation of $\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2))$ for the main-diagonal subcurves $Q_\alpha(C_k^2)$ and $Q_\beta(C_k^2)$, and let $Q_\alpha(C_k^2)$ be the lower-left subcurve/quadrant and $Q_\beta(C_k^2)$ the upper-right one. (The derivation for auxiliary-diagonal subcurves is similar.)

For a δ-pair $(v, v') \in Q_\alpha(C_k^2) \times Q_\beta(C_k^2)$, v and v' reside in the neighboring $(\delta - 1) \times (\delta - 1)$ triangular corners of $Q_\alpha(C_k^2)$ and $Q_\beta(C_k^2)$, respectively; thus the number of δ-pairs in $Q_\alpha(C_k^2) \times Q_\beta(C_k^2)$ is $\mathcal{N}_{k-1,\delta-1}$. Figure 3.3(a) shows $N_\delta(v')$ in the presence of both connecting and bottom borders for the case of contiguous subcurves, it also reveals the neighborhood of v' for this case — the eliminated "$\delta + 1 - j - i$" grid points.

We now expand $\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2))$ for main-diagonal subcurves $Q_\alpha(C_k^2)$ and $Q_\beta(C_k^2)$ as follows:

$$\begin{aligned}
&\Delta_\delta(Q_\alpha(C_k^2), Q_\beta(C_k^2)) \\
=\ & \sum_{\eta=1}^{\delta-1}\sum_{u\in Y_{k-1,\eta}^{A'}}(\delta-\eta)\hbar_k(u, \partial_2(Q_1(C_k^2)))
\end{aligned}$$

$$+\sum_{\eta=1}^{\delta-1}\sum_{u\in X^A_{k-1,\eta}}(\delta-\eta)\hbar_k(u,\partial_1(Q_3(C_k^2)))$$
$$+\mathcal{N}_{k-1,\delta-1}\hbar_k(\partial_2(Q_1(C_k^2)),\partial_1(Q_3(C_k^2))). \quad (3.5)$$

The detailed derivations for Hilbert and z-order curve families are shown in the following two sections.

3.1.2 Derivation of 2-Dimensional Hilbert Curve Family

For a 2-dimensional grid, the "orientation" of H_k^2 uniquely determines that of $Q_\alpha(H_k^2)$ for $\alpha = 1, 2, 3, 4$, and thus only one H_k^2 exists modulo symmetry (whereas there are 1536 structurally different 3-dimensional Hilbert curves [AN00]).

In order to compute the closed-form solution for $L_\delta(H_k^2)$, we develop a suite of lemmas (Lemmas 3.4, 3.5, and 3.6) to compute $\Delta_\delta(Q_\alpha(H_k^2), Q_\beta(H_k^2))$ for all $\alpha, \beta \in \{1, 2, 3, 4\}$ with $\alpha < \beta$, in which we establish a recurrence system (in k — when $2^k > \delta$) of summations with a basis system of summations computed in Lemmas 3.1, 3.2, and 3.3. Note, the notations introduced in Section 3.1.1 are applied to H_k^2 in this section.

The following three lemmas study the cumulation of indices of grid points in the row, column, diagonal, and auxiliary-diagonal structures of H_k^2.

Lemma 3.1 *The index-cumulation of a row structure of H_k^2 is independent of its row-number: for all $\alpha \in [2^k]$,*

$$\Delta(R_{k,\alpha}, \partial_1(H_k^2)) = \Delta(R_{k,\alpha}, \partial_2(H_k^2)) = \frac{1}{2}\cdot 2^{3k} - \frac{1}{2}\cdot 2^k \qquad \textit{(independent of } \alpha.\textit{)}$$

Proof. A canonical H_k^2 is left-right reflexive. For a grid point $v \in R_{k,\alpha} \cap C_{k,\beta}$, where $\alpha, \beta \in [2^k]$, its mirror point $v' \in R_{k,\alpha} \cap C_{k,2^k+1-\beta}$, and the mirror pair (v, v') satisfies that:

$$\hbar_k(v,\partial_1(H_k^2)) + \hbar_k(v',\partial_1(H_k^2)) = \hbar_k(v,\partial_2(H_k^2)) + \hbar_k(v',\partial_2(H_k^2)) = 2^{2k}-1.$$

For every $\alpha \in [2^k]$, there are 2^{k-1} mirror pairs in the row $R_{k,\alpha}$. Thus,

$$\Delta(R_{k,\alpha},\partial_1(H_k^2)) = \sum_{v\in R_{k,\alpha}} \hbar_k(v,\partial_1(H_k^2))$$

$$
\begin{aligned}
(= & \sum_{v' \in R_{k,\alpha}} \hbar_k(v', \partial_2(H_k^2)) = \Delta(R_{k,\alpha}, \partial_2(H_k^2)) \qquad (v' \text{ is the mirror point of } v)) \\
= & \sum_{\text{all mirror pairs } (v,v') \text{ in } R_{k,\alpha}} (\hbar_k(v, \partial_1(H_k^2)) + \hbar_k(v', \partial_1(H_k^2))) \\
= & \; 2^{k-1}(2^{2k} - 1) = \frac{1}{2} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k.
\end{aligned}
$$

∎

Lemma 3.2 *The binary representation of the column-number of a column structure of H_k^2 helps compute its index-cumulation as follows:*

1. *For all $\alpha \in [2^{k-1}]$, a recurrence for $\Delta(C_{k,\alpha}, \partial_1(H_k^2))$ is:*

$$\Delta(C_{k,\alpha}, \partial_1(H_k^2)) = \Delta(R_{k-1,\alpha}, \partial_1(H_{k-1}^2)) + \Delta(C_{k-1,\alpha}, \partial_1(H_{k-1}^2)) + \frac{1}{2^3} \cdot 2^{3k}.$$

For $\alpha = 1$, a closed-form solution for $\Delta(C_{k,1}, \partial_1(H_k^2))$ from the recurrence above (in k) is:

$$\frac{3}{2 \cdot 7} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k + \frac{2}{7}.$$

For those α satisfying $2^{q-1} < \alpha \leq 2^q$ for some integer $q \in [k-1]$ (that is, $q = \lceil \log \alpha \rceil$), the recurrence above (in k) yields a recurrence:

$$\Delta(C_{k,\alpha}, \partial_1(H_k^2)) = \Delta(C_{q,\alpha}, \partial_1(H_q^2)) + \sum_{\eta=q+1}^{k} (\frac{3}{2^4} \cdot 2^{3\eta} - \frac{1}{2^2} \cdot 2^{\eta}),$$

where the summation is $\frac{3}{2 \cdot 7}(2^{3k} - 2^{3q}) - \frac{1}{2}(2^k - 2^q) = \frac{3}{2 \cdot 7} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k - \frac{3}{2 \cdot 7} \cdot 2^{3\lceil \log \alpha \rceil} + \frac{1}{2} \cdot 2^{\lceil \log \alpha \rceil}$.

2. *For all $\alpha, \beta \in [2^k]$ such that $\alpha < \beta$ and the binary representations of $\alpha - 1$ and $\beta - 1$ differ only at the i-th low-order bit, where $i \in \{0, 1, \ldots, k-1\}$ (that is, $(\alpha - 1) \oplus (\beta - 1) = 2^i$, where $\oplus$ denotes the binary exclusive-or operator),*

$$\Delta(C_{k,\beta}, \partial_1(H_k^2)) = \Delta(C_{k,\alpha}, \partial_1(H_k^2)) + 2^2 \cdot 2^{3i}.$$

Proof. Consider a canonical H_k^2.

Part 1: We construct the general recurrence (in k) for $\Delta(C_{k,\alpha}, \partial_1(H_k^2))$ for arbitrary $\alpha \in [2^{k-1}]$ as follows. The column $C_{k,\alpha}$ is in the left-half of the canonical H_k^2, and can be decomposed into the α-th column of $Q_1(H_k^2)$ ($(-\frac{\pi}{2})$-rotating and then left-right reflecting into a canonical H_{k-1}^2) and the α-th column of $Q_2(H_k^2)$, that is, the α-th row of a canonical H_{k-1}^2 and the α-th column of a canonical H_{k-1}^2, respectively. By noting that for all grid points $v \in Q_2(H_k^2)$,

$$\begin{aligned} & \hbar_k(v, \partial_1(H_k^2)) = \hbar_k(v, \partial_1(Q_2(H_k^2))) + \hbar_k(\partial_1(Q_2(H_k^2)), \partial_1(H_k^2)) \\ = \;& \hbar_k(v, \partial_1(Q_2(H_k^2))) + 2^{2(k-1)}, \end{aligned}$$

and translating the index-cumulation of $C_{k,\alpha}$ (in H_k^2) in the two H_{k-1}^2-subcurves, we have:

$$\begin{aligned} & \Delta(C_{k,\alpha}, \partial_1(H_k^2)) = \sum_{v \in C_{k,\alpha}} \hbar_k(v, \partial_1(H_k^2)) \\ = \;& (\Delta(C_{k,\alpha} \cap Q_1(H_k^2), \partial_1(Q_1(H_k^2))) + 2^{k-1}\hbar_k(\partial_1(Q_1(H_k^2)), \partial_1(H_k^2))) \\ & +(\Delta(C_{k,\alpha} \cap Q_2(H_k^2), \partial_1(Q_2(H_k^2))) + 2^{k-1}\hbar_k(\partial_1(Q_2(H_k^2)), \partial_1(H_k^2))) \\ = \;& (\Delta(R_{k-1,\alpha}, \partial_1(H_{k-1}^2)) + 2^{k-1} \cdot 0) \quad \text{(after } (+\frac{\pi}{2})\text{-rotating and} \\ & \quad \text{then left-right reflecting } Q_1(H_k^2) \text{ into a canonical } H_k^2) \\ & +(\Delta(C_{k-1,\alpha}, \partial_1(H_{k-1}^2)) + 2^{k-1} \cdot 2^{2(k-1)}) \\ = \;& \Delta(R_{k-1,\alpha}, \partial_1(H_{k-1}^2)) + \Delta(C_{k-1,\alpha}, \partial_1(H_{k-1}^2)) + \frac{1}{2^3} \cdot 2^{3k} \\ (= \;& \Delta(C_{k-1,\alpha}, \partial_1(H_{k-1}^2)) + \frac{3}{2^4} \cdot 2^{3k} - \frac{1}{2^2} \cdot 2^k \quad \text{(by Lemma 3.1)}). \end{aligned}$$

Now we iterate the general recurrence (in k) from order k to order $q+1$ (where $q = \lceil \log \alpha \rceil$), which yields the recurrence (in k) for $\Delta(C_{k,\alpha}, \partial_1(H_k^2))$:

$$\begin{aligned} & \Delta(C_{k,\alpha}, \partial_1(H_k^2)) \\ = \;& \Delta(C_{k-1,\alpha}, \partial_1(H_{k-1}^2)) + \frac{3}{2^4} \cdot 2^{3k} - \frac{1}{2^2} \cdot 2^k \\ = \;& \Delta(C_{q,\alpha}, \partial_1(H_q^2)) + \sum_{\eta=q+1}^{k} (\frac{3}{2^4} \cdot 2^{3\eta} - \frac{1}{2^2} \cdot 2^{\eta}) \\ = \;& \Delta(C_{q,\alpha}, \partial_1(H_q^2)) + \frac{3}{2 \cdot 7}(2^{3k} - 2^{3q}) - \frac{1}{2}(2^k - 2^q) \\ = \;& \Delta(C_{q,\alpha}, \partial_1(H_q^2)) + \frac{3}{2 \cdot 7} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k - \frac{3}{2 \cdot 7} \cdot 2^{3\lceil \log \alpha \rceil} + \frac{1}{2} \cdot 2^{\lceil \log \alpha \rceil}. \end{aligned}$$

When $\alpha = 1$, the general recurrence yields a recurrence (in k):

$$\Delta(C_{k,1}, \partial_1(H_k^2)) = \Delta(C_{1,1}, \partial_1(H_1^2)) + \sum_{\eta=2}^{k} (\frac{3}{2^4} \cdot 2^{3\eta} - \frac{1}{2^2} \cdot 2^{\eta})$$
$$= \frac{3}{2 \cdot 7} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k + \frac{2}{7}.$$

Part 2: Let $\alpha, \beta \in [2^k]$ such that $\alpha < \beta$ and $(\alpha - 1) \oplus (\beta - 1) = 2^i$, where $i \in \{0, 1, \ldots, k-1\}$. Consider the two columns $C_{k,\alpha}$ and $C_{k,\beta}$ of the canonical H_k^2, which are 2^i columns apart. Denote the suffixes of the lower $2^k - 2^{i+1}$ grid points of $C_{k,\alpha}$ and $C_{k,\beta}$ by C_α and C_β, respectively (both are empty when and only when $i = k-1$). From the assumption that $(\alpha-1) \oplus (\beta-1) = 2^i$ and the successive subcurve/quadrant orientations of the canonical H_k^2, we observe the following:

1. The two suffixes C_α and C_β are the longest possible suffixes of $C_{k,\alpha}$ and $C_{k,\beta}$, respectively, such that both are in the same successive subcurves/quadrants enumerated as in the sequence $Q_{\{1,4\}}(Q^0_{\{2,3\}}(H_k^2)), Q_{\{1,4\}}(Q^1_{\{2,3\}}(H_k^2)), \ldots,$ $Q_{\{1,4\}}(Q^{k-(i+1)-1}_{\{2,3\}}(H_k^2))$. (When both C_α and C_β are empty, the sequence is void.)

2. For each $j \in \{0, 1, \ldots, k-i-2\}$, the subcurve $Q_{\{1,4\}}(Q^j_{\{2,3\}}(H_k^2))$ is an H^2_{k-j-1}-subcurve (of the canonical H_k^2), the two segments $C_\alpha \cap Q_{\{1,4\}}(Q^j_{\{2,3\}}(H_k^2))$ and $C_\beta \cap Q_{\{1,4\}}(Q^j_{\{2,3\}}(H_k^2))$ can be viewed as two rows in a canonical H^2_{k-j-1}, as illustrated in Figure 3.4(a). By Lemma 3.1, the index-cumulation (of all grid points) of the two segments $C_\alpha \cap Q_{\{1,4\}}(Q^j_{\{2,3\}}(H_k^2))$ and $C_\beta \cap Q_{\{1,4\}}(Q^j_{\{2,3\}}(H_k^2))$ are equal.

3. The difference in the index-cumulation between the columns $C_{k,\alpha}$ and $C_{k,\beta}$ in the canonical H_k^2 is equal to that of the corresponding columns, C'_α and C'_β, respectively, in the canonically oriented H^2_{i+1}-subcurve $Q_{\{2,3\}}(Q^{k-i-2}_{\{2,3\}}(H_k^2))$, which is denoted by $\mathcal{Q}$. When zooming in on $\mathcal{Q}$ (see Figure 3.4(b)), we partition the difference in the index-cumulation of C'_α and C'_β into the following two parts:

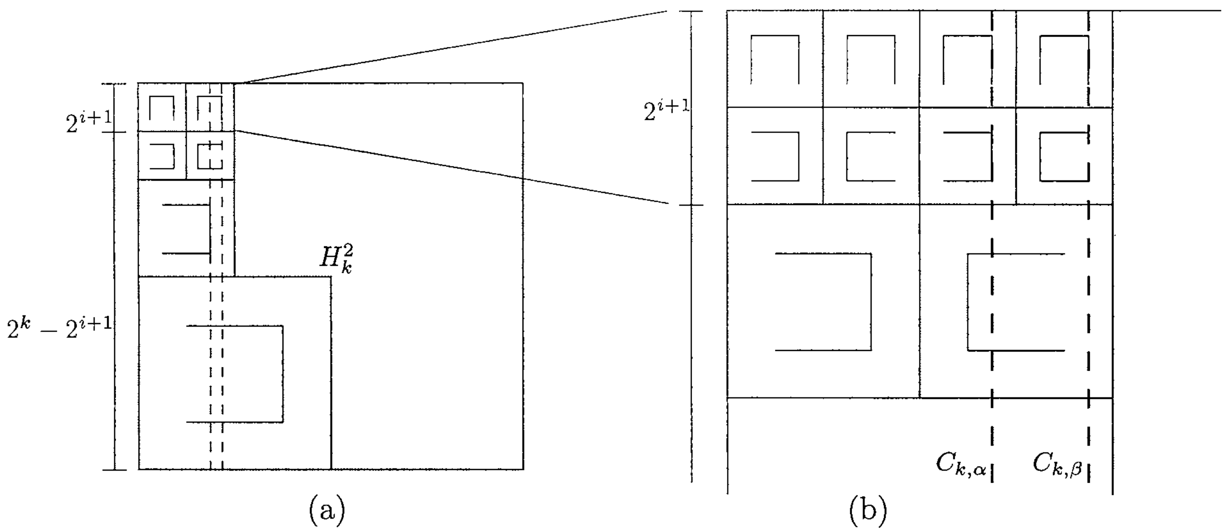


Figure 3.4: (a) Two columns $C_{k,\alpha}$ and $C_{k,\beta}$ in a canonical H_k^2 with$(\alpha-1)\oplus(\beta-1) = 2^i$, where $i \in \{0, 1, \ldots, k-1\}$; (b) zoom-in illustration of the prefixes of $C_{k,\alpha}$ and $C_{k,\beta}$ in the H_{i+1}^2-subcurve.

(a) Between the two segments $C'_\alpha \cap Q_1(\mathcal{Q})$ and $C'_\beta \cap Q_4(\mathcal{Q})$ in the 2^i-subcurves $Q_1(\mathcal{Q})$ and $Q_4(\mathcal{Q})$, respectively:

These two segments correspond to two rows in a canonical H_i^2, in which they have equal index-cumulation with respect to $\partial_1(H_i^2)$ by Lemma 3.1. Therefore the difference in the index-cumulation of these two segments in $\mathcal{Q}$ is effectively $\sum_{v \in C'_\beta \cap Q_4(\mathcal{Q})} \hbar_{i+1}(\partial_1(Q_4(\mathcal{Q})), \partial_1(\mathcal{Q})) = 2^i(3 \cdot 2^{2i})$, since there are 2^i grid points in $C_\beta \cap Q_4(\mathcal{Q})$ and $\hbar_{i+1}(\partial_1(Q_4(\mathcal{Q})), \partial_1(\mathcal{Q})) = 3 \cdot 2^i$.

(b) Between the two segments $C'_\alpha \cap Q_2(\mathcal{Q})$ and $C'_\beta \cap Q_3(\mathcal{Q})$ in the 2^i-subcurves $Q_2(\mathcal{Q})$ and $Q_3(\mathcal{Q})$, respectively:

The assumption on the column indices α and β gives that the two segments/columns $C'_\alpha \cap Q_2(\mathcal{Q})$ and $C'_\beta \cap Q_3(\mathcal{Q})$ in the canonical $Q_2(\mathcal{Q})$ and $Q_3(\mathcal{Q})$, respectively, have the same column indices. Therefore the difference in the index-cumulation of these two segments in $\mathcal{Q}$ is effectively $2^i \cdot 2^{2i}$.

Thus, $\Delta(C_{k,\beta}, \partial_1(H_k^2)) - \Delta(C_{k,\alpha}, \partial_1(H_k^2)) = 2^i(3 \cdot 2^{2i}) + 2^i \cdot 2^{2i} = 2^2 \cdot 2^{3i}$. ∎

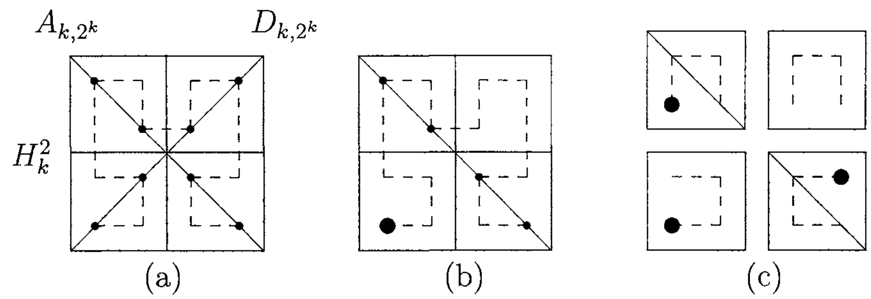


Figure 3.5: (a) Mirror bijection between $A_{k,2^k}$ and $D_{k,2^k}$ in a canonical H_k^2; (b) decomposition of $A_{k,2^k}$ into two segments in $Q_2(H_k^2)$ and $Q_4(H_k^2)$; (c) viewing the segments as the auxiliary diagonal $A_{k-1,2^{k-1}}$ in the two canonically oriented H_{k-1}^2.

Lemma 3.3 *For* $k \geq 1$*,*

1. $\Delta(A_{k,2^k}, \partial_1(H_k^2)) + \Delta(D_{k,2^k}, \partial_1(H_k^2)) = 2^{3k} - 2^k$*, and*

2. $\Delta(A_{k,2^k}, \partial_1(H_k^2)) = \frac{2}{3} \cdot 2^{3k} - \frac{2}{3} \cdot 2^k$.

Proof. Let k be an arbitrary positive integer.

Part 1: The mirror relation yields a bijection between the grid points of $A_{k,2^k}$ and $D_{k,2^k}$, as shown in Figure 3.5(a). By enumerating all mirror pairs in $A_{k,2^k} \times D_{k,2^k}$, we have :

$$
\begin{aligned}
& \Delta(A_{k,2^k}, \partial_1(H_k^2)) + \Delta(D_{k,2^k}, \partial_1(H_k^2)) \\
= & \sum_{\text{all mirror pairs } (v,v') \text{ in } A_{k,2^k} \times D_{k,2^k}} (\hbar_k(v, \partial_1(H_k^2)) + \hbar_k(v', \partial_1(H_k^2))) \\
= & \sum_{\text{all mirror pairs } (v,v') \text{ in } A_{k,2^k} \times D_{k,2^k}} (2^{2k} - 1) \\
= & 2^k(2^{2k} - 1) = 2^{3k} - 2^k.
\end{aligned}
$$

Part 2: The auxiliary diagonal $A_{k,2^k}$ in a canonical H_k^2 can be decomposed into two auxiliary-diagonal segments in $Q_2(H_k^2)$ and $Q_4(H_k^2)$ as shown in Figure 3.5(b) and (c). The segment $A_{k,2^k} \cap Q_2(H_k^2)$ is the auxiliary diagonal $A_{k-1,2^{k-1}}$ in the canonically oriented $Q_2(H_k^2)$. After $(+\frac{\pi}{2})$-rotating and then left-right reflecting $Q_4(H_k^2)$ into a canonical H_{k-1}^2, the segment $A_{k,2^k} \cap Q_4(H_k^2)$ is the auxiliary diagonal of the canonical H_{k-1}^2 (see Figure 3.5(c)). We partition $\Delta(A_{k,2^k}, \partial_1(H_k^2))$ into two parts:

From the segment $A_{k,2^k} \cap Q_2(H_k^2)$:

1. $\Delta(A_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2))$ and the cumulation of index-adjustment from $\partial_1(Q_2(H_k^2))$ to $\partial_1(H_k^2)$, which is

$$\begin{aligned} & \Delta(A_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2)) + \sum_{v \in A_{k,2^k} \cap Q_2(H_k^2)} \hbar_k(\partial_1(Q_2(H_k^2)), \partial_1(H_k^2)) \\ = \; & \Delta(A_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2)) + 2^{k-1} \cdot 2^{2(k-1)}, \end{aligned}$$

   and

2. From the segment $A_{k,2^k} \cap Q_4(H_k^2)$:

   $\Delta(A_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2)) + 2^{k-1}(3 \cdot 2^{2(k-1)})$, similarly.

Now we establish a recurrence (in k) for $\Delta(A_{k,2^k}, \partial_1(H_k^2))$ as follows:

$$\Delta(A_{k,2^k}, \partial_1(H_k^2)) = \begin{cases} 2\Delta(A_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2)) + 2^2 \cdot 2^{3(k-1)} & \text{if } k > 1 \\ 2^2 & \text{if } k = 1. \end{cases}$$

The closed-form solution for $\Delta(A_{k,2^k}, \partial_1(H_k^2))$ is $\frac{2}{3} \cdot 2^{3k} - \frac{2}{3} \cdot 2^k$. ∎

In the following lemma, we investigate the summations resulting from restricting to the six subcases of $\sum_{\alpha,\beta \in \{1,2,3,4\} | \alpha < \beta} \Delta_\delta(Q_\alpha(H_k^2), Q_\beta(H_k^2))$. For the case of contiguous subcurves, the boundary rows/columns and the boundary corners (of main- and auxiliary-diagonals) are involved; and for the case of diagonal subcurves, the boundary corners (of main- and auxiliary-diagonals) are involved.

Lemma 3.4 *For δ that is a positive integral power of 2, and $1 \leq \delta < 2^k$,*

$$\begin{aligned} \Delta_\delta(Q_1(H_k^2), Q_4(H_k^2)) &= 2(\overline{\mathcal{R}}_{k-1,\delta} + 2\mathcal{R}_{k-1,\delta-1} - \mathcal{D}'_{k-1,\delta-1} - \mathcal{A}'_{k-1,\delta-1}) \\ &\quad + (\frac{1}{2} \cdot 2^{2k} + 1)(\frac{1}{2} \cdot 2^k \delta^2 - 2\mathcal{N}_{k-1,\delta-1}), \end{aligned}$$

$$\begin{aligned} \Delta_\delta(Q_1(H_k^2), Q_2(H_k^2)) &= \Delta_\delta(Q_3(H_k^2), Q_4(H_k^2)) \\ &= \overline{\mathcal{C}}_{k-1,\delta} + 2\mathcal{C}_{k-1,\delta-1} - \mathcal{D}'_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1} \\ &\quad + \overline{\mathcal{R}}_{k-1,\delta} + 2\mathcal{R}_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1} - \mathcal{D}_{k-1,\delta-1} \\ &\quad + (\frac{1}{2} \cdot 2^k \delta^2 - 2\mathcal{N}_{k-1,\delta-1}), \end{aligned}$$

$$\Delta_\delta(Q_2(H_k^2), Q_3(H_k^2)) = 2(\overline{\mathcal{C}}_{k-1,\delta} + 2\mathcal{C}_{k-1,\delta-1} - \mathcal{D}'_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1}) + (\frac{1}{2} \cdot 2^k \delta^2 - 2\mathcal{N}_{k-1,\delta-1}),$$

and

$$\begin{aligned}\Delta_\delta(Q_1(H_k^2), Q_3(H_k^2)) &= \Delta_\delta(Q_2(H_k^2), Q_4(H_k^2)) \\ &= (\frac{1}{2^2} \cdot 2^{2k} + 1)\mathcal{N}_{k-1,\delta-1} + \mathcal{D}'_{k-1,\delta-1} + \mathcal{A}_{k-1,\delta-1}.\end{aligned}$$

Proof. From the illustration in Section 3.1.1.2 and from Equation 3.1, we expand $\Delta_\delta(Q_\alpha(H_k^2), Q_\beta(H_k^2))$ into three cumulations of index-differences:

$$\begin{aligned}\Delta_\delta(Q_\alpha(H_k^2), Q_\beta(H_k^2)) &= \sum_{(v,v')\in Q_\alpha(H_k^2)\times Q_\beta(H_k^2)|d_1(v,v')=\delta} \hbar_k(v,v') \\ &= \sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(v, \partial_2(Q_\alpha(H_k^2))) \\ &+ \sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(\partial_2(Q_\alpha(H_k^2)), \partial_1(Q_\beta(H_k^2))) \\ &+ \sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(v', \partial_1(Q_\beta(H_k^2))).\end{aligned}$$

For the case of contiguous subcurves, we compute the three cumulations of index-differences from Equations 3.2, 3.3, and 3.4 in Section 3.1.1.2. (All the notations in Section 3.1.1.2 applies for H_k^2 in this section.)

Consider the case when $(\alpha, \beta) = (1, 4)$. After $(+\frac{\pi}{2})$-rotating and then left-right reflecting $Q_4(H_k^2)$ into a canonical H_{k-1}^2, the structures c_η^4, $X_{k-1,\eta}^{D'}$, and $X_{k-1,\eta}^{A}$ are transformed into $R_{k-1,2^{k-1}+1-\eta}$, $D'_{k-1,\eta}$, and $A'_{k-1,\eta}$, respectively, for all $\eta \in [2^{k-1}]$, and $\partial_1(Q_4(H_k^2))$ into $\partial_1(H_{k-1}^2)$. From Equation 3.2,

$$\begin{aligned}&\sum_{(v,v')\in Q_{1,\delta}\times Q_{4,\delta}|d_1(v,v')=\delta} \hbar_k(v', \partial_1(Q_4(H_k^2))) \\ &= \sum_{j=1}^{\delta} \sum_{u\in R_{k-1,2^{k-1}+1-j}} (2(\delta+1-j)-1)\hbar_{k-1}(u, \partial_1(H_{k-1}^2)) \\ &\quad -\sum_{\eta=1}^{\delta} \sum_{u\in D'_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_1(H_{k-1}^2)) - \sum_{\eta=1}^{\delta} \sum_{u\in A'_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_1(H_{k-1}^2))\end{aligned}$$

$$\begin{aligned}
&= 2\sum_{j=1}^{\delta-1}\sum_{u\in R_{k-1,2^{k-1}+1-j}}(\delta-j)\hbar_{k-1}(u,\partial_1(H^2_{k-1}))\\
&\quad+\sum_{j=1}^{\delta}\sum_{u\in R_{k-1,2^{k-1}+1-j}}\hbar_{k-1}(u,\partial_1(H^2_{k-1}))\\
&\quad-\mathcal{D}'_{k-1,\delta-1}-\mathcal{A}'_{k-1,\delta-1}\\
&= 2\mathcal{R}_{k-1,\delta-1}+\overline{\mathcal{R}}_{k-1,\delta}-\mathcal{D}'_{k-1,\delta-1}-\mathcal{A}'_{k-1,\delta-1}.
\end{aligned}$$

For $\sum_{(v,v')\in Q_{1,\delta}\times Q_{4,\delta}|d_1(v,v')=\delta}\hbar_k(v,\partial_2(Q_1(H^2_k)))$, We may compute it according to Equation 3.3 for $(\alpha,\beta)=(1,4)$. The left-right reflection symmetry gives that

$$\sum_{(v,v')\in Q_{1,\delta}\times Q_{4,\delta}|d_1(v,v')=\delta}\hbar_k(v,\partial_2(Q_1(H^2_k)))=\sum_{(v,v')\in Q_{1,\delta}\times Q_{4,\delta}|d_1(v,v')=\delta}\hbar_k(v',\partial_1(Q_4(H^2_k))).$$

Since $\hbar_k(\partial_2(Q_1(H^2_k)),\partial_1(Q_4(H^2_k)))=2\cdot 2^{2(k-1)}+1$, we have (according to Equation 3.4):

$$\begin{aligned}
&\sum_{(v,v')\in Q_{1,\delta}\times Q_{4,\delta}|d_1(v,v')=\delta}\hbar_k(\partial_2(Q_1(H^2_k)),\partial_1(Q_4(H^2_k)))\\
&= (2^{k-1}\delta^2-2\mathcal{N}_{k-1,\delta-1})\hbar_k(\partial_2(Q_1(H^2_k)),\partial_1(Q_4(H^2_k)))\\
&= (2^{k-1}\delta^2-2\mathcal{N}_{k-1,\delta-1})(2\cdot 2^{2(k-1)}+1).
\end{aligned}$$

Summing up the three expansions, we have:

$$\begin{aligned}
&\Delta_\delta(Q_1(H^2_k),Q_4(H^2_k))\\
&= \sum_{(v,v')\in Q_{1,\delta}\times Q_{4,\delta}|d_1(v,v')=\delta}\hbar_k(v',\partial_1(Q_4(H^2_k)))\\
&\quad+\sum_{(v,v')\in Q_{1,\delta}\times Q_{4,\delta}|d_1(v,v')=\delta}\hbar_k(v,\partial_2(Q_1(H^2_k)))\\
&\quad+(2^{k-1}\delta^2-2\mathcal{N}_{k-1,\delta-1})(2\cdot 2^{2(k-1)}+1)\\
&= 2(2\mathcal{R}_{k-1,\delta-1}+\overline{\mathcal{R}}_{k-1,\delta}-\mathcal{D}'_{k-1,\delta-1}-\mathcal{A}'_{k-1,\delta-1})\\
&\quad+(2^{k-1}\delta^2-2\mathcal{N}_{k-1,\delta-1})(2(2^{k-1})^2+1)\\
&= 2(\overline{\mathcal{R}}_{k-1,\delta}+2\mathcal{R}_{k-1,\delta-1}-\mathcal{D}'_{k-1,\delta-1}-\mathcal{A}'_{k-1,\delta-1})\\
&\quad+(\frac{1}{2}\cdot 2^{2k}+1)(\frac{1}{2}\cdot 2^k\delta^2-2\mathcal{N}_{k-1,\delta-1}).
\end{aligned}$$

The derivations for the other three cases: $(\alpha, \beta) = (1,2)$, $(2,3)$, and $(3,4)$) are similar.

For the case of diagonal subcurves, we first observe that every δ-pair $(v, v') \in Q_1(H_k^2) \times Q_3(H_k^2)$ and its unique top-down reflection δ-pair $(u, u') \in Q_2(H_k^2) \times Q_4(H_k^2)$ have equal contribution in its own cumulation of index-differences, which implies that $\Delta_\delta(Q_1(H_k^2), Q_3(H_k^2)) = \Delta_\delta(Q_2(H_k^2), Q_4(H_k^2))$. It suffices to consider the derivation for $\Delta_\delta(Q_1(H_k^2), Q_3(H_k^2))$.

In deriving $\Delta_\delta(Q_1(H_k^2), Q_3(H_k^2))$, an index-difference computation $\hbar_{k-1}(v, \partial_2(H_{k-1}^2))$ appears. Since all our derivations and computations so far have involved $\partial_1(H_{k-1}^2)$, we observe that in a canonical H_{k-1}^2, for every $\eta \in [\delta-1]$, every grid point $u \in A'_{k-1,\eta}$ and its unique left-right reflection grid point $u' \in D'_{k-1,\eta}$ satisfy that $\hbar_{k-1}(u, \partial_2(H_{k-1}^2)) = \hbar_{k-1}(u', \partial_1(H_{k-1}^2))$.

According to Equation 3.5, we now expand $\Delta_\delta(Q_1(H_k^2), Q_3(H_k^2))$ as follows:

$$
\begin{aligned}
& \Delta_\delta(Q_1(H_k^2), Q_3(H_k^2)) \\
= & \sum_{\eta=1}^{\delta-1} \sum_{u \in Y_{k-1,\eta}^{A'}} (\delta-\eta)\hbar_k(u, \partial_2(Q_1(H_k^2))) \\
& + \sum_{\eta=1}^{\delta-1} \sum_{u \in X_{k-1,\eta}^{A}} (\delta-\eta)\hbar_k(u, \partial_1(Q_3(H_k^2))) \\
& + \mathcal{N}_{k-1,\delta-1}\hbar_k(\partial_2(Q_1(H_k^2)), \partial_1(Q_3(H_k^2))) \\
= & \sum_{\eta=1}^{\delta-1} \sum_{u \in A'_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_2(H_{k-1}^2)) \\
& \qquad \text{(after } (+\tfrac{\pi}{2})\text{-rotating and then left-right reflecting} \\
& \qquad Q_1(H_k^2) \text{ into a canonical } H_{k-1}^2) \\
& + \sum_{\eta=1}^{\delta-1} \sum_{u \in A_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_1(H_{k-1}^2)) \\
& + (2^{2(k-1)} + 1)\mathcal{N}_{k-1,\delta-1} \\
= & \sum_{\eta=1}^{\delta-1} \sum_{u \in D'_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_1(H_{k-1}^2)) + (\frac{1}{2^2} \cdot 2^{2k} + 1)\mathcal{N}_{k-1,\delta-1} + \mathcal{A}_{k-1,\delta-1}
\end{aligned}
$$

$$= (\frac{1}{2^2} \cdot 2^{2k} + 1)\mathcal{N}_{k-1,\delta-1} + \mathcal{D}'_{k-1,\delta-1} + \mathcal{A}_{k-1,\delta-1}.$$

The derivation for $\Delta_\delta(Q_2(H_k^2), Q_4(H_k^2))$ are the same as described above. ∎

The following two lemmas allow us to simplify the overall summation of $L_\delta(H_k^2)$.

Lemma 3.5 *For all integers q with $0 \le q \le k$,*

1. $\overline{\mathcal{W}}_k = \frac{1}{2} \cdot 2^{4k} - \frac{1}{2} \cdot 2^{2k}$,
2. $\overline{\mathcal{N}}_{k,2^q} = \frac{1}{2} \cdot 2^{2q} + \frac{1}{2} \cdot 2^q$,
3. $\overline{\mathcal{R}}_{k,2^q} = 2^q \Delta(R_{k,1}, \partial_1(H_k^2))$,
4. $\overline{\mathcal{C}}_{k,2^q} = 2^q \Delta(C_{k,1}, \partial_1(H_k^2)) + \frac{2}{7}(2^{4q} - 2^q)$,
5. $\overline{\mathcal{A}}_{k,2^k} + \overline{\mathcal{A}'}_{k,2^k} = \overline{\mathcal{W}}_k + \Delta(A_{k,2^k}, \partial_1(H_k^2))$,
6. $\overline{\mathcal{A}}_{k,2^k} = 2\overline{\mathcal{W}}_{k-1} + \Delta(A_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2)) + 2^{2k}\overline{\mathcal{N}}_{k-1,2^{k-1}}$,
7. *Each of the pairs: $(\overline{\mathcal{D}}_{k,2^q}, \overline{\mathcal{A}}_{k,2^q})$ and $(\overline{\mathcal{D}'}_{k,2^q}, \overline{\mathcal{A}'}_{k,2^q})$ are related via $\overline{\mathcal{N}}_{k,2^q}$ as follows:*

$$\overline{\mathcal{D}}_{k,2^q} + \overline{\mathcal{A}}_{k,2^q} = \overline{\mathcal{D}'}_{k,2^q} + \overline{\mathcal{A}'}_{k,2^q} = (2^{2k} - 1)\overline{\mathcal{N}}_{k,2^q},$$

   and

8. $\overline{\mathcal{D}'}_{k,2^q} = \frac{1}{3}(2^{2k} - 2^{2q})\overline{\mathcal{N}}_{k,2^q} + \overline{\mathcal{D}'}_{q,2^q}$

Proof. let q be an arbitrary integer with $0 \le q \le k$.

Part 1: $\overline{\mathcal{W}}_k = \sum_{v \in H_k^2} \hbar_k(v, \partial_1(H_k^2)) = \sum_{\eta \in [2^{2k}]} (\eta - 1) = \frac{1}{2} \cdot 2^{4k} - \frac{1}{2} \cdot 2^{2k}$.

Part 2: For the symbol X denoting D, D', A, or A' (main or auxiliary diagonals), the number of grid points in $X_{k,\beta}$, where $\beta \in [2^k]$, is β. Therefore,

$$\overline{\mathcal{N}}_{k,2^q} = \sum_{\beta=1}^{2^q} \sum_{v \in X_{k,\beta}} 1 = \sum_{\beta=1}^{2^q} \beta = \frac{1}{2} \cdot 2^{2q} + \frac{1}{2} \cdot 2^q.$$

Part 3: Lemma 3.1 says that for all $\beta \in [2^k]$, $\Delta(R_{k,\beta}, \partial_1(H_k^2)) = \Delta(R_{k,\beta}, \partial_2(H_k^2)) = \frac{1}{2} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k$ (independent of β). Therefore,

$$\begin{aligned}\overline{\mathcal{R}}_{k,2^q} &= \sum_{\beta=1}^{2^q}\sum_{v\in R_{k,\beta}} \hbar_k(v,\partial_1(H_k^2)) = \sum_{\beta=1}^{2^q}\Delta(R_{k,\beta},\partial_1(H_k^2)) \\ &= 2^q\Delta(R_{k,1},\partial_1(H_k^2))(= 2^q(\frac{1}{2}\cdot 2^{3k}-\frac{1}{2}\cdot 2^k)).\end{aligned}$$

Part 4: Lemma 3.2 (part 2) says that for all $\alpha,\beta\in[2^k]$ with $\alpha<\beta$ and $(\alpha-1)\oplus(\beta-1)=2^i$, where $i\in\{0,1,\ldots,k-1\}$, $\Delta(C_{k,\beta},\partial_1(H_k^2))=\Delta(C_{k,\alpha},\partial_1(H_k^2))+2^2\cdot 2^{3i}$; hence for all $\eta\in[2^{q-1}]$,

$$\Delta(C_{k,2^{q-1}+\eta},\partial_1(H_k^2)) = \Delta(C_{k,\eta},\partial_1(H_k^2)) + 2^2\cdot 2^{3(q-1)}.$$

We establish a recurrence (in q) for $C_{k,2^q}$ as follows:

$$\begin{aligned}\overline{\mathcal{C}}_{k,2^q} &= \sum_{\eta=1}^{2^q}\Delta(C_{k,\eta},\partial_1(H_k^2)) \\ &= \sum_{\eta=1}^{2^{q-1}}\Delta(C_{k,\eta},\partial_1(H_k^2)) + \sum_{\eta=2^{q-1}+1}^{2^q}\Delta(C_{k,\eta},\partial_1(H_k^2)) \\ &= \overline{\mathcal{C}}_{k,2^{q-1}} + \sum_{\eta=2^{q-1}+1}^{2^q}\Delta(C_{k,\eta},\partial_1(H_k^2)) \\ &= \overline{\mathcal{C}}_{k,2^{q-1}} + \sum_{\eta=1}^{2^{q-1}}\Delta(C_{k,2^{q-1}+\eta},\partial_1(H_k^2)) \\ &= \overline{\mathcal{C}}_{k,2^{q-1}} + \sum_{\eta=1}^{2^{q-1}}(\Delta(C_{k,\eta},\partial_1(H_k^2)) + 2^2\cdot 2^{3(q-1)}) \\ &= 2\overline{\mathcal{C}}_{k,2^{q-1}} + 2^{q-1}\cdot 2^{2+3(q-1)} \\ &= 2\overline{\mathcal{C}}_{k,2^{q-1}} + 2^{4q-2}.\end{aligned}$$

Iterating the recurrence in descending q (to 0), we have:

$$\overline{\mathcal{C}}_{k,2^q} = 2^q\overline{\mathcal{C}}_{k,2^0} + \sum_{\eta=1}^{q}2^{q-\eta}\cdot 2^{4\eta-2} = 2^q\Delta(C_{k,1},\partial_1(H_k^2)) + \frac{2}{7}(2^{4q}-2^q).$$

Part 5: In a canonical H_k^2, for all $\alpha\in[2^k]$, the "structures" $\overline{\mathcal{A}}_{k,2^k}$ and $\overline{\mathcal{A}'}_{k,2^k}$ cover the lower-left and upper-right corners of α auxiliary diagonals, respectively (see Figure 3.1(c)); and the coverages of $\overline{\mathcal{A}}_{k,2^k}$ and $\overline{\mathcal{A}'}_{k,2^k}$ have a common auxiliary diagonal

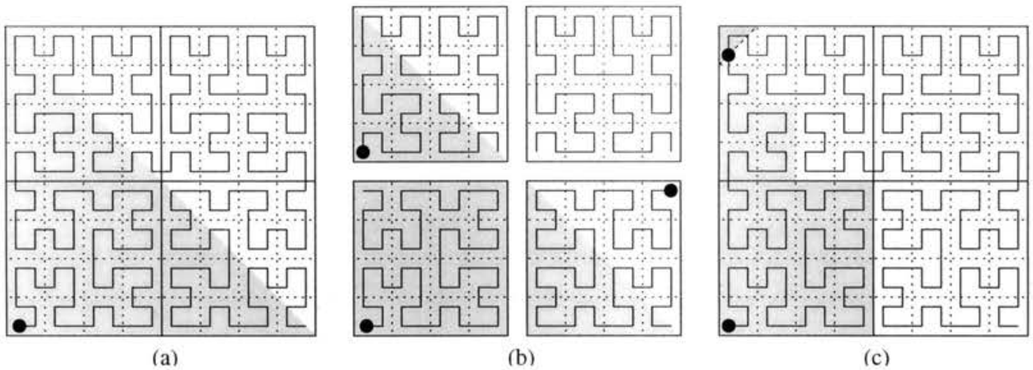


Figure 3.6: Coverages and decompositions of auxiliary and main diagonal structures in a canonical H_k^2: (a) and (b) for $\overline{\mathcal{A}}_{k,2^k}$; (c) for$\overline{\mathcal{D}'}_{k,2^k}$.

$A_{k,2^k}(= A'_{k,2^k})$. Therefore,

$$\overline{\mathcal{A}}_{k,2^k} + \overline{\mathcal{A}'}_{k,2^k} = \sum_{\beta=1}^{2^k} \Delta(A_{k,\beta}, \partial_1(H_k^2)) + \sum_{\beta=1}^{2^k} \Delta(A'_{k,\beta}, \partial_1(H_k^2))$$

$$= \sum_{\beta=1}^{2^k-1} \Delta(A_{k,\beta}, \partial_1(H_k^2)) + \Delta(A_{k,2^k}, \partial_1(H_k^2)) + \sum_{\beta=1}^{2^k} \Delta(A'_{k,\beta}, \partial_1(H_k^2))$$

$$= \sum_{v\in H_k^2} \hbar_k(v, \partial_1(H_k^2)) + \Delta(A_{k,2^k}, \partial_1(H_k^2)) = \overline{\mathcal{W}}_k + \Delta(A_{k,2^k}, \partial_1(H_k^2)).$$

Part 6: The structure $\overline{\mathcal{A}}_{k,2^k}$ covers the lower-left half of a canonical H_k^2. We decompose its coverage into three parts as shown in Figure 3.6(a): $Q_1(H_k^2)$, the lower-left half of $Q_2(H_k^2)$, and the lower-left half of $Q_4(H_k^2)$. Accordingly, we partition $\overline{\mathcal{A}}_{k,2^k}$ into three parts (see Figure 3.6(b)):

1. From $Q_1(H_k^2)$: the index-cumulation of all grid points in $Q_1(H_k^2)$ is $\overline{\mathcal{W}}_{k-1}$,

2. From the lower-left half of $Q_2(H_k^2)$ (covered by $\overline{\mathcal{A}}_{k-1,2^{k-1}}$): the index-cumulation of the lower-left half of $Q_2(H_k^2)$ with respect to $\partial_1(Q_2(H_k^2))$ and the cumulation of index-adjustment from $\partial_1(Q_2(H_k^2))$ to $\partial_1(H_k^2)$; that is, $\overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot 2^{2(k-1)}$, and

3. From the lower-left half of $Q_4(H_k^2)$ (covered by $\overline{\mathcal{A}'}_{k-1,2^{k-1}}$, after $(+\frac{\pi}{2})$-rotating and then left-right reflecting $Q_4(H_k^2)$ and its lower-left half into a canonical H_{k-1}^2 and its upper-left half, respectively): $\overline{\mathcal{A}'}_{k-1,2^{k-1}}$ and the cumulation of

index-adjustment from $\partial_1(Q_4(H_k^2))$ to $\partial_1(H_k^2)$; that is, $\overline{\mathcal{A}'}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot (3 \cdot 2^{2(k-1)})$.

Thus,

$$\overline{\mathcal{A}}_{k,2^k} = \sum_{\beta=1}^{2^k} \sum_{v \in A_{k,\beta}} \hbar_k(v, \partial_1(H_k^2))$$

$$= \sum_{\beta=1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_1(H_k^2)} \hbar_k(v, \partial_1(H_k^2)) + \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_2(H_k^2)} \hbar_k(v, \partial_1(H_k^2))$$

$$+ \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_4(H_k^2)} \hbar_k(v, \partial_1(H_k^2))$$

$$= \overline{\mathcal{W}}_{k-1} + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_2(H_k^2)} \hbar_k(v, \partial_1(H_k^2)) + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_4(H_k^2)} \hbar_k(v, \partial_1(H_k^2))$$

$$= \overline{\mathcal{W}}_{k-1}$$

$$+ \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_2(H_k^2)} (\hbar_k(v, \partial_1(Q_2(H_k^2))) + \hbar_k(\partial_1(Q_2(H_k^2)), \partial_1(H_k^2)))$$

("$v \in A_{k,2^{k-1}+\beta} \cap Q_2(H_k^2)$" is equivalent to "$v \in A_{k-1,\beta}$ in a canonical H_{k-1}^2")

$$+ \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_4(H_k^2)} (\hbar_k(v, \partial_1(Q_4(H_k^2))) + \hbar_k(\partial_1(Q_4(H_k^2)), \partial_1(H_k^2)))$$

("$v \in A_{k,2^{k-1}+\beta} \cap Q_4(H_k^2)$" is equivalent to "$v \in A'_{k-1,\beta}$in a canonical H_{k-1}^2")

$$= \overline{\mathcal{W}}_{k-1} + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k-1,\beta}} (\hbar_{k-1}(v, \partial_1(H_{k-1}^2)) + 2^{2(k-1)})$$

$$+ \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (\hbar_{k-1}(v, \partial_1(H_{k-1}^2)) + 3 \cdot 2^{2(k-1)})$$

$$= \overline{\mathcal{W}}_{k-1} + (\overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot 2^{2(k-1)}) + (\overline{\mathcal{A}'}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot (3 \cdot 2^{2(k-1)}))$$

$$= \overline{\mathcal{W}}_{k-1} + \overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{A}'}_{k-1,2^{k-1}} + 2^{2k}\overline{\mathcal{N}}_{k-1,2^{k-1}}$$

$$= 2\overline{\mathcal{W}}_{k-1} + \Delta(A_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2)) + 2^{2k}\overline{\mathcal{N}}_{k-1,2^{k-1}} \quad \text{(by part 5).}$$

Part 7: We prove the equality $\overline{\mathcal{D}}_{k,2^q} + \overline{\mathcal{A}}_{k,2^q} = (2^{2k} - 1)\overline{\mathcal{N}}_{k,2^q}$, and the proof of the equality $\overline{\mathcal{D}'}_{k,2^q} + \overline{\mathcal{A}'}_{k,2^q} = (2^{2k} - 1)\overline{\mathcal{N}}_{k,2^q}$ is similar. We proceed as in the proof of

Lemma 3.3 (part 1), by considering the mirror bijection between the grid points of $A_{k,\beta}$ and $D_{k,\beta}$ for all $\beta \in [2^k]$:

$$\overline{\mathcal{D}}_{k,2^q} + \overline{\mathcal{A}}_{k,2^q} = \sum_{\beta=1}^{2^q} \sum_{v \in D_{k,\beta}} \hbar_k(v, \partial_1(H_k^2)) + \sum_{\beta=1}^{2^q} \sum_{v \in A_{k,\beta}} \hbar_k(v, \partial_1(H_k^2))$$

$$= \sum_{\beta=1}^{2^q} \sum_{\text{all mirror pairs } (v,v') \in D_{k,\beta} \times A_{k,\beta}} (\hbar_k(v, \partial_1(H_k^2)) + \hbar_k(v', \partial_1(H_k^2)))$$

$$= \sum_{\beta=1}^{2^q} \sum_{\text{all mirror pairs } (v,v') \in D_{k,\beta} \times A_{k,\beta}} (2^{2k} - 1)$$

$$= (2^{2k} - 1) \sum_{\beta=1}^{2^q} \sum_{\text{all mirror pairs } (v,v') \in D_{k,\beta} \times A_{k,\beta}} 1 = (2^{2k} - 1)\overline{\mathcal{N}}_{k,2^q}.$$

Part 8: If $q = 1$, then the desired equality is obviously true. Consider that $q < k$. The structure $\overline{\mathcal{D}'}_{k,2^q}$ covers the upper-left corner of 2^q main diagonals in a canonical H_k^2. In fact, $\overline{\mathcal{D}'}_{k,2^q}$ is the upper-left half of $Q_2^{k-q}(H_k^2)$, which is an H_q^2-subcurve canonically oriented at the upper-left corner of the H_k^2 (see Figure 3.6(c)). Therefore,

$$\overline{\mathcal{D}'}_{k,2^q} = \sum_{\beta=1}^{2^q} \sum_{v \in D'_{k,\beta}} \hbar_k(v, \partial_1(H_k^2))$$

$$= \sum_{\beta=1}^{2^q} \sum_{v \in D'_{k,\beta}} (\hbar_k(v, \partial_1(Q_2^{k-q}(H_k^2))) + \hbar_k(\partial_1(Q_2^{k-q}(H_k^2)), \partial_1(H_k^2)))$$

$$= \sum_{\beta=1}^{2^q} \sum_{v \in D'_{q,\beta}} (\hbar_q(v, \partial_1(H_q^2)) + \sum_{\eta=q}^{k-1} 2^{2\eta}) = \sum_{\beta=1}^{2^q} \sum_{v \in D'_{q,\beta}} \hbar_q(v, \partial_1(H_q^2)) + \sum_{\beta=1}^{2^q} \sum_{v \in D'_{q,\beta}} \sum_{\eta=q}^{k-1} 2^{2\eta}$$

$$= \overline{\mathcal{D}'}_{q,2^q} + \frac{1}{3}(2^{2k} - 2^{2q}) \sum_{\beta=1}^{2^q} \sum_{v \in D'_{q,\beta}} 1 = \frac{1}{3}(2^{2k} - 2^{2q})\overline{\mathcal{N}}_{k,2^q} + \overline{\mathcal{D}'}_{q,2^q}.$$

∎

Computations of various $\mathcal{X}_{k,2^q}$ are similar to those for $\overline{\mathcal{X}}_{k,2^q}$.

Lemma 3.6 *For all integers q with $0 \leq q \leq k$,*

1. $\mathcal{N}_{k,2^q} = \frac{1}{2 \cdot 3} \cdot 2^{3q} + \frac{1}{2} \cdot 2^{2q} + \frac{1}{3} \cdot 2^q$;

2. $\mathcal{R}_{k,2^q} = \Delta(R_{k,1}, \partial_1(H_k^2)) \sum_{\beta=1}^{2^q} \beta$,

3. $\mathcal{C}_{k,2^q} = \frac{3}{2^2\cdot 7}\cdot 2^{3k+2q} + \frac{3}{2^2\cdot 7}\cdot 2^{3k+q} - \frac{1}{2^2}\cdot 2^{k+2q} - \frac{1}{2^2}\cdot 2^{k+q} + \frac{2^3}{3\cdot 5\cdot 7}\cdot 2^{5q} + \frac{1}{7}\cdot 2^{4q} + \frac{1}{3\cdot 5}\cdot 2^q$,

4. *Each of the pairs:* $(\mathcal{D}_{k,2^q}, \mathcal{A}_{k,2^q})$ *and* $(\mathcal{D}'_{k,2^q}, \mathcal{A}'_{k,2^q}))$ *are related via* $\mathcal{N}_{k,2^q}$ *as follows:*

$$\mathcal{D}_{k,2^q} + \mathcal{A}_{k,2^q} = \mathcal{D}'_{k,2^q} + \mathcal{A}'_{k,2^q} = (2^{2k}-1)\mathcal{N}_{k,2^q},$$

5.

$$\mathcal{A}_{k,2^k} = \begin{cases} 0 & \textit{if } k=0 \\ 4 & \textit{if } k=1 \\ \frac{7}{2^2\cdot 3^2\cdot 5}\cdot 2^{5k} + \frac{3}{2^4}\cdot 2^{4k} + \frac{5}{2^2\cdot 3^2}\cdot 2^{3k} - \frac{1}{2^2}\cdot 2^{2k} - \frac{2^3}{3^2\cdot 5}\cdot 2^k & \textit{otherwise,} \end{cases}$$

6. $\mathcal{A}_{k,2^q} = \mathcal{A}_{q,2^q}$,

7.

$$\mathcal{D}'_{k,2^k} = \begin{cases} 0 & \textit{if } k=0 \\ 4 & \textit{if } k=1 \\ \frac{11}{2^2\cdot 3^2\cdot 5}\cdot 2^{5k} + \frac{3}{2^4}\cdot 2^{4k} + \frac{1}{2^2\cdot 3^2}\cdot 2^{3k} - \frac{1}{2^2}\cdot 2^{2k} - \frac{2^2}{3^2\cdot 5}\cdot 2^k & \textit{otherwise,} \end{cases}$$

8. $\mathcal{D}'_{k,2^q} = \frac{1}{3}(2^{2k} - 2^{2q})\mathcal{N}_{k,2^q} + \mathcal{D}'_{q,2^q}$.

Proof. Let q be an arbitrary integer with $0 \le q \le k$.

Part 1: Similar to the proof of Lemma 3.5 (part 2). For the symbol X denoting D, D', A, or A', the number of grid points in $X_{k,\beta}$, where $\beta \in [2^k]$, is β. Therefore,

$$\mathcal{N}_{k,2^q} = \sum_{\beta=1}^{2^q}\sum_{v\in X_{k,\beta}}(2^q+1-\beta) = \sum_{\beta=1}^{2^q}(2^q+1-\beta)\beta = \frac{1}{2\cdot 3}\cdot 2^{3q} + \frac{1}{2}\cdot 2^{2q} + \frac{1}{3}\cdot 2^q.$$

Part 2: Similar to the proof of Lemma 3.5 (part 3). Lemma 3.1 says that for all $\beta \in [2^k]$, $\Delta(R_{k,\beta}, \partial_1(H_k^2)) = \Delta(R_{k,\beta}, \partial_2(H_k^2)) = \frac{1}{2}\cdot 2^{3k} - \frac{1}{2}\cdot 2^k$ (independent of β). Therefore,

$$\begin{aligned} & \mathcal{R}_{k,2^q} \\ = \ & \sum_{\beta=1}^{2^q}\sum_{v\in R_{k,\beta}}(2^q+1-\beta)\hbar_k(v,\partial_1(H_k^2)) = \sum_{\beta=1}^{2^q}((2^q+1-\beta)\sum_{v\in R_{k,\beta}}\hbar_k(v,\partial_1(H_k^2))) \\ = \ & \sum_{\beta=1}^{2^q}(2^q+1-\beta)\Delta(R_{k,\beta},\partial_1(H_k^2)) = \Delta(R_{k,1},\partial_1(H_k^2))\sum_{\beta=1}^{2^q}(2^q+1-\beta) \\ = \ & \Delta(R_{k,1},\partial_1(H_k^2))\sum_{\beta=1}^{2^q}\beta. \end{aligned}$$

Part 3: We first express $C_{k,2^q}$ in terms of $\overline{C}_{k,2^q}$ (computed in Lemma 3.5 (part 4) and Lemma 3.2 (part 1)) and $\sum_{\beta=1}^{2^q} \beta\Delta(C_{k,\beta}, \partial_1(H_k^2))$:

$$
\begin{aligned}
&\mathcal{C}_{k,2^q} \\
= \; & \sum_{\beta=1}^{2^q} \sum_{v \in C_{k,\beta}} (2^q + 1 - \beta)\hbar_k(v, \partial_1(H_k^2)) = \sum_{\beta=1}^{2^q}((2^q+1-\beta) \sum_{v\in C_{k,\beta}} \hbar_k(v, \partial_1(H_k^2))) \\
= \; & \sum_{\beta=1}^{2^q}(2^q+1-\beta)\Delta(C_{k,\beta}, \partial_1(H_k^2)) = \sum_{\beta=1}^{2^q}((2^q+1)\Delta(C_{k,\beta}, \partial_1(H_k^2)) - \beta\Delta(C_{k,\beta}, \partial_1(H_k^2))) \\
= \; & (2^q+1)\sum_{\beta=1}^{2^q}\Delta(C_{k,\beta}, \partial_1(H_k^2)) - \sum_{\beta=1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(H_k^2)) \\
= \; & (2^q+1)\overline{\mathcal{C}}_{k,2^q} - \sum_{\beta=1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(H_k^2)).
\end{aligned}
$$

Let $\mathcal{U}_{k,2^q}$ denote $\sum_{\beta=1}^{2^q} \beta\Delta(C_{k,\beta}, \partial_1(H_k^2))$, so $\mathcal{U}_{k,2^0} = \Delta(C_{k,1}, \partial_1(H_k^2))$, which is computed in Lemma 3.2 (part 1). We establish a recurrence (in q) for $\mathcal{U}_{k,2^q}$, similar to that for $\overline{\mathcal{C}}_{k,2^q}$ in the proof of Lemma 3.5 (part 4), as follows:

$$
\begin{aligned}
&\mathcal{U}_{k,2^q} \\
= \; & \sum_{\beta=1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(H_k^2)) = \sum_{\beta=1}^{2^{q-1}}\beta\Delta(C_{k,\beta}, \partial_1(H_k^2)) + \sum_{\beta=2^{q-1}+1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(H_k^2)) \\
= \; & \mathcal{U}_{k,2^{q-1}} + \sum_{\beta=1}^{2^{q-1}}(2^{q-1}+\beta)\Delta(C_{k,2^{q-1}+\beta}, \partial_1(H_k^2)) \\
= \; & \mathcal{U}_{k,2^{q-1}} + \sum_{\beta=1}^{2^{q-1}}(2^{q-1}+\beta)(\Delta(C_{k,\beta}, \partial_1(H_k^2)) + 2^2 \cdot 2^{3(q-1)}) \qquad \text{by Lemma 3.2 (part 2)} \\
= \; & \mathcal{U}_{k,2^{q-1}} + 2^{q-1}\sum_{\beta=1}^{2^{q-1}}\Delta(C_{k,\beta}, \partial_1(H_k^2)) + \sum_{\beta=1}^{2^{q-1}}\beta\Delta(C_{k,\beta}, \partial_1(H_k^2)) + \sum_{\beta=1}^{2^{q-1}}(2^{q-1}+\beta)2^{3q-1} \\
= \; & 2\mathcal{U}_{k,2^{q-1}} + 2^{q-1}\overline{\mathcal{C}}_{k,2^{q-1}} + 3 \cdot 2^{5q-4} + 2^{4q-3}.
\end{aligned}
$$

With $\overline{\mathcal{C}}_{k,2^{q-1}}$ and $\mathcal{U}_{k,2^0} = \Delta(C_{k,1}, \partial_1(H_k^2))$ computed in Lemma 3.5 (part 4) and Lemma 3.2 (part 1), the closed-form solution for $\mathcal{U}_{k,2^q}$ is:

$$
\frac{3}{2^2 \cdot 7} \cdot 2^{3k+2q} + \frac{3}{2^2 \cdot 7} \cdot 2^{3k+q} - \frac{1}{2^2} \cdot 2^{k+2q} - \frac{1}{2^2} \cdot 2^{k+q} + \frac{2 \cdot 11}{3 \cdot 5 \cdot 7} \cdot 2^{5q} + \frac{1}{7} \cdot 2^{4q} - \frac{1}{3 \cdot 5} \cdot 2^q.
$$

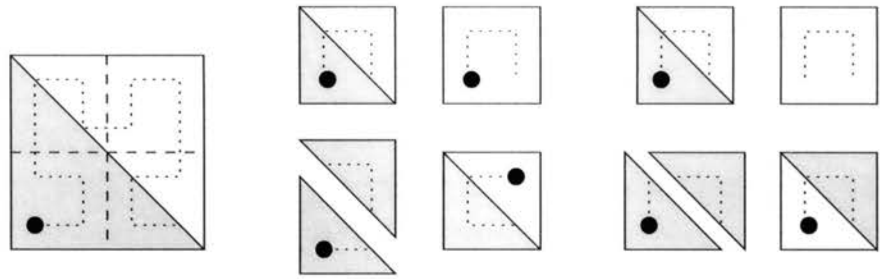


Figure 3.7: (a) Coverage of $\mathcal{A}_{k,2^k}$ in a canonical H_k^2; (b) decomposition of $\mathcal{A}_{k,2^k}$ into four triangular halves; (c) rotating and then reflecting $Q_1(H_k^2)$ and $Q_4(H_k^2)$ into two canonical H_{k-1}^2-subcurves.

Now,

$$\begin{aligned} & \mathcal{C}_{k,2^q} = (2^q+1)\overline{\mathcal{C}}_{k,2^q} - \mathcal{U}_{k,2^q} \\ = & \frac{3}{2^2 \cdot 7} \cdot 2^{3k+2q} + \frac{3}{2^2 \cdot 7} \cdot 2^{3k+q} - \frac{1}{2^2} \cdot 2^{k+2q} - \frac{1}{2^2} \cdot 2^{k+q} + \frac{2^3}{3 \cdot 5 \cdot 7} \cdot 2^{5q} \\ & + \frac{1}{7} \cdot 2^{4q} + \frac{1}{3 \cdot 5} \cdot 2^q. \end{aligned}$$

Part 4: Similar to the proof of Lemma 3.5 (part 7):

$$\begin{aligned} & \mathcal{D}_{k,2^q} + \mathcal{A}_{k,2^q} \\ = & \sum_{\beta=1}^{2^q} \sum_{v \in D_{k,\beta}} (2^q+1-\beta)\hbar_k(v, \partial_1(H_k^2)) + \sum_{\beta=1}^{2^q} \sum_{v \in A_{k,\beta}} (2^q+1-\beta)\hbar_k(v, \partial_1(H_k^2)) \\ = & \sum_{\beta=1}^{2^q} \sum_{\text{all mirror pairs } (v,v') \in D_{k,\beta} \times A_{k,\beta}} (2^q+1-\beta)(\hbar_k(v, \partial_1(H_k^2)) + \hbar_k(v', \partial_1(H_k^2))) \\ = & \sum_{\beta=1}^{2^q} \sum_{\text{all mirror pairs } (v,v') \in D_{k,\beta} \times A_{k,\beta}} (2^q+1-\beta)(2^{2k}-1) = (2^{2k}-1)\mathcal{N}_{k,2^q}. \end{aligned}$$

Part 5: We proceed as in the proof of Lemma 3.5 (part 6). The structure $\mathcal{A}_{k,2^k}$ covers the lower-left half of a canonical H_k^2. We decompose its coverages into four parts (non-empty when $k \geq 2$) as shown in Figure 3.7(a) and (b): the lower-left half of $Q_1(H_k^2)$, the upper-right half of $Q_1(H_k^2)$ without the auxiliary diagonal $A_{k,2^k}(= A'_{k,2^k})$, the lower-left half of $Q_2(H_k^2)$, and the lower-left half of $Q_4(H_k^2)$. Accordingly, we partition $\mathcal{A}_{k,2^k}$ into four parts (see Figure 3.7(c)) as follows. In the notation,

$\mathcal{A}_{k,\alpha} = \sum_{\beta=1}^{\alpha}\sum_{v\in A_{k,\beta}}(\alpha+1-\beta)\hbar_k(v,\partial_1(H_k^2))$, we associate the weight $\alpha+1-\beta$ with $A_{k,\beta}$ in the summation.

1. From the lower-left half of $Q_1(H_k^2)$ — consisting of auxiliary diagonals $A_{k,\beta}$ with weights $2^k+1-\beta$ for $\beta = 1, 2, \ldots, 2^{k-1}$ (indexed from the lower-left corner of $Q_1(H_k^2)$):

   After $(-\frac{\pi}{2})$-rotating and then left-right reflecting $Q_1(H_k^2)$ into a canonical H_{k-1}^2, we can see that these weighted auxiliary diagonals are transformed into auxiliary diagonals $A_{k-1,\beta}$ with weights $2^k+1-\beta$ for $\beta = 1, 2, \ldots, 2^{k-1}$ (indexed from the lower-left corner of the canonical H_{k-1}^2). Its contribution in $\mathcal{A}_{k,2^k}$ is:

$$\begin{aligned}
&\sum_{\beta=1}^{2^{k-1}}\sum_{v\in A_{k,\beta}\cap Q_1(H_k^2)}(2^k+1-\beta)\hbar_k(v,\partial_1(H_k^2))\\
=&\sum_{\beta=1}^{2^{k-1}}\sum_{v\in A_{k,\beta}\cap Q_1(H_k^2)}(2^{k-1}+1-\beta)\hbar_k(v,\partial_1(H_k^2))\\
&+\sum_{\beta=1}^{2^{k-1}}\sum_{v\in A_{k,\beta}\cap Q_1(H_k^2)}2^{k-1}\hbar_k(v,\partial_1(H_k^2))\\
=&\sum_{\beta=1}^{2^{k-1}}\sum_{v\in A_{k-1,\beta}}(2^{k-1}+1-\beta)\hbar_{k-1}(v,\partial_1(H_{k-1}^2))\\
&+2^{k-1}\sum_{\beta=1}^{2^{k-1}}\sum_{v\in A_{k-1,\beta}}\hbar_{k-1}(v,\partial_1(H_{k-1}^2))\\
=&\ \mathcal{A}_{k-1,2^{k-1}}+2^{k-1}\overline{\mathcal{A}}_{k-1,2^{k-1}}.
\end{aligned}$$

2. For the upper-right half of $Q_1(H_k^2)$ without the auxiliary diagonal $A_{k,2^k}$ — consisting of auxiliary-diagonal segments $A_{k,\beta}\cap Q_1(H_k^2)$ with weights $2^k+1-\beta$ for $\beta = 2^{k-1}+1, 2^{k-1}+2, \ldots, 2^k-1$ (indexed towards the upper-right corner of $Q_1(H_k^2)$):

   After $(-\frac{\pi}{2})$-rotating and then left-right reflecting $Q_1(H_k^2)$ into a canonical H_{k-1}^2, we can see that these weighted auxiliary-diagonal segments are transformed into auxiliary diagonals $A'_{k-1,\beta}$ with weights $\beta+1$ for $\beta = 1, 2, \ldots, 2^{k-1}-1$

(indexed from the upper-right corner of the canonical H^2_{k-1}). Its contribution in $\mathcal{A}_{k,2^k}$ is:

$$\begin{aligned}
& \sum_{\beta=2^{k-1}+1}^{2^k-1} \sum_{v \in A_{k,\beta} \cap Q_1(H_k^2)} (2^k+1-\beta)\hbar_k(v, \partial_1(H_k^2)) \\
= & \sum_{\gamma=1}^{2^{k-1}-1} \sum_{v \in A_{k,2^{k-1}+\gamma} \cap Q_1(H_k^2)} (2^k+1-(2^{k-1}+\gamma))\hbar_k(v, \partial_1(H_k^2)) \\
& \qquad \text{(change of summation index: } \beta = 2^{k-1}+\gamma) \\
= & \sum_{\beta=1}^{2^{k-1}-1} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_1(H_k^2)} (2^{k-1}+1-\beta)\hbar_k(v, \partial_1(H_k^2)) \\
= & \sum_{\beta=1}^{2^{k-1}-1} \sum_{v \in A'_{k-1,\beta}} (\beta+1)\hbar_{k-1}(v, \partial_1(H^2_{k-1})) \qquad \text{(in the canonical } H^2_{k-1}) \\
= & \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (\beta+1)\hbar_{k-1}(v, \partial_1(H^2_{k-1})) \\
& - \sum_{v \in A'_{k-1,2^{k-1}}} (2^{k-1}+1)\hbar_{k-1}(v, \partial_1(H^2_{k-1})) \\
= & (\sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (2^{k-1}+2)\hbar_{k-1}(v, \partial_1(H^2_{k-1})) \\
& \quad - \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (2^{k-1}+1-\beta)\hbar_{k-1}(v, \partial_1(H^2_{k-1}))) \\
& -(2^{k-1}+1)\Delta(A'_{k-1,2^{k-1}}, \partial_1(H^2_{k-1})) \\
= & (2^{k-1}+2)\overline{\mathcal{A}'}_{k-1,2^{k-1}} - \mathcal{A}'_{k-1,2^{k-1}} - (2^{k-1}+1)\Delta(A'_{k-1,2^{k-1}}, \partial_1(H^2_{k-1})).
\end{aligned}$$

3. From the lower-left half of the canonically oriented H^2_{k-1}-subcurve $Q_2(H_k^2)$ — consisting of auxiliary-diagonal segments $A_{k,\beta} \cap Q_2(H_k^2)$ with weights $2^k+1-\beta$ for $\beta = 2^{k-1}+1, 2^{k-1}+2, \ldots, 2^k-1$ (indexed from the lower-left corner of $Q_2(H_k^2)$):

   Including the cumulation of index-adjustment from $\partial_1(Q_2(H_k^2))$ to $\partial_1(H_k^2)$, its contribution in $\mathcal{A}_{k,2^k}$ is :

$$\begin{aligned}
&\sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v\in A_{k,\beta}\cap Q_2(H_k^2)} (2^k+1-\beta)\hbar_k(v,\partial_1(H_k^2)) \\
= &\sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v\in A_{k,\beta}\cap Q_2(H_k^2)} (2^k+1-\beta)(\hbar_k(v,\partial_1(Q_2(H_k^2))) \\
&\quad + \hbar_k(\partial_1(Q_2(H_k^2)),\partial_1(H_k^2))) \\
= &\sum_{\gamma=1}^{2^{k-1}} \sum_{v\in A_{k,2^{k-1}+\gamma}\cap Q_2(H_k^2)} (2^k+1-(2^{k-1}+\gamma))(\hbar_k(v,\partial_1(Q_2(H_k^2)))+2^{2(k-1)}) \\
&\qquad \text{(change of summation index: } \beta = 2^{k-1}+\gamma) \\
= &\sum_{\beta=1}^{2^{k-1}} \sum_{v\in A_{k,2^{k-1}+\beta}\cap Q_2(H_k^2)} (2^{k-1}+1-\beta)(\hbar_k(v,\partial_1(Q_2(H_k^2)))+2^{2(k-1)}) \\
= &\sum_{\beta=1}^{2^{k-1}} \sum_{v\in A_{k-1,\beta}} (2^{k-1}+1-\beta)\hbar_{k-1}(v,\partial_1(H_{k-1}^2)) \\
&+\sum_{\beta=1}^{2^{k-1}} \sum_{v\in A_{k-1,\beta}} (2^{k-1}+1-\beta)2^{2(k-1)} \\
= &\mathcal{A}_{k-1,2^{k-1}} + 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}}.
\end{aligned}$$

4. From the lower-left half of $Q_4(H_k^2)$ — consisting of auxiliary-diagonal segments $A_{k,\beta}\cap Q_4(H_k^2)$ with weights $2^k+1-\beta$ for $\beta = 2^{k-1}+1, 2^{k-1}+2, \ldots, 2^k-1$ (indexed from the lower-left corner of $Q_4(H_k^2)$):

   After $(+\frac{\pi}{2})$-rotating and then left-right reflecting $Q_4(H_k^2)$ into a canonical H_{k-1}^2, we can see that these weighted auxiliary-diagonal segments are transformed into auxiliary diagonals $A'_{k-1,\beta}$ with weights $2^{k-1}+1-\beta$ (indexed from the upper-right corner of the canonical H_{k-1}^2). Including the cumulation of index-adjustment from $\partial_1(Q_4(H_k^2))$ to $\partial_1(H_k^2)$, its contribution in $\mathcal{A}_{k,2^k}$ is:

$$\begin{aligned}
&\sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v\in A_{k,\beta}\cap Q_4(H_k^2)} (2^k+1-\beta)\hbar_k(v,\partial_1(H_k^2)) \\
= &\sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v\in A_{k,\beta}\cap Q_4(H_k^2)} (2^k+1-\beta)(\hbar_k(v,\partial_1(Q_4(H_k^2)))
\end{aligned}$$

$$
\begin{aligned}
& + \hbar_k(\partial_1(Q_4(H_k^2)), \partial_1(H_k^2))) \\
= & \sum_{\gamma=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\gamma} \cap Q_4(H_k^2)} (2^k + 1 - (2^{k-1} + \gamma))(\hbar_k(v, \partial_1(Q_4(H_k^2))) + 3 \cdot 2^{2(k-1)}) \\
& \qquad \text{(change of summation index: } \beta = 2^{k-1} + \gamma) \\
= & \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_4(H_k^2)} (2^{k-1} + 1 - \beta)(\hbar_k(v, \partial_1(Q_4(H_k^2))) + 3 \cdot 2^{2(k-1)}) \\
= & \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (2^{k-1} + 1 - \beta)\hbar_{k-1}(v, \partial_1(H_{k-1}^2)) \\
& + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (2^{k-1} + 1 - \beta)(3 \cdot 2^{2(k-1)}) \\
= & \mathcal{A}'_{k-1,2^{k-1}} + 3 \cdot 2^{2(k-1)} \mathcal{N}_{k-1,2^{k-1}}.
\end{aligned}
$$

With the four contributions, we establish a recurrence (in $k \geq 2$) for $\mathcal{A}_{k,2^k}$ as follows:

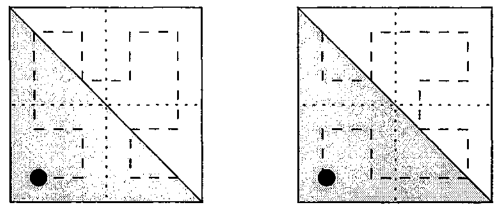
$$
\begin{aligned}
& \mathcal{A}_{k,2^k} = (\mathcal{A}_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{A}}_{k-1,2^{k-1}}) \\
& + ((2^{k-1} + 2)\overline{\mathcal{A}'}_{k-1,2^{k-1}} - \mathcal{A}'_{k-1,2^{k-1}} - (2^{k-1} + 1)\Delta(A'_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2))) \\
& + (\mathcal{A}_{k-1,2^{k-1}} + 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}}) \\
& + (\mathcal{A}'_{k-1,2^{k-1}} + 3 \cdot 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}}) \\
= \; & 2\mathcal{A}_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{A}}_{k-1,2^{k-1}} + (2^{k-1} + 2)\overline{\mathcal{A}'}_{k-1,2^{k-1}} + 2^{2k}\mathcal{N}_{k-1,2^{k-1}} \\
& - (2^{k-1} + 1)\Delta(A'_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2)).
\end{aligned}
$$

Note that $A'_{k-1,2^{k-1}}$ is identical to $A_{k-1,2^{k-1}}$, and Lemma 3.3 (part 2), Lemma 3.5 (parts 1, 2, 5, and 6), and Lemma 3.6 (part 1) are used to obtain exact formulas for $\overline{\mathcal{A}}_{k-1,2^{k-1}}$, $\overline{\mathcal{A}'}_{k-1,2^{k-1}}$, $\mathcal{N}_{k-1,2^{k-1}}$, and $\Delta(A'_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2))$ in above recurrence.

For $k < 2$, we compute $\mathcal{A}_{k,2^k}$ directly, and the recurrence for $\mathcal{A}_{k,2^k}$ is:

$$
\mathcal{A}_{k,2^k} = \begin{cases}
2\mathcal{A}_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{A}}_{k-1,2^{k-1}} + (2^{k-1} + 2)\overline{\mathcal{A}'}_{k-1,2^{k-1}} & \\
\quad + 2^{2k}\mathcal{N}_{k-1,2^{k-1}} - (2^{k-1} + 1)\Delta(A'_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2)) & \text{if } k > 1 \\
4 & \text{if } k = 1 \\
0 & \text{if } k = 0,
\end{cases}
$$

which yields the desired closed-form solution for $\mathcal{A}_{k,2^k}$.



(a) (b)

Figure 3.8: Two possible orientations of an H_q^2-subcurve for computing $\mathcal{A}_{k,2^q}$: (a) a canonical H_q^2; (b) an H_q^2-subcurve that can be reflected (with respect to the main diagonal) into a canonical H_q^2.

Part 6: The coverage of $\mathcal{A}_{k,2^q}$ in a canonical H_k^2 is an H_q^2-subcurve at the lower-left corner of the H_k^2 with two possible orientations as shown in Figure 3.8: (1) a canonically oriented H_q^2, or (2) an H_q^2-subcurve that can be $(-\frac{\pi}{2})$-rotated and then left-right reflected into a canonically H_q^2 (equivalently, a reflection with respect to the main diagonal joining the lower-left and upper-right corners of the H_q^2-subcurve). Thus, for computing $\mathcal{A}_{k,2^q}$ for both possibly oriented H_q^2-subcurves, the two sequences of auxiliary diagonals with associated weights are equal (via the reflection with respect to the main diagonal), and the two entry grid points are identical. Hence, $\mathcal{A}_{k,2^q} = \mathcal{A}_{q,2^q}$.

Part 7: Following the proof of part 5, we partition the coverage of $\mathcal{D}_{k,2^k}$ into four parts (non-empty when $k \geq 2$), and obtain:

$$
\begin{aligned}
&\mathcal{D}'_{k,2^k} \\
= \;& (\mathcal{D}_{k-1,2^{k-1}}) \\
&+(2^{k-1}\overline{\mathcal{D}'}_{k-1,2^{k-1}} + \mathcal{D}'_{k-1,2^{k-1}} + 2^{k-1}(\overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot 2^{2(k-1)}) + \mathcal{N}_{k-1,2^{k-1}} \cdot 2^{2(k-1)}) \\
&+((2^{k-1}+2)\overline{\mathcal{D}}_{k-1,2^{k-1}} - \mathcal{D}_{k-1,2^{k-1}} - \Delta(D_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2))(2^{k-1}+1) \\
&+((2^{k-1}+2)\overline{\mathcal{N}}_{k-1,2^{k-1}} - \mathcal{N}_{k-1,2^{k-1}} - 2^{k-1}(2^{k-1}+1))2^{2(k-1)}) \\
&+(\mathcal{D}'_{k-1,2^{k-1}} + \mathcal{N}_{k-1,2^{k-1}} \cdot 2 \cdot 2^{2(k-1)}).
\end{aligned}
$$

This gives a recurrence (in $k \geq 2$) for $\mathcal{D}'_{k,2^k}$ as follows:

$$
\mathcal{D}'_{k,2^k} = \begin{cases}
\begin{aligned}
&2\mathcal{D}'_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{D}'} + (2^{k-1}+2)\overline{\mathcal{D}} \\
&\;+((2 \cdot 2^{3k-3} + 2 \cdot 2^{2k-2})\overline{\mathcal{N}}_{k-1,2^{k-1}} + 2 \cdot 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}} \\
&\;-(2^{4k-4} + 2^{3k-3}) - \Delta(D_{k-1,2^{k-1}}, \partial_1(H_{k-1}^2))(2^{k-1}+1)
\end{aligned} & \text{if } k \geq 2 \\
4 & \text{if } k = 1 \\
0 & \text{if } k = 0,
\end{cases}
$$

which yields the desired closed-form solution for $\mathcal{D}'_{k,2^k}$.

Part 8: Following the proof of Lemma 3.5 (part 8), the coverage of $\mathcal{D}'_{k,2^q}$ is the upper-left half of $Q_2^{k-q}(H_k^2)$, which is an H_q^2-subcurve canonically oriented at the upper-left corner of a canonical H_k^2 (see Figure 3.6(c)). The cumulation of index-adjustment involves the summation of $\hbar_k(\partial_1(Q_2^{k-q}(H_k^2)), \partial_1(H_k^2))$, which cumulates the traversals through all the intermediate subcurves $Q_1(Q_2^{\eta}(H_k^2))$ for all $\eta \in \{0, 1, \ldots, k-q-1\}$. Thus,

$$\mathcal{D}'_{k,2^q} = \mathcal{N}_{k,2^q} \sum_{\eta=k-1}^{q} 2^{2\eta} + \mathcal{D}'_{q,2^q} = \frac{1}{3}(2^{2k} - 2^{2q})\mathcal{N}_{k,2^q} + \mathcal{D}'_{q,2^q}.$$

∎

With Lemmas 3.1 to 3.6, we obtain the closed-form formulas for $L_\delta(H_k^2)$.

Theorem 3.1 *For $\delta \in [2^k]$ that is an integral power of 2,*

$$L_\delta(H_k^2) = \begin{cases} \frac{17}{2\cdot 7} \cdot 2^{3k} - \frac{5}{2\cdot 3} \cdot 2^{2k} - \frac{2^3}{3\cdot 7} & \textit{if } \delta = 1 \\ \frac{17}{2\cdot 7} \cdot 2^{3k+2\log\delta} - \frac{2^3\cdot 3\cdot 5^2\cdot 7(k-\log\delta)+5\cdot 7\cdot 383}{2^4\cdot 3^3\cdot 5\cdot 7} \cdot 2^{2k+3\log\delta} & \\ \quad + \frac{2\cdot 3\cdot 5(k-\log\delta)-1}{2^2\cdot 3^3} \cdot 2^{2k+\log\delta} & \\ \quad - \frac{2^2\cdot 41}{3^3\cdot 5\cdot 7} \cdot 2^{5\log\delta} - \frac{2}{3^3} \cdot 2^{3\log\delta} - \frac{2}{3\cdot 5} \cdot 2^{\log\delta} & \textit{otherwise.} \end{cases}$$

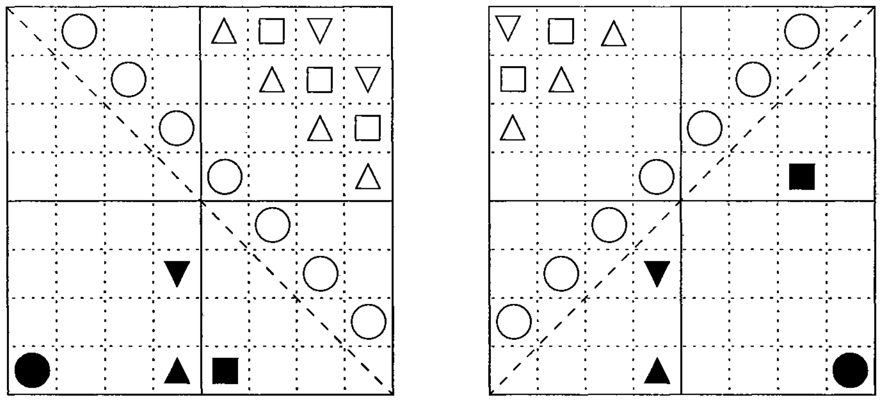
Proof. Let $\delta \in [2^k]$ that is an integral power of 2. Lemma 3.4 gives a recurrence (in k) for $L_\delta(H_k^2)$ for all $\delta \in [2^{k-1}]$ that is an integral power of 2. It suffices to determine the basis for the recurrence for all δ.

For the extreme case when $\delta = 1$, the basis for the recurrence (in k) for $L_1(H_k^2)$ occurs when $k = 1$; and $L_1(H_1^2) = 6$. The recurrence in Lemma 3.4 (for $\delta = 1$) with the basis gives the desired closed-form solution for $L_1(H_k^2)$.

For the general case when $\delta \geq 2$, the basis for the recurrence (in k) for $L_\delta(H_k^2)$ occurs when $k = \log\delta$, for which the recursive decomposition (in k) halts and we compute $L_\delta(H_{\log\delta}^2)$ based on, in a canonical $H_{\log\delta}^2$:

$$L_\delta(H_{\log\delta}^2) = \sum_{i,j\in[\delta^2] \mid i<j \text{ and } d_1(H_{\log\delta}^2(i), H_{\log\delta}^2(j))=\delta} |i-j| = \sum_{v,v'\in H_{\log\delta}^2 \mid d_1(v,v')=\delta} \hbar_{\log\delta}(v,v').$$

The summation above requires the knowledge of δ-neighborhood structures in a canonical $H_{\log\delta}^2$ and the distribution of $\hbar_{\log\delta}(v,v')$ for a δ-pair $(v,v') \in H_{\log\delta}^2 \times H_{\log\delta}^2$.



(a) (b)

Figure 3.9: (a) Examples of $N'_\delta(v)$ for a grid point v in the deleted lower-left half of a canonical $H^2_{\log\delta}$; (b) examples of $N''_\delta(v)$ for a grid point v in the deleted lower-right half of a canonical $H^2_{\log\delta}$. (Note: the geometrical shape identifies the δ-neighboring relation.)

For a grid point v in a canonical H^2_k with sufficient large k, its δ-neighborhood of cardinality 4δ, in the absence of any border/boundary, forms a square $((+\frac{\pi}{4})$-rotated) centered at v and composed of two main-diagonal and two auxiliary-diagonal segments (in which all grid points are at a 1-normed distance of δ from v). With $k = \log\delta$, in the presence of four boundaries of $H^2_{\log\delta}$, the "squared complete δ-neighborhood" is truncated by the boundaries. The δ-neighborhood of v is the truncated/incomplete square composed of at most two segments from adjacent main and auxiliary diagonals — which are two diagonals (one main and one auxiliary diagonals) in the canonical $H^2_{\log\delta}$. The δ-neighborhood of v degenerates to a single main or auxiliary diagonal of $H^2_{\log\delta}$ when and only when v is a corner grid point of $H^2_{\log\delta}$. Figure 3.9 depicts some example δ-neighborhoods in a canonical $H^2_{\log\delta}$.

In a canonical H^2_k, a deleted triangular half (lower-left, lower-right, etc.) is the corresponding triangular half without the main/auxiliary diagonal. In our derivation for $L_{\log\delta}(H^2_{\log\delta})$ below, for a grid point $v \in H^2_{\log\delta}$, we consider the contribution in $L_{\log\delta}(H^2_{\log\delta})$ of a δ-pair (v, v') from two possible sources: (1) Case of auxiliary diagonals: all δ-neighbors v' of v from an auxiliary diagonal of $H^2_{\log\delta}$, denoted by $N'_\delta(v)$ (see Figure 3.9 (a)), and (2) Case of main diagonals: those from a main diagonal of $H^2_{\log\delta}$, denoted by $N''_\delta(v)$ (see Figure 3.9 (b)). (As mentioned above, when v is a corner grid point of $H^2_{\log\delta}$, the sources degenerate to one diagonal.) The two derivations are similar, and we focus our discussion on the case of auxiliary diagonals. We observe the following δ-neighboring relation between two auxiliary diagonals in $H^2_{\log\delta}$.

For every grid point v in the deleted lower-left half of the canonical $H^2_{\log\delta}$, that is, $v \in A_{\log\delta,\alpha}$ for some $\alpha \in [\delta-1]$, we have $N'_\delta(v) = A'_{\log\delta,\beta}$ with $\alpha+\beta=\delta$. Similarly for every grid point v' in the deleted upper-right half of the canonical $H^2_{\log\delta}$, that is, $v' \in A'_{\log\delta,\beta}$ for some $\beta \in [\delta-1]$, we have $N'_\delta(v') = A_{\log\delta,\alpha}$ with $\alpha+\beta=\delta$.

After addressing the δ-neighborhood structures, we consider the organization of $\hbar_{\log\delta}(v,v')$ for all δ-pairs (v,v') for the summation in expanding $L_\delta(H^2_{\log\delta}) = \sum_{v,v'\in H^2_{\log\delta}|d_1(v,v')=\delta} \hbar_{\log\delta}(v,v')$ for the case of auxiliary diagonals. First we write that:

$$\hbar_{\log\delta}(v,v') = |\hbar_{\log\delta}(v,\partial_1(H^2_{\log\delta})) - \hbar_{\log\delta}(v',\partial_1(H^2_{\log\delta}))|,$$

and need to determine the algebraic sign of $\hbar_{\log\delta}(v,\partial_1(H^2_{\log\delta})) - \hbar_{\log\delta}(v',\partial_1(H^2_{\log\delta}))$.

We observe the following two cases based on the location of a grid point v in the quadrants of the canonical $H^2_{\log\delta}$.

1. For every grid point $v \in A_{\log\delta,\alpha} \cap (\cup_{\eta=1}^{3} Q_\eta(H^2_{\log\delta}))$, where $\alpha \in [\delta-1]$, we have $\hbar_{\log\delta}(v,\partial_1(H^2_{\log\delta})) < \hbar_{\log\delta}(v',\partial_1(H^2_{\log\delta}))$ for all $v' \in N'_\delta(v)$, and

2. For every grid point $v \in A_{\log\delta,\alpha} \cap Q_4(H^2_{\log\delta})$, where $\alpha \in \{\frac{\delta}{2}+1, \frac{\delta}{2}+2, \ldots, \delta-1\}$, we have $\hbar_{\log\delta}(v,\partial_1(H^2_{\log\delta})) > \hbar_{\log\delta}(v',\partial_1(H^2_{\log\delta}))$ for all $v' \in N'_\delta(v)$.

This suggests that we decompose the contribution in $L_{\log\delta}(H^2_{\log\delta})$ of all δ-pairs (v,v') from the case of auxiliary diagonals into four parts of index-cumulations:

1. The index-cumulation of all grid points v in the deleted lower-left half of $Q_4(H^2_{\log\delta})$:

$$\sum_{\alpha=\frac{\delta}{2}+1}^{\delta-1} \sum_{v\in A_{\log\delta,\alpha}\cap Q_4(H^2_{\log\delta})} (\delta-\alpha)\hbar_{\log\delta}(v,\partial_1(H^2_{\log\delta}))$$

$$= \sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v \in A_{\log\delta,\alpha+\frac{\delta}{2}} \cap Q_4(H^2_{\log\delta})} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta}(v, \partial_1(H^2_{\log\delta}))$$

$$= \sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v \in A_{\log\delta,\alpha+\frac{\delta}{2}} \cap Q_4(H^2_{\log\delta})} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta}(v, \partial_1(Q_4(H^2_{\log\delta})))$$

$$+\sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v \in A_{\log\delta,\alpha+\frac{\delta}{2}} \cap Q_4(H^2_{\log\delta})} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta}(\partial_1(Q_4(H^2_{\log\delta})), \partial_1(H^2_{\log\delta}))$$

$$= \sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v \in A'_{\log\delta-1,\alpha}} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta-1}(v, \partial_1(H^2_{\log\delta-1}))$$

(after $(-\frac{\pi}{2})$-rotating and then left-right reflecting $Q_4(H^2_{\log\delta})$

into a canonical $H^2_{\log\delta-1}$)

$$+3(\frac{\delta}{2})^2 \sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v \in A'_{\log\delta-1,\alpha}} (\frac{\delta}{2}-\alpha)$$

$$= \mathcal{A}'_{\log\delta-1,\frac{\delta}{2}-1} + 3(\frac{\delta}{2})^2 \mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}$$

(Note: $\mathcal{A}'_{\log\delta-1,\frac{\delta}{2}} = \mathcal{A}'_{\log\delta-1,\frac{\delta}{2}-1} + \overline{\mathcal{A}'}_{\log\delta-1,\frac{\delta}{2}}$)

$$= \mathcal{A}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{A}'}_{\log\delta-1,\frac{\delta}{2}} + 3(\frac{\delta}{2})^2 \mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}.$$

2. The index-cumulation of all δ-neighbors $v' \in N'_\delta(v)$ for all grid points v in the deleted lower-left half of $Q_4(H^2_{\log\delta})$:

$$\sum_{\beta=1}^{\frac{\delta}{2}-1} \sum_{v' \in A'_{\log\delta,\beta}} (\frac{\delta}{2}-\beta)\hbar_{\log\delta}(v', \partial_1(H^2_{\log\delta}))$$

$$= \mathcal{A}'_{\log\delta,\frac{\delta}{2}-1} = \mathcal{A}'_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{A}'}_{\log\delta,\frac{\delta}{2}} = \mathcal{A}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{A}'}_{\log\delta-1,\frac{\delta}{2}} + 2(\frac{\delta}{2})^2 \mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}.$$

3. The index-cumulation of all δ-neighbors $v' \in N'_\delta(v)$ for all grid points v in the deleted lower-left half of $H^2_{\log\delta}$ and not in the deleted lower-left half of $Q_4(H^2_{\log\delta})$:

$$\sum_{\beta=1}^{\delta-1} \sum_{v' \in A'_{\log\delta,\beta}} (\delta-\beta)\hbar_{\log\delta}(v', \partial_1(H^2_{\log\delta}))$$

$$-\sum_{\beta=1}^{\frac{\delta}{2}-1}\sum_{v'\in A'_{\log\delta,\beta}}(\frac{\delta}{2}-\beta)\hbar_{\log\delta}(v',\partial_1(H^2_{\log\delta}))$$

$$= \mathcal{A}'_{\log\delta,\delta-1}-\sum_{\beta=1}^{\frac{\delta}{2}-1}\sum_{v'\in A'_{\log\delta,\beta}}((\delta-\beta)\hbar_{\log\delta}(v',\partial_1(H^2_{\log\delta})))$$

(Note: $\mathcal{A}'_{\log\delta,\delta}=\mathcal{A}'_{\log\delta,\delta-1}+\overline{\mathcal{A}'}_{\log\delta,\delta}$)

$$= \mathcal{A}'_{\log\delta,\delta}-\overline{\mathcal{A}'}_{\log\delta,\delta}-(\mathcal{A}'_{\log\delta-1,\frac{\delta}{2}}-\overline{\mathcal{A}'}_{\log\delta-1,\frac{\delta}{2}}+2(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}).$$

4. The index-cumulation of all grid points v in the deleted lower-left half of $H^2_{\log\delta}$ and not in the deleted lower-left half of $Q_4(H^2_{\log\delta})$:

$$\sum_{\alpha=1}^{\delta-1}\sum_{v\in A_{\log\delta,\alpha}}(\delta-\alpha)\hbar_{\log\delta}(v,\partial_1(H^2_{\log\delta}))$$

$$-\sum_{\alpha=\frac{\delta}{2}+1}^{\delta-1}\sum_{v\in A_{\log\delta,\alpha}\cap Q_4(H^2_{\log\delta})}(\delta-\alpha)\hbar_{\log\delta}(v,\partial_1(H^2_{\log\delta}))$$

$$= \mathcal{A}_{\log\delta,\delta-1}-\sum_{\alpha=\frac{\delta}{2}+1}^{\delta-1}\sum_{v\in A_{\log\delta,\alpha}\cap Q_4(H^2_{\log\delta})}(\delta-\alpha)\hbar_{\log\delta}(v,\partial_1(H^2_{\log\delta}))$$

$$= \mathcal{A}_{\log\delta,\delta}-\overline{\mathcal{A}}_{\log\delta,\delta}-(\mathcal{A}'_{\log\delta-1,\frac{\delta}{2}}-\overline{\mathcal{A}'}_{\log\delta-1,\frac{\delta}{2}}+3(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}).$$

By combining the four index-cumulations obtained above, we have the contribution in $L_{\log\delta}(H^2_{\log\delta})$ of all δ-pairs (v,v') from the case of auxiliary diagonals:

$$\mathcal{A}'_{\log\delta,\delta}-\overline{\mathcal{A}'}_{\log\delta,\delta}-(\mathcal{A}_{\log\delta,\delta}-\overline{\mathcal{A}}_{\log\delta,\delta})$$

$$+2(\mathcal{A}'_{\log\delta-1,\frac{\delta}{2}}-\overline{\mathcal{A}'}_{\log\delta-1,\frac{\delta}{2}}+3(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1})$$

$$-2(\mathcal{A}'_{\log\delta-1,\frac{\delta}{2}}-\overline{\mathcal{A}'}_{\log\delta-1,\frac{\delta}{2}}+2(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1})$$

$$= \mathcal{A}'_{\log\delta,\delta}-\mathcal{A}_{\log\delta,\delta}-\overline{\mathcal{A}'}_{\log\delta,\delta}+\overline{\mathcal{A}}_{\log\delta,\delta}+2(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}.$$

Following a similar strategy, the contribution in $L_{\log\delta}(H^2_{\log\delta})$ of all δ-pairs (v,v') from the case of main diagonals is:

$$\mathcal{D}_{\log\delta,\delta}-\mathcal{D}'_{\log\delta,\delta}-\overline{\mathcal{D}}_{\log\delta,\delta}+\overline{\mathcal{D}'}_{\log\delta,\delta}+2(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}.$$

These two contributions are combined into the basis:

$$\begin{aligned} L_\delta(H^2_{\log\delta}) &= \mathcal{A}'_{\log\delta,\delta} - \mathcal{A}_{\log\delta,\delta} - \overline{\mathcal{A}'}_{\log\delta,\delta} + \overline{\mathcal{A}}_{\log\delta,\delta} + 2(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1} \\ &\quad +\mathcal{D}_{\log\delta,\delta} - \mathcal{D}'_{\log\delta,\delta} - \overline{\mathcal{D}}_{\log\delta,\delta} + \overline{\mathcal{D}'}_{\log\delta,\delta} + 2(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1} \\ &= \frac{37}{2^4\cdot 3\cdot 5}\delta^5 - \frac{1}{2^2\cdot 3}\delta^3 - \frac{2}{3\cdot 5}\delta. \end{aligned}$$

The recurrence in Lemma 3.4 (for arbitrary δ) with the basis give the desired closed-form solution for $L_\delta(H^2_k)$. ∎

3.1.3 Derivation of 2-Dimensional z-Order Curve Family

For a 2-dimensional z-order curve Z^2_k indexing the grid $[2^k]^2$, with a canonical orientation shown in Figure 2.3(a), we denote by $\partial_1(Z^2_k)$ and $\partial_2(Z^2_k)$ the entry and exit, respectively, grid point in $[2^k]^2$ (with respect to the canonical orientation). Figure 2.3(b) depicts the decomposition of Z^2_k and the ∂_1- and ∂_2-labels of four Z^2_{k-1}-subcurves. In this section, we are going to derive the exact formula for $L_\delta(Z^2_k)$ as follows (similar to the derivations of $L_\delta(H^2_k)$). The recursive decomposition (in k) of Z^2_k is

$$\begin{aligned} L_\delta(Z^2_k) &= \sum_{i,j\in[2^{2k}]|i<j \text{ and } d_1(Z^2_k(i),Z^2_k(j))=\delta} |i-j| \\ &= 4L_\delta(Z^2_{k-1}) + \sum_{\alpha,\beta\in\{1,2,3,4\}|\alpha<\beta} \Delta_\delta(Q_\alpha(Z^2_k), Q_\beta(Z^2_k)). \end{aligned}$$

Note, the notations introduced in Section 3.1.1 are for Z^2_k in this section.

We study the following three lemmas for the cumulation of indices of grid points in the row, column, diagonal, and auxiliary-diagonal structures of Z^2_k.

Lemma 3.7 *The binary representation of the row-number of a row structure of Z^2_k helps compute its index-cumulation as follows:*

1. *For all $\alpha \in [2^{k-1}]$, a recurrence for $\Delta(R_{k,\alpha}, \partial_1(Z^2_k))$ is:*

$$\Delta(R_{k,\alpha}, \partial_1(Z^2_k)) = 2\Delta(R_{k-1,\alpha}, \partial_1(Z^2_{k-1})) + \frac{1}{2^2}\cdot 2^{3k}.$$

For $\alpha = 1$, a closed-form solution for $\Delta(R_{k,1}, \partial_1(Z_k^2))$ from the recurrence above (in k) is:

$$\frac{1}{3} \cdot 2^{3k} - \frac{1}{3} \cdot 2^k.$$

For those α satisfying $2^{q-1} < \alpha \leq 2^q$ for some integer $q \in [k-1]$ (that is, $q = \lceil \log \alpha \rceil$), the recurrence above (in k) yields a recurrence:

$$\Delta(R_{k,\alpha}, \partial_1(Z_k^2)) = 2^{k-q}\Delta(R_{q,\alpha}, \partial_1(Z_q^2)) + \sum_{\eta=q+1}^{k} 2^{k-\eta}(\frac{1}{2^2} \cdot 2^{3\eta}),$$

where the summation is $\frac{1}{3}(2^{3k} - 2^{k+2q}) = \frac{1}{3} \cdot 2^{3k} - \frac{1}{3} \cdot 2^{k+2\lceil \log \alpha \rceil}$.

2. *For all $\alpha, \beta \in [2^k]$ such that $\alpha < \beta$ and the binary representations of $\alpha - 1$ and $\beta - 1$ differ only at the i-th low-order bit, where $i \in \{0, 1, \ldots, k-1\}$ (that is, $(\alpha - 1) \oplus (\beta - 1) = 2^i$, where $\oplus$ denotes the binary exclusive-or operator),*

$$\Delta(R_{k,\beta}, \partial_1(Z_k^2)) = \Delta(R_{k,\alpha}, \partial_1(Z_k^2)) + 2^{k+2i}.$$

Proof. Consider a canonical Z_k^2.

Part 1: We construct the general recurrence (in k) for $\Delta(R_{k,\alpha}, \partial_1(Z_k^2))$ for arbitrary $\alpha \in [2^{k-1}]$ as follows. The row $R_{k,\alpha}$ is in the lower-half of the canonical Z_k^2, and can be decomposed into the α-th row of $Q_1(Z_k^2)$ (a canonical Z_{k-1}^2) and the α-th row of $Q_3(Z_k^2)$ (a canonical Z_{k-1}^2). By noting that for all grid points $v \in Q_3(Z_k^2)$,

$$\begin{aligned} \hbar_k(v, \partial_1(Z_k^2)) &= \hbar_k(v, \partial_1(Q_3(Z_k^2))) + \hbar_k(\partial_1(Q_3(Z_k^2)), \partial_1(Z_k^2)) \\ &= \hbar_k(v, \partial_1(Q_3(Z_k^2))) + 2 \cdot 2^{2(k-1)}, \end{aligned}$$

and translating the index-cumulation of $R_{k,\alpha}$ (in Z_k^2) in the two Z_{k-1}^2-subcurves, we have:

$$\begin{aligned} & \Delta(R_{k,\alpha}, \partial_1(Z_k^2)) \\ = & \sum_{v \in R_{k,\alpha}} \hbar_k(v, \partial_1(Z_k^2)) \\ = & (\Delta(R_{k,\alpha} \cap Q_1(Z_k^2), \partial_1(Q_1(Z_k^2))) + 2^{k-1}\hbar_k(\partial_1(Q_1(Z_k^2)), \partial_1(Z_k^2))) \end{aligned}$$

$$
\begin{aligned}
& +(\Delta(R_{k,\alpha} \cap Q_3(Z_k^2), \partial_1(Q_3(Z_k^2))) + 2^{k-1}\hbar_k(\partial_1(Q_3(Z_k^2)), \partial_1(Z_k^2))) \\
= \; & (\Delta(R_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + 2^{k-1} \cdot 0) \\
& +(\Delta(R_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + 2^{k-1} \cdot 2 \cdot 2^{2(k-1)}) \\
= \; & 2\Delta(R_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + \frac{1}{2^2} \cdot 2^{3k}.
\end{aligned}
$$

Now we iterate the general recurrence (in k) from order k to order $q+1$ (where $q = \lceil \log \alpha \rceil$), which yields the recurrence (in k) for $\Delta(R_{k,\alpha}, \partial_1(Z_k^2))$:

$$
\begin{aligned}
& \Delta(R_{k,\alpha}, \partial_1(Z_k^2)) = 2\Delta(R_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + \frac{1}{2^2} \cdot 2^{3k} \\
= \; & 2\Delta(R_{q,\alpha}, \partial_1(Z_q^2)) + \sum_{\eta=q+1}^{k} 2^{k-\eta}(\frac{1}{2^2} \cdot 2^{3\eta}) = 2\Delta(R_{q,\alpha}, \partial_1(Z_q^2)) + \frac{1}{3}(2^{3k} - 2^{k+2q}) \\
= \; & 2\Delta(R_{q,\alpha}, \partial_1(Z_q^2)) + \frac{1}{3} \cdot 2^{3k} - \frac{1}{3} \cdot 2^{k+2\lceil \log \alpha \rceil}.
\end{aligned}
$$

When $\alpha = 1$, the general recurrence yields a recurrence (in k):

$$
\Delta(R_{k,1}, \partial_1(Z_k^2)) = 2\Delta(R_{1,1}, \partial_1(Z_1^2)) + \sum_{\eta=2}^{k}(2^{k-\eta} \cdot \frac{1}{2^2} \cdot 2^{3\eta}) = \frac{1}{3} \cdot 2^{3k} - \frac{1}{3} \cdot 2^k.
$$

Part 2: A z-order curve is a bit-interleave curve: the index of a grid point is assigned with the number that interleaves the bits of its coordinates (by starting the first bit of y-coordinate (axis-1)). Let $\alpha, \beta \in [2^k]$ such that $\alpha < \beta$ and $(\alpha - 1) \oplus (\beta - 1) = 2^i$, where $i \in \{0, 1, \ldots, k-1\}$. Consider the two rows $R_{k,\alpha}$ and $R_{k,\beta}$ of the canonical Z_k^2, which are 2^i rows apart. For points $v \in R_{k,\alpha}$, $v' \in R_{k,\beta}$ and v, v' are in the same column, the index difference of these two points is 2^{2i}. The reason is that the index in binary format for a point is the binary digits of y-coordinate interleaved with that of x-coordinate, and the i-th binary digit of y-coordinate becomes the $2i$-th binary digit of the index. For a row, there are 2^k points, so the cumulation of index differences between the points in $R_{k,\alpha}$ and $R_{k,\beta}$ is $2^k \cdot 2^{2i}$; that is $\Delta(R_{k,\beta}, \partial_1(Z_k^2)) = \Delta(R_{k,\alpha}, \partial_1(Z_k^2)) + 2^{k+2i}$. ∎

Lemma 3.8 *The binary representation of the column-number of a column structure of Z_k^2 helps compute its index-cumulation as follows:*

1. *For all* $\alpha \in [2^{k-1}]$*, a recurrence for* $\Delta(C_{k,\alpha}, \partial_1(Z_k^2))$ *is:*

$$\Delta(C_{k,\alpha}, \partial_1(Z_k^2)) = 2\Delta(C_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + \frac{1}{2^3} \cdot 2^{3k}.$$

For $\alpha = 1$*, a closed-form solution for* $\Delta(C_{k,1}, \partial_1(Z_k^2))$ *from the recurrence above (in* k*) is:*

$$\frac{1}{2 \cdot 3} \cdot 2^{3k} - \frac{1}{2 \cdot 3} \cdot 2^k.$$

For those α *satisfying* $2^{q-1} < \alpha \leq 2^q$ *for some integer* $q \in [k-1]$ *(that is,* $q = \lceil \log \alpha \rceil$*), the recurrence above (in* k*) yields a recurrence:*

$$\Delta(C_{k,\alpha}, \partial_1(Z_k^2)) = 2^{k-q}\Delta(C_{q,\alpha}, \partial_1(Z_q^2)) + \sum_{\eta=q+1}^{k} 2^{k-\eta}(\frac{1}{2^3} \cdot 2^{3\eta}),$$

where the summation is $\frac{1}{2 \cdot 3}(2^{3k} - 2^{k+2q}) = \frac{1}{2 \cdot 3} \cdot 2^{3k} - \frac{1}{2 \cdot 3} \cdot 2^{k+2\lceil \log \alpha \rceil}$.

2. *For all* $\alpha, \beta \in [2^k]$ *such that* $\alpha < \beta$ *and the binary representations of* $\alpha - 1$ *and* $\beta - 1$ *differ only at the* i*-th low-order bit, where* $i \in \{0, 1, \ldots, k-1\}$ *(that is,* $(\alpha - 1) \oplus (\beta - 1) = 2^i$*, where* $\oplus$ *denotes the binary exclusive-or operator),*

$$\Delta(C_{k,\beta}, \partial_1(Z_k^2)) = \Delta(C_{k,\alpha}, \partial_1(Z_k^2)) + 2 \cdot 2^{k+2i}.$$

Proof. Consider a canonical Z_k^2.

Part 1: We construct the general recurrence (in k) for $\Delta(C_{k,\alpha}, \partial_1(Z_k^2))$ for arbitrary $\alpha \in [2^{k-1}]$ as follows. The column $C_{k,\alpha}$ is in the left-half of the canonical Z_k^2, and can be decomposed into the α-th column of $Q_1(Z_k^2)$ (a canonical Z_{k-1}^2) and the α-th column of $Q_2(Z_k^2)$ (a canonical Z_{k-1}^2). By noting that for all grid points $v \in Q_2(Z_k^2)$,

$$\hbar_k(v, \partial_1(Z_k^2)) = \hbar_k(v, \partial_1(Q_2(Z_k^2))) + \hbar_k(\partial_1(Q_2(Z_k^2)), \partial_1(Z_k^2)) = \hbar_k(v, \partial_1(Q_2(Z_k^2))) + 2^{2(k-1)},$$

and translating the index-cumulation of $C_{k,\alpha}$ (in Z_k^2) in the two Z_{k-1}^2-subcurves, we have:

$$\Delta(C_{k,\alpha}, \partial_1(Z_k^2)) = \sum_{v \in C_{k,\alpha}} \hbar_k(v, \partial_1(Z_k^2))$$
$$= (\Delta(C_{k,\alpha} \cap Q_1(Z_k^2), \partial_1(Q_1(Z_k^2))) + 2^{k-1}\hbar_k(\partial_1(Q_1(Z_k^2)), \partial_1(Z_k^2)))$$

$$
\begin{aligned}
&+(\Delta(C_{k,\alpha} \cap Q_2(Z_k^2), \partial_1(Q_2(Z_k^2))) + 2^{k-1}\hbar_k(\partial_1(Q_2(Z_k^2)), \partial_1(Z_k^2))) \\
= &\ (\Delta(C_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + 2^{k-1} \cdot 0) + (\Delta(C_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + 2^{k-1} \cdot 2^{2(k-1)}) \\
= &\ 2\Delta(C_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + \frac{1}{2^3} \cdot 2^{3k}.
\end{aligned}
$$

Now we iterate the general recurrence (in k) from order k to order $q+1$ (where $q = \lceil \log \alpha \rceil$), which yields the recurrence (in k) for $\Delta(C_{k,\alpha}, \partial_1(Z_k^2))$:

$$
\begin{aligned}
\Delta(C_{k,\alpha}, \partial_1(Z_k^2)) &= 2\Delta(C_{k-1,\alpha}, \partial_1(Z_{k-1}^2)) + \frac{1}{2^3} \cdot 2^{3k} \\
&= 2\Delta(C_{q,\alpha}, \partial_1(Z_q^2)) + \sum_{\eta=q+1}^{k} 2^{k-\eta}(\frac{1}{2^3} \cdot 2^{3\eta}) \\
&= 2\Delta(C_{q,\alpha}, \partial_1(Z_q^2)) + \frac{1}{2 \cdot 3}(2^{3k} - 2^{k+2q}) \\
&= 2\Delta(C_{q,\alpha}, \partial_1(Z_q^2)) + \frac{1}{2 \cdot 3} \cdot 2^{3k} - \frac{1}{2 \cdot 3} \cdot 2^{k+2\lceil \log \alpha \rceil}.
\end{aligned}
$$

When $\alpha = 1$, the general recurrence yields a recurrence (in k):

$$
\Delta(C_{k,1}, \partial_1(Z_k^2)) = 2\Delta(C_{1,1}, \partial_1(Z_1^2)) + \sum_{\eta=2}^{k}(2^{k-\eta} \cdot \frac{1}{2^3} \cdot 2^{3\eta}) = \frac{1}{2 \cdot 3} \cdot 2^{3k} - \frac{1}{2 \cdot 3} \cdot 2^k.
$$

Part 2: Similar to the proof in Lemma 3.7 (part 2), Let $\alpha, \beta \in [2^k]$ such that $\alpha < \beta$ and $(\alpha - 1) \oplus (\beta - 1) = 2^i$, where $i \in \{0, 1, \ldots, k-1\}$. Consider the two columns $C_{k,\alpha}$ and $C_{k,\beta}$ of the canonical Z_k^2, which are 2^i columns apart. For points $v \in C_{k,\alpha}$ and $v' \in C_{k,\beta}$ and v, v' are in the same row, the index difference of these two points is 2^{2i+1}. The reason is that the index in binary format for a point is the binary digits of y-coordinate interleaved with that of x-coordinate and started by y-coordinate. Therefore, i-th binary digit of x-coordinate becomes the $2i+1$-st binary digit of the index. For a column, there are 2^k points, so the cumulation of index differences between the points in $C_{k,\alpha}$ and $C_{k,\beta}$ is $2^k \cdot 2^{2i+1}$; that is $\Delta(C_{k,\beta}, \partial_1(Z_k^2)) = \Delta(C_{k,\alpha}, \partial_1(Z_k^2)) + 2 \cdot 2^{k+2i}$. ∎

Lemma 3.9 *For $k \geq 1$,*

$\Delta(A_{k,2^k}, \partial_1(Z_k^2)) = \Delta(D_{k,2^k}, \partial_1(Z_k^2)) = \frac{1}{2} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k.$

Proof. A canonical Z_k^2 can be $(+\pi)$-rotated (left-right reflected and then top-down reflected) to get a same Z_k^2 structure with indices reversed. Thus, for a grid point $v \in R_{k,\alpha} \cap C_{k,\alpha}$, where $\alpha \in [2^k]$, its mirror point $v' \in R_{k,2^k+1-\alpha} \cap C_{k,2^k+1-\alpha}$, and the mirror pair (v, v') satisfies that:

$$\hbar_k(v, \partial_1(Z_k^2)) + \hbar_k(v', \partial_1(Z_k^2)) = \hbar_k(v, \partial_2(Z_k^2)) + \hbar_k(v', \partial_2(Z_k^2)) = 2^{2k} - 1.$$

In addition, $D_{k,2^k} = \{v | v \in R_{k,\alpha} \cap C_{k,\alpha} \text{ where } \alpha \in [2^k]\} = (D_{k,2^k} \cap Q_1(Z_k^2)) \cup (D_{k,2^k} \cap Q_4(Z_k^2))$, and there are 2^{k-1} mirror pairs (v, v') in the main diagonal $D_{k,2^k}$ (let $v \in (D_{k,2^k} \cap Q_1(Z_k^2))$ and $v' \in (D_{k,2^k} \cap Q_4(Z_k^2))$). Thus,

$$\begin{aligned} \Delta(D_{k,2^k}, \partial_1(Z_k^2)) &= \sum_{v \in D_{k,2^k}} \hbar_k(v, \partial_1(Z_k^2)) \\ (&= \sum_{v' \in D_{k,2^k}} \hbar_k(v', \partial_2(Z_k^2)) = \Delta(D_{k,2^k}, \partial_2(Z_k^2)) \quad (v' \text{ is the mirror point of } v)) \\ &= \sum_{\text{all the mirror pairs } (v,v') \text{ in } D_{k,2^k}} (\hbar_k(v, \partial_1(Z_k^2)) + \hbar_k(v', \partial_1(Z_k^2))) \\ &= 2^{k-1}(2^{2k} - 1) = \frac{1}{2} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k. \end{aligned}$$

Similarly, for a grid point $v \in R_{k,\alpha} \cap C_{k,2^k+1-\alpha}$, where $\alpha \in [2^k]$, its mirror point $v' \in R_{k,2^k+1-\alpha} \cap C_{k,\alpha}$, and $A_{k,2^k} = \{v | v \in R_{k,\alpha} \cap C_{k,2^k+1-\alpha} \text{ where } \alpha \in [2^k]\} = (A_{k,2^k} \cap Q_2(Z_k^2)) \cup (A_{k,2^k} \cap Q_3(Z_k^2))$,

$$\begin{aligned} \Delta(A_{k,2^k}, \partial_1(Z_k^2)) &= \sum_{v \in A_{k,2^k}} \hbar_k(v, \partial_1(Z_k^2)) \\ (&= \sum_{v' \in A_{k,2^k}} \hbar_k(v', \partial_2(Z_k^2)) = \Delta(A_{k,2^k}, \partial_2(Z_k^2)) \quad (v' \text{ is the mirror point of } v)) \\ &= \sum_{\text{all mirror pairs } (v,v') \text{ in } A_{k,2^k}} (\hbar_k(v, \partial_1(Z_k^2)) + \hbar_k(v', \partial_1(Z_k^2))) \\ &= 2^{k-1}(2^{2k} - 1) = \frac{1}{2} \cdot 2^{3k} - \frac{1}{2} \cdot 2^k. \end{aligned}$$

∎

Now we partition the summation $\sum_{\alpha,\beta \in \{1,2,3,4\} | \alpha < \beta} \Delta_\delta(Q_\alpha(Z_k^2), Q_\beta(Z_k^2))$ according to the two cases: $(\alpha, \beta) \in \{(1,2), (1,3), (2,4), (3,4)\}$ (four subcases of contiguous curves), and $(\alpha, \beta) \in \{(1,4), (2,3)\}$ (two subcases of diagonal subcurves). The summations resulting from restricting to these four subcases are investigated in the following lemma.

Lemma 3.10 *For δ that is a positive integral power of 2, and $1 \leq \delta < 2^k$,*

$$\begin{aligned}\Delta_\delta(Q_1(Z_k^2), Q_4(Z_k^2)) &= \Delta_\delta(Q_2(Z_k^2), Q_3(Z_k^2)) \\ &= 2(\overline{\mathcal{C}}_{k-1,\delta} + 2\mathcal{C}_{k-1,\delta-1} - \mathcal{D}'_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1}) \\ &\quad +(\frac{1}{2^2} \cdot 2^{2k} + 1)(\frac{1}{2} \cdot 2^k\delta^2 - 2\mathcal{N}_{k-1,\delta-1}),\end{aligned}$$

$$\begin{aligned}\Delta_\delta(Q_1(Z_k^2), Q_2(Z_k^2)) &= \Delta_\delta(Q_3(Z_k^2), Q_4(Z_k^2)) \\ &= 2(\overline{\mathcal{R}}_{k-1,\delta} + 2\mathcal{R}_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1} - \mathcal{D}_{k-1,\delta-1}) \\ &\quad +(\frac{1}{2} \cdot 2^k\delta^2 - 2\mathcal{N}_{k-1,\delta-1}),\end{aligned}$$

$$\Delta_\delta(Q_1(Z_k^2), Q_3(Z_k^2)) = (\frac{1}{2} \cdot 2^{2k} + 1)\mathcal{N}_{k-1,\delta-1} + 2\mathcal{A}_{k-1,\delta-1},$$

and

$$\Delta_\delta(Q_2(Z_k^2), Q_4(Z_k^2)) = \mathcal{N}_{k-1,\delta-1} + 2\mathcal{D}'_{k-1,\delta-1}.$$

Proof. Similar to the proof in Lemma 3.4 for Hilbert curve family (from the illustration in Section 3.1.1.2 and Equations 3.2, 3.3 and 3.4) except that z-order curve has no rotation or reflection for the subcurves and it has different distances between subcurves. We expand $\Delta_\delta(Q_\alpha(H_k^2), Q_\beta(H_k^2))$ into three cumulations of index-differences:

$$\begin{aligned}\Delta_\delta(Q_\alpha(Z_k^2), Q_\beta(Z_k^2)) &= \sum_{(v,v')\in Q_\alpha(Z_k^2)\times Q_\beta(Z_k^2)|d_1(v,v')=\delta} \hbar_k(v,v') \\ = &\sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(v, \partial_2(Q_\alpha(Z_k^2))) \\ &+ \sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(\partial_2(Q_\alpha(Z_k^2)), \partial_1(Q_\beta(Z_k^2))) \\ &+ \sum_{(v,v')\in Q_{\alpha,\delta}\times Q_{\beta,\delta}|d_1(v,v')=\delta} \hbar_k(v', \partial_1(Q_\beta(Z_k^2))).\end{aligned}$$

First we derive $\Delta_\delta(Q_\alpha(Z_k^2), Q_\beta(Z_k^2))$ for the case of contiguous subcurves ($(\alpha,\beta) \in \{(1,2),(1,3),(2,4),(3,4)\}$). We compute the three cumulations of index-differences from Equations 3.2, 3.3, and 3.4 by the derivations in Section 3.1.1.2. (All the notations for C_k^2 in Section 3.1.1.2 applies for Z_k^2 in this section.)

Consider the case when $(\alpha,\beta) = (1,3)$. Because the subcurves $Q_1(Z_k^2)$ and $Q_3(Z_k^2)$ are all canonical z-order curves Z_{k-1}^2, the structures c_η^4, $X_{k-1,\eta}^{D'}$, and $X_{k-1,\eta}^{A}$ for $\sum_{(v,v')\in Q_{1,\delta}\times Q_{3,\delta}|d_1(v,v')=\delta} \hbar_k(v',\partial_1(Q_3(Z_k^2)))$ are transformed into $C_{k-1,\eta}$, $D'_{k-1,\eta}$, and $A_{k-1,\eta}$, respectively, for all $\eta \in [2^{k-1}]$, and $\partial_1(Q_3(Z_k^2))$ into $\partial_1(Z_{k-1}^2)$. From Equation 3.2,

$$\begin{aligned}
&\sum_{(v,v')\in Q_{1,\delta}\times Q_{3,\delta}|d_1(v,v')=\delta} \hbar_k(v',\partial_1(Q_3(Z_k^2)))\\
=\ & \sum_{j=1}^{\delta}\sum_{u\in C_{k-1,j}} (2(\delta+1-j)-1)\hbar_{k-1}(u,\partial_1(Z_{k-1}^2))\\
&-\sum_{\eta=1}^{\delta}\sum_{u\in D'_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u,\partial_1(Z_{k-1}^2)) - \sum_{\eta=1}^{\delta}\sum_{u\in A_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u,\partial_1(Z_{k-1}^2))\\
=\ & 2\sum_{j=1}^{\delta-1}\sum_{u\in C_{k-1,j}} (\delta-j)\hbar_{k-1}(u,\partial_1(Z_{k-1}^2)) + \sum_{j=1}^{\delta}\sum_{u\in C_{k-1,j}} \hbar_{k-1}(u,\partial_1(Z_{k-1}^2))\\
&-\mathcal{D}'_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1}\\
=\ & 2\mathcal{C}_{k-1,\delta-1} + \overline{\mathcal{C}}_{k-1,\delta} - \mathcal{D}'_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1}.
\end{aligned}$$

We may compute $\sum_{(v,v')\in Q_{1,\delta}\times Q_{3,\delta}|d_1(v,v')=\delta} \hbar_k(v,\partial_2(Q_1(Z_k^2)))$ according to the obtained expansion for $(\alpha,\beta) = (1,3)$. The reflection symmetry (left-right and then top-down reflection) gives that

$$\sum_{(v,v')\in Q_{1,\delta}\times Q_{3,\delta}|d_1(v,v')=\delta} \hbar_k(v,\partial_2(Q_1(Z_k^2))) = \sum_{(v,v')\in Q_{1,\delta}\times Q_{3,\delta}|d_1(v,v')=\delta} \hbar_k(v',\partial_1(Q_4(Z_k^2))).$$

Since $\hbar_k(\partial_2(Q_1(Z_k^2)),\partial_1(Q_3(Z_k^2))) = 2^{2(k-1)}+1$, we have:

$$\begin{aligned}
&\sum_{(v,v')\in Q_{1,\delta}\times Q_{3,\delta}|d_1(v,v')=\delta} \hbar_k(\partial_2(Q_1(Z_k^2)),\partial_1(Q_3(Z_k^2)))\\
=\ & (2^{k-1}\delta^2 - 2\mathcal{N}_{k-1,\delta-1})\hbar_k(\partial_2(Q_1(Z_k^2)),\partial_1(Q_3(Z_k^2)))\\
=\ & (2^{k-1}\delta^2 - 2\mathcal{N}_{k-1,\delta-1})(2^{2(k-1)}+1).
\end{aligned}$$

Summing up the three expansions, we have:

$$\Delta_\delta(Q_1(Z_k^2),Q_3(Z_k^2))$$

$$= \sum_{(v,v')\in Q_{1,\delta}\times Q_{3,\delta}|d_1(v,v')=\delta} \hbar_k(v', \partial_1(Q_3(Z_k^2))) + \sum_{(v,v')\in Q_{1,\delta}\times Q_{3,\delta}|d_1(v,v')=\delta} \hbar_k(v, \partial_2(Q_1(Z_k^2)))$$
$$+(2^{k-1}\delta^2 - 2\mathcal{N}_{k-1,\delta-1})(2^{2(k-1)}+1)$$
$$= 2(2\mathcal{C}_{k-1,\delta-1} + \overline{\mathcal{C}}_{k-1,\delta} - \mathcal{D}'_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1}) + (2^{k-1}\delta^2 - 2\mathcal{N}_{k-1,\delta-1})((2^{k-1})^2+1)$$
$$= 2(\overline{\mathcal{C}}_{k-1,\delta} + 2\mathcal{C}_{k-1,\delta-1} - \mathcal{D}'_{k-1,\delta-1} - \mathcal{A}_{k-1,\delta-1}) + (\frac{1}{2^2}\cdot 2^{2k}+1)(\frac{1}{2}\cdot 2^k\delta^2 - 2\mathcal{N}_{k-1,\delta-1}).$$

The derivations for the other three cases: $(\alpha,\beta) = (1,2)$, $(2,4)$, or $(3,4)$ are similar to this one.

For the case of diagonal subcurves, we first consider the main-diagonal subcurves $\Delta_\delta(Q_1(Z_k^2), Q_4(Z_k^2))$. Because the subcurves $Q_1(Z_k^2)$ and $Q_3(Z_k^2)$ are all canonical z-order curves Z_{k-1}^2, the structures $Y_{k-1,\eta}^{A'}$ and $X_{k-1,\eta}^{A}$ are transformed into $A'_{k-1,\eta}$ and $A_{k-1,\eta}$, respectively, for all $\eta \in [2^{k-1}]$, and $\partial_1(Q_4(Z_k^2))$ into $\partial_1(Z_{k-1}^2)$. According to Equation 3.5,

$$\Delta_\delta(Q_1(Z_k^2), Q_4(Z_k^2))$$
$$= \sum_{\eta=1}^{\delta-1}\sum_{u\in Y_{k-1,\eta}^{A'}} (\delta-\eta)\hbar_k(u, \partial_2(Q_1(Z_k^2)))$$
$$+\sum_{\eta=1}^{\delta-1}\sum_{u\in X_{k-1,\eta}^{A}} (\delta-\eta)\hbar_k(u, \partial_1(Q_4(Z_k^2))) + \mathcal{N}_{k-1,\delta-1}\hbar_k(\partial_2(Q_1(Z_k^2)), \partial_1(Q_4(Z_k^2)))$$
$$= \sum_{\eta=1}^{\delta-1}\sum_{u\in A_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_1(Z_{k-1}^2))$$
$$+\sum_{\eta=1}^{\delta-1}\sum_{u\in A'_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_2(Z_{k-1}^2)) + (2\cdot 2^{2(k-1)}+1)\mathcal{N}_{k-1,\delta-1}$$
$$= \sum_{\eta=1}^{\delta-1}\sum_{u\in A_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_1(Z_{k-1}^2))$$
$$+\sum_{\eta=1}^{\delta-1}\sum_{u\in A_{k-1,\eta}} (\delta-\eta)\hbar_{k-1}(u, \partial_1(Z_{k-1}^2))$$
(after $(+\pi)$-rotating $Q_1(Z_k^2)$ into a canonical Z_{k-1}^2)
$$+(2\cdot 2^{2(k-1)}+1)\mathcal{N}_{k-1,\delta-1}$$
$$= (\frac{1}{2}\cdot 2^{2k}+1)\mathcal{N}_{k-1,\delta-1} + 2\mathcal{A}_{k-1,\delta-1}.$$

The derivation for $\Delta_\delta(Q_2(Z_k^2), Q_3(Z_k^2))$ is similar. ∎

The following two lemmas allow us to simplify the overall summation of $L_\delta(Z_k^2)$.

Lemma 3.11 *For all integers q with $0 \leq q \leq k$,*

1. $\overline{\mathcal{W}}_k = \frac{1}{2} \cdot 2^{4k} - \frac{1}{2} \cdot 2^{2k}$,
2. $\overline{\mathcal{N}}_{k,2^q} = \frac{1}{2} \cdot 2^{2q} + \frac{1}{2} \cdot 2^q$,
3. $\overline{\mathcal{R}}_{k,2^q} = 2^q \Delta(R_{k,1}, \partial_1(Z_k^2)) + \frac{1}{2 \cdot 3}(2^{k+3q} - 2^{k+q})$,
4. $\overline{\mathcal{C}}_{k,2^q} = 2^q \Delta(C_{k,1}, \partial_1(Z_k^2)) + \frac{1}{3}(2^{k+3q} - 2^{k+q})$,
5. $\overline{\mathcal{A}}_{k,2^k} + \overline{\mathcal{A}'}_{k,2^k} = \overline{\mathcal{W}}_k + \Delta(A_{k,2^k}, \partial_1(Z_k^2))$,
6. $\overline{\mathcal{A}}_{k,2^k} = \frac{1}{7} \cdot 2^{4k} + \frac{1}{2^2} \cdot 2^{3k} - \frac{1}{2^2} \cdot 2^{2k} - \frac{1}{7} \cdot 2^k$,
7. $\overline{\mathcal{D}}_{k,2^k} + \overline{\mathcal{D}'}_{k,2^k} = \overline{\mathcal{W}}_k + \Delta(D_{k,2^k}, \partial_1(Z_k^2))$,
8. $\overline{\mathcal{D}'}_{k,2^k} = \frac{3}{2 \cdot 7} \cdot 2^{4k} + \frac{1}{2^2} \cdot 2^{3k} - \frac{1}{2^2} \cdot 2^{2k} - \frac{3}{2 \cdot 7} \cdot 2^k$, *and*
9. $\overline{\mathcal{D}'}_{k,2^q} = \frac{1}{3}(2^{2k} - 2^{2q})\overline{\mathcal{N}}_{k,2^q} + \overline{\mathcal{D}'}_{q,2^q}$.

Proof. Let q be an arbitrary integer with $0 \leq q \leq k$.
Part 1: $\overline{\mathcal{W}}_k = \sum_{v \in Z_k^2} \hbar_k(v, \partial_1(Z_k^2)) = \sum_{\eta \in [2^{2k}]}(\eta - 1) = \frac{1}{2} \cdot 2^{4k} - \frac{1}{2} \cdot 2^{2k}$.
Part 2: For the symbol X denoting D, D', A, or A' (main or auxiliary diagonals), the number of grid points in $X_{k,\beta}$, where $\beta \in [2^k]$, is β. Therefore,

$$\overline{\mathcal{N}}_{k,2^q} = \sum_{\beta=1}^{2^q} \sum_{v \in X_{k,\beta}} 1 = \sum_{\beta=1}^{2^q} \beta = \frac{1}{2} \cdot 2^{2q} + \frac{1}{2} \cdot 2^q.$$

Part 3: Lemma 3.7 (part 2) says that for all $\alpha, \beta \in [2^k]$ with $\alpha < \beta$ and $(\alpha - 1) \oplus (\beta - 1) = 2^i$, where $i \in \{0, 1, \ldots, k-1\}$, $\Delta(R_{k,\beta}, \partial_1(Z_k^2)) = \Delta(R_{k,\alpha}, \partial_1(Z_k^2)) + 2^{k+2i}$; hence for all $\eta \in [2^{q-1}]$,

$$\Delta(R_{k,2^{q-1}+\eta}, \partial_1(Z_k^2)) = \Delta(R_{k,\eta}, \partial_1(Z_k^2)) + 2^{k+2(q-1)}.$$

We establish a recurrence (in q) for $R_{k,2^q}$ as follows:

$$\begin{aligned}
\overline{\mathcal{R}}_{k,2^q} &= \sum_{\eta=1}^{2^q} \Delta(R_{k,\eta}, \partial_1(Z_k^2)) \\
&= \sum_{\eta=1}^{2^{q-1}} \Delta(R_{k,\eta}, \partial_1(Z_k^2)) + \sum_{\eta=2^{q-1}+1}^{2^q} \Delta(R_{k,\eta}, \partial_1(Z_k^2)) \\
&= \overline{\mathcal{R}}_{k,2^{q-1}} + \sum_{\eta=2^{q-1}+1}^{2^q} \Delta(R_{k,\eta}, \partial_1(Z_k^2)) \\
&= \overline{\mathcal{R}}_{k,2^{q-1}} + \sum_{\eta=1}^{2^{q-1}} \Delta(R_{k,2^{q-1}+\eta}, \partial_1(Z_k^2)) \\
&= \overline{\mathcal{R}}_{k,2^{q-1}} + \sum_{\eta=1}^{2^{q-1}} (\Delta(R_{k,\eta}, \partial_1(Z_k^2)) + 2^{k+2(q-1)}) \\
&= 2\overline{\mathcal{R}}_{k,2^{q-1}} + 2^{q-1} \cdot 2^{k+2(q-1)} \\
&= 2\overline{\mathcal{R}}_{k,2^{q-1}} + 2^{k+3q-3}.
\end{aligned}$$

Iterating the recurrence in descending q (to 0), we have:

$$\overline{\mathcal{R}}_{k,2^q} = 2^q\overline{\mathcal{R}}_{k,2^0} + \sum_{\eta=1}^{q} 2^{q-\eta} \cdot 2^{k+3\eta-3} = 2^q\Delta(R_{k,1}, \partial_1(Z_k^2)) + \frac{1}{2\cdot 3}(2^{k+3q} - 2^{k+q}).$$

Part 4: Lemma 3.8 (part 2) says that for all $\alpha, \beta \in [2^k]$ with $\alpha < \beta$ and $(\alpha-1) \oplus (\beta-1) = 2^i$, where $i \in \{0, 1, \ldots, k-1\}$, $\Delta(C_{k,\beta}, \partial_1(Z_k^2)) = \Delta(C_{k,\alpha}, \partial_1(Z_k^2)) + 2 \cdot 2^{k+2i}$; hence for all $\eta \in [2^{q-1}]$,

$$\Delta(C_{k,2^{q-1}+\eta}, \partial_1(Z_k^2)) = \Delta(C_{k,\eta}, \partial_1(Z_k^2)) + 2 \cdot 2^{k+2(q-1)}.$$

We establish a recurrence (in q) for $C_{k,2^q}$ as follows:

$$\begin{aligned}
\overline{\mathcal{C}}_{k,2^q} &= \sum_{\eta=1}^{2^q} \Delta(C_{k,\eta}, \partial_1(Z_k^2)) \\
&= \sum_{\eta=1}^{2^{q-1}} \Delta(C_{k,\eta}, \partial_1(Z_k^2)) + \sum_{\eta=2^{q-1}+1}^{2^q} \Delta(C_{k,\eta}, \partial_1(Z_k^2)) \\
&= \overline{\mathcal{C}}_{k,2^{q-1}} + \sum_{\eta=2^{q-1}+1}^{2^q} \Delta(C_{k,\eta}, \partial_1(Z_k^2))
\end{aligned}$$

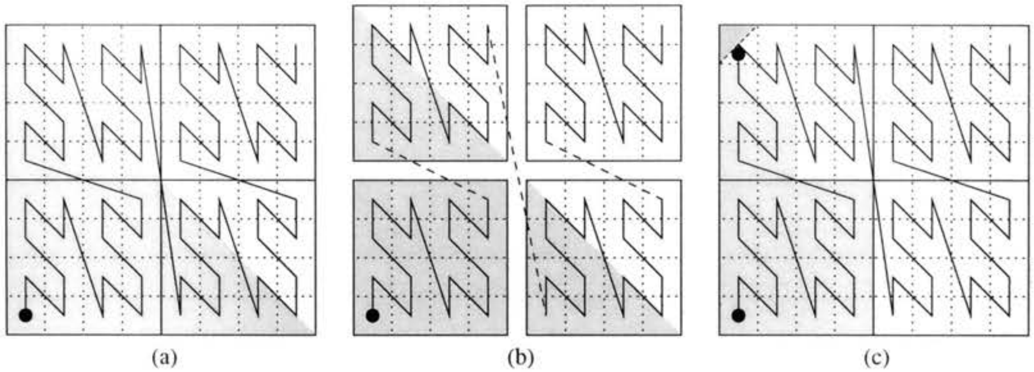


Figure 3.10: Coverages and decompositions of auxiliary and main diagonal structures in a canonical Z_k^2: (a) and (b) for $\overline{\mathcal{A}}_{k,2^k}$; (c) for$\overline{\mathcal{D}'}_{k,2^k}$.

$$\begin{aligned}
&= \overline{\mathcal{C}}_{k,2^{q-1}} + \sum_{\eta=1}^{2^{q-1}} \Delta(C_{k,2^{q-1}+\eta}, \partial_1(Z_k^2)) \\
&= \overline{\mathcal{C}}_{k,2^{q-1}} + \sum_{\eta=1}^{2^{q-1}} (\Delta(C_{k,\eta}, \partial_1(Z_k^2)) + 2 \cdot 2^{k+2(q-1)}) \\
&= 2\overline{\mathcal{C}}_{k,2^{q-1}} + 2^{q-1} \cdot 2 \cdot 2^{k+2(q-1)} \\
&= 2\overline{\mathcal{C}}_{k,2^{q-1}} + 2^{k+3q-2}.
\end{aligned}$$

Iterating the recurrence in descending q (to 0), we have:

$$\overline{\mathcal{C}}_{k,2^q} = 2^q\overline{\mathcal{C}}_{k,2^0} + \sum_{\eta=1}^{q} 2^{q-\eta} \cdot 2^{k+3\eta-2} = 2^q\Delta(C_{k,1}, \partial_1(Z_k^2)) + \frac{1}{3}(2^{k+3q} - 2^{k+q}).$$

Part 5: This proof is same as the proof in Lemma 3.5 (part 5).

Part 6: Similar to the proof in Lemma 3.5 (part 6) without any rotation or reflection operations. The structure $\overline{\mathcal{A}}_{k,2^k}$ covers the lower-left half of a canonical Z_k^2. We decompose its coverage into three parts as shown in Figure 3.10(a): $Q_1(Z_k^2)$, the lower-left half of $Q_2(Z_k^2)$, and the lower-left half of $Q_3(Z_k^2)$. Accordingly, we partition $\overline{\mathcal{A}}_{k,2^k}$ into three parts (see Figure 3.10(b)):

1. From $Q_1(Z_k^2)$: the index-cumulation of all grid points in $Q_1(Z_k^2)$ is $\overline{\mathcal{W}}_{k-1}$,

2. From the lower-left half of $Q_2(Z_k^2)$ (covered by $\overline{\mathcal{A}}_{k-1,2^{k-1}}$): the index-cumulation of the lower-left half of $Q_2(Z_k^2)$ with respect to $\partial_1(Q_2(Z_k^2))$ and the cumulation

of index-adjustment from $\partial_1(Q_2(Z_k^2))$ to $\partial_1(Z_k^2)$; that is, $\overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot 2^{2(k-1)}$, and

3. From the lower-left half of $Q_3(Z_k^2)$: $\overline{\mathcal{A}}_{k-1,2^{k-1}}$ and the cumulation of index-adjustment from $\partial_1(Q_3(Z_k^2))$ to $\partial_1(Z_k^2)$; that is, $\overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot (2 \cdot 2^{2(k-1)})$.

Thus,

$$\begin{aligned}
\overline{\mathcal{A}}_{k,2^k} &= \sum_{\beta=1}^{2^k} \sum_{v \in A_{k,\beta}} \hbar_k(v, \partial_1(Z_k^2)) \\
&= \sum_{\beta=1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_1(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2)) + \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_2(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2)) \\
&\quad + \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_3(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2)) \\
&= \overline{\mathcal{W}}_{k-1} + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_2(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2)) + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_3(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2)) \\
&= \overline{\mathcal{W}}_{k-1} \\
&\quad + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_2(Z_k^2)} (\hbar_k(v, \partial_1(Q_2(Z_k^2))) + \hbar_k(\partial_1(Q_2(Z_k^2)), \partial_1(Z_k^2))) \\
&\quad (\text{“}v \in A_{k,2^{k-1}+\beta} \cap Q_2(Z_k^2)\text{” is equivalent to “}v \in A_{k-1,\beta} \text{ in a canonical } Z_{k-1}^2\text{”}) \\
&\quad + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_3(Z_k^2)} (\hbar_k(v, \partial_1(Q_3(Z_k^2))) + \hbar_k(\partial_1(Q_3(Z_k^2)), \partial_1(Z_k^2))) \\
&\quad (\text{“}v \in A_{k,2^{k-1}+\beta} \cap Q_3(Z_k^2)\text{” is equivalent to “}v \in A_{k-1,\beta} \text{ in a canonical } Z_{k-1}^2\text{”}) \\
&= \overline{\mathcal{W}}_{k-1} + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k-1,\beta}} (\hbar_{k-1}(v, \partial_1(Z_{k-1}^2)) + 2^{2(k-1)}) \\
&\quad + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k-1,\beta}} (\hbar_{k-1}(v, \partial_1(Z_{k-1}^2)) + 2 \cdot 2^{2(k-1)}) \\
&= \overline{\mathcal{W}}_{k-1} + (\overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot 2^{2(k-1)}) + (\overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot (2 \cdot 2^{2(k-1)})) \\
&= 2\overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{W}}_{k-1} + 3 \cdot 2^{2k-2} \overline{\mathcal{N}}_{k-1,2^{k-1}}.
\end{aligned}$$

Now we establish a recurrence (in k) for $\overline{\mathcal{A}}_{k,2^k}$ as follows:

$$\overline{\mathcal{A}}_{k,2^k} = \begin{cases} 2\overline{\mathcal{A}}_{k-1,2^{k-1}} + \overline{\mathcal{W}}_{k-1} + 3 \cdot 2^{2k-2}\overline{\mathcal{N}}_{k-1,2^{k-1}} & \text{if } k > 1 \\ 3 & \text{if } k = 1. \end{cases}$$

The closed-form of $\overline{\mathcal{A}}_{k,2^k}$ is $\frac{1}{7} \cdot 2^{4k} + \frac{1}{2^2} \cdot 2^{3k} - \frac{1}{2^2} \cdot 2^{2k} - \frac{1}{7} \cdot 2^k$.

Part 7: Similar to part 5.

Part 8: similar to part 6. The structure $\overline{\mathcal{D}'}_{k,2^k}$ covers the upper-left half of a canonical Z_k^2. We decompose its coverage into three parts: $Q_2(Z_k^2)$, upper-left half of $Q_1(Z_k^2)$, and the upper-left half of $Q_4(Z_k^2)$.

$$\overline{\mathcal{D}'}_{k,2^k} = \sum_{\beta=1}^{2^k} \sum_{v \in A_{k,\beta}} \hbar_k(v, \partial_1(Z_k^2))$$

$$= \sum_{\beta=1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_2(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2)) + \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_1(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2))$$

$$+ \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_4(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2))$$

$$= \overline{\mathcal{W}}_{k-1} + 2^{2(k-1)} \cdot 2^{2(k-1)} + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_1(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2))$$

$$+ \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_4(Z_k^2)} \hbar_k(v, \partial_1(Z_k^2))$$

$$= \overline{\mathcal{W}}_{k-1} + 2^{2(k-1)} \cdot 2^{2(k-1)}$$

$$+ \sum_{\beta=1}^{2^{k-1}} \sum_{v \in D'_{k,2^{k-1}+\beta} \cap Q_1(Z_k^2)} (\hbar_k(v, \partial_1(Q_1(Z_k^2))) + \hbar_k(\partial_1(Q_1(Z_k^2)), \partial_1(Z_k^2)))$$

("$v \in D'_{k,2^{k-1}+\beta} \cap Q_1(Z_k^2)$" is equivalent to "$v \in D'_{k-1,\beta}$ in a canonical Z_{k-1}^2")

$$+ \sum_{\beta=1}^{2^{k-1}} \sum_{v \in D'_{k,2^{k-1}+\beta} \cap Q_4(Z_k^2)} (\hbar_k(v, \partial_1(Q_4(Z_k^2))) + \hbar_k(\partial_1(Q_4(Z_k^2)), \partial_1(Z_k^2)))$$

("$v \in D'_{k,2^{k-1}+\beta} \cap Q_4(Z_k^2)$" is equivalent to "$v \in D'_{k-1,\beta}$ in a canonical Z_{k-1}^2")

$$= \overline{\mathcal{W}}_{k-1} + 2^{2(k-1)} \cdot 2^{2(k-1)} + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in D'_{k-1,\beta}} \hbar_{k-1}(v, \partial_1(Z_{k-1}^2))$$

$$+\sum_{\beta=1}^{2^{k-1}} \sum_{v \in D'_{k-1,\beta}} (\hbar_{k-1}(v, \partial_1(Z^2_{k-1})) + 3 \cdot 2^{2(k-1)})$$

$$= \overline{\mathcal{W}}_{k-1} + 2^{2(k-1)} \cdot 2^{2(k-1)} + (\overline{\mathcal{D}'}_{k-1,2^{k-1}}) + (\overline{\mathcal{D}'}_{k-1,2^{k-1}} + \overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot (3 \cdot 2^{2(k-1)}))$$

$$= 2\overline{\mathcal{D}'}_{k-1,2^{k-1}} + 2^{2(k-1)} \cdot 2^{2(k-1)} + \overline{\mathcal{W}}_{k-1} + 3 \cdot 2^{2k-2}\overline{\mathcal{N}}_{k-1,2^{k-1}}.$$

Now we establish a recurrence (in k) for $\overline{D'}_{k,2^k}$ as follows:

$$\overline{D'}_{k,2^k} = \begin{cases} 2\overline{\mathcal{D}'}_{k-1,2^{k-1}} + 2^{2(k-1)} \cdot 2^{2(k-1)} + \overline{\mathcal{W}}_{k-1} + 3 \cdot 2^{2k-2}\overline{\mathcal{N}}_{k-1,2^{k-1}} & \text{if } k > 1 \\ 4 & \text{if } k = 1. \end{cases}$$

The closed-form of $\overline{D'}_{k,2^k}$ is $\frac{3}{2\cdot 7} \cdot 2^{4k} + \frac{1}{2^2} \cdot 2^{3k} - \frac{1}{2^2} \cdot 2^{2k} - \frac{3}{2\cdot 7} \cdot 2^k$.

Part 9: If $q = 1$, then the desired equality is obviously true. Consider that $q < k$. The structure $\overline{\mathcal{D}'}_{k,2^q}$ covers the upper-left corner of 2^q main diagonals in a canonical Z^2_k. Similar to Lemma 3.5 (part 8), $\overline{\mathcal{D}'}_{k,2^q}$ is the upper-left half of $Q^{k-q}_2(Z^2_k)$, which is an Z^2_q-subcurve canonically oriented at the upper-left corner of the Z^2_k (see Figure 3.10(c)). Therefore,

$$\overline{\mathcal{D}'}_{k,2^q} = \sum_{\beta=1}^{2^q} \sum_{v \in D'_{k,\beta}} \hbar_k(v, \partial_1(Z^2_k))$$

$$= \sum_{\beta=1}^{2^q} \sum_{v \in D'_{k,\beta}} (\hbar_k(v, \partial_1(Q^{k-q}_2(Z^2_k))) + \hbar_k(\partial_1(Q^{k-q}_2(Z^2_k)), \partial_1(Z^2_k)))$$

$$= \sum_{\beta=1}^{2^q} \sum_{v \in D'_{q,\beta}} (\hbar_q(v, \partial_1(Z^2_q)) + \sum_{\eta=q}^{k-1} 2^{2\eta}) = \sum_{\beta=1}^{2^q} \sum_{v \in D'_{q,\beta}} \hbar_q(v, \partial_1(Z^2_q)) + \sum_{\beta=1}^{2^q} \sum_{v \in D'_{q,\beta}} \sum_{\eta=q}^{k-1} 2^{2\eta}$$

$$= \overline{\mathcal{D}'}_{q,2^q} + \frac{1}{3}(2^{2k} - 2^{2q}) \sum_{\beta=1}^{2^q} \sum_{v \in D'_{q,\beta}} 1 = \frac{1}{3}(2^{2k} - 2^{2q})\overline{\mathcal{N}}_{k,2^q} + \overline{\mathcal{D}'}_{q,2^q}.$$

∎

Computations of various $\mathcal{X}_{k,2^q}$ are similar to those for $\overline{\mathcal{X}}_{k,2^q}$.

Lemma 3.12 *For all integers q with $0 \le q \le k$,*

1. $\mathcal{N}_{k,2^q} = \frac{1}{2\cdot 3} \cdot 2^{3q} + \frac{1}{2} \cdot 2^{2q} + \frac{1}{3} \cdot 2^q$,

2. $\mathcal{R}_{k,2^q} = \frac{1}{2\cdot 3} \cdot 2^{3k+2q} + \frac{1}{2\cdot 3} \cdot 2^{3k+q} + \frac{1}{3\cdot 7} \cdot 2^{k+4q} + \frac{1}{2^2\cdot 3} \cdot 2^{k+3q} - \frac{1}{2^2} \cdot 2^{k+2q} - \frac{3}{2\cdot 7} \cdot 2^{k+q}$,

3. $\mathcal{C}_{k,2^q} = \frac{1}{2^2 \cdot 3} \cdot 2^{3k+2q} + \frac{1}{2^2 \cdot 3} \cdot 2^{3k+q} + \frac{2}{3 \cdot 7} \cdot 2^{k+4q} + \frac{1}{2 \cdot 3} \cdot 2^{k+3q} - \frac{1}{2^2} \cdot 2^{k+2q} - \frac{5}{2^2 \cdot 7} \cdot 2^{k+q}$,

4. *Each of the pairs:* $(\mathcal{D}_{k,2^q}, \mathcal{D}'_{k,2^q})$ *and* $(\mathcal{A}_{k,2^q}, \mathcal{A}'_{k,2^q}))$ *are related via* $\mathcal{N}_{k,2^q}$ *as follows:*

$$\mathcal{A}_{k,2^q} + \mathcal{A}'_{k,2^q} = \mathcal{D}_{k,2^q} + \mathcal{D}'_{k,2^q} = (2^{2k} - 1)\mathcal{N}_{k,2^q},$$

5.

$$\mathcal{A}_{k,2^k} = \begin{cases} 0 & \textit{if } k = 0 \\ 3 & \textit{if } k = 1 \\ \frac{5}{2^3 \cdot 3 \cdot 7} \cdot 2^{5k} + \frac{1}{7} \cdot 2^{4k} + \frac{1}{2^2 \cdot 3} \cdot 2^{3k} - \frac{11}{2^3 \cdot 7} \cdot 2^{2k} - \frac{5}{2^2 \cdot 3 \cdot 7} \cdot 2^k & \textit{otherwise,} \end{cases}$$

6. $\mathcal{A}_{k,2^q} = \mathcal{A}_{q,2^q}$,

7.

$$\mathcal{D}'_{k,2^k} = \begin{cases} 0 & \textit{if } k = 0 \\ 5 & \textit{if } k = 1 \\ \frac{11}{2^3 \cdot 3 \cdot 7} \cdot 2^{5k} + \frac{3}{2 \cdot 7} \cdot 2^{4k} + \frac{1}{2^2 \cdot 3} \cdot 2^{3k} - \frac{13}{2^3 \cdot 7} \cdot 2^{2k} - \frac{11}{2^2 \cdot 3 \cdot 7} \cdot 2^k & \textit{otherwise,} \end{cases}$$

8. $\mathcal{D}'_{k,2^q} = \frac{1}{3}(2^{2k} - 2^{2q})\mathcal{N}_{k,2^q} + \mathcal{D}'_{q,2^q}$.

Proof. Let q be an arbitrary integer with $0 \leq q \leq k$.

Part 1: Similar to the proof of Lemma 3.5 (part 2). For the symbol X denoting D, D', A, or A', the number of grid points in $X_{k,\beta}$, where $\beta \in [2^k]$, is β. Therefore,

$$\mathcal{N}_{k,2^q} = \sum_{\beta=1}^{2^q} \sum_{v \in X_{k,\beta}} (2^q + 1 - \beta) = \sum_{\beta=1}^{2^q} (2^q + 1 - \beta)\beta = \frac{1}{2 \cdot 3} \cdot 2^{3q} + \frac{1}{2} \cdot 2^{2q} + \frac{1}{3} \cdot 2^q.$$

Part 2: We first express $\mathcal{R}_{k,2^q}$ in terms of $\overline{\mathcal{R}}_{k,2^q}$ (computed in Lemma 3.11 (part 3) and Lemma 3.7 (part 1)) and $\sum_{\beta=1}^{2^q} \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2))$:

$$\begin{aligned} \mathcal{R}_{k,2^q} &= \sum_{\beta=1}^{2^q} \sum_{v \in R_{k,\beta}} (2^q + 1 - \beta)\hbar_k(v, \partial_1(Z_k^2)) \\ &= \sum_{\beta=1}^{2^q} ((2^q + 1 - \beta) \sum_{v \in R_{k,\beta}} \hbar_k(v, \partial_1(Z_k^2))) = \sum_{\beta=1}^{2^q} (2^q + 1 - \beta)\Delta(R_{k,\beta}, \partial_1(Z_k^2)) \\ &= \sum_{\beta=1}^{2^q} ((2^q + 1)\Delta(R_{k,\beta}, \partial_1(Z_k^2)) - \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2))) \\ &= (2^q + 1) \sum_{\beta=1}^{2^q} \Delta(R_{k,\beta}, \partial_1(Z_k^2)) - \sum_{\beta=1}^{2^q} \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2)) \end{aligned}$$

$$= (2^q+1)\overline{\mathcal{R}}_{k,2^q} - \sum_{\beta=1}^{2^q} \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2)).$$

Let $\mathcal{U}_{k,2^q}$ denote $\sum_{\beta=1}^{2^q} \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2))$, so $\mathcal{U}_{k,2^0} = \Delta(R_{k,1}, \partial_1(Z_k^2))$, which is computed in Lemma 3.7 (part 1). We establish a recurrence (in q) for $\mathcal{U}_{k,2^q}$, similar to that for $\overline{\mathcal{R}}_{k,2^q}$ in the proof of Lemma 3.11 (part 3), as follows:

$$\begin{aligned}
&\mathcal{U}_{k,2^q}\\
=\;& \sum_{\beta=1}^{2^q} \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2)) = \sum_{\beta=1}^{2^{q-1}} \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2)) + \sum_{\beta=2^{q-1}+1}^{2^q} \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2))\\
=\;& \mathcal{U}_{k,2^{q-1}} + \sum_{\beta=1}^{2^{q-1}} (2^{q-1}+\beta)\Delta(R_{k,2^{q-1}+\beta}, \partial_1(Z_k^2))\\
=\;& \mathcal{U}_{k,2^{q-1}} + \sum_{\beta=1}^{2^{q-1}} (2^{q-1}+\beta)(\Delta(R_{k,\beta}, \partial_1(Z_k^2)) + 2^{k+2(q-1)})\\
& \qquad \text{(by Lemma 3.7 (part 2))}\\
=\;& \mathcal{U}_{k,2^{q-1}} + 2^{q-1}\sum_{\beta=1}^{2^{q-1}} \Delta(R_{k,\beta}, \partial_1(Z_k^2)) + \sum_{\beta=1}^{2^{q-1}} \beta\Delta(R_{k,\beta}, \partial_1(Z_k^2))\\
& + \sum_{\beta=1}^{2^{q-1}} (2^{q-1}+\beta)2^{k+2q-2}\\
=\;& 2\mathcal{U}_{k,2^{q-1}} + 2^{q-1}\overline{\mathcal{R}}_{k,2^{q-1}} + 3\cdot 2^{k+4q-5} + 2^{k+3q-4}.
\end{aligned}$$

With $\overline{\mathcal{R}}_{k,2^{q-1}}$ and $\mathcal{U}_{k,2^0} = \Delta(R_{k,1}, \partial_1(Z_k^2))$ computed in Lemma 3.11 (part 3) and Lemma 3.7 (part 1), the closed-form solution for $\mathcal{U}_{k,2^q}$ is:

$$\frac{1}{2\cdot 3}\cdot 2^{3k+2q} + \frac{1}{2\cdot 3}\cdot 2^{3k+q} + \frac{5}{2\cdot 3\cdot 7}\cdot 2^{k+4q} + \frac{1}{2^2\cdot 3}\cdot 2^{k+3q} - \frac{1}{2^2}\cdot 2^{k+2q} - \frac{2}{7}\cdot 2^{k+q}.$$

Now,

$$\begin{aligned}
&\mathcal{R}_{k,2^q} = (2^q+1)\overline{\mathcal{R}}_{k,2^q} - \mathcal{U}_{k,2^q}\\
=\;& \frac{1}{2\cdot 3}\cdot 2^{3k+2q} + \frac{1}{2\cdot 3}\cdot 2^{3k+q} + \frac{1}{3\cdot 7}\cdot 2^{k+4q} + \frac{1}{2^2\cdot 3}\cdot 2^{k+3q} - \frac{1}{2^2}\cdot 2^{k+2q}\\
& - \frac{3}{2\cdot 7}\cdot 2^{k+q}.
\end{aligned}$$

Part 3: We first express $\mathcal{C}_{k,2^q}$ in terms of $\overline{\mathcal{C}}_{k,2^q}$ (computed in Lemma 3.11 (part 4) and Lemma 3.8 (part 1)) and $\sum_{\beta=1}^{2^q} \beta\Delta(C_{k,\beta}, \partial_1(Z_k^2))$:

$$\begin{aligned}
\mathcal{C}_{k,2^q} &= \sum_{\beta=1}^{2^q} \sum_{v \in C_{k,\beta}} (2^q+1-\beta)\hbar_k(v, \partial_1(Z_k^2)) \\
&= \sum_{\beta=1}^{2^q}((2^q+1-\beta)\sum_{v\in C_{k,\beta}} \hbar_k(v, \partial_1(Z_k^2))) \\
&= \sum_{\beta=1}^{2^q}(2^q+1-\beta)\Delta(C_{k,\beta}, \partial_1(Z_k^2)) \\
&= \sum_{\beta=1}^{2^q}((2^q+1)\Delta(C_{k,\beta}, \partial_1(Z_k^2)) - \beta\Delta(C_{k,\beta}, \partial_1(Z_k^2))) \\
&= (2^q+1)\sum_{\beta=1}^{2^q}\Delta(C_{k,\beta}, \partial_1(Z_k^2)) - \sum_{\beta=1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(Z_k^2)) \\
&= (2^q+1)\overline{\mathcal{C}}_{k,2^q} - \sum_{\beta=1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(Z_k^2)).
\end{aligned}$$

Let $\mathcal{U}_{k,2^q}$ denote $\sum_{\beta=1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(Z_k^2))$, so $\mathcal{U}_{k,2^0} = \Delta(C_{k,1}, \partial_1(Z_k^2))$, which is computed in Lemma 3.8 (part 1). We establish a recurrence (in q) for $\mathcal{U}_{k,2^q}$, similar to that for $\overline{\mathcal{C}}_{k,2^q}$ in the proof of Lemma 3.11 (part 4), as follows:

$$\begin{aligned}
&\mathcal{U}_{k,2^q} \\
= &\sum_{\beta=1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(Z_k^2)) = \sum_{\beta=1}^{2^{q-1}}\beta\Delta(C_{k,\beta}, \partial_1(Z_k^2)) + \sum_{\beta=2^{q-1}+1}^{2^q}\beta\Delta(C_{k,\beta}, \partial_1(Z_k^2)) \\
= &\mathcal{U}_{k,2^{q-1}} + \sum_{\beta=1}^{2^{q-1}}(2^{q-1}+\beta)\Delta(C_{k,2^{q-1}+\beta}, \partial_1(Z_k^2)) \\
= &\mathcal{U}_{k,2^{q-1}} + \sum_{\beta=1}^{2^{q-1}}(2^{q-1}+\beta)(\Delta(C_{k,\beta}, \partial_1(Z_k^2)) + 2\cdot 2^{k+2(q-1)}) \\
&\quad\text{(by Lemma 3.8 (part 2))} \\
= &\mathcal{U}_{k,2^{q-1}} + 2^{q-1}\sum_{\beta=1}^{2^{q-1}}\Delta(C_{k,\beta}, \partial_1(Z_k^2)) + \sum_{\beta=1}^{2^{q-1}}\beta\Delta(C_{k,\beta}, \partial_1(Z_k^2)) \\
&+\sum_{\beta=1}^{2^{q-1}}(2^{q-1}+\beta)2^{k+2q-1} \\
= &2\mathcal{U}_{k,2^{q-1}} + 2^{q-1}\overline{\mathcal{C}}_{k,2^{q-1}} + 3\cdot 2^{k+4q-4} + 2^{k+3q-3}.
\end{aligned}$$

With $\overline{\mathcal{C}}_{k,2^{q-1}}$ and $\mathcal{U}_{k,2^0} = \Delta(C_{k,1}, \partial_1(Z_k^2))$ computed in Lemma 3.11 (part 4) and Lemma 3.8 (part 1), the closed-form solution for $\mathcal{U}_{k,2^q}$ is:

$$\frac{1}{2^2\cdot 3}\cdot 2^{3k+2q}+\frac{1}{2^2\cdot 3}\cdot 2^{3k+q}+\frac{5}{3\cdot 7}\cdot 2^{k+4q}+\frac{1}{2\cdot 3}\cdot 2^{k+3q}-\frac{1}{2^2}\cdot 2^{k+2q}-\frac{3^2}{2^2\cdot 7}\cdot 2^{k+q}.$$

Now,

$$\begin{aligned}
\mathcal{C}_{k,2^q} &= (2^q+1)\overline{\mathcal{C}}_{k,2^q}-\mathcal{U}_{k,2^q}\\
&= \frac{1}{2^2\cdot 3}\cdot 2^{3k+2q}+\frac{1}{2^2\cdot 3}\cdot 2^{3k+q}+\frac{2}{3\cdot 7}\cdot 2^{k+4q}+\frac{1}{2\cdot 3}\cdot 2^{k+3q}-\frac{1}{2^2}\cdot 2^{k+2q}\\
&\quad -\frac{5}{2^2\cdot 7}\cdot 2^{k+q}.
\end{aligned}$$

Part 4: Similar to the proof of Lemma 3.11 (part 5): for every point $v \in A_{k,\alpha}$ and its mirror point $v' \in A'_{k,\alpha}$, $\hbar_k(v,\partial_1(Z_k^2)) + \hbar_k(v',\partial_1(Z_k^2)) = 2^{2k}-1$.

$$\begin{aligned}
&\mathcal{A}_{k,2^q}+\mathcal{A}'_{k,2^q}\\
&= \sum_{\beta=1}^{2^q}\sum_{v\in A_{k,\beta}}(2^q+1-\beta)\hbar_k(v,\partial_1(Z_k^2))+\sum_{\beta=1}^{2^q}\sum_{v\in A'_{k,\beta}}(2^q+1-\beta)\hbar_k(v,\partial_1(Z_k^2))\\
&= \sum_{\beta=1}^{2^q}\sum_{\text{all mirror pairs } (v,v')\in A_{k,\beta}\times A'_{k,\beta}}(2^q+1-\beta)(\hbar_k(v,\partial_1(Z_k^2))+\hbar_k(v',\partial_1(Z_k^2)))\\
&= \sum_{\beta=1}^{2^q}\sum_{\text{all mirror pairs } (v,v')\in A_{k,\beta}\times A'_{k,\beta}}(2^q+1-\beta)(2^{2k}-1) = (2^{2k}-1)\mathcal{N}_{k,2^q}.
\end{aligned}$$

Similarly, we can derive $\mathcal{D}_{k,2^q}+\mathcal{D}'_{k,2^q} = (2^{2k}-1)\mathcal{N}_{k,2^q}$

Part 5: We proceed as in the proof of Lemmas 3.11 (part 6) and 3.6 (part 5). The structure $\mathcal{A}_{k,2^k}$ covers the lower-left half of a canonical Z_k^2. We decompose its coverage into four parts (non-empty when $k \geq 2$) as shown in Figure 3.11(a) and (b): the lower-left half of $Q_1(Z_k^2)$, the upper-right half of $Q_1(Z_k^2)$ without the auxiliary diagonal $A_{k,2^k}(= A'_{k,2^k})$, the lower-left half of $Q_2(Z_k^2)$, and the lower-left half of $Q_3(Z_k^2)$. Accordingly, we partition $\mathcal{A}_{k,2^k}$ into four parts (see Figure 3.11(b)) as follows. In the notation, $\mathcal{A}_{k,\alpha} = \sum_{\beta=1}^{\alpha}\sum_{v\in A_{k,\beta}}(\alpha+1-\beta)\hbar_k(v,\partial_1(Z_k^2))$, we associate the weight $\alpha+1-\beta$ with $A_{k,\beta}$ in the summation.

1. From the lower-left half of $Q_1(Z_k^2)$:

$$\begin{aligned}
&\sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,\beta} \cap Q_1(Z_k^2)} (2^k + 1 - \beta)\hbar_k(v, \partial_1(Z_k^2)) \\
= &\sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,\beta} \cap Q_1(Z_k^2)} (2^{k-1} + 1 - \beta)\hbar_k(v, \partial_1(Z_k^2)) \\
&+ \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,\beta} \cap Q_1(Z_k^2)} 2^{k-1}\hbar_k(v, \partial_1(Z_k^2)) \\
= &\sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k-1,\beta}} (2^{k-1} + 1 - \beta)\hbar_{k-1}(v, \partial_1(Z_{k-1}^2)) \\
&+ 2^{k-1} \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k-1,\beta}} \hbar_{k-1}(v, \partial_1(Z_{k-1}^2)) \\
= &\mathcal{A}_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{A}}_{k-1,2^{k-1}}.
\end{aligned}$$

2. For the upper-right half of $Q_1(Z_k^2)$ without the auxiliary diagonal $A_{k,2^k}$:

$$\begin{aligned}
&\sum_{\beta=2^{k-1}+1}^{2^k-1} \sum_{v \in A_{k,\beta} \cap Q_1(Z_k^2)} (2^k + 1 - \beta)\hbar_k(v, \partial_1(Z_k^2)) \\
= &\sum_{\gamma=1}^{2^{k-1}-1} \sum_{v \in A_{k,2^{k-1}+\gamma} \cap Q_1(Z_k^2)} (2^k + 1 - (2^{k-1} + \gamma))\hbar_k(v, \partial_1(Z_k^2)) \\
&\text{(change of summation index: } \beta = 2^{k-1} + \gamma) \\
= &\sum_{\beta=1}^{2^{k-1}-1} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_1(Z_k^2)} (2^{k-1} + 1 - \beta)\hbar_k(v, \partial_1(Z_k^2)) \\
= &\sum_{\beta=1}^{2^{k-1}-1} \sum_{v \in A'_{k-1,\beta}} (\beta + 1)\hbar_{k-1}(v, \partial_1(Z_{k-1}^2)) \quad \text{(in a canonical } Z_{k-1}^2) \\
= &\sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (\beta + 1)\hbar_{k-1}(v, \partial_1(Z_{k-1}^2)) \\
&- \sum_{v \in A'_{k-1,2^{k-1}}} (2^{k-1} + 1)\hbar_{k-1}(v, \partial_1(Z_{k-1}^2)) \\
= &(\sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (2^{k-1} + 2)\hbar_{k-1}(v, \partial_1(Z_{k-1}^2))
\end{aligned}$$

$$
\begin{aligned}
& -\sum_{\beta=1}^{2^{k-1}} \sum_{v \in A'_{k-1,\beta}} (2^{k-1}+1-\beta)\hbar_{k-1}(v, \partial_1(Z^2_{k-1}))) \\
& -(2^{k-1}+1)\Delta(A'_{k-1,2^{k-1}}, \partial_1(Z^2_{k-1})) \\
= \; & (2^{k-1}+2)\overline{\mathcal{A}'}_{k-1,2^{k-1}} - \mathcal{A}'_{k-1,2^{k-1}} - (2^{k-1}+1)\Delta(A'_{k-1,2^{k-1}}, \partial_1(Z^2_{k-1})).
\end{aligned}
$$

3. From the lower-left half of the canonically oriented Z^2_{k-1}-subcurve $Q_2(Z^2_k)$ — consisting of auxiliary-diagonal segments $A_{k,\beta} \cap Q_2(Z^2_k)$ with weights $2^k+1-\beta$ for $\beta = 2^{k-1}+1, 2^{k-1}+2, \ldots, 2^k-1$ (indexed from the lower-left corner of $Q_2(Z^2_k)$):

   Including the cumulation of index-adjustment from $\partial_1(Q_2(Z^2_k))$ to $\partial_1(Z^2_k)$, its contribution in $\mathcal{A}_{k,2^k}$ is :

$$
\begin{aligned}
& \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_2(Z^2_k)} (2^k+1-\beta)\hbar_k(v, \partial_1(Z^2_k)) \\
= \; & \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v \in A_{k,\beta} \cap Q_2(Z^2_k)} (2^k+1-\beta)(\hbar_k(v, \partial_1(Q_2(Z^2_k))) \\
& \quad + \hbar_k(\partial_1(Q_2(Z^2_k)), \partial_1(Z^2_k))) \\
= \; & \sum_{\gamma=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\gamma} \cap Q_2(Z^2_k)} (2^k+1-(2^{k-1}+\gamma))(\hbar_k(v, \partial_1(Q_2(Z^2_k))) + 2^{2(k-1)}) \\
& \text{(change of summation index: } \beta = 2^{k-1}+\gamma) \\
= \; & \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k,2^{k-1}+\beta} \cap Q_2(Z^2_k)} (2^{k-1}+1-\beta)(\hbar_k(v, \partial_1(Q_2(Z^2_k))) + 2^{2(k-1)}) \\
= \; & \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k-1,\beta}} (2^{k-1}+1-\beta)\hbar_{k-1}(v, \partial_1(Z^2_{k-1})) \\
& + \sum_{\beta=1}^{2^{k-1}} \sum_{v \in A_{k-1,\beta}} (2^{k-1}+1-\beta)2^{2(k-1)} \\
= \; & \mathcal{A}_{k-1,2^{k-1}} + 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}}.
\end{aligned}
$$

4. From the lower-left half of $Q_3(Z^2_k)$ — consisting of auxiliary-diagonal segments $A_{k,\beta} \cap Q_3(Z^2_k)$ with weights $2^k+1-\beta$ for $\beta = 2^{k-1}+1, 2^{k-1}+2, \ldots, 2^k-1$

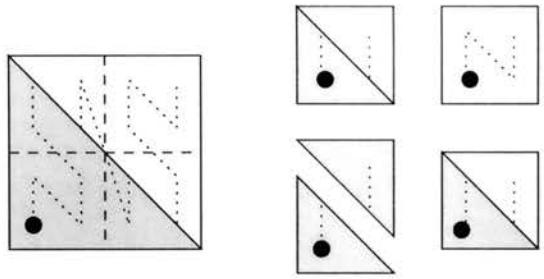
(indexed from the lower-left corner of $Q_3(Z_k^2)$):

Including the cumulation of index-adjustment from $\partial_1(Q_3(Z_k^2))$ to $\partial_1(Z_k^2)$, its contribution in $\mathcal{A}_{k,2^k}$ is:

$$
\begin{aligned}
& \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v\in A_{k,\beta}\cap Q_3(Z_k^2)} (2^k+1-\beta)\hbar_k(v,\partial_1(Z_k^2)) \\
= & \sum_{\beta=2^{k-1}+1}^{2^k} \sum_{v\in A_{k,\beta}\cap Q_3(Z_k^2)} (2^k+1-\beta)(\hbar_k(v,\partial_1(Q_3(Z_k^2))) \\
& \quad + \hbar_k(\partial_1(Q_3(Z_k^2)),\partial_1(Z_k^2))) \\
= & \sum_{\gamma=1}^{2^{k-1}} \sum_{v\in A_{k,2^{k-1}+\gamma}\cap Q_3(Z_k^2)} (2^k+1-(2^{k-1}+\gamma))(\hbar_k(v,\partial_1(Q_3(Z_k^2)))+2\cdot 2^{2(k-1)}) \\
& \text{(change of summation index: } \beta = 2^{k-1}+\gamma) \\
= & \sum_{\beta=1}^{2^{k-1}} \sum_{v\in A_{k,2^{k-1}+\beta}\cap Q_3(Z_k^2)} (2^{k-1}+1-\beta)(\hbar_k(v,\partial_1(Q_3(Z_k^2)))+2\cdot 2^{2(k-1)}) \\
= & \sum_{\beta=1}^{2^{k-1}} \sum_{v\in A_{k-1,\beta}} (2^{k-1}+1-\beta)\hbar_{k-1}(v,\partial_1(Z_{k-1}^2)) \\
& + \sum_{\beta=1}^{2^{k-1}} \sum_{v\in A_{k-1,\beta}} (2^{k-1}+1-\beta)(2\cdot 2^{2(k-1)}) \\
= & \mathcal{A}_{k-1,2^{k-1}} + 2\cdot 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}}.
\end{aligned}
$$

With the four contributions, we establish a recurrence (in $k \geq 2$) for $\mathcal{A}_{k,2^k}$ as follows:

$$
\begin{aligned}
& \mathcal{A}_{k,2^k} \\
= & (\mathcal{A}_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{A}}_{k-1,2^{k-1}}) \\
& +((2^{k-1}+2)\overline{\mathcal{A}'}_{k-1,2^{k-1}} - \mathcal{A}'_{k-1,2^{k-1}} - (2^{k-1}+1)\Delta(A_{k-1,2^{k-1}},\partial_1(Z_{k-1}^2))) \\
& +(\mathcal{A}_{k-1,2^{k-1}} + 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}}) \\
& +(\mathcal{A}_{k-1,2^{k-1}} + 2\cdot 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}}) \\
= & 3\mathcal{A}_{k-1,2^{k-1}} - \mathcal{A}'_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{A}}_{k-1,2^{k-1}} + (2^{k-1}+2)\overline{\mathcal{A}'}_{k-1,2^{k-1}} \\
& +3\cdot 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}} - (2^{k-1}+1)\Delta(A_{k-1,2^{k-1}},\partial_1(Z_{k-1}^2)).
\end{aligned}
$$



(a) (b)

Figure 3.11: (a) Coverage of $\mathcal{A}_{k,2^k}$ in a canonical Z_k^2; (b) decomposition of $\mathcal{A}_{k,2^k}$ into four triangular halves.

Note that Lemma 3.9 (part 2), Lemma 3.11 (part 6), and Lemma 3.12 (part 1) are used to obtain exact formulas for $\Delta(A_{k-1,2^{k-1}}, \partial_1(Z_{k-1}^2))$, $\overline{\mathcal{A}}_{k-1,2^{k-1}}$, and $\mathcal{N}_{k-1,2^{k-1}}$ in above recurrence.

For $k < 2$, we compute $\mathcal{A}_{k,2^k}$ directly, and the recurrence for $\mathcal{A}_{k,2^k}$ is:

$$\mathcal{A}_{k,2^k} = \begin{cases} 3\mathcal{A}_{k-1,2^{k-1}} - \mathcal{A}'_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{A}}_{k-1,2^{k-1}} + (2^{k-1}+2)\overline{\mathcal{A}'}_{k-1,2^{k-1}} & \\ \quad + 3 \cdot 2^{2k-2}\mathcal{N}_{k-1,2^{k-1}} - (2^{k-1}+1)\Delta(A_{k-1,2^{k-1}}, \partial_1(Z_{k-1}^2)) & \text{if } k > 1 \\ 3 & \text{if } k = 1 \\ 0 & \text{if } k = 0, \end{cases}$$

which yields the desired closed-form solution for $\mathcal{A}_{k,2^k}$.

Part 6: The coverage of $\mathcal{A}_{k,2^q}$ in a canonical Z_k^2 is a canonical Z_q^2-subcurve at the lower-left corner of the Z_k^2, and this sub-curve has the same entry point as Z_k^2 does. Thus, $\mathcal{A}_{k,2^q} = \mathcal{A}_{q,2^q}$.

Part 7: Following the proof of part 5, we partition the coverage of $\mathcal{D}_{k,2^k}$ into four parts (non-empty when $k \geq 2$), and obtain:

$$\begin{aligned} & \mathcal{D}'_{k,2^k} \\ = \; & (\mathcal{D}'_{k-1,2^{k-1}}) \\ & +(2^{k-1}\overline{\mathcal{D}'}_{k-1,2^{k-1}} + \mathcal{D}'_{k-1,2^{k-1}} + 2^{k-1}(\overline{\mathcal{N}}_{k-1,2^{k-1}} \cdot 2^{2(k-1)}) + \mathcal{N}_{k-1,2^{k-1}} \cdot 2^{2(k-1)}) \\ & +((2^{k-1}+2)\overline{\mathcal{D}}_{k-1,2^{k-1}} - \mathcal{D}_{k-1,2^{k-1}} - \Delta(D_{k-1,2^{k-1}}, \partial_1(Z_{k-1}^2))(2^{k-1}+1) \\ & +((2^{k-1}+2)\overline{\mathcal{N}}_{k-1,2^{k-1}} - \mathcal{N}_{k-1,2^{k-1}} - 2^{k-1}(2^{k-1}+1))2^{2(k-1)}) \\ & +(\mathcal{D}'_{k-1,2^{k-1}} + \mathcal{N}_{k-1,2^{k-1}} \cdot 3 \cdot 2^{2(k-1)}). \end{aligned}$$

This gives a recurrence (in $k \geq 2$) for $\mathcal{D}'_{k,2^k}$ as follows:

$$\mathcal{D}'_{k,2^k} = \begin{cases} 3\mathcal{D}'_{k-1,2^{k-1}} - \mathcal{D}_{k-1,2^{k-1}} + 2^{k-1}\overline{\mathcal{D}'} + (2^{k-1}+2)\overline{\mathcal{D}} & \\ \quad + ((2 \cdot 2^{3k-3} + 2 \cdot 2^{2k-2})\overline{\mathcal{N}}_{k-1,2^{k-1}} & \\ \quad + 3 \cdot 2^{2(k-1)}\mathcal{N}_{k-1,2^{k-1}} - (2^{4k-4} + 2^{3k-3}) & \\ \quad - \Delta(D_{k-1,2^{k-1}}, \partial_1(Z^2_{k-1}))(2^{k-1}+1) & \text{if } k \geq 2 \\ 5 & \text{if } k = 1 \\ 0 & \text{if } k = 0, \end{cases}$$

which yields the desired closed-form solution for $\mathcal{D}'_{k,2^k}$.

Part 8: Following the proof of Lemma 3.11 (part 9), the coverage of $\mathcal{D}'_{k,2^q}$ is the upper-left half of $Q_2^{k-q}(Z^2_k)$, which is an Z^2_q-subcurve canonically oriented at the upper-left corner of a canonical Z^2_k. The cumulation of index-adjustment involves the summation of $\hbar_k(\partial_1(Q_2^{k-q}(Z^2_k)), \partial_1(Z^2_k))$, which cumulates the traversals through all the intermediate subcurves $Q_1(Q_2^{\eta}(Z^2_k))$ for all $\eta \in \{0, 1, \ldots, k-q-1\}$. Thus,

$$\mathcal{D}'_{k,2^q} = \mathcal{N}_{k,2^q} \sum_{\eta=k-1}^{q} 2^{2\eta} + \mathcal{D}'_{q,2^q} = \frac{1}{3}(2^{2k} - 2^{2q})\mathcal{N}_{k,2^q} + \mathcal{D}'_{q,2^q}.$$

∎

Theorem 3.2 *For $\delta \in [2^k]$ that is an integral power of 2,*

$$L_\delta(Z^2_k) = \begin{cases} 2^{3k} - 2^k & \textit{if } \delta = 1 \\ 2^{3k+2\log\delta} - (\frac{2}{3^2}(k - \log\delta) + \frac{1949}{2^5 \cdot 3^3 \cdot 7})2^{2k+3\log\delta} & \\ \quad + (\frac{2}{3^2}(k - \log\delta) + \frac{7}{2^2 \cdot 3^3})2^{2k+\log\delta} & \\ \quad + \frac{19}{2^2 \cdot 3 \cdot 7} \cdot 2^{2k} - \frac{2^2}{7} \cdot 2^{k+4\log\delta} - \frac{3}{7} \cdot 2^{k+\log\delta} & \\ \quad + \frac{2 \cdot 5}{3^3 \cdot 7} \cdot 2^{5\log\delta} - \frac{2^2}{3^3} \cdot 2^{3\log\delta} + \frac{2}{3 \cdot 7} \cdot 2^{2\log\delta} & \textit{otherwise.} \end{cases}$$

Proof. Let $\delta \in [2^k]$ that is an integral power of 2. Lemma 3.10 gives a recurrence (in k) for $L_\delta(Z^2_k)$ for all $\delta \in [2^{k-1}]$ that is an integral power of 2. It suffices to determine the basis for the recurrence for all δ.

In the extreme case when $\delta = 1$, the basis for the recurrence (in k) for $L_1(Z^2_k)$ occurs when $k = 1$; and $L_1(Z^2_1) = 6$. The recurrence in Lemma 3.10 (for $\delta = 1$) with the basis gives the desired closed-form solution for $L_1(Z^2_k)$.

In the general case when $\delta \geq 2$, the basis for the recurrence (in k) for $L_\delta(Z_k^2)$ occurs when $k = \log\delta$, for which the recursive decomposition (in k) halts and we compute $L_\delta(Z^2_{\log\delta})$ based on, in a canonical $Z^2_{\log\delta}$:

$$\begin{aligned} L_\delta(Z^2_{\log\delta}) &= \sum_{i,j\in[\delta^2]\mid i<j \text{ and } d_1(Z^2_{\log\delta}(i), Z^2_{\log\delta}(j))=\delta} |i-j| \\ &= \sum_{v,v'\in Z^2_{\log\delta}\mid d_1(v,v')=\delta} \hbar_{\log\delta}(v,v'). \end{aligned}$$

The summation above requires the knowledge of δ-neighborhood structures in a canonical $Z^2_{\log\delta}$ and the distribution of $\hbar_{\log\delta}(v,v')$ for a δ-pair $(v,v') \in Z^2_{\log\delta} \times Z^2_{\log\delta}$, which is similar to the analysis in the proof of Theorem 3.1. The δ-neighborhood structures shown in Figure 3.9 is also applied to a canonical $Z^2_{\log\delta}$.

Same as in the proof of Theorem 3.1, for the case of auxiliary diagonals, we consider the organization of $\hbar_{\log\delta}(v,v')$ for all δ-pairs (v,v') for the summation in expanding $L_\delta(Z^2_{\log\delta}) = \sum_{v,v'\in Z^2_{\log\delta}\mid d_1(v,v')=\delta} \hbar_{\log\delta}(v,v')$. First we write that:

$$\hbar_{\log\delta}(v,v') = |\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) - \hbar_{\log\delta}(v', \partial_1(Z^2_{\log\delta}))|,$$

and need to determine the algebraic sign of $\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) - \hbar_{\log\delta}(v', \partial_1(Z^2_{\log\delta}))$.

We observe that for every grid point $v \in A_{\log\delta,\alpha}$, where $\alpha \in [\delta-1]$, we have $\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) < \hbar_{\log\delta}(v', \partial_1(Z^2_{\log\delta}))$ for all $v' \in N'_\delta(v)$. This observation indicates that in $L_{\log\delta}(Z^2_{\log\delta})$ of all δ-pairs (v,v') from the case of auxiliary diagonals: $\mathcal{A}'_{\log\delta,\delta-1} - \mathcal{A}_{\log\delta,\delta-1}$.

For the contribution in $L_{\log\delta}(Z^2_{\log\delta})$ of all δ-pairs (v,v') from the case of main diagonals, we observe the following two cases based on the location of a grid point v in the quadrants of the canonical $Z^2_{\log\delta}$.

1. For grid point $v \in D_{\log\delta,\alpha} \cap (\cup_{\eta=3}^4 Q_\eta(Z^2_{\log\delta}))$ and for all $v' \in N'_\delta(v) \cap (\cup_{\eta=1}^2 Q_\eta(Z_k^2))$, where $\alpha \in [\delta-1]$, we have $\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) > \hbar_{\log\delta}(v', \partial_1(Z^2_{\log\delta}))$,

2. For grid point $v \in D_{\log\delta,\alpha} \cap Q_3(Z^2_{\log\delta})$ and for all $v' \in N'_\delta(v) \cap Q_4(Z_k^2)$, where $\alpha \in [\frac{\delta}{2}-1]$, we have $\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) < \hbar_{\log\delta}(v', \partial_1(Z^2_{\log\delta}))$, and

3. For every grid point $v \in D_{\log\delta,\alpha} \cap Q_1(Z^2_{\log\delta})$ and for all $v' \in N'_\delta(v)$, where $\alpha \in \{\frac{\delta}{2}+1, \frac{\delta}{2}+2, \dots, \delta-1\}$, we have $\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) < \hbar_{\log\delta}(v', \partial_1(Z^2_{\log\delta}))$.

This suggests that we decompose the contribution in $L_{\log\delta}(Z^2_{\log\delta})$ of all δ-pairs (v, v') in the case of main diagonals into six parts of index-cumulations:

1. The index-cumulation of all grid points v in the deleted lower-right half of $Q_1(Z^2_{\log\delta})$:

$$
\begin{aligned}
& \sum_{\alpha=\frac{\delta}{2}+1}^{\delta-1} \sum_{v\in D_{\log\delta,\alpha}\cap Q_1(Z^2_{\log\delta})} (\delta-\alpha)\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) \\
= & \sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v\in D_{\log\delta,\alpha+\frac{\delta}{2}}\cap Q_1(Z^2_{\log\delta})} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) \\
= & \sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v\in D_{\log\delta,\alpha+\frac{\delta}{2}}\cap Q_1(Z^2_{\log\delta})} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta}(v, \partial_1(Q_1(Z^2_{\log\delta}))) \\
= & \sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v\in D_{\log\delta-1,\alpha}} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta-1}(v, \partial_1(Z^2_{\log\delta-1})) \\
= & \mathcal{D}_{\log\delta-1,\frac{\delta}{2}-1} \\
= & \mathcal{D}_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta-1,\frac{\delta}{2}}.
\end{aligned}
$$

2. The index-cumulation of all δ-neighbors $v' \in N'_\delta(v)$ for all grid points v in the deleted lower-right half of $Q_1(Z^2_{\log\delta})$:

$$
\begin{aligned}
& \sum_{\beta=1}^{\frac{\delta}{2}-1} \sum_{v'\in D'_{\log\delta,\beta}} (\frac{\delta}{2}-\beta)\hbar_{\log\delta}(v', \partial_1(Z^2_{\log\delta})) \\
= & \mathcal{D}'_{\log\delta,\frac{\delta}{2}-1} = \mathcal{D}'_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta,\frac{\delta}{2}}.
\end{aligned}
$$

3. The index-cumulation of all grid points v in the deleted lower-right half of $Q_3(Z^2_{\log\delta})$ and their δ-neighbors $v' \in N'_\delta(v) \cap Q_4(Z^2_{\log\delta})$:

$$
\sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v\in D_{\log\delta,\alpha}} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta}))
$$

$$= \mathcal{D}_{\log\delta,\frac{\delta}{2}-1}$$
$$= \mathcal{D}_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta,\frac{\delta}{2}}.$$

4. The index-cumulation of all δ-neighbors $v' \in N'_\delta(v) \cap Q_4(Z^2_{\log\delta})$ for the grid points v in the deleted lower-right half of $Q_3(Z^2_{\log\delta})$:

$$\sum_{\beta=\frac{\delta}{2}+1}^{\delta-1} \sum_{v'\in D'_{\log\delta,\beta}\cap Q_4(Z_k^2)} (\delta-\beta)\hbar_{\log\delta}(v',\partial_1(Z^2_{\log\delta}))$$
$$= \sum_{\beta=1}^{\frac{\delta}{2}-1} \sum_{v'\in D'_{\log\delta,\beta+\frac{\delta}{2}}\cap Q_4(Z_k^2)} (\frac{\delta}{2}-\beta)\hbar_{\log\delta}(v',\partial_1(Z^2_{\log\delta}))$$
$$= \sum_{\beta=1}^{\frac{\delta}{2}-1} \sum_{v'\in D'_{\log\delta,\beta+\frac{\delta}{2}}\cap Q_4(Z_k^2)} (\frac{\delta}{2}-\beta)(\hbar_{\log\delta}(v',\partial_1(Q_4(Z^2_{\log\delta})))$$
$$+ \hbar_{\log\delta}(\partial_1(Q_4(Z^2_{\log\delta})),\partial_1(Z^2_{\log\delta})))$$
$$= \sum_{\beta=1}^{\frac{\delta}{2}-1} \sum_{v'\in D'_{\log\delta-1,\beta}} (\frac{\delta}{2}-\beta)(\hbar_{\log\delta-1}(v',\partial_1(Z^2_{\log\delta-1}))+3\cdot(\frac{\delta}{2})^2)$$
$$= \mathcal{D}'_{\log\delta-1,\frac{\delta}{2}-1} + \mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}\cdot 3\cdot(\frac{\delta}{2})^2$$
$$= \mathcal{D}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta-1,\frac{\delta}{2}} + \mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}\cdot 3\cdot(\frac{\delta}{2})^2.$$

5. The index-cumulation of all δ-neighbors $v' \in N'_\delta(v) \cap (\cup_{\eta=1}^2 Q_\eta(Z_k^2))$ for all grid points v in the deleted lower-right half of $Z^2_{\log\delta}$ and not in the deleted lower-right half of $Q_1(Z^2_{\log\delta})$:

$$\sum_{\beta=1}^{\delta-1} \sum_{v'\in D'_{\log\delta,\beta}} (\delta-\beta)\hbar_{\log\delta}(v',\partial_1(Z^2_{\log\delta}))$$
$$-\sum_{\beta=1}^{\frac{\delta}{2}-1} \sum_{v'\in D'_{\log\delta,\beta}} (\frac{\delta}{2}-\beta)\hbar_{\log\delta}(v',\partial_1(Z^2_{\log\delta}))$$
$$-\sum_{\beta=\frac{\delta}{2}+1}^{\delta-1} \sum_{v'\in D'_{\log\delta,\beta}\cap Q_4(Z_k^2)} (\delta-\beta)\hbar_{\log\delta}(v',\partial_1(Z^2_{\log\delta}))$$
$$= D'_{\log\delta,\delta-1} - (\mathcal{D}'_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta,\frac{\delta}{2}})$$

$$-(\mathcal{D}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta-1,\frac{\delta}{2}} + 3(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1})$$

(see parts 2, 4 above)

$$\begin{aligned} = \; & \mathcal{D}'_{\log\delta,\delta} - \overline{\mathcal{D}'}_{\log\delta,\delta} - (\mathcal{D}'_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta,\frac{\delta}{2}}) \\ & -(\mathcal{D}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta-1,\frac{\delta}{2}} + 3(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}). \end{aligned}$$

6. The index-cumulation of all grid points v in the deleted lower-right half of $Z^2_{\log\delta}$ and not in the deleted lower-right half of $Q_1(Z^2_{\log\delta})$ and not in the deleted lower-right half of $Q_3(Z^2_k)$ that their δ-neighbors $v' \in N'_\delta(v) \cap Q_4(Z^2_k)$:

$$\begin{aligned} & \sum_{\alpha=1}^{\delta-1} \sum_{v\in D_{\log\delta,\alpha}} (\delta-\alpha)\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) \\ & - \sum_{\alpha=\frac{\delta}{2}+1}^{\delta-1} \sum_{v\in D_{\log\delta,\alpha}\cap Q_1(Z^2_{\log\delta})} (\delta-\alpha)\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) \\ & - \sum_{\alpha=1}^{\frac{\delta}{2}-1} \sum_{v\in D_{\log\delta,\alpha}} (\frac{\delta}{2}-\alpha)\hbar_{\log\delta}(v, \partial_1(Z^2_{\log\delta})) \\ = \; & \mathcal{D}_{\log\delta,\delta-1} - (\mathcal{D}_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta-1,\frac{\delta}{2}}) - (\mathcal{D}_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta,\frac{\delta}{2}}) \\ = \; & \mathcal{D}_{\log\delta,\delta} - \overline{\mathcal{D}}_{\log\delta,\delta} - (\mathcal{D}_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta-1,\frac{\delta}{2}}) - (\mathcal{D}_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta,\frac{\delta}{2}}). \end{aligned}$$

By combining the six index-cumulations obtained above, we have the contribution in $L_{\log\delta}(Z^2_{\log\delta})$ of all δ-pairs (v, v') in the case of auxiliary diagonals:

$$\begin{aligned} & (\mathcal{D}'_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta,\frac{\delta}{2}}) + (\mathcal{D}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta-1,\frac{\delta}{2}} + 3(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}) \\ & +(\mathcal{D}_{\log\delta,\delta} - \overline{\mathcal{D}}_{\log\delta,\delta} - (\mathcal{D}_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta-1,\frac{\delta}{2}}) - (\mathcal{D}_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta,\frac{\delta}{2}})) \\ & -(\mathcal{D}'_{\log\delta,\delta} - \overline{\mathcal{D}'}_{\log\delta,\delta} - (\mathcal{D}'_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta,\frac{\delta}{2}}) \\ & -(\mathcal{D}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta-1,\frac{\delta}{2}} + 3(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1})) \\ & -(\mathcal{D}_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta-1,\frac{\delta}{2}}) - (\mathcal{D}_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta,\frac{\delta}{2}}) \\ = \; & \mathcal{D}_{\log\delta,\delta} - \overline{\mathcal{D}}_{\log\delta,\delta} - 2(\mathcal{D}_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta-1,\frac{\delta}{2}}) - 2(\mathcal{D}_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta,\frac{\delta}{2}}) \\ & -(\mathcal{D}'_{\log\delta,\delta} - \overline{\mathcal{D}'}_{\log\delta,\delta}) + 2(\mathcal{D}'_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta,\frac{\delta}{2}}) \\ & +2(\mathcal{D}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta-1,\frac{\delta}{2}} + 3(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}). \end{aligned}$$

These two contributions are combined into the basis:

$$\begin{aligned} L_\delta(Z^2_{\log\delta}) &= (\mathcal{A}'_{\log\delta,\delta} - \overline{\mathcal{A}'}_{\log\delta,\delta}) - (\mathcal{A}_{\log\delta,\delta} - \overline{\mathcal{A}}_{\log\delta,\delta}) \\ &\quad + \mathcal{D}_{\log\delta,\delta} - \overline{\mathcal{D}}_{\log\delta,\delta} - 2(\mathcal{D}_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta-1,\frac{\delta}{2}}) - 2(\mathcal{D}_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}}_{\log\delta,\frac{\delta}{2}}) \\ &\quad - (\mathcal{D}'_{\log\delta,\delta} - \overline{\mathcal{D}'}_{\log\delta,\delta}) + 2(\mathcal{D}'_{\log\delta,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta,\frac{\delta}{2}}) \\ &\quad + 2(\mathcal{D}'_{\log\delta-1,\frac{\delta}{2}} - \overline{\mathcal{D}'}_{\log\delta-1,\frac{\delta}{2}} + 3(\frac{\delta}{2})^2\mathcal{N}_{\log\delta-1,\frac{\delta}{2}-1}). \end{aligned}$$

The recurrence in Lemma 3.10 (for arbitrary δ) with the basis give the desired closed-form solution for $L_\delta(Z^2_k)$. ∎

3.2 Locality Measures of 3-Dimensional Space-Filling Curve Families

For measures L_δ in the case of dimensionality 3, we derive the exact formulas for the canonical Hilbert and z-order curve families for $\delta = 1$. The canonical Hilbert and z-order curve are shown in Figures 3.12 and 3.13 (see Chapter II for the constructions). For these types of 3-dimensional self-similar curves, the self-similar structural property guides us to decompose C^3_k into eight identical C^3_{k-1}-subcurves (via reflection and/or rotation), which are amalgamated together by an C^3_1-curve. Following the linear order along this C^3_1-curve, the eight C^3_{k-1}-subcurves are $\{Q_\alpha(C^3_k) | \alpha = 1, 2, \ldots, 8\}$. The directions of rotations about axes are shown in Figure 3.14: arrow denotes (+)-rotation and its reverse denotes (−)-rotation.

3.2.1 Approach

The recursive decomposition (in k) of C^3_k gives that

$$\begin{aligned} L_\delta(C^3_k) &= \sum_{i,j\in[2^{3k}] | i<j \text{ and } d_1(C^3_k(i),C^3_k(j))=\delta} |i-j| \\ &= 8L_\delta(C^3_{k-1}) + \sum_{\alpha,\beta\in\{1,2,3,4,5,6,7,8\} | \alpha<\beta} \Delta_\delta(Q_\alpha(C^3_k), Q_\beta(C^3_k)), \end{aligned}$$

where $\Delta_\delta(Q_\alpha(C^3_k), Q_\beta(C^3_k))$ denotes the cumulative contribution of $|i-j|$ from the two subcurves $Q_\alpha(C^3_k)$ and $Q_\beta(C^3_k)$, that is, for all $i, j \in [2^{3k}]$ such that $i < j$,

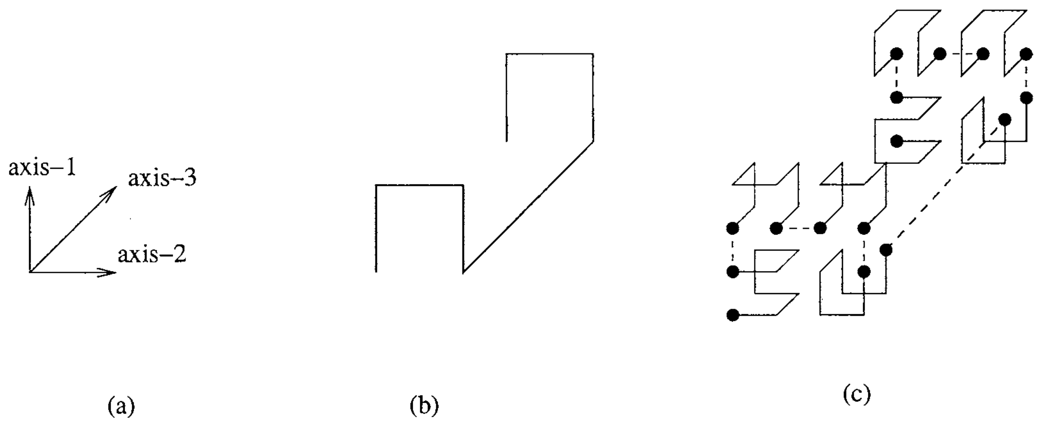


Figure 3.12: Canonical 3-dimensional Hilbert curves. (a) Coordinate system; (b) canonical H_1^3; (c) canonical H_2^3.

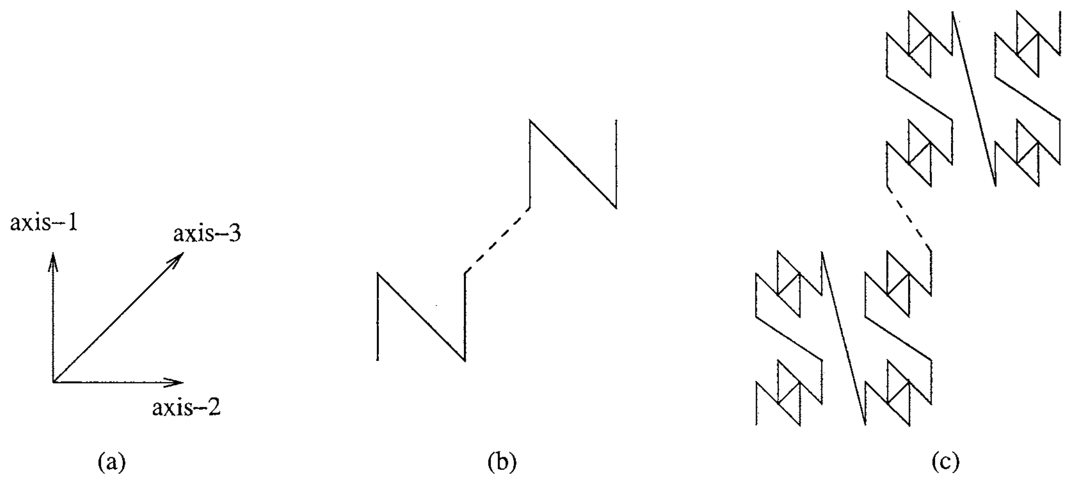


Figure 3.13: Canonical 3-dimensional z-order curves. (a) Coordinate system; (b) canonical Z_1^3; (c) canonical Z_2^3.

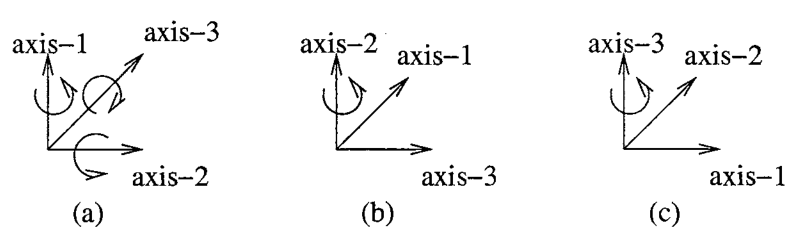


Figure 3.14: Demonstration for (+)-rotations about axes (arrows represent (+)-rotations about axes) in 3-dimensional space. (a) All the (+)-rotations about axes; (b) (+)-rotation about axis-2 viewed from different direction; (c) (+)-rotation about axis-3 viewed from different direction.

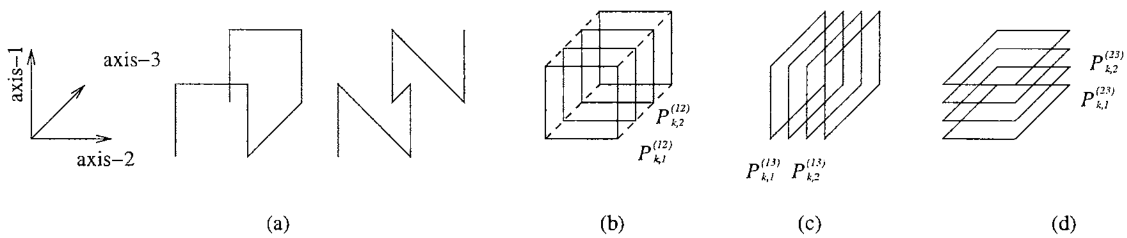


Figure 3.15: Structures of planes in 3-dimensional space: (a) Coordinate system and two space-filling curves: H_1^3, Z_1^3; (b) $P_{k,\alpha}^{(12)}$; (c) $P_{k,\alpha}^{(13)}$; (b) $P_{k,\alpha}^{(23)}$.

$d_1(C_k^3(i), C_k^3(j)) = \delta$, and i and j appear in (the index ranges of) $Q_\alpha(C_k^3)$ and $Q_\beta(C_k^3)$, respectively.

With respect to the canonical orientation of C_k^3 shown in Figure 3.15(a), we cover the 3-dimensional k-order grid with:

1. 2^k planes $(P_{k,1}^{(12)}, P_{k,2}^{(12)}, \ldots, P_{k,2^k}^{(12)})$, indexed by the coordinate of axis-3 (i.e., $P_{k,\alpha}^{(12)} = \{v | v \in [2^k]^3$ and the coordinate in axis-3 by v is $\alpha\}$) (see Figure 3.15(b)),

2. 2^k planes $(P_{k,1}^{(13)}, P_{k,2}^{(13)}, \ldots, P_{k,2^k}^{(13)})$, indexed by the coordinate of axis-2 (i.e., $P_{k,\alpha}^{(13)} = \{v | v \in [2^k]^3$ and the coordinate in axis-2 by v is $\alpha\}$) (see Figure 3.15(c)),

3. 2^k planes $(P_{k,1}^{(23)}, P_{k,2}^{(23)}, \ldots, P_{k,2^k}^{(23)})$, indexed by the coordinate of axis-1 (i.e., $P_{k,\alpha}^{(23)} = \{v | v \in [2^k]^3$ and the coordinate in axis-1 by v is $\alpha\}$) (see Figure 3.15(d)).

For $\alpha \in [2^k]$ and a grid point $p \in [2^k]^3$, we denote:

1. $\hbar_k(v, v') = |(C_k^3)^{-1}(v) - (C_k^3)^{-1}(v')|$, the index-difference between two grid points $v, v' \in [2^k]^3$.

2. $\Delta(X_{k,\alpha}, p) = \sum_{v \in X_{k,\alpha}} \hbar_k(v, p)$, where the symbol $X_{k,\alpha}$ denotes $P_{k,\alpha}^{(12)}, P_{k,\alpha}^{(13)}$, or $P_k^{(23)}$ (for example, $\Delta(P_{k,\alpha}^{(12)}, p) = \sum_{v \in P_{k,\alpha}^{(12)}} \hbar_k(v, p)$). That is, $\Delta(X_{k,\alpha}, p)$ cumulates all index-differences of all grid points in the structure $X_{k,\alpha}$ with respect to p; when $p = \partial_1(C_k^3)$, $\Delta(X_{k,\alpha}, p)$ is the index-cumulation of all grid points in $X_{k,\alpha}$.

3.2.2 Derivation of 3-Dimensional Hilbert Curve Family

In this section, we work on 3-dimensional Hilbert curve, and C_k^3 in Section 3.2.1 is replaced by H_k^3 now. Since we consider the 3-dimensional Hilbert curve that its quadrants 5,6,7,8 are the reflection of quadrants 4,3,2,1, respectively (see Chapter II), the mirror pairs for 3-dimensional space are defined with respect to the axis-3: for p in a canonical H_k^3, we have $\hbar_k(p, \partial_1(H_k^3)) + \hbar_k(p, \partial_2(H_k^3)) = 2^{3k} - 1$, and there exists a unique grid point p' (of same coordinates in axis-1 and axis-2 in H_k^3) such that $\hbar_k(p, \partial_1(H_k^3)) = \hbar_k(p', \partial_2(H_k^3))$.

$$\hbar_k(p, \partial_1(H_k^3)) + \hbar_k(p', \partial_1(H_k^3)) = \hbar_k(p, \partial_2(H_k^3)) + \hbar_k(p', \partial_2(H_k^3)) = 2^{3k} - 1.$$

The following three lemmas study the cumulation of indices of grid points in the planes of H_k^3.

Lemma 3.13 *The index-cumulation for the planes of H_k^3:*

1. For the planes $P_{k,\alpha}^{(13)}$ and $P_{k,\alpha}^{(23)}$, where $\alpha \in [2^k]$,

$$\begin{aligned}\Delta(P_{k,\alpha}^{(13)}, \partial_1(H_k^3)) &= \Delta(P_{k,\alpha}^{(13)}, \partial_2(H_k^3)) = \Delta(P_{k,\alpha}^{(23)}, \partial_1(H_k^3)) = \Delta(P_{k,\alpha}^{(23)}, \partial_2(H_k^3)) \\ &= \frac{1}{2} \cdot 2^{5k} - \frac{1}{2} \cdot 2^{2k},\end{aligned}$$

2. For the planes $P_{k,\alpha}^{(12)}$, where $\alpha \in [2^k]$,

$$\Delta(P_{k,\alpha}^{(12)}, \partial_1(H_k^3)) + \Delta(P_{k,2^k+1-\alpha}^{(12)}, \partial_1(H_k^3)) = 2^{2k}(2^{3k} - 1),$$

and

3. For the planes $P_{k,\alpha}^{(12)}$, where $\alpha \in [2^k]$,

$$\Delta(P_{k,\alpha}^{(12)}, \partial_1(H_k^3)) = \Delta(P_{k,2^k+1-\alpha}^{(12)}, \partial_2(H_k^3)).$$

Proof. Note that a canonical H_k^3 is inside-outside symmetric (with respect to axis-3).

Part 1: For a grid point $v \in P^{(13)}_{k,\alpha} \cap P^{(23)}_{k,\beta} \cap P^{(12)}_{k,\gamma}$, where $\alpha, \beta, \gamma \in [2^k]$, its mirror point $v' \in P^{(13)}_{k,\alpha} \cap P^{(23)}_{k,\beta} \cap P^{(12)}_{k,2^k+1-\gamma}$, and the mirror pair (v, v') satisfies that:

$$\hbar_k(v, \partial_1(H^3_k)) + \hbar_k(v', \partial_1(H^3_k)) = \hbar_k(v, \partial_2(H^3_k)) + \hbar_k(v', \partial_2(H^3_k)) = 2^{3k} - 1.$$

For every $\alpha \in [2^k]$, there are 2^{2k-1} mirror pairs in the plane $P^{(13)}_{k,\alpha}$. Thus,

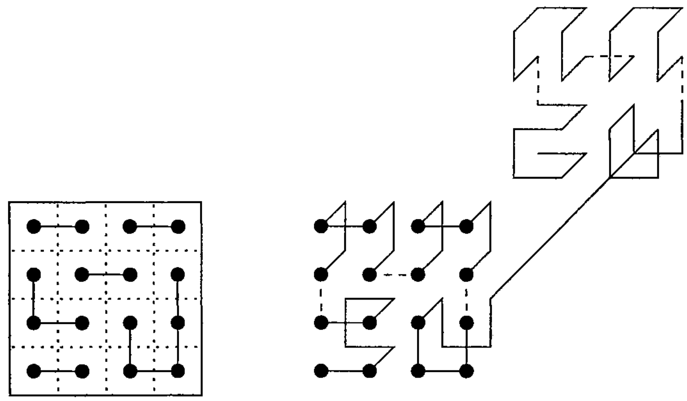
$$\begin{aligned}
& \Delta(P^{(13)}_{k,\alpha}, \partial_1(H^3_k)) = \sum_{v \in P^{13}_{k,\alpha}} \hbar_k(v, \partial_1(H^3_k)) \\
(= & \sum_{v' \in P^{13}_{k,\alpha}} \hbar_k(v', \partial_2(H^3_k)) = \Delta(P^{(13)}_{k,\alpha}, \partial_2(H^3_k)) \qquad (v' \text{is the mirror point of } v)) \\
= & \sum_{\text{all mirror pairs } (v,v') \text{ in } P^{(13)}_{k,\alpha}} (\hbar_k(v, \partial_1(H^3_k)) + \hbar_k(v', \partial_1(H^3_k))) \\
(= & \sum_{\text{all mirror pairs } (v,v') \text{ in } P^{(13)}_{k,\alpha}} (\hbar_k(v, \partial_2(H^3_k)) + \hbar_k(v', \partial_2(H^3_k)))) \\
= & 2^{2k-1}(2^{3k} - 1) = \frac{1}{2} \cdot 2^{5k} - \frac{1}{2} \cdot 2^{2k}.
\end{aligned}$$

Similarly, for points in $P^{(23)}_{k,\alpha}$, the mirror pairs are in the same plane, so they have the same cumulation of index differences. That is,

$$\begin{aligned}
\Delta(P^{(13)}_{k,\alpha}, \partial_1(H^3_k)) &= \Delta(P^{(13)}_{k,\alpha}, \partial_2(H^3_k)) = \Delta(P^{(23)}_{k,\alpha}, \partial_1(H^3_k)) = \Delta(P^{(23)}_{k,\alpha}, \partial_2(H^3_k)) \\
&= \frac{1}{2} \cdot 2^{5k} - \frac{1}{2} \cdot 2^{2k},
\end{aligned}$$

Part 2: $P^{(12)}_{k,\alpha}$ and $P^{(12)}_{k,2^k+1-\alpha}$ are reflective planes (inside-outside symmetric with respect to axis-3). So, for every point $v \in P^{(12)}_{k,\alpha}$, its mirror point $v' \in P^{(12)}_{k,2^k+1-\alpha}$, $\hbar_k(v, \partial_1(H^3_k)) + \hbar_k(v', \partial_1(H^3_k)) = 2^{3k} - 1$. Thus,

$$\begin{aligned}
& \Delta(P^{(12)}_{k,\alpha}, \partial_1(H^3_k)) + \Delta(P^{(12)}_{k,2^k+1-\alpha}, \partial_1(H^3_k)) \\
= & \sum_{v \in P^{(12)}_{k,\alpha}} \hbar_k(v, \partial_1(H^3_k)) + \sum_{v' \in P^{(12)}_{k,2^k+1-\alpha}} \hbar_k(v', \partial_1(H^3_k)) \\
= & \sum_{\text{all mirror pairs } (v,v') \in P^{(12)}_{k,\alpha} \times P^{(12)}_{k,2^k+1-\alpha}} (\hbar_k(v, \partial_1(H^3_k)) + \hbar_k(v', \partial_1(H^3_k))) \\
= & 2^{2k}(2^{3k} - 1).
\end{aligned}$$



(a) (b)

Figure 3.16: The plane structure for $P_{k,1}^{(12)}$ in a canonical H_k^3. (a) The structure for $P_{k,1}^{(12)}$; (b) the structures of underlying quadrants for $P_{k,1}^{(12)}$.

Part 3: $P_{k,\alpha}^{(12)}$ and $P_{k,2^k+1-\alpha}^{(12)}$ are reflective planes (inside-outside symmetric with respect to axis-3). So, for every point $v \in P_{k,\alpha}^{(12)}$, its mirror point $v' \in P_{k,2^k+1-\alpha}^{(12)}$and $\hbar_k(v, \partial_1(H_k^3)) = \hbar_k(v', \partial_2(H_k^3))$. Thus,

$$\begin{aligned} & \Delta(P_{k,\alpha}^{(12)}, \partial_1(H_k^3)) \\ = & \sum_{v \in P_{k,\alpha}^{(12)}} \hbar_k(v, \partial_1(H_k^3)) = \sum_{v' \in P_{k,2^k+1-\alpha}^{(12)}} \hbar_k(v', \partial_2(H_k^3)) \\ = & \Delta(P_{k,2^k+1-\alpha}^{(12)}, \partial_1(H_k^3)). \end{aligned}$$

∎

In computing $L_1(H_k^3)$, the points involved in the point-pairs across quadrants are in the exterior planes of the quadrants that these exterior planes neighbor to each other. We call these exterior planes the boundary planes.

Lemma 3.14 *For a canonical H_k^3, $\Delta(P_{k,1}^{(12)}, \partial_1(H_k^3)) = \frac{3 \cdot 5}{2 \cdot 31} \cdot 2^{5k} - \frac{1}{2} \cdot 2^{2k} + \frac{2^3}{31}$.* **Proof.** Figure 3.16(a) shows the exterior plane $P_{k,1}^{(12)}$, and Figure 3.16(b) illustrates the structures of the underlying quadrants. We see that the plane of the first quadrant is the structure of $P_{k-1,1}^{(13)}$, those of the second and the third quadrants are the structures of $P_{k-1,1}^{(23)}$, and that of the fourth quadrant is the structure of $P_{k-1,1}^{(12)}$. Thus, we can decompose the exterior plane into exterior planes of four quadrants $Q_\alpha(H_k^3)$, where

$\alpha \in [4]$, and the cumulation of index-adjustment from $\partial_1(Q_\alpha(H_k^3))$ to $\partial_1(H_k^3)$. We establish a recurrence (in k) for $\Delta(P_{k,1}^{(12)}, \partial_1(H_k^3))$ as follows:

$$
\begin{aligned}
\Delta(P_{k,1}^{(12)}, \partial_1(H_k^3)) &= \sum_{v \in P_{k,1}^{(12)}} \hbar_k(v, \partial_1(H_k^3)) \\
= & \sum_{v \in P_{k,1}^{(12)} \cap Q_1(H_k^3)} \hbar_k(v, \partial_1(H_k^3)) + \sum_{v \in P_{k,1}^{(12)} \cap Q_2(H_k^3)} \hbar_k(v, \partial_1(H_k^3)) \\
& + \sum_{v \in P_{k,1}^{(12)} \cap Q_3(H_k^3)} \hbar_k(v, \partial_1(H_k^3)) + \sum_{v \in P_{k,1}^{(12)} \cap Q_4(H_k^3)} \hbar_k(v, \partial_1(H_k^3)) \\
= & \sum_{v \in P_{k,1}^{(12)} \cap Q_1(H_k^3)} (\hbar_k(v, \partial_1(Q_1(H_k^3))) + \hbar_k(\partial_1(Q_1(H_k^3)), \partial_1(H_k^3))) \\
& + \sum_{v \in P_{k,1}^{(12)} \cap Q_2(H_k^3)} (\hbar_k(v, \partial_1(Q_2(H_k^3))) + \hbar_k(\partial_1(Q_2(H_k^3)), \partial_1(H_k^3))) \\
& + \sum_{v \in P_{k,1}^{(12)} \cap Q_3(H_k^3)} (\hbar_k(v, \partial_1(Q_3(H_k^3))) + \hbar_k(\partial_1(Q_3(H_k^3)), \partial_1(H_k^3))) \\
& + \sum_{v \in P_{k,1}^{(12)} \cap Q_4(H_k^3)} (\hbar_k(v, \partial_1(Q_4(H_k^3))) + \hbar_k(\partial_1(Q_4(H_k^3)), \partial_1(H_k^3))) \\
= & \sum_{v \in P_{k-1,1}^{(13)}} \hbar_{k-1}(v, \partial_1(H_{k-1}^3)) + 0 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)} \\
& \qquad \text{(after } (-\tfrac{\pi}{2})\text{-rotating } Q_1(H_k^3) \text{ about axis-1 and then} \\
& \qquad (-\tfrac{\pi}{2})\text{-rotating it about axis-2 into a canonical } H_{k-1}^3) \\
& + \sum_{v \in P_{k-1,1}^{(23)}} \hbar_{k-1}(v, \partial_1(H_{k-1}^3)) + 1 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)} \\
& \qquad \text{(after } (+\tfrac{\pi}{2})\text{-rotating } Q_2(H_k^3) \text{ about axis-1 and then} \\
& \qquad (+\tfrac{\pi}{2})\text{-rotating it about axis-3 into a canonical } H_{k-1}^3) \\
& + \sum_{v \in P_{k-1,1}^{(23)}} \hbar_{k-1}(v, \partial_1(H_{k-1}^3)) + 2 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)} \\
& \qquad \text{(after } (+\tfrac{\pi}{2})\text{-rotating } Q_3(H_k^3) \text{ about axis-1 and then} \\
& \qquad (+\tfrac{\pi}{2})\text{rotating it about axis-3 into a canonical } H_{k-1}^3) \\
& + \sum_{v \in P_{k-1,1}^{(12)}} \hbar_{k-1}(v, \partial_1(H_{k-1}^3)) + 3 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)}
\end{aligned}
$$

(after $(+\pi)$-rotating $Q_4(H_k^3)$ into a canonical H_{k-1}^3)

$$
\begin{aligned}
&= (\Delta(P_{k-1,1}^{(13)}, \partial_1(H_{k-1}^3)) + 0 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)}) \\
&\quad +(\Delta(P_{k-1,1}^{(23)}, \partial_1(H_{k-1}^3)) + 1 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)}) \\
&\quad +(\Delta(P_{k-1,1}^{(23)}, \partial_1(H_{k-1}^3)) + 2 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)}) \\
&\quad +(\Delta(P_{k-1,1}^{(12)}, \partial_1(H_{k-1}^3)) + 3 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)}) \\
&= \Delta(P_{k-1,1}^{(12)}, \partial_1(H_{k-1}^3)) + 3(\frac{1}{2} \cdot 2^{5(k-1)} - \frac{1}{2} \cdot 2^{2(k-1)}) + 6 \cdot 2^{5(k-1)} \\
&= \Delta(P_{k-1,1}^{(12)}, \partial_1(H_{k-1}^3)) + \frac{15}{2} \cdot 2^{5(k-1)} - \frac{3}{2} \cdot 2^{2(k-1)}.
\end{aligned}
$$

Iterating the recurrence in descending k (to 1) with $\Delta(P_{1,1}^{(12)}, \partial_1(H_1^3)) = 6$, we have:

$$
\begin{aligned}
\Delta(P_{k,1}^{(12)}, \partial_1(H_k^3)) &= \Delta(P_{1,1}^{(12)}, \partial_1(H_1^3)) + \sum_{\eta=1}^{k-1}(\frac{15}{2} \cdot 2^{5\eta} - \frac{3}{2} \cdot 2^{2\eta}) \\
&= \frac{3 \cdot 5}{2 \cdot 31} \cdot 2^{5k} - \frac{1}{2} \cdot 2^{2k} + \frac{2^3}{31}.
\end{aligned}
$$

∎

Now we partition the summation $\sum_{\alpha,\beta\in[8]|\alpha<\beta} \Delta_1(Q_\alpha(H_k^3), Q_\beta(H_k^3))$ according to the three cases: (1) for $\alpha, \beta \in \{1,2,3,4\}$, contiguous subcurves ($\alpha+1 \equiv \beta \pmod 4$) with four similar subcases, (2) for $\alpha, \beta \in \{5,6,7,8\}$, contiguous subcurves ($\alpha + 1 \equiv \beta \pmod 4$) with four similar subcases (in which $\Delta_1(Q_\alpha(H_k^3), Q_\beta(H_k^3))$ is same as the corresponding subcase of the contiguous subcurves in case (1) because of reflective (inside-outside symmetric with respect to axis-3) structures), and (3) for $\alpha, \beta \in \{1,2,3,4,5,6,7,8\}$, reflective (inside-outside symmetric with respect to axis-3) subcurves ($\alpha + \beta = 9$) with four similar subcases.

Lemma 3.15 *For a canonical* H_k^3*,*

$$
\begin{aligned}
&\Delta_1(Q_1(H_k^3), Q_4(H_k^3)) = \Delta_1(Q_5(H_k^3), Q_8(H_k^3)) \\
&= \Delta(P_{k-1,2^{k-1}}^{(23)}, \partial_2(H_{k-1}^3)) + \Delta(P_{k-1,2^{k-1}}^{(13)}, \partial_1(H_{k-1}^3)) + 2^{2k-2}(2 \cdot 2^{3(k-1)} + 1) \\
&= 2(\frac{1}{2} \cdot 2^{5(k-1)} - \frac{1}{2} \cdot 2^{2(k-1)}) + 2^{2k-2}(2 \cdot 2^{3(k-1)} + 1),
\end{aligned}
$$

$$
\Delta_1(Q_1(H_k^3), Q_2(H_k^3)) = \Delta_1(Q_7(H_k^3), Q_8(H_k^3))
$$

$$= \Delta(P^{(12)}_{k-1,2^{k-1}}, \partial_2(H^3_{k-1})) + \Delta(P^{(13)}_{k-1,1}, \partial_1(H^3_{k-1})) + 2^{2k-2}(1)$$
$$= (\Delta(P^{(12)}_{k-1,1}, \partial_1(H^3_{k-1}))) + (\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + (2^{2k-2}),$$

$$\Delta_1(Q_2(H^3_k), Q_3(H^3_k)) = \Delta_1(Q_6(H^3_k), Q_7(H^3_k))$$
$$= \Delta(P^{(12)}_{k-1,2^{k-1}}, \partial_2(H^3_{k-1})) + \Delta(P^{(12)}_{k-1,1}, \partial_1(H^3_{k-1})) + 2^{2k-2}(1)$$
$$= 2\Delta(P^{(12)}_{k-1,1}, \partial_1(H^3_{k-1})) + (2^{2k-2}),$$

$$\Delta_1(Q_3(H^3_k), Q_4(H^3_k)) = \Delta_1(Q_5(H^3_k), Q_6(H^3_k))$$
$$= \Delta(P^{(13)}_{k-1,1}, \partial_2(H^3_{k-1})) + \Delta(P^{(23)}_{k-1,1}, \partial_1(H^3_{k-1})) + 2^{2k-2}(1)$$
$$= 2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + (2^{2k-2}),$$

$$\Delta_1(Q_1(H^3_k), Q_8(H^3_k))$$
$$= \Delta(P^{(13)}_{k-1,2^{k-1}}, \partial_2(H^3_{k-1})) + \Delta(P^{(13)}_{k-1,2^{k-1}}, \partial_1(H^3_{k-1})) + 2^{2k-2}(6\cdot 2^{3(k-1)}+1)$$
$$= 2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + 2^{2k-2}(6\cdot 2^{3(k-1)}+1),$$

$$\Delta_1(Q_2(H^3_k), Q_7(H^3_k))$$
$$= \Delta(P^{(23)}_{k-1,2^{k-1}}, \partial_2(H^3_{k-1})) + \Delta(P^{(23)}_{k-1,2^{k-1}}, \partial_1(H^3_{k-1})) + 2^{2k-2}(4\cdot 2^{3(k-1)}+1)$$
$$= 2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + 2^{2k-2}(4\cdot 2^{3(k-1)}+1),$$

$$\Delta_1(Q_3(H^3_k), Q_6(H^3_k))$$
$$= \Delta(P^{(23)}_{k-1,2^{k-1}}, \partial_2(H^3_{k-1})) + \Delta(P^{(23)}_{k-1,2^{k-1}}, \partial_1(H^3_{k-1})) + 2^{2k-2}(2\cdot 2^{3(k-1)}+1)$$
$$= 2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + 2^{2k-2}(2\cdot 2^{3(k-1)}+1),$$

and

$$\Delta_1(Q_4(H^3_k), Q_5(H^3_k))$$
$$= \Delta(P^{(12)}_{k-1,2^{k-1}}, \partial_2(H^3_{k-1})) + \Delta(P^{(12)}_{k-1,1}, \partial_1(H^3_{k-1})) + 2^{2k-2}(1)$$
$$= 2\Delta(P^{(12)}_{k-1,1}, \partial_1(H^3_{k-1})) + 2^{2k-2}.$$

Proof. The derivations are straightforward. The point-pairs across quadrants involved in computing $L_1(H_k^3)$ are in the boundary planes. So, the cumulation of the index differences for $\Delta_1(Q_\alpha(H_k^3), Q_\beta(H_k^3))$, where $\alpha < \beta$, is the cumulations of index differences for points in the boundary plane of $Q_\alpha(H_k^3)$ respect to $\partial_2(Q_\alpha(H_k^3))$, those for points in the boundary plane of $Q_\beta(H_k^3)$ respect to $\partial_1(Q_\beta(H_k^3))$, and those between $\partial_2(Q_\alpha(H_k^3))$ and $\partial_1(Q_\beta(H_k^3))$. For $\Delta_1(Q_1(H_k^3), Q_4(H_k^3))$,

$$\begin{aligned}
&\Delta_1(Q_1(H_k^3), Q_4(H_k^3)) \\
= &\sum_{v \in P^{(13)}_{k,2^k-1} \cap Q_1(H_k^3)} \hbar_k(v, \partial_2(Q_1(H_k^3))) + \sum_{v' \in P^{(13)}_{k,2^{k-1}+1} \cap Q_4(H_k^3)} \hbar_k(v, \partial_1(Q_4(H_k^3))) \\
&+ \sum_{v \in P^{(13)}_{k,2^k-1} \cap Q_1(H_k^3)} \hbar_k(\partial_2(Q_1(H_k^3)), \partial_1(Q_4(H_k^3))) \\
= &\sum_{v \in P^{(23)}_{k-1,2^k-1}} \hbar_{k-1}(v, \partial_2(H_{k-1}^3)) \\
&\qquad \text{(after } (-\frac{\pi}{2})\text{-rotating } Q_1(H_k^3) \text{ about axis-1 and then} \\
&\qquad (-\frac{\pi}{2})\text{rotating it about axis-2 into a canonical } H_{k-1}^3) \\
&+ \sum_{v' \in P^{(13)}_{k-1,2^k-1}} \hbar_{k-1}(v', \partial_1(H_{k-1}^3)) \\
&\qquad \text{(after } (+\pi)\text{-rotating } Q_4(H_k^3) \text{ into a canonical } H_{k-1}^3) \\
&+ \sum_{v \in P^{(23)}_{k-1,2^k-1}} (2 \cdot 2^{3(k-1)} + 1) \\
= &\ \Delta(P^{(23)}_{k-1,2^{k-1}}, \partial_2(H_{k-1}^3)) + \Delta(P^{(13)}_{k-1,2^{k-1}}, \partial_1(H_{k-1}^3)) + 2^{2k-2}(2 \cdot 2^{3(k-1)} + 1) \\
= &\ 2(\frac{1}{2} \cdot 2^{5(k-1)} - \frac{1}{2} \cdot 2^{2(k-1)}) + 2^{2k-2}(2 \cdot 2^{3(k-1)} + 1).
\end{aligned}$$

Derivations for other cases are similar. ∎

Theorem 3.3 *For a 3-dimensional Hilbert curve family,*

$$L_1(H_k^3) = \frac{67}{2 \cdot 31} \cdot 2^{5k} - \frac{11}{2 \cdot 7} \cdot 2^{3k} - \frac{2^6}{7 \cdot 31}.$$

Proof. For a 3-dimensional Hilbert curve family,

$$\begin{aligned} L_\delta(H_k^3) &= \sum_{i,j\in[2^{3k}]\mid i<j \text{and} d_1(H_k^3(i),H_k^3(j))=\delta} |i-j| \\ &= 8L_\delta(H_{k-1}^3) + \sum_{\alpha,\beta\in\{1,2,3,4,5,6,7,8\}\mid\alpha<\beta} \Delta_\delta(Q_\alpha(H_k^3), Q_\beta(H_k^3)). \end{aligned}$$

By Lemma 3.15, for $\delta = 1$, the cumulation of index differences:

$$\begin{aligned} L_1(H_k^3) &= 8L_1(H_{k-1}^3) + \sum_{\alpha,\beta\in\{1,2,3,4,5,6,7,8\}\mid\alpha<\beta} \Delta_1(Q_\alpha(H_k^3), Q_\beta(H_k^3)) \\ &= 8L_1(H_{k-1}^3) \\ &\quad +2(2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + 2^{2k-2}(2\cdot 2^{3(k-1)}+1)) \\ &\quad +2(\Delta(P_{k-1,1}^{(12)}, \partial_1(H_{k-1}^3)) + (\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + (2^{2k-2})) \\ &\quad +2(2\Delta(P_{k-1,1}^{(12)}, \partial_1(H_{k-1}^3)) + (2^{2k-2})) \\ &\quad +2(2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + (2^{2k-2})) \\ &\quad +(2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + 2^{2k-2}(6\cdot 2^{3(k-1)}+1)) \\ &\quad +(2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + 2^{2k-2}(4\cdot 2^{3(k-1)}+1)) \\ &\quad +(2(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) + 2^{2k-2}(2\cdot 2^{3(k-1)}+1)) \\ &\quad +(2\Delta(P_{k-1,1}^{(12)}, \partial_1(H_{k-1}^3)) + (2^{2k-2})) \\ &= 8L_1(H_{k-1}^3) + 8\Delta(P_{k-1,1}^{(12)}, \partial_1(H_{k-1}^3)) + 16(\frac{1}{2}\cdot 2^{5(k-1)} - \frac{1}{2}\cdot 2^{2(k-1)}) \\ &\quad +(16\cdot 2^{3(k-1)}+12)(2^{2k-2}) \\ &= 8L_1(H_{k-1}^3) + \frac{201}{248}\cdot 2^{5k} + \frac{64}{31}. \end{aligned}$$

We establish a recurrence (in k) for $L_1(H_k^3)$ as follows:

$$L_1(H_k^3) = \begin{cases} 8L_1(H_{k-1}^3) + \frac{201}{248}\cdot 2^{5k} + \frac{64}{31} & \text{if } k>1 \\ 28 & \text{if } k=1, \end{cases}$$

which yields the desired closed-form solution for $L_1(H_k^3)$. ∎

3.2.3 Derivation of 3-Dimensional z-Order Curve Family

In this section, we focus on 3-dimensional z-order curve, and the notations for C_k^3 in Section 3.2.1 is now for Z_k^3. Since we consider the 3-dimensional space, the mirror pairs for z-order curve are described as follow: for p with coordinate (α, β, γ) in a canonical Z_k^3, we have $\hbar_k(p, \partial_1(Z_k^3)) + \hbar_k(p, \partial_2(Z_k^3)) = 2^{3k} - 1$, and there exists a unique grid point p' (with coordinate $(2^k+1-\gamma, 2^k+1-\beta, 2^k+1-\alpha)$ in Z_k^3 such that $\hbar_k(p, \partial_1(Z_k^3)) = \hbar_k(p', \partial_2(Z_k^3))$. (See constructing z-order in Chapter II for details.)

$$\hbar_k(p, \partial_1(Z_k^3)) + \hbar_k(p', \partial_1(Z_k^3)) = \hbar_k(p, \partial_2(Z_k^3)) + \hbar_k(p', \partial_2(Z_k^3)) = 2^{3k} - 1.$$

The following three lemmas study the cumulation of indices of grid points in the planes of Z_k^3.

Lemma 3.16 *The index-cumulation for the planes of* Z_k^3*:*

1. For the planes $P_{k,\alpha}^{(12)}$, $P_{k,\alpha}^{(13)}$, and $P_{k,\alpha}^{(23)}$, where $\alpha \in [2^k]$,

$$\begin{aligned} & \Delta(P_{k,\alpha}^{(12)}, \partial_1(Z_k^3)) + \Delta(P_{k,2^k+1-\alpha}^{(12)}, \partial_1(Z_k^3)) \\ = \; & \Delta(P_{k,\alpha}^{(13)}, \partial_1(Z_k^3)) + \Delta(P_{k,2^k+1-\alpha}^{(13)}, \partial_1(Z_k^3)) \\ = \; & \Delta(P_{k,\alpha}^{(23)}, \partial_1(Z_k^3)) + \Delta(P_{k,2^k+1-\alpha}^{(23)}, \partial_1(Z_k^3)) \\ = \; & 2^{2k}(2^{3k} - 1), \end{aligned}$$

and

2. For the planes $P_{k,\alpha}^{(12)}$, $P_{k,\alpha}^{(13)}$, and $P_{k,\alpha}^{(23)}$, where $\alpha \in [2^k]$,

$$\begin{aligned} \Delta(P_{k,\alpha}^{(12)}, \partial_1(Z_k^3)) &= \Delta(P_{k,2^k+1-\alpha}^{(12)}, \partial_2(Z_k^3)) \\ \Delta(P_{k,\alpha}^{(13)}, \partial_1(Z_k^3)) &= \Delta(P_{k,2^k+1-\alpha}^{(13)}, \partial_2(Z_k^3)) \\ \Delta(P_{k,\alpha}^{(23)}, \partial_1(Z_k^3)) &= \Delta(P_{k,2^k+1-\alpha}^{(23)}, \partial_2(Z_k^3)). \end{aligned}$$

Proof. Note that a canonical Z_k^3 is inside-outside symmetric (with respect to axis-3 via $+\pi$-rotation about axis-3).

Part 1: Planes $P^{(12)}_{k,\alpha}$ and $P^{(12)}_{k,2^k+1-\alpha}$ (after $+\pi$-rotated about axis-3) are reflective planes. So, for every point $v \in P^{(12)}_{k,\alpha}$, its mirror point $v' \in P^{(12)}_{k,2^k+1-\alpha}$, $\hbar_k(v, \partial_1(Z^3_k)) + \hbar_k(v', \partial_1(Z^3_k)) = 2^{3k} - 1$. Thus,

$$\begin{aligned} & \Delta(P^{(12)}_{k,\alpha}, \partial_1(Z^3_k)) + \Delta(P^{(12)}_{k,2^k+1-\alpha}, \partial_1(Z^3_k)) \\ = & \sum_{v \in P^{(12)}_{k,\alpha}} \hbar_k(v, \partial_1(Z^3_k)) + \sum_{v' \in P^{(12)}_{k,2^k+1-\alpha}} \hbar_k(v', \partial_1(Z^3_k)) \\ = & \sum_{\text{all mirror pairs } (v,v') \in P^{(12)}_{k,\alpha} \times P^{(12)}_{k,2^k+1-\alpha}} (\hbar_k(v, \partial_1(Z^3_k)) + \hbar_k(v', \partial_1(Z^3_k))) \\ = & \ 2^{2k}(2^{3k} - 1) = 2^{5k} - 2^{2k}. \end{aligned}$$

Similarly, Planes $P^{(13)}_{k,\alpha}$ and $P^{(13)}_{k,2^k+1-\alpha}$ (after $+\pi$-rotated along axis-2) are reflective planes, planes $P^{(23)}_{k,\alpha}$ and $P^{(23)}_{k,2^k+1-\alpha}$ (after $+\pi$-rotated along axis-1) are reflective planes. Thus,

$$\begin{aligned} \Delta(P^{(13)}_{k,\alpha}, \partial_1(Z^3_k)) + \Delta(P^{(13)}_{k,2^k+1-\alpha}, \partial_1(Z^3_k)) &= 2^{2k}(2^{3k} - 1) = 2^{5k} - 2^{2k}, \\ \Delta(P^{(23)}_{k,\alpha}, \partial_1(Z^3_k)) + \Delta(P^{(23)}_{k,2^k+1-\alpha}, \partial_1(Z^3_k)) &= 2^{2k}(2^{3k} - 1) = 2^{5k} - 2^{2k}. \end{aligned}$$

Part 2: $P^{(12)}_{k,\alpha}$ and $P^{(12)}_{k,2^k+1-\alpha}$ (after $+\pi$-rotated about axis-3) are reflective planes. So, for every point $v \in P^{(12)}_{k,\alpha}$, its mirror point $v' \in P^{(12)}_{k,2^k+1-\alpha}$and $\hbar_k(v, \partial_1(Z^3_k)) = \hbar_k(v', \partial_2(Z^3_k))$. Thus,

$$\begin{aligned} & \Delta(P^{(12)}_{k,\alpha}, \partial_1(Z^3_k)) \\ = & \sum_{v \in P^{(12)}_{k,\alpha}} \hbar_k(v, \partial_1(Z^3_k)) = \sum_{v' \in P^{(12)}_{k,2^k+1-\alpha}} \hbar_k(v', \partial_2(Z^3_k)) \\ = & \ \Delta(P^{(12)}_{k,2^k+1-\alpha}, \partial_1(Z^3_k)). \end{aligned}$$

In a similar way, $\Delta(P^{(13)}_{k,\alpha}, \partial_1(Z^3_k)) = \Delta(P^{(13)}_{k,2^k+1-\alpha}, \partial_2(Z^3_k))$ and $\Delta(P^{(23)}_{k,\alpha}, \partial_1(Z^3_k)) = \Delta(P^{(23)}_{k,2^k+1-\alpha}, \partial_2(Z^3_k))$. ∎

In computing $L_1(Z^3_k)$, the points involved in the point-pairs across quadrants are in the exterior planes of the quadrants that these exterior planes neighbor to each other. We call these exterior planes the boundary planes.

Lemma 3.17 *For a canonical Z_k^3,*

$$\begin{aligned}
\Delta(P_{k,1}^{(12)}, \partial_1(Z_k^3)) &= \frac{3}{2\cdot 7}\cdot 2^{5k} - \frac{3}{2\cdot 7}\cdot 2^{2k} \\
\Delta(P_{k,1}^{(13)}, \partial_1(Z_k^3)) &= \frac{5}{2\cdot 7}\cdot 2^{5k} - \frac{5}{2\cdot 7}\cdot 2^{2k} \\
\Delta(P_{k,1}^{(23)}, \partial_1(Z_k^3)) &= \frac{3}{7}\cdot 2^{5k} - \frac{3}{7}\cdot 2^{2k}.
\end{aligned}$$

Proof. Figure 3.17(a) shows the exterior plane $P_{k,1}^{(12)}$, and Figure 3.17(b) illustrates the structures of the underlying quadrants. It is obvious that the plane of the first quadrant is the structure of $P_{k-1,1}^{(13)}$, those of the second and the third quadrants are the structures of $P_{k-1,1}^{(23)}$, and that of the fourth quadrant is the structure of $P_{k-1,1}^{(12)}$. Thus, we can decompose the exterior plane into exterior planes of four quadrants $Q_\alpha(Z_k^3)$, where $\alpha \in [4]$, and the cumulations of index-adjustment from $\partial_1(Q_\alpha(Z_k^3))$ to $\partial_1(Z_k^3)$. A recurrence (in k) for $\Delta(P_{k,1}^{(12)}, \partial_1(Z_k^3))$ is established as follows:

$$\begin{aligned}
& \Delta(P_{k,1}^{(12)}, \partial_1(Z_k^3)) \\
= & \sum_{v\in P_{k,1}^{(12)}} \hbar_k(v, \partial_1(Z_k^3)) \\
= & \sum_{v\in P_{k,1}^{(12)}\cap Q_1(Z_k^3)} \hbar_k(v, \partial_1(Z_k^3)) + \sum_{v\in P_{k,1}^{(12)}\cap Q_2(Z_k^3)} \hbar_k(v, \partial_1(Z_k^3)) \\
& + \sum_{v\in P_{k,1}^{(12)}\cap Q_3(Z_k^3)} \hbar_k(v, \partial_1(Z_k^3)) + \sum_{v\in P_{k,1}^{(12)}\cap Q_4(Z_k^3)} \hbar_k(v, \partial_1(Z_k^3)) \\
= & \sum_{v\in P_{k,1}^{(12)}\cap Q_1(Z_k^3)} (\hbar_k(v, \partial_1(Q_1(Z_k^3))) + \hbar_k(\partial_1(Q_1(Z_k^3)), \partial_1(Z_k^3))) \\
& + \sum_{v\in P_{k,1}^{(12)}\cap Q_2(Z_k^3)} (\hbar_k(v, \partial_1(Q_2(Z_k^3))) + \hbar_k(\partial_1(Q_2(Z_k^3)), \partial_1(Z_k^3))) \\
& + \sum_{v\in P_{k,1}^{(12)}\cap Q_3(Z_k^3)} (\hbar_k(v, \partial_1(Q_3(Z_k^3))) + \hbar_k(\partial_1(Q_3(Z_k^3)), \partial_1(Z_k^3))) \\
& + \sum_{v\in P_{k,1}^{(12)}\cap Q_4(Z_k^3)} (\hbar_k(v, \partial_1(Q_4(Z_k^3))) + \hbar_k(\partial_1(Q_4(Z_k^3)), \partial_1(Z_k^3))) \\
= & \sum_{v\in P_{k-1,1}^{(12)}} \hbar_{k-1}(v, \partial_1(Z_{k-1}^3)) + 0\cdot 2^{2(k-1)}\cdot 2^{3(k-1)}
\end{aligned}$$

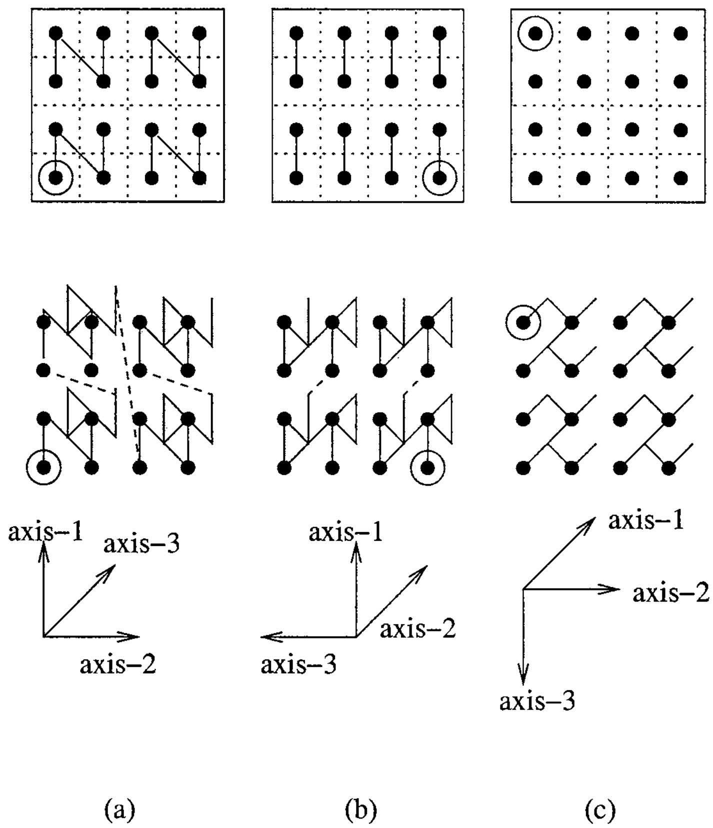


Figure 3.17: The structures of planes and their underlying quadrants for (a) $P_{k,1}^{(12)}$; (b) $P_{k,1}^{(13)}$; (c) $P_{k,1}^{(13)}$. (The circled solid circles denote the entry points.)

$$
\begin{aligned}
&+ \sum_{v \in P^{(12)}_{k-1,1}} \hbar_{k-1}(v, \partial_1(Z^3_{k-1})) + 1 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)} \\
&+ \sum_{v \in P^{(12)}_{k-1,1}} \hbar_{k-1}(v, \partial_1(Z^3_{k-1})) + 2 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)} \\
&+ \sum_{v \in P^{(12)}_{k-1,1}} \hbar_{k-1}(v, \partial_1(Z^3_{k-1})) + 3 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)} \\
&\text{(every quadrant is a canonical } Z^3_{k-1}) \\
= \ & 4\Delta(P^{(12)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 6 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)}.
\end{aligned}
$$

Iterating the recurrence in descending k (to 1) with $\Delta(P^{(12)}_{1,1}, \partial_1(Z^3_1)) = 6$, we have:

$$
\begin{aligned}
\Delta(P^{(12)}_{k,1}, \partial_1(Z^3_k)) &= 4^{k-1}\Delta(P^{(12)}_{1,1}, \partial_1(Z^3_1)) + \sum_{\eta=1}^{k-1} 4^{k-\eta-1}(6 \cdot 2^{5\eta}) \\
&= \frac{3}{2 \cdot 7} \cdot 2^{5k} - \frac{3}{2 \cdot 7} \cdot 2^{2k}.
\end{aligned}
$$

Similarly, we have

$$
\Delta(P^{(13)}_{k,1}, \partial_1(Z^3_k)) = \begin{cases} 4\Delta(P^{(13)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 10 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)} & \text{if } k > 1 \\ 10 & \text{if } k = 1, \end{cases}
$$

$$
\Delta(P^{(23)}_{k,1}, \partial_1(Z^3_k)) = \begin{cases} 4\Delta(P^{(23)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 12 \cdot 2^{2(k-1)} \cdot 2^{3(k-1)} & \text{if } k > 1 \\ 12 & \text{if } k = 1. \end{cases}
$$

These yield the closed-form solutions for $\Delta(P^{(13)}_{k,1}, \partial_1(Z^3_k))$ and $\Delta(P^{(23)}_{k,1}, \partial_1(Z^3_k))$. ∎

Now we partition the summation $\sum_{\alpha,\beta \in [8] | \alpha < \beta} \Delta_1(Q_\alpha(Z^3_k), Q_\beta(Z^3_k))$ according to the three cases: (1) for $\alpha, \beta \in \{1, 2, 3, 4\}$, contiguous subcurves $((\alpha, \beta) \in \{(1, 2), (1, 3), (2, 4), (3, 4)\})$ with four similar subcases, (2) for $\alpha, \beta \in \{5, 6, 7, 8\}$, contiguous subcurves $((\alpha, \beta) \in \{(5, 6), (5, 7), (6, 8), (7, 8)\})$ with four similar subcases (in which are same as the subcases of the contiguous subcurves in case (1) because of reflective (inside-outside symmetric with respect to axis-3 after $+\pi$-rotation) structures), and (3) for $\alpha, \beta \in \{1, 2, 3, 4, 5, 6, 7, 8\}$, diagonal subcurves $(\alpha + \beta = 9)$ with four similar subcases.

Lemma 3.18 *For a canonical* Z^3_k,

$$
\Delta_1(Q_1(Z^3_k), Q_3(Z^3_k)) = \Delta_1(Q_2(Z^3_k), Q_4(Z^3_k))
$$

$$= \Delta_1(Q_5(Z_k^3), Q_7(Z_k^3)) = \Delta_1(Q_6(Z_k^3), Q_8(Z_k^3))$$
$$= \Delta(P^{(13)}_{k-1,2^{k-1}}, \partial_2(Z^3_{k-1})) + \Delta(P^{(13)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(2^{3(k-1)}+1)$$
$$= 2\Delta(P^{(13)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(2^{3(k-1)}+1),$$

$$\Delta_1(Q_1(Z_k^3), Q_2(Z_k^3)) = \Delta_1(Q_3(Z_k^3), Q_4(Z_k^3))$$
$$= \Delta_1(Q_5(Z_k^3), Q_6(Z_k^3)) = \Delta_1(Q_7(Z_k^3), Q_8(Z_k^3))$$
$$= \Delta(P^{(23)}_{k-1,2^{k-1}}, \partial_2(Z^3_{k-1})) + \Delta(P^{(23)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(1)$$
$$= 2\Delta(P^{(23)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(1),$$

and,

$$\Delta_1(Q_1(Z_k^3), Q_5(Z_k^3)) = \Delta_1(Q_2(Z_k^3), Q_6(Z_k^3))$$
$$= \Delta_1(Q_3(Z_k^3), Q_7(Z_k^3)) = \Delta_1(Q_4(Z_k^3), Q_8(Z_k^3))$$
$$= \Delta(P^{(12)}_{k-1,2^{k-1}}, \partial_2(Z^3_{k-1})) + \Delta(P^{(12)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(3\cdot 2^{3(k-1)}+1)$$
$$= 2\Delta(P^{(12)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(3\cdot 2^{3(k-1)}+1),$$

Proof. The derivations are straightforward. The point-pairs across quadrants involved in computing L_1 are in the boundary planes. So, the cumulation of the index differences for $\Delta_1(Q_\alpha(Z_k^3), Q_\beta(Z_k^3))$, where $\alpha < \beta$, is the cumulations of index differences for points in the boundary plane of $Q_\alpha(Z_k^3)$ respect to $\partial_2(Q_\alpha(Z_k^3))$, those for points in the boundary plane of $Q_\beta(Z_k^3)$ respect to $\partial_1(Q_\beta(Z_k^3))$, and those between $\partial_2(Q_\alpha(Z_k^3))$ and $\partial_1(Q_\beta(Z_k^3))$. For $\Delta_1(Q_1(Z_k^3), Q_3(Z_k^3))$,

$$\Delta_1(Q_1(Z_k^3), Q_3(Z_k^3))$$
$$= \sum_{v\in P^{(13)}_{k,2^{k-1}}\cap Q_1(Z_k^3)} \hbar_k(v, \partial_2(Q_1(Z_k^3))) + \sum_{v'\in P^{(13)}_{k,2^{k-1}+1}\cap Q_3(Z_k^3)} \hbar_k(v, \partial_1(Q_3(Z_k^3)))$$
$$+ \sum_{v\in P^{(13)}_{k,2^{k-1}}\cap Q_1(Z_k^3)} \hbar_k(\partial_2(Q_1(Z_k^3)), \partial_1(Q_3(Z_k^3)))$$
$$= \sum_{v\in P^{(13)}_{k-1,2^{k-1}}} \hbar_{k-1}(v, \partial_2(Z^3_{k-1})) + \sum_{v'\in P^{(13)}_{k-1,1}} \hbar_{k-1}(v', \partial_1(Z^3_{k-1}))$$

(the connecting planes between 1st and 3rd subcurves

$$\text{are } P^{(13)}_{k-1,2^{k-1}}, P^{(13)}_{k-1,1}\ )$$

$$+ \sum_{v \in P^{(13)}_{k-1,2^{k-1}}} (1 \cdot 2^{3(k-1)} + 1)$$

$$= \Delta(P^{(13)}_{k-1,2^{k-1}}, \partial_2(Z^3_{k-1})) + \Delta(P^{(13)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(1 \cdot 2^{3(k-1)} + 1)$$

$$\text{(Lemma 3.16 (part 2): } \Delta(P^{(13)}_{k-1,2^{k-1}}, \partial_2(Z^3_{k-1})) = \Delta(P^{(13)}_{k-1,1}, \partial_1(Z^3_{k-1})))$$

$$= 2 \cdot \Delta(P^{(13)}_{k-1,2^{k-1}}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(2^{3(k-1)} + 1).$$

Derivations for other cases are similar. ∎

Theorem 3.4 *For a 3-dimensional z-order curve family,*

$$L_1(Z^3_k) = 2^{5k} - 2^{2k}.$$

Proof. For a 3-dimensional z-order curve family,

$$\begin{aligned} L_\delta(Z^3_k) &= \sum_{i,j \in [2^{3k}] | i<j \text{ and } d_1(Z^3_k(i), Z^3_k(j)) = \delta} |i-j| \\ &= 8L_\delta(Z^3_{k-1}) + \sum_{\alpha,\beta \in \{1,2,3,4,5,6,7,8\} | \alpha<\beta} \Delta_\delta(Q_\alpha(Z^3_k), Q_\beta(Z^3_k)). \end{aligned}$$

By Lemma 3.18, for $\delta = 1$, the cumulation of index differences:

$$L_1(Z^3_k) = 8L_1(Z^3_{k-1}) + \sum_{\alpha,\beta \in \{1,2,3,4,5,6,7,8\} | \alpha<\beta} \Delta_1(Q_\alpha(Z^3_k), Q_\beta(Z^3_k))$$

$$\begin{aligned} &= 8L_1(Z^3_{k-1}) \\ &\quad +4(2\Delta(P^{(13)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(2^{3(k-1)} + 1)) \\ &\quad +4(2\Delta(P^{(23)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(1)) \\ &\quad +4(2\Delta(P^{(12)}_{k-1,1}, \partial_1(Z^3_{k-1})) + 2^{2k-2}(3 \cdot 2^{3(k-1)} + 1)) \\ &= 8L_1(Z^3_{k-1}) + 8(\Delta(P^{(12)}_{k-1,1}, \partial_1(Z^3_{k-1})) + \Delta(P^{(13)}_{k-1,1}, \partial_1(Z^3_{k-1})) \\ &\quad +\Delta(P^{(23)}_{k-1,1}, \partial_1(Z^3_{k-1}))) + (16 \cdot 2^{3(k-1)} + 12)(2^{2k-2}) \\ &= 8L_1(Z^3_{k-1}) + \frac{3}{4} \cdot 2^{5k} + 2^{2k}. \end{aligned}$$

A recurrence (in k) for $L_1(Z^3_k)$ is established as follows:

$$L_1(Z^3_k) = \begin{cases} 8L_1(Z^3_{k-1}) + \frac{3}{4} \cdot 2^{5k} + 2^{2k} & \text{if } k > 1 \\ 28 & \text{if } k = 1, \end{cases}$$

which yields the desired closed-form solution for $L_1(Z_k^3)$. ∎

3.3 Comparison and Verification

We summarize our analyses of $L_\delta(H_k^m)$ and $L_\delta(Z_k^m)$ as follows:

$$L_\delta(H_k^m) = \begin{cases} \frac{17}{2\cdot 7}\cdot 2^{3k+2\log\delta} + O(2^{2k+3\log\delta}) & \text{for } m=2 \text{ and } \delta \text{ that is an integral power of 2} \\ \frac{67}{2\cdot 31}2^{5k} + O(2^{3k}) & \text{for } m=3 \text{ and } \delta=1, \end{cases}$$

and

$$L_\delta(Z_k^m) = \begin{cases} 2^{3k+2\log\delta} + O(2^{2k+3\log\delta}) & \text{for } m=2 \text{ and } \delta \text{ that is an integral power of 2} \\ 2^{5k} + O(2^{2k}) & \text{for } m=3 \text{ and } \delta=1. \end{cases}$$

Thus, for sufficiently large k and $\delta \ll 2^k$,

$$\frac{L_\delta(H_k^m)}{L_\delta(Z_k^m)} \approx \begin{cases} \frac{17}{2\cdot 7} \approx 1.2143 & \text{for } m=2 \text{ and } \delta \text{ that is an integral power of 2} \\ \frac{67}{2\cdot 31} \approx 1.0806 & \text{for } m=3. \end{cases}$$

With respect to the locality measure L_δ and for sufficiently large k and $\delta \ll 2^k$, the z-order curve family performs better than the Hilbert curve family for $m=2$ and over the δ-spectrum of integral powers of 2.

When $\delta = 2^k$, the domination reverses as:

$$L_\delta(H_k^2) = \frac{37}{240}\cdot 2^{5k} - \frac{1}{12}\cdot 2^{3k} - \frac{2}{15}\cdot 2^k,$$

and

$$L_\delta(Z_k^2) = \frac{107}{672}\cdot 2^{5k} - \frac{1}{12}\cdot 2^{3k} - \frac{3}{28}\cdot 2^{2k}.$$

These give that

$$\frac{L_\delta(H_k^m)}{L_\delta(Z_k^m)} \approx \frac{2\cdot 7\cdot 37}{5\cdot 107} \approx 0.9682.$$

The superiority of the z-order curve family persists but declines for $m=3$ with unit 1-normed distance for L_δ.

For the extreme case $m = 2$ and $\delta = 1$, the locality measure L_δ in our study degenerates to $L_{\text{MD},1}$ in [MD86]. Their analysis shows that for a 2-dimensional curve C for the grid $[n]^2$, $L_{\text{MD},1}$ attains its minimum $\frac{4-\sqrt{2}}{3}n^3 + O(n^2)$ ($\approx 0.8619n^2 + O(n^2)$) when C and its equivalent variants assume the following characteristics: (1) Within the four $(1 - \frac{1}{\sqrt{2}})n \times (1 - \frac{1}{\sqrt{2}})n$ corner-subgrids, the sequence of 1-normed distances between adjacent points in $[n]^2$, $d_1(C(i), C(i+1))$, incrementally increases and/or decreases in the range $[1, (1 - \frac{1}{\sqrt{2}})n]$ (while interleaving with segments of 1s), and (2) Within the central region interconnecting the corner-subgrids, the 1-normed distances are in $\{(1 - \frac{1}{\sqrt{2}})n, n\}$ (while interleaving with segments of 1s). As the z-order curve family shares some of these characteristics, the asymptotic ratios (constants greater than 1) obtained above are not surprising.

We have verified all the exact formulas (intermediate and final) involved in the derivations with computer programs over various grid-orders and 1-normed distances: ($m = 2$, $k \in \{1, 2, 3, 4, 5, 6, 7\}$, and $\delta \in \{1, 2^1, 2^2, 2^3, \ldots, 2^k\}$), and ($m = 3$, $k \in \{1, 2, 3, 4, 5, 6\}$, and $\delta = 1$).

3.4 Summary

Our analytical study of the locality properties of the Hilbert and z-order curve families, $\{H_k^m \mid k = 1, 2, \ldots\}$ and $\{Z_k^m \mid k = 1, 2, \ldots\}$, respectively, is based on the locality measure L_δ, which cumulates all index-differences between point-pairs at a common 1-normed distance δ. We have derived the exact formulas for $L_\delta(H_k^m)$ and $L_\delta(H_k^m)$ for $m = 2$ and arbitrary δ that is an integral power of 2, and $m = 3$ and $\delta = 1$. The results allow us to gauge the two curve families relative to the optimal curves with respect to L_δ, and show that the z-order curve family performs better than the Hilbert curve family over the considered ranges of dimension, grid-order, and 1-normed distance. We have verified all the exact formulas (intermediate and final) involved in the derivations with computer programs for $m = 2, 3$ and over various grid-orders and all possible 1-normed distances.

CHAPTER IV

LOCALITY MEASURES BASED ON p-NORM METRICS

To measure the proximity preservation of close-by points in the indexing space $[n^m]$, Gotsman and Lindenbaum [GL96] develop the following measures:

$$L_{\text{GL,max}}(C) = \max_{i,j \in [n^m] | i<j} \frac{d_2(C(i), C(j))^m}{|i-j|}, \text{ for } C \in \mathcal{C}.$$

Specifically, they apply the measure to the 2-dimensional Hilbert curve family, and obtain tight bounds:

$$6(1 - O(2^{-k})) \leq L_{\text{GL,max}}(H_k^2) \leq 6\frac{2}{3}.$$

Later, Alber and Niedermeier [Alb97, AN00] generalize $L_{\text{GL,max}}$ to $L_{\text{AN},p}$ by employing the p-normed metric d_p in place of the Euclidean distance d_2. They improve and extend the above tight bounds for the 2-dimensional Hilbert curve family to:

$$\begin{aligned} & L_{\text{AN},1}(H_k^2) \leq 9\frac{3}{5}, \\ 6(1 - O(2^{-k})) \leq \; & L_{\text{AN},2}(H_k^2) \leq 6\frac{1}{2}, \text{ and} \\ 6(1 - O(2^{-k})) \leq \; & L_{\text{AN},\infty}(H_k^2) \leq 6\frac{2}{5}. \end{aligned}$$

In this chapter, we close the gaps between the current best lower and upper bounds for $L_{\text{AN},p}(H_k^2)$ for $p = 1$ and all reals $p \geq 2$ by exact formulas.

4.1 Approach

For a space-filling curve C indexing an m-dimensional grid space with side length n, the notation "$v \in C$" refers to "grid point v indexed by C", and $C^{-1}(v)$ gives the index of v in the 1-dimensional index space. The locality measure $L_{\mathrm{AN},p}(C)$ from Alber and Niedermeier can be expressed in terms of grid points:

$$\begin{aligned} L_{\mathrm{AN},p}(C) &= \max_{i,j\in[n^m]|i<j} \frac{d_p(C(i),C(j))^m}{d_p(i,j)} (= \max_{i,j\in[n^m]|i<j} \frac{d_p(C(i),C(j))^m}{|i-j|}) \\ &= \max_{v,u\in C} \frac{d_p(v,u)^m}{|C^{-1}(v)-C^{-1}(u)|}. \end{aligned}$$

When $m = 2$, we write $\mathcal{L}_{C,p}(v,u) = \frac{d_p(v,u)^2}{\delta_C(v,u)}$, where $\delta_C(v,u)$ denotes the index-difference $|C^{-1}(v) - C^{-1}(u)|$. A pair of grid points v and u is representative for C with respect to $L_{\mathrm{AN},p}$ if $\mathcal{L}_{C,p}(v,u) = L_{\mathrm{AN},p}(C)$, and the pair (v,u) is called a representative pair for C with respect to $L_{\mathrm{AN},p}$.

To obtain the exact formula for $L_{\mathrm{AN},p}(H_k^2)$, we identify representative pairs (v,u) that yield $\mathcal{L}_{H_k^2,p}(v,u) = L_{\mathrm{AN},p}(H_k^2)$. Our approach that covers several cases for different values of p is stated as follows.

1. For $p = \{1,2\}$:

   (a) Follow the steps in the upper-bound argument in [GL96] by considering an arbitrary subcurve/subpath P of length l along H_k^2, where $(2^{r-1})^2 < l < (2^r)^2$ for some sufficiently large integer $r < k$. This subcurve P is contained in two adjacent quadrants Q' and Q'', each with size $(2^r)^2$ (grid points). Let D denote the diameter (with respect to the p-normed distance d_p) of the set of grid points in P. A case analysis of subpath containment (of P) in subquadrants of size $(2^{r-1})^2$ within $Q' \cup Q''$ results in six cases for $\frac{D^2}{l}$ (see [GL96] or Section 4.2.2).

   (b) In order to obtain the desired $L_{\mathrm{AN},2}$-bound, it suffices to refine the analysis of subpath containment in subquadrants of size $(2^{r-2})^2$. By comparing the values of $\frac{D^2}{l}$, the refined analysis obtains the subcurve (from Case 5 in [GL96]) in which a representative pair resides after ruling out other cases. This obtained subcurve is a structure of four linearly adjacent Hilbert subcurves (see Figure 4.1).

(c) Identify a representative pair that has the maximum of $\frac{D^2}{l}$ within the subcurve of four linearly adjacent Hilbert subcurves: across the first H_k^2-structured subcurve (denoted by ${}_1H_k^2$) and the fourth one (denoted by ${}_4H_k^2$), there exist pairs of grid points that have greater value of $\frac{D^2}{l}$ than any pairs of grid points across other pairs of subcurves. A more refined analysis yields a pair of grid points across $Q_3({}_1H_k^2)$ and $Q_2({}_4H_k^2)$ as the representative pair.

2. For arbitrary real number $p > 2$, we identify a representative pair same as the one for $p = 2$ by the observation that the value $\frac{D^2}{l}$ of this representative pair remains unchanged when p increases while the values of others decrease.

Prior to discussing details, first extend notations to identify all C_l^m-structured subcurves of a structured C_k^m for all $l \in [k]$ inductively on the order. Let $Q_\alpha(C_k^m)$ denote the α-th C_{k-1}^m-structured subcurve (along the amalgamating C_1^m-curve) for all $\alpha \in [2^m]$. Then for the α-th C_{l-1}^m-structured subcurve, $Q_\alpha(C_l^m)$, of C_l^m, where $2 < l \leq k$ and $\alpha \in [2^m]$, let $Q_\beta(Q_\alpha(C_l^m))$ denote the β-th C_{l-2}^m-structured subcurve of $Q_\alpha(C_l^m)$ for all $\beta \in [2^m]$. We write $Q_\alpha^{q+1}(C_l^m)$ for $Q_\alpha(Q_\alpha^q(C_l^m))$ for all $l \in [k]$ and all positive integers $q < l$. For the two extreme cases: $Q_\alpha^0(C_l^m)$ denotes C_l^m (when $q = 0$), and $Q_\alpha^l(C_l^m)$ (when $q = l$) identifies the α-th grid point in the C_1^m-structured subcurve $Q_\alpha^{l-1}(C_l^m)$. We write $Q_\alpha^{q+1}(C_l^m)$ for an iteration $Q_\alpha(Q_\alpha^q(C_l^m))$ for all $l \in [k]$ and all positive integers $q < l$.

For an H_l^m-structured subcurve C of a 2-dimensional Hilbert curve H_k^m in Cartesian x-y coordinates, where $l \in [k]$, notice that $\partial_1(C)$ and $\partial_2(C)$ differ exactly in one coordinate, say $z \in \{x, y\}$. It is said that the subcurve C is z^+-oriented (respectively, z^--oriented) if the z-coordinate of $\partial_1(C)$ is less than (respectively, greater than) that of $\partial_2(C)$. Note that for a 2-dimensional Hilbert curve H_k^m, its two subcurves $Q_2(H_k^m)$ and $Q_3(H_k^m)$ inherit the orientation from their supercurve H_k^2.

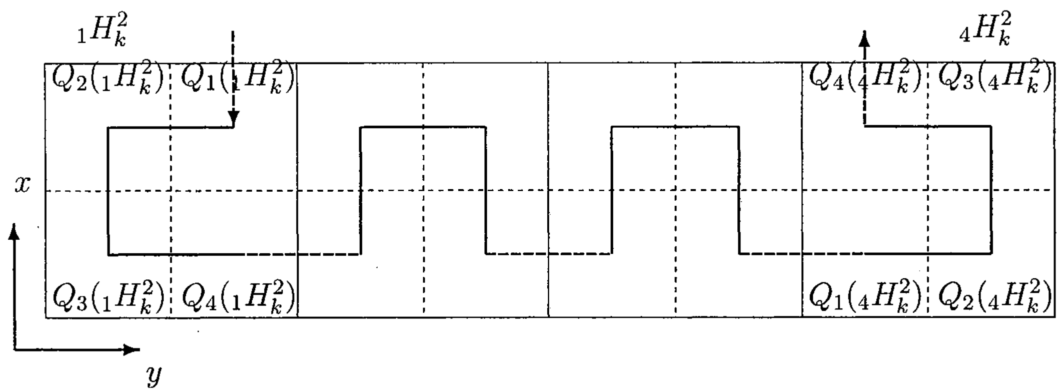


Figure 4.1: Four linearly adjacent H_k^2-structured subcurves.

4.2 Exact Formula for $L_{\mathrm{AN},p}(H_k^2)$ with $p = 2$

According to locality measure in [Alb97, AN00], we consider the case of $p = 2$ first by following the argument in [GL96] with a refined analysis and then extend p to be arbitrary real $p > 2$. As mentioned in Section 4.1, we seek the representative pair in the curve that is composed of four linearly adjacent H_k^2-structured subcurve before others and after that, we attempt to refine the case analysis of subpath containment in [GL96] by narrowing down the possible subcurve that contains representative pair. As a matter of fact, this subcurve is precisely the subcurve composed of four linearly adjacent H_k^2-structured subcurve that has been sought previously, so the exact formula for $L_{\mathrm{AN},p}(H_k^2)$ is corollarily derived.

4.2.1 Locality of Four Linearly Adjacent Hilbert Subcurves

For a 2-dimensional Hilbert curve H_l^2 with $l \geq 3$, there exists a subcurve C that is composed of four linearly adjacent H_k^2-structured subcurves with $k \leq l-3$. Figure 4.1 depicts the arrangement in Cartesian coordinates. Denote the leftmost and rightmost (first and fourth in the traversal order) H_k^2-structured subcurves by $_1H_k^2$ (x^--oriented) and $_4H_k^2$ (x^+-oriented), respectively.

For a grid point v, denotes by $X(v)$ and $Y(v)$ the x- and y-coordinate of v, respectively, and denotes by $(X(v), Y(v))$ the grid point v in the coordinate system. In this subsection, we assume that the lower-left corner grid point of $_1H_k^2$ is the origin (1, 1) of the coordinate system. In the following analysis, we identify a pair of grid points $v' \in {}_1H_k^2$ and $u' \in {}_4H_k^2$ such that $\mathcal{L}_{C,2}(v',u') = \max\{\mathcal{L}_{C,2}(v,u) \mid v \in {}_1H_k^2 \text{ and } u \in {}_4H_k^2\}$. Later we see that (v',u') serves as the representative pair for C with respect to $L_{\mathrm{AN},2}$.

To locate a potential representative pair $v \in {}_1H_k^2$ and $u \in {}_4H_k^2$, the following three lemmas show that the possibility "$v \in Q_3({}_1H_k^2)$ and $u \in Q_3({}_4H_k^2)$" is reduced to seeking v in successive Q_3-subcurves of ${}_1H_k^2$.

Lemma 4.1 *For all $v \in Q_3({}_1H_k^2) - Q_3(Q_3({}_1H_k^2))$ and all $u \in Q_3({}_4H_k^2)$, there exists $v' \in Q_3(Q_3({}_1H_k^2))$ such that $\mathcal{L}_{C,2}(v',u) \geq \mathcal{L}_{C,2}(v,u)$.*

Proof. Since $Q_3({}_1H_k^2) - Q_3(Q_3({}_1H_k^2)) = Q_1(Q_3({}_1H_k^2)) \cup Q_2(Q_3({}_1H_k^2)) \cup Q_4(Q_3({}_1H_k^2))$, we consider the following three cases.

Case 1: $v \in Q_2(Q_3({}_1H_k^2))$. Consider $v' \in Q_3(Q_3({}_1H_k^2))$ with $Y(v') = Y(v)$, then $d_2(v',u)^2 > d_2(v,u)^2$ and $\delta_C(v',u) < \delta_C(v,u)$, and we have $\mathcal{L}_{C,2}(v',u) > \mathcal{L}_{C,2}(v,u)$.

Case 2: $v \in Q_1(Q_3({}_1H_k^2))$. Consider $v'' \in Q_2(Q_3({}_1H_k^2))$ with $X(v'') = X(v)$, then $\mathcal{L}_{C,2}(v'',u) \geq \mathcal{L}_{C,2}(v,u)$. From Case 1, there exists $v' \in Q_3(Q_3({}_1H_k^2))$ such that $\mathcal{L}_{C,2}(v',u) > \mathcal{L}_{C,2}(v'',u) > \mathcal{L}_2(v,u)$.

Case 3: $v \in Q_4(Q_3({}_1H_k^2))$. Consider $v' \in Q_3(Q_3({}_1H_k^2))$ with $Y(v') = 1$ and $X(v') = X(v)$, we have:

$$
\begin{aligned}
& d_2(v',u)^2 \cdot \delta_C(v,u) - d_2(v,u)^2 \cdot \delta_C(v',u) \\
= \; & ((Y(u) - Y(v'))^2 + (X(u) - X(v'))^2) \cdot \\
& \quad (\delta_C(v, \partial_2({}_1H_k^2)) + 2 \cdot 2^{2k} + \delta_C(u, \partial_1({}_4H_k^2)) + 1) \\
& -((Y(u) - Y(v))^2 + (X(u) - X(v))^2) \cdot \\
& \quad (\delta_C(v', \partial_2({}_1H_k^2)) + 2 \cdot 2^{2k} + \delta_C(u, \partial_1({}_4H_k^2)) + 1) \\
= \; & ((Y(u) - 1)^2)(\delta_C(v, \partial_2({}_1H_k^2)) + 2 \cdot 2^{2k} + \delta_C(u, \partial_1({}_4H_k^2)) + 1) \\
& +(X(u) - X(v))^2(\delta_C(v, \partial_2({}_1H_k^2)) + 2 \cdot 2^{2k} + \delta_C(u, \partial_1({}_4H_k^2)) + 1) \\
& (\text{because } Y(v') = 1, X(v') = X(v)) \\
& -((Y(u) - Y(v))^2)(\delta_C(v', \partial_2({}_1H_k^2)) + 2 \cdot 2^{2k} + \delta_C(u, \partial_1({}_4H_k^2)) + 1)
\end{aligned}
$$

$$
\begin{aligned}
&-(X(u)-X(v))^2(\delta_C(v',\partial_2({}_1H_k^2))+2\cdot 2^{2k}+\delta_C(u,\partial_1({}_4H_k^2))+1)\\
= \;& Y(u)^2(\delta_C(v,\partial_2({}_1H_k^2))-\delta_C(v',\partial_2({}_1H_k^2)))\\
&+(-2Y(u)+1+2Y(u)\cdot Y(v)-Y(v)^2)(2\cdot 2^{2k}+1)\\
&+(-2Y(u)+1+2Y(u)\cdot Y(v)-Y(v)^2)(\delta_C(v',\partial_2({}_1H_k^2))+\delta_C(u,\partial_1({}_4H_k^2)))\\
&+(X(u)-X(v))^2\cdot\delta_C(v,\partial_2({}_1H_k^2))-(X(u)-X(v))^2\cdot\delta_C(v',\partial_2({}_1H_k^2))\\
= \;& Y(u)^2(\delta_C(v,\partial_2({}_1H_k^2))-\delta_C(v',\partial_2({}_1H_k^2)))\\
&+(2Y(u)-1)(Y(v)-1)(2\cdot 2^{2k}+1)\\
&+(2Y(u)-1)(Y(v)-1)(\delta_C(v',\partial_2({}_1H_k^2))+\delta_C(u,\partial_1({}_4H_k^2)))\\
&+(X(u)-X(v))^2(\delta_C(v,\partial_2({}_1H_k^2))-\delta_C(v',\partial_2({}_1H_k^2))). \qquad (4.1)
\end{aligned}
$$

The ranges for the values related to u, v are:

$$
\begin{aligned}
4\cdot 2^k \geq \;& Y(u) \geq \frac{7}{2}\cdot 2^k+1,\\
2^k \geq \;& X(u) \geq \frac{1}{2}\cdot 2^k+1,\\
\frac{1}{2}\cdot 2^k \geq \;& Y(v) \geq \frac{1}{4}\cdot 2^k+1,\\
\frac{1}{4}\cdot 2^k \geq \;& X(v) \geq 1,
\end{aligned}
$$

$$
\begin{aligned}
\frac{5}{16}\cdot 2^{2k} > \;& \delta_C(v,\partial_2({}_1H_k^2)) \geq \frac{1}{4}\cdot 2^{2k},\\
\frac{6}{16}\cdot 2^{2k} > \;& \delta_C(v',\partial_2({}_1H_k^2)) \geq \frac{5}{16}\cdot 2^{2k}.
\end{aligned}
$$

Thus, the ranges of the four terms in the last statement in Equation 4.1 are

$$
\begin{aligned}
& Y(u)^2(\delta_C(v,\partial_2({}_1H_k^2))-\delta_C(v',\partial_2({}_1H_k^2)))\\
&\quad \geq -2\cdot 2^{4k},\\
&(2Y(u)-1)(Y(v)-1)(2\cdot 2^{2k}+1)\\
&\quad \geq (7\cdot 2^k+1)(\frac{1}{4}\cdot 2^k)(2\cdot 2^{2k}+1) > \frac{7}{2}\cdot 2^{4k},\\
&(2Y(u)-1)(Y(v)-1)(\delta_C(v',\partial_2({}_1H_k^2))+\delta_C(u,\partial_1({}_4H_k^2)))\\
&\quad \geq (7\cdot 2^k+1)(\frac{1}{4}\cdot 2^k)(\frac{9}{16}\cdot 2^{2k}) > 0,
\end{aligned}
$$

$$(X(u) - X(v))^2(\delta_C(v, \partial_2({}_1H_k^2)) - \delta_C(v', \partial_2({}_1H_k^2)))$$
$$\geq (2^k - 1)^2(-\frac{2}{16} \cdot 2^{2k}) > -\frac{1}{8} \cdot 2^{4k}.$$

Combining these four terms together, Equation 4.1 is greater than 0. Therefore, $d_2(v', u)^2 \cdot \delta_C(v, u) - d_2(v, u)^2 \cdot \delta_C(v', u) > 0$.

This gives that $\mathcal{L}_{C,2}(v', u) > \mathcal{L}_{C,2}(v, u)$.

Combining the three cases, the lemma is proved. ∎

Lemma 4.2 *For all integers h with $1 \leq h < k$, and all $v \in Q_3^h({}_1H_k^2) - Q_3^{h+1}({}_1H_k^2)$ and all $u \in Q_3({}_4H_k^2)$, there exists $v' \in Q_3^{h+1}({}_1H_k^2)$ such that $\mathcal{L}_{C,2}(v', u) > \mathcal{L}_{C,2}(v, u)$.*

Proof. Similar to the proof of the previous lemma. ∎

Lemma 4.3 *For all integers h with $1 \leq h < k$, and all $v \in Q_3^h({}_1H_k^2) - Q_3^k({}_1H_k^2)$ and all $u \in Q_3({}_4H_k^2)$, there exists $v' \in Q_3^k({}_1H_k^2)$ such that $\mathcal{L}_{C,2}(v', u) > \mathcal{L}_{C,2}(v, u)$.*

Proof. By induction on $k - h$. For the basis of the induction ($k - h = 1$), apply Lemma 4.2 with $h = k - 1$.

For the induction step, suppose that the statement in the lemma is true for all integers h with $1 \leq k-h < n$, where $n > 1$. Consider the case when $k-h = n$. Let $v \in Q_3^h({}_1H_k^2) - Q_3^k({}_1H_k^2)$ and $u \in Q_3({}_4H_k^2)$ be arbitrary. Since $Q_3^h({}_1H_k^2) = Q_3(Q_3^h({}_1H_k^2)) \cup (Q_1(Q_3^h({}_1H_k^2)) \cup Q_2(Q_3^h({}_1H_k^2)) \cup Q_4(Q_3^h({}_1H_k^2))) = Q_3^{h+1}({}_1H_k^2) \cup (Q_3^h({}_1H_k^2) - Q_3^{h+1}({}_1H_k^2))$, we consider the following two cases.

Case 1: $v \in Q_3^{h+1}({}_1H_k^2)$. Notice that $k - (h+1) < n$. Apply the induction hypothesis for the case of $k - (h+1)$, we obtain a desired v'.

Case 2: $v \in Q_3^h({}_1H_k^2) - Q_3^{h+1}({}_1H_k^2)$. By Lemma 4.2, there exists $v' \in Q_3^{h+1}({}_1H_k^2))$ such that $\mathcal{L}_{C,2}(v', u) > \mathcal{L}_{C,2}(v, u)$. If $v' \in Q_3^k({}_1H_k^2)$, then v' is a desired grid point. Otherwise ($v \in Q_3^{h+1}({}_1H_k^2) - Q_3^k({}_1H_k^2)$), this is reduced to Case 1.

This completes the induction step, and the lemma is proved. ∎

Lemma 4.3 says that the lower-left corner grid point v' with coordinates $(1, 1)$

is unique in $Q_3({}_1H_k^2)$ such that $\mathcal{L}_{C,2}(v',u) = \max\{\mathcal{L}_{C,2}(v,u) \mid v \in Q_3({}_1H_k^2)\}$ for arbitrary $u \in Q_3({}_4H_k^2)$.

The search for a potential representative pair can be reduced to a case analysis for all possible pair-combinations $(Q_i({}_1H_k^2), Q_j({}_4H_k^2))$ for all $i, j \in \{1,2,3,4\}$. After eliminating symmetrical cases and grouping, it suffices to consider the analysis for five major cases: $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$, $(Q_3({}_1H_k^2), Q_3({}_4H_k^2))$, $(Q_3({}_1H_k^2), Q_4({}_4H_k^2))$, $(Q_4({}_1H_k^2), {}_4H_k^2)$, and $(Q_1({}_1H_k^2) \cup Q_2({}_1H_k^2), Q_3({}_4H_k^2) \cup Q_4({}_4H_k^2))$. We show that the analysis for each pair is reduced to that for the pair $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$ in the following lemmas (Lemmas 4.4, 4.5, 4.7, and 4.8).

Lemma 4.4 *For all $v \in Q_3({}_1H_k^2)$ and all $u \in Q_3({}_4H_k^2)$, there exist $v' \in Q_3({}_1H_k^2)$ and $u' \in Q_2({}_4H_k^2)$ such that $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v,u)$.*

Proof. Consider $v' \in Q_3^k({}_1H_k^2)$ $(= (1,1))$ and $u' \in Q_2({}_4H_k^2)$ with $Y(u') = Y(u)$ and $X(u') = 1$. A case analysis for $u \in Q_i(Q_3({}_4H_k^2))$ with $i \in \{1,2,3,4\}$ can show that $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v',u)$. By Lemma 4.3, $\mathcal{L}_{C,2}(v',u) \geq \mathcal{L}_{C,2}(v,u)$; therefore $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v,u)$. ∎

Lemma 4.5 *For all $v \in Q_3({}_1H_k^2)$ and all $u \in Q_4({}_4H_k^2)$, there exist $v' \in Q_3({}_1H_k^2)$ and $u' \in Q_2({}_4H_k^2)$, such that $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v,u)$.*

Proof. Consider $u'' \in Q_3({}_4H_k^2)$ with $X(u'') = X(u)$. Notice that $d_2(v,u'') > d_2(v,u)$ and $\delta_C(v,u'') < \delta_C(v,u)$, we have $\mathcal{L}_{C,2}(v,u'') > \mathcal{L}_{C,2}(v,u)$. By Lemma 4.4, there exist $v' \in Q_3({}_1H_k^2)$ and $u' \in Q_2({}_4H_k^2)$ such that $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v,u'') > \mathcal{L}_{C,2}(v,u)$. ∎

Lemma 4.6 *For all $v \in Q_4({}_1H_k^2)$ and all $u \in {}_4H_k^2$ $(= Q_1({}_4H_k^2) \cup Q_2({}_4H_k^2) \cup Q_3({}_4H_k^2) \cup Q_4({}_4H_k^2))$, there exists $v' \in Q_3({}_1H_k^2)$ such that $\mathcal{L}_{C,2}(v',u) > \mathcal{L}_{C,2}(v,u)$.*

Proof. Consider $v' \in Q_3({}_1H_k^2)$ with $X(v') = X(v)$ and $Y(v') = 1$. A case analysis for $u \in Q_i({}_4H_k^2)$ with $i \in \{1,2,3,4\}$ can show that $\mathcal{L}_{C,2}(v',u) > \mathcal{L}_{C,2}(v,u)$. ∎

Lemma 4.7 *For all $v \in Q_4({}_1H_k^2)$ and all $u \in {}_4H_k^2$, there exists $v' \in Q_3({}_1H_k^2)$ and*

$u' \in Q_2({}_4H_k^2)$ such that $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v,u)$.

Proof. Lemma 4.6 says that there exists $v' \in Q_3({}_1H_k^2)$ such that $\mathcal{L}_{C,2}(v',u) > \mathcal{L}_{C,2}(v,u)$. Since $u \in {}_4H_k^2 = Q_1({}_4H_k^2) \cup Q_2({}_4H_k^2) \cup Q_3({}_4H_k^2) \cup Q_4({}_4H_k^2)$, consider four pair-combinations for (v',u): $(Q_3({}_1H_k^2), Q_i({}_4H_k^2))$ with $i \in \{1,2,3,4\}$. The analysis for the pair $(Q_3({}_1H_k^2), Q_1({}_4H_k^2))$ is equivalent to that for $(Q_4({}_1H_k^2), Q_2({}_4H_k^2))$, which is reduced to $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$ by applying Lemma 4.6. The pair $(Q_3({}_1H_k^2), Q_3({}_4H_k^2))$ is reduced to $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$ by Lemma 4.4, and the pair $(Q_3({}_1H_k^2), Q_4({}_4H_k^2))$ is reduced to $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$ by Lemma 4.5. ∎

Lemma 4.8 *For all $v \in Q_1({}_1H_k^2) \cup Q_2({}_1H_k^2)$ and all $u \in Q_3({}_4H_k^2) \cup Q_4({}_4H_k^2)$, there exist $v' \in Q_3({}_1H_k^2)$ and $u' \in Q_2({}_4H_k^2)$ such that $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v,u)$.*

Proof. Consider $v'' \in Q_3({}_1H_k^2) \cup Q_4({}_1H_k^2)$ with $Y(v'') = Y(v)$ and $X(v'') = X(v) - 2^{k-1}$, and $u'' \in Q_1({}_4H_k^2) \cup Q_2({}_4H_k^2)$ with $Y(u'') = Y(u)$ and $X(u'') = X(u) - 2^{k-1}$. Since $d_2(v'',u'') = d_2(v,u)$ and $\delta_C(v'',u'') < \delta_C(v,u)$, we have $\mathcal{L}_{C,2}(v'',u'') > \mathcal{L}_{C,2}(v,u)$. It suffices to consider two pair-combinations for (v'',u''): $(Q_3({}_1H_k^2), Q_1({}_4H_k^2))$ and $(Q_4({}_1H_k^2), Q_1({}_4H_k^2) \cup Q_2({}_4H_k^2))$. The analysis for the pair $(Q_3({}_1H_k^2), Q_1({}_4H_k^2))$ is equivalent to that for $(Q_4({}_1H_k^2), Q_2({}_4H_k^2))$, which is reduced to $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$ by Lemma 4.6. The pair $(Q_4({}_1H_k^2), Q_1({}_4H_k^2) \cup Q_2({}_4H_k^2))$ is a subcase of Lemma 4.7. As a consequence, for these two pair-combinations for (v'',u''), there exists $v' \in Q_3({}_1H_k^2)$ and $u' \in Q_2({}_4H_k^2)$ such that $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v'',u'')$, then we can reach $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v,u)$. ∎

An immediate consequence of Lemmas 4.4, 4.5, 4.7 and 4.8 is summarized below — a representative pair must reside in $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$.

Lemma 4.9 *For all $v \in {}_1H_k^2 - Q_3({}_1H_k^2)$ and all $u \in {}_4H_k^2 - Q_2({}_4H_k^2)$, there exist $v' \in Q_3({}_1H_k^2)$ and $u' \in Q_2({}_4H_k^2)$ such that $\mathcal{L}_{C,2}(v',u') > \mathcal{L}_{C,2}(v,u)$.*

The following lemmas complement Lemmas 4.2 and 4.3, respectively, with similar proofs. Having reached the pair $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$ for seeking a potential representative pair (v', u'), they guide the search into successive Q_3-subcurves of ${}_1H_k^2$ for v'. The symmetry in the pair $(Q_3({}_1H_k^2), Q_2({}_4H_k^2))$ leads the search into successive Q_2-subcurves of ${}_4H_k^2$ for u'.

Lemma 4.10 *For all integers h with $1 \leq h < k$, and all $v \in Q_3^h({}_1H_k^2) - Q_3^{h+1}({}_1H_k^2)$ and all $u \in Q_2({}_4H_k^2)$, there exists $v' \in Q_3^{h+1}({}_1H_k^2)$ such that $\mathcal{L}_{C,2}(v', u) > \mathcal{L}_{C,2}(v, u)$.*

Lemma 4.11 *For all integers h with $1 \leq h < k$, and all $v \in Q_3^h({}_1H_k^2) - Q_3^k({}_1H_k^2)$ and all $u \in Q_2({}_4H_k^2)$, there exists $v' \in Q_3^k({}_1H_k^2)$ such that $\mathcal{L}_{C,2}(v', u) > \mathcal{L}_{C,2}(v, u)$.*

The following theorem summarizes our analysis above, and asserts that the unique representative pair reside at the lower-left and lower-right corners of C.

Theorem 4.1 *For all $v \in {}_1H_k^2 - Q_3^k({}_1H_k^2)$ and all $u \in {}_4H_k^2 - Q_2^k({}_4H_k^2)$, there exist $v' \in Q_3^k({}_1H_k^2)$ and $u' \in Q_2^k({}_4H_k^2)$ such that $\mathcal{L}_{C,2}(v', u') > \mathcal{L}_{C,2}(v, u)$ and $\mathcal{L}_{C,2}(v', u') = 6 \cdot \frac{2^{2k+3}-2^{k+2}+2^{-1}}{2^{2k+3}+1}$.*

Proof. By Lemmas 4.9 and 4.11 (and its symmetry), we have $v' \in Q_3^k({}_1H_k^2)$ with coordinates $(1, 1)$ and $u' \in Q_2^k(H_k^2)$ with coordinates $(1, 2^{k+2})$, which maximizes the $\mathcal{L}_{C,2}$-value.

Notice that $\delta_C(v', u') = 2(\sum_{i=0}^{k-1} 2^{2i} + 1 + 2 \cdot 2^{2k}) - 1$. This give that $\mathcal{L}_{C,2}(v', u') = \frac{d_2(v',u')^2}{\delta_C(v',u,)} = 6 \cdot \frac{2^{2k+3}-2^{k+2}+2^{-1}}{2^{2k+3}+1}$. ∎

4.2.2 Exact Formula for $L_{\mathrm{AN},2}(H_k^2)$

The current best bounds for the 2-dimensional Hilbert curve family with respect to $L_{\mathrm{AN},2}(H_k^2)$ [AN00] is:

$$6(1 - O(2^{-k})) \leq L_{\mathrm{AN},2}(H_k^2) \leq 6\frac{1}{2}.$$

Following the argument in [GL96] with a refined analysis, together with the exact formula for $\mathcal{L}_{C,2}(v', u')$ in Section 4.2.1, we merge the two bounds to an exact formula for $L_{\mathrm{AN},2}(H_k^2)$.

Theorem 4.2 *There exists a positive integer k_0 such that, for all $k \geq k_0$,*

$$L_{AN,2}(H_k^2) = 6 \cdot \frac{2^{2k-3} - 2^{k-1} + 2^{-1}}{2^{2k-3} + 1}.$$

Proof. In the upper-bound argument in [GL96], an arbitrary subcurve/subpath P of length l along H_k^2 is considered. Note that for arbitrary l, there exists a sufficiently large positive integer r such that $(2^{r-1})^2 < l \leq (2^r)^2$. This gives that P is contained in two adjacent quadrants Q' and Q'', each with size $(2^{r-1})^2$ (grid points). Let D denote the diameter (Euclidean) of the set of grid points in P. A case analysis of subpath containment (of P) in subquadrants of size $(2^{r-1})^2$ within $Q' \cup Q''$ results in the following six cases.

1. $\frac{4}{16}4^r < l \leq \frac{5}{16}4^r$: $D^2 < \frac{5}{4} \cdot 4^r$, hence $\frac{D^2}{l} \leq 5$.

2. $\frac{5}{16}4^r < l \leq \frac{6}{16}4^r$: $D^2 < \frac{29}{16} \cdot 4^r$, hence $\frac{D^2}{l} \leq 5\frac{4}{5}$.

3. $\frac{6}{16}4^r < l \leq \frac{7}{16}4^r$: $D^2 < \frac{10}{4} \cdot 4^r$, hence $\frac{D^2}{l} \leq 6\frac{2}{3}$.

4. $\frac{7}{16}4^r < l \leq \frac{8}{16}4^r$: $D^2 < \frac{10}{4} \cdot 4^r$, hence $\frac{D^2}{l} \leq 5\frac{5}{7}$.

5. $\frac{8}{16}4^r < l \leq \frac{12}{16}4^r$: $D^2 < \frac{13}{4} \cdot 4^r$, hence $\frac{D^2}{l} \leq 6\frac{1}{2}$.

6. $\frac{12}{16}4^r < l \leq 4^r$: $D^2 < 5 \cdot 4^r$, hence $\frac{D^2}{l} \leq 6\frac{2}{3}$.

In order to obtain the desired $L_{\mathrm{AN},2}$-bound, it suffices to refine the analysis of subpath containment in cases 3, 5, and 6 in subquadrants of size $(2^{r-2})^2$.

The refined analysis for case 3 yields the upper bounds on $\frac{D^2}{l}$: $\frac{29}{6}$, $\frac{137}{25}$, $\frac{141}{26}$, and $\frac{160}{27}$ (maximum is $\frac{160}{27} < 5.93$). For case 6, the upper bounds on $\frac{D^2}{l}$ are: $\frac{68}{12}$, $\frac{73}{13}$, $\frac{80}{14}$, and $\frac{80}{15}$ (maximum is $\frac{80}{14} < 5.72$).

The analysis for case 5 reveals that all but one arrangement (of subquadrants of size $(2^{r-2})^2$) yield upper bounds that are bounded above and away from 6. The exception structure is given by the subcurve C (described in Section 4.2.1) of four linearly adjacent Hilbert subcurves. By Theorem 4.1, the maximum $\frac{D^2}{l}$-value for this case is $6 \cdot \frac{2^{2k+3}-2^{k+2}+2^{-1}}{2^{2k+3}+1}$. Observe that $\frac{2^{2k+3}-2^{k+2}+2^{-1}}{2^{2k+3}+1}$ is strictly increasing (in k)

and approaching 1 (as $k \to \infty$). This establishes the theorem. ∎

For an y^+-oriented Hilbert curve H_k^2 with $\partial_1(H_k^2) = (1,1)$, where $k \geq k_0$, the representative pair for H_k^2 with respect to $L_{\mathrm{AN},2}$ reside at the lower-left corner (with coordinates $(2^{k-1}+1, 2^{k-2}+1)$) and the lower-right corner (with coordinates $(2^{k-1}+1, 2^k - 2^{k-2})$) of four linearly adjacent largest subquadrants (H_{k-3}^2-structured subcurves).

4.3 Exact Formula for $L_{\mathrm{AN},p}(H_k^2)$ with $p > 2$

In order to study $L_{\mathrm{AN},p}$ for arbitrary real p, we first investigate the monotonicity of the underlying p-normed metric.

Lemma 4.12 *The function $f : (0, +\infty) \to (1, +\infty)$ defined by $f(p) = (1 + \alpha^p)^{\frac{1}{p}}$, where α is a positive real constant, is strictly decreasing over its domain.*

Proof. It is equivalent to show that the function $g : (0, +\infty) \to (0, +\infty)$ defined by $g(p) = \log f(p)$ ("log" denotes the natural logarithm) is strictly decreasing over its domain. We consider the first derivative of g, which is defined on $(0, +\infty)$:

$$g'(p) = \frac{\frac{\alpha^p}{1+\alpha^p}\log\alpha^p - \log(1+\alpha^p)}{p^2} = \frac{\log\alpha^p - \log(1+\alpha^p) - \frac{\log\alpha^p}{1+\alpha^p}}{p^2}.$$

Clearly, $g'(p) = \frac{\frac{\alpha^p}{1+\alpha^p}\log\alpha^p - \log(1+\alpha^p)}{p^2} < 0$ for $0 < \alpha < 1$, and $g'(p) = \frac{\log\alpha^p - \log(1+\alpha^p) - \frac{\log\alpha^p}{1+\alpha^p}}{p^2}$ < 0 for $1 \leq \alpha$. This proves the strictly decreasing property of f over its domain. ∎

An immediate corollary of the previous lemma is that for all grid points v and u, the p-normed metric $d_p(v,u)$ is decreasing in $p \in (0, +\infty)$. Hence for a space-filling curve C, $\mathcal{L}_{C,p}(v,u) = \frac{d_p(v,u)^2}{\delta_C(v,u)}$ is decreasing in $p \in (0, +\infty)$, as $\delta_C(v,u)$ is independent of p.

Theorem 4.3 *There exists a positive integer k_0 such that, for all $k \geq k_0$,*

$$L_{AN,p}(H_k^2) = 6 \cdot \frac{2^{2k-3} - 2^{k-1} + 2^{-1}}{2^{2k-3} + 1}, \text{ for all real } p \geq 2.$$

Proof. Let (v', u') be the representative pair for H_k^2 with respect to $L_{\mathrm{AN},2}$, with their coordinates $v' = (2^{k-1}+1, 2^{k-2}+1)$ and $u' = (2^{k-1}+1, 2^k - 2^{k-2})$. Consider an arbitrary real $p \geq 2$. We show that (v', u') also serves as the unique representative pair for H_k^2 with respect to $L_{\mathrm{AN},p}$, that is, $\mathcal{L}_{H_k^2,p}(v', u') > \mathcal{L}_{H_k^2,p}(v, u)$ with $(v, u) \neq (v', u')$.

Observe that $X(v') = X(u')$, which implies that $d_p(v', u') = d_2(v', u')$. Then for arbitrary grid points $v, u \in H_k^2$ with $(v', u') \neq (v, u)$, we have:

$$\begin{aligned}
\mathcal{L}_{H_k^2,p}(v', u') &= \frac{d_p(v', u')^2}{\delta_{H_k^2}(v', u')} = \frac{d_2(v', u')^2}{\delta_{H_k^2}(v', u')} = \mathcal{L}_{H_k^2,2}(v', u') \\
&> \mathcal{L}_{H_k^2,2}(v, u) \quad \text{(as } (v', u') \text{ is a representative pair with respect to } \mathcal{L}_{H_k^2,2}) \\
&\geq \mathcal{L}_{H_k^2,p}(v, u) \quad \text{(by the monotonicity of } \mathcal{L}_{H_k^2,p}).
\end{aligned}$$

∎

4.4 Exact Formula for $L_{\mathrm{AN},p}(H_k^2)$ with $p = 1$

Following an argument similar to the one in Sections 4.2.1 and 4.2.2 to establish $L_{\mathrm{AN},2}(H_k^2)$, we obtain the exact formula for $L_{\mathrm{AN},1}(H_k^2)$.

Theorem 4.4 *There exists a positive integer k_0 such that, for all $k \geq k_0$,*

$$L_{AN,1}(H_k^2) = 9 - 3 \cdot 2^{-k+3} + 2^{-2k+4}.$$

There are two (symmetrical) representative pairs for H_k^2 with respect to $L_{\mathrm{AN},1}$; namely (v', u') and (v'', u''). For an y^+-oriented Hilbert curve H_k^2 with $\partial_1(H_k^2) = (1, 1)$, where $k \geq k_0$, the coordinates of (v', u') and (v'', u'') are $((1, 2^{k-1}), (2^k, 1))$ and $((1, 2^{k-1}+1), (2^k, 2^k))$, respectively. Thus $d_1(v', u') = 2^k + 2^{k-1} - 2$ and $\delta_{H_k^2}(v', u') = 2^{2k-2}$, and $L_{\mathrm{AN},1}(H_k^2) = \mathcal{L}_{H_k^2,1}(v', u') = 9 - 3 \cdot 2^{-k+3} + 2^{-2k+4}$.

4.5 Summary

The analytical study of the locality properties in this chapter is based on the locality measure $L_{\mathrm{AN},p}$, which is the maximum ratio of $d_p(v, u)^m$ to $d_p(\tilde{v}, \tilde{u})$ over all corresponding point-pairs (v, u) and $(\tilde{v}, \tilde{u})$ in the m-dimensional grid space and index space, respectively. For this locality measure of the Hilbert curve family $\{H_k^2 \mid k = 1, 2, \ldots\}$, our work merges the current best lower and upper bounds to exact formulas for $p \in \{1, 2\}$, and also extend our work to all reals $p \geq 2$. In addition, we identify all the representative pairs (which realize $L_{\text{AN},p}(H_k^2)$) for $p = 1$ and all reals $p \geq 2$. We also validate the results with computer programs over various p-values ($p \in \{1, 2, 3\}$) and grid-orders ($k \in \{4, 5, \ldots, 10\}$).

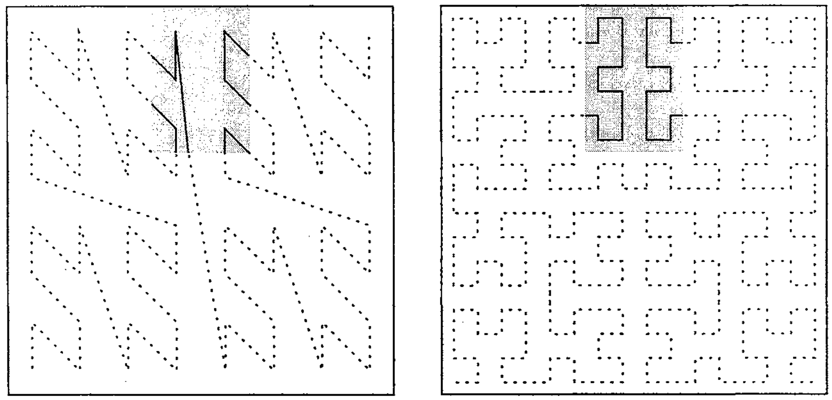
CHAPTER V

MEASURE BY MEAN NUMBER OF CLUSTERS

Evaluating clustering performance of space-filling curves is primarily to measure the distribution of continuous runs of grid points (clusters) over all identically shaped subspaces of $[n^m]$. Moreover, it can be characterized into two different measures: the mean number of clusters and the mean inter-cluster distance within a subspace. This and next chapters cover the discussions of clustering and inter-clustering, respectively.

Using the mean number of clusters within a subspace as a measure for a space-filling curve is initiated from space-filling indexing applications for multi-dimensional databases, in which a range query is mapped to a subspace (see Figure 5.1). The number of clusters within a subspace is related to the number of disk access. If a grid point corresponds to a disk page, the number of clusters within a subspace is related to the number of non-consecutive disk accesses, hence a typical query requires additional disk-seek operations. On the other hand, if many grid points are mapped to a disk page, this measure is still highly correlated to the number of disk access, since consecutive grid points are likely located in the same disk page (or neighboring disk pages) while non-consecutive grid points are likely located in different disk pages (and/or non-neighboring disk pages) — which results in more disk accesses. In other words, the less disk accesses are resulted from operations, the better performances of space-filling curves are. Based on this criterion, the statistics of mean number of clusters is used to gauge the performances of different space-filling curves.

Moon, Jagadish, Faloutsos, and Saltz [MJFS01] extend the work in [Jag97] to obtain the exact formula for the mean number of clusters over all rectangular $2^q \times 2^q$ subspaces of an H_k^2-structural grid space. Alternatively, we follow a recursive



(a) (b)

Figure 5.1: Clusters within a subspace for (a) z-order curve and (b) Hilbert curve.

approach to derive the exact formulas for the Hilbert and z-order curve families.

5.1 Approach

For an H_k^2- and Z_k^2-structural grid spaces, we obtain the exact formulas for the mean number of clusters over all rectangular $2^q \times 2^q$ subspaces by computing the edge cuts in and between its subgrids that are decomposed recursively. The idea behind this derivation is to count the the total number of edges that are cut by the sides of all possible identically shaped $2^q \times 2^q$ subspaces as observed in [MJFS01]. The reason is because the entry and exit grid points of a cluster connect to grid points outside of this subspace (two cuts by side(s) of this subspace); every cluster has two cuts by the subspace. A cut on an edge by a side of a subspace is called an "edge cut". We give an overview of the derivation for both curve families as follows.

1. Compute the number of edges that are cut by the sides of all possible $2^q \times 2^q$ subspaces, which are exactly inside of one of the four quadrants, and the number of edges cut by the sides of subspaces across different quadrants, respectively. The number of cuts on edges is twice the number of clusters over all identically shaped subspaces.

2. Categorize the edge cuts caused by subspaces across quadrants into: edge cuts within $2^q \times 2^q$ corner boundaries of the quadrants and within side boundaries

(of 2^q rows/columns) of the quadrants. (Note that the edges that are cut by subspaces across quadrants only in the boundary regions (sides and corners) of the quadrants.) By decomposing these corner and side boundaries, we derive recurrences for each of them:

(a) For edge cuts within one of the four corner boundaries: edge cuts within upper (left or right) corner and lower (left or right) corner boundaries are inter-recurrence related.

(b) For edge cuts within one of the four side boundaries:

i. Edge cuts in left boundary (same as right boundary) consists of substructures of left boundary, bottom boundary and two lower-corner boundaries,

ii. Edge cuts in bottom boundary consists of substructures of two left boundaries and two upper-corner boundaries, and

iii. Edge cuts in the top boundary consists of substructures of two top boundaries and two upper-corner boundaries. These inter-recurrent relations are based upon the construction of a canonical H_k^2 (see Figure 2.6).

3. After obtaining the numbers of edge cuts in boundaries that are derived from Step 2, we solve the recurrence for the total number of edge cuts, then divide it by 2 to get the total number of clusters. To obtain the mean number of clusters, the total number of clusters needs to be divided by the total number of subspaces of size $2^q \times 2^q$, which is $(2^k - 2^q + 1)^2$.

5.2 Analytical Study of Number of Clusters for Hilbert Curve

With respect to the canonical orientation of H_k^2 shown in Figure 2.6(a), we cover the 2-dimensional k-order grid with 2^k rows $(R_{k,1}, R_{k,2}, \ldots, R_{k,2^k})$, indexed from the bottom, and 2^k columns $(C_{k,1}, C_{k,2}, \ldots, C_{k,2^k})$, indexed from the left. We denote:

1. For a grid point $v \in [2^k]^2$, its x- and y-coordinate by $X(v)$ and $Y(v)$, respectively (that is, v is the intersection grid point of the row $R_{k,X(v)}$ and the column $C_{k,Y(v)}$),

2. For a rectangular query subgrid with its lower-left corner at grid point (x, y) and upper-right corner at grid point (x', y') ($1 \leq x \leq x' \leq 2^k$ and $1 \leq y \leq y' \leq 2^k$) covering $\cup_{\alpha=x}^{x'} R_{k,\alpha} \cap \cup_{\beta=y}^{y'} C_{k,\beta}$ by $G(x, y, x', y')$ ($= \{v \in [2^k]^2 \mid x \leq X(v) \leq x'$ and $y \leq Y(v) \leq y'\}$). The size of the query subgrid $G(x, y, x', y')$ is $(x' - x + 1) \times (y' - y + 1)$.

Notes:

1. axis-1 and axis-2 correspond to the x-axis and y-axis, respectively,

2. $(+\frac{\pi}{2})$-rotation and $(-\frac{\pi}{2})$-rotation correspond to the 90°-clockwise rotation and 90°-counterclockwise rotation from axis-1 to axis -2, respectively.

Remark 5.1 *For most self-similar m-dimensional order-k space-filling curve C_k^m indexing the grid $[2^k]^m$, we can view C_k^m as a C_{k-q}^m-curve interconnecting $2^{2(k-q)}$ C_q^m-subcurves for all $q \in [k]$.*

The remark above motivates our analytical study of clustering performances to be based upon query subgrids of size $2^q \times 2^q$.

For a 2-dimensional order-k Hilbert curve H_k^2, let $\mathcal{S}_{k,2^q}(H_k^2)$ denote the summation of all numbers of clusters over all $2^q \times 2^q$ query subgrids of an H_k^2-structural grid space $[2^k]^2$.

Remark 5.2 *Within a query subgrid G, the number of clusters is half of the number of edges of underlying space-filling curve that are cut by the sides of G [MJFS01].*

Denote by $\bar{n}(C, G)$ the number of edges within C that are cut by sides of subgrid G (without counting edge(s) connecting between C and other subcurves). Remark 5.2 translates the computation of the summation of all numbers of clusters over all identically shaped subgrids G to the computations of $\frac{1}{2}(\sum_{\text{all } G} \bar{n}(C, G) + 2)$ (the contribution of 2 is the number of cuts for connecting edges of C to other curves).

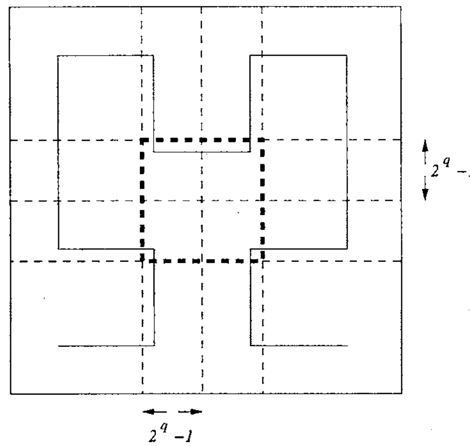


Figure 5.2: The boundary regions of neighboring quadrants are organized into nine regions.

For $2^q \times 2^q$ subgrids G, we denote by $E_q(C)$ all numbers of clusters over all identically shaped $2^q \times 2^q$ subgrids G, which is $\sum_{\text{all } G} \bar{n}(C,G) + 2$.

The recursive decomposition of H_k^2 (see Figure 2.6(b)) gives that

$$E_q(H_k^2) = 4E_q(H_{k-1}^2) + \varepsilon_{k,q}(H_k^2),$$

where $\varepsilon_{k,q}(H_k^2)$ denotes the summation of all edge cuts over all identically shaped $2^q \times 2^q$ query subgrids, each of which overlaps with more than one quadrant (that is, two or four). These query subgrids are contained in the boundary regions of neighboring quadrants as shown in Figure 5.2.

Remark 5.3 *For a 2-dimensional Hilbert curve H_k^2, the connecting edge between $Q_1(H_k^2)$ and $Q_2(H_k^2)$ is on the first column (left-most column), that between $Q_2(H_k^2)$ and $Q_3(H_k^2)$ is on the $2^{k-1}+1$-st row (the lowest row of these two quadrants), and that between $Q_3(H_k^2)$ and $Q_4(H_k^2)$ is on the 2^k-th column (right-most column).*

We denote the connecting edge between two quadrants $Q_i(H_k^2)$ and $Q_j(H_k^2)$ by a pair $(Q_i(H_k^2), Q_j(H_k^2))$. The previous remark tells the locations of the connecting edges. In addition to the cuts on connecting edges, the computation of $\varepsilon_{k,q}(H_k^2)$ is divided into two parts according to the overlaps of subspaces:

For a $2^q \times 2^q$ query subgrid G, G overlaps with:

1. exactly $Q_i(H_k^2)$ and $Q_{i \bmod 4+1}(H_k^2)$,

2. $Q_i(H_k^2)$ for all $i \in \{1, 2, 3, 4\}$.

We develop combinatorial lemmas in the following two subsections to support the computations. We denote by $G(x, y, x', y')$ the query subgrid, in which the lower-left grid point is (x, y) and the upper-right grid point (x', y').

5.2.1 $\sum \bar{n}(H_k^2, G)$ over Subgrids G Overlapping with Two Quadrants

Consider an arbitrary $2^q \times 2^q$ query subgrid G that exactly overlaps two quadrants $Q_i(H_k^2)$ and $Q_{i \bmod 4+1}(H_k^2)$, where $i \in \{1, 2, 3, 4\}$. The side-length is from 1 to $2^q - 1$ for the side across two quadrants. Since the quadrants are isomorphic to a canonical H_{k-1}^2 via symmetry (reflection and rotation), we consider the following system of summations $\Omega_{k,2^q} = (\Omega_{k,2^q}^L, \Omega_{k,2^q}^R, \Omega_{k,2^q}^B, \Omega_{k,2^q}^T)$ in a general context of a canonical H_k^2:

$$\Omega_{k,2^q}^L = \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(H_k^2, G(x, 1, x+2^q-1, y)) \text{ — for left boundary (see Figure 5.3(a)),}$$

$$\Omega_{k,2^q}^R = \sum_{x=1}^{2^k-2^q+1} \sum_{y=2^k-2^q+2}^{2^k} \bar{n}(H_k^2, G(x, y, y+2^q-1, 2^k)) \text{ — for right boundary,}$$

$$\Omega_{k,2^q}^B = \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^k-2^q+1} \bar{n}(H_k^2, G(1, y, x, y+2^q-1)) \text{ — for bottom boundary,}$$

$$\Omega_{k,2^q}^T = \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^k-2^q+1} \bar{n}(H_k^2, G(x, y, 2^k, y+2^q-1)) \text{ — for top boundary, and}$$

$$\mathcal{N}_{k,2^q}^S = \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} 1 \text{ — for the number of rectangular subgrids in a boundary for } \Omega_{k,2^q}.$$

We will establish a system of recurrences (in k) for $\Omega_{k,2^q}$ (see Lemma 5.4 below). The system of recurrence involves another system of summations as prerequisites, as demonstrated in the following example. Consider a recursive decomposition of $\Omega_{k,2^q}^L$, illustrated in Figure 5.3(b), into four parts: (1) $\Omega_{k-1,2^q}^B$, (2) $\Omega_{k-1,2^q}^{c_4}$, $\Omega_{k-1,2^q}^{c_1}$, (3) $\Omega_{k-1,2^q}^L$, and (4) the number of cuts on connecting edges. The part $\Omega_{k-1,2^q}^{c_1}$ ($\Omega_{k-1,2^q}^{c_4}$)

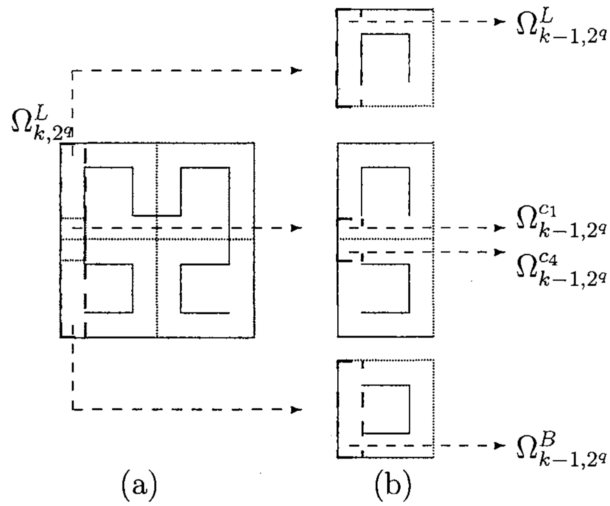


Figure 5.3: (a) $\Omega^{L}_{k,2^q}$ for a canonical H^2_k; (b) its recursive decomposition.

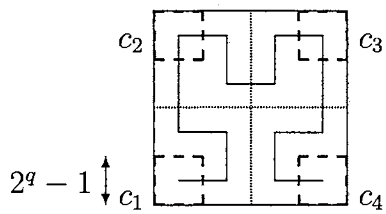


Figure 5.4: The four $(2^q-1)\times(2^q-1)$ corners of a canonical H^2_k.

computes $\sum \bar{n}(H^2_k, G)$ over all "incomplete" rectangular subgrids G (with one side-length at most 2^q-1) overlapping both $Q_1(H^2_k)$ and $Q_2(H^2_k)$. Each of the three parts $\Omega^{B}_{k-1,2^q}$, $\Omega^{c1}_{k-1,2^q}$ ($\Omega^{c4}_{k-1,2^q}$), and $\Omega^{L}_{k-1,2^q}$ is defined with respect to a canonical H^2_{k-1}. Note that $\Omega^{L}_{k,2^q} = \Omega^{R}_{k,2^q}$ because of the left-right symmetry property of H^2_k.

The recursive decompositions of all four parts in $\Omega^{L}_{k,2^q}$, $\Omega^{R}_{k,2^q}$, $\Omega^{B}_{k,2^q}$, and $\Omega^{T}_{k,2^q}$ lead us to consider a prerequisite system of summations $\Omega^{c}_{k,2^q} = (\Omega^{c1}_{k,2^q}, \Omega^{c2}_{k,2^q}, \Omega^{c3}_{k,2^q}, \Omega^{c4}_{k,2^q})$ in a more general context of a canonical H^2_k (see Figure 5.4):

$$\Omega^{c1}_{k,2^q} = \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} \bar{n}(H^2_k, G(1,1,x,y)) \text{ — for lower-left corner,}$$

$$\Omega^{c2}_{k,2^q} = \sum_{x=2^k-2^q+2}^{2^k}\ \sum_{y=1}^{2^q-1} \bar{n}(H^2_k, G(x,1,2^k,y)) \text{ — for upper-left corner,}$$

$$\Omega^{c3}_{k,2^q} = \sum_{x=2^k-2^q+2}^{2^k}\ \sum_{y=2^k-2^q+2}^{2^k} \bar{n}(H^2_k, G(x,y,2^k,2^k)) \text{ — for upper-right corner,}$$

$$\Omega^{c4}_{k,2^q} = \sum_{x=1}^{2^q-1} \sum_{y=2^k-2^q+2}^{2^k} \bar{n}(H^2_k, G(1,y,x,2^k)) \text{ — for lower-right corner,}$$

$$\mathcal{N}^c_{k,2^q} = \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} 1 \text{ — for the number of incomplete rectangular subgrids in a corner.}$$

These four summations involve rectangular subgrids contained in $(2^q - 1) \times (2^q - 1)$ corners. Note that $\Omega^{c1}_{k,2^q} = \Omega^{c4}_{k,2^q}$ and $\Omega^{c2}_{k,2^q} = \Omega^{c3}_{k,2^q}$ because of the left-right symmetry property of H^2_k. As suggested by Remark 5.1, we zoom in on the $2^q \times 2^q$ H^2_q-structural corners, and consider the following system of summations $\overline{\Omega}^c_{q,2^q} = (\overline{\Omega}^{c1}_{q,2^q}, \overline{\Omega}^{c2}_{q,2^q})$:

$$\overline{\Omega}^{c1}_{q,2^q} = \sum_{x=1}^{2^q} \sum_{y=1}^{2^q} \bar{n}(H^2_q, G(1,1,x,y)) \text{ — for lower(-left) corner,}$$

$$(= \sum_{x=1}^{2^q} \sum_{y=1}^{2^q} \bar{n}(H^2_q, G(1,y,x,2^q))) \text{ — for lower(-right) corner,}$$

$$\overline{\Omega}^{c2}_{q,2^q} = \sum_{x=1}^{2^q} \sum_{y=1}^{2^q} \bar{n}(H^2_q, G(x,1,2^q,y)) \text{ — for upper(-left) corner,}$$

$$(= \sum_{x=1}^{2^q} \sum_{y=1}^{2^q} \bar{n}(H^2_q, G(x,y,2^q,2^q))) \text{ — for upper(-right) corner,}$$

$$\overline{\mathcal{N}}^c_{q,2^q} = \sum_{x=1}^{2^q} \sum_{y=1}^{2^q} 1 \text{ — for the number of rectangular subgrids in a } 2^q \times 2^q \text{ corner.}$$

Thus far, we learn that the system of recurrences for $\Omega_{k,2^q}$ can be defined and solved via the prerequisite system $\Omega^c_{k,2^q}$, which is related to the system $\overline{\Omega}^c_{q,2^q}$ (see Lemma 5.3 below). The system $\overline{\Omega}^c_{q,2^q}$, which involves subgrids (with both side-lengths at most 2^q) of a canonical H^2_q, represents the basis of the recursive decompositions (in k to q) of $\Omega_{k,2^q}$ and $\Omega^c_{k,2^q}$. Similar to the reduction of $\Omega_{k,2^q}$ to $\Omega^c_{k,2^q}$, we develop a system of recurrences (in q) for $\overline{\Omega}^c_{q,2^q}$ via a prerequisite system, as demonstrated in the following example. Consider a recursive decomposition of $\overline{\Omega}^{c1}_{q,2^q} = \sum_{x=1}^{2^q} \sum_{y=1}^{2^q} \bar{n}(H^2_q, G(1,1,x,y))$ into four parts, based upon the overlapping scenario of the rectangular subgrid $G(1,1,x,y)$ with the four quadrants of a canonical H^2_q (see Figure 5.5).

Case 1: $G(1,1,x,y)$ is contained in $Q_1(H^2_q)$ (see Figure 5.5(a)). This part is

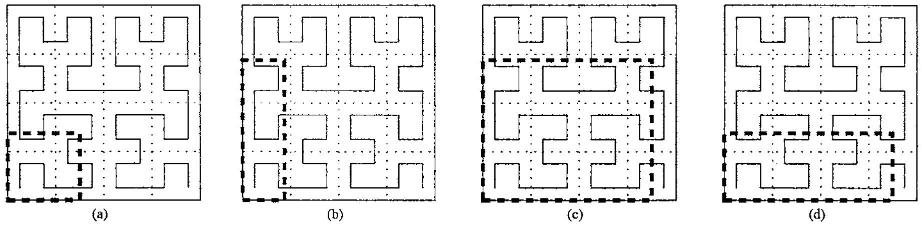


Figure 5.5: Four overlapping scenarios when decomposing $\overline{\Omega}^{c1}_{q,2^q}$ in a canonical H^2_q: (a) contained in $Q_1(H^2_q)$; (b) and (d) overlapping with exactly two quadrants; (c) overlapping with all quadrants.

reduced to $\overline{\Omega}^{c1}_{q-1,2^{q-1}}$ after $(-\frac{\pi}{2})$-rotating and then left-right reflecting $Q_1(H^2_q)$ into a canonical H^2_{q-1} and all numbers of cuts on connecting edge $(Q_1(H^2_q), Q_2(H^2_q))$ that are caused by the top (horizontal) side of G.

Case 2: $G(1,1,x,y)$ overlaps with exactly $Q_1(H^2_q)$ and $Q_2(H^2_q)$ (see Figure 5.5(b)). This part is reduced to $\overline{\Omega}^{c1}_{q-1,2^{q-1}}$ and all numbers of cuts on horizontal edge within $Q_1(H^2_q)$ and on connecting edge $(Q_2(H^2_q), Q_3(H^2_q))$ that are caused by the right (vertical) side of G.

Case 3: $G(1,1,x,y)$ overlaps with exactly $Q_1(H^2_q)$ and $Q_4(H^2_q)$ (see Figure 5.5(d)). This part is reduced to $\overline{\Omega}^{c2}_{q-1,2^{q-1}}$ after $(+\frac{\pi}{2})$-rotating and then left-right reflecting $Q_4(H^2_q)$ into a canonical H^2_{q-1} and all numbers of cuts on vertical edges within $Q_1(H^2_q)$ and on connecting edges $(Q_1(H^2_q), Q_2(H^2_q))$ and $(Q_3(H^2_q), Q_4(H^2_q))$ that are caused by the top (horizontal) side of G.

Case 4: $G(1,1,x,y)$ overlaps with exactly all the quadrants (see Figure 5.5(c)). This part is reduced to $\overline{\Omega}^{c1}_{q-1,2^{q-1}}$ and all numbers of cuts on vertical edges within $Q_2(H^2_q)$ and horizontal edges within $Q_4(H^2_q)$ that are caused by the top (horizontal) and right (vertical), respectively, sides of G.

The recursive decompositions of $\overline{\Omega}^{c1}_{q,2^q}$, and $\overline{\Omega}^{c2}_{q,2^q}$ lead us to consider a prerequisite system of summations $\overline{\Pi}_q = (\overline{\Pi}^v_q, \overline{\Pi}^h_q)$ in a general context of a canonical H^2_q:

$$\overline{\Pi}^v_q = \sum_{x=1}^{2^q} \bar{n}(H^2_q, G(1,1,x,2^q)) \text{ — number of vertical edges (edges cut by}$$

top (horizontal) sides of G that covers lower part of H^2_q)

$$\overline{\Pi}_q^h = \sum_{y=1}^{2^q} \bar{n}(H_q^2, G(1,1,2^q,y)) \text{ — number of horizontal edges (edges cut by right (vertical) sides of } G \text{ that covers left part of } H_q^2)$$

We develop and solve a system of recurrences for $\overline{\Pi}_q$ and reverse the sequence of reductions to obtain the closed-form solutions for $\Omega_{k,2^q}$, which are summarized in the following four lemmas.

Lemma 5.1 *For a canonical* H_q^2,

$$\overline{\Pi}_q^v = \begin{cases} 2\overline{\Pi}_{q-1}^v + 2\overline{\Pi}_{q-1}^h + 2 & \text{if } q > 1 \\ 2 & \text{if } q = 1 \end{cases}$$

$$\overline{\Pi}_q^h = \begin{cases} 2\overline{\Pi}_{q-1}^v + 2\overline{\Pi}_{q-1}^h + 1 & \text{if } q > 1 \\ 1 & \text{if } q = 1 \end{cases}$$

Proof. For $\overline{\Pi}_q^v$, the number of edge cuts by top (horizontal) sides of G can be computed from cuts within the four quadrants and cuts on the two connecting edges $(Q_1(H_q^2), Q_2(H_q^2))$ and $(Q_3(H_q^2), Q_4(H_q^2))$.

$$\overline{\Pi}_q^v = \sum_{x=1}^{2^q} \bar{n}(H_q^2, G(1,1,x,2^q))$$

$$= \sum_{x=1}^{2^{q-1}} \bar{n}(H_q^2, G(1,1,x,2^q)) \quad \text{(cuts on vertical edge within } Q_1(H_q^2), Q_4(H_q^2) \text{ plus cuts on two connecting edges when } x = 2^{q-1})$$

$$+ \sum_{x=2^{q-1}+1}^{2^q} \bar{n}(H_q^2, G(1,1,x,2^q)) \quad \text{(cuts on vertical edges within } Q_2(H_q^2), Q_3(H_q^2))$$

$$= \Big(\sum_{x=1}^{2^{q-1}} \bar{n}(Q_1(H_q^2), G(1,1,x,2^{q-1})) \quad \text{(cuts on vertical edges within } Q_1(H_q^2))$$

$$+ \sum_{x=1}^{2^{q-1}} \bar{n}(Q_4(H_q^2), G(1,2^{q-1}+1,x,2^q)) \quad \text{(cuts on vertical edges within } Q_4(H_q^2))$$

$$+ 2\Big) \quad \text{(cuts on connecting edges } (Q_1(H_q^2), Q_2(H_q^2)), (Q_3(H_q^2), Q_4(H_q^2)) \text{ when } x = 2^{q-1})$$

$$+(\sum_{x=2^{q-1}+1}^{2^q} \bar{n}(Q_2(H_q^2), G(2^{q-1}+1, 1, x, 2^{q-1}))$$

(cuts on vertical edges within $Q_2(H_q^2)$)

$$+ \sum_{x=2^{q-1}+1}^{2^q} \bar{n}(Q_3(H_q^2), G(2^{q-1}+1, 2^{q-1}+1, x, 2^q)))$$

(cuts on vertical edges within $Q_3(H_q^2)$)

$$= (\sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1,1,2^{q-1},y)) \quad \text{(after } (-\frac{\pi}{2})\text{-rotating and left-right}$$

reflecting $Q_1(H_q^2)$ into canonical H_{q-1}^2)

$$+ \sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1,1,2^{q-1},y)) \quad \text{(after } (+\frac{\pi}{2})\text{-rotating and left-right}$$

reflecting $Q_4(H_q^2)$ into canonical H_{q-1}^2)

$$+ 2)$$

$$+(\sum_{x=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1,1,x,2^{q-1})) \quad (Q_2(H_q^2)\text{: a canonical} H_{q-1}^2)$$

$$+ \sum_{x=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1,1,x,2^{q-1}))) \quad (Q_3(H_q^2)\text{: a canonical} H_{q-1}^2)$$

$$= (\overline{\Pi}_{q-1}^h + \overline{\Pi}_{q-1}^h + 2) + (\overline{\Pi}_{q-1}^v + \overline{\Pi}_{q-1}^v) = 2\overline{\Pi}_{q-1}^v + 2\overline{\Pi}_{q-1}^h + 2.$$

The proof of $\overline{\Pi}_q^h$ is similar to that of $\overline{\Pi}_q^v$. ∎

The closed-form solutions for $\overline{\Pi}_q$ are employed to establish a system of recurrences for $\overline{\Omega}_{q,2^q}^c$.

Lemma 5.2 *For a canonical* H_q^2,

$$\overline{\Omega}_{q,2^q}^{c_1} = \begin{cases} 3\overline{\Omega}_{q-1,2^{q-1}}^{c_1} + \overline{\Omega}_{q-1,2^{q-1}}^{c_2} + 3\cdot 2^{q-1}\cdot \overline{\Pi}_{q-1}^v + 2^{q-1}\cdot \overline{\Pi}_{q-1}^h \\ \quad + 3\cdot 2^{q-1} + 1 & \text{if } q > 1, \\ 4 & \text{if } q = 1; \end{cases}$$

$$\overline{\Omega}_{q,2^q}^{c_2} = \begin{cases} \overline{\Omega}_{q-1,2^{q-1}}^{c_1} + 3\overline{\Omega}_{q-1,2^{q-1}}^{c_2} + 2^{q-1}\cdot \overline{\Pi}_{q-1}^v + 3\cdot 2^{q-1}\cdot \overline{\Pi}_{q-1}^h \\ \quad + 3\cdot 2^{q-1} + 2 & \text{if } q > 1, \\ 5 & \text{if } q = 1. \end{cases}$$

Proof. As in Figure 5.5 and the case discussion for $\Omega_{q,2^q}^{c_1}$, we can split $\Omega_{q,2^q}^{c_1}$ into four parts. Thus,

$$\overline{\Omega}^{C_1}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \bar{n}(H_q^2, G(1,1,x,y))$$

$$= \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} \bar{n}(H_q^2, G(1,1,x,y)) \qquad \text{(Figure 5.5(a))}$$

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} \bar{n}(H_q^2, G(1,1,x,y)) \qquad \text{(Figure 5.5(b))}$$

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(H_q^2, G(1,1,x,y)) \qquad \text{(Figure 5.5(c))}$$

$$+ \sum_{x=1}^{2^{q-1}}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(H_q^2, G(1,1,x,y)) \qquad \text{(Figure 5.5(d))}$$

$$= (\sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} \bar{n}(Q_1(H_q^2), G(1,1,x,y)) \qquad \text{(cuts within } Q_1(H_q^2))$$

$$+ \sum_{x=2^{q-1}}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} 1) \qquad \text{(cuts on connecting edge } (Q_1(H_q^2), Q_2(H_q^2)))$$

$$+(\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} \bar{n}(Q_2(H_q^2), G(2^{q-1}+1,1,x,y)) \qquad \text{(cuts within } Q_2(H_q^2))$$

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} \bar{n}(Q_1(H_q^2), G(1,1,2^{q-1},y)) \qquad \text{(cuts within } Q_1(H_q^2)$$

by vertical side of G)

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}}^{2^{q-1}} 1) \qquad \text{(cuts on connecting edge } (Q_2(H_q^2), Q_3(H_q^2)))$$

$$+(\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_3(H_q^2), G(2^{q-1}+1, 2^{q-1}+1, x, y))$$

(cuts within $Q_3(H_q^2)$)

$$+ \sum_{x=1}^{2^{q-1}}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_2(H_q^2), G(2^{q-1}+1, 1, x, 2^{q-1})) \qquad \text{(cuts within } Q_2(H_q^2)$$

by horizontal side of G)

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_4(H_q^2), G(1, 2^{q-1}+1, 2^{q-1}, y)))$$

(cuts within $Q_4(H_q^2)$ by vertical side of G)

$$+(\sum_{x=1}^{2^{q-1}} \sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_4(H_q^2), G(1, 2^{q-1}+1, x, y)) \quad \text{(cuts within } Q_4(H_q^2))$$

$$+\sum_{x=1}^{2^{q-1}} \sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_1(H_q^2), G(1, 1, x, 2^{q-1})) \quad \text{(cuts within } Q_1(H_q^2)$$

by horizontal side of G)

$$+\sum_{x=2^{q-1}}^{2^{q-1}} \sum_{y=2^{q-1}+1}^{2^q} 1 \quad \text{(cuts on connecting edge } (Q_1(H_q^2), Q_2(H_q^2)))$$

$$+\sum_{x=2^{q-1}}^{2^{q-1}} \sum_{y=2^q}^{2^q} 1) \quad \text{(cuts on connecting edge } (Q_3(H_q^2), Q_4(H_q^2)))$$

$$= (\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1, 1, x, 2^{q-1})) \quad \text{(after } (-\frac{\pi}{2}) - \text{rotating (and left-right}$$

reflecting) $Q_1(H_q^2)$ into a canonical H_{q-1}^2)

$$+ (2^{q-1}))$$

$$+(\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1, 1, x, y)) \quad (Q_2(H_q^2) \text{ a canonical } H_{q-1}^2)$$

$$+\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1, 1, x, 2^{q-1})) \quad \text{(after } (-\frac{\pi}{2}) - \text{rotating (and}$$

left-right reflecting) $Q_1(H_q^2)$ into a canonical H_{q-1}^2)

$$+ (2^{q-1}))$$

$$+(\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1, 1, x, y)) \quad (Q_3(H_q^2) \text{ a canonical } H_{q-1}^2)$$

$$+\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1, 1, x, 2^{q-1})) \quad (Q_2(H_q^2)\text{: a canonical } H_{q-1}^2)$$

$$+\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1, 1, x, 2^{q-1}))) \quad \text{(after } (+\frac{\pi}{2}) - \text{rotating (and}$$

left-right reflecting) $Q_4(H_q^2)$ into a canonical H_{q-1}^2)

$$+(\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(H_{q-1}^2, G(1, 1, x, y)) \quad \text{(after } (+\frac{\pi}{2}) - \text{rotating (and left-right}$$

reflecting) $Q_4(H_q^2)$ into a canonical H_{q-1}^2)

$$+\sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}}\bar{n}(H^2_{q-1},G(1,1,2^{q-1},y)) \quad \text{(after } (-\frac{\pi}{2}) - \text{rotating (and}$$

$$\text{left-right reflecting) } Q_1(H^2_q) \text{ into a canonical } H^2_{q-1})$$

$$+ (2^{q-1}) + 1)$$

$$\begin{aligned}
&= (\overline{\Omega}^{c1}_{q-1,2^{q-1}} + 2^{q-1}) \\
&\quad +(\overline{\Omega}^{c1}_{q-1,2^{q-1}} + \overline{\Pi}^{v}_{q-1}\cdot 2^{q-1} + 2^{q-1}) \\
&\quad +(\overline{\Omega}^{c1}_{q-1,2^{q-1}} + \overline{\Pi}^{v}_{q-1}\cdot 2^{q-1} + \overline{\Pi}^{v}_{q-1}\cdot 2^{q-1}) \\
&\quad +(\overline{\Omega}^{c2}_{q-1,2^{q-1}} + \overline{\Pi}^{h}_{q-1}\cdot 2^{q-1} + 2^{q-1} + 1) \\
&= 3\overline{\Omega}^{c1}_{q-1,2^{q-1}} + \overline{\Omega}^{c2}_{q-1,2^{q-1}} + 3\cdot 2^{q-1}\cdot\overline{\Pi}^{v}_{q-1} + 2^{q-1}\cdot\overline{\Pi}^{h}_{q-1} + 3\cdot 2^{q-1} + 1.
\end{aligned}$$

The proof for $\overline{\Omega}^{c2}_{q,2^q}$ is similar to this one. ∎

The closed-form solutions for $\overline{\Omega}^{c}_{q,2^q}$ and $\overline{\Pi}_q$ are employed to obtain exact formulas for $\Omega^c_{k,2^q}$.

Lemma 5.3 *For a canonical H^2_k structured as an H^2_{k-q}-curve interconnecting $2^{2(k-q)}$ H^2_q-subcurves,*

$$\begin{aligned}
\Omega^{c1}_{k,2^q} &= \overline{\Omega}^{c1}_{q,2^q} - \overline{\Pi}^{h}_{q} - \overline{\Pi}^{v}_{q}, \\
\Omega^{c2}_{k,2^q} &= \overline{\Omega}^{c2}_{q,2^q} - \overline{\Pi}^{h}_{q} - \overline{\Pi}^{v}_{q}.
\end{aligned}$$

Proof. By the definition, we have

$$\begin{aligned}
\Omega^{c1}_{k,2^q} &= \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1}\bar{n}(H^2_k,G(1,1,x,y)) \\
&= \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q}\bar{n}(H^2_k,G(1,1,x,y)) - \sum_{x=1}^{2^q-1}\sum_{y=2^q}^{2^q}\bar{n}(H^2_k,G(1,1,x,y)) \\
&= (\sum_{x=1}^{2^q}\sum_{y=1}^{2^q}\bar{n}(H^2_k,G(1,1,x,y)) - \sum_{x=2^q}^{2^q}\sum_{y=1}^{2^q}\bar{n}(H^2_k,G(1,1,x,y))) \\
&\quad -(\sum_{x=1}^{2^q}\sum_{y=2^q}^{2^q}\bar{n}(H^2_k,G(1,1,x,y)) - \sum_{x=2^q}^{2^q}\sum_{y=2^q}^{2^q}\bar{n}(H^2_k,G(1,1,x,y))) \\
&= \overline{\Omega}^{c1}_{q,2^q} - \overline{\Pi}^{h}_{q} - \overline{\Pi}^{v}_{q}.
\end{aligned}$$

The proof of $\Omega^{c2}_{k,2^q}$ is similar to this one. ∎

The exact formulas for $\Omega^c_{k,2^q}$ are employed to establish a system of recurrences for $\Omega_{k,2^q}$.

Lemma 5.4 *For a canonical H^2_k structured as an H^2_{k-q}-curve interconnecting $2^{2(k-q)}$ H^2_q-subcurves,*

$$\Omega^L_{k,2^q} = \begin{cases} \Omega^B_{k-1,2^q} + \Omega^L_{k-1,2^q} + 2\Omega^{c1}_{k-1,2^q} + 2(2^q-1) & \text{if } k > q, \\ \overline{\Pi}^h_q & \text{if } k = q; \end{cases}$$

$$\Omega^B_{k,2^q} = \begin{cases} 2\Omega^L_{k-1,2^q} + 2\Omega^{c2}_{k-1,2^q} & \text{if } k > q, \\ \overline{\Pi}^v_q & \text{if } k = q; \end{cases}$$

$$\Omega^T_{k,2^q} = \begin{cases} 2\Omega^T_{k-1,2^q} + 2\Omega^{c2}_{k-1,2^q} & \text{if } k > q, \\ \overline{\Pi}^v_q & \text{if } k = q. \end{cases}$$

Proof. Similar to the proof of Lemma 5.3, from the definition, we have (see Figure 5.3)

$$\Omega^L_{k,2^q} = \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(H^2_k, G(x,1,x+2^q-1,y))$$

$$= \sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(H^2_k, G(x,1,x+2^q-1,y)) \quad (G \text{ in } Q_1(H^2_k))$$

$$+ \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \bar{n}(H^2_k, G(x,1,x+2^q-1,y)) \quad (G \text{ across } Q_1(H^2_k), Q_2(H^2_k))$$

$$+ \sum_{x=2^{k-1}+1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(H^2_k, G(x,1,x+2^q-1,y)) \quad (G \text{ in } Q_2(H^2_k))$$

$$= \Big(\sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Q_1(H^2_k), G(x,1,x+2^q-1,y)) \quad (\text{cuts within } Q_1(H^2_k))$$

$$+ \sum_{x=2^{k-1}-2^q+1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} 1\Big) \quad (\text{cuts on connecting edge } (Q_1(H^2_q), Q_2(H^2_q)))$$

$$+\Big(\sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \bar{n}(Q_1(H^2_k), G(x,1,2^{k-1},y)) \quad (\text{zooming in } Q_1(H^2_k))$$

$$+ \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \bar{n}(Q_2(H^2_k), G(2^{k-1}+1,1,x+2^q-1,y))\Big)$$

$$
\begin{aligned}
& \qquad \text{(zooming in } Q_2(H_k^2)) \\
& +(\sum_{x=2^{k-1}+1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Q_2(H_k^2), G(x,1,x+2^q-1,y)) \quad \text{(cuts within } Q_2(H_k^2)) \\
& \quad + \sum_{x=2^{k-1}+1}^{2^{k-1}+1} \sum_{y=1}^{2^q-1} 1) \quad \text{(cuts on connecting edge } (Q_1(H_q^2), Q_2(H_q^2))) \\
= & \ (\sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(H_{k-1}^2, G(x,1,x+2^q-1,y)) \quad \text{(after } (-\frac{\pi}{2})\text{-rotating and} \\
& \qquad \text{left-right reflecting } Q_1(H_k^2) \text{ into a canonical } H_{k-1}^2) \\
& \quad + (2^q-1)) \\
& +(\sum_{x=1}^{2^q-1} \sum_{y=2^{k-1}-2^q+2}^{2^{k-1}} \bar{n}(H_{k-1}^2, G(1,y,x,2^{k-1})) \quad \text{(after } (-\frac{\pi}{2})\text{-rotating and} \\
& \qquad \text{left-right reflecting } Q_1(H_k^2) \text{ into a canonical } H_{k-1}^2) \\
& \quad + \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} \bar{n}(Q_2(H_k^2), G(1,1,x,y))) \quad (Q_2(H_k^2)\text{: a canonical } H_{k-1}^2) \\
& +(\sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(H_{k-1}^2, G(x,1,x+2^q-1,y)) \quad (Q_2(H_k^2)\text{: a canonical } H_{k-1}^2) \\
& \quad + (2^q-1)) \\
= & \ (\Omega_{k-1,2^q}^B + (2^q-1)) \\
& +(\Omega_{k-1,2^q}^{c_1} + \Omega_{k-1,2^q}^{c_4}) \quad (\Omega_{k-1,2^q}^{c_1} = \Omega_{k-1,2^q}^{c_4}) \\
& +(\Omega_{k-1,2^q}^B + (2^q-1)) \\
= & \ \Omega_{k-1,2^q}^L + \Omega_{k-1,2^q}^B + 2\Omega_{k-1,2^q}^{c_1} + 2(2^q-1).
\end{aligned}
$$

For $\Omega_{k-1,2^q}^B$ and $\Omega_{k-1,2^q}^T$, the proofs are similar to this one. ∎

We obtain the closed-form solutions for $\Omega_{k,2^q}$ by using the mathematical software Maple.

5.2.2 Query Subgrids Overlapping with All Quadrants

For a $2^q \times 2^q$ query subgrid G that overlaps four quadrants around the center of H_k^2, when zooming in on the incomplete rectangular subgrid $G \cap G_1$ (with both side-lengths at most $2^q - 1$), where G_1 denotes the subspace of $Q_1(H_k^2)$, we reduce $\sum_{\text{all } G \cap G_1} \bar{n}(H_k^2, G \cap G_1)$ to $\Omega_{k-1,2^q}^{c3} (= \Omega_{k-1,2^q}^{c2})$ after $(-\frac{\pi}{2})$-rotating and left-right reflecting $Q_1(H_k^2)$ into a canonical H_{k-1}^2. Similar consideration leads to reductions of $\sum_{\text{all } G \cap G'} \bar{n}(H_k^2, G \cap G')$ to $\Omega_{k-1,2^q}^{c4} (= \Omega_{k-1,2^q}^{c1})$, $\Omega_{k-1,2^q}^{c1}$ and $\Omega_{k-1,2^q}^{c2}$ when $G \cap G'$ denotes the subspace for G overlapping $Q_2(H_k^2)$, $Q_3(H_k^2)$, or $Q_3(H_k^2)$, respectively.

Thus, the summation of numbers of edge cuts for all $2^q \times 2^q$ query subgrids G that overlap all four quadrants is

$$2\Omega_{k-1,2^q}^{c2} + 2\Omega_{k-1,2^q}^{c1}.$$

5.2.3 The Big Picture: Computing $E_q(H_k^2)$

The results in the previous three subsections yield $\varepsilon_{k,q}(H_k^2)$. Hence, we have the following lemma for $E_q(H_k^2)$.

Lemma 5.5 *For a canonical H_k^2, the recurrence for total number of cuts on edges by all $2^q \times 2^q$ subgrids G:*

$$E_q(H_k^2) = \begin{cases} 4E_q(H_{k-1}^2) + (\Omega_{k-1,2^q}^{L} + \Omega_{k-1,2^q}^{B} + (2^q - 1)) \\ \quad + (\Omega_{k-1,2^q}^{L} + \Omega_{k-1,2^q}^{L}) \\ \quad + (\Omega_{k-1,2^q}^{L} + \Omega_{k-1,2^q}^{B} + (2^q - 1)) \\ \quad + (\Omega_{k-1,2^q}^{T} + \Omega_{k-1,2^q}^{T}) \\ \quad + (2\Omega_{k-1,2^q}^{c1} + 2\Omega_{k-1,2^q}^{c2}) & \text{if } k > q, \\ \\ 2 & \text{if } k = q. \end{cases}$$

Proof. Similar to the proofs of Lemmas 5.1 to 5.4.

Case 1: G overlaps with exactly $Q_1(H_k^2)$ and $Q_2(H_k^2)$. This part is reduced to $\Omega_{k-1,2^q}^{R} (= \Omega_{k-1,2^q}^{L})$ (cuts on $Q_1(H_k^2)$), $\Omega_{k-1,2^q}^{B}$ (cuts on $Q_2(H_k^2)$), and $2^q - 1$ cuts on the connecting edge $(Q_2(H_k^2), Q_3(H_k^2))$.

Case 2: G overlaps with exactly $Q_2(H_k^2)$ and $Q_3(H_k^2)$. This part is reduced to $\Omega_{k-1,2^q}^{R} (= \Omega_{k-1,2^q}^{L})$ (cuts on $Q_2(H_k^2)$), and $\Omega_{k-1,2^q}^{L}$ (cuts on $Q_3(H_k^2)$).

Case 3: G overlaps with exactly $Q_3(H_k^2)$ and $Q_4(H_k^2)$. This part is reduced to $\Omega_{k-1,2^q}^{B}$ (cuts on $Q_3(H_k^2)$), $\Omega_{k-1,2^q}^{L}$ (cuts on $Q_4(H_k^2)$), and $2^q - 1$ cuts on the connecting edge $(Q_2(H_k^2), Q_3(H_k^2))$.

Case 4: G overlaps with exactly $Q_1(H_k^2)$ and $Q_4(H_k^2)$. This part is reduced to $\Omega^T_{k-1,2^q}$ (cuts on $Q_1(H_k^2)$), and $\Omega^T_{k-1,2^q}$ (cuts on $Q_4(H_k^2)$).

Case 5: G overlaps with exactly all four quadrants. This part is reduced to $\Omega^{c3}_{k-1,2^q}(=\Omega^{c2}_{k-1,2^q})$ (cuts on $Q_1(H_k^2)$), $\Omega^{c4}_{k-1,2^q}(=\Omega^{c1}_{k-1,2^q})$ (cuts on $Q_2(H_k^2)$), $\Omega^{c1}_{k-1,2^q}$ (cuts on $Q_3(H_k^2)$),and $\Omega^{c2}_{k-1,2^q}$ (cuts on $Q_4(H_k^2)$).

Combining all the five cases, complete the recurrence.

For the case of $k=q$, there are two cuts that are the edge cut between entry grid point and other curve, and the edge cut between exit grid point and other curve. ▮

Therefore, the exact formula for $E_q(H_k^2)$ is:

$$E_q(H_k^2) = 2^{2k+q+1} - 2^{k+2q+2} + 2^{k+q+1} + 2^{k-q+1} + 2^{3q+1} - 2^{2q+1}.$$

The summation of all numbers of clusters over all identically shaped $2^q \times 2^q$ query subgrids of an H_k^2-structural grid space $[2^k]^2$ is

$$\mathcal{S}_{k,2^q}(H_k^2) = \frac{E_q(H_k^2)}{2}.$$

The mean number of cluster within a subspace of size $2^q \times 2^q$ for H_k^2 is

$$\frac{\mathcal{S}_{k,2^q}(H_k^2)}{(2^k - 2^q + 1)^2} = \frac{E_q(H_k^2)}{2(2^k - 2^q + 1)^2}.$$

Thus, the exact formula for the mean number of cluster within a subspace $2^q \times 2^q$ for H_k^2 is corollarily derived.

Theorem 5.1 *The mean number of cluster over all identical subspaces $2^q \times 2^q$ for H_k^2 is*

$$\frac{2^{2k+q+1} - 2^{k+2q+2} + 2^{k+q+1} + 2^{k-q+1} + 2^{3q+1} - 2^{2q+1}}{2(2^k - 2^q + 1)^2}.$$

5.3 Analytical Study of Number of Clusters for z-Order Curve

With respect to the canonical orientation of Z_k^2 shown in Figure 2.3(a), we apply the same approach and notations as in previous section to derive the exact formula for $\mathcal{S}_{k,2^q}(Z_k^2)$, which is the summation of all numbers of clusters over all identically shaped $2^q \times 2^q$ query subgrids of an Z_k^2-structural grid space $[2^k]^2$.

Like the approach in previous section, we compute the summation of all numbers of edge cuts over all identically shaped $2^q \times 2^q$ subgrids G by the recursive decomposition of Z_k^2:

$$E_q(Z_k^2) = 4E_q(Z_{k-1}^2) + \varepsilon_{k,q}(Z_k^2),$$

where $\varepsilon_{k,q}(Z_k^2)$ denotes the summation of all edge cuts over all $2^q \times 2^q$ query subgrids, each of which overlaps with more than one quadrant (that is, two or four). These query subgrids are contained in the boundary regions of neighboring quadrants.

We set up the systems for $E_q(Z_k^2)$ similar to that for $E_q(H_k^2)$ in previous section.

5.3.1 $\sum \bar{n}(Z_k^2, G)$ over Subgrids G Overlapping with Two Quadrants

Consider an arbitrary $2^q \times 2^q$ query subgrid G that exactly overlaps two quadrant $Q_i(Z_k^2)$ and $Q_j(Z_k^2)$, where $(i,j) \in \{(1,2),(1,3),(2,4),(3,4)\}$. The side-length is from 1 to $2^q - 1$ for the side across two quadrants. Since all the quadrants are isomorphic to a canonical Z_{k-1}^2, we consider the following system of summations $\Omega_{k,2^q} = (\Omega_{k,2^q}^L, \Omega_{k,2^q}^R, \Omega_{k,2^q}^B, \Omega_{k,2^q}^T)$ in a general context of a canonical Z_k^2:

$$\begin{aligned}
\Omega_{k,2^q}^L &= \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Z_k^2, G(x,1,x+2^q-1,y)) \text{ — for left boundary} \\
&\qquad \text{(see Figure 5.6(a)),} \\
\Omega_{k,2^q}^R &= \sum_{x=1}^{2^k-2^q+1} \sum_{y=2^k-2^q+2}^{2^k} \bar{n}(Z_k^2, G(x,y,x+2^q-1,2^k)) \text{ — for right boundary,} \\
\Omega_{k,2^q}^B &= \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^k-2^q+1} \bar{n}(Z_k^2, G(1,y,x,y+2^q-1)) \text{ — for bottom boundary,} \\
\Omega_{k,2^q}^T &= \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^k-2^q+1} \bar{n}(Z_k^2, G(x,y,2^k,y+2^q-1)) \text{ — for top boundary, and} \\
\mathcal{N}_{k,2^q}^S &= \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} 1 \text{ — for the number of rectangular subgrids} \\
&\qquad \text{in a boundary for } \Omega_{k,2^q}.
\end{aligned}$$

We will establish a system of recurrences (in k) for $\Omega_{k,2^q}$ (see Lemma 5.9 below). Note, $\Omega_{k,2^q}^L = \Omega_{k,2^q}^R$ and $\Omega_{k,2^q}^B = \Omega_{k,2^q}^T$ because of the property of symmetry for Z_k^2.

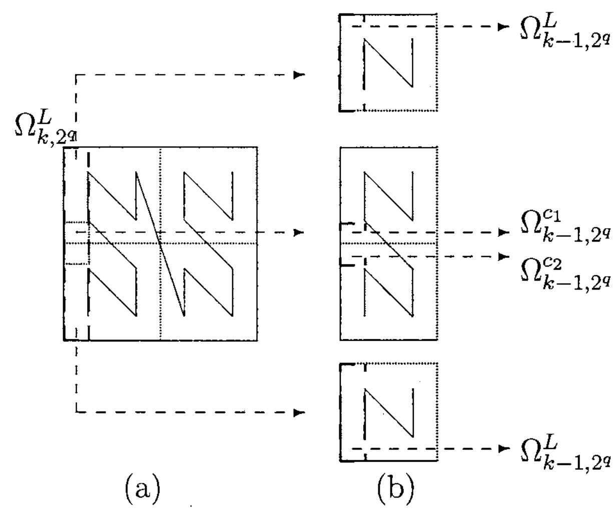


Figure 5.6: (a) $\Omega^L_{k,2^q}$ for a canonical Z^2_k; (b) its recursive decomposition.

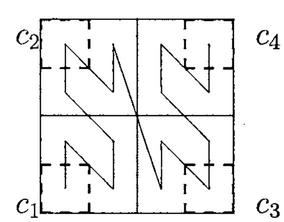


Figure 5.7: The four $(2^q - 1) \times (2^q - 1)$ corners of a canonical Z^2_k.

The recursive decompositions of all four parts in $\Omega^L_{k,2^q}$, $\Omega^R_{k,2^q}$, $\Omega^B_{k,2^q}$, and $\Omega^T_{k,2^q}$ require a prerequisite system of summations $\Omega^c_{k,2^q} = (\Omega^{c_1}_{k,2^q}, \Omega^{c_2}_{k,2^q}, \Omega^{c_3}_{k,2^q}, \Omega^{c_4}_{k,2^q})$ in a more general context of a canonical Z^2_k (see Figure 5.7):

$$\Omega^{c_1}_{k,2^q} = \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} \bar{n}(Z^2_k, G(1,1,x,y)) \text{ — for lower-left corner,}$$

$$\Omega^{c_2}_{k,2^q} = \sum_{x=2^k-2^q+2}^{2^k}\ \sum_{y=1}^{2^q-1} \bar{n}(Z^2_k, G(x,1,2^k,y)) \text{ — for upper-left corner,}$$

$$\Omega^{c_3}_{k,2^q} = \sum_{x=1}^{2^q}\ \sum_{xy=2^k-2^q+2}^{2^k} \bar{n}(Z^2_k, G(1,y,x,2^k)) \text{ — for lower-right corner,}$$

$$\Omega^{c_4}_{k,2^q} = \sum_{x=2^k-2^q+2}^{2^k}\ \sum_{y=2^k-2^q+2}^{2^k} \bar{n}(Z^2_k, G(x,y,2^k,2^k)) \text{ — for upper-right corner,}$$

$$\mathcal{N}^c_{k,2^q} = \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} 1 \text{ — for the number of incomplete rectangular subgrids in a corner.}$$

Note, $\Omega^{c_1}_{k,2^q} = \Omega^{c_4}_{k,2^q}$ and $\Omega^{c_2}_{k,2^q} = \Omega^{c_3}_{k,2^q}$ because of the symmetry property of Z^2_k. To compute $\Omega^c_{k,2^q}$, we set up the following system of summations $\overline{\Omega}^c_{q,2^q} = (\overline{\Omega}^{c_1}_{q,2^q}, \overline{\Omega}^{c_2}_{q,2^q})$:

$$\overline{\Omega}^{c_1}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \bar{n}(Z^2_q, G(1,1,x,y)) \text{ — for lower-left corner,}$$

$$(= \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \bar{n}(Z^2_q, G(x,y,2^q,2^q))) \text{ — for upper-right corner,}$$

$$\overline{\Omega}^{c_2}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \bar{n}(Z^2_q, G(x,1,2^q,y)) \text{ — for upper-left corner,}$$

$$(= \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \bar{n}(Z^2_q, G(1,y,x,2^q))) \text{ — for lower-right corner,}$$

$$\overline{\mathcal{N}}^c_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} 1 \text{ — for the number of rectangular subgrids in a } 2^q \times 2^q \text{ corner.}$$

Similar to the reduction of $\Omega_{k,2^q}$ to $\Omega^c_{k,2^q}$, we develop a system of recurrences (in q) for $\overline{\Omega}^c_{q,2^q}$ via a prerequisite system $\overline{\Pi}_q = (\overline{\Pi}^v_q, \overline{\Pi}^h_q)$.

This prerequisite system of summations in a general context of a canonical Z^2_q:

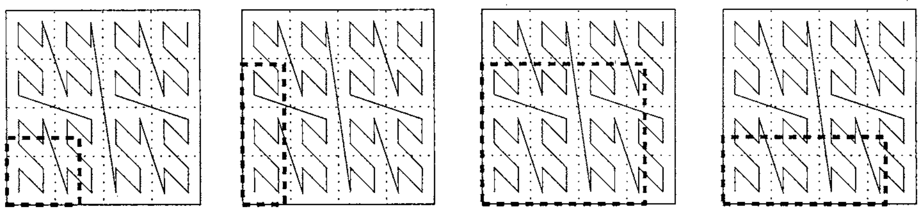


Figure 5.8: Four overlapping scenarios when decomposing $\overline{\Omega}^{c_1}_{q,2^q}$ in a canonical Z^2_q: (a) contained in $Q_1(Z^2_q)$; (b) and (d) overlapping with exactly two quadrants; (c) overlapping with all quadrants.

$$\overline{\Pi}^v_q = \sum_{x=1}^{2^q} \bar{n}(Z^2_k, G(1,1,x,2^q)) \text{ — number of edge cuts by horizontal lines, and}$$

$$\overline{\Pi}^h_q = \sum_{y=1}^{2^q} \bar{n}(Z^2_k, G(1,1,2^q,y)) \text{ — number of edge cuts by vertical lines.}$$

We develop and solve a system of recurrences for $\overline{\Pi}_q$ and reverse the sequence of reductions to obtain the closed-form solutions for $\Omega_{k,2^q}$, which are summarized in the following four lemmas.

Lemma 5.6 *For a canonical Z^2_q,*

$$\overline{\Pi}^v_q = \begin{cases} 4\overline{\Pi}^v_{q-1} + 2^q + 1 & \text{if } q > 1 \\ 3 & \text{if } q = 1 \end{cases}$$

$$\overline{\Pi}^h_q = \begin{cases} 4\overline{\Pi}^h_{q-1} + 2(2^{q-1} - 1) + 1 & \text{if } q > 1 \\ 1 & \text{if } q = 1 \end{cases}$$

Proof. The number of edges that are cut by horizontal lines can be computed from the edge cuts within four quadrants plus cuts on three connecting edges $(Q_1(Z^2_q), Q_2(Z^2_q))$, $(Q_3(Z^2_q), Q_4(Z^2_q))$, and $(Q_2(Z^2_q), Q_3(Z^2_q))$.

$$\overline{\Pi}^v_q = \sum_{x=1}^{2^q} \bar{n}(Z^2_q, G(1,1,x,2^q))$$

$$= \sum_{x=1}^{2^{q-1}} \bar{n}(Z^2_q, G(1,1,x,2^q)) \quad \text{(cuts within } Q_1(Z^2_q), Q_3(Z^2_q)$$

plus the cuts on connecting edges)

$$+ \sum_{x=2^{q-1}+1}^{2^q} \bar{n}(Z^2_q, G(1,1,x,2^q)) \quad \text{(cuts within } Q_2(Z^2_q), Q_4(Z^2_q)$$

plus the cuts on connecting edge)

$$
\begin{aligned}
= \;& (\sum_{x=1}^{2^{q-1}} \bar{n}(Q_1(Z_q^2), G(1,1,x,2^{q-1})) \quad \text{(cuts within } Q_1(Z_q^2))\\
& \quad + \sum_{x=1}^{2^{q-1}} \bar{n}(Q_3(Z_q^2), G(1,2^{q-1}+1,x,2^q)) \quad \text{(cuts within } Q_3(Z_q^2))\\
& \quad + \sum_{x=2^{q-1}}^{2^{q-1}} 1 \quad \text{(cuts on connecting edge } (Q_1(Z_q^2), Q_2(Z_q^2)))\\
& \quad + \sum_{x=1}^{2^{q-1}} 1 \quad \text{(cuts on connecting edge } (Q_2(Z_q^2), Q_3(Z_q^2)))\\
& \quad + \sum_{x=2^{q-1}}^{2^{q-1}} 1) \quad \text{(cuts on connecting edge } (Q_3(Z_q^2), Q_4(Z_q^2)))\\
& +(\sum_{x=2^{q-1}+1}^{2^q} \bar{n}(Q_2(Z_q^2), G(2^{q-1}+1,1,x,2^{q-1})) \quad \text{(cuts within } Q_2(Z_q^2))\\
& \quad + \sum_{x=2^{q-1}+1}^{2^q} \bar{n}(Q_4(Z_q^2), G(2^{q-1}+1,2^{q-1}+1,x,2^q))\\
& \qquad \text{(cuts within } Q_4(Z_q^2))\\
& \quad + \sum_{x=2^{q-1}+1}^{2^q-1} 1) \quad \text{(cuts on connecting edges } (Q_2(Z_q^2), Q_3(Z_q^2)))\\
= \;& (\sum_{x=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,2^{q-1}))) \quad (Q_1(Z_q^2)\text{: a conanical } Z_{q-1}^2)\\
& +(\sum_{x=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,2^{q-1})) \quad (Q_3(Z_q^2)\text{: a conanical } Z_{q-1}^2)\\
& \quad + 1 + 2^{q-1} + 1)\\
& +(\sum_{x=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,2^{q-1}))) \quad (Q_2(Z_q^2)\text{: a conanical } Z_{q-1}^2)\\
& +(\sum_{x=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,2^{q-1})) \quad (Q_4(Z_q^2)\text{: a conanical } Z_{q-1}^2)\\
& \quad + 2^{q-1} - 1)\\
= \;& \overline{\Pi}_{q-1}^v + (\overline{\Pi}_{q-1}^v + 2^{q-1} + 2) + \overline{\Pi}_{q-1}^v + (\overline{\Pi}_{q-1}^v + 2^{q-1} - 1) = 4 \cdot \overline{\Pi}_{q-1}^v + 2^q + 1.
\end{aligned}
$$

The proof of $\overline{\Pi}_q^h$ is similar to that of $\overline{\Pi}_q^v$. ∎

The closed-form solutions for $\overline{\Pi}_q$ are employed to establish a system of recurrences for $\overline{\Omega}^c_{q,2^q}$.

Lemma 5.7 *For a canonical* Z_q^2,

$$\overline{\Omega}^{c_1}_{q,2^q} = \begin{cases} 4\overline{\Omega}^{c_1}_{q-1,2^{q-1}} + 2^q \cdot \overline{\Pi}^v_{q-1} + 2^q \cdot \overline{\Pi}^h_{q-1} + 2^{2q} - 2^q + 3 & \text{if } q > 1, \\ 5 & \text{if } q = 1; \end{cases}$$

$$\overline{\Omega}^{c_2}_{q,2^q} = \begin{cases} 4\overline{\Omega}^{c_2}_{q-1,2^{q-1}} + 2^q \cdot \overline{\Pi}^v_{q-1} + 2^q \cdot \overline{\Pi}^h_{q-1} + 2^{2q} + 2^q & \text{if } q > 1, \\ 6 & \text{if } q = 1. \end{cases}$$

Proof. As in Figure 5.8 and the case discussion for $\Omega^{c_1}_{q,2^q}$, we can split $\Omega^{c_1}_{q,2^q}$ into four parts:

$$\begin{aligned}
\overline{\Omega}^{c_1}_{q,2^q} &= \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \bar{n}(Z_q^2, G(1,1,x,y)) \\
&= \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} \bar{n}(Z_q^2, G(1,1,x,y)) \quad \text{(Figure 5.8(a))} \\
&\quad + \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} \bar{n}(Z_q^2, G(1,1,x,y)) \quad \text{(Figure 5.8(b))} \\
&\quad + \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Z_q^2, G(1,1,x,y)) \quad \text{(Figure 5.8(c))} \\
&\quad + \sum_{x=1}^{2^{q-1}}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Z_q^2, G(1,1,x,y)) \quad \text{(Figure 5.8(d))} \\
&= \Big(\sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} \bar{n}(Q_1(Z_q^2), G(1,1,x,y)) \quad \text{(cuts within } Q_1(Z_q^2)) \\
&\quad\quad + \sum_{x=2^{q-1}}^{2^{q-1}}\sum_{y=2^{q-1}}^{2^{q-1}} 1\Big) \quad \text{(cut on connecting edge } (Q_1(Z_q^2), Q_2(Z_q^2))) \\
&\quad + \Big(\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} \bar{n}(Q_2(Z_q^2), G(1,1,x,y)) \quad \text{(cuts within } Q_2(Z_q^2)) \\
&\quad\quad + \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} \bar{n}(Q_1(Z_q^2), G(1,1,2^{q-1},y)) \quad \text{(cuts within } Q_1(Z_q^2) \\
&\quad\quad\quad \text{by vertical side of } G) \\
&\quad\quad + \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}-1} 1 \quad \text{(cuts on connecting edge } (Q_1(Z_q^2), Q_2(Z_q^2)))
\end{aligned}$$

$$+\sum_{x=2^q}^{2^q}\sum_{y=2^{q-1}}^{2^{q-1}} 1) \quad \text{(cut on connecting edge } (Q_2(Z_q^2), Q_3(Z_q^2)))$$

$$+(\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_4(Z_q^2), G(1,1,x,y)) \quad \text{(cuts within } Q_4(Z_q^2))$$

$$+\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_2(Z_q^2), G(1,1,x,2^{q-1})) \quad \text{(cuts within } Q_2(Z_q^2) \text{ by horizontal side of } G)$$

$$+\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_3(Z_q^2), G(1,1,2^{q-1},y)) \quad \text{(cuts within } Q_3(Z_q^2) \text{ by vertical side of } G)$$

$$+\sum_{x=2^{q-1}+1}^{2^q-1}\sum_{y=2^{q-1}+1}^{2^q} 1 \quad \text{(cuts on connecting edge } (Q_2(Z_q^2), Q_3(Z_q^2)))$$

$$+\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q-1} 1) \quad \text{(cuts on connecting edge } (Q_3(Z_q^2), Q_4(Z_q^2)))$$

$$+(\sum_{x=1}^{2^{q-1}}\sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Q_3(Z_q^2), G(1,1,x,y)) \quad \text{(cuts within } Q_3(Z_q^2))$$

$$+\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} \bar{n}(Q_1(Z_q^2), G(1,1,x,2^{q-1})) \quad \text{(cuts within } Q_1(Z_q^2) \text{ by horizontal side of } G)$$

$$+\sum_{x=2^{q-1}}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} 1 \quad \text{(cuts on connecting edge } (Q_1(Z_q^2), Q_2(Z_q^2)))$$

$$+\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} 1 \quad \text{(cuts on connecting edge } (Q_2(Z_q^2), Q_3(Z_q^2)))$$

$$+\sum_{x=2^{q-1}}^{2^{q-1}}\sum_{y=2^q}^{2^q} 1) \quad \text{(cut on connecting edge } (Q_3(Z_q^2), Q_4(Z_q^2)))$$

$$= (\sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,y)) \quad (Q_1(Z_q^2)\text{: a canonical } Z_{q-1}^2)$$
$$+1)$$

$$+(\sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,y)) \quad (Q_2(Z_q^2)\text{: a canonical } Z_{q-1}^2)$$

$$+ (2^{q-1}) \sum_{y=1}^{2^{q-1}} \bar{n}(Z_q^2, G(1,1,2^{q-1},y)) \qquad (Q_1(Z_q^2)\text{: a canonical } Z_{q-1}^2)$$
$$+ (2^{q-1}(2^{q-1}-1))$$
$$+ 1)$$
$$+(\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,y))(Q_4(Z_q^2)\text{: a canonical } Z_{q-1}^2)$$
$$+ (2^{q-1}) \sum_{x=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,2^{q-1})) \qquad (Q_2(Z_q^2)\text{: a canonical } Z_{q-1}^2)$$
$$+ (2^{q-1}) \sum_{y=2^{q-1}+1}^{2^q} \bar{n}(Z_{q-1}^2, G(1,1,2^{q-1},y)) \qquad (Q_3(Z_q^2)\text{: a canonical } Z_{q-1}^2)$$
$$+ (2^{q-1}(2^{q-1}-1))$$
$$+ (2^{q-1}(2^{q-1}-1)))$$
$$+(\sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,y)) \qquad (Q_3(Z_q^2)\text{: a canonical } Z_{q-1}^2)$$
$$+ \sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \bar{n}(Z_{q-1}^2, G(1,1,x,2^{q-1})) \qquad (Q_1(Z_q^2)\text{: a canonical } Z_{q-1}^2)$$
$$+ 2^{q-1} + (2^{q-1})^2 + 1)$$

$$\begin{aligned}
= \;& (\overline{\Omega}_{q-1,2^{q-1}}^{c_1} + 1) \\
& +(\overline{\Omega}_{q-1,2^{q-1}}^{c_1} + 2^{q-1} \cdot \overline{\Pi}_{q-1}^{h} + 2^{q-1}(2^{q-1}-1) + 1) \\
& +(\overline{\Omega}_{q-1,2^{q-1}}^{c_1} + 2^{q-1} \cdot \overline{\Pi}_{q-1}^{v} + 2^{q-1} \cdot \overline{\Pi}_{q-1}^{h} + 2^{q-1}(2^{q-1}-1) + 2^{q-1}(2^{q-1}-1)) \\
& +(\overline{\Omega}_{q-1,2^{q-1}}^{c_1} + 2^{q-1} \cdot \overline{\Pi}_{q-1}^{v} + 2^{q-1} + (2^{q-1})^2 + 1) \\
= \;& 4\overline{\Omega}_{q-1,2^{q-1}}^{c_1} + 2^q \cdot \overline{\Pi}_{q-1}^{v} + 2^q \cdot \overline{\Pi}_{q-1}^{h} + 2^{2q} - 2^q + 3.
\end{aligned}$$

The proof for $\overline{\Omega}_{q,2^q}^{c_2}$ is similar to this one. ∎

The closed-form solutions for $\overline{\Omega}_{q,2^q}^{c}$ and $\overline{\Pi}_q$ are employed to obtain exact formulas for $\Omega_{k,2^q}^{c}$.

Lemma 5.8 *For a canonical Z_k^2 structured as an Z_{k-q}^2-curve interconnecting $2^{2(k-q)}$*

Z_q^2*-subcurves,*

$$\Omega^{c1}_{k,2^q} = \overline{\Omega}^{c1}_{q,2^q} - \overline{\Pi}^h_q - \overline{\Pi}^v_q,$$
$$\Omega^{c2}_{k,2^q} = \overline{\Omega}^{c2}_{q,2^q} - \overline{\Pi}^h_q - \overline{\Pi}^v_q.$$

Proof. By the definition, we have

$$\begin{aligned}
\Omega^{c1}_{k,2^q} &= \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} \bar{n}(Z_k^2, G(1,1,x,y)) \\
&= \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q} \bar{n}(Z_k^2, G(1,1,x,y)) - \sum_{x=1}^{2^q-1}\sum_{y=2^q}^{2^q} \bar{n}(Z_k^2, G(1,1,x,y)) \\
&= (\sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \bar{n}(Z_k^2, G(1,1,x,y)) - \sum_{x=2^q}^{2^q}\sum_{y=1}^{2^q} \bar{n}(Z_k^2, G(1,1,x,y))) \\
&\quad -(\sum_{x=1}^{2^q}\sum_{y=2^q}^{2^q} \bar{n}(Z_k^2, G(1,1,x,y)) - \sum_{x=2^q}^{2^q}\sum_{y=2^q}^{2^q} \bar{n}(Z_k^2, G(1,1,x,y))) \\
&= \overline{\Omega}^{c1}_{q,2^q} - \overline{\Pi}^h_q - \overline{\Pi}^v_q.
\end{aligned}$$

The proof of $\Omega^{c2}_{k,2^q}$ is similar to this one. ▮

The exact formulas for $\Omega^{c}_{k,2^q}$ are employed to establish a system of recurrences for $\Omega_{k,2^q}$.

Lemma 5.9 *For a canonical Z_k^2 structured as an Z_{k-q}^2-curve interconnecting $2^{2(k-q)}$ Z_q^2-subcurves,*

$$\Omega^{L}_{k,2^q} = \begin{cases} 2\Omega^{L}_{k-1,2^q} + \Omega^{c1}_{k-1,2^q} + \Omega^{c2}_{k-1,2^q} + (2^q-1)^2 + (2^q-1) & \text{if } k > q, \\ \overline{\Pi}^h_q & \text{if } k = q; \end{cases}$$
$$\Omega^{B}_{k,2^q} = \begin{cases} 2\Omega^{B}_{k-1,2^q} + \Omega^{c1}_{k-1,2^q} + \Omega^{c2}_{k-1,2^q} + (2^q-1)^2 + (2^q-1) & \text{if } k > q, \\ \overline{\Pi}^v_q & \text{if } k = q. \end{cases}$$

Proof. Similar to the proof of Lemma 5.8, from the definition, we have (see Figure 5.6)

$$\Omega^{L}_{k,2^q} = \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \bar{n}(Z_k^2, G(x,1,x+2^q-1,y))$$

$$
\begin{aligned}
= & \sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Z_k^2, G(x,1,x+2^q-1,y)) \quad (\text{in } Q_1(Z_k^2)) \\
& + \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \bar{n}(Z_k^2, G(x,1,x+2^q-1,y)) \quad (\text{across } Q_1(Z_k^2), Q_2(Z_k^2)) \\
& + \sum_{x=2^{k-1}+1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Z_k^2, G(x,1,x+2^q-1,y)) \quad (\text{in } Q_2(Z_k^2)) \\
= & \, (\sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Q_1(Z_k^2), G(x,1,x+2^q-1,y))) \quad (\text{cuts within } Q_1(Z_k^2)) \\
& +(\sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \bar{n}(Q_1(Z_k^2), G(x,1,2^{k-1},y)) \quad (\text{zooming in } Q_1(Z_k^2)) \\
& \quad + \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \bar{n}(Q_2(Z_k^2), G(2^{k-1}+1,1,x+2^q-1,y)) \\
& \qquad (\text{zooming in } Q_2(Z_k^2)) \\
& \quad + \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} 1) \quad (\text{cuts on connecting edge } (Q_1(Z_k^2), Q_2(Z_k^2))) \\
& +(\sum_{x=2^{k-1}+1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Q_2(Z_k^2), G(x,1,x+2^q-1,y)) \quad (\text{cuts within } Q_2(Z_k^2)) \\
& \quad \sum_{x=2^{k-1}+1}^{2^{k-1}+1} \sum_{y=1}^{2^{q-1}-1} 1) \quad (\text{cuts on connecting edge } (Q_1(Z_k^2), Q_2(Z_k^2))) \\
= & \, (\sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Z_{k-1}^2, G(x,1,x+2^q-1,x))) \quad (Q_1(Z_k^2)\text{: a canonical } Z_{k-1}^2) \\
& +(\sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \bar{n}(Z_{k-1}^2, G(x,1,2^{k-1},y)) \quad (Q_1(Z_k^2)\text{: a canonical } Z_{k-1}^2) \\
& \quad + \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} \bar{n}(Z_{k-1}^2, G(1,1,x,y)) \quad (Q_1(Z_k^2)\text{: a canonical } Z_{k-1}^2) \\
& \quad + (2^q-1)^2) \\
& (\sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \bar{n}(Z_{k-1}^2, G(x,1,x+2^q-1,y)) \quad (Q_2(Z_k^2)\text{: a canonical } Z_{k-1}^2) \\
& \quad + (2^q-1))
\end{aligned}
$$

$$= \Omega^{L}_{k-1,2^q}$$
$$+(\Omega^{c1}_{k-1,2^q} + \Omega^{c2}_{k-1,2^q} + (2^q - 1)^2)$$
$$+(\Omega^{L}_{k-1,2^q} + (2^q - 1))$$
$$= 2\Omega^{L}_{k-1,2^q} + \Omega^{c1}_{k-1,2^q} + \Omega^{c2}_{k-1,2^q} + (2^q - 1)^2 + (2^q - 1).$$

For $\Omega^{B}_{k-1,2^q}$, the proof is similar to this one. ∎

We obtain the closed-form solutions for $\Omega_{k,2^q}$ by using the mathematical software Maple.

5.3.2 Query Subgrids Overlapping with All Quadrants

For a $2^q \times 2^q$ query subgrid G that overlaps four quadrants around the center of Z_k^2, when zooming in on the incomplete rectangular subgrid $G \cap G_1$, where G_1 denotes the subspace of $Q_1(Z_k^2)$ (with both side-lengths at most $2^q - 1$), we reduce $\sum_{\text{all } G \cap G_1} \bar{n}(Z_k^2, G \cap G_1)$ to $\Omega^{c4}_{k-1,2^q}(= \Omega^{c1}_{k-1,2^q})$. Similar consideration leads to a reduction of $\sum_{\text{all } G \cap G'} \bar{n}(Z_k^2, G \cap G')$ to $\Omega^{c3}_{k-1,2^q}(= \Omega^{c2}_{k-1,2^q})$, $\Omega^{c2}_{k-1,2^q}$ and $\Omega^{c1}_{k-1,2^q}$ when $G \cap G'$ denotes the subspace for G overlapping $Q_2(Z_k^2)$, $Q_3(Z_k^2)$, or $Q_3(Z_k^2)$, respectively.

Thus, the summation of numbers of edge cuts for all identically shaped $2^q \times 2^q$ query subgrids G that overlap all four quadrants is

$$2\Omega^{c2}_{k-1,2^q} + 2\Omega^{c1}_{k-1,2^q}.$$

5.3.3 The Big Picture: Computing $E_q(Z_k^2)$

The results in the previous three subsections yield $\varepsilon_{k,q}(Z_k^2)$. Hence, we have the following lemma for $E_q(Z_k^2)$:

Lemma 5.10 *For a canonical Z_k^2, the recurrence for total number of cuts on edges* *by all identically shaped $2^q \times 2^q$ subgrids G:*

$$E_q(Z_k^2) = \begin{cases} 4E_q(Z_{k-1}^2) + 2(\Omega_{k-1,2^q}^B + \Omega_{k-1,2^q}^B + 2(2^q-1)) \\ \quad + 2(\Omega_{k-1,2^q}^L + \Omega_{k-1,2^q}^L + 2(2^q-1)) \\ \quad + (2\Omega_{k-1,2^q}^{c1} + 2\Omega_{k-1,2^q}^{c2} + 2(2^q-1)^2) & \text{if } k > q+1, \\ \\ 4E_q(Z_{k-1}^2) + 2(\Omega_{k-1,2^q}^B + \Omega_{k-1,2^q}^B) \\ \quad + 2(\Omega_{k-1,2^q}^L + \Omega_{k-1,2^q}^L + 2(2^q-1)) \\ \quad + (2\Omega_{k-1,2^q}^{c1} + 2\Omega_{k-1,2^q}^{c2} + 2(2^q-1)^2) & \text{if } k = q+1, \\ \\ 2 & \text{if } k = q. \end{cases}$$

Proof. For $k > q+1$, the proof is similar to the proof of Lemma 5.5:

Case 1: G overlaps with exactly $Q_1(Z_k^2)$ and $Q_2(Z_k^2)$. This part is reduced to $\Omega_{k-1,2^q}^T (= \Omega_{k-1,2^q}^B)$ (cuts on $Q_1(Z_k^2)$), $\Omega_{k-1,2^q}^B$ (cuts on $Q_2(Z_k^2)$), $2(2^q-1)$ cuts on the connecting edge $(Q_1(Z_k^2), Q_2(Z_k^2))$ when subgrids G align left-most and right-most side of these two quadrants.

Case 2: G overlaps with exactly $Q_2(Z_k^2)$ and $Q_4(Z_k^2)$. This part is reduced to $\Omega_{k-1,2^q}^R (= \Omega_{k-1,2^q}^L)$ (cuts on $Q_2(Z_k^2)$), $\Omega_{k-1,2^q}^L$ (cuts on $Q_4(Z_k^2)$), (2^q-1) cuts on the connecting edge $(Q_2(Z_k^2), Q_3(Z_k^2))$ when subgrids G align top-most side of these two quadrants, and (2^q-1) cuts on the connecting edge $(Q_3(Z_k^2), Q_4(Z_k^2))$ when subgrids G align bottom-most side of these two quadrants.

Case 3: G overlaps with exactly $Q_3(Z_k^2)$ and $Q_4(Z_k^2)$. This part is reduced to $\Omega_{k-1,2^q}^T (= \Omega_{k-1,2^q}^B)$ (cuts on $Q_3(Z_k^2)$), $\Omega_{k-1,2^q}^B$ (cuts on $Q_4(Z_k^2)$), and $2(2^q-1)$ cuts on the connecting edge $(Q_3(Z_k^2), Q_4(Z_k^2))$ when subgrids G align left-most and right-most side of these two quadrants.

Case 4: G overlaps with exactly $Q_1(Z_k^2)$ and $Q_3(Z_k^2)$. This part is reduced to $\Omega_{k-1,2^q}^R (= \Omega_{k-1,2^q}^L)$ (cuts on $Q_1(Z_k^2)$), $\Omega_{k-1,2^q}^L$ (cuts on $Q_3(Z_k^2)$), (2^q-1) cuts on the connecting edge $(Q_2(Z_k^2), Q_3(Z_k^2))$ when subgrids G align bottom-most side of these two quadrants, and (2^q-1) cuts on the connecting edge $(Q_1(Z_k^2), Q_2(Z_k^2))$ when subgrids G align top-most side of these two quadrants.

Case 5: G overlaps with exactly all four quadrants. This part is reduced to $\Omega_{k-1,2^q}^{c4} (= \Omega_{k-1,2^q}^{c1})$ (cuts on $Q_1(Z_k^2)$), $\Omega_{k-1,2^q}^{c3} (= \Omega_{k-1,2^q}^{c2})$ (cuts on $Q_2(Z_k^2)$), $\Omega_{k-1,2^q}^{c2}$ (cuts on $Q_3(Z_k^2)$), $\Omega_{k-1,2^q}^{c1}$ (cuts on $Q_4(Z_k^2)$), and $2(2^q-1)^2$ cuts on the connecting edges $(Q_1(Z_k^2), Q_2(Z_k^2))$ and $(Q_3(Z_k^2), Q_4(Z_k^2))$.

Combining all the five cases, complete the recurrence for $k > q+1$.

For $k = q+1$, there are no cuts on connecting edges in Cases 1 and 3 because the connecting edges are inside of all the subspaces that are across exactly two quadrants (first and second, or third and fourth quadrants).

For the case of $k = q$, there are two cuts that are the edge cut between entry grid point and other curve, and the edge cut between exit grid point and other curve. ∎

The exact formula for $E_q(Z_k^2)$ is:

$$\begin{aligned} E_q(Z_k^2) &= 2^{2k+q+2} - 2^{2k+2} + 3 \cdot 2^{2k-q} - 2^{2k-2q} - 2^{k+2q+3} + 3 \cdot 2^{k+q+2} - 2^{k+3} \\ &\quad +2^{k-q+2} + 2^{3q+2} - 2^{2q+3} + 2^{q+2}. \end{aligned}$$

The summation of all numbers of clusters over all identically shaped $2^q \times 2^q$ query subgrids of an Z_k^2-structural grid space $[2^k]^2$ is

$$\mathcal{S}_{k,2^q}(Z_k^2) = \frac{E_q(Z_k^2)}{2}.$$

The mean number of cluster within a subspace $2^q \times 2^q$ is

$$\frac{\mathcal{S}_{k,2^q}(Z_k^2)}{(2^k - 2^q + 1)^2} = \frac{E_q(Z_k^2)}{2(2^k - 2^q + 1)^2}.$$

Thus, the exact formula for the mean number of cluster within a subspace $2^q \times 2^q$ for Z_k^2 is corollarily derived.

Theorem 5.2 *The mean number of cluster over all identical subspaces $2^q \times 2^q$ for Z_k^2 is*

$$\begin{aligned} &(2^{2k+q+2} - 2^{2k+2} + 3 \cdot 2^{2k-q} + 2^{2k-2q} - 2^{k+2q+3} + 3 \cdot 2^{k+q+2} + 2^{k+3} \\ &+2^{k-q+2} + 2^{3q+2} + 2^{2q+3} + 2^{q+2})/(2(2^k - 2^q + 1)^2). \end{aligned}$$

5.4 Comparisons and Validation

For a space-filling curve C_k indexing the grid space $[2^k]^2$, denote by $\mathcal{S}_{k,q}(C_k)$ the mean number of clusters over all $2^q \times 2^q$ subgrids of the C_k-structural grid space.

The exact formulas for $E_q(H_k^2)$ and $E_q(Z_k^2)$ give the exact formulas for $\mathcal{S}_{k,2^q}(H_k^2)$ and $\mathcal{S}_{k,2^q}(Z_k^2)$. We simplify the exact results asymptotically as follows. For sufficiently large k and q with $k \gg q$ (typical scenario for range queries),

$$\frac{\mathcal{S}_{k,2^q}(Z_k^2)}{\mathcal{S}_{k,2^q}(H_k^2)} = \frac{E_q(Z_k^2)}{E_q(H_k^2)} \approx 2$$

With respect to the $\mathcal{S}_{k,q}$-statistics, the Hilbert curve family clearly performs better than the z-order curve family over the considered ranges for k and q.

5.5 Summary

In this chapter, the analytical study of the clustering performances of 2-dimensional order-k Hilbert and z-order curve families are based upon the clustering statistics $\mathcal{S}_{k,2^q}$ — mean number of clusters over all $2^q \times 2^q$ identically shaped subgrids, respectively. By taking advantage of self-similar properties of Hilbert and z-order curve, we derive their exact formulas for $\mathcal{S}_{k,2^q}$. The results are same as those derived by Moon, Jagadish, Faloutsos, and Saltz [MJFS01]. The exact results allow us to compare their relative performances with respect to this measure. For sufficiently large k and q with $k \gg q$, Hilbert curve family performs significantly better than z-order curve family with respect to $\mathcal{S}_{k,2^q}$.

CHAPTER VI

MEASURE BY MEAN INTER-CLUSTER DISTANCE

In addition to considering the number of clusters to optimize a range query over space-filling curves in multi-dimensional databases, the mean inter-cluster distance within a subspace is adopted as another measure of performances for space-filling indexing methods. This idea is stemmed from Asano, Ranjan, Roos, Welzl, and Widmayer [ARR+97]. They consider the problem of minimizing the number of seek operations, mainly because there is a tradeoff between seek time to proper block (cluster) and latency/transfer time for unnecessary blocks (inter-cluster gap). Thus, good bounds on the inter-clustering statistics can translate into good bounds on the average tolerance of unnecessary block transfers.

Similar to Chapter V, we propose a measure by the mean inter-cluster distance within a subspace for space-filling curves and use it to compare Hilbert and z-order curve families.

6.1 Approach

Note that there are two statistics for the mean inter-cluster distance within a subspace: one is the mean inter-cluster distance over all inter-cluster gaps, the other one is the mean total inter-cluster distance over all identically shaped subgrids. The derivations for both statistics involve the computation of the total inter-cluster distance over the identically shaped subspaces and the computations of calculates the number of inter-cluster gaps and the number of subspaces.

Within a subspace G, we denote the first entry (lowest-indexed) and last exit (highest-indexed) grid points by $\theta_1(G)$ and $\theta_2(G)$, the number of clusters and the number of grid points inside of G by $\bar{n}(G)$ and $|G|$, respectively. The following

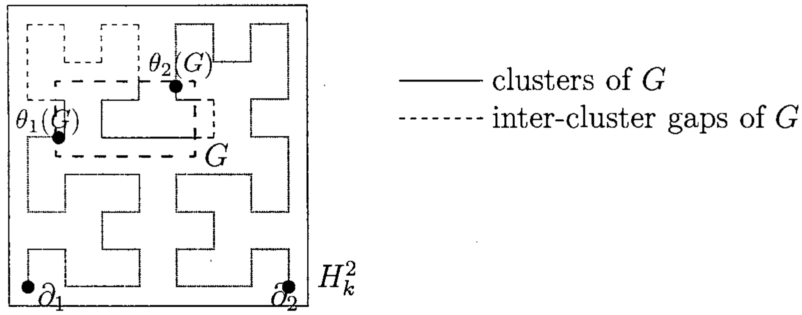


Figure 6.1: An example query subgrid G inducing its first entrance $\theta_1(G)$ and last exit $\theta_2(G)$, and interleaving clusters and inter-cluster gaps.

observations helps the derivations of mean inter-cluster distance:

1. Sum of inter-cluster distances within a subspace G equals to $\theta_2(G) - \theta_1(G) - |G| + 1$.
2. For a subspace G overlapping with more than one quadrant, $\theta_1(G)$ must be in the lowest numbered quadrant, and $\theta_2(G)$ must be in the highest numbered quadrant.
3. Number of inter-cluster gaps within a subspace G equals to $\bar{n}(G) - 1$ (see Figure 6.1).

With the above observations, we can translate the computation of inter-cluster distances into the computation of the index cumulations of $\theta_1(G)$ and $\theta_2(G)$, respectively, for all identically shaped subspaces G.

The index-cumulation for $\theta_2(G)$ for all identically shaped subspace G is based on the construction framework by recursion:

1. Decompose the index-cumulation into index-cumulations within the four quadrants and index-cumulation for subspaces G across different quadrants.
2. Categorize the $\theta_2(G)$ of a subspace across quadrants into: $\theta_2(G)$ in four corner boundaries and four side boundaries that are inter-recurrence related.

Similarly, we can derive the index-cumulation for all $\theta_1(G)$.

6.2 Analytical Study of Inter-Clustering Performances for Hilbert Curve

With respect to the canonical orientation of H_k^2 shown in Figure 2.6(a), we cover the 2-dimensional k-order grid with 2^k rows $(R_{k,1}, R_{k,2}, \ldots, R_{k,2^k})$, indexed from the bottom, and 2^k columns $(C_{k,1}, C_{k,2}, \ldots, C_{k,2^k})$, indexed from the left. We denote:

1. For a grid point $v \in [2^k]^2$, its x- and y-coordinate by $X(v)$ and $Y(v)$, respectively (that is, v is the intersection grid point of the row $R_{k,X(v)}$ and the column $C_{k,Y(v)}$,

2. For the grid points $v, v' \in [2^k]^2$, their index-difference by $\hbar_k(v, v')$ $(= |(H_k^2)^{-1}(v) - (H_k^2)^{-1}(v')|)$, and

3. For a rectangular query subgrid with its lower-left corner at grid point (x, y) and upper-right corner at grid point (x', y') $(1 \le x \le x' \le 2^k$ and $1 \le y \le y' \le 2^k)$ covering $\cup_{\alpha=x}^{x'} R_{k,\alpha} \cap \cup_{\beta=y}^{y'} C_{k,\beta}$, its set of grid points by $G_k(x, y, x', y')$ $(= \{v \in [2^k]^2 \mid x \le X(v) \le x'$ and $y \le Y(v) \le y'\})$. The size of the query subgrid $G_k(x, y, x', y')$ is $(x' - x + 1) \times (y' - y + 1)$.

As suggested in Remark 5.1, we study the inter-clustering performances based upon query subgrids of size $2^q \times 2^q$.

For a 2-dimensional order-k Hilbert curve H_k^2, let $\Psi_{k,q}(H_k^2)$ denote the summation of all inter-cluster distances over all $2^q \times 2^q$ query subgrids of an H_k^2-structural grid space $[2^k]^2$. For a subgrid G, let $\theta_1(G)$ denote the first entrance (the lowest H_k^2-indexed grid point) into G and $\theta_2(G)$ denote the last exit (the highest H_k^2-indexed grid point) out of G. Figure 6.1 illustrates an example query subgrid and its induced first entrance and last exit, and interleaving clusters and inter-cluster gaps.

Remark 6.1 *Within a query subgrid G (with $|G|$ grid points), the summation of all its inter-cluster distances is $\hbar_k(\theta_1(G), \theta_2(G)) - |G| + 1$. In developing the supporting lemmas, we express $\hbar_k(\theta_1(G), \theta_2(G))$ as $\hbar_k(\theta_2(G), v) - \hbar_k(\theta_1(G), v)$ for a suitably chosen grid point v.*

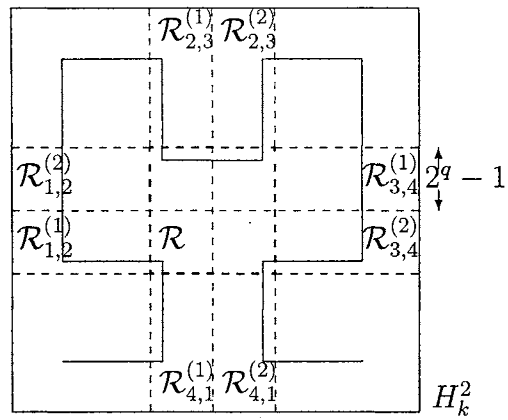


Figure 6.2: The boundary regions of neighboring quadrants are organized into nine disjoint regions: $\mathcal{R}^{(1)}_{i,i \bmod 4+1}$, $\mathcal{R}^{(2)}_{i,i \bmod 4+1}$ for $i = 1,2,3,4$, and $\mathcal{R}$.

Thus, Remark 6.1 reduces the computation of the summation of all inter-cluster distances over all identically shaped subgrids G to the computations of $\sum_{\text{all } G} \hbar_k(\theta_j(G), v)$ for $j = 1,2$ and a suitably chosen v.

The recursive decomposition of H_k^2 (see Figure 2.6(b)) gives that

$$\Psi_{k,q}(H_k^2) = 4\Psi_{k,q}(H_{k-1}^2) + \varepsilon_{k,q}(H_k^2),$$

where $\varepsilon_{k,q}(H_k^2)$ denotes the summation of all inter-cluster distances over all $2^q \times 2^q$ query subgrids, each of which overlaps with more than one quadrant (that is, two or four). These query subgrids are contained in the boundary regions of neighboring quadrants, which can be organized into nine disjoint regions: $\mathcal{R}^{(1)}_{i,i \bmod 4+1}$, $\mathcal{R}^{(2)}_{i,i \bmod 4+1}$ for $i = 1,2,3,4$, and R, as shown in Figure 6.2.

Remark 6.2 *For a query subgrid G overlapping with more than one quadrant, $\theta_1(G)$ is in the lowest-numbered quadrant, and $\theta_2(G)$ is in the highest-numbered quadrant.* For a $2^q \times 2^q$ query subgrid G, G overlaps with:

1. Exactly $Q_i(H_k^2)$ and $Q_{i \bmod 4+1}(H_k^2)$ if and only if $G \subseteq \mathcal{R}^{(1)}_{i,i \bmod 4+1} \cup \mathcal{R}^{(2)}_{i,i \bmod 4+1}$ for every $i \in \{1,2,3,4\}$. In this case, $\theta_\eta(G) \in \mathcal{R}^{(\eta)}_{i,i \bmod 4+1}$ for $\eta \in \{1,2\}$ by Remark 6.2.

2. $Q_i(H_k^2)$ for all $i \in \{1,2,3,4\}$ if and only if $G \subseteq \mathcal{R}$. In this case, $\theta_1(G) \in Q_1(H_k^2)$

(upper-right corner) and $\theta_2(G) \in Q_4(H_k^2)$ (upper-left corner) by Remark 6.2.

We divide the computation of $\varepsilon_{k,q}(H_k^2)$ into three parts:

1. $\sum \hbar_k(\theta_2(G), \partial_1(H_k^2))$ over all $2^q \times 2^q$ query subgrids $G \subseteq \mathcal{R}^{(1)}_{i,i \bmod 4+1} \cup \mathcal{R}^{(2)}_{i,i \bmod 4+1}$ for $i \in \{1,2,3,4\}$,

2. $\sum \hbar_k(\theta_1(G), \partial_1(H_k^2))$ over all $2^q \times 2^q$ query subgrids $G \subseteq \mathcal{R}^{(1)}_{i,i \bmod 4+1} \cup \mathcal{R}^{(2)}_{i,i \bmod 4+1}$ for $i \in \{1,2,3,4\}$, and

3. the summation of all inter-cluster distances over all $2^q \times 2^q$ query subgrids contained in $\mathcal{R}$.

We develop combinatorial lemmas in the following three subsections to support the computations.

6.2.1 $\sum \hbar_k(\theta_2(G), \partial_1(H_k^2))$ over Subgrids G Overlapping with Two Quadrants

Consider an arbitrary $2^q \times 2^q$ query subgrid $G \subseteq \mathcal{R}^{(1)}_{i,i \bmod 4+1} \cup \mathcal{R}^{(2)}_{i,i \bmod 4+1}$ where $i \in \{1,2,3,4\}$. Remark 6.2 gives that $\theta_2(G) \in \mathcal{R}^{(2)}_{i,i \bmod 4+1}$, and we zoom in on the "incomplete" rectangular subgrid $G \cap \mathcal{R}^{(2)}_{i,i \bmod 4+1}$ (with one side-length at most $2^q - 1$). Observe that for $i = 1,2,3,4$, $\mathcal{R}^{(2)}_{i,i \bmod 4+1}$ aggregates the $2^q - 1$ bottom rows, leftmost columns, top rows, and leftmost columns of $Q_2(H_k^2)$, $Q_3(H_k^2)$, $Q_4(H_k^2)$, and $Q_4(H_k^2)$, respectively. Since the quadrants are isomorphic to a canonical H_{k-1}^2 via symmetry (reflection and rotation), we consider the following system of summations $\Omega_{k,2^q} = (\Omega^L_{k,2^q}, \Omega^R_{k,2^q}, \Omega^B_{k,2^q}, \Omega^T_{k,2^q})$ in a general context of a canonical H_k^2:

$$\Omega^L_{k,2^q} = \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)), \partial_1(H_k^2))$$

— for left boundary (see Figure 6.3(a)),

$$\Omega^R_{k,2^q} = \sum_{x=1}^{2^k-2^q+1} \sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_2(G_k(x,y,x+2^q-1,2^k)), \partial_1(H_k^2))$$

— for right boundary,

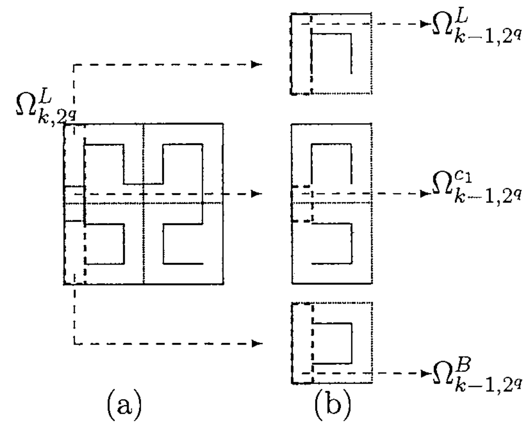


Figure 6.3: (a) $\Omega^L_{k,2^q}$ for a canonical H^2_k; (b) its recursive decomposition.

$$\Omega^B_{k,2^q} = \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^k-2^q+1} \hbar_k(\theta_2(G_k(1,y,x,y+2^q-1)), \partial_1(H^2_k))$$

— for bottom boundary,

$$\Omega^T_{k,2^q} = \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^k-2^q+1} \hbar_k(\theta_2(G_k(x,y,2^k,y+2^q-1)), \partial_1(H^2_k))$$

— for top boundary, and

$$\mathcal{N}^S_{k,2^q} = \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} 1$$

— for the number of incomplete rectangular subgrids in a boundary.

We will establish a system of recurrences (in k) for $\Omega_{k,2^q}$ (see Lemma 6.4 below). The system of recurrence involves another system of summations as prerequisites, as demonstrated in the following example. Consider a recursive decomposition of $\Omega^L_{k,2^q}$, illustrated in Figure 6.3(b), into four parts: (1) $\Omega^B_{k-1,2^q}$, (2) $\Omega^{c_1}_{k-1,2^q}$, (3) $\Omega^L_{k-1,2^q}$, and (4) adjustments for the previous three parts. The part $\Omega^{c_1}_{k-1,2^q}$ helps compute $\sum \hbar_k(\theta_2(G), \partial_1(H^2_k))$ over all incomplete rectangular subgrids G (with one side-length at most $2^q - 1$) overlapping both $Q_1(H^2_k)$ and $Q_2(H^2_k)$. According to Remark 6.2, the computation of this summation is reduced to $\sum \hbar_{k-1}(\theta_2(G), \partial_1(H^2_{k-1}))$ over all incomplete rectangular subgrids G (with both side-lengths at most $2^q - 1$) in the c_1-corner (lower-left corner) of a canonical H^2_{k-1} (that is, $Q_2(H^2_k)$). Each of the three parts $\Omega^B_{k-1,2^q}$, $\Omega^{c_1}_{k-1,2^q}$, and $\Omega^L_{k-1,2^q}$ is defined with respect to $\partial_1(H^2_{k-1})$ of a canonical

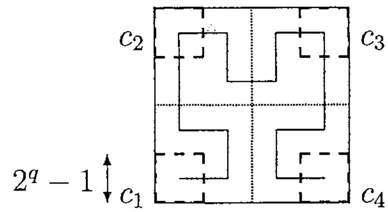


Figure 6.4: The four $(2^q-1)\times(2^q-1)$ corners of a canonical H_k^2.

H_{k-1}^2, we need to adjust each part with distance cumulation between the entry/exit of the underlying quadrant and $\partial_1(H_k^2)$.

The recursive decompositions of all four parts in $\Omega_{k,2^q}^L$, $\Omega_{k,2^q}^R$, $\Omega_{k,2^q}^B$, and $\Omega_{k,2^q}^T$ lead us to consider a prerequisite system of summations $\Omega_{k,2^q}^c = (\Omega_{k,2^q}^{c_1}, \Omega_{k,2^q}^{c_2}, \Omega_{k,2^q}^{c_3}, \Omega_{k,2^q}^{c_4})$ in a more general context of a canonical H_k^2 (see Figure 6.4):

$$\Omega_{k,2^q}^{c_1} = \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)),\partial_1(H_k^2)) \text{ — for lower-left corner,}$$

$$\Omega_{k,2^q}^{c_2} = \sum_{x=2^k-2^q+2}^{2^k}\ \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,2^k,y)),\partial_1(H_k^2)) \text{ — for upper-left corner,}$$

$$\Omega_{k,2^q}^{c_3} = \sum_{x=2^k-2^q+2}^{2^k}\ \sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_2(G_k(x,y,2^k,2^k)),\partial_1(H_k^2)) \text{ — for upper-right corner,}$$

$$\Omega_{k,2^q}^{c_4} = \sum_{x=1}^{2^q-1}\ \sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_2(G_k(1,y,x,2^k)),\partial_1(H_k^2)) \text{ — for lower-right corner, and}$$

$$\mathcal{N}_{k,2^q}^c = \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} 1 \text{ — for the number of incomplete rectangular subgrids in a corner.}$$

Note that in $\Omega_{k,2^q}^{c_4}$, $\theta_2(G_k(1,y,x,2^k)) = \partial_2(H_k^2)$ for all x and y in the summation-index ranges, hence $\Omega_{k,2^q}^{c_4} = (2^q-1)^2\hbar_k(\partial_2(H_k^2),\partial_1(H_k^2)) = (2^q-1)^2(2^{2k}-1)$. All other three summations involve rectangular subgrids contained in $(2^q-1)\times(2^q-1)$ corners. As suggested by Remark 5.1, we zoom in on the $2^q\times 2^q$ H_q^2-structural corners, and consider the following system of summations $\overline{\Omega}_{q,2^q}^c = (\overline{\Omega}_{q,2^q}^{c_1}, \overline{\Omega}_{q,2^q}^{c_2}, \overline{\Omega}_{q,2^q}^{c_3})$:

$$\overline{\Omega}_{q,2^q}^{c_1} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,y)),\partial_1(H_q^2)) \text{ — for lower-left corner,}$$

$$\overline{\Omega}_{q,2^q}^{c_2} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \hbar_q(\theta_2(G_q(x,1,2^q,y)),\partial_1(H_q^2)) \text{ — for upper-left corner,}$$

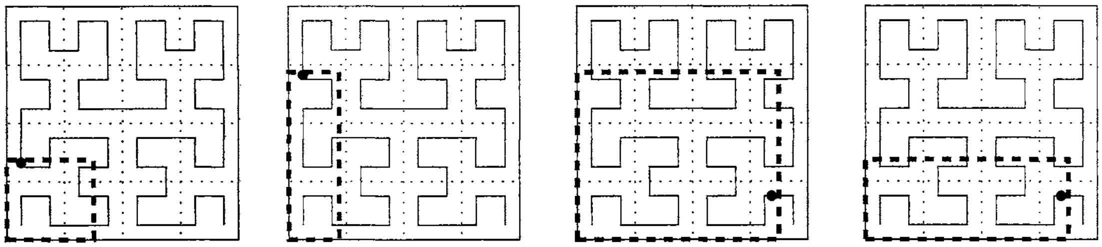


Figure 6.5: Four overlapping scenarios when decomposing $\overline{\Omega}^{c_1}_{q,2^q}$ in a canonical H^2_q: (a) contained in $Q_1(H^2_q)$; (b) and (d) overlapping with exactly two quadrants; (c) overlapping with all quadrants.

$$\overline{\Omega}^{c_3}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \hbar_q(\theta_2(G_q(x,y,2^q,2^q)),\partial_1(H^2_q)) \text{ — for upper-right corner, and}$$

$$\overline{\mathcal{N}}^{c}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} 1 \text{ — for the number of rectangular subgrids in a } 2^q \times 2^q \text{ corner.}$$

Thus far, we learn that the system of recurrences for $\Omega_{k,2^q}$ can be defined and solved via the prerequisite system $\Omega^c_{k,2^q}$, which is related to the system $\overline{\Omega}^{c}_{q,2^q}$ (see Lemma 6.3 below). The system $\overline{\Omega}^{c}_{q,2^q}$, which involves subgrids (with both side-lengths at most 2^q) of a canonical H^2_q, represents the basis of the recursive decompositions (in k to q) of $\Omega_{k,2^q}$ and $\Omega^c_{k,2^q}$. Similar to the reduction of $\Omega_{k,2^q}$ to $\Omega^c_{k,2^q}$, we develop a system of recurrences (in q) for $\overline{\Omega}^{c}_{q,2^q}$ via a prerequisite system, as demonstrated in the following example. Consider a recursive decomposition of $\overline{\Omega}^{c_1}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \hbar_k(\theta_2(G_q(1,1,x,y)),\partial_1(H^2_q))$ into four parts (together with adjustments), based upon the overlapping scenario of the rectangular subgrid $G_q(1,1,x,y)$ with the four quadrants of a canonical H^2_q (see Figure 6.5).

Case 1: $G_q(1,1,x,y)$ is contained in $Q_1(H^2_q)$ (see Figure 6.5(a)). This part is reduced to $\overline{\Omega}^{c_1}_{q-1,2^{q-1}}$ after $(-\frac{\pi}{2})$-rotating and then left-right reflecting $Q_1(H^2_q)$ into a canonical H^2_{q-1}.

Case 2: $G_q(1,1,x,y)$ overlaps with exactly $Q_1(H^2_q)$ and $Q_2(H^2_q)$ (see Figure 6.5(b)). This part is reduced to $\overline{\Omega}^{c_1}_{q-1,2^{q-1}}$ (with adjustment of distance cumulation).

Case 3: $G_q(1,1,x,y)$ overlaps with exactly $Q_1(H^2_q)$ and $Q_4(H^2_q)$ (see Figure 6.5(d)). This part is reduced to $\overline{\Omega}^{c_3}_{q-1,2^{q-1}}$ after $(+\frac{\pi}{2})$-rotating and then left-right reflecting

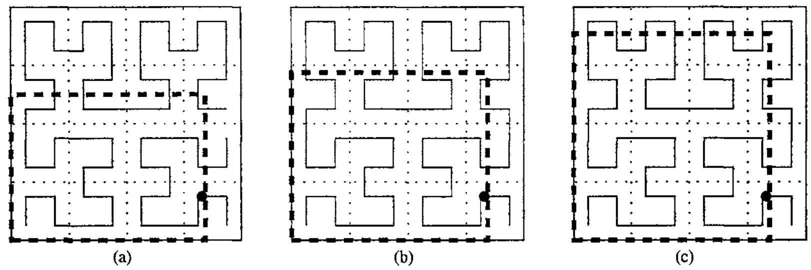


Figure 6.6: For subgrids overlapping with all quadrants of a canonical H_q^2, their last exits are the same.

$Q_4(H_q^2)$ into a canonical H_{q-1}^2 (with adjustment of distance cumulation).

Case 4: $G_q(1,1,x,y)$ overlaps with all quadrants (see Figure 6.5(c)). The overlapping condition gives that $x, y \in \{2^{q-1}+1, 2^{q-1}+2, \ldots, 2^q\}$. According to Remark 6.2, $\theta_2(G_q(1,1,x,y)) \in Q_4(H_q^2)$. Observe that, as shown in Figure 6.6, for every $y \in \{2^{q-1}+1, 2^{q-1}+2, \ldots, 2^q\}$, the subgrids $G_q(1,1,x,y)$ for all $x \in \{2^{q-1}+1, 2^{q-1}+2, \ldots, 2^q\}$ have the same $\theta_2(G_q(1,1,x,y))$ (independent of x).

The recursive decompositions of $\overline{\Omega}_{q,2^q}^{c_1}$, $\overline{\Omega}_{q,2^q}^{c_2}$, and $\overline{\Omega}_{q,2^q}^{c_3}$ lead us to consider a prerequisite system of summations $\overline{\Pi}_q = (\overline{\Pi}_q^T, \overline{\Pi}_q^L)$ in a general context of a canonical H_q^2:

$$\begin{aligned}
\overline{\Pi}_q^T &= \sum_{x=2^q}^{1} \hbar_q(\theta_2(G_q(x,1,2^q,2^q)), \partial_1(H_q^2))\text{—top to bottom incrementally, and} \\
\overline{\Pi}_q^L &= \sum_{y=1}^{2^q} \hbar_q(\theta_2(G_q(1,1,2^q,y)), \partial_1(H_q^2))\text{—left to right incrementally.}
\end{aligned}$$

We develop and solve a system of recurrences for $\overline{\Pi}_q$ and reverse the sequence of reductions to obtain the closed-form solutions for $\Omega_{k,2^q}$, which are summarized in the following four lemmas. Note that we present the systems of recurrences only (which are solved by a mathematical and analytical software such as Maple).

Lemma 6.1 *For a canonical H_q^2,*

$$\begin{aligned}
\overline{\Pi}_q^T &= \begin{cases} \overline{\Pi}_{q-1}^T + 2(2^{q-1})^3 + \overline{\Pi}_{q-1}^L + 3(2^{q-1})^3 & \text{if } q > 1 \\ 5 & \text{if } q = 1 \end{cases} \\
\overline{\Pi}_q^L &= \begin{cases} \overline{\Pi}_{q-1}^L + (2^{q-1})^3 + \overline{\Pi}_{q-1}^T + 3(2^{q-1})^3 & \text{if } q > 1 \\ 4 & \text{if } q = 1 \end{cases}
\end{aligned}$$

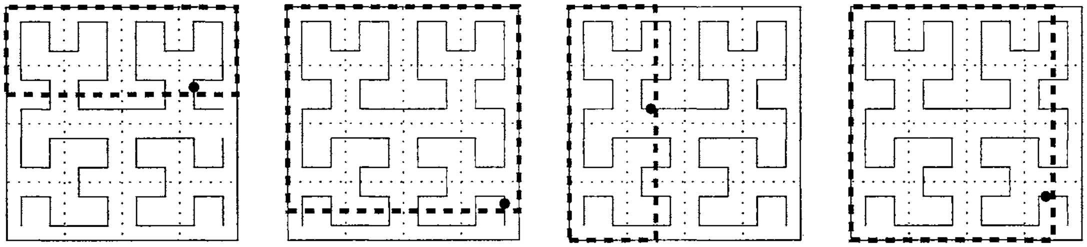


Figure 6.7: (a),(b) scenarios for $\overline{\Pi}_q^T$; (c),(d) scenarios for $\overline{\Pi}_q^L$.

Proof. The scenarios for $\overline{\Pi}_q$ are shown in Figure 6.7: (a) and (b) for $\overline{\Pi}_q^T$: $G_q(x, 1, 2^q, 2^q)$ (a) overlaps with $Q_2(H_q^2)$ and $Q_3(H_q^2)$, and (b) overlaps with $Q_1(H_q^2)$ and $Q_4(H_q^2)$; (c) and (d) for $\overline{\Pi}_q^L$: $G_q(x, 1, 2^q, 2^q)$ (c) overlaps with $Q_1(H_q^2)$ and $Q_2(H_q^2)$, and (d) overlaps with $Q_3(H_q^2)$ and $Q_4(H_q^2)$.

$$
\begin{aligned}
\overline{\Pi}_q^T &= \sum_{x=2^q}^{1} \hbar_q(\theta_2(G_q(x, 1, 2^q, 2^q)), \partial_1(H_q^2)) \\
&= \sum_{x=2^q}^{2^{q-1}+1} \hbar_q(\theta_2(G_q(x, 1, 2^q, 2^q)), \partial_1(H_q^2)) \quad \text{(overlaps with } Q_2(H_q^2), Q_3(H_q^2) \text{ only)} \\
&\quad + \sum_{x=2^{q-1}}^{1} \hbar_q(\theta_2(G_q(x, 1, 2^q, 2^q)), \partial_1(H_q^2)) \quad \text{(overlaps with all four quadrants)} \\
&= \sum_{x=2^q}^{2^{q-1}+1} \hbar_q(\theta_2(G_q(x, 2^{q-1}+1, 2^q, 2^q)), \partial_1(H_q^2)) \quad \text{(Remark 6.2)} \\
&\quad + \sum_{x=2^{q-1}}^{1} \hbar_q(\theta_2(G_q(x, 1, 2^q, 2^q)), \partial_1(H_q^2)) \quad \text{(Remark 6.2)} \\
&= \sum_{x=2^q}^{2^{q-1}+1} (\hbar_q(\theta_2(G_q(x, 2^{q-1}+1, 2^q, 2^q)), \partial_1(Q_3(H_q^2))) + \hbar_q(\partial_1(Q_3(H_q^2), \partial_1(H_q^2))) \\
&\quad + \sum_{x=2^{q-1}}^{1} (\hbar_q(\theta_2(G_q(x, 2^{q-1}+1, 2^q, 2^q)), \partial_1(Q_4(H_q^2))) + \hbar_q(\partial_1(Q_4(H_q^2), \partial_1(H_q^2))) \\
&= \sum_{x=2^{q-1}}^{1} (\hbar_{q-1}(\theta_2(G_{q-1}(x, 1, 2^{q-1}, 2^{q-1})), \partial_1(H_{q-1}^2)) + 2(2^{q-1})^2 \\
&\quad + \sum_{y=2^{q-1}}^{1} (\hbar_{q-1}(\theta_2(G_{q-1}(1, 1, 2^{q-1}, y)), \partial_1(H_{q-1}^2)) + 3(2^{q-1})^2
\end{aligned}
$$

(after $(+\frac{\pi}{2})$-rotating and then left-right reflecting $Q_4(H_q^2)$

into a canonical H^2_{q-1})

$$= \overline{\Pi}^T_{q-1} + 2^{q-1} \cdot 2(2^{q-1})^2 + \overline{\Pi}^L_{q-1} + 2^{q-1} \cdot 3(2^{q-1})^2$$
$$= \overline{\Pi}^T_{q-1} + 2(2^{q-1})^3 + \overline{\Pi}^L_{q-1} + 3(2^{q-1})^3.$$

The proof of $\overline{\Pi}^L_q$ is similar to that of $\overline{\Pi}^T_q$. ∎

The closed-form solutions for $\overline{\Pi}_q$ are employed to establish a system of recurrences for $\overline{\Omega}^c_{q,2^q}$.

Lemma 6.2 *For a canonical H^2_q,*

$$\overline{\Omega}^{c_1}_{q,2^q} = \begin{cases} 2\overline{\Omega}^{c_1}_{q-1,2^{q-1}} + \overline{\Omega}^{c_3}_{q-1,2^{q-1}} + \frac{5^3}{2^8} \cdot 2^{4q} - \frac{3}{2^4} \cdot 2^{2q} & \text{if } q > 1, \\ 7 & \text{if } q = 1; \end{cases}$$

$$\overline{\Omega}^{c_2}_{q,2^q} = \begin{cases} 3\overline{\Omega}^{c_2}_{q-1,2^{q-1}} + \frac{3 \cdot 41}{2^8} \cdot 2^{4q} - \frac{3}{2^4} \cdot 2^{2q} & \text{if } q > 1, \\ 7 & \text{if } q = 1; \end{cases}$$

$$\overline{\Omega}^{c_3}_{q,2^q} = \begin{cases} \overline{\Omega}^{c_1}_{q-1,2^{q-1}} + \overline{\Omega}^{c_3}_{q-1,2^{q-1}} + \frac{23}{2^5} \cdot 2^{4q} - \frac{3}{2^3} \cdot 2^{2q} & \text{if } q > 1, \\ 10 & \text{if } q > 1. \end{cases}$$

Proof. As illustrated in Figure 6.5 and in the case discussion for $\Omega^{c_1}_{q,2^q}$, we split $\overline{\Omega}^{c_1}_{q,2^q}$ into four parts:

$$\overline{\Omega}^{c_1}_{q,2^q} = \sum_{x=1}^{2^q} \sum_{y=1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,y)), \partial_1(H^2_q))$$

$$= \sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} \hbar_q(\theta_2(G_q(1,1,x,y), \partial_1(H^2_q)) \quad \text{(Figure 6.5(a))}$$
$$+ \sum_{x=2^{q-1}+1}^{2^q} \sum_{y=1}^{2^{q-1}} \hbar_q(\theta_2(G_q(1,1,x,y), \partial_1(H^2_q)) \quad \text{(Figure 6.5(b))}$$
$$+ \sum_{x=2^{q-1}+1}^{2^q} \sum_{y=2^{q-1}+1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,y), \partial_1(H^2_q)) \quad \text{(Figure 6.5(c))}$$
$$+ \sum_{x=1}^{2^{q-1}} \sum_{y=2^{q-1}+1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,y), \partial_1(H^2_q)) \quad \text{(Figure 6.5(d))}$$

$$= \sum_{x=1}^{2^{q-1}} \sum_{y=1}^{2^{q-1}} (\hbar_q(\theta_2(G_q(1,1,x,y)), \partial_1(Q_1(H^2_q))) + \hbar_q(\partial_1(Q_1(H^2_q)), \partial_1(H^2_q)))$$
$$+ \sum_{x=2^{q-1}+1}^{2^q} \sum_{y=1}^{2^{q-1}} (\hbar_q(\theta_2(G_q(1,1,x,y)), \partial_1(Q_2(H^2_q))) + \hbar_q(\partial_1(Q_2(H^2_q)), \partial_1(H^2_q)))$$

$$+\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q}(\hbar_q(\theta_2(G_q(1,1,x,y)),\partial_1(Q_4(H_q^2)))$$

$$+\hbar_q(\partial_1(Q_4(H_q^2)),\partial_1(H_q^2))) \qquad (\theta_2 \text{ in } Q_4(H_q^2) \text{ by Remark 6.2})$$

$$+\sum_{x=1}^{2^{q-1}}\sum_{y=2^{q-1}+1}^{2^q}(\hbar_q(\theta_2(G_q(1,1,x,y)),\partial_1(Q_4(H_q^2)))+\hbar_q(\partial_1(Q_4(H_q^2)),\partial_1(H_q^2)))$$

$$= \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}}(\hbar_{q-1}(\theta_2(G_{q-1}(1,1,x,y)),\partial_1(H_{q-1}^2))+0\cdot(2^{q-1})^2)$$

$$+\sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}}(\hbar_{q-1}(\theta_2(G_{q-1}(1,1,x,y)),\partial_1(H_{q-1}^2))+1\cdot(2^{q-1})^2)$$

$$+\sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}}(\hbar_{q-1}(\theta_2(G_{q-1}(1,1,2^{q-1},y)),\partial_1(H_{q-1}^2))+3\cdot(2^{q-1})^2)$$

$$\text{(after } (+\frac{\pi}{2})\text{-rotating and then left-right reflecting } Q_4(H_q^2) \text{ into a canonical } H_{q-1}^2)$$

$$+\sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}}(\hbar_{q-1}(\theta_2(G_{q-1}(x,y,2^{q-1},2^{q-1})),\partial_1(H_{q-1}^2))+3\cdot(2^{q-1})^2)$$

$$\text{(after } (+\frac{\pi}{2})\text{-rotating and then left-right reflecting } Q_4(H_q^2) \text{ into a canonical } H_{q-1}^2)$$

$$= \overline{\Omega}_{q-1,2^{q-1}}^{c_1}+\overline{\Omega}_{q-1,2^{q-1}}^{c_1}+(2^{q-1})^4+(\overline{\Pi}_{q-1}^L+3(2^{q-1})^2)(2^{q-1})^2$$

$$+\overline{\Omega}_{q-1,2^{q-1}}^{c_3}+3(2^{q-1})^4$$

$$= 2\overline{\Omega}_{q-1,2^{q-1}}^{c_1}+\overline{\Omega}_{q-1,2^{q-1}}^{c_3}+(2^{q-1})^2\overline{\Pi}_{q-1}^L+7(2^{q-1})^4$$

$$= 2\overline{\Omega}_{q-1,2^{q-1}}^{c_1}+\overline{\Omega}_{q-1,2^{q-1}}^{c_3}+\frac{5^3}{2^8}\cdot 2^{4q}-\frac{3}{2^4}\cdot 2^{2q} \qquad \text{(Lemma 6.1)}.$$

The proofs for $\overline{\Omega}_{q,2^q}^{c_2}$ and $\overline{\Omega}_{q,2^q}^{c_3}$ are similar to this one. ∎

The closed-form solutions for $\overline{\Omega}_{q,2^q}^{c}$ and $\overline{\Pi}_q$ are employed to obtain exact formulas for $\Omega_{k,2^q}^c$.

Lemma 6.3 *For a canonical H_k^2 structured as an H_{k-q}^2-curve interconnecting $2^{2(k-q)}$ H_q^2-subcurves,*

$$\Omega_{k,2^q}^{c_1} = \overline{\Omega}_{q,2^q}^{c_1}-\overline{\Pi}_q^L-(2^q-1)(2^{2q}-1),$$

$$\Omega^{c_2}_{k,2^q} = \overline{\Omega}^{c_2}_{q,2^q} - \overline{\Pi}^T_q - \overline{\Pi}^L_q + (2^{2q}-1) + (2^q-1)^2 \sum_{i=q}^{k-1} 2^{2i},$$

$$\Omega^{c_3}_{k,2^q} = \overline{\Omega}^{c_3}_{q,2^q} - \overline{\Pi}^T_q - (2^q-1)(2^{2q}-1) + (2^q-1)^2 (2^{2k} - \sum_{i=q}^{k-1} 2^{2i} - 2^{2q}).$$

Proof. By the definition, we have:

$$\Omega^{c_1}_{k,2^q} = \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(H_k^2))$$

$$= \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(H_k^2))$$
$$- \sum_{x=1}^{2^q-1}\sum_{y=2^q}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(H_k^2))$$
$$= (\sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(H_k^2))$$
$$- \sum_{x=2^q}^{2^q}\sum_{y=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(H_k^2)))$$
$$-(\sum_{x=1}^{2^q}\sum_{y=2^q}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(H_k^2))$$
$$- \sum_{x=2^q}^{2^q}\sum_{y=2^q}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(H_k^2)))$$
$$= \overline{\Omega}^{c_1}_{q,2^q}$$
$$- \sum_{y=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,2^q,y)), \partial_1(H_k^2))$$
$$- \sum_{x=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,2^q)), \partial_1(H_k^2)) \qquad (G_k \text{ containing } \partial_2(H_q^2))$$
$$+\hbar_{kq}(\theta_2(G_k(1,1,2^q,2^q)), \partial_1(H_k^2))) \qquad (\theta_2(G(1,1,2^q,2^q)) = \partial_2(H_q^2))$$
$$= \overline{\Omega}^{c_1}_{q,2^q} - \overline{\Pi}^L_q - (2^{2q}-1)\cdot 2^q + (2^{2q}-1)$$
$$= \overline{\Omega}^{c_1}_{q,2^q} - \overline{\Pi}^L_q - (2^q-1)(2^{2q}-1).$$

The proofs of $\Omega^{c_2}_{k,2^q}$ and $\Omega^{c_3}_{k,2^q}$ are similar to this one. ∎

The exact formulas for $\Omega^c_{k,2^q}$ are employed to establish a system of recurrences for $\Omega_{k,2^q}$.

Lemma 6.4 *For a canonical H^2_k structured as an H^2_{k-q}-curve interconnecting $2^{2(k-q)}$ H^2_q-subcurves,*

$$\Omega^L_{k,2^q} = \begin{cases} \Omega^B_{k-1,2^q} + (\Omega^{c_1}_{k-1,2^q} + (2^q-1)^2(2^{k-1})^2) \\ \quad + (\Omega^L_{k-1,2^q} + (2^{k-1}-2^q+1)(2^q-1)(2^{k-1})^2) & \text{if } k > q, \\ \overline{\Pi}^L_q - (2^{2q}-1) & \text{if } k = q; \end{cases}$$

$$\Omega^R_{k,2^q} = \begin{cases} (\Omega^B_{k-1,2^q} + 3(2^{k-1}-2^q+1)(2^q-1)(2^{k-1})^2) \\ \quad + (\Omega^{c_1}_{k-1,2^q} + 3(2^q-1)^2(2^{k-1})^2) \\ \quad + (\Omega^R_{k-1,2^q} + 2(2^{k-1}-2^q+1)(2^q-1)(2^{k-1})^2) & \text{if } k > q, \\ (2^q-1)(2^{2q}-1) & \text{if } k = q; \end{cases}$$

$$\Omega^B_{k,2^q} = \begin{cases} \Omega^L_{k-1,2^q} + (\Omega^{c_3}_{k-1,2^q} + 3(2^q-1)^2(2^{k-1})^2) \\ \quad + (\Omega^R_{k-1,2^q} + 3(2^{k-1}-2^q+1)(2^q-1)(2^{k-1})^2) & \text{if } k > q, \\ (2^q-1)(2^{2q}-1) & \text{if } k = q; \end{cases}$$

$$\Omega^T_{k,2^q} = \begin{cases} (\Omega^T_{k-1,2^q} + (2^{k-1}-2^q+1)(2^q-1)(2^{k-1})^2) \\ \quad + (\Omega^{c_2}_{k-1,2^q} + 2(2^q-1)^2(2^{k-1})^2) \\ \quad + (\Omega^T_{k-1,2^q} + 2(2^{k-1}-2^q+1)(2^q-1)(2^{k-1})^2) & \text{if } k > q, \\ \overline{\Pi}^T_q - (2^{2q}-1) & \text{if } k = q. \end{cases}$$

Proof. Similar to the proof of Lemma 6.3, from the definition, we have (see Figure 6.3):

$$\Omega^L_{k,2^q} = \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)), \partial_1(H^2_k))$$

$$= \sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)), \partial_1(H^2_k)) \qquad (\text{in } Q_1(H^2_k))$$

$$+ \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)), \partial_1(H^2_k))$$
$$(\text{across } Q_1(H^2_k), Q_2(H^2_k))$$

$$+ \sum_{x=2^{k-1}+1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)), \partial_1(H^2_k)) \qquad (\text{in } Q_2(H^2_k))$$

$$= \sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} (\hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)), \partial_1(Q_1(H^2_k)))$$
$$+ \hbar_k(\partial_1(Q_1(H^2_k)), \partial_1(H^2_k)))$$

$$+ \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(2^{k-1}+1,1,x+2^q-1,y)),\partial_1(H_k^2))$$

(Remark 6.2)

$$+ \sum_{x=2^{k-1}+1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} (\hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)),\partial_1(Q_2(H_k^2)))$$
$$+ \hbar_k(\partial_1(Q_2(H_k^2)),\partial_1(H_k^2)))$$

$$= \sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} (\hbar_{k-1}(\theta_2(G_{k-1}(x,1,x+2^q-1,y)),\partial_1(H_{k-1}^2)) + 0\cdot(2^{k-1})^2)$$

(after $(-\frac{\pi}{2})$-rotating and then left-right reflecting $Q_1(H_k^2)$ into a canonical H_{k-1}^2)

$$+ \sum_{x=2^{k-1}+1}^{2^{k-1}+2^q-1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(2^{k-1}+1,1,x,y)),\partial_1(H_k^2)) \qquad \text{(change index of } y\text{)}$$

$$+ \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^{k-1}-2^q+1} (\hbar_{k-1}(\theta_2(G_{k-1}(1,y,x,y+2^q-1)),\partial_1(H_{k-1}^2)) + 1\cdot(2^{k-1})^2)$$

$$= \Omega^B_{k-1,2^q} + 0$$
$$+ \sum_{x=2^{k-1}+1}^{2^{k-1}+2^q-1} \sum_{y=1}^{2^q-1} (\hbar_k(\theta_2(G_k(1,2^{k-1}+1,x,y)),\partial_1(Q_2(H_k^2)))$$
$$+ \hbar_k(\partial_1(Q_2(H_k^2)),\partial_1(H_k^2)))$$
$$+\Omega^L_{k-1,2^q} + 1\cdot(2^{k-1})^2(2^q-1)(2^{k-1}-2^q+1)$$

$$= \Omega^B_{k-1,2^q} + (\Omega^L_{k-1,2^q} + (2^{k-1})^2(2^q-1)(2^{k-1}-2^q+1))$$
$$+ \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} (\hbar_{k-1}(\theta_2(G_{k-1}(1,1,x,y)),\partial_1(H_{k-1}^2)) + 1\cdot(2^{k-1})^2)$$

($Q_2(H_k^2)$ is a canonical H_{k-1}^2)

$$= \Omega^B_{k-1,2^q} + (\Omega^{c_1}_{k-1,2^q} + (2^q-1)^2(2^{k-1})^2)$$
$$+(\Omega^L_{k-1,2^q} + (2^{k-1})^2(2^q-1)(2^{k-1}-2^q+1)).$$

For $\Omega^R_{k-1,2^q}$, $\Omega^B_{k-1,2^q}$, and $\Omega^T_{k-1,2^q}$, the proofs are similar to this one. ∎

6.2.2 Computing $\sum \hbar_k(\theta_1(G), \partial_1(H_k^2))$ over Subgrids G Overlapping with Two Quadrants

We may proceed as in Section 6.2.1, based upon the following system of summations $\omega_{k,2^q} = (\omega_{k,2^q}^L, \omega_{k,2^q}^R, \omega_{k,2^q}^B, \omega_{k,2^q}^T)$:

$$\begin{aligned}
\omega_{k,2^q}^L &= \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \hbar(\theta_1(G_k(x,1,x+2^q-1,y)), \partial_1(H_k^2))\text{—for left boundary,}\\
\omega_{k,2^q}^R &= \sum_{x=1}^{2^k-2^q+1} \sum_{y=2^k-2^q+2}^{2^k} \hbar(\theta_1(G_k(x,y,x+2^q-1,2^k)), \partial_1(H_k^2))\text{—for right boundary,}\\
\omega_{k,2^q}^B &= \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^k-2^q+1} \hbar(\theta_1(G_k(1,y,x,y+2^q-1)), \partial_1(H_k^2))\text{—for bottom boundary, and}\\
\omega_{k,2^q}^T &= \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^k-2^q+1} \hbar(\theta_1(G_k(x,y,2^k,y+2^q-1)), \partial_1(H_k^2))\text{—for top boundary.}
\end{aligned}$$

Or, we apply the following lemma to relate the two systems $\omega_{k,2^q}$ and $\Omega_{k,2^q}$.

Lemma 6.5 *For a canonical H_k^2,*

$$\begin{aligned}
\omega_{k,2^q}^L + \Omega_{k,2^q}^R &= (2^{2k}-1)\mathcal{N}_{k,2^q}^S,\\
\omega_{k,2^q}^R + \Omega_{k,2^q}^L &= (2^{2k}-1)\mathcal{N}_{k,2^q}^S,\\
\omega_{k,2^q}^T + \Omega_{k,2^q}^T &= (2^{2k}-1)\mathcal{N}_{k,2^q}^S,\\
\omega_{k,2^q}^B + \Omega_{k,2^q}^B &= (2^{2k}-1)\mathcal{N}_{k,2^q}^S.
\end{aligned}$$

Proof. A canonical H_k^2 is left-right reflexive. For a grid point $v = (x, y)$, its mirror point $v' = (x, 2^k + 1 - y)$, and the mirror pair (v, v') satisfies that:

$$\hbar_k(v, \partial_1(H_k^2)) + \hbar_k(v', \partial_1(H_k^2)) = \hbar_k(v, \partial_2(H_k^2)) + \hbar_k(v', \partial_2(H_k^2)) = 2^{2k} - 1.$$

The right reflection of $G_k(x, 1, x+2^q-1, y)$, where $y \in [2^q]$ and $q < k$, is $G_k(x, 2^k+1-y, x+2^q-1, 2^k)$, and the reflection of the lowest indexed point in $G_k(x, 1, x+2^q-1, y)$ is the highest indexed in $G_k(x, 2^k+1-y, x+2^q-1, 2^k)$; that is, $\theta_1(G_k(x, 1, x+2^q-1, y))$ and $\theta_2(G_k(x, 2^k+1-y, x+2^q-1, 2^k))$ are mirror pair. Thus,

$$\omega_{k,2^q}^L + \Omega_{k,2^q}^R$$

$$\begin{aligned}
&= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar(\theta_1(G_k(x,1,x+2^q-1,y)),\partial_1(H_k^2)) \\
&\quad + \sum_{x=1}^{2^k-2^q+1}\sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_2(G_k(x,y,x+2^q-1,2^k)),\partial_1(H_k^2)) \\
&= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar(\theta_1(G_k(x,1,x+2^q-1,y)),\partial_1(H_k^2)) \\
&\quad + \sum_{x=1}^{2^k-2^q+1}\sum_{\gamma=2^q-1}^{1} \hbar_k(\theta_2(G_k(x,2^k+1-\gamma,x+2^q-1,2^k)),\partial_1(H_k^2)) \\
&\qquad\qquad \text{(change of summation index: } \gamma = 2^k+1-y) \\
&= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar(\theta_1(G_k(x,1,x+2^q-1,y)),\partial_1(H_k^2)) \\
&\quad + \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,2^k+1-y,x+2^q-1,2^k)),\partial_1(H_k^2)) \\
&\qquad\qquad \text{(change of summation index: } y = \gamma) \\
&= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} (\hbar(\theta_1(G_k(x,1,x+2^q-1,y)),\partial_1(H_k^2)) \\
&\qquad + \hbar_k(\theta_2(G_k(x,2^k+1-y,x+2^q-1,2^k)),\partial_1(H_k^2))) \\
&= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} (2^{2k}-1) \qquad \text{(mirror pair)} \\
&= (2^{2k}-1)\sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} 1 \\
&= (2^{2k}-1)\mathcal{N}_{k,2^q}^S.
\end{aligned}$$

The proofs for $\omega_{k,2^q}^L+\Omega_{k,2^q}^R$, $\omega_{k,2^q}^T+\Omega_{k,2^q}^T$, and $\omega_{k,2^q}^B+\Omega_{k,2^q}^B$ are similar to this one. ▮

6.2.3 Query Subgrids Overlapping with All Quadrants

For a $2^q \times 2^q$ query subgrid $G \subseteq \mathcal{R}$, we have: (1) $\theta_2(G) \in Q_4(H_k^2)$ and (2) $\theta_1(G) \in Q_1(H_k^2)$ by Remark 6.2.

For (1), when zooming in on the incomplete rectangular subgrid $G \cap Q_4(H_k^2)$ (with both side-lengths at most $2^q - 1$), we reduce $\sum_{\text{all} G\subseteq\mathcal{R}} \hbar_k(\theta_2(G),\partial_1(H_k^2))$ to

$\Omega^{c2}_{k-1,2^q}$ after $(+\frac{\pi}{2})$-rotating and left-right reflecting $Q_4(H^2_k)$ into a canonical H^2_{k-1} (with adjustment of distance cumulation).

For (2), similar consideration leads to a reduction of $\sum_{\mathrm{all} G \subseteq \mathcal{R}} \hbar_k(\theta_1(G), \partial_1(H^2_k))$ to $\omega^{c3}_{k-1,2^q}$, where $\omega^{c3}_{k,2^q}$ denotes $\sum_{x=2^k-2^q+2}^{2^k} \sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_1(G_k(x,y,2^k,2^k)), \partial_1(H^2_k))$ for a canonical H^2_k and is related to $\Omega^{c2}_{k,2^q}$ as follows.

Lemma 6.6 *For a canonical* H^2_k*,* $\omega^{c3}_{k,2^q} + \Omega^{c2}_{k,2^q} = (2^{2k}-1)\mathcal{N}^c_{k,2^q}$.

Proof. Similar to the proof of Lemma 6.5,

$$
\begin{aligned}
& \omega^{c3}_{k,2^q} + \Omega^{c2}_{k,2^q} \\
= & \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_1(G_k(x,y,2^k,2^k)), \partial_1(H^2_k)) \\
& + \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,2^k,y)), \partial_1(H^2_k)) \\
= & \sum_{x=2^k-2^q+2}^{2^k} \sum_{\gamma=2^q-1}^{1} \hbar_k(\theta_1(G_k(x,2^k+1-\gamma,2^k,2^k)), \partial_1(H^2_k)) \\
& \qquad \text{(change of summation index: } \gamma = 2^k+1-y) \\
& + \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,2^k,y)), \partial_1(H^2_k)) \\
= & \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^q-1} \hbar_k(\theta_1(G_k(x,2^k+1-y,2^k,2^k)), \partial_1(H^2_k)) \\
& \qquad \text{(change of summation index: } y = \gamma) \\
& + \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,2^k,y)), \partial_1(H^2_k)) \\
= & \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^q-1} (\hbar_k(\theta_1(G_k(x,2^k+1-y,2^k,2^k)), \partial_1(H^2_k)) \\
& \quad + \hbar_k(\theta_2(G_k(x,1,2^k,y)), \partial_1(H^2_k))) \\
= & \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^q-1} (2^{2k}-1) \qquad \text{(mirror pair)} \\
= & (2^{2k}-1) \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^q-1} 1
\end{aligned}
$$

$$= (2^{2k}-1)\mathcal{N}^c_{k,2^q}.$$

∎

Thus, the summation of all inter-cluster distances over all $2^q \times 2^q$ query subgrids contained in $\mathcal{R}$ is

$$(\Omega^{c2}_{k-1,2^q} + 3\cdot 2^{2k-2}\mathcal{N}^c_{k-1,2^q}) - \omega^{c3}_{k-1,2^q} - (2^{2q}-1)\mathcal{N}^c_{k-1,2^q}.$$

6.2.4 The Big Picture: Computing $\Psi_q(H^2_k)$

The results in the previous three subsections yield $\varepsilon_{k,q}(H^2_k)$. Hence, we have the following theorem for $\Psi_q(H^2_k)$.

Theorem 6.1 *For a canonical H^2_k, the recurrence for summation of all inter-cluster distances over all $2^q \times 2^q$ query subgrids of an H^2_k-structural grid space $[2^k]^2$ is:*

$$\Psi_q(H^2_k) = \begin{cases} 4\Psi_q(H^2_{k-1}) + (\Omega^B_{k-1,2^q} + 2^{2k-2}\mathcal{N}^S_{k-1,2^q}) \\ \quad - \omega^R_{k-1,2^q} - (2^{2q}-1)\mathcal{N}^S_{k-1,2^q} \\ \quad + (\Omega^L_{k-1,2^q} + 2\cdot 2^{2k-2}\mathcal{N}^S_{k-1,2^q}) \\ \quad - (\omega^R_{k-1,2^q} + 2^{2k-2}\mathcal{N}^S_{k-1,2^q}) - (2^{2q}-1)\mathcal{N}^S_{k-1,2^q} \\ \quad + (\Omega^L_{k-1,2^q} + 3\cdot 2^{2k-2}\mathcal{N}^S_{k-1,2^q}) \\ \quad - (\omega^B_{k-1,2^q} + 2\cdot 2^{2k-2}\mathcal{N}^S_{k-1,2^q}) - (2^{2q}-1)\mathcal{N}^S_{k-1,2^q} \\ \quad + (\Omega^T_{k-1,2^q} + 3\cdot 2^{2k-2}\mathcal{N}^S_{k-1,2^q}) \\ \quad - (\omega^T_{k-1,2^q}) - (2^{2q}-1)\mathcal{N}^S_{k-1,2^q} \\ \quad + (\Omega^{c2}_{k-1,2^q} + 3\cdot 2^{2k-2}\mathcal{N}^c_{k-1,2^q}) \\ \quad - (\omega^{c3}_{k-1,2^q}) - (2^{2q}-1)\mathcal{N}^c_{k-1,2^q} & \text{if } k > q, \\ 0 & \text{if } k = q. \end{cases}$$

The exact formula for $\Psi_q(H^2_k)$ is:

$$\begin{aligned} \Psi_q(H^2_k) = {} & \frac{17}{14}\cdot 2^{3k+q} - \frac{17}{14}\cdot 2^{3k} - \frac{20885}{8151}\cdot 2^{2k+2q} + \frac{139}{48}\cdot 2^{2k+q} + \frac{7}{39}\cdot 2^{2k-2q}\cdot 3^q \\ & - 2^{2k-1} - \frac{1}{3}\cdot 2^{2k-q-2} + \frac{31}{1254}\cdot 2^{2k-2q}((\frac{3+\sqrt{5}}{2})^q + (\frac{3-\sqrt{5}}{2})^q) \\ & + \frac{21\cdot\sqrt{5}}{2090}\cdot 2^{2k-2q}((\frac{3+\sqrt{5}}{2})^q + (\frac{3-\sqrt{5}}{2})^q) + \frac{29767}{21736}\cdot 2^{k+3q} - 13\cdot 2^{k+2q-4} \\ & - 2^{k+q} - \frac{7}{39}\cdot 2^{k-q}\cdot 3^q + 3\cdot 2^{k-2} - \frac{63\cdot\sqrt{5}}{2090}\cdot 2^{k-q}((\frac{3+\sqrt{5}}{2})^q - (\frac{3-\sqrt{5}}{2})^q) \\ & - \frac{31}{418}\cdot 2^{k-q}((\frac{3+\sqrt{5}}{2})^q + (\frac{3-\sqrt{5}}{2})^q) - \frac{755}{35112}\cdot 2^{4q} - \frac{73}{21}\cdot 2^{3q-2} + 3\cdot 2^{2q-1} \end{aligned}$$

$$+\frac{21\cdot\sqrt{5}}{1045}((\frac{3+\sqrt{5}}{2})^q-(\frac{3-\sqrt{5}}{2})^q)+\frac{31}{627}((\frac{3+\sqrt{5}}{2})^q+(\frac{3-\sqrt{5}}{2})^q)-\frac{1}{3}\cdot 2^{q+1}.$$

6.2.5 Total Number of Inter-cluster Gaps

In order to compute the (universe) mean inter-cluster distance over all inter-cluster gaps from all identically shaped subgrids, we need to derive the total number of inter-cluster gaps, denoted by $\Phi_{k,q}(H_k^2)$ for a canonical H_k^2.

For a grid space indexed by a space-filling curve, since the clusters interleave with the inter-cluster gaps of every query subgrid, we have:

$$\begin{aligned}&\text{total number of inter-cluster gaps}\\ =\ &\text{total number of clusters} - \text{total number of query subgrids.}\end{aligned}$$

As discussed in Chapter V,

$$\text{total number of clusters} = \text{total number of edges cut by all query subgrids}/2,$$

and the exact formula that we have derived for the total number of edges that cut all $2^q \times 2^q$ query subgrids is:

$$E_{k,q}(H_k^2) = 2^{2k+q+1} - 2^{k+2q+2} + 2^{k+q+1} + 2^{k-q+1} + 2^{3q+1} - 2^{2q+1}.$$

Hence, the total number of inter-cluster gaps:

$$\begin{aligned}\Phi_{k,q}(H_k^2) &= \frac{E_{k,q}(H_k^2)}{2} - (2^k - 2^q + 1)^2\\ &= 2^{2k+q} - 2^{2k} - 2^{k+2q+1} + 3\cdot 2^{k+q} - 2^{k+1} + 2^{k-q}\\ &\quad +2^{3q} - 2^{2q+1} + 2^{q+1} - 1.\end{aligned}$$

6.3 Analytical Study of Inter-Clustering Performances for z-Order Curve

Similar to the approach in the previous section for Hilber curve H_k^2, we now adopt the same approach for z-order curve Z_k^2. Following the approach in Section 6.2, we apply same notations to computations in Z_k^2. For a 2-dimensional z-order curve of order-k Z_k^2, let $\Psi_{k,q}(Z_k^2)$ denote the summation of all inter-cluster distances over all $2^q \times 2^q$

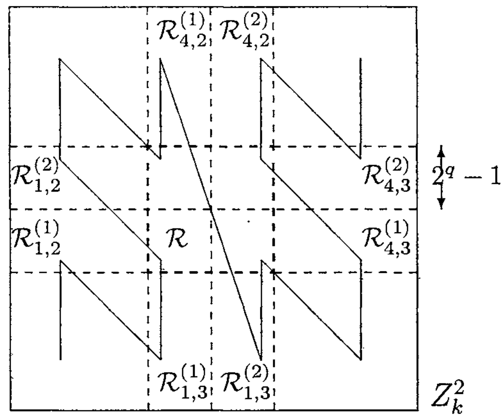


Figure 6.8: The boundary regions of neighboring quadrants are organized into nine disjoint regions: $\mathcal{R}^{(1)}_{i,j}$, $\mathcal{R}^{(2)}_{i,j}$ for $(i,j) \in \{1,4\} \times \{2,3\}$, and $\mathcal{R}$.

query subgrids of an Z^2_k-structural grid space $[2^k]^2$. The recursive decomposition of Z^2_k (see Figure 2.3) gives that:

$$\Psi_{k,q}(Z^2_k) = 4\Psi_{k,q}(Z^2_{k-1}) + \varepsilon_{k,q}(Z^2_k),$$

where $\varepsilon_{k,q}(Z^2_k)$ denotes the summation of all inter-cluster distances over all $2^q \times 2^q$ query subgrids, each of which overlaps with more than one quadrant (that is, two or four). These query subgrids are contained in the boundary regions of neighboring quadrants, which can be organized into nine disjoint regions: $\mathcal{R}^{(1)}_{i,j}$, $\mathcal{R}^{(2)}_{i,j}$ for $(i,j) \in \{1,4\} \times \{2,3\}$, and R, as shown in Figure 6.8.

For a $2^q \times 2^q$ query subgrid G, G overlaps with:

1. Exactly $Q_i(Z^2_k)$ and $Q_j(Z^2_k)$ if and only if $G \subseteq \mathcal{R}^{(1)}_{i,j} \cup \mathcal{R}^{(2)}_{i,j}$ for every $(i,j) \in \{1,4\} \times \{2,3\}$. In this case, $\theta_\eta(G) \in \mathcal{R}^{(\eta)}_{i,j}$ for $\eta \in \{1,2\}$ by Remark 6.2.

2. $Q_i(Z^2_k)$ for all $i \in \{1,2,3,4\}$ if and only if $G \subseteq \mathcal{R}$. In this case, $\theta_1(G) \in Q_1(Z^2_k)$ (upper-right corner) and $\theta_2(G) \in Q_4(Z^2_k)$ (lower-left corner) by Remark 6.2.

We divide the computation of $\varepsilon_{k,q}(Z^2_k)$ into three parts:

1. $\sum \hbar_k(\theta_2(G), \partial_1(Z^2_k))$ over all $2^q \times 2^q$ query subgrids $G \subseteq \mathcal{R}^{(1)}_{i,j} \cup \mathcal{R}^{(2)}_{i,j}$ for $(i,j) \in \{1,4\} \times \{2,3\}$,

2. $\sum \hbar_k(\theta_1(G), \partial_1(Z_k^2))$ over all $2^q \times 2^q$ query subgrids $G \subseteq \mathcal{R}_{i,j}^{(1)} \cup \mathcal{R}_{i,j}^{(2)}$ for $(i,j) \in \{1,4\} \times \{2,3\}$, and

3. the summation of all inter-cluster distances over all $2^q \times 2^q$ query subgrids contained in $\mathcal{R}$.

As in the previous section, we develop combinatorial lemmas in the following three subsections to support the computations.

6.3.1 $\sum \hbar_k(\theta_2(G), \partial_1(Z_k^2))$ over Subgrids G Overlapping with Two Quadrants

We set up the system for $\Omega_{k,2^q} = (\Omega_{k,2^q}^L, \Omega_{k,2^q}^R, \Omega_{k,2^q}^B, \Omega_{k,2^q}^T)$ in a general context of a canonical Z_k^2:

$$\begin{aligned}
\Omega_{k,2^q}^L &= \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)), \partial_1(Z_k^2)) \\
&\quad \text{— for left boundary (see Figure 6.9(a)),} \\
\Omega_{k,2^q}^R &= \sum_{x=1}^{2^k-2^q+1} \sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_2(G_k(x,y,x+2^q-1,2^k)), \partial_1(Z_k^2)) \\
&\quad \text{— for right boundary,} \\
\Omega_{k,2^q}^B &= \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^k-2^q+1} \hbar_k(\theta_2(G_k(1,y,x,y+2^q-1)), \partial_1(Z_k^2)) \\
&\quad \text{— for bottom boundary,} \\
\Omega_{k,2^q}^T &= \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^k-2^q+1} \hbar_k(\theta_2(G_k(x,y,2^k,y+2^q-1)), \partial_1(Z_k^2)) \\
&\quad \text{— for top boundary, and} \\
\mathcal{N}_{k,2^q}^S &= \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} 1 \\
&\quad \text{— for the number of incomplete rectangular subgrids in a boundary.}
\end{aligned}$$

We will establish a system of recurrences (in k) for $\Omega_{k,2^q}$ (see Lemma 6.10 below). The system of recurrence involves another system of summations as prerequisites $\Omega_{k,2^q}^c = (\Omega_{k,2^q}^{c_1}, \Omega_{k,2^q}^{c_2}, \Omega_{k,2^q}^{c_3}, \Omega_{k,2^q}^{c_4})$, as demonstrated in the example in the computation for Hilbert curve.

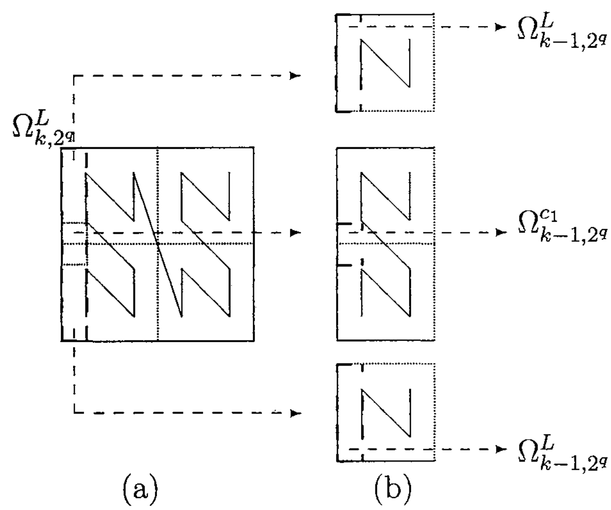


Figure 6.9: (a) $\Omega^L_{k,2^q}$ for a canonical Z^2_k; (b) its recursive decomposition.

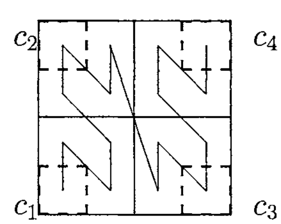


Figure 6.10: The four $(2^q - 1) \times (2^q - 1)$ corners of a canonical Z^2_k.

The system for $\Omega^{c}_{k,2^q} = (\Omega^{c1}_{k,2^q}, \Omega^{c2}_{k,2^q}, \Omega^{c3}_{k,2^q}, \Omega^{c4}_{k,2^q})$ in a general context of a canonical Z^2_k (see Figure 6.10) are:

$$\Omega^{c1}_{k,2^q} = \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z^2_k)) \text{ — for lower-left corner,}$$

$$\Omega^{c2}_{k,2^q} = \sum_{x=2^k-2^q+2}^{2^k}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,2^k,y)), \partial_1(Z^2_k)) \text{ — for upper-left corner,}$$

$$\Omega^{c3}_{k,2^q} = \sum_{x=1}^{2^q-1}\sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_2(G_k(1,y,x,2^k)), \partial_1(Z^2_k)) \text{ — for lower-right corner, and}$$

$$\Omega^{c4}_{k,2^q} = \sum_{x=2^k-2^q+2}^{2^k}\sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_2(G_k(x,y,2^k,2^k)), \partial_1(Z^2_k)) \text{ — for upper-right corner,}$$

$$\mathcal{N}^{c}_{k,2^q} = \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} 1 \text{ — for the number of incomplete rectangular subgrids in a corner.}$$

Unlike the structure of Hilbert curve, there are no reflection and rotation required for subcurves in z-order to be a canonical z-order curve. The $\theta_2(G)$ is always in the upper-right corner of G; $\theta_1(G)$ is always in the lower-left corner of G. When G overlaps more than one quadrants, $\theta_2(G)$ can only be in left side, bottom side or lower-left corner of the right, upper or upper-right quadrant, respectively; $\theta_1(G)$ can only be in right side, top side or upper-right corner of the left, bottom or upper-right quadrant, respectively. Thus, for $\Omega^{c}_{k,2^q}$, we focus on $\Omega^{c1}_{k,2^q}$ only; for $\Omega_{k,2^q}$, we focus on $\Omega^{L}_{k,2^q}, \Omega^{B}_{k,2^q}$.

The prerequisite system of summations $\overline{\Omega}^{c}_{q,2^q} = (\overline{\Omega}^{c1}_{q,2^q})$:

$$\overline{\Omega}^{c1}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \hbar_k(\theta_2(G_q(1,1,x,y)), \partial_1(Z^2_q)) \text{ — for lower-left corner,}$$

$$\overline{\mathcal{N}}^{c}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} 1 \text{ — for the number of rectangular subgrids in a } 2^q \times 2^q \text{ corner.}$$

In a general context of a canonical Z^2_q, the recursive decompositions of $\overline{\Omega}^{c1}_{q,2^q}$ need a prerequisite system of summations $\overline{\Pi}_q = (\overline{\Pi}^{B}_q, \overline{\Pi}^{L}_q)$:

$$\overline{\Pi}^{B}_q = \sum_{x=1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,2^q)), \partial_1(Z^2_q)) \text{ — bottom to top incrementally, and}$$

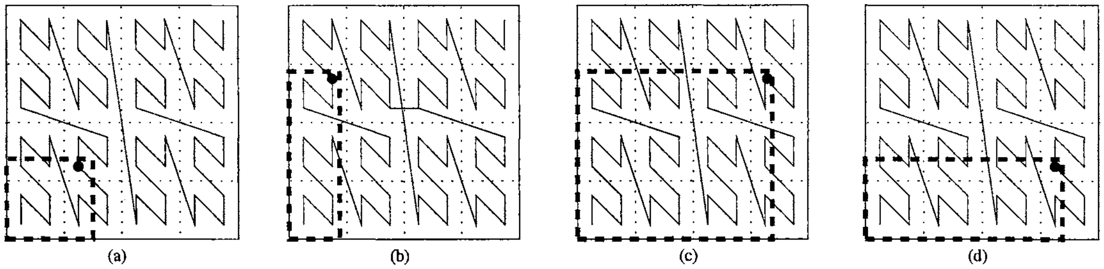


Figure 6.11: Four overlapping scenarios when decomposing $\overline{\Omega}^{c_1}_{q,2^q}$ in a canonical Z^2_q: (a) contained in $Q_1(Z^2_q)$; (b) and (d) overlapping with exactly two quadrants; (c) overlapping with all quadrants.

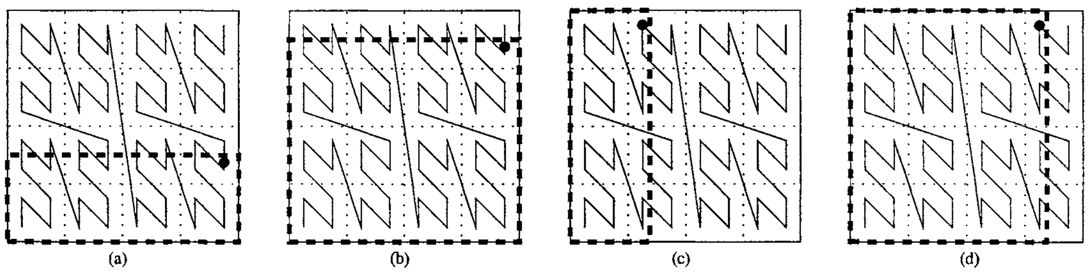


Figure 6.12: (a),(b) scenarios for $\overline{\Pi}^B_q$; (c),(d) scenarios for $\overline{\Pi}^L_q$.

$$\overline{\Pi}^L_q = \sum_{y=1}^{2^q} \hbar_q(\theta_2(G_q(1,1,2^q,y)), \partial_1(Z^2_q)) \text{ — left to right incrementally.}$$

As in the derivation for Hilbert curve, we develop and solve a system of recurrences for $\overline{\Pi}_q$ and reverse the sequence of reductions to obtain the closed-form solutions for $\Omega_{k,2^q}$, which are summarized in the following four lemmas.

Lemma 6.7 *For a canonical Z^2_q,*

$$\overline{\Pi}^B_q = \begin{cases} 2\overline{\Pi}^B_{q-1} + 5(2^{q-1})^3 & \text{if } q > 1 \\ 5 & \text{if } q = 1 \end{cases}$$

$$\overline{\Pi}^L_q = \begin{cases} 2\overline{\Pi}^L_{q-1} + 4(2^{q-1})^3 & \text{if } q > 1 \\ 4 & \text{if } q = 1 \end{cases}$$

Proof. The scenarios for $\overline{\Pi}_q$ are shown in Figure 6.12: (a) and (b) for $\overline{\Pi}^B_q$: $G_q(1,1,x,2^q)$ (a) overlaps with $Q_1(Z^2_q)$ and $Q_3(Z^2_q)$, and (b) overlaps with $Q_2(Z^2_q)$ and $Q_4(Z^2_q)$; (c) and (d) for $\overline{\Pi}^L_q$: $G_q(x,1,2^q,2^q)$ (c) overlaps with $Q_1(Z^2_q)$ and $Q_2(Z^2_q)$, and (d) overlaps with $Q_3(Z_q^2)$ and $Q_4(Z_q^2)$.

$$
\begin{aligned}
\overline{\Pi}_q^B &= \sum_{x=1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,2^q)), \partial_1(Z_q^2)) \\
&= \sum_{x=1}^{2^{q-1}} \hbar_q(\theta_2(G_q(1,1,x,2^q)), \partial_1(Z_q^2)) \quad \text{(overlapping with } Q_1(Z_q^2), Q_3(Z_q^2) \text{ only)} \\
&\quad + \sum_{x=2^{q-1}+1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,2^q)), \partial_1(Z_q^2)) \quad \text{(overlapping with four quadrants)} \\
&= \sum_{x=1}^{2^{q-1}} \hbar_q(\theta_2(G_q(1,2^{q-1}+1,x,2^q)), \partial_1(Z_q^2)) \quad \text{(Remark 6.2)} \\
&\quad + \sum_{x=2^{q-1}+1}^{2^q} \hbar_q(\theta_2(G_q(2^{q-1}+1,2^{q-1}+1,x,2^q)), \partial_1(Z_q^2)) \quad \text{(Remark 6.2)} \\
&= \sum_{x=1}^{2^{q-1}} (\hbar_q(\theta_2(G_q(1,2^{q-1}+1,x,2^q)), \partial_1(Q_3(Z_q^2))) + \hbar_q(\partial_1(Q_3(Z_q^2), \partial_1(Z_q^2))) \\
&\quad + \sum_{x=2^{q-1}+1}^{2^q} (\hbar_q(\theta_2(G_q(2^{q-1}+1,2^{q-1}+1,x,2^q)), \partial_1(Q_4(Z_q^2))) \\
&\qquad + \hbar_q(\partial_1(Q_4(Z_q^2), \partial_1(Z_q^2))) \\
&= \sum_{x=1}^{2^{q-1}} (\hbar_{q-1}(\theta_2(G_{q-1}(1,1,x,2^{q-1})), \partial_1(Z_{q-1}^2)) + 2(2^{q-1})^2) \\
&\quad + \sum_{x=1}^{2^{q-1}} (\hbar_{q-1}(\theta_2(G_{q-1}(1,1,x,2^{q-1})), \partial_1(Z_{q-1}^2)) + 3(2^{q-1})^2) \\
&\quad (Q_3(Z_k^2), Q_4(Z_q^2)\text{: canonical } Z_{q-1}^2) \\
&= \overline{\Pi}_{q-1}^B + 2^{q-1} \cdot 2(2^{q-1})^2 + \overline{\Pi}_{q-1}^B + 2^{q-1} \cdot 3(2^{q-1})^2 \\
&= 2\overline{\Pi}_{q-1}^B + 5(2^{q-1})^3.
\end{aligned}
$$

The proof of $\overline{\Pi}_q^L$ is similar to that of $\overline{\Pi}_q^B$. ∎

The closed-form solutions for $\overline{\Pi}_q$ are employed to establish a system of recurrences for $\overline{\Omega}_{q,2^q}^c$.

Lemma 6.8 *For a canonical Z_q^2,*

$$
\overline{\Omega}_{q,2^q}^{c_1} = \begin{cases} 4\overline{\Omega}_{q-1,2^{q-1}}^{c_1} + 6(2^{q-1})^4 & \textit{if } q > 1, \\ 6 & \textit{if } q = 1. \end{cases}
$$

Proof. As in Figure 6.11 and the case discussion for $\Omega^{c_1}_{q,2^q}$, we can split $\overline{\Omega}^{c_1}_{q,2^q}$ into four parts by similar way:

$$\overline{\Omega}^{c_1}_{q,2^q} = \sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,y)), \partial_1(Z_q^2))$$

$$= \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} \hbar_q(\theta_2(G_q(1,1,x,y), \partial_1(Z_q^2)) \quad \text{(Figure 6.11(a))}$$

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} \hbar_q(\theta_2(G_q(1,1,x,y), \partial_1(Z_q^2)) \quad \text{(Figure 6.11(b))}$$

$$+ \sum_{x=1}^{2^{q-1}}\sum_{y=2^{q-1}+1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,y), \partial_1(Z_q^2)) \quad \text{(Figure 6.11(d))}$$

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} \hbar_q(\theta_2(G_q(1,1,x,y), \partial_1(Z_q^2)) \quad \text{(Figure 6.11(c))}$$

$$= \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} (\hbar_q(\theta_2(G_q(1,1,x,y)), \partial_1(Q_1(Z_q^2))) + \hbar_q(\partial_1(Q_1(Z_q^2)), \partial_1(Z_q^2)))$$

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=1}^{2^{q-1}} (\hbar_q(\theta_2(G_q(1,1,x,y)), \partial_1(Q_2(Z_q^2))) + \hbar_q(\partial_1(Q_2(Z_q^2)), \partial_1(Z_q^2)))$$

$$+ \sum_{x=1}^{2^{q-1}}\sum_{y=2^{q-1}+1}^{2^q} (\hbar_q(\theta_2(G_q(1,1,x,y)), \partial_1(Q_3(Z_q^2))) + \hbar_q(\partial_1(Q_3(Z_q^2)), \partial_1(Z_q^2)))$$

$$+ \sum_{x=2^{q-1}+1}^{2^q}\sum_{y=2^{q-1}+1}^{2^q} (\hbar_q(\theta_2(G_q(1,1,x,y)), \partial_1(Q_4(Z_q^2))) + \hbar_q(\partial_1(Q_4(Z_q^2)), \partial_1(Z_q^2)))$$

$$= \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} (\hbar_{q-1}(\theta_2(G_{q-1}(1,1,x,y)), \partial_1(Z_{q-1}^2)) + 0\cdot(2^{q-1})^2)$$

$$+ \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} (\hbar_{q-1}(\theta_2(G_{q-1}(1,1,x,y)), \partial_1(Z_{q-1}^2)) + 1\cdot(2^{q-1})^2)$$

$$+ \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} (\hbar_{q-1}(\theta_2(G_{q-1}(1,1,x,y)), \partial_1(Z_{q-1}^2)) + 2\cdot(2^{q-1})^2)$$

$$+ \sum_{x=1}^{2^{q-1}}\sum_{y=1}^{2^{q-1}} (\hbar_{q-1}(\theta_2(G_{q-1}(1,1,x,y)), \partial_1(Z_{q-1}^2)) + 3\cdot(2^{q-1})^2)$$

$$= \overline{\Omega}^{c_1}_{q-1,2^{q-1}} + \overline{\Omega}^{c_1}_{q-1,2^{q-1}} + (2^{q-1})^4$$

$$
\begin{aligned}
&+\overline{\Omega}^{c_1}_{q-1,2^{q-1}} + 2(2^{q-1})^4 + \overline{\Omega}^{c_1}_{q-1,2^{q-1}} + 3(2^{q-1})^4 \\
= \;& 4\overline{\Omega}^{c_1}_{q-1,2^{q-1}} + 6(2^{q-1})^4.
\end{aligned}
$$

▮

The closed-form solutions for $\overline{\Omega}^{c}_{q,2^q}$ and $\overline{\Pi}_q$ are employed to obtain exact formulas for $\Omega^c_{k,2^q}$.

Lemma 6.9 *For a canonical Z^2_k structured as an Z^2_{k-q}-curve interconnecting $2^{2(k-q)}$ Z^2_q-subcurves,*

$$\Omega^{c_1}_{k,2^q} \;=\; \overline{\Omega}^{c_1}_{q,2^q} - \overline{\Pi}^B_q - \overline{\Pi}^L_q + (2^{2q} - 1).$$

Proof. By following the definition, we have

$$
\begin{aligned}
\Omega^{c_1}_{k,2^q} &= \sum_{x=1}^{2^q-1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z^2_k)) \\
&= \sum_{x=1}^{2^q}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z^2_k)) \\
&\quad - \sum_{x=2^q}^{2^q}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z^2_k)) \\
&= (\sum_{x=1}^{2^q}\sum_{y=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z^2_k)) \\
&\qquad - \sum_{x=1}^{2^q}\sum_{y=2^q}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z^2_k))) \\
&\quad -(\sum_{x=2^q}^{2^q}\sum_{y=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z^2_k)) \\
&\qquad - \sum_{x=2^q}^{2^q}\sum_{y=2^q}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z^2_k))) \\
&= \overline{\Omega}^{c_1}_{q,2^q} \\
&\quad - \sum_{x=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,x,2^q)), \partial_1(Z^2_k)) \\
&\quad - \sum_{y=1}^{2^q} \hbar_k(\theta_2(G_k(1,1,2^q,y)), \partial_1(Z^2_k))
\end{aligned}
$$

$$+\hbar_k(\theta_2(G_k(1,1,2^q,2^q)),\partial_1(Z_k^2))) \qquad (\theta_2(G(1,1,2^q,2^q))=\partial_2(Z_q^2))$$
$$= \overline{\Omega}^{c_1}_{q,2^q} - \overline{\Pi}^B_q - \overline{\Pi}^L_q + (2^{2q}-1).$$

∎

The exact formulas for $\Omega^c_{k,2^q}$ are employed to establish a system of recurrences for $\Omega_{k,2^q}$.

Lemma 6.10 *For a canonical Z_k^2 structured as an Z_{k-q}^2-curve interconnecting $2^{2(k-q)}$ Z_q^2-subcurves,*

$$\Omega^L_{k,2^q} = \begin{cases} \Omega^L_{k-1,2^q} + (\Omega^{c_1}_{k-1,2^q} + (2^q-1)^2(2^{k-1})^2) \\ \quad + (\Omega^L_{k-1,2^q} + (2^{k-1}-2^q+1)(2^q-1)(2^{k-1})^2) & \text{if } k>q, \\ \overline{\Pi}^L_q - (2^{2q}-1) & \text{if } k=q; \end{cases}$$

$$\Omega^B_{k,2^q} = \begin{cases} \Omega^B_{k-1,2^q} + (\Omega^{c_1}_{k-1,2^q} + 2(2^q-1)^2(2^{k-1})^2) \\ \quad + (\Omega^B_{k-1,2^q} + 2(2^{k-1}-2^q+1)(2^q-1)(2^{k-1})^2) & \text{if } k>q, \\ \overline{\Pi}^B_q - (2^{2q}-1) & \text{if } k=q. \end{cases}$$

Proof. Similar to the proof of Lemma 6.9, from the definition, we have (see Figure 6.9)

$$\Omega^L_{k,2^q} = \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)),\partial_1(Z_k^2))$$
$$= \sum_{x=1}^{2^{k-1}-2^q+1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)),\partial_1(Z_k^2)) \qquad (\text{in } Q_1(Z_k^2))$$
$$+ \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)),\partial_1(Z_k^2))$$
$$(\text{across } Q_1(Z_k^2), Q_2(Z_k^2))$$
$$+ \sum_{x=2^{k-1}+1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)),\partial_1(Z_k^2)) \qquad (\text{in } Q_2(Z_k^2))$$
$$= \sum_{x=1}^{2^{k-1}-2^q+1}\sum_{y=1}^{2^q-1} (\hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)),\partial_1(Q_1(Z_k^2)))$$
$$+ \hbar_k(\partial_1(Q_1(Z_k^2)),\partial_1(Z_k^2)))$$
$$+ \sum_{x=2^{k-1}-2^q+2}^{2^{k-1}}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(2^{k-1}+1,1,x+2^q-1,y)),\partial_1(Z_k^2))$$

$$
\begin{aligned}
& \text{(Remark 6.2)} \\
& \sum_{x=2^{k-1}+1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} (\hbar_k(\theta_2(G_k(x,1,x+2^q-1,y)), \partial_1(Q_2(Z_k^2))) \\
& \quad + \hbar_k(\partial_1(Q_2(Z_k^2)), \partial_1(Z_k^2))) \\
= & \sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} (\hbar_{k-1}(\theta_2(G_{k-1}(x,1,x+2^q-1,y)), \partial_1(Z_{k-1}^2)) + 0 \cdot (2^{k-1})^2) \\
& + \sum_{x=2^{k-1}+1}^{2^{k-1}+2^q-1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(2^{k-1}+1,1,x,y)), \partial_1(Z_k^2)) \qquad \text{(change index of } x) \\
& + \sum_{x=1}^{2^{k-1}-2^q+1} \sum_{y=1}^{2^q-1} (\hbar_{k-1}(\theta_2(G_{k-1}(x,1,x+2^q-1,y)), \partial_1(Z_{k-1}^2)) + 1 \cdot (2^{k-1})^2) \\
= & \ \Omega^L_{k-1,2^q} + 0 \\
& + \sum_{x=2^{k-1}+1}^{2^{k-1}+2^q-1} \sum_{y=1}^{2^q-1} (\hbar_k(\theta_2(G_k(2^{k-1}+1,1,x,y)), \partial_1(Q_2(Z_k^2))) \\
& \quad + \hbar_k(\partial_1(Q_2(Z_k^2)), \partial_1(Z_k^2))) \\
& + \Omega^L_{k-1,2^q} + 1 \cdot (2^{k-1})^2(2^q-1)(2^{k-1}-2^q+1) \\
= & \ \Omega^L_{k-1,2^q} + (\Omega^L_{k-1,2^q} + (2^{k-1})^2(2^q-1)(2^{k-1}-2^q+1)) \\
& + \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} (\hbar_{k-1}(\theta_2(G_{k-1}(1,1,x,y)), \partial_1(Z_{k-1}^2)) + 1 \cdot (2^{k-1})^2) \\
= & \ \Omega^L_{k-1,2^q} + (\Omega^{c_1}_{k-1,2^q} + (2^q-1)^2(2^{k-1})^2) \\
& + (\Omega^L_{k-1,2^q} + (2^{k-1})^2(2^q-1)(2^{k-1}-2^q+1)).
\end{aligned}
$$

For $\Omega^B_{k-1,2^q}$, the proof is similar to this one. ∎

6.3.2 Computing $\sum \hbar_k(\theta_1(G), \partial_1(Z_k^2))$ over Subgrids G Overlapping with Two Quadrants

We may proceed as in Section 6.3.1, based upon the following system of summations $\omega_{k,2^q} = (\omega^L_{k,2^q}, \omega^R_{k,2^q}, \omega^B_{k,2^q}, \omega^T_{k,2^q})$:

$$
\omega^L_{k,2^q} = \sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \hbar(\theta_1(G_k(x,1,x+2^q-1,y)), \partial_1(Z_k^2)) \text{—for left boundary,}
$$

$$\omega^R_{k,2^q} = \sum_{x=1}^{2^k-2^q+1} \sum_{y=2^k-2^q+2}^{2^k} \hbar(\theta_1(G_k(x,y,x+2^q-1,2^k)),\partial_1(Z_k^2))\text{—for right boundary,}$$

$$\omega^B_{k,2^q} = \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^k-2^q+1} \hbar(\theta_1(G_k(1,y,x,y+2^q-1)),\partial_1(Z_k^2))\text{—for bottom boundary,}$$

$$\omega^T_{k,2^q} = \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=1}^{2^k-2^q+1} \hbar(\theta_1(G_k(x,y,2^k,y+2^q-1)),\partial_1(Z_k^2))\text{—for top boundary.}$$

Or, we apply the following lemma to relate the two systems $\omega_{k,2^q}$ and $\Omega_{k,2^q}$. As mentioned before, the system need only the computations for $\theta_1(G)$ in right side or top side in a quadrant. We consider $\omega^R_{k,2^q}$, $\omega^T_{k,2^q}$ in the following lemma.

Lemma 6.11 *For a canonical Z_k^2,*

$$\begin{aligned}\omega^R_{k,2^q} + \Omega^L_{k,2^q} &= (2^{2k}-1)\mathcal{N}^S_{k,2^q},\\ \omega^T_{k,2^q} + \Omega^B_{k,2^q} &= (2^{2k}-1)\mathcal{N}^S_{k,2^q}.\end{aligned}$$

Proof. A canonical Z_k^2 is a reflexive. (After top-down and then left-right reflect Z_k^2, we get the one of same structure.) For a grid point $v = (x,y)$, its mirror point $v' = (2^k+1-x, 2^k+1-y)$, and the mirror pair (v,v') satisfies that:

$$\hbar_k(v,\partial_1(Z_k^2)) + \hbar_k(v',\partial_1(Z_k^2)) = \hbar_k(v,\partial_2(Z_k^2)) + \hbar_k(v',\partial_2(Z_k^2)) = 2^{2k}-1.$$

Top-down and then left-right reflect $G_k(x,1,x+2^q-1,y)$, where $y \in [2^q]$ and $q < k$, we get $G_k(2^k-2^q+2-x, 2^k+1-y, 2^k+1-x, 2^k)$, and the reflection of the lowest indexed point in $G_k(x,1,x+2^q-1,y)$ is the highest indexed in $G_k(2^k-2^q+2-x, 2^k+1-y, 2^k+1-x, 2^k)$; that is, $\theta_1(G_k(x,1,x+2^q-1,y))$ and $\theta_2(G_k(2^k-2^q+2-x, 2^k+1-y, 2^k+1-x, 2^k))$ are mirror pair. Thus,

$$\begin{aligned}&\omega^L_{k,2^q} + \Omega^R_{k,2^q}\\ = &\sum_{x=1}^{2^k-2^q+1} \sum_{y=1}^{2^q-1} \hbar(\theta_1(G_k(x,1,x+2^q-1,y)),\partial_1(Z_k^2))\\ &+ \sum_{x=1}^{2^k-2^q+1} \sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_2(G_k(x,y,x+2^q-1,2^k)),\partial_1(Z_k^2))\end{aligned}$$

$$= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar(\theta_1(G_k(x,1,x+2^q-1,y)),\partial_1(Z_k^2))$$
$$+ \sum_{\gamma=2^k-2^q+1}^{1}\sum_{\gamma'=2^q-1}^{1} \hbar_k(\theta_2(G_k(2^k-2^q+2-\gamma,2^k+1-\gamma',2^k+1-\gamma,2^k)),\partial_1(Z_k^2))$$

(change of summation index: $\gamma = 2^k - 2^q + 2 - x, \gamma' = 2^k + 1 - y$)

$$= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar(\theta_1(G_k(x,1,x+2^q-1,y)),\partial_1(Z_k^2))$$
$$+ \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(2^k-2^q+2-x,2^k+1-y,2^k+1-x,2^k)),\partial_1(Z_k^2))$$

(change of summation index: $x = \gamma, y = \gamma'$)

$$= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} (\hbar(\theta_1(G_k(x,1,x+2^q-1,y)),\partial_1(Z_k^2))$$
$$+ \hbar_k(\theta_2(G_k(2^k-2^q+2-x,2^k+1-y,2^k+1-x,2^k)),\partial_1(Z_k^2)))$$
$$= \sum_{x=1}^{2^k-2^q+1}\sum_{y=1}^{2^q-1} (2^{2k}-1) \quad \text{(mirror pair)}$$
$$= (2^{2k}-1)\mathcal{N}^S_{k,2^q}.$$

The proof for $\omega^T_{k,2^q} + \Omega^B_{k,2^q}$ is similar to this one. ∎

6.3.3 Query Subgrids Overlapping with All Quadrants

For a $2^q \times 2^q$ query subgrid $G \subseteq \mathcal{R}$, we have: (1) $\theta_2(G) \in Q_4(Z_k^2)$ and (2) $\theta_1(G) \in Q_1(Z_k^2)$ by Remark 6.2.

For (1), when zooming in on the incomplete rectangular subgrid $G \cap Q_4(Z_k^2)$ (with both side-lengths at most $2^q - 1$), we reduce $\sum_{\text{all} G \subseteq \mathcal{R}} \hbar_k(\theta_2(G), \partial_1(Z_k^2))$ to $\Omega^{c_1}_{k-1,2^q}$ (with adjustment of distance cumulation).

For (2), similar consideration leads to a reduction of $\sum_{\text{all} G \subseteq \mathcal{R}} \hbar_k(\theta_1(G), \partial_1(Z_k^2))$ to $\omega^{c_4}_{k-1,2^q}$, where $\omega^{c_4}_{k,2^q}$ denotes $\sum_{x=2^k-2^q+2}^{2^k}\sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_1(G_k(x,y,2^k,2^k)),\partial_1(Z_k^2))$ for a canonical Z_k^2 and is related to $\Omega^{c_1}_{k,2^q}$ as follows.

Lemma 6.12 *For a canonical Z_k^2, $\omega^{c_4}_{k,2^q} + \Omega^{c_1}_{k,2^q} = (2^{2k}-1)\mathcal{N}^c_{k,2^q}$.*

Proof. Similar to the proof of Lemma 6.11,

$$
\begin{aligned}
& \omega^{c_4}_{k,2^q} + \Omega^{c_1}_{k,2^q} \\
= & \sum_{x=2^k-2^q+2}^{2^k} \sum_{y=2^k-2^q+2}^{2^k} \hbar_k(\theta_1(G_k(x,y,2^k,2^k)), \partial_1(Z_k^2)) \\
& + \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z_k^2)) \\
= & \sum_{\gamma=2^q-1}^{1} \sum_{\gamma'=2^q-1}^{1} \hbar_k(\theta_1(G_k(2^k+1-\gamma, 2^k+1-\gamma', 2^k, 2^k)), \partial_1(Z_k^2)) \\
& \qquad \text{((change of summation index: } \gamma = 2^k+1-x, \gamma' = 2^k+1-y) \\
& + \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z_k^2)) \\
= & \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_1(G_k(2^k+1-x, 2^k+1-y, 2^k, 2^k)), \partial_1(Z_k^2)) \\
& \qquad \text{(change of summation index: } x = \gamma, y = \gamma') \\
& + \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z_k^2)) \\
= & \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} (\hbar_k(\theta_1(G_k(2^k+1-x, 2^k+1-y, 2^k, 2^k)), \partial_1(Z_k^2)) \\
& \quad + \hbar_k(\theta_2(G_k(1,1,x,y)), \partial_1(Z_k^2))) \\
= & \sum_{x=1}^{2^q-1} \sum_{y=1}^{2^q-1} (2^{2k}-1) \qquad \text{(mirror pair)} \\
= & (2^{2k}-1)\mathcal{N}^c_{k,2^q}.
\end{aligned}
$$

∎

6.3.4 The Big Picture: Computing $\Psi_q(Z_k^2)$

The results in the previous three subsections yield $\varepsilon_{k,q}(Z_k^2)$. Hence, we have the following theorem for $\Psi_q(Z_k^2)$:

Theorem 6.2 *For a canonical Z_k^2, the recurrence for summation of all inter-cluster* *distances over all $2^q \times 2^q$ query subgrids of an Z_k^2-structural grid space $[2^k]^2$ is:*

$$\Psi_q(Z_k^2) = \begin{cases} 4\Psi_q(Z_{k-1}^2) + (\Omega_{k-1,2^q}^B + 2^{2k-2}\mathcal{N}_{k-1,2^q}^S) \\ \quad - \omega_{k-1,2^q}^T - (2^{2q}-1)\mathcal{N}_{k-1,2^q}^S \\ \quad + (\Omega_{k-1,2^q}^L + 3\cdot 2^{2k-2}\mathcal{N}_{k-1,2^q}^S) \\ \quad - (\omega_{k-1,2^q}^R + 2^{2k-2}\mathcal{N}_{k-1,2^q}^S) - (2^{2q}-1)\mathcal{N}_{k-1,2^q}^S \\ \quad + (\Omega_{k-1,2^q}^B + 3\cdot 2^{2k-2}\mathcal{N}_{k-1,2^q}^S) \\ \quad - (\omega_{k-1,2^q}^T + 2\cdot 2^{2k-2}\mathcal{N}_{k-1,2^q}^S) - (2^{2q}-1)\mathcal{N}_{k-1,2^q}^S \\ \quad + (\Omega_{k-1,2^q}^L + 2\cdot 2^{2k-2}\mathcal{N}_{k-1,2^q}^S) \\ \quad - (\omega_{k-1,2^q}^R) - (2^{2q}-1)\mathcal{N}_{k-1,2^q}^S \\ \quad + (\Omega_{k-1,2^q}^{c_1} + 3\cdot 2^{2k-2}\mathcal{N}_{k-1,2^q}^c) \\ \quad - (\omega_{k-1,2^q}^{c_4}) - (2^{2q}-1)\mathcal{N}_{k-1,2^q}^c & \textit{if } k > q, \\ 0 & \textit{if } k = q. \end{cases}$$

The exact formula for $\Psi_q(Z_k^2)$ is:

$$\begin{aligned} \Psi_q(Z_k^2) &= 2^{3k+q} - 2^{3k} - 2^{2k+2q+1} + 2^{2k+q+1} + 2^{k+3q} - 2^{k+q+1} + 2^k \\ &\quad -2^{3q} + 2^{2q+1} - 2^q \end{aligned}$$

6.3.5 Total Number of Inter-cluster Gaps

In order to compute the (universe) mean inter-cluster distance over all inter-cluster gaps from all identically shaped subgrids, we need to derive the total number of inter-cluster gaps, denoted by $\Phi_{k,q}(Z_k^2)$ for a canonical Z_k^2.

In Chapter V, we have derived the exact formula for $E_{k,q}(Z_k^2)$, which denotes the total number of edges cut all $2^q \times 2^q$ query subgrids, and the closed-form solution for $E_{k,q}(Z_k^2)$ is

$$\begin{aligned} E_q(Z_k^2) &= 2^{2k+q+2} - 2^{2k+2} + 3\cdot 2^{2k-q} - 2^{2k-2q} - 2^{k+2q+3} + 3\cdot 2^{k+q+2} - 2^{k+3} \\ &\quad +2^{k-q+2} + 2^{3q+2} - 2^{2q+3} + 2^{q+2}. \end{aligned}$$

Hence, the total number of inter-cluster gaps:

$$\begin{aligned} \Phi_{k,q}(Z_k^2) &= \frac{E_{k,q}(Z_k^2)}{2} - (2^k - 2^q + 1)^2 \\ &= 2^{2k+q+1} - 3\cdot 2^{2k} + 3\cdot 2^{2k-q-1} - 2^{2k-2q-1} - 2^{k+2q+2} + 2^{k+q+3} \\ &\quad -3\cdot 2^{k+1} + 2^{k-q+1} + 2^{3q+1} - 5\cdot 2^{2q} + 2^{q+2} - 1. \end{aligned}$$

6.4 Comparisons and Validation

For a space-filling curve C_k^2 indexing the grid space $[2^k]^2$, denote by $\Delta_{k,q}(C_k^2)$ the universe mean inter-cluster distance over all inter-cluster gaps from all $2^q \times 2^q$ subgrids of the C_k^2-structural grid space, and by $\widetilde{\Delta}_{k,q}(C_k^2)$ the mean total inter-cluster distance over all $2^q \times 2^q$ subgrids of the C_k^2-structural grid space.

The exact formulas for $\Psi_q(H_k^2)$, $\Phi_{k,q}(H_k^2)$, $\Psi_q(Z_k^2)$, and $\Phi_{k,q}(Z_k^2)$ give the exact formulas for $\Delta_{k,q}(H_k^2)$, $\Delta_{k,q}(Z_k^2)$, $\widetilde{\Delta}_{k,q}(H_k^2)$, and $\widetilde{\Delta}_{k,q}(Z_k^2)$. We simplify the exact results asymptotically as follows. For sufficiently large k and q with $k \gg q$ (typical scenario for range queries),

$$\Delta_{k,q}(C_k^2) \approx \begin{cases} \frac{17}{14} \cdot 2^k & \text{if } C_k^2 \text{ is } H_k^2, \\ \frac{1}{2} \cdot 2^k & \text{if } C_k^2 \text{ is } Z_k^2; \end{cases} \qquad \widetilde{\Delta}_{k,q}(C_k^2) \approx \begin{cases} \frac{17}{14} \cdot 2^{k+q} & \text{if } C_k^2 \text{ is } H_k^2, \\ 2^{k+q} & \text{if } C_k^2 \text{ is } Z_k^2; \end{cases}$$

$$\frac{\Delta_{k,q}(H_k^2)}{\Delta_{k,q}(Z_k^2)} \approx \frac{17}{7} \approx 2.43, \qquad \frac{\widetilde{\Delta}_{k,q}(H_k^2)}{\widetilde{\Delta}_{k,q}(Z_k^2)} \approx \frac{17}{14} \approx 1.21.$$

With respect to the $\Delta_{k,q}$-statistics, the z-order curve family clearly performs better than the Hilbert curve family over the considered ranges for k and q. With respect to the $\widetilde{\Delta}_{k,q}$-statistics, the superiority of z-order curve family persists but declines significantly.

We have validated all the exact formulas (intermediate and final) involved in the derivations in the analytical study with computer programs over various grid- and subgrid-orders: $k \in \{3, 4, \dots, 10\}$ and $q \in \{2, 3, \dots, k\}$.

6.5 Summary

Our analytical study of the inter-clustering performances of 2-dimensional order-k Hilbert and z-order curve families are based upon the two inter-clustering statistics $\Delta_{k,q}$ and $\widetilde{\Delta}k, q$ — universe mean inter-cluster distance over all inter-cluster gaps and mean total inter-cluster distance over all subgrids of size $2^q \times 2^q$, respectively. The exact results allow us to compare their relative performances with respect to these two measures. For sufficiently large k and q with $k \gg q$, z-order curve family performs significantly (marginally) better than Hilbert curve family with respect to $\Delta_{k,q}$-statistics ($\widetilde{\Delta}_{k,q}$-statistics, respectively). We also validate the results with computer programs over various grid- and subgrid-orders.

CHAPTER VII

CONCLUSION

Evaluating either locality preservation or clustering performance is considered as a criterion as far as the applicabilities of space-filling curves are concerned. In this research, we focus on the relevant work for these two categories of measures. The objectives of this research are to (1) investigate the locality preservation and clustering performances for space-filling curves, especially the most popular ones, z-order and Hilbert space-filling curve families, (2) derive the closed-form formulas for different measures to quantify the qualities of space-filling curves, and (3) compare the z-order curve family with the Hilbert curve family by the derived formulas.

For the research related to locality preservation, we propose a new locality measure L_δ that quantifies the locality preservation of space-filling curves by considering the mean absolute index-difference for two grid points at a common Manhattan distance in multi-dimensional space (quantifying the applicabilities of the space-filling curves is the initial step in the whole work for evaluating their performances). Then comparisons have been made between Hilbert and z-order curve families. Basically, we have derived the exact formulas for $L_\delta(H_k^m)$ and $L_\delta(Z_k^m)$ for m=2 and arbitrary δ that is an integral power of 2, and m=3 and $\delta = 1$. The results obtained from the exact formulas allow us to gauge the two curve families relative to the optimal curves with respect to L_δ, and indicate that the z-order curve family performs better than the Hilbert curve family over the considered ranges of dimension, grid-order, and 1-normed distance. Besides, we have verified all the exact formulas (intermediate and final) involved in the derivations with computer programs for $m = 2, 3$ and over various grid-orders and all possible 1-normed distances.

In addition to proposing a new locality measures, we have closed the bounds of the well-known p-normed metric measure for the 2-dimensional Hilbert curve family. For p-normed metric measure, the analytical study of the locality properties is constructed on the locality measure $L_{\mathrm{AN},p}$, which is the maximum ratio of $d_p(v,u)^m$ to $d_p(\tilde{v},\tilde{u})$ over all corresponding point-pairs (v,u) and $(\tilde{v},\tilde{u})$ in the m-dimensional grid space and index space, respectively. Regarding this locality measure of the Hilbert curve family, our work merges the current best lower and upper bounds to exact formulas for $p \in \{1,2\}$, and also extend our work to all real $p \geq 2$. Moreover, we identify all the representative pairs (which realize $L_{\mathrm{AN},p}(H_k^2)$) for $p = 1$ and all real $p \geq 2$ and also validate the results with computer programs over various p-values.

In the part of research related to clustering performance, there are two kinds of measures that have been developed. One is built on the mean number of clusters within a subgrid; the other one is built on the mean inter-cluster distance within a subgrid. We have derived the exact formulas for the mean numbers of clusters and the mean inter-cluster distances within subspaces for a 2-dimensional space.

Our first analytical study of the clustering performances of 2-dimensional order-k Hilbert and z-order curve families are based upon clustering statistics — a mean number of clusters over all subgrids of size $2^q \times 2^q$, respectively. The quantified results allow us to compare the relative performances of these two families. For sufficiently large k and q with $k \gg q$, the Hilbert curve family performs significantly better than the z-order curve family with respect to the mean numbers of clusters within subspaces.

The other analytical study of the clustering performances are based upon the two inter-clustering statistics — universe mean inter-cluster distance over all inter-cluster gaps and mean total inter-cluster distance over all subgrids of size $2^q \times 2^q$. Similarly, the results quantified by our formulas allow us to compare their relative performances as well: for sufficiently large k and q with $k \gg q$, the z-order curve family performs significantly (marginally) better than the Hilbert curve family in this respect. We also validate the results with computer programs over various grid- and subgrid-orders.

In summary, we study both locality preservation and clustering performances by proposing various measures and also providing their derivations of exact formulas for the Hilbert and z-order curve families. Having closed-form formulas allows us to compare the relative performances of locality or clustering between these two space-filling curve families. A practical implication of our results is that the exact formulas provide good bounds on measuring the loss in data locality in the index space, while spatial correlation exists in the 2-dimensional grid space. For higher dimensionality, it becomes much more difficult due to the loss of geometric intuition. Nevertheless, Alber and Niedermeier [AN00] provide a mathematical mechanism to describe and analyze the combinatorial properties of continuous curves such as the Hilbert curves and non-continuous ones such as z-order curves in arbitrary dimensions. Their structure-theoretic viewpoint may shed some light on our future study with arbitrary dimensions.

BIBLIOGRAPHY

[ACE+02] W. Aref, A.C. Catlin, A. Elmagarmid, J. Fan, M. Hammad, I.F. Ilyas, M.S. Marzouk, and X.Q. Zhu. A video database management system for advanding video database research. In *Proceedings of the Internationl Workshop on Multimedia Information Systems*, 2002.

[AE99] P. Agarwal and J. Erickson. Geometric range searching and its relatives. In *Advances in Discrete and Computational Geometry*, volume 23 of *Contemporary Mathematics*, pages 1–56, 1999.

[Alb97] J. Alber. Locality properties of discrete space-filling curves: Results with relevance for computer science (in German). Studienarbeit Universität Tübingen, Wilhelm-Schickard-Institut für Informatik. July 1997.

[AM90] D. J. Abel and D. M. Mark. A comparative analysis of some two-dimensional orderings. *International Journal of Geographical Information Systems*, 4(1):21–31, January 1990.

[AN98] J. Alber and R. Niedermeier. On multi-dimensional Hilbert indexings. In *Computing and Combinatorics*, volume 1449 of *Lecture Notes in Computer Science*, pages 329–338, August 1998.

[AN00] J. Alber and R. Niedermeier. On multi-dimensional curves with Hilbert property. *Theory of Computing Systems*, 33(4):295–312, 2000.

[ARR+97] T. Asano, D. Ranjan, T. Roos, E. Welzl, and P. Widmayer. Space-filling curves and their use in the design of geometric data structures. *Theoretical Computer Science*, 181(1):3–15, 1997.

[BBB+97] S. Berchtold, C. Böhm, B. Braunmüller, D. A. Keim, and H.-P. Kriegel. Fast parallel similarity search in multimedia databases. In *Proceedings of the 1997 ACM SIGMOD international conference on Management of data*, pages 1–12, 1997.

[BBK01] C. Böhm, S. Berchtold, and D. A. Keim. Searching in high-dimensional spaces — index structures for improving the performance of multimedia databases. *ACM Computing Surveys*, 33(3):322–373, 2001.

[BC98] G. Breinholt and C.Schierz. Algorithm 781: Generating Hilbert's space-filling curve by recursion. *ACM Transactions on Mathematical Software*, 24(2):184–189, June 1998.

[BKK99] C. Böhm, G. Klump, and H.-P. Kriegel. XZ-ordering: A space-filling curve for objects with spatial extension. In *Spatial Databases, the Sixth International Symposium*, volume 1651 of *Lecture Notes in Computer Science*, pages 75–90. Springer, July 1999.

[BP82] J. J. Bartholdi and L. K. Platzman. An $O(n \log n)$ planar traveling salesman heuristic based on spacefilling curves. *Operations Research Letters*, 1(4):121–125, 1982.

[BRWW97] E. Bugnion, T. Roos, R. Wattenhofer, and P. Widmayer. Space filling curves versus random walks. In van Kreveld, Nievergelt, Roos, and Widmayer, editors, *Algorithmic Foundations of Geographic Information Systems*, volume 1340 of *Lecture Notes in Computer Science*, pages 199–211, Springer-Verlag, Berlin Heidelberg, 1997.

[But69] A. R. Butz. Convergence with Hilbert's space filling curve. *Journal of Computer and System Sciences*, 3(2):128–146, May 1969.

[FR89] C. Faloutsos and S. Roseman. Fractals for secondary key retrieval. In *Proceedings of the eighth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 247–252, 1989.

[GG98] V. Gaede and O. Günther. Multidimensional access methods. *ACM Computing Surveys*, 30(2):170–231, 1998.

[GL96] C. Gotsman and M. Lindenbaum. On the metric properties of discrete space-filling curves. *IEEE Transactions on Image Processing*, 5(5):794–797, 1996.

[Har64] L. H. Harper. Optimal assignments of numbers to vertices. *Journal of the Society for Industrial and Applied Mathematics*, 12(1):131–135, 1964.

[HW79] D. H. Hubel and T. N. Wiesel. Brain mechanisms of vision. *Scientific American*, 241(3):150–162, 1979.

[HW02] J. Hungershöfer and J. M. Wierum. On the quality of partitions based on space filling curves. In *Proceedings of the International Conference on Computational Science*, pages 36–45, 2002.

[Jag90] H. V. Jagadish. Linear clustering of objects with multiple attributes. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 332–342. Association for Computing Machinery, May 1990.

[Jag97] H. V. Jagadish. Analysis of the Hilbert curve for representing two-dimensional space. *Information Processing Letters*, 62(1):17–22, 1997.

[KOR95] M. Kaddoura, C.-W. Ou, and S. Ranka. Partitioning unstructured computational graphs for non-uniform and adaptive environments. *IEEE Parallel and Distributed Technology*, 3(3):63–69, 1995.

[LK00] J. K. Lawder and P. J. H. King. Querying multi-dimensional data indexed using the Hilbert space-filling curve. Research Report JL3/00, School of Computer Science and Information Systems, Birkbeck College, University of London. 2000.

[LK01] J. K. Lawder and P. J. H. King. Using state diagrams for Hilbert curve mappings. *International Journal of Computer Mathematics*, 78(3):327–342, 2001.

[LLL01] S. Liao, M. A. Lopez, and S. T. Leutenegger. High dimensional similarity search with space filling curves. In *Proceedings of the International Conference on Data Engineering*, pages 615–622. IEEE Computer Society, 2001.

[LS94] W. M. Lam and J. M. Shapiro. Class of fast algorithms for the Peano-Hilbert space-filling curve. In *Proceedings of the International Conference on Image Processing*, pages 636–641. IEEE Computer Society, November 1994.

[LS97] X. Liu and G. F. Schrack. An algorithm for encoding and decoding the 3-D Hilbert order. *IEEE Transactions on Image Processing*, 6(9):1333–1337, 1997.

[LZ84] A. Lempel and J. Ziv. Compression of two-dimensional images. In A. Apostolico and Z. Galil, editors, *Combinatorial Algorithms on Words*, pages 141–156, ASI Series, Volume F12, Springer-Verlag, Berlin Heidelberg, 1984.

[MA01] M. F. Mokbel and W. G. Aref. Irregularity in multi-dimensional space-filling curves with applications in multimedia databases. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 512–519, November 2001.

[MAK02] M. F. Mokbel, W. G. Aref, and I. Kamel. Performance of multi-dimensional space-filling curves. In *Proceedings of the tenth ACM international symposium on Advances in geographic information systems*, pages 149–154, 2002.

[Max98] Nelson Max. Visualizing Hilbert curves. In *Proceedings of the conference on Visualization*, pages 447–450, 1998.

[MD86] G. Mitchison and R. Durbin. Optimal numberings of an $N \times N$ array. *SIAM Journal on Algebraic and Discrete Methods*, 7(4):571–582, 1986.

[MJFS01] B. 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*, 13(1):124–141, 2001.

[Ore86] J. A. Orenstein. Spatial query processing in an object-oriented database system. In *Proceedings of the 1986 ACM SIGMOD international conference on Management of data*, pages 326–336. ACM Press, 1986.

[PKK92] A. Pérez, S. Kamata, and E. Kawaguchi. Peano scanning of arbitrary size images. In *Proceedings of the International Conference on Pattern Recognition*, pages 565–568. IEEE Computer Society, 1992.

[Ras01] R. Rasala. Exploring recursion in Hilbert curves. In *Proceedings of the sixth Annual Conference on Innovation and Technology in Computer Science Education*, page 194, June 2001.

[Sag94] H. Sagan. *Space-Filling Curves.* Springer-Verlag, New York, 1994.

[SR00] Z. Song and N. Roussopoulos. Using Hilbert curve in image storing and retrieving. In *Proceedings of the 2000 ACM workshops on Multimedia*, pages 167–170, 2000.

[VG91] L. Velho and J. D. M. Gomes. Digital halftoning with space filling curves. In *Proceedings of the 18th annual conference on Computer graphics and interactive techniques*, pages 81–90, 1991.

[Voo91] D. Voorhies. Space-filling curves and a measure of coherence. In J. Arvo, editor, *Graphics Gems II*, pages 26–30, Academic Press, 1991.

[Wie01] J.-M. Wierum. Average case quality of partitions induced by the Lebesgue indexing. Technical Report TR-002-01, Paderborn Center for Parallel Computing, November 2001.

[Zum01] G. Zumbusch. On the quality of space-filing curve induced partitions. *Zeitschrift für Angewandte Mathematik und Mechanik*, 81(Suppl. 1):25–28, 2001.

APPENDIX A

MAPLE SOURCE CODES

A.1 Number of Edge Cuts over All $2^q \times 2^q$ Subspaces in H_k^2

```
> # of Edge Cuts For 2-dimensional Hilbert Curves.
# Space is 2^k * 2^k , the rectangular query is 2^q * 2^q
# where 1<=q<=k
# E(k,q) denotes the number of edge cuts by 2^q * 2^q rectangular
# query in a 2^k * 2^k space.
# If first character of a variable is r for a variable, it is a
# statement for recurrent equation
# If the first two characters are r for a variable, it is the
# intermediate variable for the notations that we want to solve.
# The intermediate results will appear if replacing ":" with ";".
#
# Lemma 5.1: For an Hilbert curve of order q, Pi {h,v}:
> rrv:='rrv': rrh:='rrh': ## reset variables
> rv:=rrv(q)=2*rrv(q-1)+2*rrh(q-1)+2:
> rvbase:=rrv(1)=2:
> rh:=rrh(q)=2*rrv(q-1)+2*rrh(q-1)+1:
> rhbase:=rrh(1)=1:
> rsolve({rv,rvbase,rh,rhbase},{rrv(q),rrh(q)}):
> assign(%); v:=unapply(rrv(q),q): h:=unapply(rrh(q),q):
# Lemma 5.2: For an Hilbert curve of order q, Omega bar {c1,c2}:
> rrbar_c1:='rrbar_c1': rrbar_c2:='rrbar_c2': ## reset the variables
> rbar_c1:=rrbar_c1(q)=3*rrbar_c1(q-1)+rrbar_c2(q-1)
> +3*2^(q-1)*v(q-1)+2^(q-1)*h(q-1)+3*2^(q-1)+1:
> rbar_c1base:=rrbar_c1(1)=4:
> rbar_c2:=rrbar_c2(q)=rrbar_c1(q-1)+3*rrbar_c2(q-1)+2^(q-1)*v(q-1)
> +3*2^(q-1)*h(q-1)+3*2^(q-1)+2:
> rbar_c2base:=rrbar_c2(1)=5:
> rsolve({rbar_c1,rbar_c1base,rbar_c2,rbar_c2base},{rrbar_c1(q),
> rrbar_c2(q)}):
> assign(%): bar_c1:=unapply(rrbar_c1(q),q):
> bar_c2:=unapply(rrbar_c2(q),q):
# Lemma 5.3: For an Hilbert curve of order k, Omega {c1,c2}:
> c1:=(k,q)->bar_c1(q)-h(q)-v(q):
> c2:=(k,q)->bar_c2(q)-h(q)-v(q):
```

```
# Lemma 5.4: For an Hilbert curve of order k, Omega {L,B,T}:
#       Let z=k-q (because maple only takes one variable for
#           recurrent equation).
> rrL:='rrL': rrT:='rrT': rrB:='rrB':
> rL:=rrL(z)=rrB(z-1)+rrL(z-1)+2*c1(z+q-1,q)+2*(2^q-1):
> rLbase:=rrL(0)=h(q):
> rB:=rrB(z)=2*rrL(z-1)+2*c2(z+q-1,q):
> rBbase:=rrB(0)=v(q):
> rT:=rrT(z)=2*rrT(z-1)+2*c2(z+q-1,q):
> rTbase:=rrT(0)=v(q):
> rsolve({rL,rLbase,rB,rBbase,rT,rTbase},{rrL(z),rrB(z),rrT(z)}):
> assign(%): L:=unapply(rrL(z),z,q): B:=unapply(rrB(z),z,q):
> T:=unapply(rrT(z),z,q):
# Lemma 5.5: For an Hilbert curve of order k, the number of edge cuts
#       Let z=k-q (because maple only takes one variable for
#           recurrent equation)
> rrE:='rrE': ## reset the variable
> rE:=rrE(z)=4*rrE(z-1)+L(z-1,q)+B(z-1,q)+2^q-1
>    +L(z-1,q)+L(z-1,q)
>    +L(z-1,q)+B(z-1,q)+2^q-1
>    +T(z-1,q)+T(z-1,q)
>    +2*c1(z-1,q)+2*c2(z-1,q):
> rEbase:=rrE(0)=2:
> rsolve({rE,rEbase},rrE(z)):
> Ez:=unapply(%,z,q):
> E:=unapply(simplify(Ez(k-q,q)),k,q);
```

$$E := (k, q) \rightarrow 2\, 4^{(k - q)}\, 8^{q} - 2\, 4^{q} + 2\, 8^{q} + 2^{(1 + k - q)}\, 4^{q} - 4\, 2^{(k - q)}\, 8^{q} + 2^{(1 + k - q)}$$

```
# The above is the result for 2^k * 2^k space
# Compare to Moon's paper, they use the space as 2^(k+q) x 2^(k+q)
> E(k+q,q): simplify(%);
```

$$2\, 4^{k}\, 8^{q} - 2\, 4^{q} + 2\, 8^{q} + 2^{(1 + k)}\, 4^{q} - 4\, 2^{k}\, 8^{q} + 2^{(1 + k)}$$

A.2 Number of Edge Cuts over All $2^q \times 2^q$ Subspaces in Z_k^2

```
> # of Edge Cuts For 2-dimensional z-Order Curves.
```

```
# Space is 2^k * 2^k , the rectangular query is 2^q * 2^q
# where 1<=q <=k
# E(k,q) denotes the number of edge cuts by 2^q * 2^q rectangular
# query in a 2^k * 2^k space.
# If first character of a variable is r for a variable, it is a
# statement for recurrent equation
# If the first two characters are r for a variable, it is the
# intermediate variable for the notations that we want to solve.
# The intermediate results will appear if replacing ":" by ";".
# Lemma 5.6: For a z-order curve of order q, Pi {h,v}:
> rrv:='rrv': rrh:='rrh': ## reset variables
> rv:=rrv(q)=4*rrv(q-1)+2^q+1:
> rvbase:=rrv(1)=3:
> rh:=rrh(q)=4*rrh(q-1)+2*(2^(q-1)-1)+1:
> rhbase:=rrh(1)=1:
> rsolve({rv,rvbase,rh,rhbase},{rrv(q),rrh(q)}):
> assign(%); v:=unapply(rrv(q),q): h:=unapply(rrh(q),q):
# Lemma 5.7: For a z-order curve of order q, Omega bar {c1,c2}:
> rrbar_c1:='rrbar_c1': rrbar_c2:='rrbar_c2': ## reset the variables
> rbar_c1:=rrbar_c1(q)=4*rrbar_c1(q-1)+2^q*v(q-1)+2^q*h(q-1)+2^(2*q)
> -2^q+3:
> rbar_c1base:=rrbar_c1(1)=5:
> rbar_c2:=rrbar_c2(q)=4*rrbar_c2(q-1)+2^q*v(q-1)+2^q*h(q-1)+2^(2*q)
> +2^q:
> rbar_c2base:=rrbar_c2(1)=6:
> rsolve({rbar_c1,rbar_c1base,rbar_c2,rbar_c2base},{rrbar_c1(q),
> rrbar_c2(q)}):
> assign(%): bar_c1:=unapply(rrbar_c1(q),q):
> bar_c2:=unapply(rrbar_c2(q),q):
# Lemma 5.8: For a z-order curve of order k, Omega {c1,c2}:
> c1:=(k,q)->bar_c1(q)-h(q)-v(q):
> c2:=(k,q)->bar_c2(q)-h(q)-v(q):
# Lemma 5.9: For a z-order curve of order k, Omega {L,B}:
#       Let z=k-q (because maple only takes one variable for
#         recurrent equation).
> rrL:='rrL': rrB:='rrB':
> rL:=rrL(z)=2*rrL(z-1)+c1(z+q-1,q)+c2(z+q-1,q)+(2^q-1)^2+(2^q-1):
> rLbase:=rrL(0)=h(q):
> rB:=rrB(z)=2*rrB(z-1)+c1(z+q-1,q)+c2(z+q-1,q)+(2^q-1)^2+(2^q-1):
> rBbase:=rrB(0)=v(q):
> rsolve({rL,rLbase,rB,rBbase},{rrL(z),rrB(z)}):
> assign(%): L:=unapply(rrL(z),z,q): B:=unapply(rrB(z),z,q):
# Lemma 5.10: For a z-order curve of order k, the number of edge cuts
#       Let z=k-q (because maple only takes one variable for
```

```
#         recurrent equation)
> rrE:='rrE': ## reset the variable
> rE:=rrE(z)=4*rrE(z-1)+2*(B(z-1,q)+B(z-1,q)+2*(2^q-1))
>    +2*(L(z-1,q)+L(z-1,q)+2*(2^q-1))
>    +2*c1(z-1,q)+2*c2(z-1,q)+2*(2^q-1)^2:
> rEbase:=rrE(1)=4*2+2*(B(0,q)+B(0,q))
>    +2*(L(0,q)+L(0,q)+2*(2^q-1))
>    +2*c1(0,q)+2*c2(0,q)+2*(2^q-1)^2:
> rsolve({rE,rEbase},rrE(z)):
> Ez:=unapply(%,z,q):
> E:=unapply(simplify(Ez(k-q,q)),k,q);

                   (k - q)    (1 + k - q)  q    (1 + k)
  E := (k, q) -> -4        + 4            8  - 4

          (k - q)  q      q      q      q    (q + 1)  (k - q)
       + 3 4        2  + 4 2  - 8 4  + 4 8  - 8        2

          k      (k - q)       (k - q)  q
       - 8 2  + 4 2        + 12 2       4

# The above is the result for 2^k * 2^k space
# Compare to Moon's paper, they use the space as 2^(k+q) x 2^(k+q)
> E(k+q,q): simplify(%);

   k    (1 + k)  q    (1 + k + q)      k  q      q      q      q
  -4  + 4        8  - 4            + 3 4  2  + 4 2  - 8 4  + 4 8

        (q + 1)  k      (k + q)      k      k  q
     - 8         2  - 8 2        + 4 2  + 12 2  4
```

A.3 Total Inter-cluster Distance over All $2^q \times 2^q$ Subspaces in H_k^2

```
> Inter-cluster Distance For 2-dimensional Hilbert Curves.
# Space is 2^k * 2^k , the rectangular query is 2^q * 2^q
# where 1<=q<=k
# Psiq(k,q) denotes the total intercluster distances over all
# 2^q * 2^q rectangular query in a 2^k * 2^k space.
# If first character of a variable is r for a variable, it is a
# statement for recurrent equation
# If the first two characters are r for a variable, it is the
# intermediate variable for the notations that we want to solve.
# The intermediate results will appear if replacing ":" by ";".
```

```
# Let z=k-q (because maple only takes one variable for recurrent
# equation).
# Lemma 6.1: For an Hilbert curve of order q, Pi {T,L}:
> rrPiT:='rrPiT': rrPiL:='rrPiL': ## reset variables
> rPiT:=rrPiT(q)=rrPiT(q-1)+2*(2^(q-1))^3+rrPiL(q-1)+3*(2^(q-1))^3:
> rPiTbase:=rrPiT(1)=5:
> rPiL:=rrPiL(q)=rrPiL(q-1)+(2^(q-1))^3+rrPiT(q-1)+3*(2^(q-1))^3:
> rPiLbase:=rrPiL(1)=4:
> rsolve({rPiT,rPiTbase,rPiL,rPiLbase},{rrPiT(q),rrPiL(q)}):
> assign(%): PiT:=unapply(rrPiT(q),q): PiL:=unapply(rrPiL(q),q):
# Lemma 6.2: For an Hilbert curve of order q, Omega bar {c1,c2,c3}:
> rrbar_c1:='rrbar_c1': rrbar_c2:='rrbar_c2': rrbar_c3:='rrbar_c3':
> ## reset the variables
> rbar_c1:=rrbar_c1(q)=2*rrbar_c1(q-1)+rrbar_c3(q-1)+5^3/2^8*2^(4*q)
> -3/2^4*2^(2*q):
> rbar_c1base:=rrbar_c1(1)=7:
> rbar_c2:=rrbar_c2(q)=3*rrbar_c2(q-1)+3*41/2^8*2^(4*q)-3/2^4*2^(2*q):
> rbar_c2base:=rrbar_c2(1)=7:
> rbar_c3:=rrbar_c3(q)=rrbar_c1(q-1)+rrbar_c3(q-1)+23/2^5*2^(4*q)
> -3/2^3*2^(2*q):
> rbar_c3base:=rrbar_c3(1)=10:
> rsolve({rbar_c1,rbar_c1base,rbar_c3,rbar_c3base},{rrbar_c1(q),
> rrbar_c3(q)}):
> assign(%): bar_c1:=unapply(simplify(rrbar_c1(q)),q):
> bar_c3:=unapply(simplify(rrbar_c3(q)),q):
> rsolve({rbar_c2,rbar_c2base},rrbar_c2(q)):
> bar_c2:=unapply(simplify(%),q):
# Lemma 6.3: For an Hilbert curve of order k, Omega {c1,c2,c3}:
> c1:=(k,q)->bar_c1(q)-PiL(q)-(2^q-1)*(2^(2*q)-1):
> c2:=(k,q)->bar_c2(q)-PiT(q)-PiL(q)+(2^(2*q)-1)
> +(2^q-1)^2*sum(2^(2*i),i=q..k-1):
> c3:=(k,q)->bar_c3(q)-PiT(q)-(2^q-1)*(2^(2*q)-1)+(2^q-1)^2*(2^(2*k)
> -sum(2^(2*i),i=q..k-1)-2^(2*q)):
#     Let z=k-q
> Nc:=(k,q)->(2^q-1)^2:
> c1z:=(z,q)->c1(z+q,q):
> c2z:=(z,q)->c2(z+q,q):
> c3z:=(z,q)->c3(z+q,q):
> Ncz:=(z,q)->Nc(z+q,q):
# Lemma 6.4: For an Hilbert curve of order k, Omega {L,R,B,T}:
#       Let z=k-q (because maple only takes one variable for
#         recurrent equation).
> Ns:=(k,q)->(2^k-2^q+1)*(2^q-1):
> Nsz:=(z,q)->Ns(z+q,q):
```

```
> rrL:='rrL': rrR:='rrR': rrT:='rrT': rrB:='rrB':
> rL:=rrL(z)=rrB(z-1)+c1z(z-1,q)+Ncz(z-1,q)*(2^((z+q)-1))^2
> +rrL(z-1)+Nsz(z-1,q)*(2^((z+q)-1))^2:
> rLbase:=rrL(0)=PiL(q)-(2^(2*q)-1):
> rR:=rrR(z)=rrB(z-1)+3*Nsz(z-1,q)*(2^((z+q)-1))^2
> +c1z(z-1,q)+3*Ncz(z-1,q)*(2^((z+q)-1))^2
> +rrR(z-1)+2*Nsz(z-1,q)*(2^((z+q)-1))^2:
> rRbase:=rrR(0)=(2^q-1)*(2^(2*q)-1):
> rB:=rrB(z)=rrL(z-1)+c3z(z-1,q)+3*Ncz(z-1,q)*(2^((z+q)-1))^2
> +rrR(z-1)+3*Nsz(z-1,q)*(2^((z+q)-1))^2:
> rBbase:=rrB(0)=(2^q-1)*(2^(2*q)-1):
> rT:=rrT(z)=rrT(z-1)+(2^((z+q)-1)-2^q+1)*(2^q-1)*(2^((z+q)-1))^2
> +c2z(z-1,q)+2*(2^q-1)^2*(2^((z+q)-1))^2
> +rrT(z-1)+2*(2^((z+q)-1)-2^q+1)*(2^q-1)*(2^((z+q)-1))^2:
> rTbase:=rrT(0)=PiT(q)-(2^(2*q)-1):
> rsolve({rL,rLbase,rR,rRbase,rB,rBbase},{rrL(z),rrR(z),rrB(z)}):
> assign(%): L:=unapply(rrL(z),z,q): R:=unapply(rrR(z),z,q):
> B:=unapply(simplify(rrB(z)),z,q):
> rsolve({rT,rTbase},rrT(z)):
> T:=unapply(simplify(%),z,q):
# Lemma 6.5: For an Hilbert curve of order k,
#       Let z=k-q
> oL:=(z,q)->(2^(2*(z+q))-1)*Nsz(z,q)-R(z,q):
> oR:=(z,q)->(2^(2*(z+q))-1)*Nsz(z,q)-L(z,q):
> oB:=(z,q)->(2^(2*(z+q))-1)*Nsz(z,q)-B(z,q):
> oT:=(z,q)->(2^(2*(z+q))-1)*Nsz(z,q)-T(z,q):
# Lemma 6.6: For an Hilbert curve of order k,
#       Let z=k-q
> oc3:=(k,q)->(2^(2*k)-1)*Nc(k,q)-c2(k,q):
> oc3z:=(z,q)->oc3(z+q,q):
# Theorem 6.1: For a canonical Hilbert of order k, sum of all
# inter-cluster distances
#    Let z=k-q.
> rrPsi:='rrPsi':
> rPsi:=rrPsi(z)=4*rrPsi(z-1)+B(z-1,q)+2^(2*(z+q)-2)*Nsz(z-1,q)
>   -oR(z-1,q)-(2^(2*q)-1)*Nsz(z-1,q)
>   +L(z-1,q)+2*2^(2*(z+q)-2)*Nsz(z-1,q)
>   -(oR(z-1,q)+2^(2*(z+q)-2)*Nsz(z-1,q))-(2^(2*q)-1)*Nsz(z-1,q)
>   +L(z-1,q)+3*2^(2*(z+q)-2)*Nsz(z-1,q)
>   -(oB(z-1,q)+2*2^(2*(z+q)-2)*Nsz(z-1,q))-(2^(2*q)-1)*Nsz(z-1,q)
>   +T(z-1,q)+3*2^(2*(z+q)-2)*Nsz(z-1,q)
>   -(oT(z-1,q))-(2^(2*q)-1)*Nsz(z-1,q)
>   +c2z(z-1,q)+3*2^(2*(z+q)-2)*Ncz(z-1,q)
>   -(oc3z(z-1,q))-(2^(2*q)-1)*Ncz(z-1,q):
```

```
> rPsiBase:=rrPsi(0)=0:
> rsolve({rPsi,rPsiBase},rrPsi(z)): Psiz:=unapply(%,z,q):
# change input parameter to be k,q (k=z+q).
> Psiq:=unapply(simplify(Psiz(k-q,q)),k,q):
```

A.4 Total Inter-cluster Distance over All $2^q \times 2^q$ Subspaces in Z_k^2

```
> Inter-cluster Distance For 2-dimensional z-order Curves.
# Space is 2^k * 2^k , the rectangular query is 2^q * 2^q
# where 1<=q<=k
# Psiq(k,q) denotes the total inter-cluster distances over
# all  2^q * 2^q rectangular query in a 2^k * 2^k space.
# If first character of a variable is r for a variable, it is a
# statement for recurrent equation
# If the first two characters are r for a variable, it is the
# intermediate variable for the notations that we want to solve.
# The intermediate results will appear if replacing ":" by ";".
# Let z=k-q (because maple only takes one variable for recurrent
# equation).
# Lemma 6.7: For a z-order curve of order q, Pi {B,L}:
> rrPiB:='rrPiB': rrPiL:='rrPiL': ## reset variables
> rPiB:=rrPiB(q)=2*rrPiB(q-1)+5*(2^(q-1))^3:
> rPiBbase:=rrPiB(1)=5:
> rPiL:=rrPiL(q)=2*rrPiL(q-1)+4*(2^(q-1))^3:
> rPiLbase:=rrPiL(1)=4:
> rsolve({rPiB,rPiBbase,rPiL,rPiLbase},{rrPiB(q),rrPiL(q)}):
> assign(%): PiB:=unapply(rrPiB(q),q): PiL:=unapply(rrPiL(q),q):
# Lemma 6.8: For a z-order curve of order q, Omega bar {c1}:
> rrbar_c1:='rrbar_c1':  ## reset the variables
> rbar_c1:=rrbar_c1(q)=4*rrbar_c1(q-1)+6*(2^(q-1))^4:
> rbar_c1base:=rrbar_c1(1)=6:
> rsolve({rbar_c1,rbar_c1base},rrbar_c1(q)):
> bar_c1:=unapply(simplify(%),q):
# Lemma 6.9: For a z-order curve of order k, Omega {c1}:
> c1:=(k,q)->bar_c1(q)-PiB(q)-PiL(q)+(2^(2*q)-1):
#    Let z=k-q
> Nc:=(k,q)->(2^q-1)^2:
> c1z:=(z,q)->c1(z+q,q):
> Ncz:=(z,q)->Nc(z+q,q):
# Lemma 6.10: For a z-order curve of order k, Omega {L,B}:
#      Let z=k-q (because maple only takes one variable for
#         recurrent equation).
> Ns:=(k,q)->(2^k-2^q+1)*(2^q-1):
> Nsz:=(z,q)->Ns(z+q,q):
```

```
> rrL:='rrL': rrB:='rrB':
> rL:=rrL(z)=rrL(z-1)+c1z(z-1,q)+Ncz(z-1,q)*(2^((z+q)-1))^2
> +rrL(z-1)+Nsz(z-1,q)*(2^((z+q)-1))^2:
> rLbase:=rrL(0)=PiL(q)-(2^(2*q)-1):
> rB:=rrB(z)=rrB(z-1)+c1z(z-1,q)+2*Ncz(z-1,q)*(2^((z+q)-1))^2
> +rrB(z-1)+2*Nsz(z-1,q)*(2^((z+q)-1))^2:
> rBbase:=rrB(0)=PiB(q)-(2^(2*q)-1):
> rsolve({rL,rLbase,rB,rBbase},{rrL(z),rrB(z)}):
> assign(%): L:=unapply(rrL(z),z,q): B:=unapply(rrB(z),z,q):
# Lemma 6.11: For a z-order curve of order k,
#       Let z=k-q
> oR:=(z,q)->(2^(2*(z+q))-1)*Nsz(z,q)-L(z,q):
> oT:=(z,q)->(2^(2*(z+q))-1)*Nsz(z,q)-B(z,q):
# Lemma 6.12: For a z-order curve of order k,
#       Let z=k-q
> oc4:=(k,q)->(2^(2*k)-1)*Nc(k,q)-c1(k,q):
> oc4z:=(z,q)->oc4(z+q,q):
# Theorem 6.2: For a canonical z-order of order k, sum of all
# inter-cluster distances
#    Let z=k-q.
> rrPsi:='rrPsi':
> rPsi:=rrPsi(z)=4*rrPsi(z-1)+B(z-1,q)+2^(2*(z+q)-2)*Nsz(z-1,q)
>   -oT(z-1,q)-(2^(2*q)-1)*Nsz(z-1,q)
>   +L(z-1,q)+3*2^(2*(z+q)-2)*Nsz(z-1,q)
>   -(oR(z-1,q)+2^(2*(z+q)-2)*Nsz(z-1,q))-(2^(2*q)-1)*Nsz(z-1,q)
>   +B(z-1,q)+3*2^(2*(z+q)-2)*Nsz(z-1,q)
>   -(oT(z-1,q)+2*2^(2*(z+q)-2)*Nsz(z-1,q))-(2^(2*q)-1)*Nsz(z-1,q)
>   +L(z-1,q)+2*2^(2*(z+q)-2)*Nsz(z-1,q)
>   -(oR(z-1,q))-(2^(2*q)-1)*Nsz(z-1,q)
>   +c1z(z-1,q)+3*2^(2*(z+q)-2)*Ncz(z-1,q)
>   -(oc4z(z-1,q))-(2^(2*q)-1)*Ncz(z-1,q):
> rPsiBase:=rrPsi(0)=0:
> rsolve({rPsi,rPsiBase},rrPsi(z)): Psiz:=unapply(%,z,q):
# change input parameter to be k,q (k=z+q).
> Psiq:=unapply(simplify(Psiz(k-q,q)),k,q):
> Psiq(k,q);

   (k - q)    q     (1 + k - q)  q     k       (k - q)  q
  2        16   - 2               4   + 2   + 2 4        8

              (k - q)    q     q       q     q     k     (k - q)    q
        - 2 4         16   - 8   + 2 4   - 2   - 8   + 8        16
```

APPENDIX B

PROGRAM SOURCE CODES

B.1 2-Dimensional Space-Filling Curves

B.1.1 VectorD2D.h

```
//////////////////////////////////////////////////////////////////////
// VectorD2D: a class for the 2D vector.
//
// Note: all the indices in program starts from 0
//////////////////////////////////////////////////////////////////////
#ifndef __VECTORD2D_H
#define __VECTORD2D_H
#include <iostream>
#include <sstream>
#include <string>
#include <math.h>
using namespace std ;

class VectorD2D
{
protected:
    int mNumDim; // dimensionality
    long mE[2];     // the coordinates (axis-1, axis-2)

public:
    VectorD2D():mNumDim(2) {    //constructor, default 2-D
        for (int i=0; i<mNumDim; i++)
            mE[i]=-1;
    }

    VectorD2D(VectorD2D *other):mNumDim(2)    { //copy constructor
        for (int i=0; i<mNumDim; i++)
            mE[i]=other->mE[i];
    }

    VectorD2D(int a, int b):mNumDim(2){    //constructor
        //VectorD2D();
```

```
    mE[0]=a;
    mE[1]=b;
}

~VectorD2D() {    // destructor
    //delete[] mE;
}

int getnDim() {    // # of dimension
    return mNumDim;
}

int getVofDim(int dim)    { // coordinate
    if (dim > getnDim())
        return -9999;
    return mE[dim];
}

// addition
VectorD2D add (VectorD2D &other) {
    VectorD2D tmp(this);
    int n_lower = getnDim(); // get the lower dimensionality
    if (getnDim() >= other.getnDim()) n_lower = other.getnDim();
    for (int i=0; i< n_lower; i++){
        tmp.mE[i] += other.mE[i];
    }
    return tmp;
}

// subtraction
VectorD2D subtract (VectorD2D &other) {
    VectorD2D tmp(this);
    int n_lower = getnDim(); // get the lower dimensionality
    if (getnDim() >= other.getnDim()) n_lower = other.getnDim();
    for (int i=0; i< n_lower; i++) {
        tmp.mE[i] -=  other.mE[i];
    }
    return tmp;
}

// multiply
VectorD2D Multiply (int num) {
    VectorD2D tmp(this);
    for (int i=0; i< tmp.getnDim(); i++) {
```

```
        tmp.mE[i] *= num;
    }
    return tmp;
}

VectorD2D Multiply (double num)    {
    VectorD2D tmp(this);
    for (int i=0; i< tmp.getnDim(); i++) {
        tmp.mE[i] = (int) tmp.mE[i] * num;
    }
    return tmp;
}

VectorD2D MultiplyLeftShiftHalf (double num) {
    VectorD2D tmp(this);
    double tmpDouble =0.0;

    for (int i=0; i< tmp.getnDim(); i++) {
        tmpDouble = ((double) tmp.mE[i]) * num;
        if (-1 < tmpDouble && tmpDouble <0)
            tmp.mE[i] = -1;
        else
            tmp.mE[i] = (int)( tmp.mE[i] * num);
    }
    return tmp;
}

// dot opertation
long dot (VectorD2D *other)    {
    int tmp = 0;
    for (int i=0; i< getnDim(); i++) {
        tmp += mE[i] * other->mE[i];
    }
    return tmp;
}
long dot (VectorD2D &other)    {
    int tmp = 0;
    for (int i=0; i< getnDim(); i++) {
        tmp += mE[i] * other.mE[i];
    }
    return tmp;
}

VectorD2D& copyFrom(VectorD2D &other) {    //copy , -1 if error
```

```
    int nshortest = getnDim();
    if (getnDim() >= other.getnDim())
        nshortest = other.getnDim();    // get the shortest

    for (int i=0; i< nshortest; i++) {
        mE[i] = other.mE[i];
    }
    return *this;
}

VectorD2D& copyFrom(VectorD2D *other) {    //copy constructor
    int nshortest = getnDim();
    if (getnDim() >= other->getnDim())
        nshortest = other->getnDim();    // get the shortest

    for (int i=0; i< nshortest; i++) {
        mE[i] = other->mE[i];
    }
    return *this;
}

VectorD2D & operator = ( VectorD2D &other) {
    if (this != &other) {
        for (int i=0; i<mNumDim; i++)
            mE[i]=other.mE[i];
    }
    return *this;
}

VectorD2D Normalized() {
    VectorD2D unitVector(*this);
    long sqr_len=0;
    int i;

    for (i=0; i<mNumDim; i++)    //sqr sum
        sqr_len += (long) pow(mE[i], 2.0);

    sqr_len = (long) sqrt(sqr_len); // length(distance)

    if (sqr_len != 0) {
        for (i=0; i<mNumDim; i++)
            unitVector.mE[i] /= sqr_len;
    }
    return unitVector;
```

```
    }

    string toString() {
        std::ostringstream o;
        if (o << *this)
            return o.str();
        return "";
    }

    friend std::ostream& operator<< (std::ostream& o, VectorD2D& v) {
        o << "(" << v.getVofDim(0);
        for (int i=1; i< v.getnDim() ; i++)    {
            o <<  ", " << v.getVofDim(i) ;
        }
        return o << ")";
    }
};
#endif
```

B.1.2 Node2D.h

```
//////////////////////////////////////////////////////////////////////
// Node2D: an class for the element in the array.
//   data: coordinates, index, next element
// Note: all the indices in program starts from 0
//////////////////////////////////////////////////////////////////////
#ifndef __NODE2D_H
#define __NODE2D_H
#include <iostream>
#include <sstream>
#include <string>
#include "VectorD2D.h"
using namespace std ;

class Node2D  {
private:
    VectorD2D *mData; //coordinates (axis 1, axis 2)
    VectorD2D *mNext; //next indexed element's coordinates
    long mSFCCode;//index
    int mChecked;      // mark after visited in a traversal

public:
    Node2D() {    //constructor
        mData = new VectorD2D(-1,-1);
```

```
    mNext = NULL;
    mSFCCode = -1;
    mChecked=0;
}

// constructor with input of next node's coordinates, this index
Node2D(VectorD2D *data, VectorD2D *next, long hcode) {
    if (data != NULL)
        mData = new VectorD2D(data);
    else
        mData=NULL;
    if (next != NULL)
        mNext = new VectorD2D(next);
    else
        mData = NULL;
    mSFCCode = hcode;
    mChecked=0;
}

~Node2D() { //destructor
    delete mData;
    delete mNext;
}

void setData(long a, long b) {
    VectorD2D tmp(a,b);
    if (mData == NULL)
        cerr << "NULL for mData" << endl;
    mData->copyFrom(tmp);
}

VectorD2D* getNext() { return  mNext; }

VectorD2D* getCoor() {return mData;}

long getSFCCode() { return mSFCCode; }

void setNext(VectorD2D *next) {
    if (next != NULL)
        if (mNext == NULL)
            mNext = new VectorD2D(next);
        else
            mNext->copyFrom(next);
}
```

```
    void setSFCCode(long hcode)    { mSFCCode = hcode;}

    string toString() {
        std::ostringstream o;
        if (o << *this)
            return o.str();
        return "";
    }

    friend std::ostream& operator<< (std::ostream& o,  Node2D& v) {
        o << "Node: " << *(v.mData) << " (index code="
            << v.getSFCCode();
        if (v.mNext == NULL)
            o << ")  nextNode= (NULL)";
        else
            o << ")  nextNode= " << *(v.mNext) ;
        return o;
    }

    // The following 3 fxns are for traversal
    void setChecked() { mChecked++; } // mark for been visited
    void setUnChecked() { mChecked=0; }    // clear the mark
    int getChecked() { return mChecked; } // the status of the mark
};
#endif
```

B.1.3 SFCArray2D.h

```
//////////////////////////////////////////////////////////////////////
// SFCArray2D: a class for maintaining the indices and coordinates.
//
// Note: all the indices in program starts from 0
//////////////////////////////////////////////////////////////////////
#ifndef __SFCARRAY2D_H
#define __SFCARRAY2D_H
#include <iostream>
#include <stdio.h>
#include <malloc.h>
#include <math.h>
#include "VectorD2D.h"
#include "Node2D.h"

using namespace std ;
```

```
long two(long x) { // 2^x
    long one=1;
    long outcome=0;
    if (x<0)
        return -1;
    return one << x;
}

class SFCArray2D {
private:
    Node2D* mSFCArray; // array of grid points
    long order;     // the order
    long mWidth;     // the max width for each coordinate, =2^(order)
    int Successful; // for checking the indexing
    int mSFC; // type sfc, 0: Hilbert; 1: z-order

    // the constructor for a node in SFCArrayNode
    //     by taking different input formats.
    Node2D* SFCArrayNode (VectorD2D *location) {
        if (location != NULL)
            return &(mSFCArray[location->getVofDim(0) * mWidth
                    + location->getVofDim(1)]);
        else
            return NULL;
    }

    Node2D* SFCArrayNode (VectorD2D &location) {
         return &(mSFCArray[location.getVofDim(0) * mWidth
                + location.getVofDim(1)]);
     }

    Node2D* SFCArrayNode (const long i, const long j) {
        if (i < mWidth && i>=0 && j <mWidth && j>=0)
            return &(mSFCArray[i * mWidth + j]);
        else
            return NULL;
    }

public:
    // Hilbert array
    SFCArray2D(long max, int level, int sfc): mWidth(max), mSFC(sfc),
        Successful(0), order(level) {
        if (max < (long)pow(2.0, level)) { // Array is too small
```

```
        cerr << "Overflow in the array";
        return;
    }
    mSFCArray = new Node2D[max*max];
    // initiallize elements
    for (long i = 0; i< mWidth; i++) {
        for (long j=0; j< mWidth; j++) {
            mSFCArray[i*mWidth + j].setData(i,j);
        }
    }
     VectorD2D o(0,0);    // origin
    VectorD2D D0 (max, 0);    // D0 direction
    VectorD2D D1 (0,max);    // D1 direction
    // construct the array's nextpoint
    // recusive call to calculate the linkage and the SFC code
    if (mSFC==0) // Hilbert curve
        SFCLineArray(o, D0, D1, level, &o);
    else // z-order curve mSFC==1
        ZLineArray(o, D0, D1, level, &o);
}

 ~SFCArray2D() {
    delete[] mSFCArray;
}

long getWidth()    { return mWidth; }

VectorD2D SFCLineArray(VectorD2D &o, VectorD2D &D0,
    VectorD2D &D1, int level, VectorD2D *Prev)    {
    VectorD2D tmpo(o), tmpD0(D0), tmpD1(D1); //copy of o, D0, D1
    if (level <= 0) {
        if (Prev->getVofDim(0)<0 || Prev->getVofDim(1)<0 ||
              Prev->getVofDim(0)>=getWidth() ||
              Prev->getVofDim(1)>=getWidth())
            cerr <<"error, less:" << (*Prev) << endl;

        if (SFCArrayNode(Prev) == NULL)
             SFCArrayNode(o)->setNext(&o); //no previous node

         else {
            SFCArrayNode(o)->setSFCCode(
                SFCArrayNode(Prev)->getSFCCode()+1);
            SFCArrayNode(Prev)->setNext(&o);
         }
```

```
        return o;
    }
    else {
        VectorD2D tmpV;

        tmpV=SFCLineArray(tmpo=o,
            tmpD1=D1.Multiply(.5), tmpD0=D0.Multiply(.5),
            level-1,  Prev); //00 block
        tmpV=SFCLineArray(tmpo=o.add(D0.Multiply(.5)),
            tmpD0=D0.Multiply(.5), tmpD1=D1.Multiply(.5),
            level-1, &tmpV); //01 block
        tmpV=SFCLineArray(tmpo=o.add(D0.Multiply(.5)).
            add(D1.Multiply(.5)),
            tmpD0=D0.Multiply(.5), tmpD1=D1.Multiply( .5),
            level-1, &tmpV); //11 block
        tmpV=SFCLineArray(tmpo=o.add(D0.Multiply(.5)).
            add(D1).subtract(D0.Normalized()).
            subtract(D1.Normalized()),
            tmpD1=D1.Multiply(-.5), tmpD0=D0.Multiply(-.5),
            level-1, &tmpV); //10 block
        return tmpV;
    }
}

// z-order
VectorD2D ZLineArray(VectorD2D &o, VectorD2D &D0, VectorD2D &D1,
        int level, VectorD2D *Prev)    {
    VectorD2D tmpo(o), tmpD0(D0), tmpD1(D1);
    if (level <= 0)    {
        if (Prev->getVofDim(0)<0 || Prev->getVofDim(1)<0 ||
              Prev->getVofDim(0)>=getWidth() ||
              Prev->getVofDim(1)>=getWidth()) {
            cerr <<"error, less:" << (*Prev) << endl;
        }
        if (SFCArrayNode(Prev) == NULL) { // no previous node
             SFCArrayNode(o)->setNext(&o);
        }
         else {
            SFCArrayNode(o)->setSFCCode(SFCArrayNode(Prev)
              ->getSFCCode()+1);
            SFCArrayNode(Prev)->setNext(&o);
         }
        return o;
    }
```

```
    else {
        VectorD2D tmpV;

        tmpV=ZLineArray(tmpo=o,
            tmpD0=D0.Multiply(.5), tmpD1=D1.Multiply(.5),
            level-1,  Prev); //00 block
        tmpV=ZLineArray(tmpo=o.add(D0.Multiply(.5)),
            tmpD0=D0.Multiply(.5), tmpD1=D1.Multiply(.5),
            level-1, &tmpV); //01 block
        tmpV=ZLineArray(tmpo=o.add(D1.Multiply(.5)),
            tmpD0=D0.Multiply(.5), tmpD1=D1.Multiply( .5),
            level-1, &tmpV); //11 block
        tmpV=ZLineArray(tmpo=o.add(D0.Multiply(.5)).
                add(D1.Multiply(.5)),
            tmpD0=D0.Multiply(.5), tmpD1=D1.Multiply(.5),
            level-1, &tmpV); //10 block
        return tmpV;
    }
}
//--- end of z-order

const VectorD2D* getNextPoint(VectorD2D *cur) {
    if (cur != NULL && SFCArrayNode(cur)!= NULL)
        return SFCArrayNode(cur)->getNext();
    else
        return NULL;
}

const VectorD2D* getNextV(Node2D *cur) {
    if (cur != NULL )
        return cur->getNext();
    else
        return NULL;
}
// this will check the indexing for SFC curve
// return the # of used
long Check()
{
    VectorD2D *tmp = new VectorD2D(0,0);     // starting point
    Node2D *tmpNode = SFCArrayNode(tmp);// starting node
    long nChecked=0,     // Checked #
        nError=0,         // Error #
        old_hcode = -1;       // cur_SFCCode
    delete tmp;
```

```
    while (tmpNode != NULL ){
        if (tmpNode->getSFCCode() != old_hcode+1)
            cout << "Not cotinuous at" << tmpNode->toString()
            << " and prev:" << old_hcode <<endl;
        tmpNode->setChecked();
        if (nChecked>mWidth*mWidth) {
            cout << "Error in the setChecked "<< endl;
            return -1;
        }
      old_hcode = tmpNode->getSFCCode();
      tmpNode = SFCArrayNode(tmpNode->getNext());
    }

    for (int i=0; i< getWidth(); i++)
        for (int j=0; j<getWidth(); j++)
            if (SFCArrayNode(i,j) != NULL)
                if (SFCArrayNode(i,j)->getChecked() == 1)
                    nChecked++;
                else
                    nError--;
    cout << "init:" << *SFCArrayNode(0,0) <<
        " \nend:" << *SFCArrayNode(0,getWidth()-1) << endl;
    cout << "nError: "<<nError<<" nChecked:" << nChecked << endl;
    if (nError < 0)
        return nError;
    else
        return nChecked;
}

// created succesfully?
int IsSuccessful() { return Successful; }

long lemma3_1(int r=0) {
    long R1=0,R2; // sum for row, R1 R2 is for comparison
    if (mSFC==0) {
      for (int i=0; i< getWidth() ; i++){
        R1=0; // initialize R1
        for (int j=0; j<getWidth(); j++) {
            R1 += abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode());
        }
        if (i>0)
            if (R1 != R2)
```

```
                cerr << " row " << i-1 << " != row " << i <<endl;
        R2=R1;
      }
    }
    else { // z-order
        for (int j=0; j<getWidth(); j++)
            R1 += abs(SFCArrayNode(r,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode());
    }
    cout << " Sum at row " << r << " = " << R1 << endl;
    return R1;
}

long lemma3_2(int c) {  // c is column number (indexed from 0)
    long cSum=0; // summation of column c
     for (int i=0; i<getWidth(); i++)
        cSum += abs(SFCArrayNode(i,c)->getSFCCode() -
              SFCArrayNode(0,0)->getSFCCode());
    cout << " Sum at column " << c << " = " << cSum << endl;
    return cSum;
}

long lemma3_3() {
    long ASum=0, DSum=0; // summation of A, D
    for (int i=0; i<getWidth(); i++) {
        ASum += abs(SFCArrayNode(i,getWidth()-i-1)->
                getSFCCode() -
            SFCArrayNode(0,0)->getSFCCode());
        DSum += abs(SFCArrayNode(i,i)->getSFCCode() -
            SFCArrayNode(0,0)->getSFCCode());
    }
    if ((mSFC==0 && ASum+DSum != two(3*order)-two(order)) ||
        (mSFC==1 && (ASum!=DSum ||
            ASum!= (two(3*order)-two(order))/2)))
          cerr << "A+D is incorrect. A+D=" << ASum+DSum << endl;
    cout << " Sum for A (2^k) = " << ASum << endl;
    return ASum;
}

long lemma3_5(int q) {  // 2^q * 2^q is query size
    if (q<=0 || q>order) {
        cerr << "0<q<=k only"<<endl;
        return -1;
    }
```

```
long Wbar=0, Nbarq=0, Rbarq=0, Cbarq=0, Abarq=0,
    AbarPrimeq=0, Dbarq=0, DbarPrimeq=0; //summation of Xbar
int i, j, ri, rj;
for (i=0; i<getWidth(); i++)
    for (j=0; j<getWidth(); j++)
        Wbar += abs(SFCArrayNode(i,j)->getSFCCode() -
            SFCArrayNode(0,0)->getSFCCode());
for (i=0; i<two(q); i++)
    for (j=0; j<two(q)-i; j++)
        Nbarq ++;
for (i=0; i<two(q); i++)
    for (j=0; j<getWidth(); j++)
        Rbarq += abs(SFCArrayNode(i,j)->getSFCCode() -
            SFCArrayNode(0,0)->getSFCCode());
for (i=0; i<getWidth(); i++)
    for (j=0; j<two(q); j++)
        Cbarq += abs(SFCArrayNode(i,j)->getSFCCode() -
            SFCArrayNode(0,0)->getSFCCode());
for (i=0; i<two(q); i++)
    for (j=0; j<two(q)-i; j++) {
        rj= getWidth()-j-1;  // reverse direction of j
        ri= getWidth()-i-1;  // reverse direction of i
        Abarq += abs(SFCArrayNode(i,j)->getSFCCode() -
            SFCArrayNode(0,0)->getSFCCode());
        Dbarq += abs(SFCArrayNode(i,rj)->getSFCCode()-
            SFCArrayNode(0,0)->getSFCCode());
        DbarPrimeq += abs(SFCArrayNode(ri,j)->
                getSFCCode()-
            SFCArrayNode(0,0)->getSFCCode());
        AbarPrimeq += abs(SFCArrayNode(ri,rj)->
                getSFCCode()-
            SFCArrayNode(0,0)->getSFCCode());
    }

if (Wbar != two(4*order-1)-two(2*order-1))
    cerr << "Wbar error, Wbar= " << Wbar << endl;
if (Nbarq != two(2*q-1)+two(q-1))
    cerr << "Nbar error, Nbar= " << Nbarq << endl;
if (mSFC==0){
if (Rbarq != two(q)*lemma3_1())
    cerr << "Rbar error, Rbar= " << Rbarq << endl;
if (Cbarq != two(q)*lemma3_2(0)+2*(two(4*q)-two(q))/7)
    cerr << "Cbar error, Cbar= " << Cbarq << endl;
long Wbarq_1 = two(4*(q-1))/2-two(2*(q-1))/2;
```

```
    long Nbarq_1 = (long)(pow(2.,2*(q-1))/2.+pow(2.,q-1)/2.);
    long ASumq_1 = 0;
    for (i=0; i<two(q-1); i++)
        ASumq_1 +=abs(SFCArrayNode(i,two(q-1)-i-1)->getSFCCode()-
            SFCArrayNode(0,0)->getSFCCode());
    //Abar (k,q) == Abar(q,q)
    if (Abarq != 2*Wbarq_1+ASumq_1+two(2*q)*Nbarq_1)
        cerr << "Abar error, Abar= " << Abarq << endl;
    if (Dbarq + Abarq != (two(2*order)-1)*Nbarq)
        cerr << "Dbar+Abar error" << endl;
    if (Dbarq + Abarq != DbarPrimeq+AbarPrimeq)
        cerr << "Dbar+Abar = Abar' + Dbar' error" << endl;
    }
    else {  // z-order
    if (Rbarq != two(q)*lemma3_1(0)+
          (two(order+3*q)-two(order+q))/6)
        cerr << "Rbar error, Rbar= " << Rbarq << endl;
    if (Cbarq != two(q)*lemma3_2(0)+
            (two(order+3*q)-two(order+q))/3)
        cerr << "Cbar error, Cbar= " << Cbarq << endl;
    long ASumq = 0;
    for (i=0; i<two(q); i++)
        ASumq += abs(SFCArrayNode(i,two(q)-i-1)->getSFCCode() -
            SFCArrayNode(0,0)->getSFCCode());
    // Abar (k,q)==Abar(q,q)
    if (Abarq != (two(4*q)+7*two(3*q-2)-7*two(2*q-2)-two(q))/7)
        cerr << "Abar error, Abar= " << Abarq << endl;
    }
    cout << "Abar(q)="<<Abarq <<" Abar'(q)="<<AbarPrimeq
        << " Dbar(q)="<<Dbarq <<" Dbar'(q)="<<DbarPrimeq << endl;
    return DbarPrimeq;
}

long lemma3_6(int q) {  // 2^q * 2^q is query size
    if (q<=0 || q>order) {
        cerr << "0<q<=k only"<<endl;
        return -1;
    }
    long Nq=0, Rq=0, Cq=0, Aq=0,
        APrimeq=0, Dq=0, DPrimeq=0; //summation of X
    int i, j, ri, rj;
    for (i=0; i<two(q); i++)
        for (j=0; j<two(q)-i; j++)
            Nq += (two(q)-i-j);
```

```
for (i=0; i<two(q); i++)
    for (j=0; j<getWidth(); j++)
        Rq += abs(SFCArrayNode(i,j)->getSFCCode() -
          SFCArrayNode(0,0)->getSFCCode())*(two(q)-i);
for (i=0; i<getWidth(); i++)
    for (j=0; j<two(q); j++)
        Cq += abs(SFCArrayNode(i,j)->getSFCCode() -
          SFCArrayNode(0,0)->getSFCCode())*(two(q)-j);
for (i=0; i<two(q); i++)
    for (j=0; j<two(q)-i; j++) {
        rj= getWidth()-j-1;  // reverse direction of j
        ri= getWidth()-i-1;  // reverse direction of i
        Aq += abs(SFCArrayNode(i,j)->getSFCCode() -
          SFCArrayNode(0,0)->getSFCCode())*(two(q)-i-j);
        Dq += abs(SFCArrayNode(i,rj)->getSFCCode()-
          SFCArrayNode(0,0)->getSFCCode())*(two(q)-i-j);
        DPrimeq += abs(SFCArrayNode(ri,j)->
                getSFCCode()-
          SFCArrayNode(0,0)->getSFCCode())*(two(q)-i-j);
        APrimeq += abs(SFCArrayNode(ri,rj)->
                getSFCCode()-
          SFCArrayNode(0,0)->getSFCCode())*(two(q)-i-j);
    }
double ftmp=(pow(2.,3*q-1)+3*pow(2.,2*q-1)+pow(2.,q))/3.;
if (Nq != ftmp)
    cerr << "N error, N= "<<Nq <<" formula=" << ftmp << endl;
if (mSFC ==0 ) { // Hilbert curve
if (Rq != lemma3_1()*(two(q)+1)*two(q)/2)
    cerr << "R error, R= " << Rq << endl;
if (Cq != (3*two(3*order+2*q-2)+3*two(3*order+q-2)
      -7*two(order+2*q-2)-7*two(order+q-2)+(two(5*q+3)
      +7*two(q))/15+two(4*q))/7)
    cerr << "C error, C= " << Cq << endl;
if (Dq + Aq != (two(2*order)-1)*Nq)
    cerr << "D+A error" << endl;
if (Dq + Aq != DPrimeq+APrimeq)
    cerr << "D+A = A' + D' error" << endl;
ftmp=7./45.*pow(2.0,5*q-2)+3.*pow(2.,4*q-4)
    +5./9.*pow(2.,3*q-2)-pow(2.,2*q-2)-pow(2.,q+3)/45.;
if (Aq != ftmp)
    cerr << "A error, A= " <<Aq<<" formula=" << ftmp << endl;
ftmp=11./45.*pow(2.0,5*q-2)+3.*pow(2.,4*q-4)+pow(2.,3*q-2)/9.
    -pow(2.,2*q-2)-pow(2.,q+2)/45.;  // D'q,q
if (DPrimeq != Nq*(two(2*order)-two(2*q))/3+ftmp)
```

```
        cerr << "D' error, D'= " << DPrimeq << " formula="
        << Nq*(two(2*order)-two(2*q))/3+ftmp << endl;
    } else
    {
    if (Rq != (two(3*order+2*q-1)*7+two(3*order+q-1)*7
          +two(order+4*q)+two(order+3*q-2)*7
          -two(order+2*q-2)*21-two(order+q-1)*9)/21)
        cerr << "R error, R= " << Rq << endl;
    if (Cq != (two(3*order+2*q-2)*7+two(3*order+q-2)*7
          +two(order+4*q+1)+two(order+3*q-1)*7
          -two(order+2*q-2)*21-two(order+q-2)*15)/21)
        cerr << "C error, C= " << Cq << endl;
    if (Aq + APrimeq != (two(2*order)-1)*Nq)
        cerr << "A+A' error" << endl;
    if (Dq + DPrimeq != Aq+APrimeq)
        cerr << "D+D' = A + A' error" << endl;
    ftmp=(pow(2.,5*q-3)*5.+3.*pow(2.,4*q)+7.*pow(2.,3*q-2)
        -pow(2.,2*q-3)*33-pow(2.,q-2)*5.)/21.;
    if (Aq != ftmp)
        cerr << "A error, A= " <<Aq<<" formula=" << ftmp << endl;
    ftmp=(pow(2.,5*q-3)*11.+9.*pow(2.,4*q-1)+7.*pow(2.,3*q-2)
        -pow(2.,2*q-3)*39-pow(2.,q-2)*11.)/21.; //D'(q,q)
    if (DPrimeq != Nq*(two(2*order)-two(2*q))/3+ftmp)
        cerr << "D' error, D'= " << DPrimeq << " formula="
        << Nq*(two(2*order)-two(2*q))/3+ftmp << endl;
    }
    cout << "A(q)="<<Aq <<" A'(q)="<<APrimeq
        << " D(q)="<<Dq <<" D'(q)="<<DPrimeq << endl;
    return DPrimeq;
}

// distance for 1-normed distance
long theorem3_1(int q) {
    long delta=two(q), sumDist=0, sumNeighbor=0;
    double ftmp; //tmp variable for formula
    int ni; // ni-th upper or lower neighbor
    for (int i=0; i< getWidth() ; i++)
    for (int j=0; j<getWidth(); j++) {
        if (i+delta<getWidth()) {
            sumDist += abs(
            SFCArrayNode(i,j)->getSFCCode() -
              SFCArrayNode(i+delta,j)->getSFCCode());
            sumNeighbor++;
        }
```

```
    if (i-delta>=0) {
        sumDist += abs(
          SFCArrayNode(i,j)->getSFCCode() -
          SFCArrayNode(i-delta,j)->getSFCCode());
        sumNeighbor++;
    }
    for (ni=delta-1; ni>=0; ni--){ // upper neighbors
        if (i+ni<getWidth() && j+delta-ni<getWidth()) {
            sumDist +=abs(SFCArrayNode(i,j)->getSFCCode()-
              SFCArrayNode(i+ni,j+delta-ni)->getSFCCode());
            sumNeighbor++;
        }
        if (i+ni<getWidth() && j-delta+ni>=0) {
            sumDist +=abs(SFCArrayNode(i,j)->getSFCCode()-
              SFCArrayNode(i+ni,j-delta+ni)->getSFCCode());
            sumNeighbor++;
        }
    }
    for (ni=1; ni<=delta-1; ni++){ // lower neighbors
        if (i-ni>=0 && j+delta-ni<getWidth()) {
            sumDist += abs(SFCArrayNode(i,j)->getSFCCode()-
              SFCArrayNode(i-ni,j+delta-ni)->getSFCCode());
            sumNeighbor++;
        }
        if (i-ni>=0 && j-delta+ni>=0) {
            sumDist += abs(SFCArrayNode(i,j)->getSFCCode()-
              SFCArrayNode(i-ni,j-delta+ni)->getSFCCode());
            sumNeighbor++;
        }
    }
}
sumDist /= 2;//everyone had been summed twice
if (mSFC==0) { //Hilbert curve
if (q==0)
    ftmp = (51*two(3*order)-35*two(2*order)-16)/42;
else
    ftmp = pow(2.,3*order+2*q)*17./14.-(8.*3.*25.*7.*
      (order-q)+35.*383.)/(16.*27.*35.)*pow(2.,2*order+3*q)
      +(30.*(order-q)-1.)*pow(2.,2*order+q)/(4.*27.)-164.*
      pow(2.,5*q)/(27.*35.)-2*pow(2.,3*q)/27.-2*pow(2.,q)/15.;
} else { // z-order curve
if (q==0)
    ftmp = (two(3*order)-two(order));
else
```

```
        ftmp = pow(2.,3*order+2*q)-
          ((order-q)*2./9.+1949/(32.*27.*7.))*pow(2.,2*order+3*q)
          +(2./9.*(order-q)+7/(4.*27.))*pow(2.,2*order+q)+
          19./(4.*3.*7.)*pow(2.,2*order)-4./7.*pow(2.,order+4*q)
          -3./7.*pow(2.,order+q)+10./(27.*7.)*pow(2.,5*q)
          -pow(2.,3*q+2)/27.+pow(2.,2*q+1)/21.;
    }
    if (ftmp != sumDist)
        cerr << "Ldelta error, formula= " << ftmp << endl;
    cout <<"Total distance(delta="<<delta<<"):" << sumDist
        << " Sum of neighbor:" << sumNeighbor <<endl;

     return sumDist;
}

double pnorm(VectorD2D *v, VectorD2D *u, double p) {
    return pow(pow(abs(v->getVofDim(0)-u->getVofDim(0)),p)+
        pow(abs(v->getVofDim(1)-u->getVofDim(1)),p),1/p);
}

void theorem4_3and4() {
    VectorD2D *rep1, *rep2, *rep3, *rep4, //representative
        *tmp= new VectorD2D(0,0);
    Node2D *tmpNode, *tmpNode2; //other nodes compared with tmpNode
    double dp, maxV=0., LAN;
    int numberMax=0; // number of representative pairs

    for (int p=1; p<4; p++){ // p-normed value
        maxV=0; tmpNode= SFCArrayNode(tmp); // starting node
      while (tmpNode != NULL ) {
        tmpNode2 = SFCArrayNode(tmpNode->getNext());
        while (tmpNode2!=NULL) {
          dp=pnorm(tmpNode->getCoor(),tmpNode2->getCoor(),p);
          LAN=dp*dp/(tmpNode2->getSFCCode()-tmpNode->getSFCCode());
          if (LAN > maxV){
            maxV=LAN; numberMax=1;
            rep1=tmpNode->getCoor();
            rep2=tmpNode2->getCoor();
          } else if (LAN == maxV) {// another pair
            numberMax++;
            rep3=tmpNode->getCoor();
            rep4=tmpNode2->getCoor(); //keep 2 pairs only.
          }
          tmpNode2=SFCArrayNode(tmpNode2->getNext());
```

```
        }
        tmpNode = SFCArrayNode(tmpNode->getNext());
      }
      if ((p==1 &&
            maxV != 9.-3.*pow(2.,-order+3)+pow(2.,-2*order+4))
         || (p>1 && maxV != (pow(2.,2*order-3)-pow(2.,order-1)+
            1./2.)*6./(pow(2.,2*order-3)+1)))
             cerr << " p=" << p << " error " << endl;
      cout << "p="<< p<< " Max value = " << maxV<< endl
          << numberMax <<" representative pair(s) are "
          << rep1->toString() << rep2->toString();
      if (numberMax>1)
        cout << " and " << rep1->toString() << rep2->toString();
      cout << endl ;
    }
    delete tmp; // it was temporary starting point
}

void lemma6_1() {
    long PiT=0, PiL=0, PiB=0, max1, tmp;
    if (mSFC==0) // Hilber curve
    for (int ri=0; ri< getWidth() ; ri++){
        max1=0; // initialize max1
        for (int i=ri; i<getWidth(); i++)
        for (int j=0; j<getWidth(); j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        PiT += max1;
    }
    else //z-order curve
    for (int ri=0; ri< getWidth() ; ri++){
        max1=0; // initialize max1
        for (int i=0; i<=ri; i++)
        for (int j=0; j<getWidth(); j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        PiB += max1;
    }

    for (int rj=0; rj< getWidth() ; rj++){
        max1=0; // initialize max1
        for (int i=0; i<getWidth(); i++)
```

```
        for (int j=0; j<=rj; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        PiL += max1;
    }
    if (mSFC==0)  // Hilbert curve
        cout << "PiT= " << PiT;
    else  // z-order curve
        cout << "PiB= " << PiB;
    cout << "  PiL=" << PiL << endl;
}

void lemma6_2() {
    long barc1=0, barc2=0, barc3=0, max1, tmp;
    int ri, rj, i, j;
    for (ri=0; ri< getWidth() ; ri++)
    for (rj=0; rj< getWidth() ; rj++){
        max1=0; // initialize max1
        for (i=0; i<=ri; i++)
        for (j=0; j<=rj; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        barc1 += max1;
    }
    for (ri=0; ri< getWidth() ; ri++)
    for (rj=0; rj< getWidth() ; rj++){
        max1=0; // initialize max1
        for (i=ri; i<getWidth(); i++)
        for (j=0; j<=rj; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        barc2 += max1;
    }
    for (ri=0; ri< getWidth() ; ri++)
    for (rj=0; rj< getWidth() ; rj++){ //c3area is z-order's c4
        max1=0; // initialize max1
        for (i=ri; i<getWidth(); i++)
        for (j=rj; j<getWidth(); j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
```

```
        barc3 += max1;
    }
    if (mSFC==0) // Hilbert
        cout << "bar_c1= " << barc1 << " bar_c2= " << barc2
            << " bar_c3= " << barc3<< endl;
    else
        cout << "bar_c1= " << barc1 << endl;
}

void lemma6_3(int q) {
    long c1=0, c2=0, c3=0, max1, tmp;
    int ri, rj, i, j;
    for (ri=0; ri< two(q)-1 ; ri++)
    for (rj=0; rj< two(q)-1 ; rj++){
        max1=0; // initialize max1
        for (i=0; i<=ri; i++)
        for (j=0; j<=rj; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                    SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        c1 += max1;
    }
    for (ri=getWidth()-two(q)+2-1; ri< getWidth() ; ri++)
    for (rj=0; rj< two(q)-1 ; rj++){
        max1=0; // initialize max1
        for (i=ri; i<getWidth(); i++)
        for (j=0; j<=rj; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                    SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        c2 += max1;
    }
    for (ri=getWidth()-two(q)+2-1; ri< getWidth() ; ri++)
    for (rj=getWidth()-two(q)+2-1; rj< getWidth() ; rj++){
        max1=0; // initialize max1
        for (i=ri; i<getWidth(); i++)
        for (j=rj; j<getWidth(); j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                    SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        c3 += max1;  // z-order's c4
    }
    if (mSFC==0) //Hilbert
        cout << "c1= " << c1 << " c2= " << c2
```

```
            << " c3= " << c3<< endl;
    else //z-order
        cout << "c1= " << c1 << endl;
}

void lemma6_4(int q) {
    long L=0, R=0, B=0, max1, tmp;
    int ri, rj, i, j;
    for (ri=0; ri< getWidth()-two(q)+1 ; ri++)
    for (rj=0; rj< two(q)-1 ; rj++){
        max1=0; // initialize max1
        for (i=ri; i<=ri+two(q)-1; i++)
        for (j=0; j<=rj; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        L += max1;
    }
    for (ri=0; ri< getWidth()-two(q)+1 ; ri++)
    for (rj=getWidth()-two(q)+2-1; rj< getWidth() ; rj++){
        max1=0; // initialize max1
        for (i=ri; i<=ri+two(q)-1; i++)
        for (j=rj; j<getWidth(); j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        R += max1;
    }
    for (ri=0; ri< two(q)-1 ; ri++)
    for (rj=0; rj< getWidth()-two(q)+1; rj++){
        max1=0; // initialize max1
        for (i=0; i<=ri; i++)
        for (j=rj; j<=rj+two(q)-1; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) > max1)
                max1=tmp;
        B += max1;
    }

    cout << "L= " << L << " R= " << R
        << " B= " << B << endl;
}

void lemma6_5(int q) {
```

```
long L=0, R=0, B=0, T=0, min1, tmp;
int ri, rj, i, j;
for (ri=0; ri< getWidth()-two(q)+1 ; ri++)
for (rj=0; rj< two(q)-1 ; rj++){
    min1=getWidth()*getWidth(); // initialize min1
    for (i=ri; i<=ri+two(q)-1; i++)
    for (j=0; j<=rj; j++)
        if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
              SFCArrayNode(0,0)->getSFCCode())) <min1)
            min1=tmp;
    L += min1;
}
for (ri=0; ri< getWidth()-two(q)+1 ; ri++)
for (rj=getWidth()-two(q)+2-1; rj< getWidth() ; rj++){
    min1=getWidth()*getWidth(); // initialize min1
    for (i=ri; i<=ri+two(q)-1; i++)
    for (j=rj; j<getWidth(); j++)
        if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
              SFCArrayNode(0,0)->getSFCCode())) <min1)
            min1=tmp;
    R += min1;
}
for (ri=0; ri< two(q)-1 ; ri++)
for (rj=0; rj< getWidth()-two(q)+1; rj++){
    min1=getWidth()*getWidth(); // initialize min1
    for (i=0; i<=ri; i++)
    for (j=rj; j<=rj+two(q)-1; j++)
        if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
              SFCArrayNode(0,0)->getSFCCode())) < min1)
            min1=tmp;
    B += min1;
}
for (ri=getWidth()-two(q)+2-1; ri< getWidth() ; ri++)
for (rj=0; rj< getWidth()-two(q)+1; rj++){
    min1=getWidth()*getWidth(); // initialize min1
    for (i=ri; i<getWidth(); i++)
    for (j=rj; j<=rj+two(q)-1; j++)
        if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
              SFCArrayNode(0,0)->getSFCCode())) < min1)
            min1=tmp;
    T += min1;
}
 cout << "theta1 L= " << L << " R= " << R
    << " B= " << B << " T= " << T << endl;
```

```
}

void lemma6_6(int q) {
    long c1=0, c2=0, c3=0, min1, tmp;
    int ri, rj, i, j;
    for (ri=0; ri< two(q)-1 ; ri++)
    for (rj=0; rj< two(q)-1 ; rj++){
        min1=getWidth()*getWidth();; // initialize max1
        for (i=0; i<=ri; i++)
        for (j=0; j<=rj; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) <min1)
                min1=tmp;
        c1 += min1;
    }
    for (ri=getWidth()-two(q)+2-1; ri< getWidth() ; ri++)
    for (rj=0; rj< two(q)-1 ; rj++){
        min1=getWidth()*getWidth();; // initialize max1
        for (i=ri; i<getWidth(); i++)
        for (j=0; j<=rj; j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) <min1)
                min1=tmp;
        c2 += min1;
    }
    for (ri=getWidth()-two(q)+2-1; ri< getWidth() ; ri++)
    for (rj=getWidth()-two(q)+2-1; rj< getWidth() ; rj++){
        min1=getWidth()*getWidth();; // initialize max1
        for (i=ri; i<getWidth(); i++)
        for (j=rj; j<getWidth(); j++)
            if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                  SFCArrayNode(0,0)->getSFCCode())) <min1)
                min1=tmp;
        c3 += min1;
    }
    // area of c4 for z-order is the area of c3 for Hilbert
    if (mSFC==0)
        cout << "theta1  c1= " << c1 << " c2= " << c2
            << " c3= " << c3<< endl;
    else
        cout << "theta1  c4= " << c3 << endl;
}

void theorem6_1(int q) {
```

```
        long Psiq=0, min1, max1, tmp;
        int ri, rj, i, j;
        for (ri=0; ri< getWidth()-two(q)+1 ; ri++)
        for (rj=0; rj< getWidth()-two(q)+1 ; rj++){
            min1=getWidth()*getWidth(); // initialize min1
            max1=0; // initialize max1
            for (i=ri; i<=ri+two(q)-1; i++)
            for (j=rj; j<=rj+two(q)-1; j++) {
                if ((tmp= abs(SFCArrayNode(i,j)->getSFCCode() -
                     SFCArrayNode(0,0)->getSFCCode())) <min1)
                    min1=tmp;
                if (tmp>max1)
                    max1=tmp;
            }
            Psiq += (max1-min1-two(q)*two(q)+1);
        }
        cout << "Psiq= " << Psiq << endl;
    }
};
#endif
```

B.1.4 2D.cpp

```
//////////////////////////////////////////////////////////////////////
// 2D.cpp: the interface
//
// Note: all the indices in program starts from 0
//////////////////////////////////////////////////////////////////////
#include <iostream>
#include "Node2D.h"
#include "VectorD2D.h"
#include "SFCArray2D.h"

SFCArray2D *HA;

void init (int nLevel, int sfctype) {
    long tmp =(long) pow(2.0,nLevel);
    HA = new SFCArray2D(tmp ,nLevel, sfctype);
}

int main(char * s[]) {
    int q=2; // query size 2^q * 2^q, or distance 2^q, q must be >= 0
    int sfctype=1; //0: Hilbert, 1:z-order
    for (int k=1; k<10; k++) {
```

```
        init(k, sfctype);
        if (sfctype==0)
            cout << "============ Hilbert Curve ===========" << endl;
        else
            cout << "============ z-Order Curve ===========" << endl;
        if (HA->IsSuccessful())    {
            return -1;
        }
        cout << "k: " << k << "      q= " << q << endl;
//        HA->Check();
        cout << "Lemma 3.1 (3.7 for z-order)" << endl;
        HA->lemma3_1(0); // parameter:row number (indexed from 0)
        cout << "Lemma 3.2 (3.8 for z-order)" << endl;
        HA->lemma3_2(0);  // parameter:column number (indexed from 0)
        cout << "Lemma 3.3 (3.9 for z-order)" << endl;
        HA->lemma3_3();
        cout << "Lemma 3.5 (3.11 for z-order)" << endl;
        HA->lemma3_5(q);
        cout << "Lemma 3.6 (3.12 for z-order)" << endl;
        HA->lemma3_6(q);
        cout << "Theorem 3.1 (3.2 for z-order)" << endl;
        HA->theorem3_1(q);
        cout << "Theorem 4.3 and 4.4" << endl; //Not for z-order
        HA->theorem4_3and4();
        cout << "Lemma 6.1 (6.7 for z-order)" << endl;
        HA->lemma6_1();
        cout << "Lemma 6.2 (6.8 for z-order)" << endl;
        HA->lemma6_2();
        cout << "Lemma 6.3 (6.9 for z-order)" << endl;
        HA->lemma6_3(q);
        cout << "Lemma 6.4 (6.10 for z-order)" << endl;
        HA->lemma6_4(q);
        cout << "Lemma 6.5 (6.11 for z-order)" << endl;
        HA->lemma6_5(q);
        cout << "Lemma 6.6 (6.12 for z-order)" << endl;
        HA->lemma6_6(q);
        cout << "Theorem 6.1 (6.2 for z-order)" << endl;
        HA->theorem6_1(q);

        delete HA;
    }
    return 0;
}
```

B.2 3-Dimensional Space-Filling Curves

B.2.1 VectorD3D.h

```
//////////////////////////////////////////////////////////////////////
// VectorD3D: a class for 3D vector.
//
// Note: all the indices in program starts from 0
//////////////////////////////////////////////////////////////////////
#ifndef __VECTORD3D_H
#define __VECTORD3D_H
#include <iostream>
#include <sstream>
#include <string>
#include <math.h>
using namespace std ;

class VectorD3D {
protected:
    int mNumDim;
    long mE[3];     // the value for each dimension

public:
    VectorD3D():mNumDim(3) {//constructor
        for (int i=0; i<mNumDim; i++)
            mE[i]=-1;
    }

    VectorD3D(VectorD3D *other):mNumDim(3) {  // constructor
        for (int i=0; i<mNumDim; i++)
            mE[i]=other->mE[i];
    }

    VectorD3D(int a, int b, int c):mNumDim(3){ // for coordinate
        mE[0]=a;
        mE[1]=b;
        mE[2]=c;
    }

    ~VectorD3D() { /*delete[] mE;*/ }    // destructor

    int getnDim(){    // # of dimension
        return mNumDim;
    }
```

```
int getVofDim(int dim) {//value in each dimension
    if (dim > getnDim())
        return -9999;
    return mE[dim];
}

// addition
VectorD3D add (VectorD3D &other) {
    VectorD3D tmp(this);
    int nshortest = getnDim();
    if (getnDim() >= other.getnDim())
        nshortest = other.getnDim();    // get the shortest
    for (int i=0; i< nshortest; i++) {
        tmp.mE[i] += other.mE[i];
    }
    return tmp;
}

// subtraction
VectorD3D subtract (VectorD3D &other) {
    VectorD3D tmp(this);
    int nshortest = getnDim();
    if (getnDim() >= other.getnDim())
        nshortest = other.getnDim();    // get the shortest
    for (int i=0; i< nshortest; i++) {
        tmp.mE[i] -=  other.mE[i];
    }
    return tmp;
}

// multiply
VectorD3D Multiply (int num) {
    VectorD3D tmp(this);
    for (int i=0; i< tmp.getnDim(); i++)
        tmp.mE[i] *= num;
    return tmp;
}

VectorD3D Multiply (double num) {
    VectorD3D tmp(this);
    for (int i=0; i< tmp.getnDim(); i++)
        tmp.mE[i] = (int) tmp.mE[i] * num;
    return tmp;
```

```
}

VectorD3D MultiplyLeftShiftHalf (double num) {
    VectorD3D tmp(this);
    double tmpDouble =0.0;
    for (int i=0; i< tmp.getnDim(); i++) {
        tmpDouble = ((double) tmp.mE[i]) * num;
        if (-1 < tmpDouble  && tmpDouble <0)
            tmp.mE[i] = -1;
        else
            tmp.mE[i] = (int)( tmp.mE[i] * num);
    }
    return tmp;
}

// dot opertation
long dot (VectorD3D *other) {
    int tmp = 0;
    for (int i=0; i< getnDim(); i++)
        tmp += mE[i] * other->mE[i];
    return tmp;
}

long dot (VectorD3D &other)    {
    int tmp = 0;
    for (int i=0; i< getnDim(); i++)
        tmp += mE[i] * other.mE[i];
    return tmp;
}

VectorD3D& copyFrom(VectorD3D &other){//copy, return -1 for error
    int nshortest = getnDim();
    if (getnDim() >= other.getnDim())
        nshortest = other.getnDim();    // get the shortest

    for (int i=0; i< nshortest; i++)
        mE[i] = other.mE[i];
    return *this;
}

VectorD3D& copyFrom(VectorD3D *other){    // constructor
    int nshortest = getnDim();
    if (getnDim() >= other->getnDim())
        nshortest = other->getnDim();    // get the shortest
```

```
    for (int i=0; i< nshortest; i++)
        mE[i] = other->mE[i];
    return *this;
}

VectorD3D & operator = ( VectorD3D &other) {
    if (this != &other) {
        for (int i=0; i<mNumDim; i++)
            mE[i]=other.mE[i];
    }
    return *this;
}

VectorD3D Normalized()    {
    VectorD3D unitVector(*this);
    long sqr_len=0;
    int i;

    for (i=0; i<mNumDim; i++)    //sqr sum
        sqr_len += (long) pow(mE[i], 2.0);

    sqr_len = (long) sqrt(sqr_len); // length(distance)

    if (sqr_len != 0) {
        for (i=0; i<mNumDim; i++)
            unitVector.mE[i] /= sqr_len;
    }
    return unitVector;
}

string toString() {
    std::ostringstream o;
    if (o << *this)
        return o.str();
    return "";
}

friend std::ostream& operator<< (std::ostream& o,  VectorD3D& v){
    o << "(" << v.getVofDim(0);
    for (int i=1; i< v.getnDim() ; i++)
        o <<  ", " << v.getVofDim(i) ;
    return o << ")";
}
```

```
};
#endif
```

B.2.2 Node3D.h

```
//////////////////////////////////////////////////////////////////////
// Node3D: a class for the element in the array.
//   data: coordinates, index, next element
// Note: all the indices in program starts from 0
//////////////////////////////////////////////////////////////////////
#ifndef __NODE3D_H
#define __NODE3D_H
#include <iostream>
#include <sstream>
#include <string>
#include "VectorD3D.h"
using namespace std ;

class Node3D  {
private:
    VectorD3D *mData;
    VectorD3D *mNext;
    long mSFCCode;
    int mChecked;
public:
    Node3D() {    //constructor
        mData = new VectorD3D(-1,-1,-1);
        mNext = NULL;
        mSFCCode = -1;
        mChecked=0;
    }

    Node3D(VectorD3D *data, VectorD3D *next, long hcode){//constructor
        if (data != NULL)
            mData = new VectorD3D(data);
        else
            mData=NULL;
        if (next != NULL)
            mNext = new VectorD3D(next);
        else
            mData = NULL;
        mSFCCode = hcode;
        mChecked=0;
    }
```

```
~Node3D() {
    delete mData;
    delete mNext;
}

void setData(long a, long b, long c) {
    VectorD3D tmp(a,b, c);
    if (mData == NULL)
        cerr << "NULL for mData" << endl;
    mData->copyFrom(tmp);
}

VectorD3D* getNext() {
    return  mNext;
}

long getSFCCode() {
    return mSFCCode;
}

void setNext(VectorD3D *next)    {
    if (next != NULL) {
        if (mNext == NULL)
            mNext = new VectorD3D(next);
        else
            mNext->copyFrom( next);
    }
}

void setSFCCode(long hcode)    {
     mSFCCode = hcode;
}

string toString() {
    std::ostringstream o;
    if (o << *this)
        return o.str();
    return "";
}

friend std::ostream& operator<< (std::ostream& o,  Node3D& v) {
    o << "Node: " << *(v.mData) << " (SFCCode=" << v.getSFCCode();
    if (v.mNext == NULL)
```

```
            o << ")  nextNode= (NULL)";
        else
            o << ")  nextNode= " << *(v.mNext) ;
        return o;
    }

    int getVofDim(int dim) { //value in each dimension
        if (dim > mData->getnDim())
            return -9999;
        return mData->getVofDim(dim);
    }

    void setChecked() { // set for mChecked
        mChecked++;
    }
    void setUnChecked() { // set for unchecked
        mChecked=0;
    }
    int getChecked()    {
        return mChecked;
    }
};
#endif
```

B.2.3 SFCArray3D.h

```
//////////////////////////////////////////////////////////////////////
// SFCArray3D: a class for maintaining the indices and coordinates.
//
// Note: all the indices in program starts from 0
//////////////////////////////////////////////////////////////////////
#ifndef __SFCARRAY3D_H
#define __SFCARRAY3D_H
#include <iostream>
#include <stdio.h>
#include <malloc.h>
#include <math.h>
#include "VectorD3D.h"
#include "Node3D.h"
using namespace std ;

class SFCArray3D {
private:
    Node3D* mSFCArray; // array to keep the indices and coordinates
```

```
long mWidth;    // the max width for each coordinate
int order; // order
int mSFC; // 0: Hilbert, 1: z-order
int Successful;

long arrayIndex(VectorD3D *location){
    long arrayIndex=0;
    if (location == NULL)
        return -1;
    for (int i=0; i< location->getnDim(); i++){
        if (location->getVofDim(i) < 0)    {
            arrayIndex = -1;
            break;
        }
        arrayIndex =arrayIndex * mWidth + location->getVofDim(i);
    }
    return arrayIndex;
}

Node3D* SFCArrayNode (VectorD3D *location) {
    if (location != NULL) {
        long arrayInd = arrayIndex(location);
        if (arrayInd < 0)
            return NULL;
        return &(mSFCArray[arrayInd]);
    } else {
        return NULL;
    }
}

Node3D* SFCArrayNode (VectorD3D &location)  {
     long arrayInd = arrayIndex(&location);
    if (arrayInd < 0)
        return NULL;
    return &(mSFCArray[arrayInd]);
 }

Node3D* SFCArrayNode (const long i, const long j, const long k) {
    if (i < mWidth && i>=0 && j <mWidth && j>=0 &&
        k<mWidth && k>=0)
        return &(mSFCArray[(i * mWidth + j)*mWidth +k]);
    else
        return NULL;
}
```

```
public:
    SFCArray3D(int max, int level, int sfctype):mWidth(max),
            Successful(0), mSFC(sfctype), order(level)   {
        if (max < (int)pow(2.0, level)) {
            cerr << "Too few of elements in the array";
            return;
        }
        mSFCArray = new Node3D[max*max*max];
        // initiallize the Node3D
        for (long i = 0; i< mWidth; i++)
          for (long j=0; j< mWidth; j++)
            for (long k=0; k< mWidth; k++)
                mSFCArray[(i*mWidth+j)*mWidth+k].setData(i,j,k);
         VectorD3D o(0,0,0);     // origin
        VectorD3D D0 (max, 0, 0);     // D0 direction
        VectorD3D D1 (0,max, 0);     // D1 direction
        VectorD3D D2 (0, 0, max);     // D2 direction
        // construct the array's nextpoint
        // recusive call to calculate the linkage and the SFC code
        SFCLineArray(o, D0, D1, D2, level, &o);
    }

     ~SFCArray3D() {    delete[] mSFCArray;}

    long getWidth()    {return mWidth;    }

    VectorD3D SFCLineArray(VectorD3D &o, VectorD3D &D0,
          VectorD3D &D1, VectorD3D &D2, int level, VectorD3D *Prev) {
        VectorD3D tmpo(o), tmpD0(D0), tmpD1(D1), tmpD2(D2);
        // copy of o, D0, D1, D2
        if (level <= 0) {
            if (Prev->getVofDim(0)<0 ||    Prev->getVofDim(1)<0 ||
                  Prev->getVofDim(2)<0 ||
                  Prev->getVofDim(0)>=getWidth()  ||
                  Prev->getVofDim(1)>=getWidth() ||
                  Prev->getVofDim(2)>=getWidth()) {
                cerr <<"error, less:" << (*Prev) << endl;
            }

            if (SFCArrayNode(Prev) == NULL)    { // no previous node
                 SFCArrayNode(o)->setNext(&o);
            } else {
                SFCArrayNode(o)->setSFCCode(SFCArrayNode(Prev)->
```

```
            getSFCCode()+1);
        SFCArrayNode(Prev)->setNext(&o);
      }
     return o;
} else    {
    VectorD3D tmpV;
    tmpV=SFCLineArray( tmpo=o,    tmpD1=D1.Multiply(.5),
        tmpD2=D2.Multiply(.5),    tmpD0=D0.Multiply(.5),
        level-1,  Prev); //00 block
    tmpV=SFCLineArray(    tmpo=o.add(D0.Multiply(.5)),
        tmpD2=D2.Multiply(.5),    tmpD0=D0.Multiply(.5),
        tmpD1=D1.Multiply(.5),    level-1, &tmpV); //01 block
    tmpV=SFCLineArray(
        tmpo=o.add(D0.Multiply(.5)).add(D1.Multiply(.5)),
        tmpD2=D2.Multiply(.5), tmpD0=D0.Multiply(.5),
        tmpD1=D1.Multiply(.5),    level-1, &tmpV); //11 block
    tmpV=SFCLineArray(tmpo=o.add(D0.Multiply(.5)).
        subtract(D0.Normalized()).add(D1).
        subtract(D1.Normalized()),    tmpD0=D0.Multiply(-.5),
        tmpD1=D1.Multiply(-.5),  tmpD2=D2.Multiply(.5),
        level-1, &tmpV); //10 block
    // the other side of D2 (3rd D)
    tmpV=SFCLineArray(tmpo=o.add(D0.Multiply(.5)).
        subtract(D0.Normalized()).add(D1).
        subtract(D1.Normalized()).add(D2.Multiply(.5)),
        tmpD0=D0.Multiply(-.5),    tmpD1=D1.Multiply(-.5),
        tmpD2=D2.Multiply(.5),    level-1, &tmpV); //10 block
    tmpV=SFCLineArray(tmpo=o.add(D0.Multiply(.5)).add(D1).
        subtract(D1.Normalized()).add(D2).
        subtract(D2.Normalized()),tmpD2=D2.Multiply(-.5),
        tmpD0=D0.Multiply(.5),     tmpD1=D1.Multiply(-.5),
        level-1, &tmpV); //11 block
    tmpV=SFCLineArray(tmpo=o.add(D0.Multiply(.5)).
        add(D1.Multiply(.5)).subtract(D1.Normalized()).
        add(D2).subtract(D2.Normalized()),
        tmpD2=D2.Multiply(-.5),    tmpD0=D0.Multiply(.5),
        tmpD1=D1.Multiply(-.5),    level-1, &tmpV); //01 block
    tmpV=SFCLineArray(tmpo=o.add(D0.Multiply(.5)).
        subtract(D0.Normalized()).add(D2).
        subtract(D2.Normalized()),    tmpD1=D1.Multiply(.5),
        tmpD2=D2.Multiply(-.5),    tmpD0=D0.Multiply(-.5),
        level-1,  &tmpV); //00 block
    return tmpV;
}
```

```
}

const VectorD3D* getNextPoint(VectorD3D *cur)   {
    if (cur != NULL &&  SFCArrayNode(cur)!= NULL) {
        return SFCArrayNode(cur)->getNext();
    }  else  {
        return NULL;
    }
}

const VectorD3D* getNextV(Node3D *cur) {
    if (cur != NULL ) {
        return cur->getNext();
    }  else {
        return NULL;
    }
}
// return the # of points visited
long Check(){
    VectorD3D *tmp = new VectorD3D(0,0,0);    // start point
    Node3D *tmpNode = SFCArrayNode(tmp);
    long nChecked=0,    // Checked #
        nError=0,         // Error #
        old_hcode = -1;       // cur_SFCCode
    while (tmpNode != NULL ) {
        if (tmpNode->getSFCCode() != old_hcode+1) {
            cout << "Not cotinuous at" << *tmpNode
                << " and prev:" << old_hcode <<endl;
        }
        tmpNode->setChecked();
        if (nChecked>mWidth*mWidth*mWidth) {
            cout << "Error in the setChecked "<< endl;
            return -1;
        }
        cout << tmp->toString() << endl;
        old_hcode = tmpNode->getSFCCode();
        tmp = tmpNode->getNext();    //
        tmpNode = SFCArrayNode(tmp);
    }

    for (int i=0; i< getWidth(); i++)
      for (int j=0; j<getWidth(); j++)
        for (int k=0; k<getWidth(); k++)
           if (SFCArrayNode(i,j, k) != NULL )
```

```
            if (SFCArrayNode(i,j,k)->getChecked() == 1 )
                nChecked++;
            else
                nError--;
    cout << "init:" << *SFCArrayNode(0,0,0) <<
        " \nend:" << *SFCArrayNode(0,0,getWidth()-1) << endl
        <<"nError: "<<nError<<" nChecked:" << nChecked << endl;
    if (nError < 0)
        return nError;
    else
        return nChecked;
}

// created succesfully?
int IsSuccessful()  {  return Successful;  }

double lemma3_13(int r=0) {
    double P1=0.,P2=0.,P3=0.; //for plane13,23, P2 is for comparison
    if (mSFC==0) {
      for (int i=0; i< getWidth() ; i++){
        P1=0.; // initialize P1
        for (int j=0; j<getWidth(); j++)
        for (int k=0; k<getWidth(); k++) {
            P1 += (double) abs(SFCArrayNode(i,j,k)->getSFCCode()-
                  SFCArrayNode(0,0,0)->getSFCCode());
        }
        if (i>0)
          if (P1 != P2)
            cerr << " plane " << i-1 << " != plane " << i <<endl;
        P2=P1;
      }
    }
    else { // z-order
        for (int j=0; j<getWidth(); j++)
            P1 += abs(SFCArrayNode(r,j,0)->getSFCCode() -
                  SFCArrayNode(0,0,0)->getSFCCode());
    }
    P3=P1;
    cout << " Sum at P(23) " << r << " = " << P1 << endl;
    if (mSFC==0) {
      for (int j=0; j< getWidth() ; j++){
        P1=0.; // initialize P1
        for (int i=0; i<getWidth(); i++)
        for (int k=0; k<getWidth(); k++) {
```

```
            P1 += (double) abs(SFCArrayNode(i,j,k)->getSFCCode()-
                  SFCArrayNode(0,0,0)->getSFCCode());
        }
        if (j>0)
          if (P1 != P2)
            cerr << " plane " << j-1 << " != plane " << j <<endl;
        P2=P1;
      }
      if (P3 !=P1 || P3 !=pow(2.,5*order-1)-pow(2.,2*order-1))
        cerr << "P13 != P23 error " << endl;
    }
    else { // z-order
        for (int j=0; j<getWidth(); j++)
            P1 += abs(SFCArrayNode(r,j,0)->getSFCCode() -
                  SFCArrayNode(0,0,0)->getSFCCode());
    }
    cout << " Sum at P(13) " << r << " = " << P1 << endl;
    return P1;
}

double lemma3_14() {
    double P1=0.,P2=0.; //for plane12, P2 is for comparison
    if (mSFC==0) {
      for (int i=0; i< getWidth() ; i++)
        for (int j=0; j<getWidth(); j++)
            P1 += (double) abs(SFCArrayNode(i,j,0)->getSFCCode()-
                  SFCArrayNode(0,0,0)->getSFCCode());
      P2=15./62.*pow(2.,5*order)-pow(2.,2*order-1)+8./31.;
    }
    else { // z-order
        for (int j=0; j<getWidth(); j++)
            P1 += abs(SFCArrayNode(0,j,0)->getSFCCode() -
                  SFCArrayNode(0,0,0)->getSFCCode());
    }
    if (P1 != P2)
        cerr << "P12 0 error" << endl;
    cout << " Sum at P(12) 0= " << P1 << endl;
    return P1;
}

// for  1-normed distance
void theorem3_3() {
    double sumDist=0;
    long sumNeighbor=0;
```

```
        for (int i=0; i< getWidth() ; i++)
          for (int j=0; j<getWidth(); j++) {
            for (int k=0; k<getWidth(); k++) {
              if (i+1 < getWidth())    {
                sumDist += abs(SFCArrayNode(i,j, k)->getSFCCode() -
                            SFCArrayNode(i+1,j, k)->getSFCCode());
                sumNeighbor ++;
              }
              if (j+1 < mWidth) {
                sumDist += abs(SFCArrayNode(i,j, k)->getSFCCode() -
                            SFCArrayNode(i,j+1, k)->getSFCCode());
                sumNeighbor ++;
              }
              if (k+1 < mWidth)    {
                sumDist += abs(SFCArrayNode(i,j, k)->getSFCCode() -
                            SFCArrayNode(i,j, k+1)->getSFCCode());
                sumNeighbor ++;
              }
            }
          }
          if (sumDist != 67./62*pow(2.,5*order)
              -11./14.*pow(2.,3*order)-64./7./31.)
              cerr << "Tatal distance error" << endl;
          cout <<"Total distance(width="<<getWidth()<<"):" << sumDist
              << "  Sum of neighbor:"  << sumNeighbor <<endl;
          cout << "average distance of 1 for HA:"
              << (double)sumDist/sumNeighbor <<endl;
          return;
    }
 };
#endif
```

B.2.4 3D.cpp

```
////////////////////////////////////////////////////////////////////////
// 3D.cpp: the interface
//
// Note: all the indices in program starts from 0
////////////////////////////////////////////////////////////////////////
#include <iostream>
#include "Node3D.h"
#include "VectorD3D.h"
#include "SFCArray3D.h"
```

```
SFCArray3D *HA;

void init (int nLevel,int sfctype) {
    long tmp =(long) pow(2.0,nLevel);
    HA = new SFCArray3D(tmp ,nLevel,sfctype);
}

int main(char * s[]) {
    int sfctype=0; //0: Hilbert, 1: z-order
    for (int k=1; k<8; k++)    {
        init(k,sfctype);
        if (sfctype==0)
            cout << "============ Hilbert Curve ===========" << endl;
        else
            cout << "============ z-Order Curve ===========" << endl;
        if (HA->IsSuccessful())    {
            return -1;
        }
        cout << "k: " << k << endl;
//        HA->Check();
//        cout << "Lemma 3.1 (3.7 for z-order)" << endl;
        cout << "Lemma 3.13 (3.16 for z-order)" << endl;
        HA->lemma3_13(0); // parameter:row number (indexed from 0)
        cout << "Lemma 3.14 (3.17 for z-order)" << endl;
        HA->lemma3_14(); // parameter:row number (indexed from 0)
        cout << "Theorem 3.3 (3.4 for z-order)" << endl;
        HA->theorem3_3(); // parameter:row number (indexed from 0)
        delete HA;
    }
    return 0;
}
```

VITA

Hung-Chi Su

Candidate for the Degree of

Doctor of Philosophy

Thesis: LOCALITY AND CLUSTERING PERFORMANCES OF SPACE-FILLING CURVES

Major Field: Computer Science

Biographical:

Education: Received Bachelor of Chemical Engineering from National Cheng-Kung University, Taiwan in June 1986; received the Master of Science degree at Oklahoma State University, Stillwater, Oklahoma in May 1999 with a major in Computer Science. Completed requirements for the Doctor of Philosophy degree with a major in Computer Science at Oklahoma State University, Stillwater, Oklahoma in December 2003.

Professional Experience: System Analyst, Shin-Kung Computer Service/Hess Bookstore Chains/Best Color Enterprise Co., Taiwan, from 1989 to 1995; Research Assistant, Oklahoma State University, from 1995 to 2003; Teaching Assistant, Oklahoma State University, 1995/1997/1999-2003.