実数列の大域的丸めの数え上げ

Nadia Takki Chebihi　定兼 邦彦　徳山 豪
東北大学大学院情報科学研究科システム情報科学専攻
〒980-8577 仙台市青葉区片平 2-1-1

長さ n の実数列に対し，$[1,n]$ の部分区間からなる区間族に対して，丸め誤差の区間和がどの区間に関しても 1 以下になるものを大域的丸めと呼ぶ．どのような区間族に対しても，大域的丸めが存在することは知られているが，特に，$[1,n]$ の全ての区間からなる区間族に関しては，大域的丸めは $n+1$ 個しかなく，$O(n^2)$ 時間で列挙が出来ること，また，全ての長さ k 以下の区間族に関しては，大域的丸めを一様ランダムに生成するシステム (ネットワーク) を $O(kn)$ 時間で生成できることを示す．

Enumerating all low-discrepancy roundings of a real sequence

Nadia Takki Chebihi　Kunihiko Sadakane　Takeshi Tokuyama
Department of System Information Sciences
Graduate School of Information Sciences, Tohoku University
Katahira 2-1-1, Aoba-ku, Sendai 980-8577, Japan
nadia@na.dais.is.tohoku.ac.jp, {sada,tokuyama}@dais.is.tohoku.ac.jp

In this paper, we discuss the problem of computing all the integral sequences obtained by rounding an input real valued sequence such that the discrepancy between the input sequence and each output integral sequence is less than one. We show that the number of such roundings is $n+1$ if we consider the discrepancy with respect to the set of all subintervals, and give an efficient algorithm to report all of them. Then, we give an optimal method to construct a compact graph to represent the set of global roundings satisfying a weaker discrepancy condition.

1 Introduction

For a given real number α, its *rounding* is either $\lfloor \alpha \rfloor$ or $\lceil \alpha \rceil$. Given a sequence $\boldsymbol{a} = (a_i)_{1\le i\le n}$ of real numbers, its rounding is an integral sequence $\boldsymbol{b} = (b_i)_{1\le i\le n}$ such that each entry b_i is a rounding of a_i. Without loss of generality, we can assume that each entry of $\boldsymbol{a}$ is in the closed interval $[0,1]$. Thus, the rounding of $\boldsymbol{a}$ becomes a binary array.

There are 2^n possible roundings of a given $\boldsymbol{a}$, and we would like to compute good-quality roundings with respect to a given criterion. The problem is not only combinatorially interesting but also related to coding theory, data compression, computer vision, operations research, and Monte Carlo simulation.

In order to give a criterion to determine quality of roundings, we define a distance in the space $\mathcal{A}$ of all $[0,1]$-valued sequences of n real numbers. For an element $\boldsymbol{a} \in \mathcal{A}$, let $\boldsymbol{a}(I) = \sum_{i\in I} a_i$ be the sum of entries of $\boldsymbol{a}$ whose indices are located in an interval $I \subset [1,n]$.

We fix a family of $\mathcal{F}$ of intervals. The l_∞ distance between two elements $\boldsymbol{a}$ and $\boldsymbol{a}'$ in $\mathcal{A}$ with respect to $\mathcal{F}$ is defined by

$$Dist_\infty^{\mathcal{F}}(\boldsymbol{a}, \boldsymbol{a}') = \max_{I \in \mathcal{F}} |\boldsymbol{a}(I) - \boldsymbol{a}'(I)|.$$

$Dist_\infty^{\mathcal{F}}(\boldsymbol{a}, \boldsymbol{b})$ is the rounding error of a rounding $\boldsymbol{b}$ of a given $[0,1]$-valued sequence $\boldsymbol{a}$ measured by using the distance. The supremum of the optimal rounding error $\sup_{\boldsymbol{a} \in \mathcal{A}} \min_{\boldsymbol{b} \in \mathcal{B}} Dist_\infty^{\mathcal{F}}(\boldsymbol{a}, \boldsymbol{b})$ is called the *inhomogeneous discrepancy* of $\mathcal{A}$ with respect to the family $\mathcal{F}$ [3]. Here, $\mathcal{B}$ is the set of all binary valued sequences of length n. The most popular case is where $\mathcal{F}$ is the set $\mathcal{I}_n$ of all integral subintervals of $[1, n]$, and the discrepancy of with respect to $\mathcal{I}_n$ is sometimes called the *1-dimentional discrepancy* in the literature.

Abusing the notation, we often call the error $Dist_\infty^{\mathcal{F}}(\boldsymbol{a}, \boldsymbol{b})$ the *discrepancy between* $\boldsymbol{a}$ *and* $\boldsymbol{b}$ with respect to $\mathcal{F}$.

We say that a rounding $\boldsymbol{b}$ of $\boldsymbol{a}$ is an $\mathcal{F}$*-global rounding* if $Dist_\infty^{\mathcal{F}}(\boldsymbol{a}, \boldsymbol{b}) < 1$ holds; in other words, $\boldsymbol{b}$ is a global rounding of $\boldsymbol{a}$ if and only if $\boldsymbol{b}[I]$ is a rounding of $\boldsymbol{a}[I]$ for every $I \in \mathcal{F}$. It is known that for any $\mathcal{F}$, an $\mathcal{F}$*-global rounding* exists. On the other hand, for any constant $\epsilon > 0$, there exists an input $\boldsymbol{a}$ which has no rounding with a discrepancy less than $1 - \epsilon$ even if we consider the family of all intervals of length 2 [1].

There are two classical algorithms to compute an $\mathcal{F}$-global rounding (the output sequences depend on the algorithms): One is the error-diffusion algorithm, and the other is Viterbi's decoding algorithm. Moreover, Asano et al. [1] have recently shown that for any given input sequence $\boldsymbol{a}$, a binary sequence $\boldsymbol{b}$ minimizing the discrepancy can be computed in time $O(\sqrt{n}|\mathcal{F}|\log^2 n)$, where $|\mathcal{F}|$ is the cardinality of $\mathcal{F}$, and hence $O(n^2)$.

A major defect of the above algorithms is that each of them outputs only one particular $\mathcal{F}$-global rounding. This lack of flexibility causes some serious problems in some applications such as image processing. Therefore, it is desired to design efficient algorithms to output either (1) all $\mathcal{F}$-global roundings or (2) a system so that one can efficiently select a given number of $\mathcal{F}$-global roundings uniformly random from the set of all global roundings.

In this paper, we consider the family $\mathcal{I}_k$ consisting of all intervals of length at most k in $[1, n]$. The family is natural and important in several applications. We first consider the special case where $k = n$, and show that we can report all $\mathcal{I}_n$-global roundings in $O(n^2)$ time. This implies that the number of $\mathcal{I}_n$ global roundings is bounded by a polynomial; indeed, it is at most $n + 1$, and exactly $n + 1$ under a non-degeneracy condition. Next, we give an $O(nk)$ time algorithm to output an acyclic network with $O(nk)$ nodes so that the set of all $\mathcal{I}_k$-global roundings equals the set of all directed s-t paths in the network. As byproducts, we show that several optimization rounding problems that can be solved in $O(2^k qn)$ time by using Viterbi's dynamic programming algorithm can be solved in $O(kqn)$ if we restrict the solution space to the set of $\mathcal{I}_k$ global roundings. Here, q is the time to do some basic operations depending on problems. This includes an improved $O(nk)$ time complexity of computing the rounding $\boldsymbol{b}$ minimizing $Dist_\infty^{\mathcal{I}_k}(\boldsymbol{a}, \boldsymbol{b})$.

The present paper mainly focuses on theoretical aspect of the problem; however, our motivation comes from digital halftoning, which is one of the most fundamental techniques in image processing. An intensity image can be considered as a $[0,1]$-valued $n \times n$ array $\boldsymbol{A}$ where each entry $a_{i,j}$ corresponds to a brightness level (gray level) of the (i, j) pixel of the pixel grid. For a color image, we consider an overlay of three such matrices representing red, green, and blue color components, respectively. The digital halftoning is to compute a binary $n \times n$ array $\boldsymbol{B}$ "approximating" $\boldsymbol{A}$. The intention of this method is to convert a given image which consists of several bits for brightness levels into a binary image having only black and white pixels. This kind of technique is indispensable to print an image on an output device that produces black dots only, such as facsimiles and laser printers. The problem is not easy; for example, neither simple rounding nor randomized rounding (round each entry $a_{i,j}$ to 1 with probability $a_{i,j}$) generates a good halftoning image.

Up to now, a large number of methods and algorithms for digital halftoning have been proposed (see, e.g., [8, 4, 9, 10]). The ordered dither method [10] and the two-dimensional error diffusion method [4] are quite popular methods. By the nature of the problem, we need help of human's decision to judge the quality of halftoning; however, a nice mathematical measurement for automatically evaluating the quality is desired. Discrepancy is a nice mathemetical measurement for the halftoning [11]. However, two dimensional rounding problem minimizing the discrepancy is NP-hard, and even its approximation is theoretically difficult [1, 2].

The concept and algorithms for global roundings given in this paper will be useful tools for designing nice halftoning methods. Every $\mathcal{I}_k$-global rounding (for a suitable k) gives a good quality rounding for each row. However, if we further consider the side-effect, it is not wise to round each row independently and combine them, since it often causes some systematic patterns (that do not exist in the input image) in the output image: Such a pattern is called a *regular pattern* created by a rounding.

We can avoid generating regular patterns if we have many candidate global roundings for each row and select a suitable one considering the relation to the neighbor rows. Even a random choice of a global rounding works well in our preliminary experiments: Compared to the randomized rounding, the method to choose a global rounding randomly in each row decreases the randomness, and hence tends to keep features of the original image better. Moreover, we can consider several bicriteria optimization problems to compute global a rounding of each row that simultaneously minimizes two-dimensional side effects.

2 Structure of the set of global roundings

2.1 Preliminaries

Let $S(\boldsymbol{a},\mathcal{F})$ be the set of all $\mathcal{F}$-global roundings of $\boldsymbol{a}$, and let $N(\boldsymbol{a},\mathcal{F}) = |S(\boldsymbol{a},\mathcal{F})|$ be the number of different roundings. The discrepancy satisfies the monotonicity by definition; i.e., $Dist^{\mathcal{F}}_{\infty}(\boldsymbol{a},\boldsymbol{b}) \geq Dist^{\mathcal{J}}_{\infty}(\boldsymbol{a},\boldsymbol{b})$ if $\mathcal{F} \supset \mathcal{J}$. Therefore, $S(\boldsymbol{a},\mathcal{F}) \subset S(\boldsymbol{a},\mathcal{J})$ if $\mathcal{F} \supset \mathcal{J}$.

For a sequence $\boldsymbol{c}$ of length n, let $\boldsymbol{c}(\leq k)$ be its prefix of length k. Thus, $\boldsymbol{a}(\leq k)$ is the prefix of the input sequence $\boldsymbol{a}$ of length k. Abusing the notation, we say that a binary sequence of length k is a $\mathcal{F}$-global rounding of a prefix of $\boldsymbol{a}$ if it is a global rounding of $\boldsymbol{a}(\leq k)$ with respect to $\mathcal{F}(\leq k) = \{I \cap [1,k] : I \in \mathcal{F}\}$. The following lemma is trivial, but useful:

Lemma 1 *The prefix of length k of a $\mathcal{F}$-global rounding $\boldsymbol{b}$ of $\boldsymbol{a}$ is a $\mathcal{F}$-global rounding of the prefix $\boldsymbol{a}(\leq k)$ of $\boldsymbol{a}$. Moreover, for every $\mathcal{F}$-global rounding $\boldsymbol{c}$ of a prefix $\boldsymbol{a}(\leq k)$, its prefix of length $\ell < k$ is a $\mathcal{F}$-global rounding of the prefix $\boldsymbol{a}(\leq \ell)$.*

Definition 1 *A family $\mathcal{F}$ is called prefix-complete if for any $m \leq n$ and for any $I \in \mathcal{F}$, $I \cap [1,m] \in \mathcal{F}$.*

We mainly consider prefix-complete families in this paper. Obviously, $\mathcal{I}_k$, which we focus on, is a prefix-complete family.

2.2 Rounding graph

Definition 2 *A* rounding graph *of $\boldsymbol{a}$ with respect to $\mathcal{F}$ is a directed acyclic graph G with a source node such that each edge contains either* 0 *or* 1 *as a label, every path from its source to a sink gives a global rounding (if we read the labels at edges on the path sequentially) of $\boldsymbol{a}$, and every global rounding appears as such a path.*

There may be several different roundings graphs for a set of global roundings. We first consider one particular rounding graph (indeed, a binary tree) of an input sequence $\boldsymbol{a}$ with respect to a prefix-complete family $\mathcal{F}$ of intervals. The graph is often called the *keyword tree* in the literature [6], if we consider the set of global roundings as a set of binary keywords.

The construction is as follows: We denote $\boldsymbol{b} \bullet 0$ and $\boldsymbol{b} \bullet 1$ as the sequence obtained by appending 0 and 1 to the end of $\boldsymbol{b}$, respectively. We consider a node $v(\boldsymbol{c})$ associated with an integral sequence $\boldsymbol{c}$, and let $V(\boldsymbol{a}, \mathcal{F}) = \{v(\boldsymbol{c}) : \boldsymbol{c} \text{ is a } \mathcal{F}\text{-global rounding of a prefix of } \boldsymbol{a}\}$. Here, we use a convention that $\emptyset$ is a global rounding of the empty "prefix" of length 0 of $\boldsymbol{a}$. Consider a graph $\tilde{T}(\boldsymbol{a}, \mathcal{F})$, which has $V(\boldsymbol{a}, \mathcal{F})$ as its node set, and has an arc from $v(\boldsymbol{c})$ to $v(\boldsymbol{d})$ if and only if either $\boldsymbol{d} = \boldsymbol{c} \bullet 0$ or $\boldsymbol{d} = \boldsymbol{c} \bullet 1$: the arc has 0 (resp. 1) as its label in the former (resp. latter) case. The following lemma is immediately obtained from the construction and the definition of a prefix-complete family:

Lemma 2 *$\tilde{T}(\boldsymbol{a}, \mathcal{F})$ is a binary directed tree rooted at $v(\emptyset)$ such that if we read the labels at edges on the path from $v(\emptyset)$ to a node $v(\boldsymbol{c})$ sequentially, we have the binary string $\boldsymbol{c}$.*

The depth of the tree $\tilde{T}(\boldsymbol{a}, \mathcal{F})$ is n by the construction, and we ignore the leaves at shallower levels, if any. In precise, let $T(\boldsymbol{a}, \mathcal{F})$ be the induced subgraph of $\tilde{T}(\boldsymbol{a}, \mathcal{F})$ consisting of nodes on the paths from leaves of level n towards the root. $T(\boldsymbol{a}, \mathcal{F})$ is a rounding graph, since the set of paths from the root to leaves of depth n is exactly the set of $\mathcal{F}$-global roundings. Note that the size of the tree may be exponential in general.

3 $\mathcal{I}_n$-global roundings

3.1 Combinatorial results

We consider the case where $\mathcal{F} = \mathcal{I}_n$. If $N(\boldsymbol{a}, \mathcal{F})$ is very large (say, exponential in n), we have no hope to report all the $\mathcal{F}$-global roundings in polynomial time. The following lemma is easy to prove, but it was a surprising discovery for the authors:

Lemma 3 *For any real sequence $\boldsymbol{a}$ of length n, $N(\boldsymbol{a}, \mathcal{I}_n) \leq n + 1$.*

Definition 3 *A real sequence $\boldsymbol{a}$ is called non-degenerate if $a(I)$ is non-integral for every interval $I \in \mathcal{I}_n$.*

Lemma 4 *$T(\boldsymbol{a}, \mathcal{I}_n) = \tilde{T}(\boldsymbol{a}, \mathcal{I}_n)$, and if the sequence $\boldsymbol{a}$ is non-degenerate, $N(\boldsymbol{a}, \mathcal{I}_n) = n+1$.*

These two lemmas imply that, if we apply a symbolic perturbation method to modify the input sequence $\boldsymbol{a}$ such that $\boldsymbol{a}(I)$ is non-integral for every I, we can always have exactly $n + 1$ global roundings of $\boldsymbol{a}$ with respect to $\mathcal{I}_n$.

One natural question is whether we can obtain a polynomial bound of the number of binary sequences if we relax the discrepancy bound. The answer is negative: suppose that we consider the relaxed condition $Dist_{\infty}^{\mathcal{I}_n}(\boldsymbol{a}, \boldsymbol{b}) \leq 1$, instead of $Dist_{\infty}^{\mathcal{I}_n}(\boldsymbol{a}, \boldsymbol{b}) < 1$. Consider the input sequence $\boldsymbol{a}$ of even length whose every entry is 0.5. Then, we can observe that every binary sequences satisfying that $b_{2i-1} + b_{2i} = 1$ for $i = 1, 2, \ldots n/2$ are included in the solution set. There are $2^{n/2}$ such sequences.

3.2 Algorithm for reporting all $\mathcal{I}_n$- global roundings

For the family $\mathcal{I}_n$ of all intervals, we compute all $n + 1$ sequences. We indeed construct the rounding graph $T = T(\boldsymbol{a}, \mathcal{I}_n)$ in $O(n^2)$ time and $O(n)$ working space (ignoring the space to store the tree). The tree T is a binary tree of height n with at most $n + 1$ leaves, and it has $\Theta(n^2)$ nodes.

For simplicity, we simply call a global rounding for an $\mathcal{I}_n$-global rounding in this subsection. For each global rounding $\boldsymbol{c}$ of a prefix (say, $\boldsymbol{a}(\leq i)$) of $\boldsymbol{a}$, let $\mathit{diff}(\boldsymbol{c}) = \boldsymbol{a}([1, i]) - \boldsymbol{c}([1, i])$.

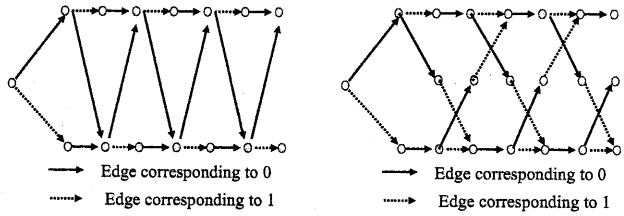

Figure 1: Rounding graphs for $\mathcal{I}_2$ (left drawing) and $\mathcal{I}_3$ (right drawing).

We define $maxdiff(\boldsymbol{c}) = \max\{diff(\boldsymbol{d}) : \boldsymbol{d}$ is a prefix of $\boldsymbol{c}\}$ and $mindiff(\boldsymbol{c}) = \min\{diff(\boldsymbol{d}) : \boldsymbol{d}$ is a prefix of $\boldsymbol{c}\}$.

Starting from $\emptyset$, we construct the tree from top to bottom, increasing the depth one by one. The level which is under construction in the algorithm is called the *current level.* If the current level has a depth i, we construct nodes corresponding to global roundings of $\boldsymbol{a}(\leq i)$. We compute $diff(\boldsymbol{c})$, $maxdiff(\boldsymbol{c})$, and $mindiff(\boldsymbol{c})$ for the nodes in the current level of the tree by using the information of the previous level. Note that $maxdiff(\boldsymbol{c}) < mindiff(\boldsymbol{c}) + 2$ holds.

Suppose that the current level is at depth i, and let $v(\boldsymbol{c})$ be a node of T with depth $i-1$ (the level with depth $i-1$ has been already constructed). We want to decide whether $\boldsymbol{c} \bullet 0$ and/or $\boldsymbol{c} \bullet 1$ are global roundings of $\boldsymbol{a}(\leq i)$. The following result is obtained in a routine way from the definition of a global rounding:

Lemma 5 *Let $\tilde{\boldsymbol{c}}$ be either $\boldsymbol{c} \bullet 0$ or $\boldsymbol{c} \bullet 1$. The sequence $\tilde{\boldsymbol{c}}$ is a global rounding of $\boldsymbol{a}(\leq i)$ if and only if $maxdiff(\boldsymbol{c}) - 1 < diff(\tilde{\boldsymbol{c}}) < mindiff(\boldsymbol{c}) + 1$.*

Since $diff(\boldsymbol{c} \bullet 0) = diff(\boldsymbol{c}) - \boldsymbol{a}(i)$ and $diff(\boldsymbol{c} \bullet 1) = diff(\boldsymbol{c}) + 1 - \boldsymbol{a}(i)$, they can be computed in $O(1)$ time. Thus, we can decide in $O(1)$ time whether $\tilde{\boldsymbol{c}}$ is a global rounding or not. it is easy to see that $maxdiff(\tilde{\boldsymbol{c}})$ and $mindiff(\tilde{\boldsymbol{c}})$ can be computed in $O(1)$ time. Hence, we spend $O(1)$ time to creating a node in the graph T. Thus, the time complexity of our algorithm is $O(n^2)$. Since we only use the information stored in the $(i-1)$-th revel to compute the i-th level, we use $O(n)$ working space (ignoring the space to store the output).

3.3 Compact rounding graph for a smaller family of intervals

In some applications, we do not care very long intervals. Hence, instead of $\mathcal{I}_n$, we would like to consider $\mathcal{I}_k$ for $k < n$. Unfortunately, the number of $\mathcal{I}_k$-global roundings is $\Omega((k+1)^{\lfloor n/2k \rfloor})$, and hence exponential in $n/2k$. Therefore, it is too expensive to report all the $\mathcal{I}_k$-global roundings explicitly. Instead, we construct a rounding graph of size $O(nk)$, so that we can generate global roundings in a uniformly random fashion.

Let us learn from the following simple example: Consider a fixed input $\boldsymbol{a} = (0.4, 0.4, \ldots, 0.4)$ consisting of n entries with a value 0.4. A binary string is an $\mathcal{I}_2$-global rounding of $\boldsymbol{a}$ if and only if it contains no two consecutive entries 1, 1. Such binary sequences correspond to vertices of Fibonacci cube [7], and the number of such sequences equals the $(n+2)$-th Fibonacci number; Hence it is exponential. However, we have a compact rounding graph with $2n+1$ nodes illustrated in the left drawing of the Figure 1. If we consider $\mathcal{I}_3$, we have a rounding graph in the right drawing.

Theorem 1 *For any input sequence $\boldsymbol{a}$, we can construct its rounding graph with at most $nk + 1 - \lceil k(k+1)/2 \rceil$ nodes representing the set of all $\mathcal{I}_k$-global roundings.*

The rest of this subsection is devoted to the proof of the above theorem. The proof is constructive, and similar to the construction of a BDD (bounded decision diagram) from a decision tree. First, we consider the tree $T = \tilde{T}(\boldsymbol{a}, \mathcal{I}_k)$ defined in the previous section. We say two sequences c and c' are $(k-1)$-similar to each other if they have the same length $\ell \geq k-1$, and they have the same suffix of length $k-1$. The equivalence class of a sequence $\boldsymbol{c}$ under the $(k-1)$-similarlity is denoted by $class(\boldsymbol{c})$. In this subsection, we concentrate on the family $\mathcal{I}_k$, and hence simply write "global roundings" for $\mathcal{I}_k$-global roundings.

Two nodes $v(c)$ and $v(c')$ in T are called similar to each other if c and c' are $(k-1)$-similar. The following claim is easy to verify:

Claim A: If v and v' in T are similar, there is an one-to-one matching between the set of descendants of v and that of v' such that each matching nodes are similar to each other.

We fold the tree T to obtain a graph $G(\boldsymbol{a}, \mathcal{I}_k)$ such that similar nodes are identified and unified into a single node of $G(\boldsymbol{a}, \mathcal{I}_k)$. The edges of T is also unified without causing conflict because of Claim A. Inherited fom T, The graph $G(\boldsymbol{a}, \mathcal{I}_k)$ is a layered directed acyclic graph with $n+1$ layers. From the definition of similarity, the unified edges should have the same label. Due to Claim A, all the outgoing edge with a same label must be unified; thus, each node has at most two outgoing edges. Also, each edge has a label 0 or 1 inherited from T without causing any conflict.

From Lemma 3.1, there are at most k different binary sequences which is a global rounding of a subsequence $a_i, a_{i+1}, \ldots, a_{i+k-2}$ with respect to $\mathcal{I}_{k-1}$. Hence, at each layer of T, there are at most k different suffixes of the sequences associated to node in the layer. Hence, there are at most k nodes in each layer of G. We can also easily see that the first i-th layer has at most $i+1$ nodes for $i \leq k-1$. This proves the theorem.

3.4 Algorithm to compute a compact rounding graph

We want to compute $G(\boldsymbol{a}, \mathcal{I}_k)$ efficiently. Since, $\mathcal{I}_k$ is prefix complete, we can apply a similar sweeping strategy to the case of $\mathcal{I}_n$.

Each node of $G(\boldsymbol{a}, \mathcal{I}_k)$ corresponds to an equivalence class of a prefix of $\boldsymbol{a}$, and wrote as $v(\boldsymbol{c})$, where $\boldsymbol{c}$ is the representative of the equivalence class, which is the lexicographically smallest member (in other words, the smallest member if we regard binary sequences as integers in binary forms) in the class.

Starting from $\emptyset$, we construct $G(\boldsymbol{a}, \mathcal{I}_k)$ from the source to sinks, increasing the level (i.e., depth) one by one. If the current level has depth i, we construct vertices corresponding equivalence classes of the global roundings of $\boldsymbol{a}(\leq i)$. As we have shown in the previous subsection, there are at most k such equivalence classes. We maintain $\mathit{diff}(\boldsymbol{c})$, $\mathit{maxdiff}_k(\boldsymbol{c})$, and $\mathit{mindiff}_k(\boldsymbol{c})$ for the representative $\boldsymbol{c}$ of the equivalence class corresponding to each node in the current level of the graph by using the information of the previous level. Let $L(m)$ be the set of representatives of the equivalence classes corresponding to nodes of the m-th level of $G(\boldsymbol{a}, \mathcal{I}_k)$.

Let $\ell(\boldsymbol{c})$ be the length of a sequence $\boldsymbol{c}$. We define

$$\begin{aligned} \mathit{maxdiff}_k(\boldsymbol{c}) &= \max\{\mathit{diff}(\boldsymbol{d}) : \boldsymbol{d} \text{ is a prefix of } \boldsymbol{c} \text{ such that } \ell(\boldsymbol{d}) \geq \ell(\boldsymbol{c}) - k + 1\} \text{ and} \\ \mathit{mindiff}_k(\boldsymbol{c}) &= \min\{\mathit{diff}(\boldsymbol{d}) : \boldsymbol{d} \text{ is a prefix of } \boldsymbol{c} \text{ such that } \ell(\boldsymbol{d}) \geq \ell(\boldsymbol{c}) - k + 1\}. \end{aligned}$$

Lemma 6 *If $\boldsymbol{c} = (c_1, c_2, \ldots, c_m)$ is a prefix of a global rounding with respect to $\mathcal{I}_k$, $\boldsymbol{c} \bullet c_{m+1}$ ($c_{m+1} = 1$ or 0) is a prefix of a global rounding if and only if $\mathit{maxdiff}_{k-1}(\boldsymbol{c}) + a_{m+1} - 1 < c_{m+1} < \mathit{mindiff}_{k-1}(\boldsymbol{c}) + a_{m+1} + 1$*

Hence, we can select all the global roundings among $\{\boldsymbol{c} \bullet 0 : \boldsymbol{c} \in L(m)\}$ and $\{\boldsymbol{c} \bullet 1 : \boldsymbol{c} \in L(m)\}$ in $O(k)$ time. Thus, we can construct $G(\boldsymbol{a}, \mathcal{I}_k)$ in $O(nk + nq)$ time if the following operations can be done in $O(q)$ amortized time for each level: (1): Classify the set of global roundings among $\{\boldsymbol{c} \bullet 0 : \boldsymbol{c} \in L(m)\} \cup \{\boldsymbol{c} \bullet 1 : \boldsymbol{c} \in L(m)\}$ into equivalence classes, and choose representatives. (2): Compute information of diff, $\mathit{mindiff}_k$ and $\mathit{maxdiff}_k$ for all representatives in $L(m+1)$.

In order to implement the operation (1), we consider a tree $T(m)$ from the set of representatives $\boldsymbol{c}$ in $L(m)$. The tree has a leaf $l(\boldsymbol{c})$ for each $\boldsymbol{c} \in L(m)$, and each edge has either 0 or 1 as its label, and the path from the root to $l(\boldsymbol{c})$ gives the suffix of length $k-1$ of $\boldsymbol{c}$ in the reverse order. For example, if $k = 4$ and $\boldsymbol{c} = 0, 0, 1, 1, 0, 1, 1$, the path from the root gives the sequence $1, 1, 0$. It is clear that $T(m)$ has $O(k^2)$ edges. From $T(m)$, we can construct $T(m+1)$ by making two copies of $T(m)$, joining them at a new root with edges of labels 0 and 1 respectively, remove leaves which do not correspond to global roundings, and upgrades each other leaf to its parent's place. If two leaves are upgraded to the same position (i.e., if they have the same parent), we know that these two leaves are corresponding to sequences with a same equivalence class.

In order to attain the $O(k)$ time complexity, we use a compressed form $H(m)$ of $T(m)$. Since $T(m)$ has only k leaves, it has at most $k-1$ branching nodes. Only at most $O(k)$ cells storing label sequences are updated, and an update of the label sequences is either removing the last bit of the sequence, or appending sequences in two cells; Hence, each such operation can be done in $O(1)$ time. Thus, we can do the operation (1) in $O(k)$ time.

The operation (2) can be implemented in $O(k \log k)$ time by using a dynamic tree data structure. Instead, we do it in $O(k)$ amortized time without using a complicated data structure. Hence, we have obtained the following theorem:

Theorem 2 *The graph $G(\boldsymbol{a}, \mathcal{I}_k)$ can be constructed in $O(nk)$ time using $O(k^2)$ working space.*

We can compute for every node $v(\boldsymbol{c})$ of $G(\boldsymbol{a}, \mathcal{I}_k)$ the number $n(v(\boldsymbol{c}))$ of global roundings of $\boldsymbol{a}$ that have $\boldsymbol{c}$ as their prefix. This can be done in $O(nk)$ time by using a dynamic programming procedure. By using this information, we can generate global roundings uniformly random by walking on the directed acyclic graph $G(\boldsymbol{a}, \mathcal{I}_k)$ (directed from the source to sinks) using $n(v(\boldsymbol{c}))$ as the probability for choosing the next branch (i.e., next bit of the rounding).

4 Fast Viterbi-type algorithms and bicriteria optimization

Let us review the Viterbi's algorithm in a general form. For each integral subinterval $J = [i+1, i+k] \subset [1, n]$ of length k, let us consider a function f_J assigning a real value $f_J(\boldsymbol{a}, \boldsymbol{x})$ for each pair of a real sequence $\boldsymbol{a} \in [0,1]^n$ and a binary sequence $\boldsymbol{x} \in \{0,1\}^n$ of length n. The function f_J is called *local* if $f_J(\boldsymbol{a}, \boldsymbol{x})$ is determined by the entries of $\boldsymbol{a}$ and $\boldsymbol{x}$ located in the interval J.

Consider a commutative semigroup operation $\oplus$ satisfying the monotonicity, i.e., if $x_1 \geq y_1$ and $x_2 \geq y_2$ then $x_1 \oplus x_2 \geq y_1 \oplus y_2$. Examples of such operations are $+$, max, min, and taking the L_p norm $(|x_1|^p + |x_2|^p)^{1/p}$. Let us consider the sum (under the $\oplus$ operation) $F(\boldsymbol{a}, \boldsymbol{x}) = \bigoplus_{i=0}^{n-k} f_{[i+1,i+k]}(\boldsymbol{a}, \boldsymbol{x})$, and would like to find a binary sequence $\boldsymbol{x}$ minimizing $F(\boldsymbol{a}, \boldsymbol{x})$.

Viterbi's dynamic programming algorithm can be applied to the above problem. It is easy to see the following: Suppose that $f_J(\boldsymbol{a}, \boldsymbol{x})$ is local and computable in $O(q)$ amortized time if we run the dynamic programming. Then, the binary sequence x minimizing $F(\boldsymbol{a}, \boldsymbol{x})$ can be computed in $O(2^k nq)$ time. If we further combine our global rounding condition, we have the following:

Theorem 3 *Under the assumption as above, the global rounding sequence x of $\boldsymbol{a}$ with respect to $\mathcal{I}_k$ minimizing $F(\boldsymbol{a}, \boldsymbol{x})$ can be computed in $O(knq)$ time.*

Corollary 1 *The rounding minimizing the L_∞ rounding error with respect to $\mathcal{I}_k$ can be computed in $O(kn)$ time.*

For a family of interval $\mathcal{F}$, we can consider a nonnegative valued function w on $\mathcal{F}$ and define the weighted l_p distance $Dist_p^{\mathcal{F},w}(\boldsymbol{a},\boldsymbol{b}) = (\sum_{I\in\mathcal{F}} |\boldsymbol{a}(I) - \boldsymbol{b}(I)|^p w(I))^{1/p}$ between $\boldsymbol{a}$ and its rounding $\boldsymbol{b}$. Although a weighted l_p distance is a nice measure of quality of a rounding if we choose suitable w and p, it is time consuming to compute the optimal rounding with respect to this measure [2]. However, if we restrict the solution space to the set of global roundings with respect to $\mathcal{I}_k$, we have the following:

Corollary 2 *Given any weight function w, the global rounding minimizing the weighted l_p error with respect to $\mathcal{I}_k$ can be computed in $O(k^2n)$ time.*

5 Remarks on Digital halftoning applications

From the viewpoint of practical applications, our main target is digital halftoning: We would like to approximate a $[0,1]$-valued matrix $\boldsymbol{A}$ with a binary matrix $\boldsymbol{B}$. One natural formulation is that we define $Dist_\infty^{\mathcal{F}}(\boldsymbol{A},\boldsymbol{B}) = \max_{R\in\mathcal{F}} |\boldsymbol{A}(R) - \boldsymbol{B}(R)|$ for a family F of subarrays, and find $\boldsymbol{B}$ minimizing this distance. However, this problem is NP-hard, and even an approximation algorithm with a provable constant approximation ratio is difficult to design [1]. One heuristics method is to round rows one by one, considering the relations to roundings of previous rows. Here, we must keep the rounding of the current row to be similar to the input sequence (the global rounding property certifies it) to reduce the side-effect of roundings of forthcoming rows, and also minimize the two-dimensional error effect in the part of the matrix rounded so far (together with the current row). For the purpose, the bicriteria method given in the preceding section will be suitable. Our experimental results will be reported elsewhere.

References

[1] T. Asano, T. Matsui, and T. Tokuyama: "On the complexity of the optimal rounding problems of sequences and matrices," *Proceedings of SWAT00, LNCS1851* (2000), pp. 476-489.

[2] T. Asano et al, "Digital Halftoning: Formulation as a combinatorial optimization problem and approximation algorithms based on network flow", working paper, 2000 November.

[3] J. Beck and V. T. Sös, *Discrepancy Theory,* in *Handbook of Combinatorics* Volume II, (ed. R.Graham, M. Grötschel, and L Lovász) 1995, Elsevier.

[4] R. W. Floyd and L. Steinberg: "An adaptive algorithm for spatial gray scale," *SID 75 Digest, Society for Information Display* (1975), pp. 36–37.

[5] H. N. Gabow and R. E. Tarjan: "Faster scaling algorithms for network problems," *SIAM J. Comp.,* 18 (1989), pp. 1013–1036.

[6] D. Gusfield, *Algorithms on Strings, Trees and Sequences: Computer science and computational biology,* Cambridge U.P. 1997.

[7] W. J. Hsu, "Fibonacci cubes – a new interconnection topology," IEEE Trans. Parallel and Distributed Systems, 4 (1993) pp.2-12.

[8] D. E. Knuth: "Digital halftones by dot diffusion," ACM Trans. Graphics, 6-4 (1987), pp. 245–273.

[9] J. O. Limb: "Design of dither waveforms for quantized visual signals," *Bell Syst. Tech. J.,* 48-7 (1969), pp. 2555–2582.

[10] B. Lippel and M. Kurland: "The effect of dither on luminance quantization of pictures," *IEEE Trans. Commun. Tech.,* COM-19 (1971), pp.879-888.

[11] V. Rödl and P. Winkler: "Concerning a matrix approximation problem", *Crux Mathmaticorum,* 1990, pp. 76–79.