IEICE TRANS. FUNDAMENTALS, VOL. E78-A, NO. 8 AUGUST 1995

PAPER *Special Section on Digital Signal Processing*

Parallel Processing Techniques for Multidimensional Sampling Lattice Alteration Based on Overlap-Add and Overlap-Save Methods

Shogo MURAMATSU†, ***Student Member and*** **Hitoshi KIYA**†, ***Member***

SUMMARY In this paper, we propose two parallel processing methods for multidimensional (MD) sampling lattice alteration. The use of our proposed methods enables us to alter the sampling lattice of a given MD signal sequence in parallel without any redundancy caused by up- and down-sampling, even if the alteration is rational and non-separable. Our proposed methods are provided by extending two conventional block processing techniques for FIR filtering: the overlap-add method and the overlap-save method. In these proposed methods, firstly a given signal sequence is segmented into some blocks, secondly sampling lattice alteration is implemented for each block data individually, and finally the results are fitted together to obtain the output sequence which is identical to the sequence obtained from the direct sampling lattice alteration. Besides, we provide their efficient implementation: the DFT-domain approach, and give some comments on the computational complexity in order to show the effectiveness of our proposed methods.

key words: *multidimensional signal processing, multirate signal processing, parallel processing, overlap-add/save method*

1. Introduction

Multidimensional (MD) multirate signal processing, such as MD filter banks and MD wavelet transforms, finds an increasing number of applications in various important areas [1]–[6], for example motion analysis, image sequence processing and so forth [7], [8]. Sampling lattice alteration is an indispensable technique for MD multirate signal processing which deals with MD digital data at several sampling lattices.

Most applications of MD signal processing require real-time processing, and is often accompanied by the problem how to realize a large amount of the computation within few specified processing time. Because of the problem, it is hardly possible to avoid the use of the parallel processing approach, which attempts to increase computing speed by using a number of processor elements (PE) simultaneously [9]–[11]. For finite-extent impulse response (FIR) filtering, the overlap-add method (OLA) and the overlap-save method (OLS) [12]–[14] are exploited as the basic techniques of parallel filtering [10], [11]. Both OLA and OLS are regarded as block processing or data distribution technique, which segments a given signal sequence into some blocks and handles each block independently.

Since sampling lattice alteration includes filtering operation, both OLA and OLS can be applied to the processing. Due to up- and down-sampling involved in the alteration, however, the redundant operations are caused in the approach. In order to eliminate the redundancy, we have to take account of up- and down-sampling. Besides, in order for the effective parallel processing, we have to consider how to operate each segmented block data independently. The parallel sampling lattice alteration based on the above consideration, however, have not been formulated yet.

In this paper, in order to alter MD sampling lattices in parallel without the redundancy caused by up- and down-sampling, we propose two parallel processing methods by extending the conventional OLA and OLS [15]. As a general case, we deal with a rational and non-separable sampling lattice alteration, such as orthogonal-to-quincunx and orthogonal-to-face-centered-orthorhombic (FCO) alterations [7]. The theory of this work is applicable to any number of dimension and is regarded as the generalized version of the one-dimensional (1D) extended OLA and OLS [16], [17].

As a preliminary, in Sect. 2, some fundamentals of MD signal processing and sampling lattice alteration are reviewed. In Sect. 3, the application of OLA and OLS to the filtering of sampling lattice alteration is considered. In Sect. 4, the extension of OLA and OLS to sampling lattice alteration is proposed. And finally, in Sect. 5, the efficient implementation of our proposed methods is provided, followed by the conclusion of this work.

2. Review of Basic Concepts

As a preliminary, we firstly indicate some important notations, and review some fundamentals of MD signal processing and rational sampling lattice alteration.

Let D, $\boldsymbol{N}$ and $\boldsymbol{V}$ be the number of dimensions, a $D \times D$ nonsingular integer matrix and a $D \times D$ nonsingular real matrix, respectively.

Manuscript received January 4, 1995.
Manuscript revised March 8, 1995.

†The authors are with the Faculty of Technology, Tokyo Metropolitan University, Hachioji-shi, 192-03 Japan.

2.1 Notations

All through this work, the following notations are used.

$\mathcal{N}$: set of $D \times 1$ integer vectors.

$\mathcal{N}(\boldsymbol{N})$: set of all integer vectors of the form $\boldsymbol{N}\boldsymbol{x}$ where $\boldsymbol{x}$ is a real vector with components x_i in the range $0 \leq x_i < 1$.

$J(\boldsymbol{N}) \triangleq |\det \boldsymbol{N}|$: the absolute determinant of $\boldsymbol{N}$, which is also equal to the number of elements in $\mathcal{N}(\boldsymbol{N})$.

$\mathcal{L}(\boldsymbol{V})$: set of all vectors of the form $\boldsymbol{V}\boldsymbol{n}, \boldsymbol{n} \in \mathcal{N}$, that is, the lattice generated by $\boldsymbol{V}$.

$((\boldsymbol{n}))_{\boldsymbol{N}}$: the remainder obtained by dividing some integer vector $\boldsymbol{n} \in \mathcal{N}$ with $\boldsymbol{N}$, which is derived from the division theorem for integer vectors.

Some of the above notations are similar to those in Refs. [1]–[4], and more detailed discussions can be found in those articles.

2.2 MD Linear Convolution [4], [13]

MD linear filtering is done by calculating the corresponding MD linear convolution. Let $f(\boldsymbol{k})$ and $h(\boldsymbol{k})$ be a D-dimensional signal sequence and the impulse response of a linear shift-invariant (LSI) system, respectively. The definition of D-dimensional linear convolution is expressed as follows:

$$g(\boldsymbol{k}) = \sum_{\boldsymbol{n} \in \mathcal{N}} f(\boldsymbol{n})h(\boldsymbol{k}-\boldsymbol{n}), \ \boldsymbol{k} \in \mathcal{N}. \tag{1}$$

In the following, this operation is briefly represented as $g(\boldsymbol{k}) = f(\boldsymbol{k}) * h(\boldsymbol{k})$.

Suppose that $h(\boldsymbol{k})$ is an FIR filter defined over a region of support $\mathcal{N}_h \subset \mathcal{N}$. Provided $f(\boldsymbol{k})$ is a signal sequence defined a finite-extent region of support $\mathcal{N}_f \subset \mathcal{N}$, then the result $g(\boldsymbol{k})$ also has a finite-extent and the region of support $\mathcal{N}_g \subset \mathcal{N}$ results in

$$\mathcal{N}_g = \bigcup_{\boldsymbol{n} \in \mathcal{N}_f} \mathcal{N}_{h\boldsymbol{n}}, \tag{2}$$

where $\mathcal{N}_{h\boldsymbol{n}} \triangleq \{\boldsymbol{k} | \boldsymbol{k}-\boldsymbol{n} \in \mathcal{N}_h\}$.

On the contrary, for computing the result $g(\boldsymbol{k})$ only within some region of support $\bar{\mathcal{N}}_g \subset \mathcal{N}$, it is sufficient that there exists the signal $f(\boldsymbol{k})$ only within the region of support $\bar{\mathcal{N}}_f \subset \mathcal{N}$, which is derived from

$$\bar{\mathcal{N}}_f = \bigcup_{\boldsymbol{k} \in \bar{\mathcal{N}}_g} \bar{\mathcal{N}}_{h\boldsymbol{k}}, \tag{3}$$

where $\bar{\mathcal{N}}_{h\boldsymbol{k}} \triangleq \{\boldsymbol{n} | \boldsymbol{k}-\boldsymbol{n} \in \mathcal{N}_h\}$.

2.3 MD Discrete Fourier Transform [13]

Let $\boldsymbol{N}$ be a nonsingular integer matrix and $f(\boldsymbol{n})$ be a D-dimensional signal defined over a region of support $\mathcal{N}(\boldsymbol{N})$. The matrix-$\boldsymbol{N}$ discrete Fourier transform (DFT) of $f(\boldsymbol{n})$ and the inverse discrete Fourier transform (IDFT) are defined as follows, respectively:

$$F(\boldsymbol{k}) = \sum_{\boldsymbol{n} \in \mathcal{N}(\boldsymbol{N})} f(\boldsymbol{n}) \left[\boldsymbol{W}_{\boldsymbol{N}}^{(\mathrm{g})}\right]_{\boldsymbol{k},\boldsymbol{n}}, \quad \boldsymbol{k} \in \mathcal{N}(\boldsymbol{N}^T), \tag{4}$$

$$f(\boldsymbol{n}) = \frac{1}{J(\boldsymbol{N})} \sum_{\boldsymbol{k} \in \mathcal{N}(\boldsymbol{N}^T)} F(\boldsymbol{k}) \left[\boldsymbol{W}_{\boldsymbol{N}}^{(\mathrm{g})}\right]_{\boldsymbol{k},\boldsymbol{n}}^{*}, \quad \boldsymbol{n} \in \mathcal{N}(\boldsymbol{N}), \tag{5}$$

where the superscripts 'T' and '$*$' denote the transposition and the conjugation, respectively. $F(\boldsymbol{k})$ is the matrix-$\boldsymbol{N}$ DFT of $f(\boldsymbol{n})$ and $\boldsymbol{W}^{(\mathrm{g})}$ denotes the generalized DFT matrix [4] defined as

$$\left[\boldsymbol{W}_{\boldsymbol{N}}^{(\mathrm{g})}\right]_{\boldsymbol{k},\boldsymbol{n}} = \exp\left(-j2\pi \boldsymbol{k}^T \boldsymbol{N}^{-1} \boldsymbol{n}\right), \quad \boldsymbol{k} \in \mathcal{N}(\boldsymbol{N}^T), \boldsymbol{n} \in \mathcal{N}(\boldsymbol{N}), \tag{6}$$

where the superscript '-1' denotes the inversion. The matrix $\boldsymbol{N}$ is referred to as a periodicity matrix.

Both of the DFT in Eq. (4) and the IDFT in Eq. (5) can be computed efficiently by using the fast Fourier transform (FFT) algorithms, such as the row-column, vector-radix and prime-factor algorithms [13].

2.4 Sampling Lattice Alteration [2], [4]–[6]

Sampling lattice alteration system is an indispensable element for MD multirate systems [2], [4]–[6]. Unlike 1D sampling rate conversion [18], the lattice alteration is expressed by means of matrix instead of scalar. Figure 1 shows the basic structure of the sampling lattice alteration system with a rational matrix $\boldsymbol{L}^{-1}\boldsymbol{M}$, where $\boldsymbol{L}$ and $\boldsymbol{M}$ are $D \times D$ nonsingular integer matrices. When $\boldsymbol{L}$ is chosen as the identity matrix (denoted $\boldsymbol{I}$), the system is reduced to the decimator with the integer matrix $\boldsymbol{M}$. On the other hand, it comes to the interpolator with the integer matrix $\boldsymbol{L}$, when $\boldsymbol{M} = \boldsymbol{I}$.

In Fig. 1, $x(\boldsymbol{n})$, $y(\boldsymbol{m})$ and $h(\boldsymbol{k})$ denote an D-dimensional input signal sequence, the output signal sequence and the impulse response of an LSI system. These sequences are related as follows:

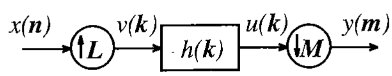


Fig. 1 A sampling lattice alteration system.

$$v(\boldsymbol{k}) = \begin{cases} x(\boldsymbol{L}^{-1}\boldsymbol{k}), & \boldsymbol{k} \in \mathcal{L}(\boldsymbol{L}) \\ 0, & \text{elsewhere} \end{cases}, \quad (7)$$

$$u(\boldsymbol{k}) = v(\boldsymbol{k}) * h(\boldsymbol{k}),\ \boldsymbol{k} \in \mathcal{N}, \quad (8)$$

$$y(\boldsymbol{m}) = u(\boldsymbol{M}\boldsymbol{m}),\ \boldsymbol{m} \in \mathcal{N}. \quad (9)$$

The operations in Eqs. (7) and (9) are referred to as up-sampling and down-sampling, respectively. Note that the filtering operation has a lot of redundancy, since some samples of the sequence $v(\boldsymbol{k})$ to be filtered are zero-values due to up-sampling, and some samples of filtered sequence $u(\boldsymbol{k})$ is discarded by down-sampling.

In the following, the processes defined as Eqs. (7) and (9) are represented as $v(\boldsymbol{k}) = [x]_{\uparrow \boldsymbol{L}}(\boldsymbol{k})$ and $y(\boldsymbol{m}) = [u]_{\downarrow \boldsymbol{M}}(\boldsymbol{m})$, respectively.

3. Parallel Filtering Approach

In MD real-time processing, it is hardly possible to avoid the use of the parallel processing approach [9]–[11]. For FIR filtering, OLA and OLS are used as the basic techniques of the parallel processing [10], [11]. In this section, as the first step for altering sampling lattices in parallel, we consider applying OLA and OLS to the filtering as shown in Fig. 2. Although the structure has redundant operations, it plays an important role for the later expansion.

Let $\boldsymbol{N}_S$ be a $D \times D$ nonsingular integer matrix, and $h(\boldsymbol{k})$ be an impulse response defined over a region of support $\mathcal{N}_h \subset \mathcal{N}$, that is, an FIR filter. The followings show the procedures of the sampling lattice alteration with OLA and OLS, respectively.

3.1 Application of the Overlap-Add Method (OLA)

By using OLA, in the system as shown in Fig. 1, the filtering of the up-sampled sequence $v(\boldsymbol{k})$ can be implemented with the following procedure: Firstly, segment $v(\boldsymbol{k})$ into blocks generated by the matrix $\boldsymbol{N}_S$ as

$$v_i(\boldsymbol{k}') = \begin{cases} v(\boldsymbol{N}_S \boldsymbol{i} + \boldsymbol{k}'), & \boldsymbol{k}' \in \mathcal{N}(\boldsymbol{N}_S) \\ 0, & \text{elsewhere} \end{cases}, \quad (10)$$

where $\boldsymbol{i} \in \mathcal{N}$ and $v_i(\boldsymbol{k}')$ is an $\boldsymbol{i}$-th segmented block data. Secondly, execute the linear convolution of each block data $v_i(\boldsymbol{k}')$ with the filter $h(\boldsymbol{k}')$ individually as

$$u_i(\boldsymbol{k}') = \begin{cases} v_i(\boldsymbol{k}') * h(\boldsymbol{k}'), & \boldsymbol{k}' \in \mathcal{N}_h(\boldsymbol{N}_S) \\ 0, & \text{elsewhere} \end{cases}, \quad (11)$$

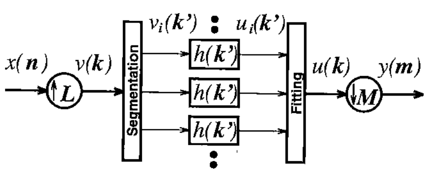


Fig. 2 A sampling lattice alteration system with the parallel filtering approach.

where $\mathcal{N}_h(\boldsymbol{N}_S)$ is derived from Eq. (2) as

$$\mathcal{N}_h(\boldsymbol{N}_S) = \bigcup_{\boldsymbol{k}' \in \mathcal{N}(\boldsymbol{N}_S)} \mathcal{N}_{h\boldsymbol{k}'}. \quad (12)$$

And finally, add the results $u_i(\boldsymbol{k}'), \boldsymbol{i} \in \mathcal{N}$ together to obtain the overall filtered sequence $u(\boldsymbol{k})$ as

$$u(\boldsymbol{k}) = \sum_{\boldsymbol{i} \in \mathcal{N}} u_i(\boldsymbol{k} - \boldsymbol{N}_S \boldsymbol{i}),\ \boldsymbol{k} \in \mathcal{N}. \quad (13)$$

In this work, we refer to $\boldsymbol{N}_S$ as a segmentation matrix.

Figure 3 shows an example of the above procedure, where we choose $D = 2$, $\mathcal{N}_h = \mathcal{N}(\boldsymbol{N}_S)$ and

$$\boldsymbol{N}_S = \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}. \quad (14)$$

In Fig. 3, k_ℓ denotes the component of the variable $\boldsymbol{k}$, that is, $\boldsymbol{k} = (k_0, k_1)^T$, and the shaded area indicates the region assigned for an $\boldsymbol{i}$-th block, where $(1,1)^T$-st block is remarked. Note that the illustrations are two-dimensional, although the theory is applicable to any number of dimensions.

3.2 Application of the Overlap-Save Method (OLS)

Another approach for filtering in parallel is OLS, of which procedure is similar to OLA. The procedure is shown as follows: Firstly, segment $v(\boldsymbol{r})$ into overlapping blocks as

$$v_i(\boldsymbol{k}') = \begin{cases} v(\boldsymbol{N}_S \boldsymbol{i} + \boldsymbol{k}'), & \boldsymbol{k}' \in \bar{\mathcal{N}}_h(\boldsymbol{N}_S) \\ 0, & \text{elsewhere} \end{cases}, \quad (15)$$

where $\boldsymbol{i} \in \mathcal{N}$ and $\bar{\mathcal{N}}_h(\boldsymbol{N}_S)$ is derived from Eq. (3) as

$$\bar{\mathcal{N}}_h(\boldsymbol{N}_S) = \bigcup_{\boldsymbol{k}' \in \mathcal{N}(\boldsymbol{N}_S)} \bar{\mathcal{N}}_{h\boldsymbol{k}'}. \quad (16)$$

Secondly execute the linear convolution of each $v_i(\boldsymbol{k}')$ with $h(\boldsymbol{k}')$ individually to obtain the result $u_i(\boldsymbol{k}')$ only within the region of support $\mathcal{N}(\boldsymbol{N}_S)$ as

$$u_i(\boldsymbol{k}') = \begin{cases} v_i(\boldsymbol{k}') * h(\boldsymbol{k}'), & \boldsymbol{k}' \in \mathcal{N}(\boldsymbol{N}_S) \\ 0, & \text{elsewhere} \end{cases}. \quad (17)$$

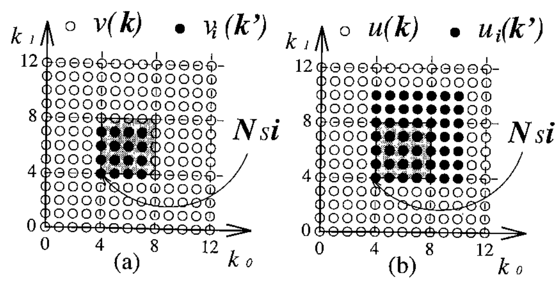


Fig. 3 The overlap-add method, (a) Segmentation, (b) Fitting.

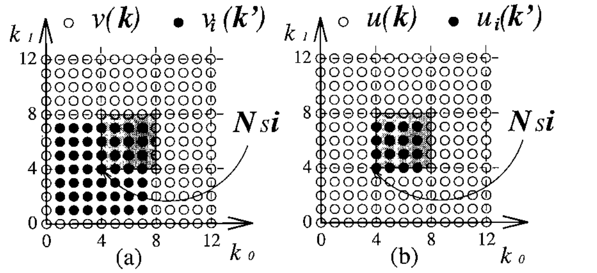


Fig. 4 The overlap-save method, (a) Segmentation, (b) Fitting.

And finally, fit the results $u_i(\boldsymbol{k}'), \boldsymbol{i} \in \mathcal{N}$ to obtain the overall filtered sequence $u(\boldsymbol{k})$ as Eq. (13). Actually, this fitting operation does not require any additions, because there is no overlapping block.

As well as Fig. 3, we show the procedure of OLS in Fig. 4, as an example.

In this section, we showed that the use of OLA or OLS in sampling lattice alteration enables us to implement the filtering in parallel. In the procedure, however, it is impossible to avoid the redundancy caused by up- and down-sampling. The redundancy is not desirable in terms of the computational efficiency. The aim of this work is to eliminate the redundancy even under the parallel processing.

4. Parallel Sampling Lattice Alteration

As we mentioned before, there is redundancy in the structure as shown in Fig. 2. In this section, taking account of the up- and down-sampling, we propose to extend the parallel processing in Fig. 2 to the whole system as shown in Fig. 5, so that both the elimination of the redundancy and the parallel processing can be simultaneously established.

The theory of this work is regarded as the generalized version of the 1D EOLA and EOLS [16], [17].

4.1 The Extended Overlap-Add Method (EOLA)

For the sampling lattice alteration with OLA as shown in Fig. 2, the extension of the parallel processing can be done by choosing the segmentation matrix $\boldsymbol{N}_S$ as a common right multiple (crm) of $\boldsymbol{L}$ and $\boldsymbol{M}$ [2], [5], [6], that is,

$$\boldsymbol{N}_S \in \operatorname{crm}(\boldsymbol{L}, \boldsymbol{M}), \tag{18}$$

from the following lemmas.

Lemma 1: If and only if $\boldsymbol{N}_S$ is a right multiple (rm) of $\boldsymbol{L}$ [2], [5], [6], the sequence $v_i(\boldsymbol{k}')$ provided in Eq. (10) can be obtained as

$$v_i(\boldsymbol{k}') = [x_i]_{\uparrow \boldsymbol{L}}(\boldsymbol{k}'), \tag{19}$$

after the input sequence $x(\boldsymbol{n})$ is segmented into blocks as

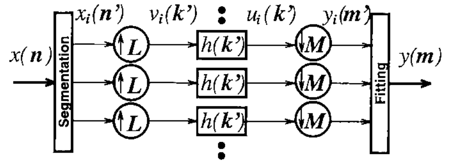


Fig. 5 A sophisticated parallel sampling lattice alteration system.

$$x_i(\boldsymbol{n}') = \begin{cases} x(\boldsymbol{Q}_S\boldsymbol{i} + \boldsymbol{n}'), & \boldsymbol{n}' \in \mathcal{N}(\boldsymbol{Q}_S) \\ 0, & \text{elsewhere} \end{cases}, \tag{20}$$

where $\boldsymbol{Q}_S = \boldsymbol{L}^{-1}\boldsymbol{N}_S$.

Proof: From Eq. (7), Eq. (10) is represented in terms of $x(\boldsymbol{n})$ as

$$v_i(\boldsymbol{k}') = \begin{cases} x(\boldsymbol{L}^{-1}(\boldsymbol{N}_S\boldsymbol{i} + \boldsymbol{k}')), \\ \qquad \boldsymbol{k}' \in \mathcal{L}_{\boldsymbol{N}_S\boldsymbol{i}}(\boldsymbol{L}) \cap \mathcal{N}(\boldsymbol{N}_S) \\ 0, \qquad \text{elsewhere} \end{cases}, \tag{21}$$

where $\mathcal{L}_{\boldsymbol{n}}(\boldsymbol{L}) \triangleq \{\boldsymbol{k}' | \boldsymbol{n} + \boldsymbol{k}' \in \mathcal{L}(\boldsymbol{L})\}$. Now, we can verify that $\mathcal{L}_{\boldsymbol{N}_S\boldsymbol{i}}(\boldsymbol{L})$ always is equal to $\mathcal{L}(\boldsymbol{L})$ independently of the block index $\boldsymbol{i}$, there exists some integer matrix $\boldsymbol{Q}_S = \boldsymbol{L}^{-1}\boldsymbol{N}_S$ and Eq. (21) is rewritten as

$$v_i(\boldsymbol{k}') = \begin{cases} x(\boldsymbol{Q}_S\boldsymbol{i} + \boldsymbol{L}^{-1}\boldsymbol{k}'), \\ \qquad \boldsymbol{k}' \in \mathcal{L}(\boldsymbol{L}) \cap \mathcal{N}(\boldsymbol{L}\boldsymbol{Q}_S) \\ 0, \qquad \text{elsewhere} \end{cases}, \tag{22}$$

if and only if $\boldsymbol{N}_S$ is an rm of $\boldsymbol{L}$. Since $\mathcal{L}(\boldsymbol{L}) \cap \mathcal{N}(\boldsymbol{L}\boldsymbol{Q}_S)$ is equal to $\boldsymbol{L}\mathcal{N}(\boldsymbol{Q}_S)$, we can notice that $\boldsymbol{L}^{-1}\boldsymbol{k}' \in \mathcal{N}(\boldsymbol{Q}_S)$. In the result, $v_i(\boldsymbol{k}')$ can be obtained from $x_i(\boldsymbol{n}')$ provided in Eq. (20) as Eq. (19). □

Equation (19) implies that the up-sampling can be implemented in parallel individually.

Lemma 2: If and only if $\boldsymbol{N}_S$ is an rm of $\boldsymbol{M}$, then the overall final output sequence $y(\boldsymbol{m})$ can be obtained as

$$y(\boldsymbol{m}) = \sum_{\boldsymbol{i} \in \mathcal{N}} y_i(\boldsymbol{m} - \boldsymbol{P}_S\boldsymbol{i}), \ \boldsymbol{m} \in \mathcal{N}, \tag{23}$$

from the sequence $y_i(\boldsymbol{m}')$ provided as follows:

$$y_i(\boldsymbol{m}') = \begin{cases} [u_i]_{\downarrow \boldsymbol{M}}(\boldsymbol{m}'), & \boldsymbol{m}' \in [\mathcal{N}_h]_{\downarrow \boldsymbol{M}}(\boldsymbol{P}_S) \\ 0, & \text{elsewhere} \end{cases}, \tag{24}$$

where $\boldsymbol{P}_S = \boldsymbol{L}^{-1}\boldsymbol{N}_S$ and $[\mathcal{N}_h]_{\downarrow \boldsymbol{M}}(\boldsymbol{P}_S) \triangleq \{\boldsymbol{m}' \in \mathcal{N} | \boldsymbol{M}\boldsymbol{m}' \in \mathcal{N}_h(\boldsymbol{M}\boldsymbol{P}_S)\}$.

Proof: From Eqs. (9) and (13), the overall final output sequence $y(\boldsymbol{m})$ is represented in terms of $u_i(\boldsymbol{k}')$ as

$$y(\boldsymbol{m}) = \sum_{\boldsymbol{i} \in \mathcal{N}} u_i(\boldsymbol{M}\boldsymbol{m} - \boldsymbol{N}_S\boldsymbol{i}), \ \boldsymbol{m} \in \mathcal{N}. \tag{25}$$

Now, we can verify that there exists some integer matrix $\boldsymbol{P}_S = \boldsymbol{M}^{-1}\boldsymbol{N}_S$ and Eq. (25) is rewritten as

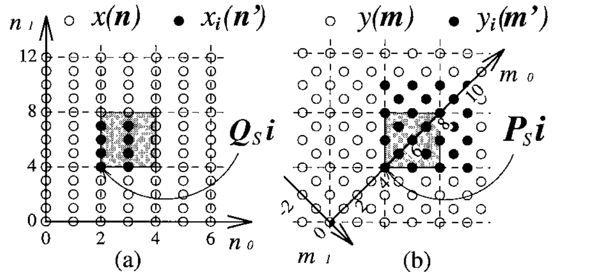


Fig. 6 The extended overlap-add method, (a) Segmentation, (b) Fitting.

$$y(\boldsymbol{m}) = \sum_{\boldsymbol{i}\in\mathcal{N}} u_i(\boldsymbol{M}(\boldsymbol{m} - \boldsymbol{P}_S\boldsymbol{i})),\ \boldsymbol{m} \in \mathcal{N}, \tag{26}$$

if and only if $\boldsymbol{N}_S$ is an rm $\boldsymbol{M}$. In the result, $y(\boldsymbol{m})$ can be obtained from $y_i(\boldsymbol{m}')$ provided in Eq. (24) as Eq. (23), where we note that the decimated sequence $[u_i]_{\downarrow\boldsymbol{M}}(\boldsymbol{m}')$ has the region of support $[\mathcal{N}_h]_{\downarrow\boldsymbol{M}}(\boldsymbol{P}_S)$ decimated from $\mathcal{N}_h(\boldsymbol{M}\boldsymbol{P}_S)$ which $u_i(\boldsymbol{k}')$ provided in Eq. (11) has. □

Equation (24) implies that the down-sampling can be implemented in parallel individually.

From **Lemma 1** and **2**, by choosing $\boldsymbol{N}_S$ to satisfy Eq. (18), we can implement the whole operation of sampling lattice alteration in parallel as shown in Fig. 5. It is known that there exists at least one crm$(\boldsymbol{L},\boldsymbol{M})$, which is $\boldsymbol{R} = \boldsymbol{L}(J(\boldsymbol{M})\boldsymbol{I})$ [2]. Therefore, one of the simple ways to choose the segmentation matrix $\boldsymbol{N}_S$ satisfying the condition in Eq. (18) is as follows: Let $\boldsymbol{N}_S = \boldsymbol{R}\boldsymbol{S}$, where $\boldsymbol{S}$ is an arbitrary nonsingular integer matrix.

In Fig. 6, for example, we show the illustrations of the segmentation and fitting operations, respectively, where we choose

$$\boldsymbol{L} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix},\ \boldsymbol{M} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{27}$$

in addition to $\boldsymbol{N}_S$ in Eq. (14). This example is an orthogonal-to-quincunx alteration. For the above choice, the condition in Eq. (18) holds and the matrices $\boldsymbol{Q}_S$ and $\boldsymbol{P}_S$ result in

$$\boldsymbol{Q}_S = \begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix},\ \boldsymbol{P}_S = \begin{pmatrix} 2 & 2 \\ 2 & -2 \end{pmatrix}. \tag{28}$$

In Fig. 6, n_ℓ and m_ℓ are the components of the variables $\boldsymbol{n}$ and $\boldsymbol{m}$, that is, $\boldsymbol{n} = (n_0, n_1)^T$ and $\boldsymbol{m} = (m_0, m_1)^T$, respectively, and the shaded area indicates the region assigned for an $\boldsymbol{i}$-th block, where $(1,1)^T$-st block is remarked.

4.2 The Extended Overlap-Save Method (EOLS)

As well as the OLA case, when we choose $\boldsymbol{N}_S$ to satisfy Eq. (18), it is possible to extend OLS as shown in Fig. 5. Figure 7 shows the segmentation and fitting operations of the extended overlap-save technique, where the matrices $\boldsymbol{N}_S$,$\boldsymbol{L}$ and $\boldsymbol{M}$ are chosen as Eqs. (14) and (27), respectively.

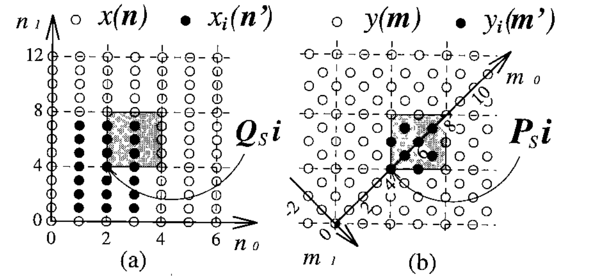


Fig. 7 The extended overlap-save method, (a) Segmentation, (b) Fitting.

In EOLS, the segmentation of the input sequence $x(\boldsymbol{n})$ is implemented as

$$x_i(\boldsymbol{n}') = \begin{cases} x(\boldsymbol{Q}_S\boldsymbol{i} + \boldsymbol{n}'), & \boldsymbol{n}' \in [\bar{\mathcal{N}}_h]_{\downarrow\boldsymbol{L}}(\boldsymbol{Q}_S) \\ 0, & \text{elsewhere} \end{cases}, \tag{29}$$

where $\boldsymbol{Q}_S = \boldsymbol{L}^{-1}\boldsymbol{N}_S$ and $[\bar{\mathcal{N}}_h]_{\downarrow\boldsymbol{L}}(\boldsymbol{Q}_S) \triangleq \{\boldsymbol{n}' \in \mathcal{N} | \boldsymbol{L}\boldsymbol{n}' \in \bar{\mathcal{N}}_h(\boldsymbol{L}\boldsymbol{Q}_S)\}$. The above segmentation is derived from Eqs. (7) and (15) with the similar discussion to **Lemma 1**, where we note that $\mathcal{L}(\boldsymbol{L}) \cap \bar{\mathcal{N}}_h(\boldsymbol{L}\boldsymbol{Q}_S)$ is equal to $\boldsymbol{L}\,[\bar{\mathcal{N}}_h]_{\downarrow\boldsymbol{L}}(\boldsymbol{Q}_S)$.

Also, the overall final output sequence $y(\boldsymbol{m})$ is obtained as Eq. (23) from the results $y_i(\boldsymbol{m}'), \boldsymbol{i} \in \mathcal{N}$ of

$$y_i(\boldsymbol{m}') = \begin{cases} [u_i]_{\downarrow\boldsymbol{M}}(\boldsymbol{m}'), & \boldsymbol{m}' \in \mathcal{N}(\boldsymbol{P}_S) \\ 0, & \text{elsewhere} \end{cases}, \tag{30}$$

where $\boldsymbol{P}_S = \boldsymbol{M}^{-1}\boldsymbol{N}_S$. Actually, this fitting operation does not require any additions. The above operation is derived from the similar discussion to **Lemma 2**, where we note that the decimated sequence $[u_i]_{\downarrow\boldsymbol{M}}(\boldsymbol{m}')$ has the region of support $\mathcal{N}(\boldsymbol{P}_S)$ decimated from $\mathcal{N}(\boldsymbol{M}\boldsymbol{P}_S)$ which $u_i(\boldsymbol{k}')$ provided in Eq. (17) has.

5. Efficient Implementation

From Fig. 5, we can see that the processing for each block data in EOLA and EOLS is identical to that of the basic structure as shown in Fig. 1. This fact implies that a number of conventional techniques for sampling lattice alteration, such as polyphase and multistage techniques [2], [4], can be applied to our proposed methods directly. Therefore, the problem of the redundancy can easily be solved even under the parallel processing.

In this section, we provide the DFT-domain approach of EOLA and EOLS without the redundancy caused by up- and down-sampling, and give some comments on the computational complexity. According to the previous discussion, we assume that $\boldsymbol{N}_S$ is a segmentation matrix at the filtering operation and satisfies Eq. (18), and that $h(\boldsymbol{k})$ is an FIR filter defined over a region of support $\mathcal{N}_h \subset \mathcal{N}$.

5.1 The DFT-Domain Approach

It is known that the the conventional OLA and OLS can be implemented with the DFT-domain approach, which is computationally efficient when a high-order FIR filter is employed [12]–[14]. This approach uses the fact that linear convolution can be implemented by the product of the corresponding DFTs. In the following, taking account of up- and down-sampling, we provide the procedure modified for EOLA and EOLS.

Suppose that $\boldsymbol{N}$ is a nonsingular integer matrix satisfying the following conditions: in EOLA case,

$$\mathcal{N}_{\boldsymbol{s}}(\boldsymbol{N}) \supseteqq \mathcal{N}_h(\boldsymbol{N}_S), \text{ for some } \boldsymbol{s} \in \mathcal{N}, \tag{31}$$

in EOLS case,

$$\mathcal{N}_{\boldsymbol{s}}(\boldsymbol{N}) \supseteqq \bar{\mathcal{N}}_h(\boldsymbol{N}_S) \text{ and } \mathcal{N}_{\boldsymbol{s}'}(\boldsymbol{N}) \supseteqq \mathcal{N}_h, \quad \text{for some } \boldsymbol{s} \in \mathcal{N} \text{ and some } \boldsymbol{s}' \in \mathcal{N}, \tag{32}$$

and besides, in either case,

$$\boldsymbol{N} = \boldsymbol{N}_S \boldsymbol{\Lambda}, \tag{33}$$

for some diagonal matrix $\boldsymbol{\Lambda}$ with nonzero rational elements larger than one on the diagonal, where $\mathcal{N}_{\boldsymbol{s}}(\boldsymbol{N}) \triangleq \{\boldsymbol{k} | \boldsymbol{k} - \boldsymbol{s} \in \mathcal{N}(\boldsymbol{N})\}$, $\mathcal{N}_h(\boldsymbol{N}_S)$ and $\bar{\mathcal{N}}_h(\boldsymbol{N}_S)$ are provided in Eqs. (12) and (16), respectively. Using the above $\boldsymbol{N}$ as a periodicity matrix, we can implement both EOLA and EOLS in the DFT domain.

Equations (31) and (32) correspond to sufficient conditions that the conventional OLA and OLS can be implemented with the DFT-domain approach, respectively (see Appendix A). Equation (33) assures that $\boldsymbol{N}$ is a crm of $\boldsymbol{L}$ and $\boldsymbol{M}$, which is the necessary and sufficient condition that the up- and down-sampling can be implemented in the DFT domain without the redundancy (see Appendix B), and that $\mathcal{N}(\boldsymbol{N}_S) \subset \mathcal{N}(\boldsymbol{N})$. One of the simple ways to choose the periodicity matrix $\boldsymbol{N}$ satisfying the above conditions is as follows: Let $\boldsymbol{\Lambda} = \lambda \boldsymbol{I}$, where λ is a rational scalar factor which holds $\boldsymbol{N}$ an integer matrix and is sufficiently large to satisfy the condition in Eq. (31) for EOLA or Eq. (32) for EOLS.

Let $\boldsymbol{Q}$ and $\boldsymbol{P}$ be $\boldsymbol{L}^{-1}\boldsymbol{N}$ and $\boldsymbol{M}^{-1}\boldsymbol{N}$, respectively, which are guaranteed to be nonsingular integer matrices from Eq. (33). In the following, we show the procedures of the DFT-domain approach for EOLA and EOLS.

A. The DFT-domain approach for EOLA

Step 1: Obtain the finite-extent sequence $\hat{x}_i(\boldsymbol{n}')$ from $x_i(\boldsymbol{n}')$ provided in Eq. (20) as

$$\hat{x}_i(\boldsymbol{n}') = x_i(\boldsymbol{n}'), \ \boldsymbol{n}' \in \mathcal{N}(\boldsymbol{Q}), \tag{34}$$

where we note that $\mathcal{N}(\boldsymbol{Q}_S) \subset \mathcal{N}(\boldsymbol{Q})$ since $\mathcal{N}(\boldsymbol{N}_S) \subset \mathcal{N}(\boldsymbol{N})$.
Step 2: Compute the matrix-$\boldsymbol{Q}$ DFT $X_i(\boldsymbol{q})$ of $\hat{x}_i(\boldsymbol{n}')$.
Step 3: Periodically extend $X_i(\boldsymbol{q})$ as

$$V_i(\boldsymbol{\ell}) = X_i(((\boldsymbol{\ell}))_{\boldsymbol{Q}^T}), \ \boldsymbol{\ell} \in \mathcal{N}(\boldsymbol{N}^T). \tag{35}$$

This operation corresponds to up-sampling.
Step 4: Execute the product of $V_i(\boldsymbol{\ell})$ and the matrix-$\boldsymbol{N}$ DFT $H(\boldsymbol{\ell})$ of $\hat{h}(\boldsymbol{k}')$ as

$$U_i(\boldsymbol{\ell}) = V_i(\boldsymbol{\ell}) H(\boldsymbol{\ell}), \ \boldsymbol{\ell} \in \mathcal{N}(\boldsymbol{N}^T), \tag{36}$$

where

$$\hat{h}(\boldsymbol{k}') = \sum_{\boldsymbol{r} \in \mathcal{N}} h(\boldsymbol{k}' - \boldsymbol{N}\boldsymbol{r}), \ \boldsymbol{k}' \in \mathcal{N}(\boldsymbol{N}). \tag{37}$$

This operation corresponds to circular convolution [13].
Step 5: Subdivide $U_i(\boldsymbol{\ell})$ and add the sub-sequences together as

$$Y_i(\boldsymbol{p}) = \frac{1}{J(\boldsymbol{M})} \sum_{\boldsymbol{r} \in \mathcal{N}(\boldsymbol{M}^T)} U_i(\boldsymbol{P}^T \boldsymbol{r} + \boldsymbol{p}), \quad \boldsymbol{p} \in \mathcal{N}(\boldsymbol{P}^T). \tag{38}$$

This operation corresponds to down-sampling.
Step 6: Compute the matrix-$\boldsymbol{P}$ IDFT $\hat{y}_i(\boldsymbol{m}')$ of $Y_i(\boldsymbol{p})$.
Step 7: Obtain the output block data $y_i(\boldsymbol{m}')$ as

$$y_i(\boldsymbol{m}') = \begin{cases} \hat{y}_i(((\boldsymbol{m}'))_{\boldsymbol{P}}), & \boldsymbol{m}' \in [\mathcal{N}_h]_{\downarrow \boldsymbol{M}}(\boldsymbol{P}_S) \\ 0, & \text{elsewhere} \end{cases}, \tag{39}$$

which is identical to the sequence provided in Eq. (24).

B. The DFT-domain approach for EOLS

Step 1: Obtain the finite-extent sequence $\hat{x}_i(\boldsymbol{n}')$ from $x_i(\boldsymbol{n}')$ provided in Eq. (29) as

$$\hat{x}_i(\boldsymbol{n}') = \sum_{\boldsymbol{r} \in \mathcal{N}} x_i(\boldsymbol{n}' - \boldsymbol{Q}\boldsymbol{r}), \ \boldsymbol{n}' \in \mathcal{N}(\boldsymbol{Q}). \tag{40}$$

Under the condition in Eq. (32), this operation does not require any additions, but circularly shifting $x_i(\boldsymbol{n}')$.
Steps 2–6: Execute the same as the EOLA case.
Step 7: Obtain the output block data $y_i(\boldsymbol{m}')$ as

$$y_i(\boldsymbol{m}') = \begin{cases} \hat{y}_i(\boldsymbol{m}'), & \boldsymbol{m}' \in \mathcal{N}(\boldsymbol{P}_S) \\ 0, & \text{elsewhere} \end{cases}, \tag{41}$$

which is identical to the sequence provided in Eq. (30), where we note that $\mathcal{N}(\boldsymbol{P}_S) \subset \mathcal{N}(\boldsymbol{P})$ since $\mathcal{N}(\boldsymbol{N}_S) \subset \mathcal{N}(\boldsymbol{N})$.

5.2 Computational Complexities

In order to show the effectiveness of the parallel sampling lattice alteration described in Sect. 4, we compare it with the parallel filtering approach described in Sect. 3 in terms of the computational complexity of the DFT-domain approach. Note that, in the parallel filtering approach as shown in Fig. 2, it is impossible to exploit the modified procedures described in Sect. 5.1. Hence, we assume that the conventional DFT-domain approach [12]–[14] is applied to it.

Let $\mu(\boldsymbol{N})$ and $\alpha(\boldsymbol{N})$ be the total number of real multiplications and additions of the matrix-$\boldsymbol{N}$ DFT (or IDFT), respectively, and assume that the input sequence $x(\boldsymbol{n})$ and the impulse response $h(\boldsymbol{k})$ be real values, $H(\boldsymbol{\ell})$ has been precomputed and stored, and complex multiplication is done by the 3/3 algorithm. Note that, when both $x(\boldsymbol{n})$ and $h(\boldsymbol{k})$ are real, the operations in the DFT domain can be reduced to almost half due to the symmetric properties. The computational load of the parallel filtering approach via the DFT is approximately represented as follows:

$$M_{\rm C} \cong \frac{\mu_{\rm F}(\boldsymbol{N})+\frac{3}{2}J(\boldsymbol{N})+\mu_{\rm I}(\boldsymbol{N})}{J(\boldsymbol{N}_S)}\cdot J(\boldsymbol{M}), \quad (42)$$

$$A_{\rm C} \cong \frac{\alpha_{\rm F}(\boldsymbol{N})+\frac{3}{2}J(\boldsymbol{N})+\alpha_{\rm I}(\boldsymbol{N})+\beta}{J(\boldsymbol{N}_S)}\cdot J(\boldsymbol{M}), \quad (43)$$

where $M_{\rm C}$ and $A_{\rm C}$ are the real multiplications per output sample and the real additions per output sample, respectively, and β is the number of elements in the region of support $\{\mathcal{N}_h(\boldsymbol{N}_S)-\mathcal{N}(\boldsymbol{N}_S)\}$ in the OLA case, whereas $\beta=0$ in the OLS case. Besides, the subscripts 'F' and 'I' indicate the DFT and IDFT, respectively. The fractional part indicates the number of operations per output sample of the parallel filtering and $J(\boldsymbol{M})$ is the factor caused by down-sampling. On the other hand, the complexity of the modified DFT-domain approach is approximately represented as follows:

$$M_{\rm M} \cong \frac{\mu_{\rm F}(\boldsymbol{Q})+\frac{3}{2}J(\boldsymbol{N})+\mu_{\rm I}(\boldsymbol{P})}{J(\boldsymbol{P}_S)}, \quad (44)$$

$$A_{\rm M} \cong \frac{\alpha_{\rm F}(\boldsymbol{Q})+\frac{5}{2}J(\boldsymbol{N})-J(\boldsymbol{P})+\alpha_{\rm I}(\boldsymbol{P})+\gamma}{J(\boldsymbol{P}_S)}, \quad (45)$$

where $M_{\rm M}$ and $A_{\rm M}$ are the real multiplications per output sample and the real additions per output sample, respectively, and γ is the number of elements in the region of support $\{[\mathcal{N}_h]_{\downarrow\boldsymbol{M}}(\boldsymbol{P}_S)-\mathcal{N}(\boldsymbol{P}_S)\}$ in the EOLA case, whereas $\gamma=0$ in the EOLS case.

As an example, let us suppose that the up-sampling matrix $\boldsymbol{L}$ and the down-sampling matrix $\boldsymbol{M}$ are provided as in Eq. (27), the filter $h(\boldsymbol{k})$ has the region of support

$$\mathcal{N}_h=\mathcal{N}\begin{pmatrix}16 & 0\\ 0 & 16\end{pmatrix}, \quad (46)$$

the segmentation matrix $\boldsymbol{N}_S$ at the filtering operation is chosen to satisfy the condition in Eq. (18) as

$$\boldsymbol{N}_S=\begin{pmatrix}16 & 0\\ 0 & 16\end{pmatrix}, \quad (47)$$

and the periodicity matrix $\boldsymbol{N}$ at the filtering operation is chosen to satisfy the conditions in Eqs. (31), (32) and (33) as

$$\boldsymbol{N}=\begin{pmatrix}32 & 0\\ 0 & 32\end{pmatrix}. \quad (48)$$

In the following, on the above suppositions, we compare the modified DFT-domain approach with the parallel filtering appraoch via the DFT.

Firstly, we investigate the complexity of the parallel filtering approach via the DFT. Let us consider implementing the matrix-$\boldsymbol{N}$ DFT via the radix-(2×2) FFT algorithm [13]. Using the algorithm, both $\mu_{\rm F}(\boldsymbol{N})$ and $\mu_{\rm I}(\boldsymbol{N})$ in Eq. (42) result in $\mu_{\rm F}(\boldsymbol{N})=\mu_{\rm I}(\boldsymbol{N})=11520$, and both $\alpha_{\rm F}(\boldsymbol{N})$ and $\alpha_{\rm I}(\boldsymbol{N})$ in Eq. (43) result in $\alpha_{\rm F}(\boldsymbol{N})=\alpha_{\rm I}(\boldsymbol{N})=32000$. Hence, from Eqs. (42) and (43), we can verify that the parallel filtering with the conventional DFT-domain approach requires $M_{\rm C}\cong 192$ and $A_{\rm C}\cong 512$ in the OLS case or $A_{\rm C}\cong 517.5$ in the OLA case, where $\beta=705$.

And secondly, we investigate the complexity of the modified DFT-domain approach. Let us consider implementing the matrix-$\boldsymbol{Q}$ DFT and the matrix-$\boldsymbol{P}$ IDFT via the vector-radix FFT algorithm [13], by decomposing $\boldsymbol{Q}$ and $\boldsymbol{P}$ into

$$\boldsymbol{Q}=\begin{pmatrix}16 & 0\\ 0 & 32\end{pmatrix}=\begin{pmatrix}1 & 0\\ 0 & 2\end{pmatrix}\begin{pmatrix}16 & 0\\ 0 & 16\end{pmatrix}, \quad (49)$$

$$\boldsymbol{P}=\begin{pmatrix}16 & 16\\ 16 & -16\end{pmatrix}=\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}16 & 0\\ 0 & 16\end{pmatrix}, \quad (50)$$

respectively, and exploiting the radix-(2×2) FFT algorithm. Using the algorithm, $\mu_{\rm F}(\boldsymbol{Q})$ and $\mu_{\rm I}(\boldsymbol{P})$ in Eq. (44) result in $\mu_{\rm F}(\boldsymbol{Q})\leqq 6144$ and $\mu_{\rm I}(\boldsymbol{P})\leqq 6144$, respectively, and $\alpha_{\rm F}(\boldsymbol{Q})$ and $\alpha_{\rm I}(\boldsymbol{P})$ in Eq. (45) result in $\alpha_{\rm F}(\boldsymbol{Q})\leqq 15360$ and $\alpha_{\rm I}(\boldsymbol{P})\leqq 15360$, respectively. Hence, from Eqs. (44) and (45), we can verify that the modified DFT-domain approach is realized at most by $M_{\rm M}\cong 108$ and $A_{\rm M}\cong 256$ in the EOLS case or $A_{\rm M}\cong 258.8$ in the EOLA case, where $\gamma=353$. Clearly, $M_{\rm M}$ is less than $M_{\rm C}$, and $A_{\rm M}$ is less than $A_{\rm C}$ in this example. The reason is that the modified approach has no redundant operations caused by up- and down-sampling.

In the following, we give some comments on the choice of the segmentation matrix $\boldsymbol{N}_S$ and the periodicity matrix $\boldsymbol{N}$. From Eqs. (42), (43), (44) and (45), the computational complexities of both approaches depend on $\boldsymbol{N}$, $\boldsymbol{N}_S$ and the employed FFT algorithms. For the conventional approach, the block size consideration in terms of the computational efficiency can be found in the article [14]. The similar discussion can be expected to appropriately choose $\boldsymbol{N}_S$ and $\boldsymbol{N}$ in the modified approach, although it also depends on $\boldsymbol{L}$ and $\boldsymbol{M}$. Note that, however, the choice of $\boldsymbol{N}_S$ and $\boldsymbol{N}$ affects not only the computational efficiency but also the memory requirement. Although there has to be a trade-off between the efficiency and the memory, it is relatively difficult to find an optimal segmentation and periodicity matrices, since it is machine-dependent [12].

6. Conclusions

In this paper, we proposed two parallel processing methods for multidimensional sampling lattice alteration by extending two conventional block processing techniques for FIR filtering: the overlap-add method and the overlap-save method, and also show that the use of our proposed methods enables us to alter sampling lattices without any redundancy caused by up- and down-sampling as well as the parallel processing, even if the alteration is rational and non-separable. The theory of this work applies to any number of dimension. Therefore, it is possible to alter sampling lattice from orthogonal to quincunx or from orthogonal to FCO in parallel without the redundancy.

In future, we will investigate the practical applications of our proposed methods, such as motion picture coding, image format conversion and so forth.

References

[1] Vixcito, E. and Allebach, J.P., "The Analysis and Design of Multidimensional FIR Perfect Reconstruction Filter Banks for Arbitrary Sampling Lattices," *IEEE Trans. on Circuits and Systems*, vol.38, no.1, pp.29–41, Jan. 1991.

[2] Chen, T. and Vaidyanathan, P.P., "The Role of Matrices in Multidimensional Multirate Systems," *IEEE Trans. on Signal Processing*, vol.41, no.3, pp.1035–1047, Mar. 1993.

[3] Chen, T. and Vaidyanathan, P.P.,"Multidimensinal Multirate Filters and Filter Banks Derived from One-Dimensional Filters," *IEEE Trans. on Signal Processing*, vol.41, no.5, pp.1749–1765, May 1993.

[4] Vaidyanathan, P.P., *Multirate Systems and Filter Banks*, Prentice Hall, Englewood Cliffs, 1993.

[5] Evans, B.L., Bamberger, R.H. and McClellan, J.H., "Rules for Multidimensional Multirate Structures," *IEEE Trans. on Signal Processing*, vol.42, no.4, pp.762–771, Apr. 1994.

[6] Gopinath, R.A. and Burrus, C.S., "On Upsampling, Downsampling, and Rational Sampling Rate Filter Banks," *IEEE Trans. on Signal Processing*, vol.42, no.4, pp.812–824, Apr. 1994.

[7] Kovačević, J. and Vetterli, M., "FCO Sampling of Digital Video Using Perfect Reconstruction Filter Banks," *IEEE Trans. on Image Processing*, vol.2, no.1, pp.118–122, Jan. 1993.

[8] Sezan, M.I. and Lagendijk, R.L., *Motion Analysis and Image Sequence Processing*, Kluwer Academic Publishers, 1993.

[9] Sawabe, T., Fujii, T., Fujii, T. and Ono, S., "Performance Evaluation of Super High Definition Image Processing on a Parallel DSP System," *IEICE Trans. on Fundamentals*, vol.E76–A, no.8, pp.1308–1315, Aug. 1993.

[10] Pirsch, P., *VLSI Implementation for Image Communication*, Elsevier Science Publishers, 1993.

[11] Tamitani, I., Harasaki, H., Nishitani, T., Endo, Y., Yamashina, M. and Enomoto, T., "A Real-Time Video Signal Processor Suitable for Motion Picture Coding Applications," *IEEE Trans. on Circuits and Systems*, vol.36, no.10, pp.1259–1266, Oct. 1989.

[12] Twogood, R.E., Ekstrom, M.P. and Mitra, S.K., "Optimal Section Procedure for the Implemetatioin of 2-D Digital Filters," *IEEE Trans. on Circuits and Systems*, vol.CAS25, no.5, pp.260–269, May 1978.

[13] Dudgeon, D.E. and Mersereau, R.M., *Multidimensional Digital Signal Processing*, Prentice Hall, Englewood Cliffs, 1984.

[14] Li, X. and Qian, G., "Block Size Consideration for Multidimensional Convolution and Correation," *IEEE Trans. on Signal Processing*, vol.40, no.5, pp.1271–1273, May 1992.

[15] Muramatsu, S. and Kiya, H., "A Multidimensional Block Operation for Arbitrary Sampling Lattice Alteration," *Proc. of 9th DSP Symposium*, pp.593–598, Jun. 1994.

[16] Muramatsu, S. and Kiya, H., "An Extended Overlap-Add Method and -Save Method for Sampling Rate Conversion," *ISCAS'94*, vol.2, no.2, pp.313–316, Jun. 1994.

[17] Muramatsu, S. and Kiya, H., "An Extended Overlap-Add Method and Save Method for Sampling Rate Conversion," *IEICE Trans. on Fundamentals*, vol.J77–A, no.8, Aug. 1994.

[18] Crochiere, R.E. and Rabiner, L.R., *Multirate Digital Signal Processing*, Prentice-Hall, Englewood Cliffts NJ, 1983.

Appendix A: Linear Convolution via the DFT

This appendix shows that Eqs. (31) and (32) are sufficient conditions that the product of the DFTs yields the correct linear convolution output of the corresponding sequences in OLA and OLS, respectively.

Let us consider the following periodic convolution with a periodicity matrix $\boldsymbol{N}$:

$$\tilde{u}_i(\boldsymbol{k}') = \sum_{\boldsymbol{n}\in\mathcal{N}_{\boldsymbol{s}'}(\boldsymbol{N})} \tilde{h}(\boldsymbol{n})\tilde{v}_i(\boldsymbol{k}'-\boldsymbol{n}),\ \boldsymbol{k}'\in\mathcal{N} \quad (\text{A}\cdot 1)$$

where $\mathcal{N}_{\boldsymbol{s}'}(\boldsymbol{N}) = \{\boldsymbol{k}|\boldsymbol{k}-\boldsymbol{s}'\in\mathcal{N}(\boldsymbol{N})\}$, $\boldsymbol{s}'\in\mathcal{N}$, and $\tilde{h}(\boldsymbol{k}')$ and $\tilde{v}_i(\boldsymbol{k}')$ are the periodic extensions of the impulse response $h(\boldsymbol{k}')$ and the sequence $v_i(\boldsymbol{k}')$ in Eq. (10) for OLA or Eq. (15) for OLS, respectively:

$$\tilde{h}(\boldsymbol{k}') = \sum_{\boldsymbol{r}\in\mathcal{N}} h(\boldsymbol{k}'-\boldsymbol{N}\boldsymbol{r}),\ \boldsymbol{k}'\in\mathcal{N}, \quad (\text{A}\cdot 2)$$

$$\tilde{v}_i(\boldsymbol{k}') = \sum_{\boldsymbol{r}\in\mathcal{N}} v_i(\boldsymbol{k}'-\boldsymbol{N}\boldsymbol{r}),\ \boldsymbol{k}'\in\mathcal{N}. \quad (\text{A}\cdot 3)$$

Note that the vector $\boldsymbol{s}'$ does not affect the result $\tilde{u}_i(\boldsymbol{k}')$ at all due to the periodic property of $\tilde{h}(\boldsymbol{k}')$ and $\tilde{v}_i(\boldsymbol{k}')$. The periodic convolution in Eq. (A·1) is equivalent to the product of the corresponding DFTs in Eq. (36), where $U_i(\boldsymbol{\ell})$, $V_i(\boldsymbol{\ell})$ and $H(\boldsymbol{\ell})$ are the matrix-$\boldsymbol{N}$ DFTs of $\tilde{u}_i(\boldsymbol{k}'), \tilde{v}_i(\boldsymbol{k}')$ and $\tilde{h}(\boldsymbol{k}')$ for $\boldsymbol{k}'\in\mathcal{N}(\boldsymbol{N})$, respectively.

Lemma A1: In OLA case, Eq. (31) is a sufficient condition that the sequence $u_i(\boldsymbol{k}')$ provided in Eq. (11) is related with $\tilde{u}_i(\boldsymbol{k}')$ in Eq. (A·1) as follows:

$$\tilde{u}_i(\boldsymbol{k}') = u_i(\boldsymbol{k}'),\ \boldsymbol{k}'\in\mathcal{N}_h(\boldsymbol{N}_S). \quad (\text{A}\cdot 4)$$

Proof: Note that $\mathcal{N}_h \subseteqq \mathcal{N}_h(\boldsymbol{N}_S)$ from Eqs. (2) and (12). If the condition in Eq. (31) is satisfied, then $\mathcal{N}_h \subseteqq \mathcal{N}_{\boldsymbol{s}'}(\boldsymbol{N})$, where $\boldsymbol{s}' = \boldsymbol{s}$, and Eq. (A·1) can be reduced to

$$\tilde{u}_i(\boldsymbol{k}') = \sum_{\boldsymbol{r}\in\mathcal{N}}\sum_{\boldsymbol{n}\in\mathcal{N}_h} h(\boldsymbol{n})v_i(\boldsymbol{k}'-\boldsymbol{n}-\boldsymbol{N}\boldsymbol{r}),$$

$$= \sum_{r\in\mathcal{N}} u_i(\boldsymbol{k}'-\boldsymbol{N}\boldsymbol{r}),\ \boldsymbol{k}'\in\mathcal{N}. \quad (A\cdot 5)$$

Besides, under the condition in Eq. (31), $u_i(\boldsymbol{k}'-\boldsymbol{N}\boldsymbol{r}) = 0$ for $\boldsymbol{k}'\in\mathcal{N}_h(\boldsymbol{N}_S), \boldsymbol{r}\neq\mathbf{o}$. Hence, from Eq. (A· 5), the relation in Eq. (A· 4) is derived. □

In the result, under the condition in Eq. (31), we can obtain the sequence $u_i(\boldsymbol{k}')$ provided in Eq. (11) via the DFT.

Lemma A2: In OLS case, Eq. (32) is a sufficient condition that the sequence $u_i(\boldsymbol{k}')$ provided in Eq. (17) is related with $\tilde{u}_i(\boldsymbol{k}')$ in Eq. (A· 1) as follows:

$$\tilde{u}_i(\boldsymbol{k}') = u_i(\boldsymbol{k}'),\ \boldsymbol{k}'\in\mathcal{N}(\boldsymbol{N}_S). \quad (A\cdot 6)$$

Proof: If the condition in Eq. (32) is satisfied, since $\mathcal{N}_h \subseteqq \mathcal{N}_{s'}(\boldsymbol{N})$ for some $\boldsymbol{s}'\in\mathcal{N}$, then Eq. (A· 1) can be reduced to

$$\tilde{u}_i(\boldsymbol{k}') = \sum_{n\in\mathcal{N}_h} h(\boldsymbol{n}) \sum_{r\in\mathcal{N}} v_i(\boldsymbol{k}'-\boldsymbol{n}-\boldsymbol{N}\boldsymbol{r}),\quad \boldsymbol{k}'\in\mathcal{N}. \quad (A\cdot 7)$$

Besides, note that $\boldsymbol{k}'-\boldsymbol{n}\in\bar{\mathcal{N}}_h(\boldsymbol{N}_S)$, provided $\boldsymbol{k}'\in\mathcal{N}(\boldsymbol{N}_S)$ and $\boldsymbol{n}\in\mathcal{N}_h$, and that $v_i(\boldsymbol{k}') = 0$ for $\boldsymbol{k}'\notin\bar{\mathcal{N}}_h(\boldsymbol{N}_S)$. In result, under the condition in Eq. (32), $v_i(\boldsymbol{k}'-\boldsymbol{n}-\boldsymbol{N}\boldsymbol{r}) = 0$ for $\boldsymbol{k}'\in\mathcal{N}(\boldsymbol{N}_S), \boldsymbol{n}\in\mathcal{N}_h$ and $\boldsymbol{r}\neq\mathbf{o}$. Hence, from Eq. (A· 7), the relation in Eq. (A· 6) is derived. □

In the result, under the condition in Eq. (32), we can obtain the sequence $u_i(\boldsymbol{k}')$ provided in Eq. (17) via the DFT.

Appendix B: Lattice Alteration via the DFT

This appendix shows that, if and only if the periodicity matrix $\boldsymbol{N}$ is a crm of $\boldsymbol{L}$ and $\boldsymbol{M}$, then the matrix-$\boldsymbol{L}$ up-sampling and the matrix-$\boldsymbol{M}$ down-sampling can be implemented in the DFT domain without the redundancy.

Lemma A3: If and only if $\boldsymbol{N}$ is an rm of $\boldsymbol{L}$, then $\tilde{v}_i(\boldsymbol{k}')$ in Eq. (A· 3) can be represented as

$$\tilde{v}_i(\boldsymbol{k}') = [\tilde{x}_i]_{\uparrow\boldsymbol{L}}(\boldsymbol{k}'), \quad (A\cdot 8)$$

$$\tilde{x}_i(\boldsymbol{n}') = \sum_{r\in\mathcal{N}} x_i(\boldsymbol{n}'-\boldsymbol{Q}\boldsymbol{r}),\ \boldsymbol{n}'\in\mathcal{N}. \quad (A\cdot 9)$$

where $\boldsymbol{Q} = \boldsymbol{L}^{-1}\boldsymbol{N}$ and $x_i(\boldsymbol{n}')$ is the sequence provided in Eq. (20) for EOLA or Eq. (29) for EOLS.

Proof: If and only if $\boldsymbol{N}$ is an rm of $\boldsymbol{L}$, then there exists some integer matrix $\boldsymbol{Q} = \boldsymbol{L}^{-1}\boldsymbol{N}$ and the following relation is derived from Eq. (19):

$$v_i(\boldsymbol{k}'-\boldsymbol{N}\boldsymbol{r}) = \begin{cases} x_i(\boldsymbol{L}^{-1}\boldsymbol{k}'-\boldsymbol{Q}\boldsymbol{r}), & \boldsymbol{k}'\in\mathcal{L}(\boldsymbol{L}) \\ 0, & \text{elsewhere} \end{cases}. \quad (A\cdot 10)$$

Substituting the above relation to Eq. (A· 3) yields the following relation:

$$\tilde{v}_i(\boldsymbol{k}') = \begin{cases} \sum_{r\in\mathcal{N}} x_i(\boldsymbol{L}^{-1}\boldsymbol{k}'-\boldsymbol{Q}\boldsymbol{r}), & \boldsymbol{k}'\in\mathcal{L}(\boldsymbol{L}) \\ 0, & \text{elsewhere} \end{cases}. \quad (A\cdot 11)$$

This relation is identical to Eq. (A· 8). □

Lemma A4: Let $V_i(\boldsymbol{\ell})$ be the matrix-$\boldsymbol{N}$ DFT of $\tilde{v}_i(\boldsymbol{k}')$ in Eq. (A· 3) for $\boldsymbol{k}'\in\mathcal{N}(\boldsymbol{N})$. If and only if $\boldsymbol{N}$ is an rm of $\boldsymbol{L}$, then $V_i(\boldsymbol{\ell})$ can be obtained as Eq. (35), where $\boldsymbol{Q} = \boldsymbol{L}^{-1}\boldsymbol{N}$ and $X_i(\boldsymbol{q})$ is the matrix-$\boldsymbol{Q}$ DFT of $\tilde{x}_i(\boldsymbol{n}')$ in Eq. (A· 9) for $\boldsymbol{n}'\in\mathcal{N}(\boldsymbol{Q})$.

Proof: From the definition in Eq. (4), $V_i(\boldsymbol{\ell})$ is represented as

$$V_i(\boldsymbol{\ell}) = \sum_{k'\in\mathcal{N}(\boldsymbol{N})} \tilde{v}_i(\boldsymbol{k}') \left[\boldsymbol{W}_{\boldsymbol{N}}^{(\mathrm{g})}\right]_{\boldsymbol{\ell},\boldsymbol{k}'},\quad \boldsymbol{\ell}\in\mathcal{N}(\boldsymbol{N}^T). \quad (A\cdot 12)$$

If and only if $\boldsymbol{N}$ is an rm of $\boldsymbol{L}$, then there exists some integer matrix $\boldsymbol{Q} = \boldsymbol{L}^{-1}\boldsymbol{N}$, and the above equation can be decomposed from the division theorem for integer vectors [2], [13] as follows:

$$V_i(\boldsymbol{\ell}) = \sum_{n'\in\mathcal{N}(\boldsymbol{Q})} \sum_{r\in\mathcal{N}(\boldsymbol{L})} \tilde{v}_i(\boldsymbol{L}\boldsymbol{n}'+\boldsymbol{r}) \left[\boldsymbol{W}_{\boldsymbol{L}\boldsymbol{Q}}^{(\mathrm{g})}\right]_{\boldsymbol{\ell},\boldsymbol{L}\boldsymbol{n}'+\boldsymbol{r}},\quad \boldsymbol{\ell}\in\mathcal{N}(\boldsymbol{N}^T). \quad (A\cdot 13)$$

From Eq. (A· 8), since $\tilde{v}_i(\boldsymbol{L}\boldsymbol{n}'+\boldsymbol{r})$ is equal to $\tilde{x}_i(\boldsymbol{n}')$ for $\boldsymbol{r} = \mathbf{o}$ and zero for $\boldsymbol{r}\neq\mathbf{o}$, Eq. (A· 13) can be reduced to

$$V_i(\boldsymbol{\ell}) = \sum_{n'\in\mathcal{N}(\boldsymbol{Q})} \tilde{x}_i(\boldsymbol{n}') \left[\boldsymbol{W}_{\boldsymbol{Q}}^{(\mathrm{g})}\right]_{\boldsymbol{\ell},\boldsymbol{n}'},\quad \boldsymbol{\ell}\in\mathcal{N}(\boldsymbol{N}^T), \quad (A\cdot 14)$$

where $\left[\boldsymbol{W}_{\boldsymbol{L}\boldsymbol{Q}}^{(\mathrm{g})}\right]_{\boldsymbol{\ell},\boldsymbol{L}\boldsymbol{n}'} = \left[\boldsymbol{W}_{\boldsymbol{Q}}^{(\mathrm{g})}\right]_{\boldsymbol{\ell},\boldsymbol{n}'}$ from Eq. (6). The above relation is identical to Eq. (35). □

Lemma A5: Let $U_i(\boldsymbol{\ell})$ be the matrix-$\boldsymbol{N}$ DFT of $\tilde{u}_i(\boldsymbol{k}')$ in Eq. (A· 1) for $\boldsymbol{k}'\in\mathcal{N}(\boldsymbol{N})$, and $\tilde{y}_i(\boldsymbol{m}')$ be $[\tilde{u}_i]_{\downarrow\boldsymbol{M}}(\boldsymbol{m}')$. If and only if $\boldsymbol{N}$ is an rm of $\boldsymbol{M}$, then $\tilde{y}_i(\boldsymbol{m}')$ can be obtained by periodically extending the matrix-$\boldsymbol{P}$ IDFT of $Y_i(\boldsymbol{p})$ provided in Eq. (38), where $\boldsymbol{P} = \boldsymbol{M}^{-1}\boldsymbol{N}$.

Proof: $\tilde{y}_i(\boldsymbol{m}')$ can be represented in terms of $U_i(\boldsymbol{\ell})$ as

$$\tilde{y}_i(\boldsymbol{m}') = \frac{1}{J(\boldsymbol{N})} \sum_{\ell\in\mathcal{N}(\boldsymbol{N}^T)} U_i(\boldsymbol{\ell}) \left[\boldsymbol{W}_{\boldsymbol{N}}^{(\mathrm{g})}\right]^*_{\boldsymbol{\ell},\boldsymbol{M}\boldsymbol{m}'},\quad \boldsymbol{m}'\in\mathcal{N}, \quad (A\cdot 15)$$

from the relation between the discrete Fourier series and the DFT [13] and the relation $\tilde{y}_i(\boldsymbol{m}') = \tilde{u}_i(\boldsymbol{M}\boldsymbol{m}')$. If and only if $\boldsymbol{N}$ is an rm of $\boldsymbol{M}$, then there exists some integer matrix $\boldsymbol{P} = \boldsymbol{M}^{-1}\boldsymbol{N}$, and the above equation can be decomposed from the division theorem for integer vectors as follows:

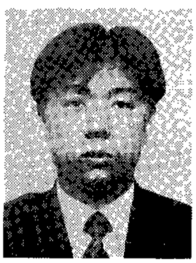
$$\tilde{y}_i(\boldsymbol{m}') = \frac{1}{J(\boldsymbol{M})} \sum_{\boldsymbol{p}\in\mathcal{N}(\boldsymbol{P}^T)} \left\{ \frac{1}{J(\boldsymbol{P})} \sum_{\boldsymbol{r}\in\mathcal{N}(\boldsymbol{M}^T)} U_i(\boldsymbol{P}^T\boldsymbol{r}+\boldsymbol{p}) \right\} \left[\boldsymbol{W}_{\boldsymbol{P}}^{(\mathrm{g})}\right]^*_{\boldsymbol{p},\boldsymbol{m}'}, \ \boldsymbol{m}' \in \mathcal{N}, \qquad (\mathrm{A\cdot 16})$$

where $\left[\boldsymbol{W}_{\boldsymbol{MP}}^{(\mathrm{g})}\right]_{\boldsymbol{P}^T\boldsymbol{r}+\boldsymbol{p},\boldsymbol{M}\boldsymbol{m}'} = \left[\boldsymbol{W}_{\boldsymbol{P}}^{(\mathrm{g})}\right]_{\boldsymbol{p},\boldsymbol{m}'}$ and $J(\boldsymbol{MP}) = J(\boldsymbol{M})J(\boldsymbol{P})$ (the multiplicative property of determinants). From the above equation, $\tilde{y}_i(\boldsymbol{m}')$ can be obtained as

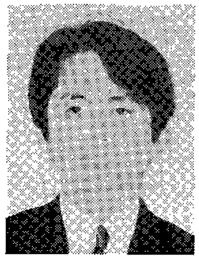
$$\tilde{y}_i(\boldsymbol{m}') = \hat{y}_i(((\boldsymbol{m}'))_{\boldsymbol{P}}), \qquad (\mathrm{A\cdot 17})$$

where $\hat{y}_i(\boldsymbol{m}')$ denotes the matrix-$\boldsymbol{P}$ IDFT of $Y_i(\boldsymbol{p})$ provided in Eq. (38). □

Note that Eq. (35) requires no operation for zero-value samples inserted by up-sampling, and that Eq. (38) requires no operation for obtaining samples to be discarded by down-sampling. Because of Lemma A4 and A5, by choosing $\boldsymbol{N}$ as a crm of $\boldsymbol{L}$ and $\boldsymbol{M}$, we can implement the whole system of sampling lattice alteration in Fig. 5 via the DFT without the redundancy.



Shogo Muramatsu was born in Tokyo, Japan, on May 2, 1970. He received the B.E. and M.E. degrees in electrical engineering from Tokyo Metropolitan University, Tokyo, Japan, in 1993 and 1995, respectively. He is currently a candidate for the D.E. degree at Tokyo Metropolitan University. His research interests are in digital signal processing, multirate systems, and image processing. Mr.Muramatsu is a Student member of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) of USA.



Hitoshi Kiya was born in Yamagata, Japan, on November 16, 1957. He received the B.E. and M.E. degrees in electrical engineering from Nagaoka University of Technology, Niigata, Japan, and the D.E. degree in electrical engineering from Tokyo Metropolitan University, Tokyo, Japan, in 1980, 1982, and 1987, respectively. In 1982, he joined Tokyo Metropolitan University, where he is currently an Associate Professor of Electronics and Information Engineering. His research interests are in digital signal processing, multirate systems, adaptive filtering, image processing, and efficient algorithms for VLSI implementation. Dr.Kiya is a Member of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) of USA, the Image Electronics Engineers of Japan and the Institute of Television Engineers of Japan.