# ЭФФЕКТИВНЫЙ АЛГОРИТМ ДВУМЕРНОЙ НЕРАЗДЕЛИМОЙ ДЕЦИМАЦИИ

Рычков А.Н.

Московский энергетический институт (технический университет)
105835, ГСП, Москва Е-250, ул. Красноказарменная, д.17
Тел.: (095)-362 7463, E-mail: ra27@mail.ru

Известно, что неразделимые многоскоростные системы, являясь общим случаем, обладают возможностью более эффективного решения задач цифровой обработки сигналов, но сложность реализации таких неразделимых операций как децимация, фильтрация и продолжение сигналов накладывают ограничения на их использование.

В данной статье предлогается алгоритм эффективной реализации операции децимации для произвольной решетки. Основная идея предлогаемого подхода заключается в представлении неразделимой решетки как суммы разделимых решеток. Это можно описать следующей формулой

$$LAT(\mathbf{M}) = \sum_{k \in N(\mathbf{S})|_{LAT(\mathbf{M})}} \mathbf{z}^{\mathbf{k}} \cdot LAT(\mathbf{S}),$$

где $N(\mathbf{S})|_{LAT(\mathbf{M})}$ классы смежности $LAT(\mathbf{S})$, определенной на $LAT(\mathbf{M})$, а $\mathbf{S}=diag(J(\mathbf{M}),J(\mathbf{M}))$ $(J(\mathbf{M})=|det(\mathbf{M})|)$ . На рис.1 показан пример такого представления для шахматной решетки.

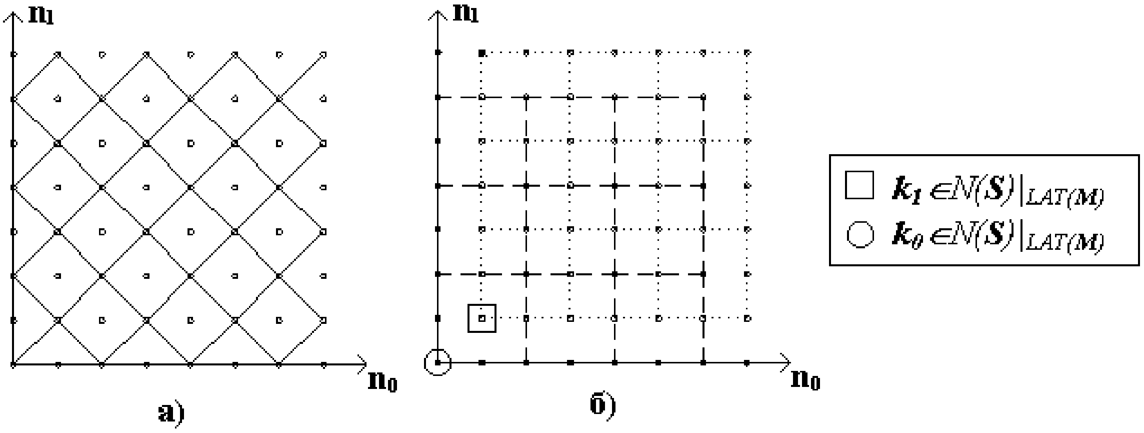

Рис. 1 Шахматная решетка (а); и ее разделимое представление (б)

Такой подход позволяет: упростить вычисления координат решетки децимации, оптимизировать преобразование координат и устранить сдвиг на этапе преобразования координат. Данный метод может быть расширен для сигналов с произвольным числом измерений.

## Литература

1. Vaidyanathan P.P, Multirate Systems and Filter Banks, Englewood Cliffs, NJ: Prentice-Hall, 1993, Chapter 12.

## EFFICIENT 2D NONSEPARABLE DECIMATION ALGORITHM

Rychkov A.

Moscow Power Engineering Institute, Department of Electrical Physics
105835, GSP, Moscow E-250, 17 Krasnokazarmennaya st.
Phone: +7 (095) 362 7463, E-mail: ra27@mail.ru

It's known that nonseparable multirate systems (being general case of such systems) are more flexible and can produce more efficient solutions of signal processing tasks, but the complexity of such nonseparable operations as decimation, filtering and signal extension place limitations on its application. This paper considers efficient decimation algorithm for an arbitrary matrix.

Decimation operator and its lattice *LAT(**M**)* are definded by nonsingular matrix [1]:

$$\mathbf{M}=\begin{bmatrix} m_{00} & m_{01} \\ m_{10} & m_{11} \end{bmatrix}. \qquad (1)$$

While decimated coordinates of *LAT(**M**)* samples are transformed by
**b**=**M**$^{-1}$**a**, (2)

where **a**, **b** – vectors-columns of initial and decimated signals coordinates.
*Fundamental parallelepiped* is defined as [1]:
*FPD(**M**)*=**Mx**, **x**$\in[0,1)^2$ (3)

Samples belonging *FPD(**M**)* are called as *cosets* and noted *N(**M**)*. Number of cosets equals *J(**M**)*=|det(**M**)|. (4)

The signals used in practice have positive coordinates but after decimation can be shifted to negative area and are to be corrected.

Simple decimation algorithm for arbitrary matrix has following steps:

1) Defining *LAT(**M**)* samples;
2) Coordinates transformation;
3) Negative shift correction;
4) Moving sample from coordinates **a** ((2)) to coordinates **b**.

The biggest part of algorithm performance time is due to first three steps. Decimation of $N_0N_1$ sized signal takes us arithmetic operations (далее operations) equals to

$$O > 3N_1N_2 + \frac{N_1N_2}{\mathbf{J}(\mathbf{M})}, \qquad (5)$$

where first term is due to first and second steps and second one is due to third step.

Main idea of proposed algorithm is *separable presentation* of nonseparable lattice. Fig. 1 shows such a presentation of a quincunx lattice.

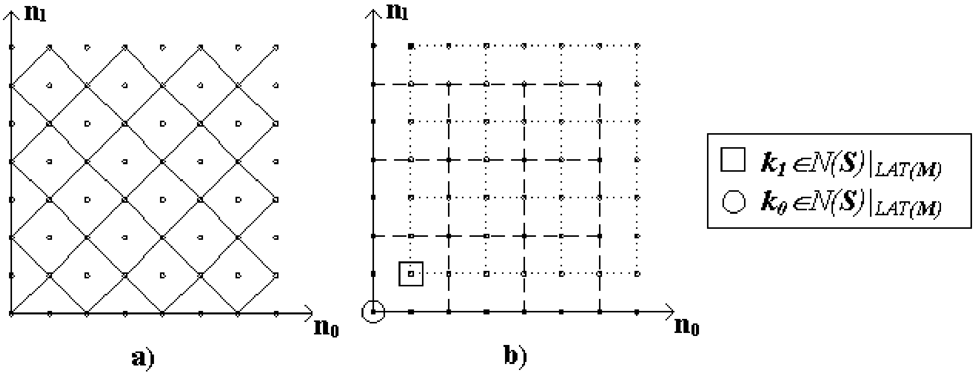

Fig. 1 Quincunx lattice (a); and its separable presentation (b)

For general case this presentation can be written as

$$LAT(\mathbf{M}) = \sum_{k\in N(\mathbf{S})|_{LAT(\mathbf{M})}} \mathbf{z}^{\mathbf{k}} \cdot LAT(\mathbf{S}), \qquad (6)$$

where $N(\boldsymbol{S})|_{LAT(\boldsymbol{M})}$ are cosets of $LAT(\boldsymbol{S})$, defined at $LAT(\boldsymbol{M})$ (Fig. 1), and $\boldsymbol{S}=diag(J(\boldsymbol{M}),J(\boldsymbol{M}))$. Formula (6) produces

$$J(\boldsymbol{S})|_{LAT(\boldsymbol{M})}=\frac{J(\mathbf{S})}{J(\mathbf{M})}=J(\mathbf{M}) \qquad (7)$$

As will be shown bellow this presentation allows us to

1) Calculate $LAT(\boldsymbol{M})$ coordinates using one operation per sample;
2) optimize coordinate transformation;
3) eliminate shift before coordinates transformation.

A decimation algorithm based on such a presentation consists of next steps:

1) Calculation of $N(\boldsymbol{S})|_{LAT(\boldsymbol{M})}$;
2) Shift eliminating and defining of memory to be allocated;
3) Defining of $LAT(\boldsymbol{M})$ samples and samples coordinates transformation.

Let's write each step.

From (3) follows, to determine the set $N(\boldsymbol{S})|_{LAT(\boldsymbol{M})}$ one should test samples of such square $0\le n_1<J(\boldsymbol{S})$, $0\le n_2<J(\boldsymbol{S})$ using (2), so (2) produces integer **b** for $LAT(\boldsymbol{M})$ samples. This step takes us $3J(\boldsymbol{M})$ operations.

If the shift of decimated signal is known one can shift initial signal in order to eliminate that shift, i.e.

$$\mathbf{b}+\mathbf{s}=\mathbf{Ma} \quad □\mathbf{b}=\mathbf{M}(\mathbf{a}-\mathbf{M}^{-1}\mathbf{s}), \qquad (8)$$

where **s** – decimated signal shift and $\mathbf{M}^{-1}\mathbf{s}$ eliminating shift of initial signal. The shift **s** and volume of memory $D_0D_1$ to be allocated are defined by rectangular fitting decimated signal. This rectangular can be found by decimating samples, defining initial signal shape defined at $LAT(\boldsymbol{M})$, and finding edge points. Initial signal shape is defined by $J(\boldsymbol{M})$ rectangulars which first points belong to $N(\boldsymbol{S})|_{LAT(\boldsymbol{M})}$ and others are defined by initial signal shape. After rectangular tops transformation using (2), the coordinates of fitting rectangular are defined which produce shift **s** and memory volume to be allocated $D_0D_1$. Then cosets and signal sizes eliminating shift are calculated as follows

$$N'(\boldsymbol{M})=N(\boldsymbol{M})+\mathbf{s}, \qquad (9)$$

$$\begin{bmatrix} N_2' & N_1' \end{bmatrix}^T=\begin{bmatrix} N_2 & N_1 \end{bmatrix}^T+\mathbf{s}.$$

This step of algorithm takes us $26J(\boldsymbol{M})$ operations.

Coordinates calculation and transformation is performed for each sublattice $LAT(\boldsymbol{S})$ separately. In physical memory 2D signal is stored as 1D signal it is efficient to be taken into account. Let decimated signal is stored in memory column by column which size is defined previously as $D_1$. Let one expand (2)

$$\begin{bmatrix} b_0 \\ b_1 \end{bmatrix}=\begin{bmatrix} m_{00} & m_{01} \\ m_{10} & m_{11} \end{bmatrix}\begin{bmatrix} a_0 \\ a_1 \end{bmatrix}=\begin{bmatrix} m_{00}a_0+m_{01}a_1 \\ m_{10}a_0+m_{11}a_1 \end{bmatrix},$$

and write it in terms of 1D signal

$$t=b_0D_1+b_1=c_0a_0+c_1a_1, \qquad (10)$$

where t – 1D coordinate, $c_0=(m_{00}D_1+m_{10})$, $c_1=(m_{01}D_1+m_{11})$. While (10) is used, one takes first coset, calculates coordinates of lattice $\boldsymbol{z}^{k0}□LAT(\boldsymbol{S})|_{LAT(\boldsymbol{M})}$ and substitutes it in (10), then one takes second coset and calculates coordinates of lattice $\boldsymbol{z}^{k1}□LAT(\boldsymbol{S})|_{LAT(\boldsymbol{M})}$ and substitutes it in (10), etc. So any coset has $N_0N_1J^{-2}(\boldsymbol{M})$ values of set $\{(a_0, a_1)\}$, while variable $a_0$ and $a_1$ have $N_0J^{-1}(\mathbf{M})$ and $N_1J^{-1}(\mathbf{M})$ different values. It is enough to calculate sets $\{c_0a_0\}$, $\{c_1a_1\}$ and substitute its combinations in (10). Calculation of such sets can be described by next formula

$$\{c_ia_i\}=[c_ik_i^j:c_iJ(\mathbf{M}):c_i N_i^{'}], \qquad (11)$$

where [a:b:c] notes the vector, changing from *a* with step *b* to *c*, and $k_i^j$ i-th component (scalar) of j-th coset belonging to $N'(\boldsymbol{S})|_{LAT(\boldsymbol{M})}$. This step of algorithm takes us $N_1+N_2$ operations.

The whole amount of operations to be done using this algorithm is

$$O'=3J^2(\mathbf{M})+26J(\mathbf{M})+N_1+N_2. \qquad (12)$$

The gain of this method (O/O′) while $N_0=N_1=N$ is

$$G>1{,}5N \qquad (13)$$

This paper shows that the nonseparable decimation has complexity comparable with separable one. However this technique was proposed for 2D signals and it can be also extended to an arbitrary number of dimensions.

## References

1. Vaidyanathan P.P, Multirate Systems and Filter Banks, Englewood Cliffs, NJ: Prentice-Hall, 1993, Chapter 12.