Lecture 4. Efficient computational algorithms for digital image processing

(2 hours)

Filtering in signal domain.

Direct digital convolution $\mathbf{b} = \{b_k\} = \left\{\sum_n h_n a_{k-n}\right\}$ .Computational complexity.

Recursive filters: $b_k = \sum_{n=0}^{N_t-1} h_n^{tr} a_{k-n} + \sum_{n=1}^{N_r-1} h_n^{rc} b_{k-n}$. Computational complexity. Recursive computing local mean.

Recursive computation of DFT and DCT in a moving window:

1-D: $\alpha_r^1 = \left[\alpha_r^0 + \frac{1}{\sqrt{N_1}}(a_N - a_0)\right] exp(-i2\pi r / N)$

2-D: $\alpha_{r,s}^{0,1} = \left\{\alpha_{r,s}^{0,0} + \frac{1}{\sqrt{N_1 N_2}} \sum_{k=0}^{N_1-1} (a_{k,N_2} - a_{k,0}) exp\left(i2\pi \frac{kr}{N_1}\right)\right\} exp\left(-i2\pi \frac{s}{N_2}\right)$

Recursive computation of histograms and order statistics in moving window

$$h^{(k)}(m) = \sum_{n=k-N}^{k+N} \delta(a_n - m);\ h^{(k+1)}(m) = \sum_{n=k-N+1}^{k+N+1} \delta(a_n - m) = h^{(n)}(m) + \delta(a_{k+N+1} - m) - \delta(a_{k-N} - m)$$

Parallel and cascade digital filters. 2-D cascade digital filters.

Parallel and recursive implementation of digital filters. Sliding window filtering in transform domain

Filtering in transform domain using using FFT-type algorithms.

The principle of Fast Fourier Transform.

2-D DFT and the basic idea of Fast Transforms. Fast Hadamard Transform:

$$\alpha_{r_{n-1},\dots,r_0} = \frac{1}{\sqrt{N}} \sum_{k_{n-1}=0}^{1} (-1)^{k_{n-1}r_{n-1}} \cdots \sum_{k_1=0}^{1} (-1)^{k_1 r_1} \sum_{k_0=0}^{1} a_{k_{n-1},\dots,k_0} (-1)^{k_0 r_0}$$

DFT of a sequence of $N = N_1 N_2$

$$\alpha_{r_1,r_2} = \frac{1}{\sqrt{N_1 N_2}} \sum_{k_2=0}^{N_2} \sum_{k_1=0}^{N_1} a_{k_1,k_2}\, exp\left[i2\pi \frac{(k_2 N_1 + k_1)(r_1 N_2 + r_2)}{N_2 N_1}\right] = DFT_{k_1}\left\{exp\left(i2\pi \frac{k_1 r_2}{N}\right) \bullet DFT_{k_2}\{a_{k_1,k_2}\}\right\}$$

Radix-2 FFT. Graph representation of fast algorithms.

Matrix techniques in the theory of fast transforms and algorithms.

Elementary matrices: $[1\ \ 0];[0\ \ 1]$; and their derivatives.

Matrix building operations: direct matrix sum; “vertical” sum; direct (Kronecker) product

Fast transforms matrices as layered Kronecker ones. Factorization theorems.

Review of Fast Transforms.

$$\mathbf{HAD}_{2^n} = \prod_{r=0}^{n-1} \left[\mathbf{I}_{2^{n-r-1}} \otimes \mathbf{h}_2 \otimes \mathbf{I}_{2^r}\right] = \prod_{r=0}^{n-1} \left\{\mathbf{I}_{2^{n-r-1}} \otimes \left[\left([11] \otimes \mathbf{I}_{2^r}\right) \Xi \left([1\bar{1}] \otimes \mathbf{I}_{2^r}\right)\right]\right\}$$

$$\mathbf{HAR}_{2^n} = \left(1 \oplus \bigotimes_{r=0}^{n-1} 2^{(r-n)/2} \mathbf{I}_{2^r}\right) \prod_{r=0}^{n-1} \left\{\left[\left(\mathbf{I}_{2^r} \otimes [11]\right) \Xi \left(\mathbf{I}_{2^r} \otimes [1\bar{1}]\right)\right] \oplus \mathbf{I}_{2^n - 2^{r+1}}\right\}$$

$$\overline{\mathbf{FOUR}}_{2^n} = \left([11] \otimes \overline{\mathbf{FOUR}}_{2^{n-1}}\right) \Xi \left\{[1\bar{1}] \otimes \left[\overline{\mathbf{FOUR}}_{2^{n-1}} \cdot \left(\bigotimes_{s=0}^{n-2} \mathbf{d}_{2^{s-n}}\right)\right]\right\}$$

$$\mathbf{FOUR}_{2^n} = \mathbf{BRP}_{2^n} \cdot \prod_{r=0}^{n-1} \left(\mathbf{I}_{2(n-1-r)} \otimes \left[\mathbf{I}_{2^r} \oplus \left(\bigotimes_{s=0}^{r-1} \mathbf{d}_{2^{s+2-n}}\right)\right]\right) \cdot \left(\mathbf{I}_{2(n-r-1)} \otimes \mathbf{h}_2 \otimes \mathbf{I}_{2^r}\right),$$

Synthesis of fast transforms and algorithms: Pruned FFT, Quantized DFT

Fast DCT and DcST algorithms

Combined algorithms of DFTs

Combined DFT of two real-valued sequences:

$$\{c_k = a_k + ib_k\} \Rightarrow DFT \Rightarrow \{\gamma_r = \alpha_r + i\beta_r\};\ \{\alpha_r = (\gamma_r + \gamma_{N-r}^*)/2\};\{\beta_r = -i(\gamma_r - \gamma_{N-r}^*)/2\}$$

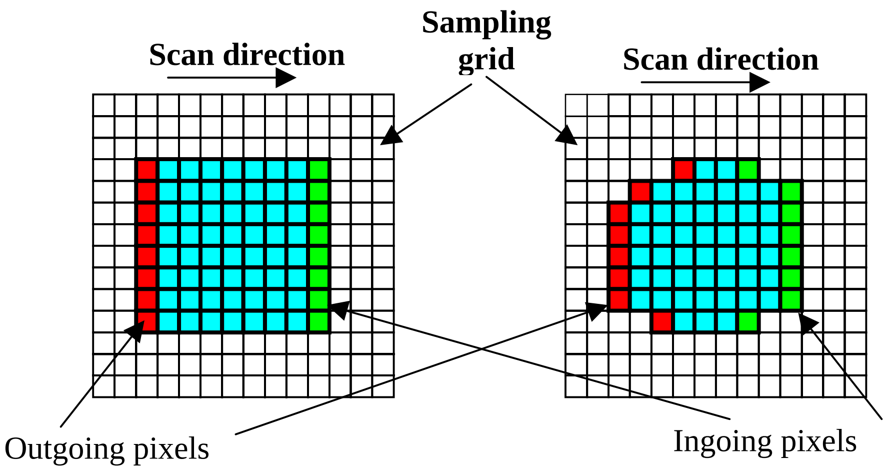
Combined DFT of a single sequence with even number of terms:

$$\{c_k = a_{2k} + ia_{2k+1}\} \Rightarrow DFT \Rightarrow \{\gamma_r = \alpha_r + i\beta_r\} \Rightarrow \{\alpha_r^{even} + i\alpha_r^{odd}\};\ \{\alpha_r = \left(\alpha_r^{even} + \alpha_r^{odd}\, exp(i\pi r / N)\right)/\sqrt{2}\};$$

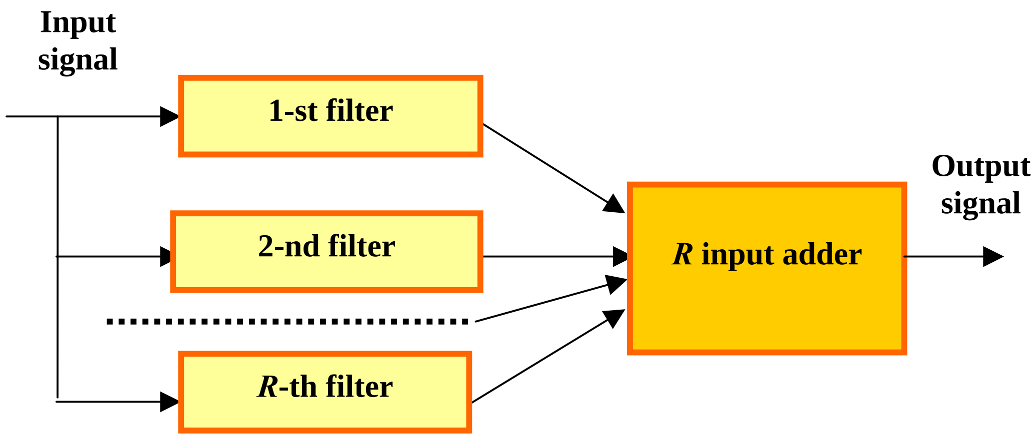
Combined DCT $\{c_k = a_k + i(-1)^k b_k\} \Rightarrow SDFT(1/2,0) \Rightarrow \{\gamma_r^{1/2,0} = \alpha_r^{DCT} + i\beta_{r+N}^{DCT}\};$

$\{\alpha_r^{DCT} = (\gamma_r^{1/2,0} - \gamma_{N-r}^*)/2\};\{\beta_r^{DCT} = (\gamma_{N-r}^{1/2,0} + \gamma_{N+r}^{1/2,0})/2\}$

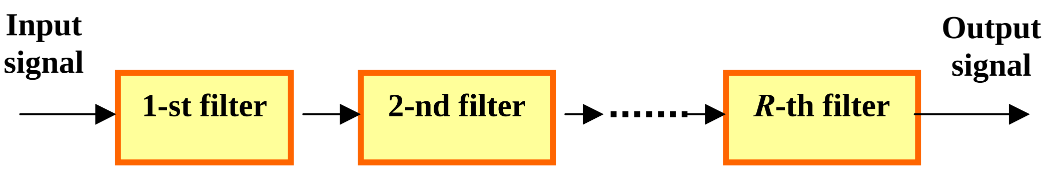
Efficient algorithms of filtering in signal domain



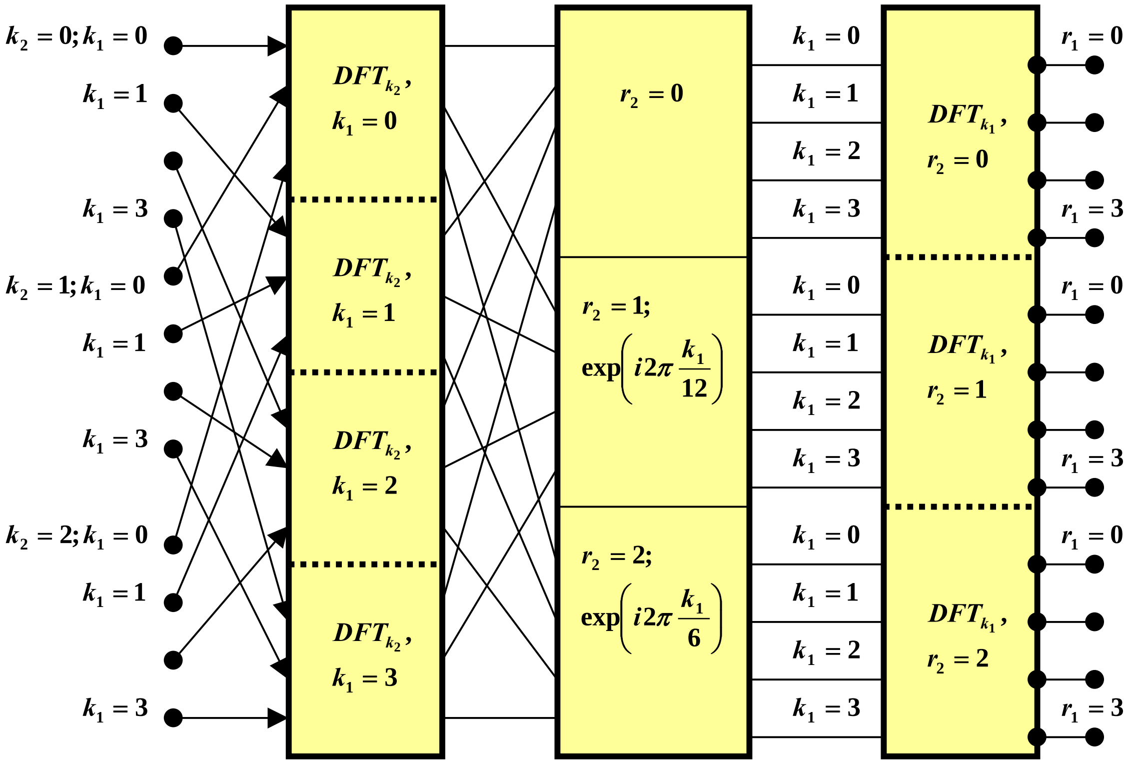
To the principle of recursive filtering in sliding window



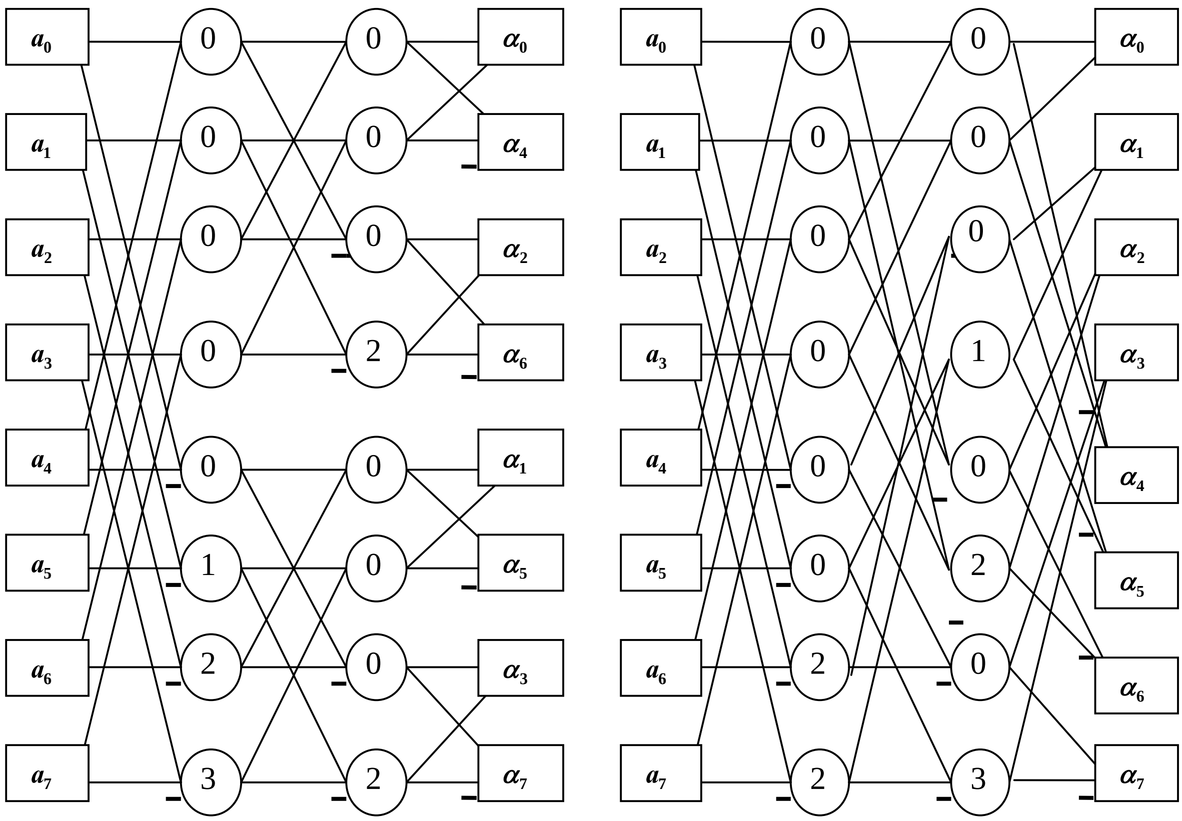
Parallel implementation of filtering



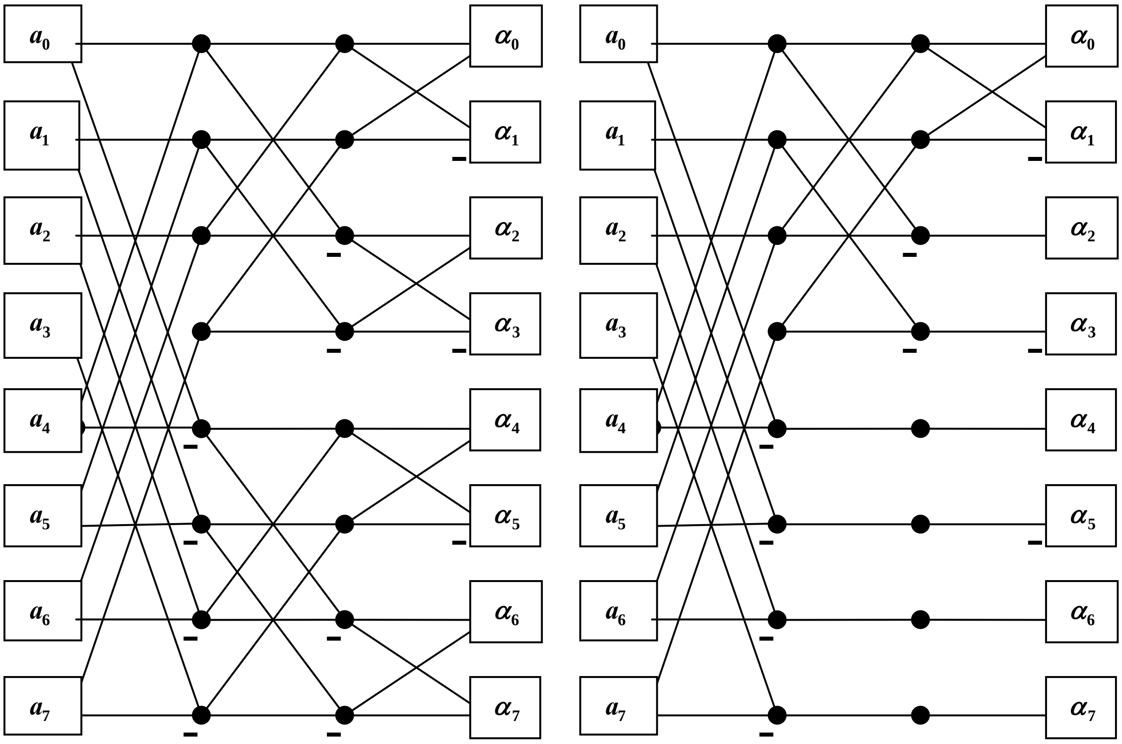
Cascade implementation of filtering



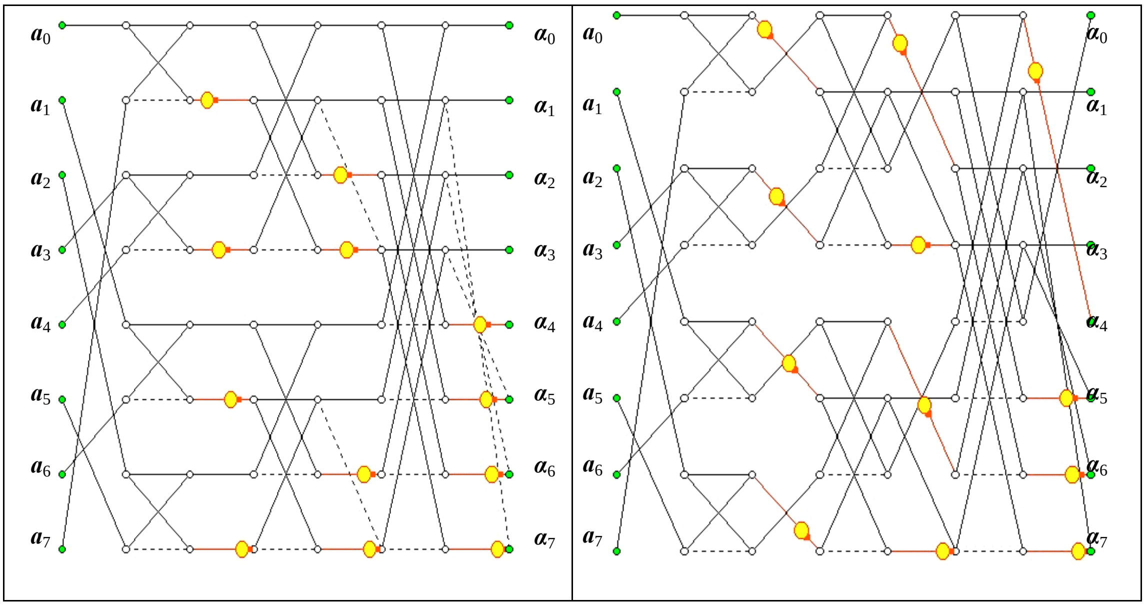
Flow diagram of FFT for $N = 12 = 3 \times 4$



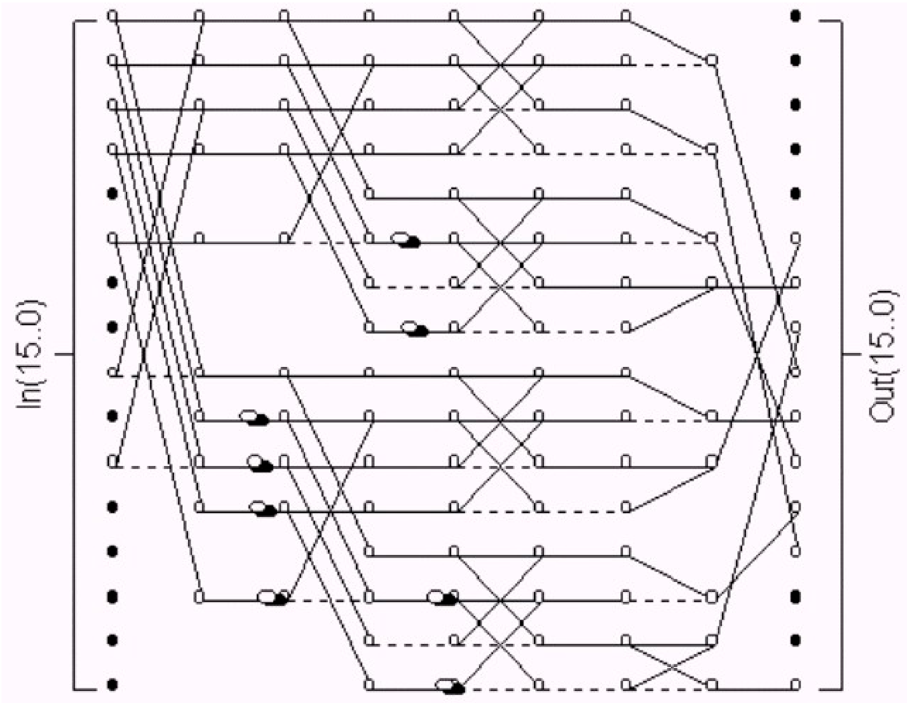
Flow charts of FFT "in place" (left) and of FFT without permutations algorithms; (numbers in nodes are values of index ***m*** in exponential factors **$\exp(i2\pi m/8)$**



Flow diagrams of Fast Hadamard (left) and Fast Haar (right) Transform algorithms



Illustrative flow diagram of the DCT (a) and DcST (b) for N=8. Solid lines denote addition at the graph nodes; dotted lines denote subtraction; circles denote multiplication by respective coefficients $\{d_{s,m}\}$ and $\{e_{s,m}\}$ (numerical values are not shown)

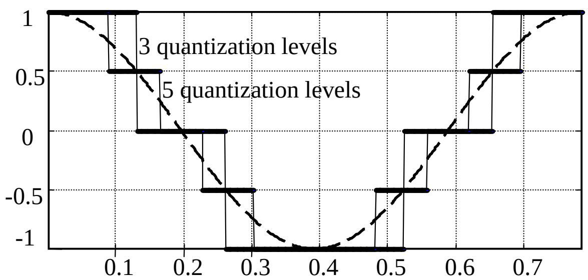


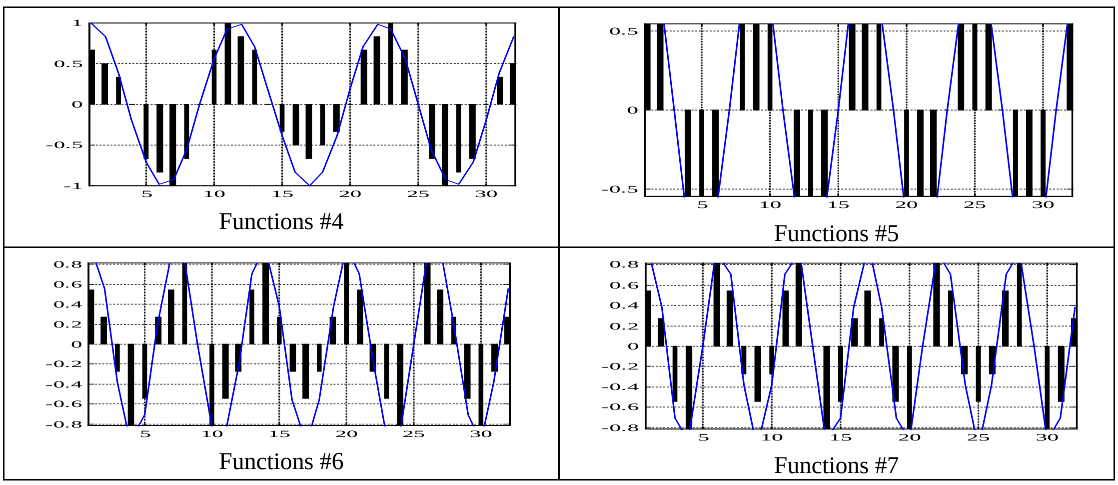
Illustrative flow diagram of a pruned DFT for N=16.

(Solid lines denote multiplication by 1, or $\sqrt{-1}$, doted lines denote multiplication by -1, or -j and solid lines with black bubbles denote multiplication by "twiddle" factors).

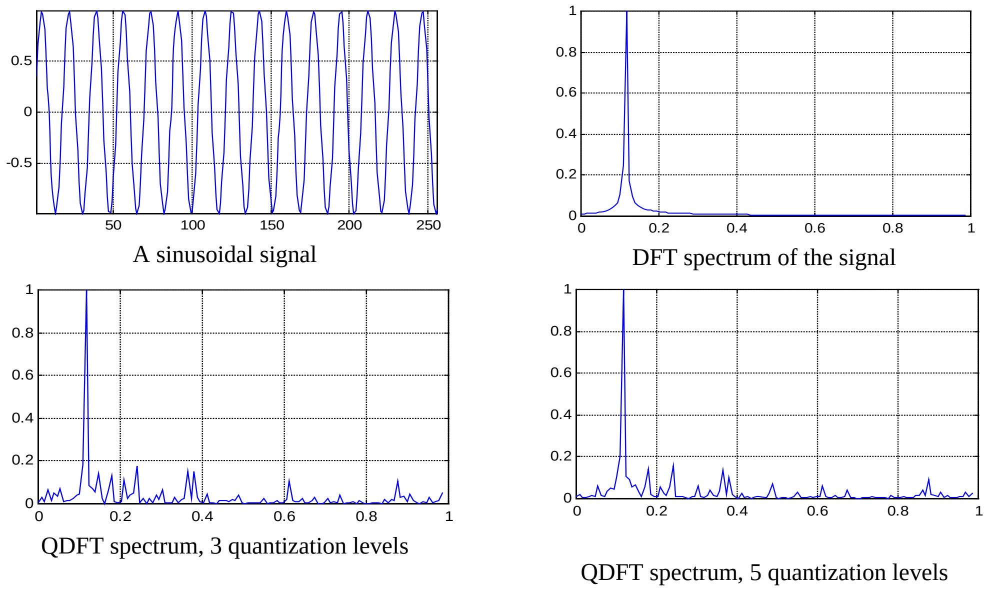
$$\mathbf{TF}_r = \begin{bmatrix} \cos\theta_r & \sin\theta_r \\ -\sin\theta_r & \cos\theta_r \end{bmatrix}$$

$$\mathbf{QTF}_r = \begin{bmatrix} \mathrm{qos}\,\theta_r & \mathrm{qin}\,\theta_r \\ -\mathrm{qin}\,\theta_r & \mathrm{qos}\,\theta_r \end{bmatrix}$$

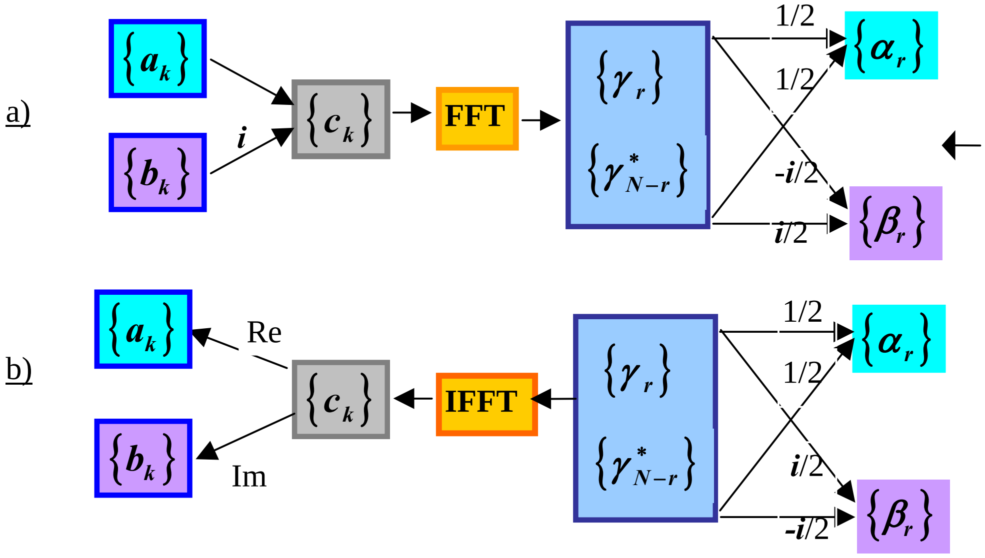




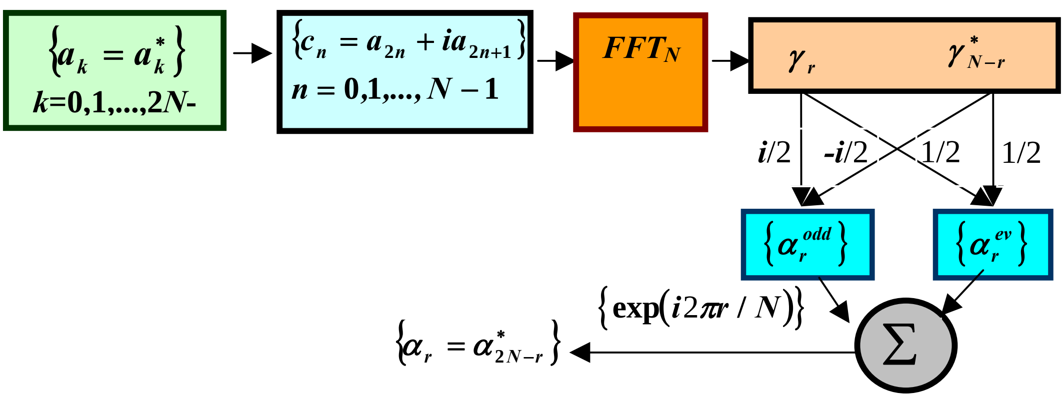
Basis functions of the DFT (linearly interpolated, solid lines) and the QDFT with 5 quantization levels (bars) for *N*=32



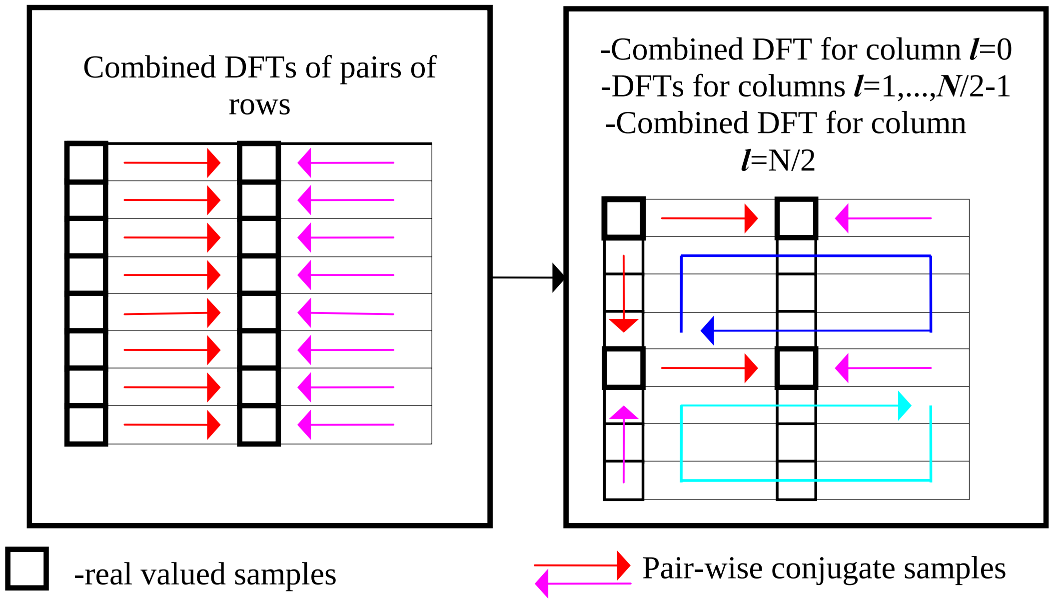
DFT and QDFT spectra of a sinusoidal signal



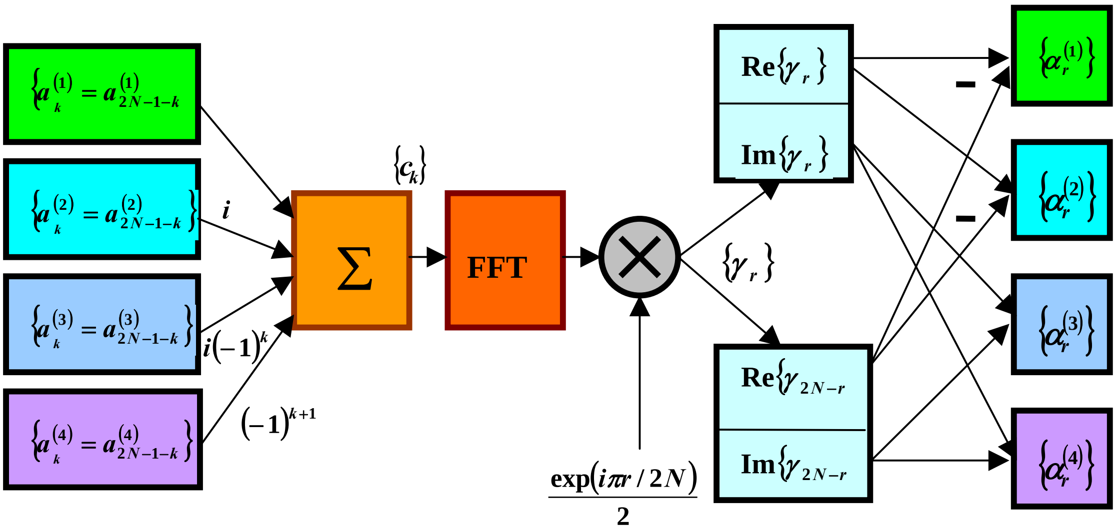
Flow diagram of the combined direct (a) and inverse (b) DFT of two real-valued signals



Flow diagram of the combined direct DFT of a real-valued signal with an even number of samples



Flow diagram of 2-D combined DFT for even number of rows and columns



Flow diagram of the fast FFT based combined algorithm for computing DCT of four signals