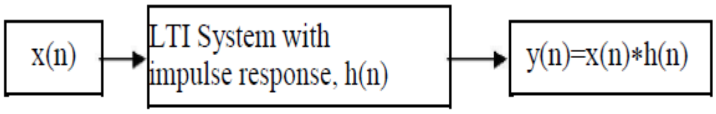
USES OF FFT IN LINEAR FILTERING

Linear filtering refers to obtaining the output, y(n) of a linear, time-invariant (LTI) system with impulse response, h(n) to an input, x(n). This process is often termed as the convolution sum as shown in the following figure.



Let us first make the following assumptions:

(i) x(n) is a sequence of length P defined for $0 \le n \le P-1$ and

(ii) h(n) is a sequence of length M defined for $0 \le n \le M-1$.

The convolution of x(n) and h(n) called the linear convolution is computed through DFT (FFT) as follows:|

Step(1): Choose $N \ge P+M-1$ (such that $N=2^r$ where r is a least positive integer). **Step(2):** Form the sequence $x^1(n)$ by padding N–P zeros to x(n).

$$x^1(n)=\begin{cases} x(n) & 0 \le n \le P-1 \\ 0 & P \le n \le N-1 \end{cases}$$

Step(3): Form the sequence $h^1(n)$ by padding N–M zeros to x(n).

$$h^1(n)=\begin{cases} h(n) & 0 \le n \le M-1 \\ 0 & M \le n \le N-1 \end{cases}$$

Step(4): Compute the N-point DFTs (FFTs), $X^1(k)$ and $H^1(k)$, of $x^1(n)$ and $h^1(n)$ i.e.,

$$X^1(k)=\sum_{n=0}^{N-1} x^1(n)W_N^{nk}, \quad k=0,1,2,\ldots,N-1$$

$$H^1(k)=\sum_{n=0}^{N-1} h^1(n)W_N^{nk}, \quad k=0,1,2,\ldots,N-1$$

where $W_N=\exp[-j(2\pi/N)]$.

Step(5): Compute the required output $y(n)$ for $0 \le n \le P+M-2$ by computing IDFT of the product, $X^1(k)H^1(k)$ and retaining the first $P+M-1$ values of the result.

$$y(n) = \begin{cases} y^1(n) & 0 \le n \le P+M-2 \\ & \end{cases}$$

Overlap-add method:

Let us first make the following assumptions:

(i) $x(n)$ is a long sequence of length $P \gg M$ defined for $0 \le n \le P-1$ and

(ii) $h(n)$ is a short sequence of length M defined for $0 \le n \le M-1$

Step(1): Choose a convenient, positive integer $L \ge 1$.

Step(2): Segment the long sequence $x(n)$ into r* sequences, each of length L. Let the segmented sequences be $x_0(n), x_1(n), x_2(n), \ldots, x_{r-1}(n)$ where, in general

$$x_k(n) = \begin{cases} x(n+kL),\ n=0,1,2,\ldots,L-1 \\ 0,\ \text{otherwise} \end{cases}$$

for $k=0,1,2,\ldots,r-1$.

*r is the smallest positive integer chosen such that $rL \ge P$.

Step(3): Choose $N \ge L+M-1$ (such that $N=2^r$ where r is a least positive integer). **Step(4):** Compute the N-point circular convolution

$$y_k(n) = x_k(n) \, \textcircled{N} \, h(n),\ 0 \le n \le N-1$$

using DFT (FFT).

Step(5): Form the sequences $y_k^1(n)$ by shifting the sequences $y_k(n)$ to the right by kL units for k=0,1,2,....,r−1 i.e.,

$$y_k^1(n) = \begin{cases} y_k(n-kL), & kL \le n \le (k+1)L+M-2 \\ 0, & \text{otherwise} \end{cases}$$

Here, the sequence $y_k^1(n)$ is nonzero for $kL \le n \le (k+1)L+M-2$, $y_{k-1}^1(n)$ is nonzero for $(k-1)L \le n \le kL+M-2$ and $y_{k+1}^1(n)$ is nonzero for $(k+1)L \le n \le (k+2)L+M-2$. This implies that the first (M−1) points of $y_k^1(n)$ overlap the last (M−1) points of $y_{k-1}^1(n)$ and the last (M−1) points of $y_k^1(n)$ overlap the first (M−1) points of $y_{k+1}^1(n)$ for all k as shown below:

Step(6): Compute the required output y(n) for $0 \le n \le P+M-2$ as follows:

$$y(n)=\sum_{k=0}^{r-1} y_k^1(n), \quad 0 \le n \le P+M-2$$

Overlap-save method:

Let us first make the following assumptions:

(i) x(n) is a long sequence of length P>>M defined for $0 \le n \le P-1$ and

(ii) h(n) is a short sequence of length M defined for $0 \le n \le M-1$

Step(1): Choose a convenient, positive integer $L \ge M$.

Step(2): Segment the long sequence x(n) into r sequences, each of length L. Let the segmented sequences be $x_0(n)$, $x_1(n)$, ..., $x_{r-1}(n)$, where, in general,

$$x_k(n)=\begin{cases} x[n+kL-(k+1)(M-1)], & n=0, 1, \dots, L-1 \\ 0, & \text{otherwise} \end{cases}$$

for $k=0, 1, \dots, r-1$.

$x_0(n)$ is chosen such that the first $(M-1)$ points are zeros and the remaining $(L-M+1)$ points are the first $(L-M+1)$ points of $x(n)$.

$x_1(n)$ is chosen such that the first $(M-1)$ points are the last $(M-1)$ points of $x_0(n)$ and the remaining $(L-M+1)$ points are the second $(L-M+1)$ points of $x(n)$.

$x_2(n)$ is chosen such that the first $(M-1)$ points are the last $(M-1)$ points of $x_1(n)$ and the remaining $(L-M+1)$ points are the third $(L-M+1)$ points of $x(n)$ and so on.

In general, $x_k(n)$ is chosen such that the first $(M-1)$ points overlap the last $(M-1)$ points of $x_{k-1}(n)$ and the last $(M-1)$ points overlap the first $(M-1)$ points of $x_{k+1}(n)$. **Step(3):** Compute the L-point circular convolution

$$y_k(n)=x_k(n) \, ⓛ \, h(n), \quad 0 \le n \le L-1$$

using DFT (FFT).

Step(4): Form the sequences $y_k^1(n)$ by discarding the first $(M-1)$ points of $y_k(n)$ for $k=0, 1, \dots, r-1$.

Step(5): Compute the required output $y(n)$ for $0 \le n \le P+M-2$ by appending the sequences $y_k^1(n)$ in order as follows:

$$y(n)=\{y_0^1(n), y_1^1(n), \dots, y_{r-1}^1(n)\}, \quad 0 \le n \le P+M-2$$

CORRELATION THROUGH FFT

The correlation of two finite length sequences, x(n) and y(n), each of length, N is the sequence, $r_{xx}(k)$ given by

$$r_{xy}(m)=\sum_{n=m}^{N-1} x(n)\ y(n-m) \quad \text{for } m\geq 0$$

$$r_{xy}(m)=\sum_{n=0}^{N-|m|-1} x(n)\ y(n-m) \quad \text{for } m<0$$

where the index, m is called the ***lag***. The correlation can be shown to be

$$r_{xy}(k)=x(k)*y(-k)$$

where y(–n) is the folded version of y(n). Hence the correlation can be computes using DFT (FFT) as follows.

The correlation of a sequence, x(n) to itself i.e., the correlation for the case when x(n)=y(n), is called the autocorrelation given by

$$r_{xx}(m)=\sum_{n=m}^{N-1} x(n)\ x(n-m) \quad \text{for } m\geq 0$$

$$r_{xx}(m)=\sum_{n=0}^{N-|m|-1} x(n)\ x(n-m) \quad \text{for } m<0$$

and

$$r_{xx}(k)=x(k)*x(-k)$$

Given x(n) and y(n), each of length, N

Step(1): Compute (2N–1)-point DFT (FFT), X(k) of x(n).
Step(2): Compute (2N–1)-point DFT (FFT), $Y_f(k)$ of $y_f(n)$ where $y_f(n)$ is the folded version of y(n).
Step(3): Compute the product of X(k) and $Y_f(k)$ and take the inverse DFT (FFT) of the result.