Filter Banks -I

Time-Frequency Representations

Dr. Yogananda Isukapalli

Time-Frequency Representation

1. Introduction
   a) Transform theory
   b) Time-frequency representation
2. Short-Time Fourier Transform
3. Evaluating STFT
4. Interpretation using bandpass filters
   a) Traditional Fourier transform
   b) STFT
5. Inversion of STFT
6. Filter bank interpretation of inverse STFT
7. Generalization of STFT
8. Passing from STFTs to Wavelets

1. Introduction

a) Transform theory

❖ It is well known that a discrete-time signal $x(n)$ can be represented as linear combination of a set of discrete-time (synthesis) vectors $b_i(n)$

$$x(n) = \sum_i c_i b_i(n) \tag{1}$$

This is known as the ***synthesis equation*.** The c_i are the weights of the vectors, which are generated by computing the inner product of the signal and a set of analysis vectors $a_i(n)$

$$c_i = \langle x(n), a_i(n) \rangle \tag{2}$$

These weights are elements of a vector ***C***, which is called the transform of a signal.

$$C(i) = \sum_n x(n) a_i(n) \tag{3}$$

This equation is known as the ***analysis equation***

❖ Transforms are important for many reasons:

1. New information is provided about the signal

   Example: The Fourier transform provides information on the frequency content of a signal

2. Processing a signal in the transform domain can be simpler or implemented more efficiently.

   Example: Convolution becomes multiplication in the Fourier domain, which is easier to perform.

3. The signal may be represented by fewer numbers in the transform domain, i.e., some $c_i = 0$.

   Example: The Discrete Cosine Transform (DCT) is widely used in speech and image compression because it does a good job of decorrelationg these types of signals (makes many $c_i = 0$).

❖ Combining the analysis and synthesis equations yields:

$$\hat{x}(n) = \sum_i \langle x(n),\, a_i(n)\rangle\, b_i(n) \tag{4}$$

$$\hat{X} = BAX \tag{5}$$

Where **A** and **B** are the analysis and synthesis matrices respectively, and $\hat{x}$ is the reconstructed signal. To get $\hat{x} = x$, **B =A^{-1}**.

❖ We may break this class of invertible transforms into three groups:

1. Orthogonal/Unitary

$$B = A^H \qquad \textbf{where } \mathbf{A}^H = A^{-1}$$
$$b_i(n) = a_i{}^*(n) \tag{6}$$

Example - DFT:

$$b_i{}^*(n) = \left(e^{\frac{j2\pi k}{N}n}\right)^* = e^{\frac{-j2\pi k}{N}n} = a_i(n) \tag{7}$$

2. Biorthogonal

$\mathbf{B} = \mathbf{A}^{-1}$ (8)

$$\langle a_i(n), b_i(n) \rangle = \delta_{j=i} \tag{9}$$

3. Non-orthogonal transform

$\mathbf{B} = \mathbf{A}^{+}$, where $\mathbf{A}^{+}$ is the pseudoinverse. (10)

Example - Lapped Orthogonal Transform (LOT)

Frames

❖ A frame is a set of functions, call them Ψ, such that a linear expansion can be performed, i.e., the set of functions spans $L^2(R)$, which is the space of finite energy signals. Hence, a signal can be represented as

$$x = A^{-1} \sum_j \langle x, \Psi_j \rangle \tilde{\Psi}_j \tag{11}$$

Mathematically, this can be expressed as

$$A\|x\|^2 \leq \sum_{j \in Z} \left|\langle x, \Psi_j \rangle\right|^2 \leq B\|x\|^2 \qquad (12)$$

for some $A > 0$ and $B < \infty$. A and B are known as the ***frame bounds***, and can be numerically calculated given a particular frame

❖ If the number of vectors in the frame exceed the number of vectors needed to span the space, or the dimension of the vectors is greater than that of the space, then the expansion is redundant and we effectively have a nonorthogonal transform. The amount of redundancy can be measured by taking the ratio of the frame bounds $B/A > 1$. Thus, the higher the ratio, the greater the redundancy.

❖ If $A = B$, then the frame is called a "tight frame", This is equivalent to having the minimum number of vectors to span a particular space and having the dimension of the vectors equal to that of the subspace i.e., the *transformation matrix is square*.

- If $A = B = 1$, then we have a "normal" tight frame, or "energy-preserving" tight frame. A tight frame may add energy in the form of a non-unity A and B. Clearly, an orthonormal set preserves the energy of the signal (it is *isometric*). We may obtain a normal tight frame from a tight frame by dividing the frame bound equation by either A or B. This is equivalent to scaling each vector by its norm (length) so that they all have unit length. A tight frame behaves as an orthonormal basis, even if the basis functions are not linearly independent.

- In general, frames are not orthonormal, do not satisfy the Parseval Identity (does not preserve energy), and the expansion using frames is not unique.

Orthonormal Basis Functions

- A basis is a special case of a tight frame in which the vectors Ψ not only span the space, but are also *linearly independent*. If the vectors are, in addition, orthonormal to each, meaning

$$\langle \Psi_k, \Psi_l \rangle = \delta_{kl} \quad \text{or} \tag{13}$$

$$\langle \Psi_k, \Psi_l \rangle = \begin{cases} 1 & k = l \\ 0 & k \neq l \end{cases} \tag{14}$$

  then we may say that the set of vectors Ψ form an *orthonormal basis* of the space. An orthogonal (or biorthogonal) transform is a mapping from a space defined by one orthonormal basis to another space defined by a new orthonormal basis. In addition, this transformation is unique.

- When we are dealing with time signals, we use the standard Euclidean basis vectors, which in matrix form is the Identity matrix **I**. Hence, when we apply an orthogonal transform like the (Discrete) Fourier Transform, we are projecting the signal from the Euclidean basis vectors to a new orthonormal basis, which are the vectors of the transform matrix.

❒ In summary, we view the classification of the frames as in the following:

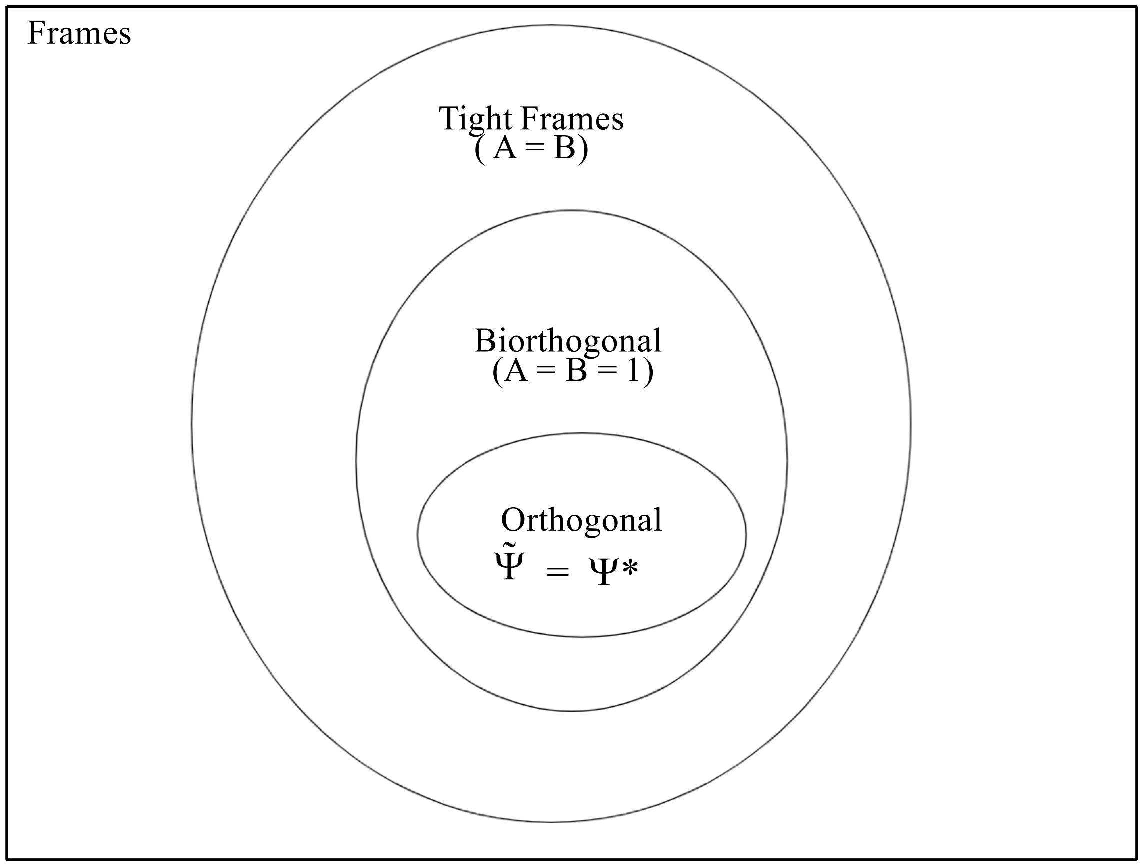

fig 1. Classification of frames

Here, a biorthogonal basis is one in which the synthesis basis is different, yet orthogonal to, the analysis basis (as in a biorthogonal transform).

b) Time-Frequency Representations

- Transforms are limited in the sense that only one view of the signal is allowed at a time. For example with the Fourier transform, we may either view how the signal varies in time or how it varies in frequency, but not both. This is shown pictorially below in *figure 2*. Transforms do not allow for a combination of the two domains. This is acceptable for stationary signals, i.e., signals whose components do not change in time. However, transforms are quite unacceptable for nonstationary signals, since we often wish to know when different frequency components occur in time

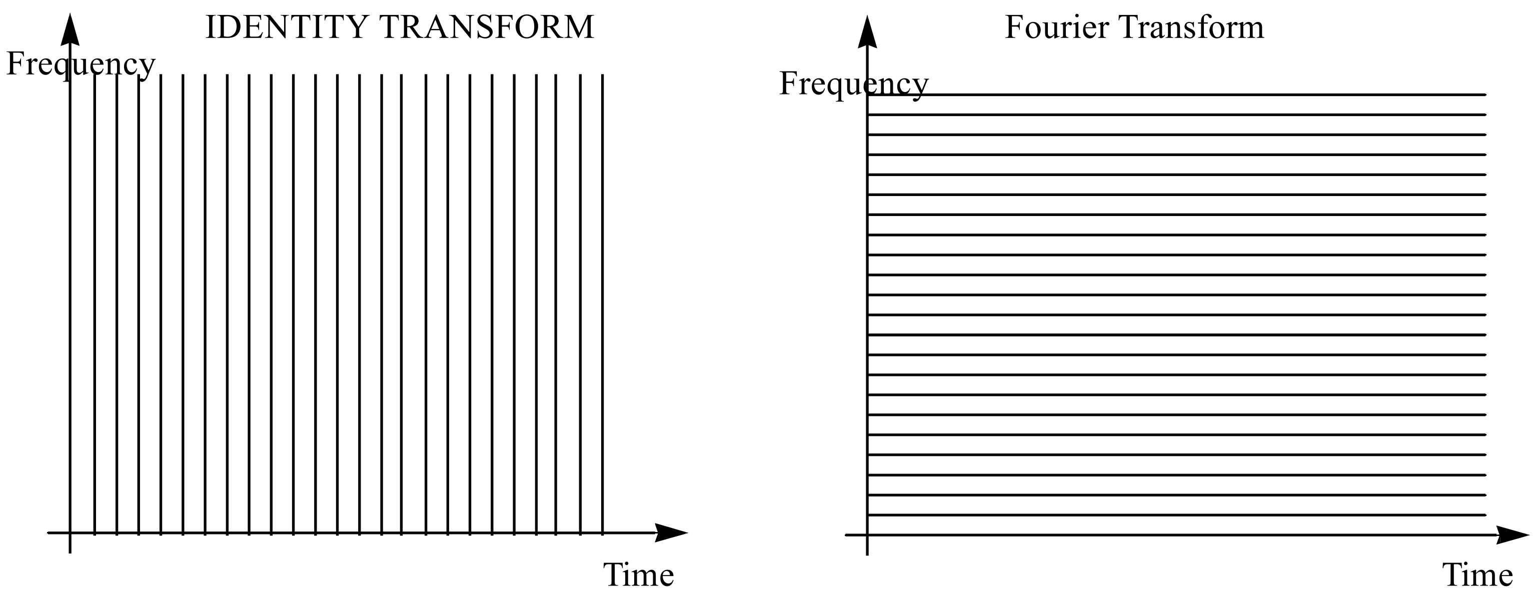

fig 2- Tilings of the time-frequency plane for the Identity & Fourier Transforms

- Time Frequency Representations (TFRs) provide this combination of domains. TFRs map a one-dimensional (continuous) signal in time (or frequency) into a two-dimensional (continuous) function in time and frequency.

$$x(t) \leftrightarrow T_x(t,f)$$

- This domain is often called the time-frequency (TF) plane.

- To demonstrate the importance of TFRs, consider the two signals in *figure 3* and their spectra. Due to the fact that time information is lost, the spectra appear identical. Ideally, we would like the TFR to display data as in *figure 4*

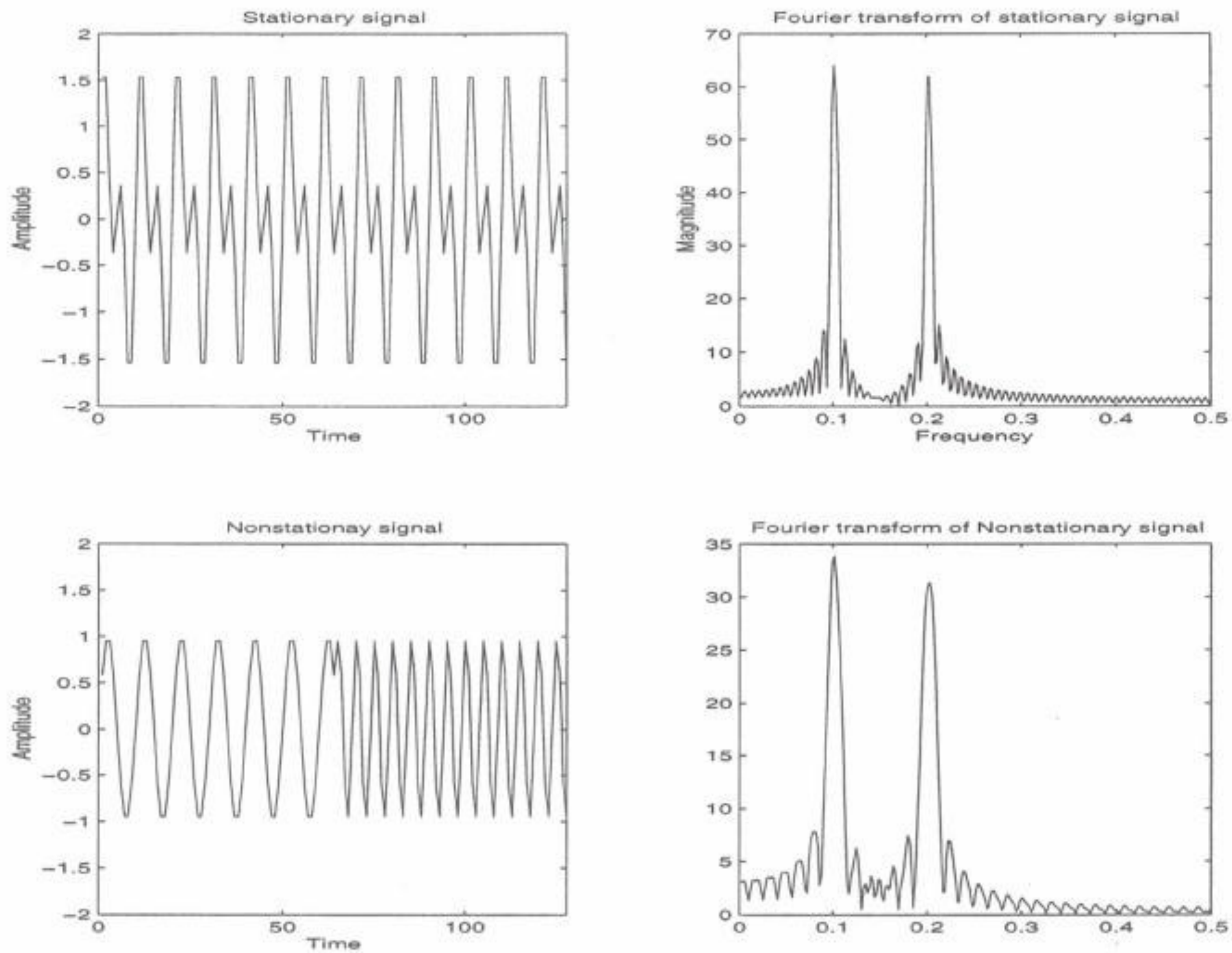

fig 3. a) Stationary signal of 2 frequencies 0.1 and 0.2 which exist for all time b) FT of a c) Nonstationalry signal with frequency 0.1 existing for half the time and frequency 0 .2 existing for other half. d) FT of c.

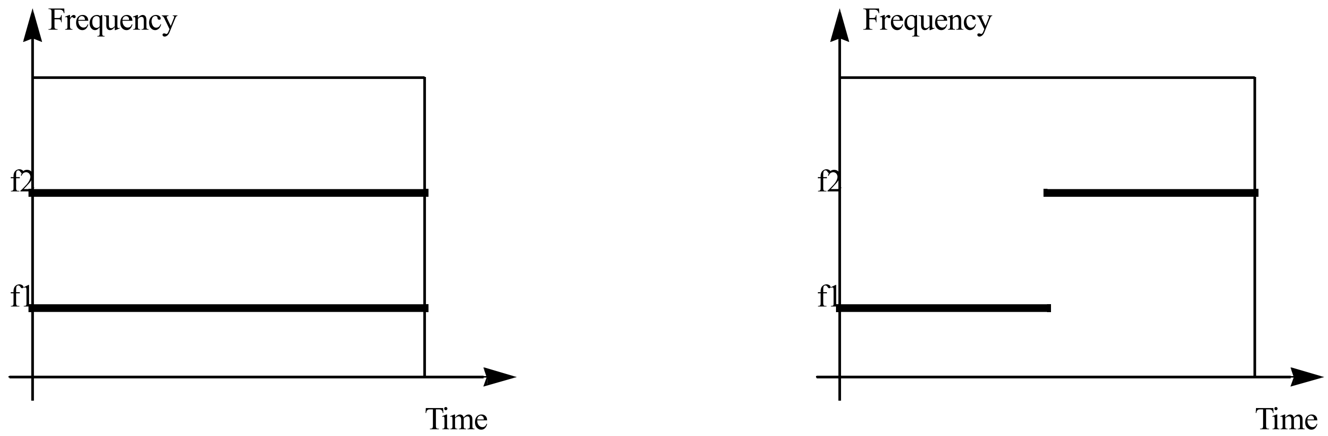

fig 4. Ideal TFRs

- In practical applications, we must deal with discrete signals *x(n)*. Hence, we must also discretize the TF plane. This can be done, in theory, by sampling the continuous TFR. In practice, we only compute the TFR at these sample locations to produce a discrete TFR. This effectively partitions the TF plane, which can easily be seen by connecting adjacent sampling points as in *figure 5*

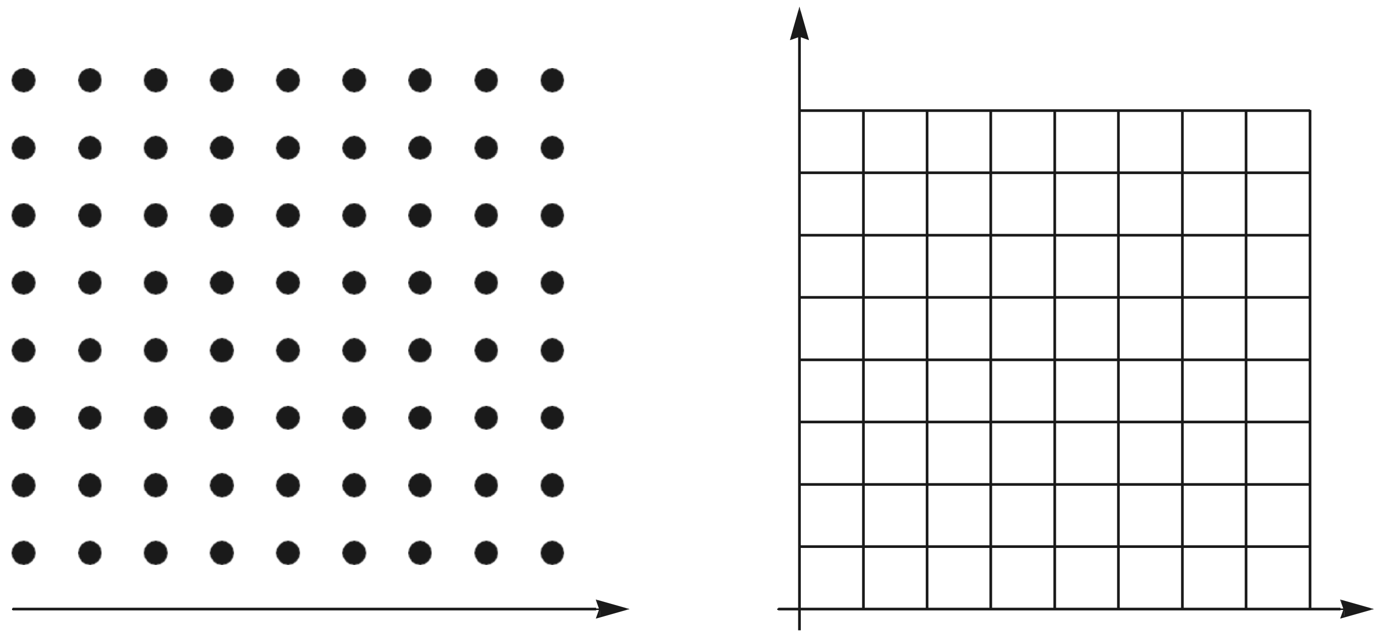

fig 5. Uniform sampling and tiling of the TF plane

- These partitions are called tilings of the TF plane. Various TFRs partition the TF plane in different ways. The tilings in *figure 5* are two extreme examples of a TF tiling.

2. Short-Time Fourier Transform (STFT)

❖ One of the most common TFRs is the short-time, or windowed, Fourier transform. It is defined as:

$$STFT_x(\tau, f) = \int_t x(t) w^*(t-\tau) e^{-j2\pi ft} dt \tag{15}$$

where w(t) is some window function. This is equivalent to computing the Fourier transform of a windowed segment of data.

❖ The window *w(n)* can be any standard window function, such as Boxcar(rectangular), Hamming, or Bartlett window. The Gabor expansion uses the Gaussian function as window. The window plays a critical role in STFT analysis, and different windows will lead to different TFRs.

❖ The discretized version of the STFT can be defined as:

$$STFT(m, k) = \sum_n [x(n) w(n-m)] e^{-j2\pi nk/N} \tag{16}$$

where N is the window length. Note that this is equivalent to computing the Discrete Fourier Transform (DFT) of a length N block of the signal *x(n)*. The discrete STFT uniformly samples the continuous STFT, as in *figure 5*.

Time-Frequency resolution trade-off.

❖ Without loss of generality, let's use a rectangular window on the data. According to the DFT, the number of frequency bins in the transformation is equal to the number of data points in the block, which in turn is equal to the length of the time window.

❒ Let's consider two cases:

1. Two sinusoids that are closely spaced in frequency:

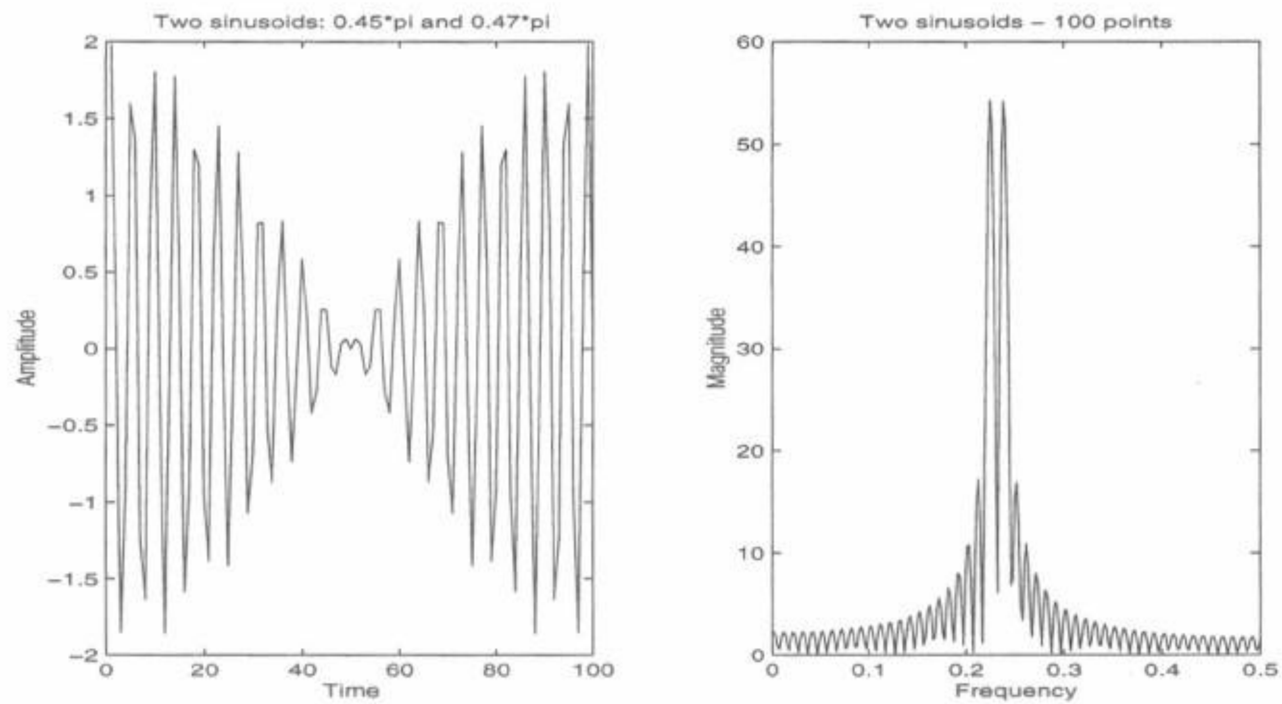

fig 6. a) Two sinusids close in frequency b) Fourier transform of a.

100pts.

❖ Looking at the figures below, we see that we need at least N samples to separate sinusoids that are π/N radians apart:

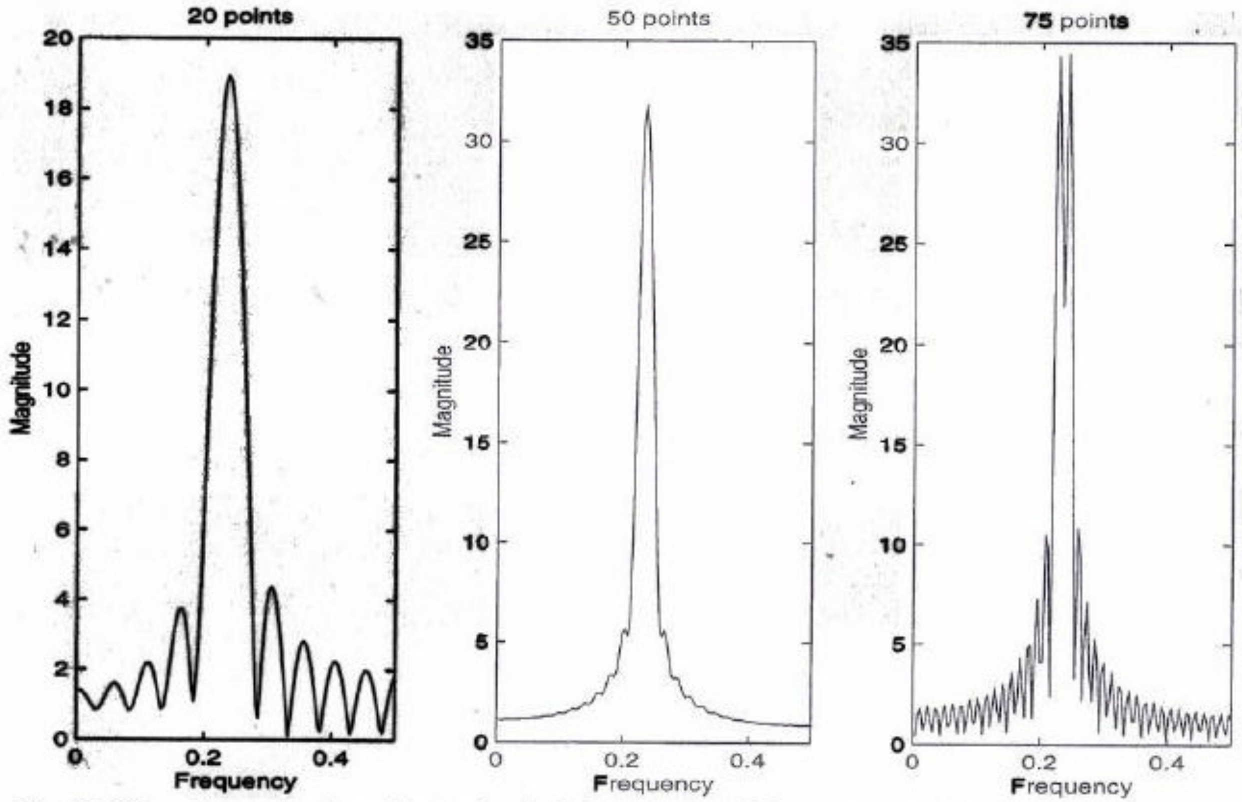

fig 7. Varying window length a) 20 point b) 50 point c) 75 point window

It is obvious that if two sinusoids are closely spaced together, we need a large time window. Hence, frequency resolution is inversely proportional to the length of the time window.

2. Two impulse functions that are closely separated in time

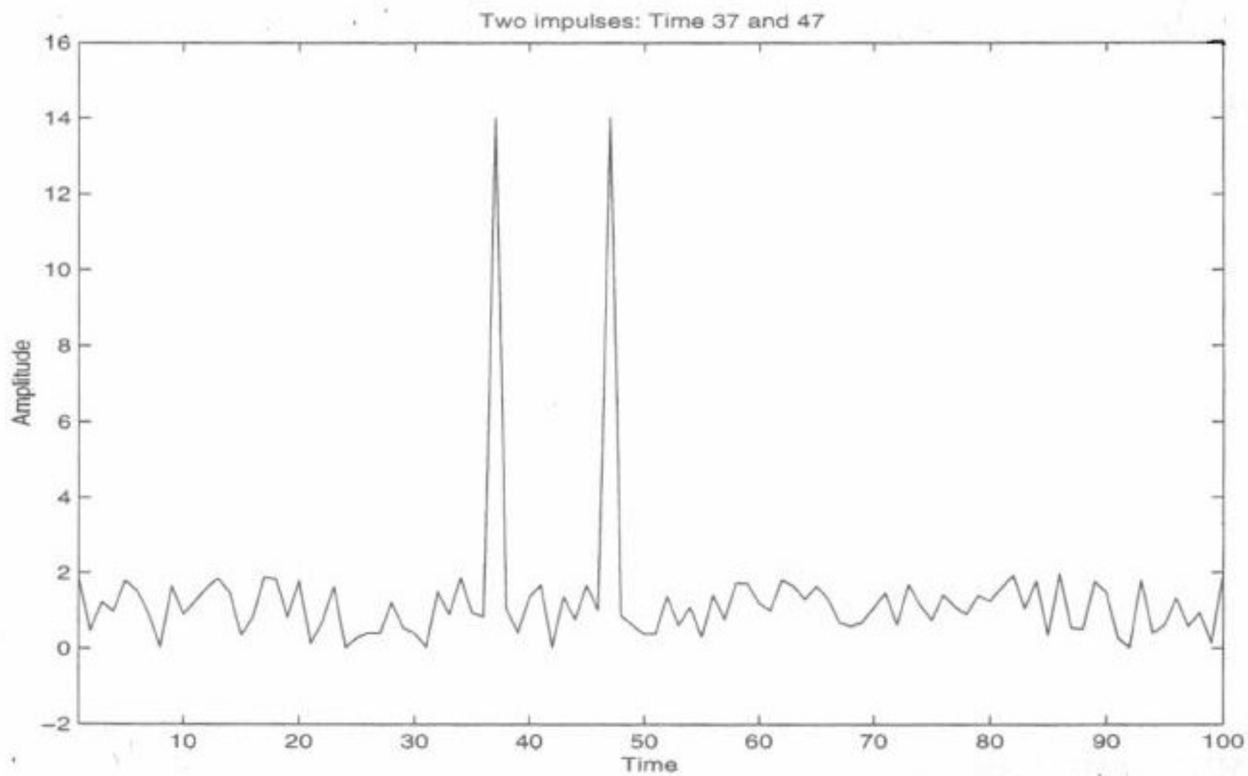

fig 8. Two impulses close in time

❖ From the figures below, we see that with a large time window, the two impulses may be combined into one if the DFT is performed on both signals (since the energy of both is spread over all frequencies).

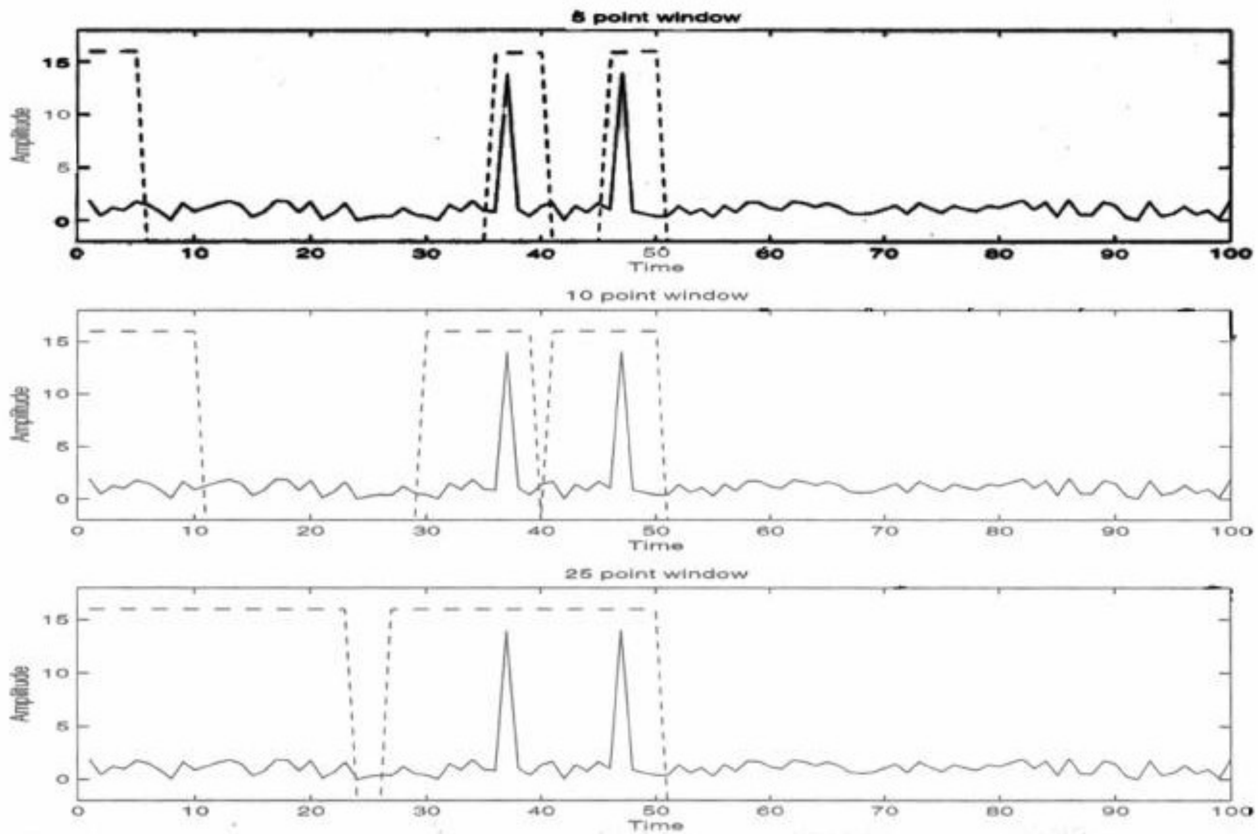

fig 9. Varying window lengths for signal of fig8.

❒ Hence, an uncertainty exists in that we cannot simultaneously know the instantaneous time occurrence and frequency of a signal component. This uncertainty is parallel to the Heisenberg's Uncertainty Principle in quantum mechanics with position and momentum of an electron.

❒ This uncertainty is due to the inherent relationship between the time and frequency domains. A signal that is shorter in time is longer in frequency, and vice verse. A clear example is the FT of a delta, which is unity. In this case, we have perfect time resolution and infinite frequency resolution. Another example is the sinuosoid, which is infinite in time and has perfect frequency resolution. This important theorem is known as the *Balian-Low Theorem,* which states that there is a bound to the amount of resolution that can exist, which can be quantitatively expressed as:

$$(\Delta t)(\Delta f) \geq \frac{1}{4\pi} \tag{17}$$

where Δt and Δf are the time and frequency resolutions, and are defined as

$$\Delta t^2 = \frac{\int t^2 |x(t)|^2 dt}{\int |x(t)|^2 dt} \quad \text{and} \quad \Delta f^2 = \frac{\int f^2 |X(f)|^2 df}{\int |X(f)|^2 df} \tag{18}$$

❒ There are three ways that we can view the calculation of a discrete-time, discrete-frequency TFR:

1. Windowing the signal

2. Windowing the basis functions

3. Applying a bank of frequency-selective filters

All of these views are equivalent and each adds to the understanding of the generation of TFRs.

1. Windowing the signal

❒ This is perhaps the most obvious approach, and it has been discussed already. We take a block of data of length N from a signal of length L and apply a transform. Since the same algorithm is used to perform the transform for each block, we can say that the basis vectors of the transform do not change. This is desirable for implementation purposes:

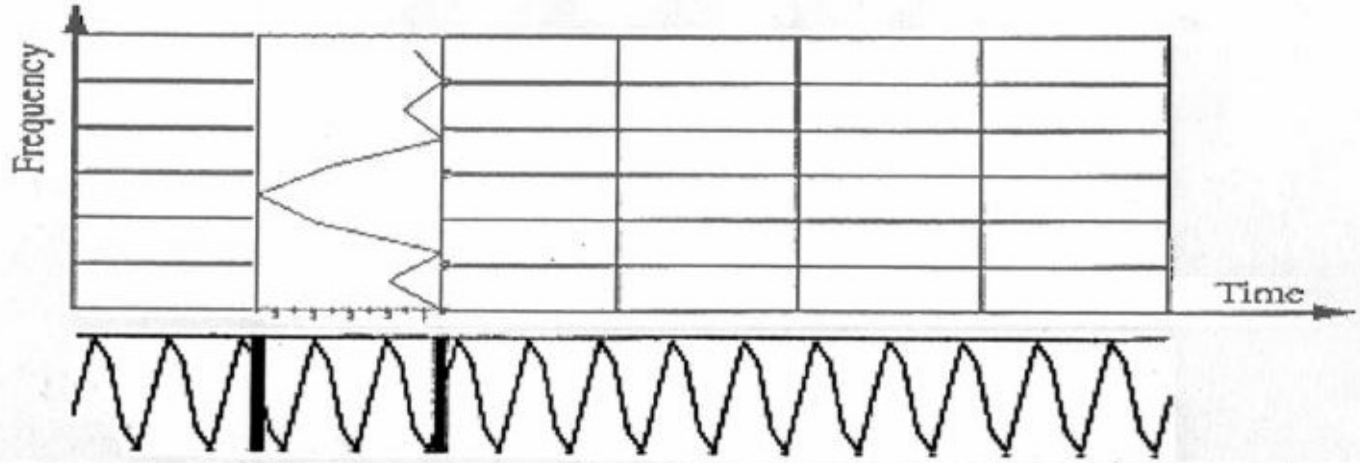

fig 10. Implementing a TFR using signal windowing method

2. Windowing the basis functions

- This second approach provides a much more interesting view of the process. In the first approach, we had a total of N basis vectors which were infinite in length. To compute the TFR, we computed the inner product of the windowed segment of data with the infinite length basis vectors. This amounted into taking the DFT of a composite signal $y(n) = x(n)w(n)$.

Similarly we can create a composite basis vector

$$\Psi(n) = w(n)e^{\frac{-j2\pi nk}{N}}$$

which results in

$$STFT_x(m, k) = \sum_n x(n)\Psi(n - m) \tag{19}$$

❐ By windowing the basis vectors, we are creating a new basis for the entire TF plane. The number of basis vectors increases from N to

$$N \times (\text{number of blocks}) = N \times (L / N) = L$$

Hence, if we have a length $L = 64$ signal, then by windowing the basis functions we create a set of 64 basis functions for the TF plane. This value is also the dimension of the space that we are trying to characterize. Remember that if we have more than these 64 vectors, we have a redundant expansion and the orthonormal basis becomes a frame $(B/A > 1)$. If we have less than 64 vectors, the system is undetermined and the transform is noninvertible.

❒ Let's apply the Uncertainty principle and *Balian-Low Theorem* to this idea of windowing the basis functions. To obtain perfect resolution in frequency, the basis functions must be of infinite length. This is due to the Balian-Low Theorem which states that a long signal in one domain(time) is short in another(frequency). Once we window these infinite length functions, we are effectively truncating the signal. This truncation will cause a spread in frequency (due to convolution with a sinc function in the case of a rectangular window). The amount of spreading is directly proportional to the amount of truncation of the time waveform: the more the truncation the more the spreading. This is another way to view the Uncertainty principle in action.

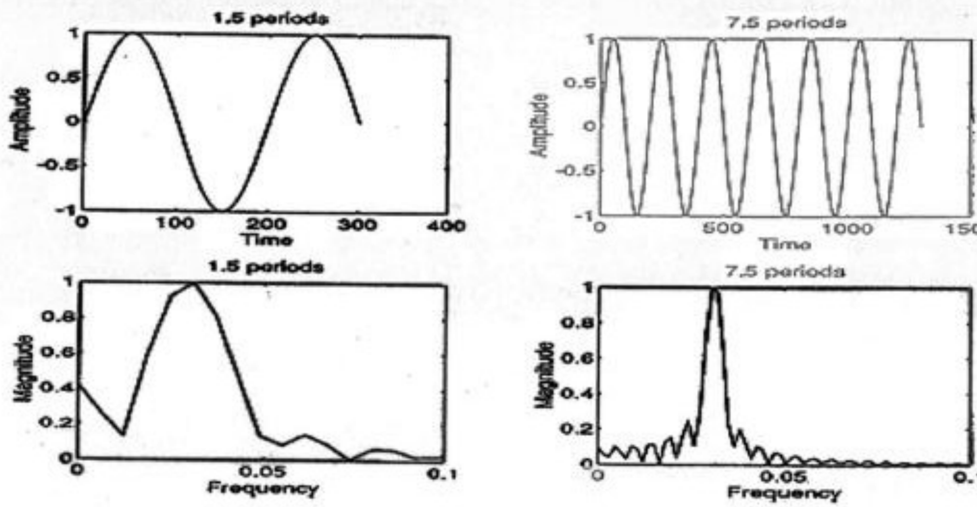

fig 11. Truncating an infinitely long signal

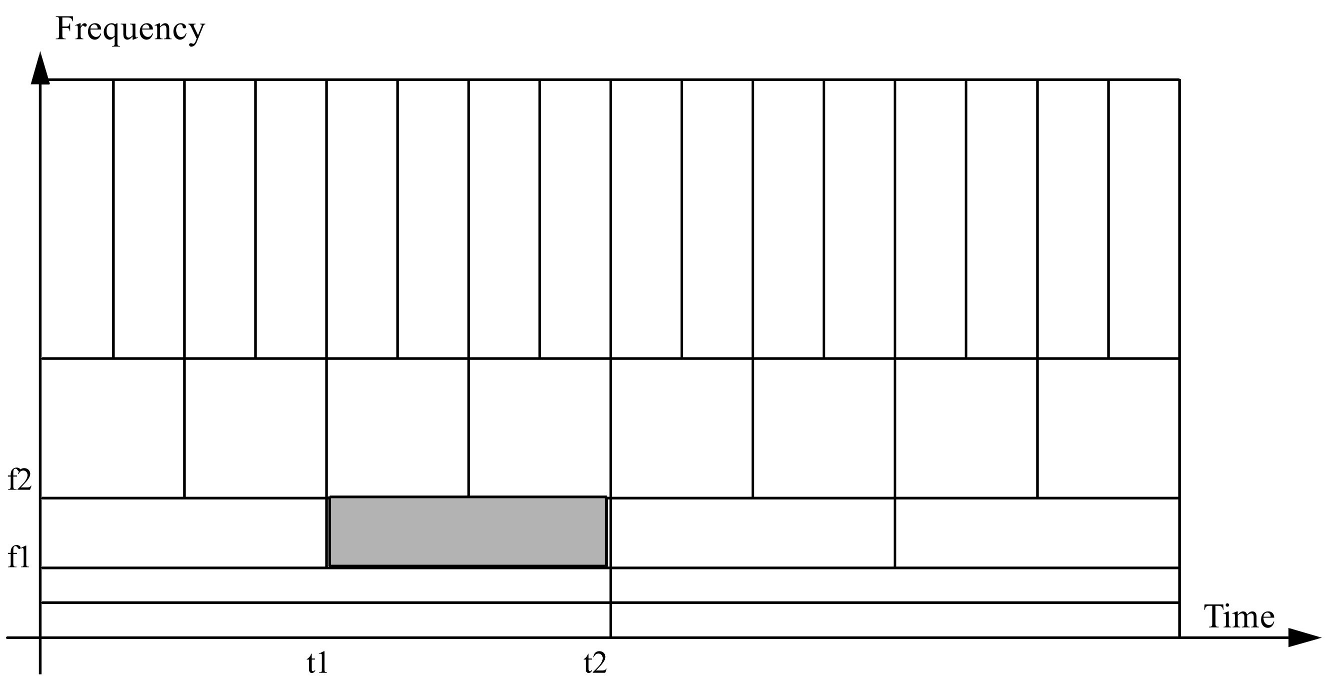

fig 12. Arbitrary TF tiling for a length L = 32 signal

- Note that we also have L tiles in the TF plane. Hence, each tile has a basis vector associated with it. For a particular tile, its basis vector will occupy the same time and frequency band as the tile. For example, the basis function corresponding to the shaded tile in figure.. above will occur from time t_1 to time t_2, and will occupy the frequency from f_1 to f_2.

❒ Hence, the second way we may compute a TFR is to find a basis for the TF plane of a dimension equal to the signal length (dimension), and then compute the inner product of the entire signal and each basis function. This is equivalent to projecting the signal onto a new L dimensional space.

Note: Each tile has a minimum area of $1/4\pi$ due to the Balian-Low Theorem. This area can be calculated by multiplying $\Delta t = t_2 - t_1$ and $\Delta f = f_2 - f_1$.

3. Applying a bank of frequency selective filters

The Fourier transform of the STFT equation yields:

$$\begin{aligned} STFT_x(n, l) &= \sum_k X(k) W(k-l) e^{\frac{-j2\pi nk}{N}} \\ &= \sum_k X(k) \Psi(k-l) \end{aligned} \tag{20}$$

❒ The quantity $W(k)$ is the Fourier transform of the time window $w(n)$. If the time window is a bandpass filter, i.e., the frequency response is limited to a finite, non-piecewise band, we may view $W(k)$ as a "frequency window". Hence, we may view TFR generation as applying a frequency window, or filter, to the data

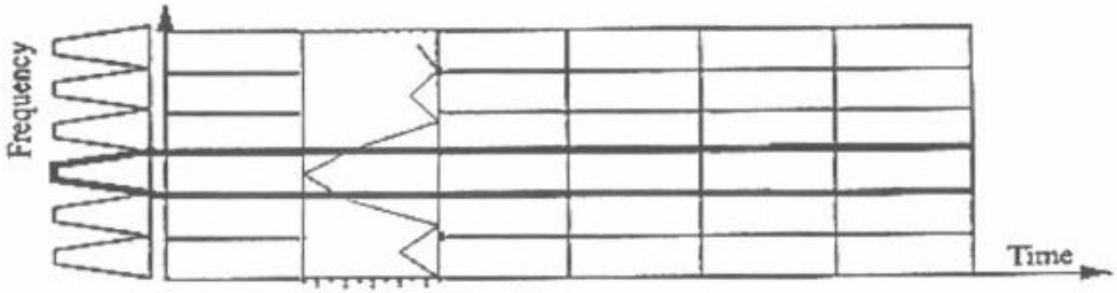

fig 13. Bank of frequency selective fiters

- To compute the TFR, we convolve the data (in time) with the impulse response of each filter. Note that we get an output from each filter every sample time. If there are M filters, we will have M sets of length N data, for a total of MN data points. However, we said earlier that we must have same number of points in the transformed signal as in the original signal. Hence, we have redundancy in the transform (frame).

- The redundancy is due to the convolution operation. Each sample is included a number of times equal to the length of the filter. Hence, to eliminate redundancy, we must:
  1. Convolve the filter with blocks of data only.
  2. After each convolution, shift the filter sequence by a number of samples greater than one sample.
  3. Decimate the output of each filter by some factor.

- Clearly, all the three methods described above will produce the same effect, and thus are equivalent. The last is easiest to implement and leads to the well-known filter bank structure shown in figure (Note $L < M$ will lead to a redundant expansion (frame)).

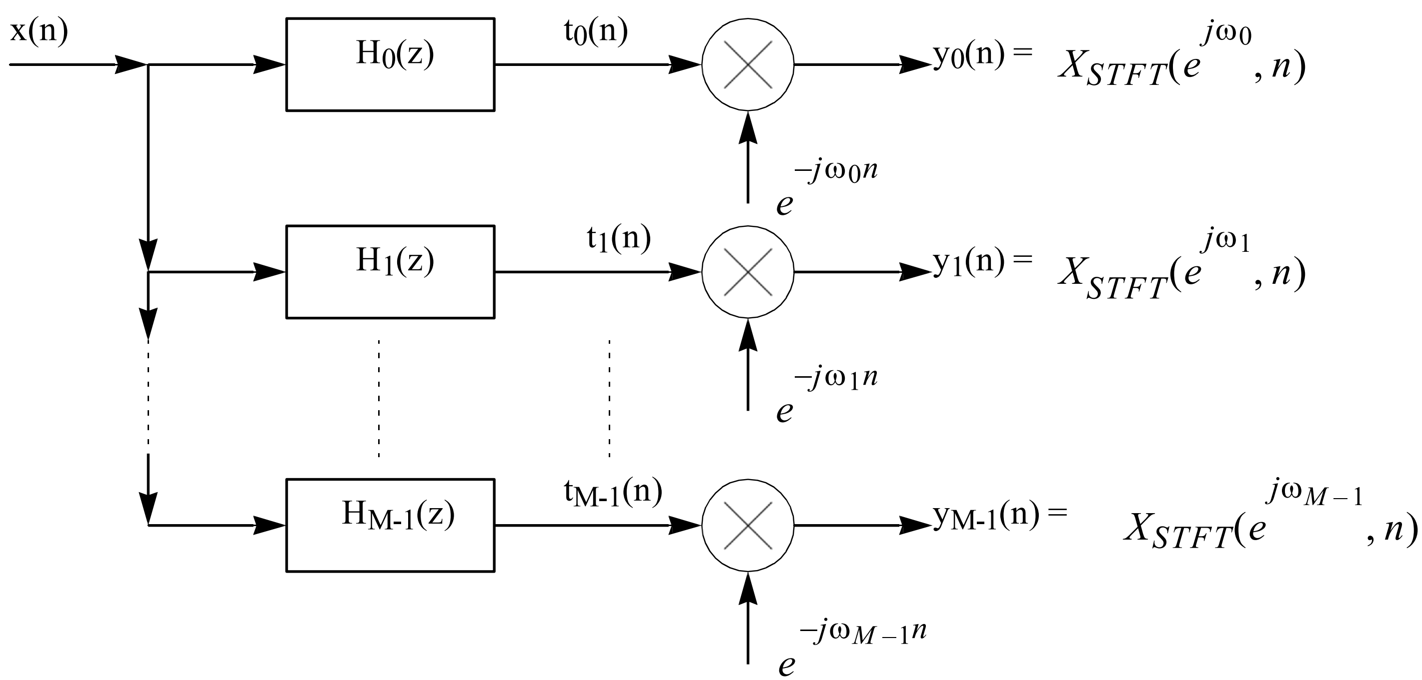

fig 14. Filter bank approach to TFR generation

- The filter bank used in *figure 14* can either be uniform or non-uniform. In a uniform filter bank, such as one used to implement the STFT, each filter has constant bandwidth:

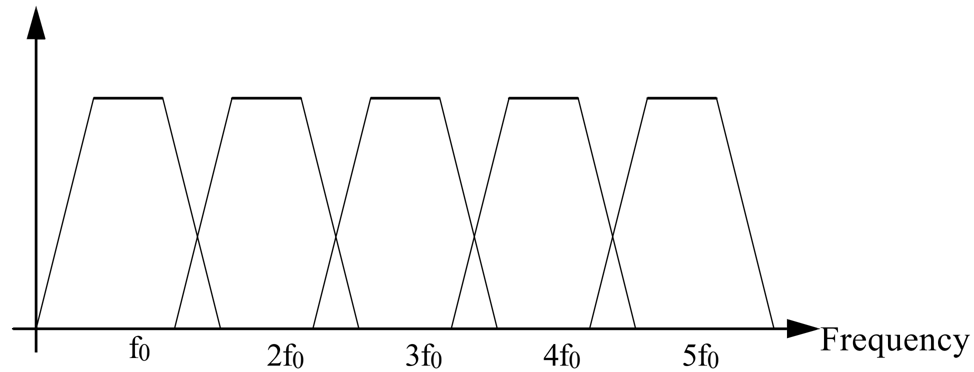

fig 15. Constant bandwidth

- One choice of a non-uniform filter bank be one that has constant relative bandwidth, meaning that the bandwidth of the filter relative to its centre frequency is constant

$$\frac{\Delta f}{f} = \text{constant}$$

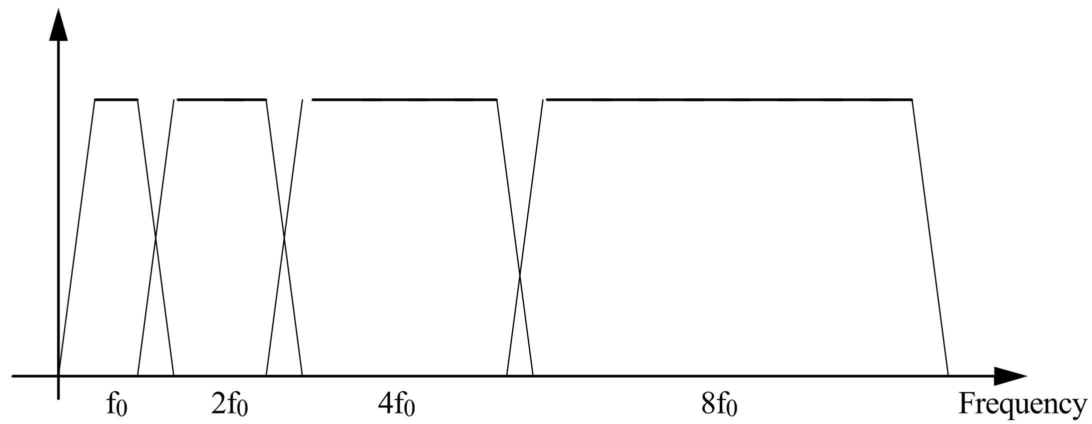

fig 16. Constant relative (constant - Q) bandwidth

- This is also called a constant-Q analysis (Q being the quality factor). Such filters can be said to be uniform on a logarithmic frequency scale.

3. The Short-Time Fourier Transform(STFT)

Steps in finding STFT of a signal

1. Multiply the signal *x(n)* with a window *w(n)*.
2. Compute the Fourier transform of the product *x(n)w(n)*
3. Shift the window *w(n)* in time
4. Go to step 1.

❒ This operation results in a separate Fourier transformation for each location m of the center of the window. In other words, we obtain a function $X_{STFT}(e^{j\omega}, m)$ of two variables *w* and *m* . The frequency variable *w* is continuous, and takes the usual range. The shift variable *m* is typically an integer multiple of some fixed integer *K*. For any fixed *m*, the window captures the features of the signal *x(n)* in the local region around *m*. The window therefore helps to localize the time domain data, before obtaining the frequency domain information.

❒ The short-time Fourier transform can be written mathematically as

$$X_{STFT}(e^{j\omega}, m) = \sum_{n=-\infty} x(n)w(n-m)e^{-j\omega n} \tag{21}$$

for $w(n) = 1$, this reduces the the traditional Fourier transform for any choice of m.

4. Interpretation using bandpass filters.

a) Traditional Fourier transform as a bank of filters

The evaluation of $X(e^{j\omega})$ at fixed frequency ω_0 can be represented as shown in *figure17*

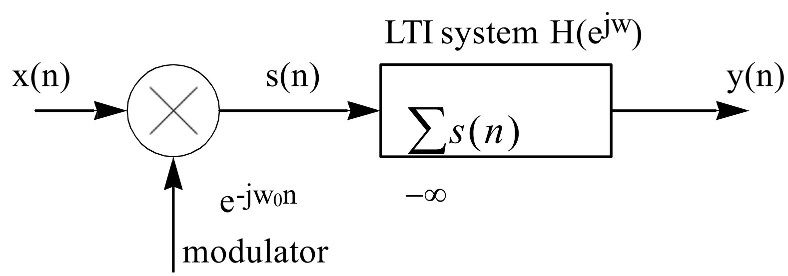

fig 17. Representation of Fourier transform in terms of linear systems

This can be considered as a cascade of two systems

1. *The modulator* $e^{-j\omega_0 n}$ - It shifts the Fourier transform towards the left by an amount ω_0 so that the zero frequency value of $S(e^{j\omega})$ is equal to $X(e^{j\omega})$.

2. *The LTI system* $H(e^{j\omega})$ - This has an impulse response *h(n) = 1* for all *n*. Its frequency response is given as

$$H(e^{j\omega}) = \sum_{n=-\infty} h(n)e^{-j\omega n} = 2\pi\delta_a(\omega) \qquad (22)$$

where $\delta_a(\,)$ is a Dirac delta function.

$H(e^{j\omega})$ can be considered as an ideal lowpass filter which passes only the zero-frequency signal. Therefore the output is given as

$$Y(e^{j\omega}) = 2\pi X(e^{j\omega_0})\delta_a(\omega) \qquad -\pi \le \omega \le \pi \qquad (23)$$

$$\text{i.e. } y(n) = X(e^{j\omega_0}) \qquad (24)$$

- Thus the process of evaluating $X(e^{j\omega})$ can be looked upon as a linear system, which takes the input x(n) and produces a constant output *y(n)* whose value is equal to $X(e^{j\omega})$ for all time *n*. Thus any sample of *y(n)* can be taken to be the value of $X(e^{j\omega})$
- The Fourier transform operator which evaluates $X(e^{j\omega})$ for all ω is, therefore, a bank of modulators followed by filters. This system has an infinite number of channels.

b) STFT as a bank of filters

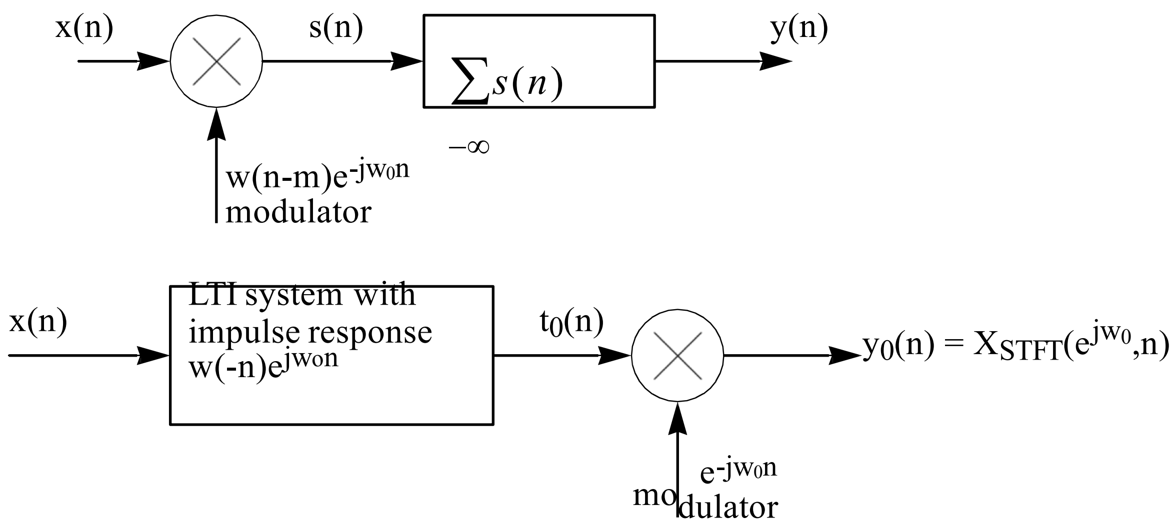

fig 18. a) The STFT represented in terms of linear system and b) a rearrangement of a

ω_0 and m are constants. So, *y(n)* is constant for all *n*. with $y(n) = X_{STFT}(e^{j\omega_0}, m)$

Equation (21) can be written as shown in equation (25)

$$X_{STFT}(e^{j\omega}, m) = e^{-j\omega m} \sum_{n=-\infty} x(n)w(n-m)e^{j\omega(m-n)} \qquad (5)$$

figure 18b. shows this interpretation

- This is a linear system with two parts. The first is an LTI filter with impulse response $w(-n)e^{j\omega_0 n}$. This is followed by the modulator $e^{-j\omega_0 n}$. The output $y_0(n)$ of this system is now a function of n. For any specific value of n, say $n = m$, this output represents the Fourier transform of $x(n)$, in the neighbourhood of m, because m represents the location of the window $w(k)$ in the time domain. For the special case where $w(k) = 1$ for all k, this output becomes a constant (equal to the traditional Fourier transform) for all n.

- The window $w(n)$ has a lowpass transform $W(e^{j\omega})$ and so does $w(-n)$.

  The modulated version represents a bandpass filter $W(e^{-j(\omega - \omega_0)})$
  The output sequence is therefore, the output of a bandpass filter, whose passband is centered around ω_0.

- The modulator $e^{-j\omega_0 n}$ re-centers the center frequency around zero frequency.

- For every frequency ω_0 the STFT performs the filtering operation to produce an output sequence $X_{STFT}(e^{j\omega_0}, m)$. So, the STFT can be looked upon as filter bank, with infinite number of filters (one per frequency). In practice, we are interested in computing the Fourier transform at a discrete set of frequencies

$0 \le \omega_0 < \omega_1 < \ldots < \omega_{M-1} < 2\pi$. In this case the STFT reduces to a filter bank with M bandpass filters $H_k(z)$ with responses $H_k(e^{j\omega}) = W(e^{-j(\omega - \omega_k)})$ and followed by modulators as shown in the figure 19.

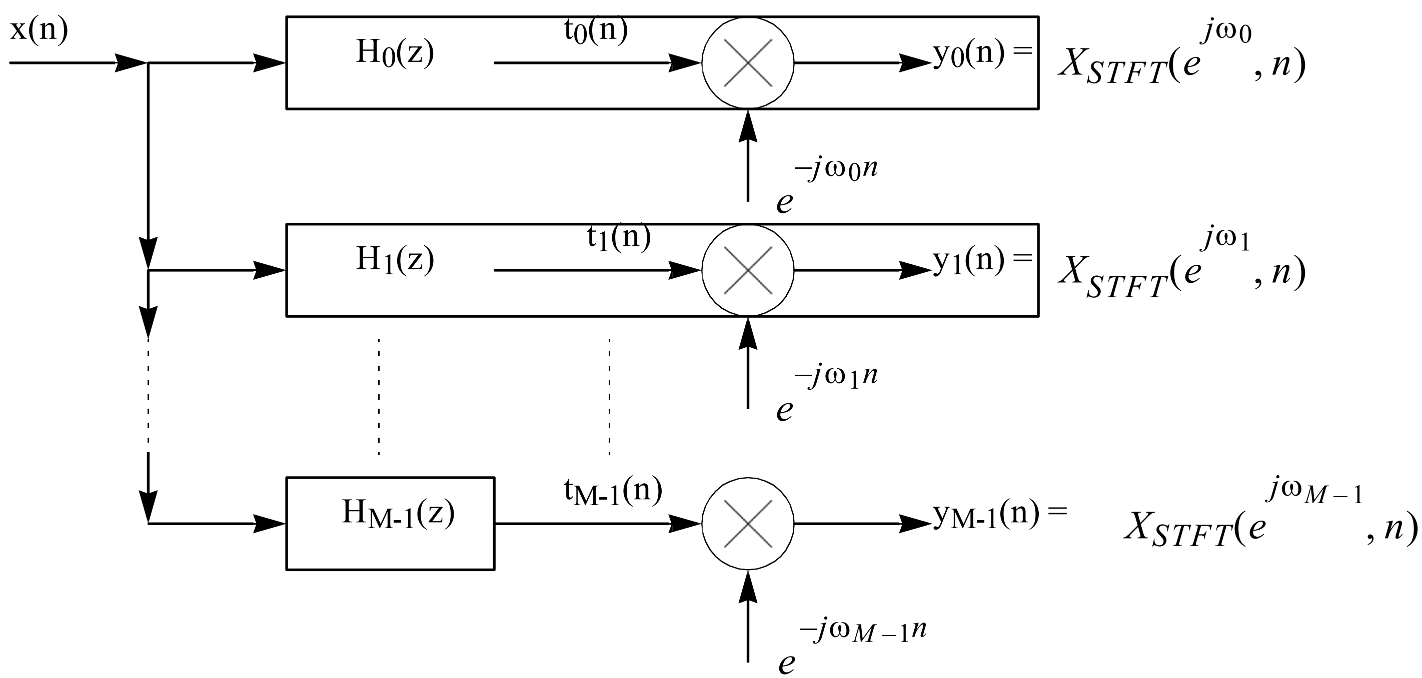

fig 19. The STFT operation viewed as a filter bank

- If the frequencies ω_k are uniformly spaced, then the above system becomes the uniform DFT bank. For this case the M filters are related as

$$H_k(z) = H_0(zW^k) \qquad 0 \le k \le M-1 \text{ where } W = e^{-j2\pi/M}$$

This implies that the uniform DFT bank is a device to compute the short-time Fourier transform at uniformly spaced frequencies.

Choice of w(n), and Time-frequency tradeoff

- The STFT is not uniquely defined unless the window *w(n)* is specified. The choice of *w(n)* governs the tradeoff between time localization and frequency resolution. The signal $y_0(m)$ represents the evolution of the Fourier transform of x(n), evaluated around frequency ω_0 . Thus $y_0(m)$ represents the local information, around time m and around frequency ω_0. As $W(e^{j\omega})$ becomes narrower, the bandpass filters get narrower, so the information in the frequency domain is more localized. However as $W(e^{j\omega})$ becomes narrower, the window w(n) gets wider (uncertainity principle) so that the localization of information in the time domain is compromised. Similarly by making the window w(n) narrower, the time domain information is more localized, while the frequency domain information is compromised.

Time-frequency representation and decimation

Instead of moving the window one sample at a time, if we move it by M samples then it is equivalent to decimating the outputs $y_0(n)$ by M. Figure 20 shows the decimated STFT system with the modulators moved past the decimators.

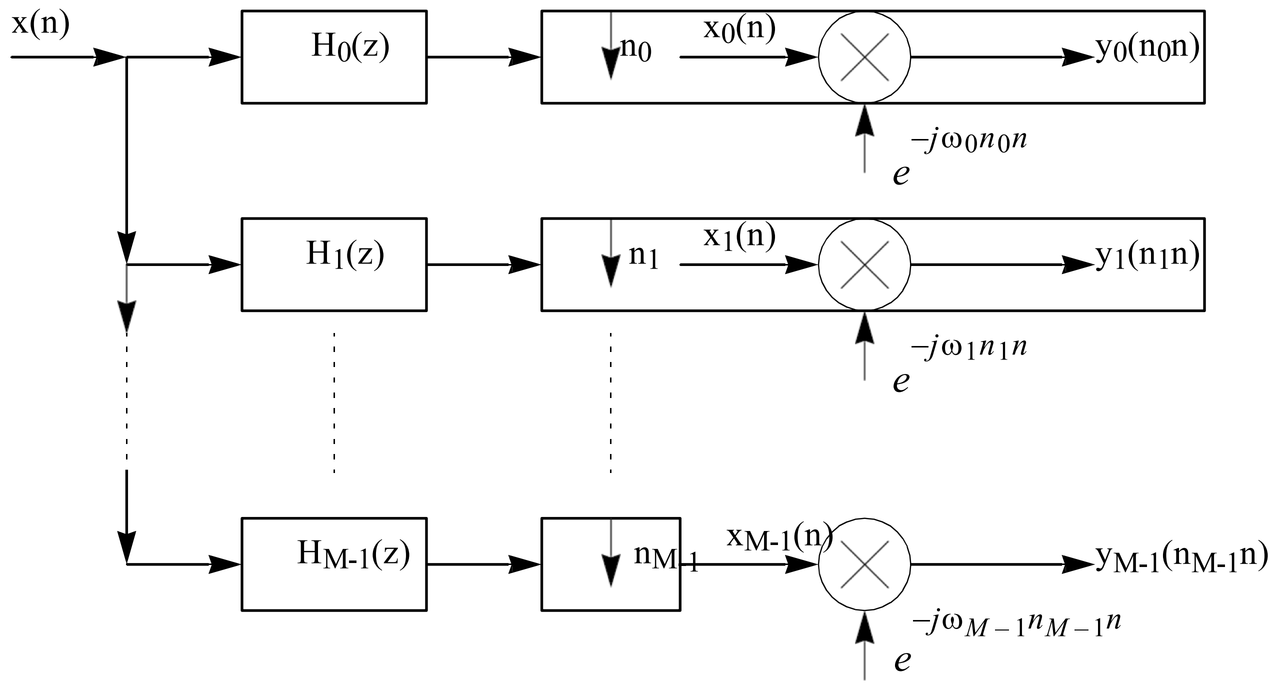

fig 20. An analysis bank with decimators and modulators

Since the filters have equal bandwidth, the decimation ratios can be taken to be equal. With $n_k = M$ (number of filters) we obtain a maximally decimated analysis bank.

5. Inversion of the STFT.

❒ As $X_{STFT}(e^{j\omega}, m)$ is the Fourier transform of $x(n)w(n-m)$, we have

$$x(n)w(n-m) = \frac{1}{2\pi}\int_0^{2\pi} X_{STFT}(e^{j\omega}, m)e^{j\omega n}d\omega \tag{26}$$

If we set $n = m$, then we obtain the STFT inversion formula as

$$x(n)w(0) = \frac{1}{2\pi}\int_0^{2\pi} X_{STFT}(e^{j\omega}, m)e^{j\omega m}d\omega \tag{27}$$

so that we can obtain $x(n)$ for all m as long as $w(0)$ is not equal to zero.

❒ A second inversion formula is given by

$$x(n) = \frac{1}{2\pi}\int_0^{2\pi}\left(\sum_{m=-\infty}^{\infty} X_{STFT}(e^{j\omega}, m)w^*(n-m)\right)e^{j\omega n}d\omega \tag{28}$$

provided $\sum_m |w(m)|^2 = 1$ (29)

6. Filter bank interpretation of the Inverse transform

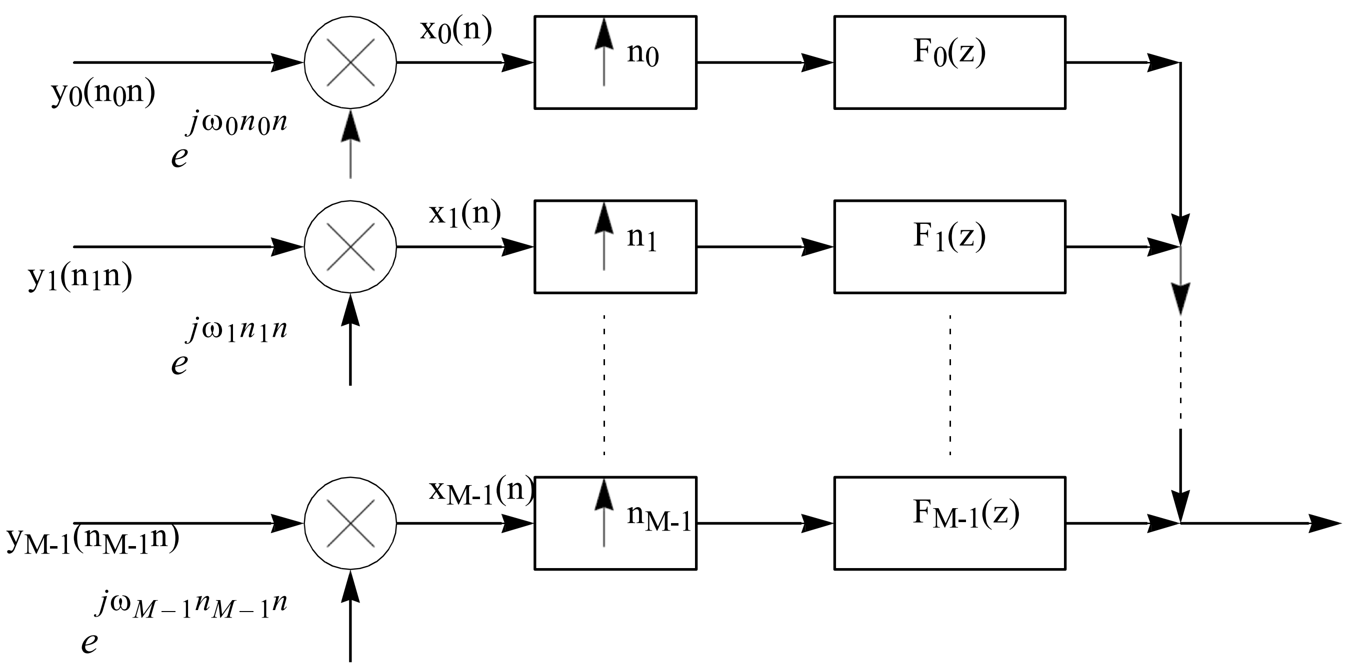

fig 21. The synthesis bank used to reconstruct x(n) from its STFT coefficients. Usually $n_k = M$ for all k

As long as the analysis filters $H_k(z)$ are chosen properly, we can find stable synthesis filters $F_k(z)$ to recover $x(n)$ perfectly.

7. Generalization of the STFT

- An arbitrary signal *x(n)* can be recovered from its decimated STFT coefficients, provided the analysis and the synthesis filter banks satisfy the perfect reconstruction(PR) property. However, the analysis filters are derived from a single prototype *w(n)* by modulation, the PR condition in turn will restrict the coefficients of *w(n)* severely. By relaxing this condition, we can obtain more flexibility. This generalized system is not derivable from a single sliding window system and is usually called “spectrum analyser”.
- The STFT pair can be written as

$$x_k(n) = \sum_{m=-\infty} x(m) h_k(n_k n - m) \tag{30}$$

$$\text{and } x(n) = \sum_{k=0}^{M-1} \sum_{m=-\infty} x_k(m) f_k(n - n_k m) \tag{31}$$

k is the filter number and *m* is the time shift

- The analysis filters $H_k(z)$ and synthesis fitlers $F_k(z)$ satisfy the PR condition.

  The decimators n_k are inversly proportional to the passband widths fo the filters $H_k(z)$.
- From an analogy of Fourier transform, we see that $f_k(n-n_k m)$ play the role of basis functions in equation (31)

  For orthonormal basis, $f_k(n) = h_k^*(-n)$.

8. Passing from STFT to Wavelets

step 1. Nonuniform filter banks

- The bandpass filters that we used for STFT have equal bandwidth as they are obtained by modulation of a single filter. As a first step, we give up this modulation scheme, and obtain filters $h_k(t)$ as

$$h_k(t) = a^{-k/2} h(a^{-k} t) \qquad \text{where } a > 1\text{, and } k = integer \tag{32}$$

In the frequency domain this can be written as

$$H_k(j\Omega) = a^{k/2} H(ja^{k}\Omega) \tag{33}$$

- Thus all the responses are obtained by frequency-scaling (unlike frequency shift for STFT) of a prototype response $H(j\Omega)$
- The scale factor $a^{-k/2}$ is meant to ensure that the energy $\int_{-\infty}^{\infty} |h_k(t)|^2 dt$ is independent of k. This can be regarded as normalizing convention.

Step 2. Nonuniform decimation

- Since the bandwidth of $H_k(j\Omega)$ is smaller for large k, we can sample its output at a corresponding lower rate. In time domain this can be seen as a large step size for large widths of $h_k(t)$. The step size for window movement is a_kT and it increases with k, that is, increases as the center frequency Ω_k (hence bandwidth) of the filter decreases.

- Therefore, we can write the wavelet transform as

$$X_{DWT}(k, n) = a^{-k/2}\int_{-\infty}^{\infty} x(t)h(nT - a^{-k}t)dt \qquad \text{where } k, n \text{ are integers} \tag{34}$$

By making the above changes, the wavelet transform can be obtained using a filter bank as shown in *figure 22*

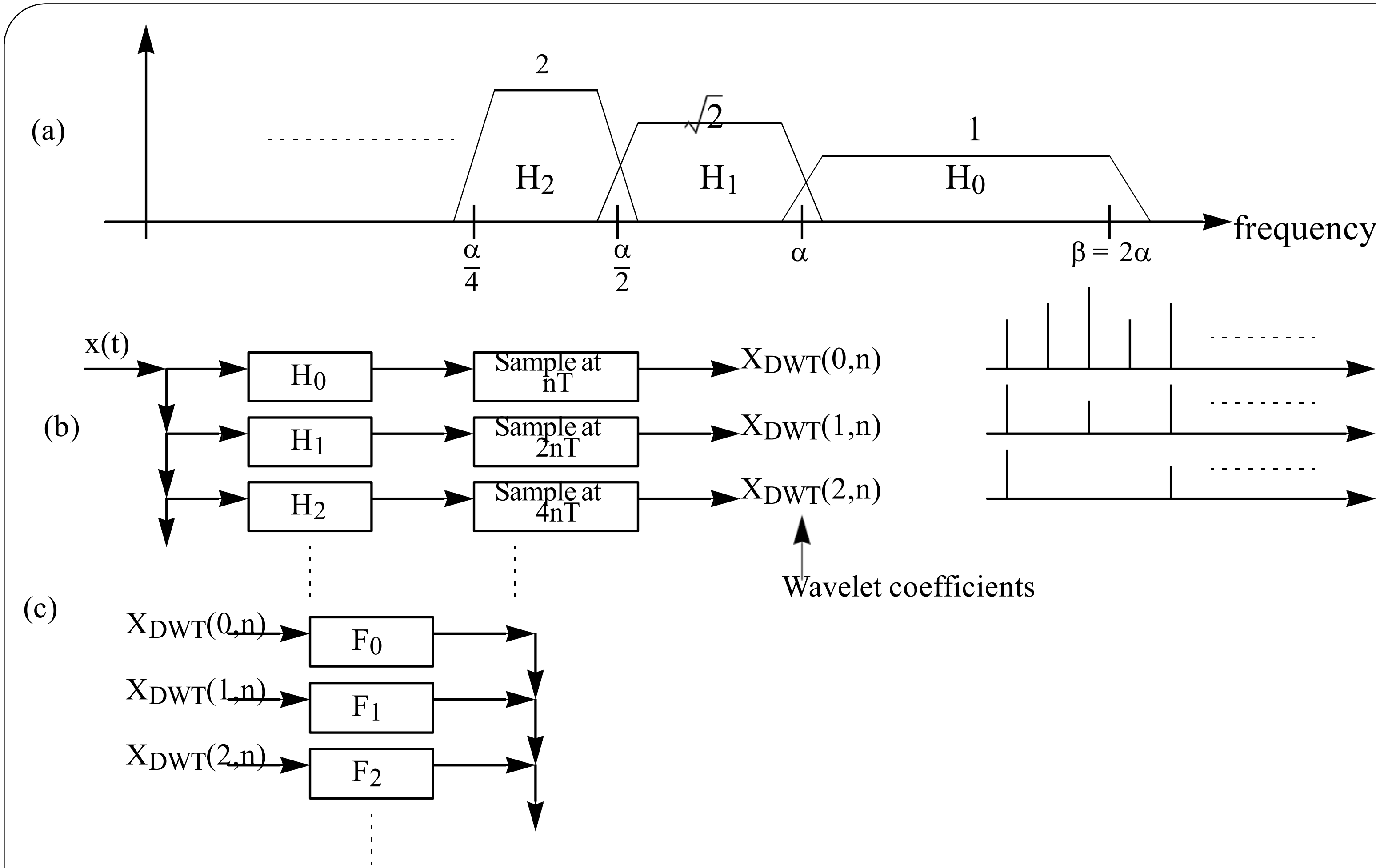

fig 22. (a) Frequency response obtained by the scaling process with a = 2. α and β are the cutoff frequencies of the bandpass filter (b) Analysis bank representation of DWT (c) Synthesis bank.