Unit 4

Vector Quantization:

Introduction:

Quantization, the process of approximating continuous-amplitude signals by digital (discrete amplitude) signals, is an important aspect of data compression or coding, the field concerned with the reduction of the number of bits necessary to transmit or store analog data, subject to a distortion or fidelity criterion. The independent quantization of each signal value or parameter is termed scalar quantization, while the joint quantization of a block of parameters is termed block or vector quantization.

Current projections for world-wide communications in the 1990s and beyond, point to a proliferation of digital transmission as a dominant means of communication for voice and data. Digital transmission is expected to provide flexibility, reliability, and cost effectiveness, with the added potential for communication privacy and security through encryption. The costs of digital storage and transmission media are generally proportional to the amount of digital data that can be stored or transmitted. While the cost of such media decreases every year, the demand for their use increases at an even higher rate. Therefore, there is a continuing need to minimize the number of bits necessary to transmit signals while maintaining acceptable signal fidelity or quality. The branch of electrical engineering that deals with the latter problem is termed data compression or coding. When applied to speech, it is known as speech compression or speech coding.

The conversion of an analog (continuous-time, continuous-amplitude) source into a digital (discrete-time, discrete amplitude) source, consists of two parts: sampling and quantization. Sampling converts a continuous-time signal into a discrete-time signal by measuring the signal value at regular intervals of time. Quantization converts a continuous-amplitude signal into one of a set of discrete amplitudes, thus resulting in a discrete-amplitude signal that is different from the continuous-amplitude signal by the quantization error or noise. When each of a set of parameters (or a sequence of signal values) is quantized separately, the process is known as scalar quantization. When the set of parameters is quantized jointly as a single vector, the process is known as vector quantization (also known as block quantization or pattern-matching quantization). We shall often abbreviate vector quantization as **VQ**.

Speech Coding:

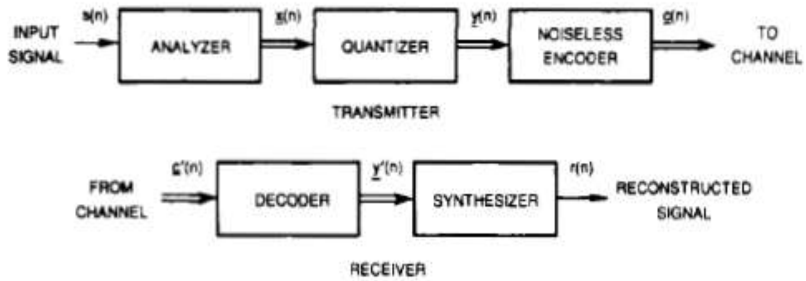

Fig. 1. Basic components of a data compression system for speech coding.

Fig. 1 shows the basic components of a data compression system appropriate for speech coding. The first component analyzes the discrete-time signal ~(n)' and extracts a vector of unquantized parameters x(n). The set of parameters x(n) is quantized into the vector y(n), which is then encoded into a sequence of bits c(n) and transmitted through the transmission channel or stored in some storage medium. (The quantizer includes any prediction and feedback loops that are an integral part of the quantization process.) In general, the output of the channel c'(n) will be different from c(n) if there are channel errors. At the receiver, the decoder converts the sequence of bits c'(n) into parameter values y'(n), which are then used as input to the synthesizer. The output r(n) is the reconstructed signal which will be an approximation to the input signal s(n)

Vector quantization (VQ) is a technique used in speech coding, which is a process of compressing speech signals to efficiently transmit or store them while maintaining acceptable quality. In VQ, speech signals are represented by vectors, and instead of encoding each individual sample, groups of samples (vectors) are encoded together.

Here's how vector quantization works in speech coding:

1. **Vector Formation**: The speech signal is divided into small frames, typically consisting of tens of milliseconds of speech. Each frame is then represented as a vector in a multidimensional space, where each dimension corresponds to a sample within the frame.
2. **Codebook Construction**: A codebook is created, consisting of a set of representative vectors. These vectors are obtained through a training process, often using techniques like clustering algorithms such as k-means, where they represent centroids of clusters of vectors from the training data. The size of the codebook (number of vectors) is a crucial parameter affecting the compression quality.
3. **Quantization**: For each frame of speech, the closest vector in the codebook is found. This process is known as quantization. The index of the closest vector is then transmitted or stored instead of the original frame. This index is typically represented using fewer bits than required to represent the original samples directly, achieving compression.
4. **Decoding**: During decoding, the index received is used to retrieve the corresponding vector from the codebook. This vector is then used to reconstruct the speech frame.

Advantages of Vector Quantization in Speech Coding:

1. **Compression Efficiency**: VQ can achieve high compression ratios compared to other techniques, as it exploits correlations between speech samples within frames.
2. **Low Complexity Decoding**: Decoding in VQ is computationally simpler compared to other compression techniques like waveform coding, making it suitable for real-time applications.
3. **Robustness**: VQ can be robust to channel errors since only the index of the codebook entry needs to be transmitted. Errors in the transmitted index can often be mitigated using error-correction techniques.

Limitations of Vector Quantization in Speech Coding:

1. **Perceptual Quality**: While VQ can achieve high compression ratios, the quality of the reconstructed speech may not be as high as other techniques like waveform coding, especially at lower bit rates.
2. **Codebook Design**: The performance of VQ heavily depends on the design of the codebook. Constructing an optimal codebook can be computationally intensive and may require substantial training data.
3. **Variable Bit Rate**: VQ may produce variable bit rate output, which can complicate transmission or storage requirements compared to fixed bit rate codecs.
4. **Complexity**: While decoding is less complex than encoding, the process of designing and optimizing the codebook can be complex and computationally expensive, especially for large codebooks.

summary,

Vector quantization offers efficient compression of speech signals with relatively low decoding complexity but requires careful design of the codebook and may exhibit lower quality compared to other compression techniques, especially at very low bit rates.

Transform based Coding:

The transform-based approach to speech coding is a method used for compressing speech signals by transforming them into a different domain where redundancies can be more efficiently represented and removed. This approach is widely used in modern speech coding standards due to its effectiveness in achieving high compression ratios while maintaining acceptable speech quality. One of the most common transform-based techniques is the Modified Discrete Cosine Transform (MDCT), which is utilized in popular speech coding standards such as MP3, AAC, and Opus.

Advantages of the transform-based approach to speech coding include:

- **High Compression Ratios**: By exploiting the redundancies present in the frequency domain, transform-based techniques can achieve high compression ratios, reducing the bitrate required to represent the speech signal.
- **Perceptual Efficiency**: Transform-based codecs can focus on preserving perceptually important components of the speech signal while discarding less relevant information, leading to improved subjective quality at low bitrates.
- **Flexibility**: Different transform sizes and quantization strategies can be employed to adapt the codec to different bitrate requirements and application scenarios.

However, there are also challenges associated with transform-based speech coding, such as managing artifacts introduced during quantization, maintaining low computational complexity for real-time encoding and decoding, and ensuring robustness to packet loss and channel errors in communication systems.

The transform commonly used in speech coding is the **Modified Discrete Cosine Transform (MDCT)**. The **MDCT** is a variant of the **Discrete Cosine Transform (DCT)** and is particularly well-suited for speech and audio coding due to its properties and efficiency.

Here's an overview of the MDCT and its mathematical formulation:

1. **Time-domain input signal**: Let $x[n]$ be a discrete-time signal representing the speech signal. This signal is usually divided into overlapping frames of length N, denoted as $x_i[n]$, where i represents the frame index.
2. **Windowing**: Before applying the MDCT, a window function $w[n]$ is applied to each frame to reduce spectral leakage and aliasing effects. Common window functions include the Hamming window or the Kaiser window.
3. **MDCT**: The MDCT is applied to each windowed frame $x_i[n]$ to obtain its frequency-domain representation. The MDCT of length N is defined as follows:
   $$X_k = \sum_{n=0}^{N-1} x[n] \cdot \cos\left(\frac{\pi}{N}\left(n + \frac{1}{2} + \frac{N}{2}\right)\left(k + \frac{1}{2}\right)\right)$$

   where X_k represents the kth MDCT coefficient, and $0 \leq k < N/2$.

4. **Frequency-domain representation**: After applying the MDCT, the speech signal is represented by $N/2$ frequency-domain coefficients. These coefficients capture the spectral content of the signal.
5. **Quantization and entropy coding**: The MDCT coefficients are quantized and further compressed using entropy coding techniques, as described earlier.

6. **Decoding**: During decoding, the compressed bitstream is processed to reconstruct the quantized MDCT coefficients. These coefficients are then inverse-transformed back into the time domain to reconstruct the original speech signal.

The MDCT is preferred over the DCT for speech and audio coding because it provides better frequency localization due to its overlapping nature, which helps reduce artifacts such as spectral leakage. Additionally, the MDCT facilitates efficient implementation through algorithms such as the Fast MDCT (FMDCT) algorithm.

Overall, the MDCT plays a crucial role in the transform-based approach to speech coding, enabling efficient compression of speech signals while maintaining acceptable audio quality.

Frequency domain coding is a technique used in speech coding (speech compression) to represent and compress digital audio signals, particularly speech, in a manner that exploits the characteristics of the frequency domain. This approach can be more efficient than time-domain coding methods like Pulse Code Modulation (PCM) because it leverages the properties of the speech signal in the frequency domain.

Here's how frequency domain coding is applied in speech coding:

Signal Analysis: The speech signal is divided into short frames or segments, typically 20-30 milliseconds in duration.

Fast Fourier Transform (FFT): Each frame is transformed from the time domain to the frequency domain using techniques like the Fast Fourier Transform (FFT). This transforms the signal from a waveform in the time domain to a representation of its frequency components.

Spectral Analysis: In the frequency domain, various spectral analysis techniques are applied to the transformed frames. These can include:
Filterbanks: The transformed signal is passed through a filterbank that splits the signal into different frequency sub-bands. These sub-bands represent different parts of the spectrum (e.g., low, mid, high frequencies).
Quantization: The magnitudes and phases of the frequency components are quantized. Quantization in the frequency domain can be more efficient, as it can exploit perceptual masking properties to allocate more bits to important components and fewer bits to less important ones.
Coding: After spectral analysis and quantization, the quantized information is transmitted or stored. Various coding schemes may be used, including entropy coding techniques to compress the data further.
Reconstruction: At the receiver end or during playback, the transmitted or stored data is decoded and used to reconstruct the original frequency-domain representation of the speech signal.
Inverse FFT: The inverse Fast Fourier Transform (IFFT) is applied to convert the frequency-domain signal back into the time domain.
Synthesis: Finally, the time-domain signal is synthesized, and the original speech waveform is reconstructed for playback.
Frequency domain coding has several advantages in speech coding:
Efficiency: It can be more efficient in terms of compression compared to time-domain coding because it can exploit the perceptual properties of human hearing and allocate bits more intelligently to different frequency components.
Robustness: Frequency domain coding can be more robust to channel errors or packet loss during transmission, making it suitable for real-time communication applications like Voice over IP (VoIP).

Quality: Depending on the implementation, frequency domain coding methods can offer good audio quality while achieving reasonable compression ratios.

Common speech coding standards that use frequency domain coding techniques include codecs like MPEG Audio (used in MP3) and various adaptive transform codecs.

In summary, frequency domain coding in speech coding leverages the frequency characteristics of speech signals to achieve efficient compression and high-quality audio reproduction, making it valuable for various communication and multimedia applications.

A transform-based approach in speech coding refers to the use of mathematical transformations, typically applied in the frequency domain, to represent and compress digital audio signals, particularly speech. These approaches transform the original time-domain speech signal into a different domain, such as the frequency domain, and then apply various techniques to encode and compress the transformed data efficiently. One of the most commonly used transformations in speech coding is the Discrete Cosine Transform (DCT) or variations of it. Here's how the transform-based approach is applied in speech coding:

Signal Segmentation: Similar to other speech coding methods, the continuous speech signal is divided into smaller frames or chunks, usually lasting 20 to 30 milliseconds.

Transform: Each frame is transformed from the time domain to the frequency domain using a mathematical transform like the Discrete Cosine Transform (DCT). The DCT is particularly popular in speech coding, as it helps concentrate most of the signal energy in a few lower-frequency coefficients, making it easier to compress.

Coefficient Quantization: The coefficients obtained after the transformation are quantized, meaning that their values are approximated to reduce the number of bits needed for representation. Quantization is a crucial step, as it determines the trade-off between audio quality and compression ratio.
Entropy Coding: To achieve further compression, entropy coding techniques, such as Huffman coding or arithmetic coding, are often applied to code the quantized coefficients efficiently.
Reconstruction: During playback or transmission, the received quantized coefficients are used to reconstruct the transformed representation of the frame.
Inverse Transform: The inverse of the original transform (e.g., inverse DCT) is applied to convert the frame back from the frequency domain to the time domain.
Overlap and Add: To ensure continuity between adjacent frames, a technique like overlap and add may be employed to combine and smooth the reconstructed frames.
Synthesis: The final reconstructed time-domain frames are combined to produce the reconstructed speech signal.
Transform-based approaches are known for their efficiency in representing speech signals, and they are used in many speech coding standards and formats, including:
MP3: MP3 (MPEG-1 Audio Layer III) uses the Modified Discrete Cosine Transform (MDCT) for transforming the signal into the frequency domain.
AAC: Advanced Audio Coding (AAC), used in formats like MPEG-4, also employs the MDCT.
G.711: The ITU-T G.711 standard for pulse code modulation (PCM) uses a simple form of DCT for speech signal encoding.

These standards balance the trade-off between compression efficiency and audio quality by carefully designing their transform-based encoding schemes.

In summary, the transform-based approach in speech coding leverages mathematical transformations, such as the Discrete Cosine Transform, to efficiently represent and compress speech signals while maintaining acceptable audio quality. This method is widely used in many audio compression standards and formats.

Model-based coding in the context of speech coding (speech compression) involves the use of mathematical or statistical models to represent and encode speech signals more efficiently than traditional waveform-based coding methods. These models can capture various aspects of speech, including the spectral envelope, excitation source, and other parameters that describe speech production.

Here's how model-based coding is applied in speech coding:

Parameter Extraction: The continuous speech signal is divided into frames, typically lasting 20-30 milliseconds. For each frame, relevant speech parameters are extracted. These parameters may include:

Spectral Parameters: These represent the spectral envelope of the speech frame, describing the distribution of energy across different frequency bands.

Excitation Parameters: These describe the excitation source, which includes pitch information and voicing characteristics.

Other Parameters: Depending on the coding method, additional parameters such as formant frequencies or LPC coefficients may be included.

Modeling: The extracted parameters are used to construct mathematical models that capture the characteristics of the speech signal. For example, a widely used model in speech coding is the Linear Predictive Coding (LPC) model, which represents the spectral envelope of the speech frame.

Quantization: The parameters obtained from the modeling stage are quantized, meaning that their values are approximated to reduce the number of bits needed for representation. The design of the quantization process is essential, as it influences audio quality and compression efficiency.

Entropy Coding: To further compress the quantized parameters efficiently, entropy coding techniques like Huffman coding or arithmetic coding may be applied.

Decoding: During playback or transmission, the received quantized parameters are used to reconstruct the models.

Synthesis: The models are used to synthesize the speech signal. For example, in LPC-based coding, the LPC model is used to resynthesize the spectral envelope, and the excitation parameters are used to generate the excitation source.

Model-based coding methods are known for their efficiency in representing speech signals and offer advantages like:

High compression efficiency: Model-based coding can achieve high compression ratios while maintaining reasonable audio quality.

Robustness: These methods can be more robust to channel errors or packet loss during transmission, making them suitable for real-time communication applications like Voice over IP (VoIP).

Parameter control: Model-based coding allows for more fine-grained control over various speech parameters, which can be useful for applications like speech synthesis and voice transformation.

Common speech coding standards that use model-based coding techniques include the Code-Excited Linear Prediction (CELP) and

Algebraic Code-Excited Linear Prediction (ACELP) coding methods, which are used in standards like G.711, G.729, and G.723.1 for various telecommunication and voice communication applications.

In summary, model-based coding in speech coding leverages mathematical or statistical models to efficiently represent and compress speech signals while maintaining acceptable audio quality. This approach is valuable for various communication and multimedia applications.

Audio compression techniques, which are used in speech coding as well as other audio-related applications, aim to reduce the amount of data required to represent and transmit audio signals while maintaining acceptable audio quality. These techniques are crucial for applications like voice communication, audio streaming, and multimedia content delivery. Here are some common audio compression techniques often used in speech coding:

Pulse Code Modulation (PCM): PCM is a straightforward method where the analog audio signal is sampled and quantized to discrete values. While it offers high audio quality, it is not very efficient in terms of data compression. It serves as the basis for other compression techniques.

Transform Coding: Transform coding is a technique that converts audio data from the time domain to the frequency domain using transforms like the Discrete Fourier Transform (DFT) or Discrete Cosine Transform (DCT). The transformed data can be quantized and compressed more efficiently.

Vector Quantization (VQ): As previously discussed, VQ is a technique that partitions the transformed data into clusters and encodes each cluster with a representative codeword. This is an efficient way to represent audio data, including speech, with reduced data rates.

Code-Excited Linear Prediction (CELP): CELP is a widely used technique for speech coding. It models the speech signal as an excitation signal (e.g., pitch and pulse) and a linear predictive coding (LPC) filter. CELP encodes the parameters of these models to achieve high compression ratios with good speech quality.

Algebraic Code-Excited Linear Prediction (ACELP): ACELP is an extension of CELP that combines algebraic codebooks with pitch and pulse excitations. This approach improves the efficiency and quality of speech coding, making it suitable for various communication systems.

Psychoacoustic Modeling: This technique is based on the principles of psychoacoustics, which study how the human auditory system perceives sound. It removes audio components that are less perceptible to the human ear, allowing for higher compression rates without significant loss in perceived quality. It's often used in perceptual audio codecs like MP3.

Lossless Compression: For applications where preserving every detail of the audio signal is critical, lossless compression techniques like FLAC and ALAC (Apple Lossless) are used. These techniques compress audio data without any loss in quality.

Variable Bit Rate (VBR) Coding: Instead of using a constant bit rate, VBR coding dynamically allocates more bits to complex parts of the audio and fewer bits to simpler sections. This improves efficiency while maintaining quality.

Speech Coding Standards: Standards developed by organizations like ITU-T and MPEG define specific algorithms and bitstream formats for audio and speech coding. Examples include G.711, G.729, and various MP3 standards.

Wavelet Transform: Wavelet transforms can be used in audio compression to represent the audio signal at various scales and resolutions. They are employed in codecs like FLAC.

The choice of audio compression technique depends on the specific application and the trade-off between audio quality and data rate. For real-time communication, speech coding methods like CELP and ACELP are often preferred, whereas perceptual audio codecs like MP3 and AAC are suitable for music and multimedia applications