Unit 2

Sound/Audio system

- Concepts of Sound System
- Music and MIDI
- Speech Generation, Speech Analysis, Speech Transmission

What is sound ?

- Sound is a physical phenomenon produced by the vibration of matter, such as guitar string or a block of wood.

- As the matter vibrates, it creates pressure waves that propagate through some medium(air, solid, or liquid). When the waves reach human ear, a sound is heard.

- Sound/audio processing techniques basically process those sound waves (acoustic signals).

- Audio systems are mainly concerned with coding, storage and processing of sound and speech.

Basic sound concepts

Sound is produced by the vibration of matter. During the vibration, pressure variations are created in the air surrounding it. The pattern of the oscillation is called a *waveform* (Figure 3.1 [Tec89]).

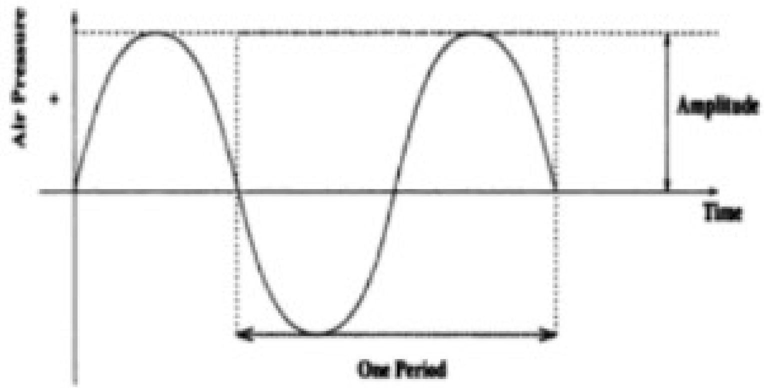


Figure 3.1: *Oscillation of an air pressure wave.*

The waveform repeats the same shape at regular intervals and this portion is called a *period*. Since sound waves occur naturally, they are never perfectly smooth or uniformly periodic. However, sounds that display a recognizable periodicity tend to be more musical than those that are nonperiodic. Examples of periodic sound sources are musical instruments, vowel sounds, the whistling wind and bird songs. Nonperiodic sound sources include unpitched percussion instruments, coughs and sneezes and rushing water.

Basic sound concepts

Frequency

The *frequency* of a sound is the reciprocal value of the period; it represents the number of periods in a second and is measured in *hertz* (Hz) or *cycles per second* (cps). A convenient abbreviation, kHz (kilohertz), is used to indicate thousands of oscillations per second: 1 kHz equals 1000 Hz [Boo87]. The frequency range is divided into:

Infra-sound	from 0 to 20 Hz
Human hearing frequency range	from 20 Hz to 20 kHz
Ultrasound	from 20 kHz to 1 GHz
Hypersound	from 1 GHz to 10 THz

Multimedia systems typically make use of sound only within the frequency range of human hearing. We will call sound within the human hearing range *audio* and the waves in this frequency range *acoustic signals* [Boo87]. For example, speech is an acoustic signal produced by humans; music signals have a frequency range between 20 Hz and 20 kHz. Besides speech and music, we denote any other audio signal as *noise*.

Basic sound concepts

Amplitude

A sound also has an *amplitude*, a property subjectively heard as loudness. The amplitude of a sound is the measure of the displacement of the air pressure wave from its mean, or quiescent state.

Amplitude is the measure of sound levels. For a digital sound, amplitude is the sample value.

The reason that sounds have different loudness is that they carry different amount of power. The unit of power is watt. The intensity of sound is the amount of power transmitted through an area of $1m^2$ oriented perpendicular to the propagation direction of the sound.

If the intensity of a sound is $1 watt/m^2$, we may start feel the sound. The ear may be damaged. This is known as the *threshold of feeling*. If the intensity is $10^{-12} watt/m^2$, we may just be able to hear it. This is know as the *threshold of hearing*.

Bandwidth: it is the range of frequencies a device can produce or a human can hear.

FM radio	50Hz – 15kHz
AM radio	80Hz – 5kHz
CD player	20Hz – 20kHz
Sound Blaster 16 sound card	30Hz – 20kHz
Inexpensive microphone	80Hz – 12kHz
Telephone	300Hz – 3kHz
Children's ears	20Hz – 20kHz
Older ears	50Hz – 10kHz
Male voice	120Hz – 7kHz
Female voice	200Hz – 9kHz

- Sound is analog in nature. To represent sound in computer, it must be converted to digital form
- Furthermore, humans can not hear a digitally represented sound. In order to hear the sound, it must be converted into analog form.

3.1.1 Computer Representation of Sound

The smooth, continuous curve of a sound waveform is not directly represented in a computer. A computer measures the amplitude of the waveform at regular time intervals to produce a series of numbers. Each of these measurements is a *sample*. Figure 3.2 illustrates one period of a digitally sampled waveform.

The mechanism that converts an audio signal into digital samples is the *Analog-to-Digital Converter (ADC)*. The reverse conversion is performed by a *Digital-to-Analog Converter (DAC)*.

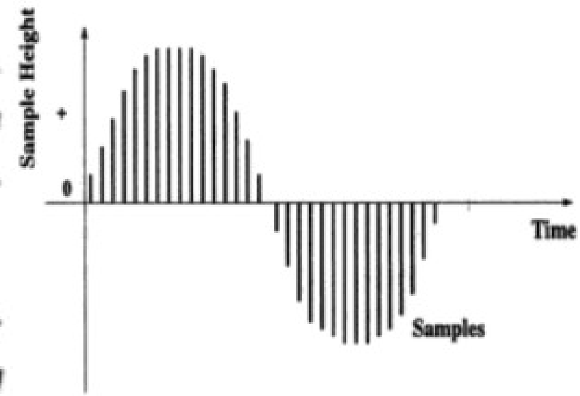


Figure 3.2: *Sampled waveform.*

Basic sound concepts

Sampling Rate

The rate at which a continuous waveform (Figure 3.1) is sampled is called the *sampling rate*. Like frequencies, sampling rates are measured in Hz. The CD standard sampling rate of 44100 Hz means that the waveform is sampled 44100 times per second. This seems to be above the frequency range the human ear can hear. However, the bandwidth (which in this case is 20000 Hz - 20 Hz = 19980 Hz) that digitally sampled audio signal can represent, is at most equal to half of the CD standard sampling rate (44100 Hz). This is an application of the Nyquist sampling theorem. ("For lossless digitization, the sampling rate should be at least twice the maximum frequency responses.") Hence, a sampling rate of 44100 Hz can only represent frequencies up to 22050 Hz, a boundary much closer to that of human hearing.

Quantization

Just as a waveform is sampled at discrete times, the value of the sample is also discrete. The resolution or *quantization* of a sample value depends on the number of bits used in measuring the height of the waveform. An 8-bit quantization yields 256 possible values; 16-bit CD-quality quantization results in over 65536 values.

Figure 3.3 presents a 3-bit quantization. The sampled waveform with a 3-bit quantization results in only eight possible values: .75, .5, .25, 0, -.25, -.5, -.75 and -1.

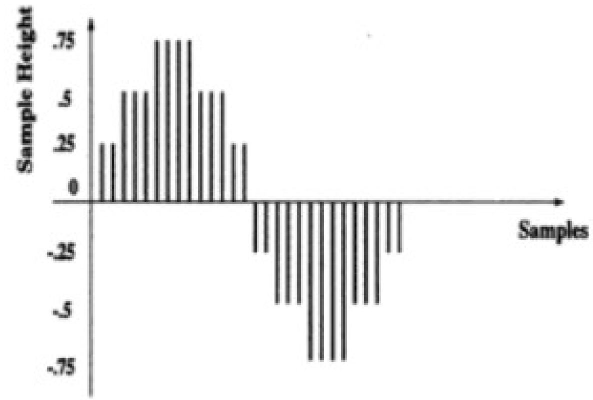


Figure 3.3: *Three-bit quantization.*

The lower the quantization, the lower the lower the quality of sound

The lower the sampling, the lower the lower the size (memory space)

Quality versus File Size

The size of a digital recording depends on the sampling rate, resolution and number of channels.

$$S = R \times (b/8) \times C \times D$$

Higher sampling rate, higher resolution gives higher quality but bigger file size.

S	file size	bytes
R	sampling rate	samples per second
b	resolution	bits
C	channels	1 - mono, 2 - stereo
D	recording duration	seconds

For example, if we record 10 seconds of stereo music at 44.1kHz, 16 bits, the size will be:

$$\begin{aligned} S &= 44100 \times (16/8) \times 2 \times 10 \\ &= 1,764,000\textbf{bytes} \\ &= 1722.7\textbf{Kbytes} \\ &= 1.68\textbf{Mbytes} \end{aligned}$$

Note: $$\begin{aligned} 1\text{Kbytes} &= 1024\text{bytes} \\ 1\text{Mbytes} &= 1024\text{Kbytes} \end{aligned}$$

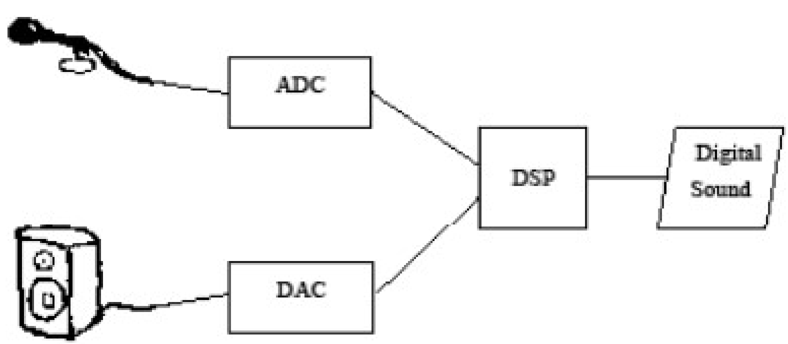
High quality sound files are very big, however, the file size can be reduced by compression.

Sound hardware

- Before the sound can be processed, a computer needs input/output devices.
- Microphones and speakers are connected to ADC and DAC, respectively for input and output of audio.

- Recording and Digitising sound:
  - An *analog-to-digital converter*(ADC) converts the analog sound signal into digital samples.
  - A *digital signal processor*(DSP) processes the sample, e.g. filtering, modulation, compression, and so on.
- Play back sound:
  - A *digital signal processor* processes the sample, e.g. decompression, demodulation, and so on
  - An *digital-to-analog converter*(DAC) converts the digital samples into sound signal
- All these hardware devices are integrated into a few chips on a sound card

- Different sound card have different capability of processing digital sounds. When buying a sound card, you should look at:
  - maximum sampling rate
  - stereo or mono
  - duplex or simplex



Prepared by: Shiv Raj Pant

Audio formats

- For audio, the minimum number of samples per second to unambiguously represent English speech is 8000 Hz. Using less than that would result in speech that might not be comprehensible.
- For telephone quality speech, 8 kHz sampling rate and 8 bit quantization is considered fast and accurate enough.
- CD and DVD quality audio uses 44100 samples per second and 16 bit PCM (pulse code modulation) quantization.
- Audio tracks in movies use 48kHz sampling.
- Audio file formats specify following elements which also differentiate between different formats:
  - Type of modulation
  - Size of the data
  - Number of channels
  - Samples per second
  - Bytes per sample

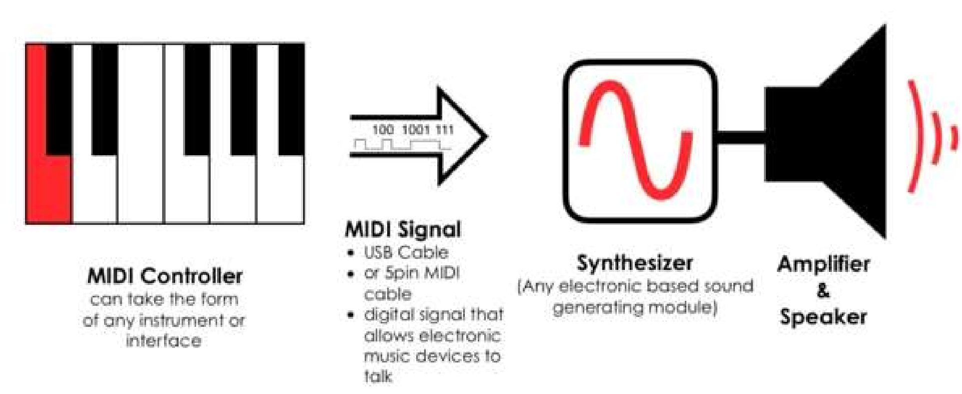
Some popular audio file formats are:

1. PCM(pulse code modulation)
2. AAC(Advanced audio coding)
3. WMA(windows media audio)
4. MP3(MPEG-1 Audio layer3)

Homework: *Write short notes (about one paragraph with important details such as file header format, sampling rate, frequency, compression type etc.*

Computer music and MIDI

- Sound can be natural(such as human voice, guitar music etc) or artificially synthesized (computer generated).
- Electronic instrument that uses some form of digital or analog processing to produce audible sound is called synthesizer. Most synthesizers artificially produce (or synthesize) the sounds of acoustic instruments.
- MIDI (musical instrument digital interface) is a protocol standard developed in the 1980's which allows electronic instruments and other digital musical tools to communicate with each other.
- In simple words, MIDI is the interface between electronic musical instrument and computer.
- MIDI is universally accepted standard for communicating musical signal information by digital means.
- Whenever, two devices want to share musical information, they follow MIDI standard to build their connection ports.



MIDI Hardware

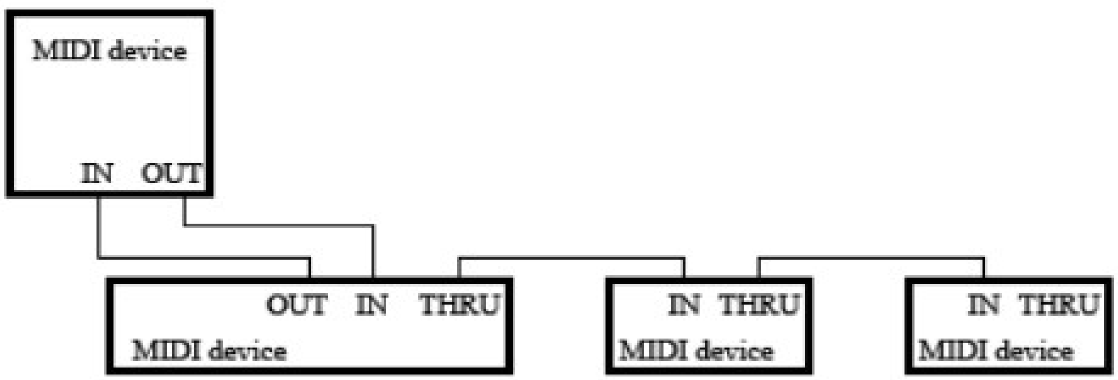
An electronic musical instrument or a computer which has MIDI interface should has one or more MIDI ports. The MIDI ports on musical instruments are usually labelled with:

IN — for receiving MIDI data;

OUT — for outputting MIDI data that are generated by the instrument;

THRU — for passing MIDI data to the next instrument.

MIDI devices can be daisy-chained together.

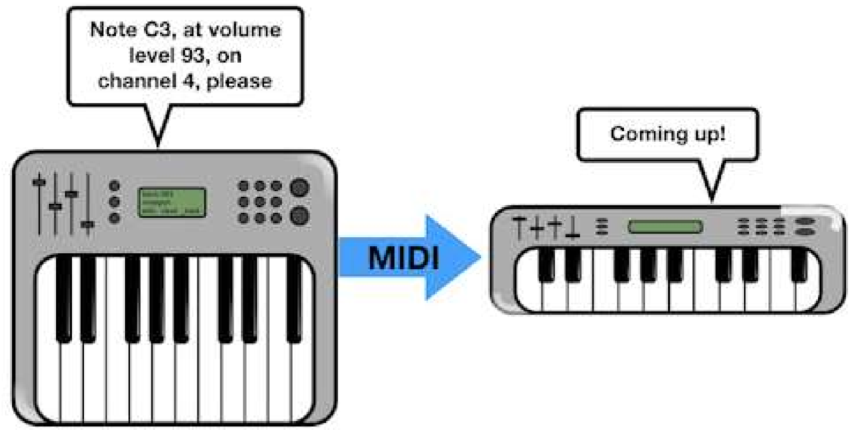


Components of MIDI

MIDI has two different components

1. *Hardware*
   - Hardware connects the equipments.
   - MIDI specifies the physical connection between musical instruments that includes the connection port, MIDI cable, and electronic signals that are sent through cable.

2. *Data format*
   - Data format is concerned with encoding the information travelling through the hardware.
   - The encoding includes:
     - instrument specification
     - The notion of beginning and end of a *note*
     - Basic frequency
     - Sound volume
   - The data format does not include encoding of individual samples (as in audio format)



Traditional MIDI cable and ports.

The MIDI data format is digital; the data are grouped into *MIDI messages*. Each MIDI message communicates one *musical event* between machines. These musical events are usually actions that a musician performs while playing a musical instrument. The action might be pressing keys, moving slider controls, setting switches and adjusting foot pedals.

When a musician presses a piano key, the MIDI interface creates a MIDI message where the beginning of the note with its stroke intensity is encoded. This message is transmitted to another machine. In the moment the key is released, a corresponding signal (MIDI message) is transmitted again. For ten minutes of music, this process creates about 200 Kbytes of MIDI data, which is essentially less than the equivalent volume of a CD-audio coded stream in the same time.

If a musical instrument satisfies both components of the MIDI standard, the instrument is a *MIDI device* (e.g., a synthesizer), capable of communicating with other MIDI devices through *channels*. The MIDI standard specifies 16 channels. A MIDI device (musical instrument) is mapped to a channel. Music data, transmitted through a channel, are reproduced at the receiver side with the synthesizer instrument. The MIDI standard identifies 128 instruments, including noise effects (e.g., telephone, air craft), with unique numbers. For example, 0 is for the Acoustic Grand Piano, 12 for the marimba, 40 for the violin, 73 for the flute, etc.

Some instruments allow only one note to be played at a time, such as the flute. Other instruments allow more than one note to be played simultaneously, such as the organ. The maximum number of simultaneously played notes per channel is a main property of each synthesizer. The range can be from 3 to 16 notes per channel.

To tune a MIDI device to one or more channels, the device must be set to one of the MIDI *reception modes*. There are four modes:

- Mode 1: Omni On/Poly;
- Mode 2: Omni On/Mono;
- Mode 3: Omni Off/Poly;
- Mode 4: Omni Off/Mono

The first part (omni) specifies how the MIDI device monitors the incoming MIDI channels.
Omni On: monitor all channels respond to all channel messages
Omni off: respond only to channel message sent on the channel device is set to receive.
The second part tells the MIDI device how to play notes coming in over the MIDI cable.
Mono- play one note at a time
Poly play many notes at a time

MIDI devices:

- MIDI system is composed of many types of devise:
  - ➢ Data generator: Usually a keyboard (digital/software or analog/hardware)
  - ➢ Sound generator: reproduces data generated by keyboard
  - ➢ MIDI interface: through the MIDI interface, computer can control output of individual instruments, receive, store and process coded information
  - ➢ Sequencer: stores and modify musical data. In multimedia system, sequencer is a computer application.

Synthesizer

- The heart of any MIDI system is the *synthesizer device*.
- A typical synthesizer device looks like a piano keyboard
- Common Synthesizer has following components:

i. Sound generator *ii. Microprocessor*

iii. Keyboard *iv. Control panel*

v. Auxiliary controllers *vi. memory*

These are very simple concepts.
Please see the book.
Available to view at google books.
https://books.google.com.np/books?hl=sv&lr=&id=Y83LQyWXzDUC&oi=fnd&pg=PA1&ots=aewZ8Ys4Me&sig=_y48fFESFJBsfoTEcAWGRnsoKwY&redir_esc=y#v=onepage&q&f=false

- *Sound Generators*

Sound generators do the actual work of synthesizing sound; the purpose of the rest of the synthesizer is to control the sound generators. The principal purpose of the generator is to produce an audio signal that becomes sound when fed into a loudspeaker. By varying the voltage oscillation of the audio signal, a sound generator changes the quality of the sound – its pitch, loudness and tone color – to create a wide variety of sounds and notes.

Internally, sound generation can be done in different ways. One way is to store the acoustic signals as MIDI data in advance. Afterwards, the stored MIDI data are transformed with a digital-analog adapter into acoustic signals. Individual notes are composed in a timely fashion. Another method is to create acoustic signals synthetically.

- *Microprocessor*

  The microprocessor communicates with the keyboard to know what notes the musician is playing, and with the control panel to know what commands the musician wants to send to the microprocessor. The microprocessor then specifies note and sound commands to the sound generators; in other words, the microprocessor sends and receives MIDI messages.

- *Keyboard*

  The keyboard affords the musician's direct control of the synthesizer. Pressing keys on the keyboard signals the microprocessor what notes to play and how long to play them. Some synthesizer keyboards can also signal to the microprocessor how loud to play the notes and whether to add *vibrato* or other effects to the notes. The sound intensity of a tone depends on the speed and acceleration of the key pressure. The keyboard should have at least five octaves with 61 keys.

- *Control Panel*

  The control panel controls those functions that are not directly concerned with notes and durations (controlled by the keyboard). Panel controls include: a slider that sets the overall volume of the synthesizer, a button that turns the synthesizer on and off, and a menu that calls up different patches for the sound generators to play.

- *Auxiliary Controllers*

  Auxiliary controllers are available to give more control over the notes played on the keyboard. Two very common variables on a synthesizer are *pitch bend* and *modulation*. Pitch bend controllers can bend pitch up and down, adding *portamento* (a smooth, uninterrupted glide in passing from one tone to another) to notes; modulation controllers can increase or decrease effects such as *vibrato*.

- *Memory*

  Synthesizer memory is used to store patches for the sound generators and settings on the control panel. Many synthesizers also have a slot for *external memory cartridges*. By using several memory cartridges, the musician can plug in a different cartridge each time s/he wants a set of new sounds for the synthesizer.

There are many other MIDI devices that augment the standard synthesizer in a MIDI system. Examples are drum machines which specialize in percussion sounds and rhythms, the master keyboard which increases the quality of the synthesizer keyboard, guitar controllers, guitar synthesizers, drum pad controllers and so on.

An important MIDI device is a *sequencer*, which can be a drum machine, computer or dedicated sequencer. A sequencer was used originally as a storage server for generated MIDI data. Today, a sequencer, being a computer, becomes additionally a music editor. Data can be modified in a proper way because of their digital data representation.

MIDI messages

MIDI messages transmit information between MIDI devices and determine what kinds of musical events can be passed from device to device.

The format of MIDI messages consists of the status byte (the first byte of any MIDI message), which describes the kind of message, and data bytes (the following bytes).

MIDI messages are divided into two different types:

1. Channel messages:

- Channel messages go only to specific devices.
- There are two types of channel messages:

i. *Channel voice messages*: send actual performance data between MIDI devices, describing keyboard action, controller action, and controller panel changes. They also describe music by defining pitch and other sound qualities. Examples of CVM are: "*note on*", "*note off*", "*control change*" etc

ii. *Channel mode messages*: determine a way that a receiving MIDI device responds to channel voice messages. They set the MIDI channel receiving modes for different devices, stop spurious notes from playing etc.. Examples: "*Omni mode off*", "*all notes off*, "*local control*" etc

2. System messsages

- System messages go to all devices in a MIDI system (rather than to specific channel).
- System messages-
  - Synchronize the timing of MIDI devices in performance.
  - Prepare sequencers and synthesizers to play a music.
  - Perform other manufacturer-specific customization tasks.
  - Examples: "*system reset*", "*Timing clock*", "*song select*", etc..

MIDI software

- MIDI softwares are applictions that help run a MIDI system through computer
- Computers provide high level control to the composer or sound designers
- There are four types of MIDI softwares:
  - *Music recording and performance applications*

    This category of applications provides functions such as recording of MIDI messages as they enter the computer from other MIDI devices, and possibly editing and playing back the messages in performance.

  - *Musical notations and printing applications*

    This category allows writing music using traditional musical notation. The user can then play back the music using a performance program or print the music on paper for live performance or publication.

  - *Synthesizer patch editors and librarians*

    These programs allow information storage of different synthesizer patches in the computer's memory and disk drives, and editing of patches in the computer.

  - *Music education applications*

    These software applications teach different aspects of music using the computer monitor, keyboard and other controllers of attached MIDI instruments.

MIDI Vs digital audio (difference between MIDI and other audio e.g. WAV, PCM etc formats)

Digital Audio

- Digital representation of physical sound waves
- File size is large if without compression
- Quality is in proportion to file size
- More software available
- Play back quality less dependent on the sound sources
- Can record and play back any sound including speech

MIDI

- Abstract representation of musical sounds and sound effects
- MIDI files are much more compact
- File size is independent to the quality
- Much better sound if the sound source is of high quality
- Need some music theory
- Cannot generate speech

Speech

- Speech can be natural (human-generated) or artificial.
- Humans can perceive and understand speech and differentiate between speech and noise.
- For perceiving, filtering and understanding speech both ears are used by human brain. Filtering with one ear is difficult.
- For generating and processing speech, we should have knowledge about how our natural speech producing organs work, and how we perceive speech signal. e.g.
    - human hear is most sensitive in the range from 600Hz to 6000Hz
    - Voiced speech signals show almost periodic behavior during certain time intervals
    - The spectrum of audio signals show a characteristic maxima (3-5 frequency bands).
    - Etc... ..

Artificial speech

- A machine can also support speech generation and recognition.
- We now have text-to-speech conversion devices where a machine can convert given text into human-like speech.
- Artificial voice are used in different public places for general information (e.g. at airports)

In this section we will focus on computer-generated speech and recognition

Speech generation

- Speech generation is one of the fascinating things for researchers from long time ago.
- In mid 19th century, Helmholtz built a mechanical vocal tract using several mechanical resonators.
- In 1940, Dudley produced the first speech synthesizer through imitation of mechanical vibration using electrical oscillation.

<u>Requirements for speech generation</u>

- ➢ Real-time signal generation
  - Must produce sound without delay especially in text-to-speech conversion.
- ➢ Understandability and naturality
  - Generated speech must be understandable and must sound natural to increase user acceptance

Some basic notions about speech

- The lowest periodic spectral component of the speech signal is called the *fundamental frequency*. It is present in a voiced sound.
- A *phone* is the smallest speech unit, such as the *m* of *mat* and the *b* of *bat* in English, that distinguish one utterance or word from another in a given language.
- *Allophones* mark the variants of a phone. For example, the aspirated *p* of *pit* and the unaspirated *p* of *spit* are allophones of the English phoneme *p*.
- The *morph* marks the smallest speech unit which carries a meaning itself. Therefore, *consider* is a morph, but *reconsideration* is not.
- A *voiced sound* is generated through the vocal cords. *m*, *v* and *l* are examples of voiced sounds. The pronunciation of a voiced sound depends strongly on each speaker.
- During the generation of an *unvoiced sound*, the vocal cords are opened. *f* and *s* are unvoiced sounds. Unvoiced sounds are relatively independent from the speaker.

How to generate sound?

1. Reproduced speech output
2. Time-dependent sound concatenation
3. Frequency-dependent sound concatenation

1. Reproduced speech output

- The easiest method out sound generation on output is to use prerecorded speech and play it back in timely fashion
- The speech can be stored as PCM in compressed form

2. Time-dependent sound concatenation

- Speech can also be produced by sound concatenation in timely fashion.
- Individual speech units are composed like building blocks.

Three approaches are possible

i. Phone sound concatenation

- individual phones are used as speech units
- Transition between phones is problematic

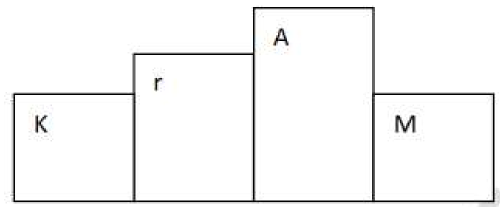


Fig: Phone sound concatenation

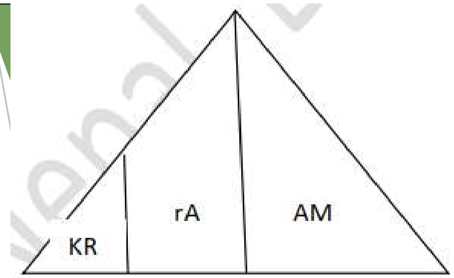


Fig: Di-phone concatenation

Syllable sound.

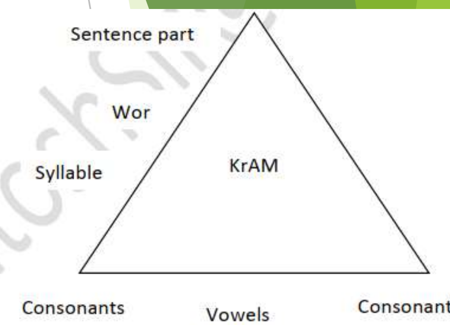


Fig: Word sound concatenation

ii. Di-phone concatenation

- Using two phones together as units

iii. Word sound concatenation

- The speech is generated through syllables.
- The pronounciation is achieved through storage of whole word
- Solves transition problem

3. Frequency-dependent sound concatenation

- This method of speech generation uses formant synthesis.
- Formants are frequency maxima in the speech signal
- Formant synthesis simulates the vocal tract through a filter.
- A pulse signal with a frequency corresponding to the fundamental speech frequency is chosen as a simulation of voice sound
- Unvoiced sounds are created through a noise generator
- Individual speech units (e.g. phones) are defined through characteristic value of formants

Speech synthesis:

Using speech synthesis, a text can be transformed into acoustic signal

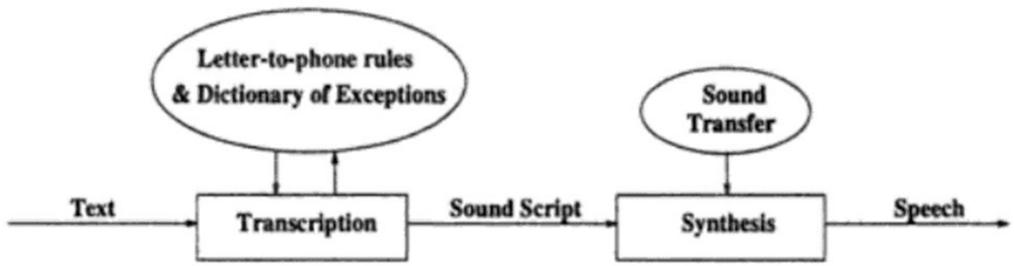


Figure 3.8: *Components of a speech synthesis system with time-dependent sound concatenation.*

Step 1: translate text into sound script using letter-to-phone rules and dictionary of exceptions.

Step 2: translate sound script into speech signal using time-dependent or frequency-dependent concatenation

Step 1 is always performed with software. But step 2 is implemented with signal processors or even dedicated processors.

Speech analysis

- Speech analysis has numerous application potentials.
- Specifically, speech analysis deals with the following topics (shown in figure).

1. *Identificatin and verification of speaker*
2. *Recognition and understanding of speech signal*
3. *Identification of mood/mental state of speaker(e.g. whether speaker is lying)*

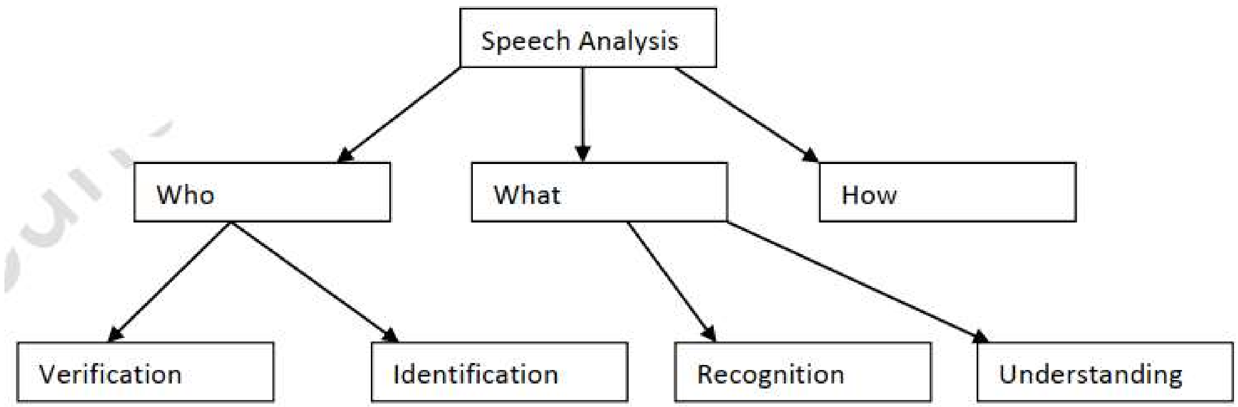


Fig. Speech analysis deals with research area shown in the above figure.

Details of above topics are presented as screenshots from the book in further slides

- Human speech has certain characteristics determined by a speaker. Hence, speech analysis can serve to analyze *who* is speaking, i.e., to *recognize a speaker* for his/her *identification* and *verification*. The computer identifies and verifies the speaker using an acoustic fingerprint. An acoustic fingerprint is a digitally stored speech probe (e.g., certain statement) of a person; for example, a company that uses speech analysis for identification and verification of employees. The speaker has to say a certain sentence into a microphone. The computer system gets the speaker's voice, identifies and verifies the spoken statement, i.e., determines if the speech probe matches the speaker's spoken statement.

- Another main task of speech analysis is to analyze *what* has been said, i.e., to recognize and understand the speech signal itself. Based on speech sequence, the corresponding text is generated. This can lead to a speech-controlled typewriter, a translation system or part of a workplace for the handicapped.

- Another area of speech analysis tries to research speech patterns with respect to *how* a certain statement was said. For example, a spoken sentence sounds differently if a person is angry or calm. An application of this research could be a lie detector.

Speech recognition and understanding

- The general three steps in speech recognition and understanding are shown in figure.
- The system repeatedly goes through these steps several times before fully understanding the speech.
-

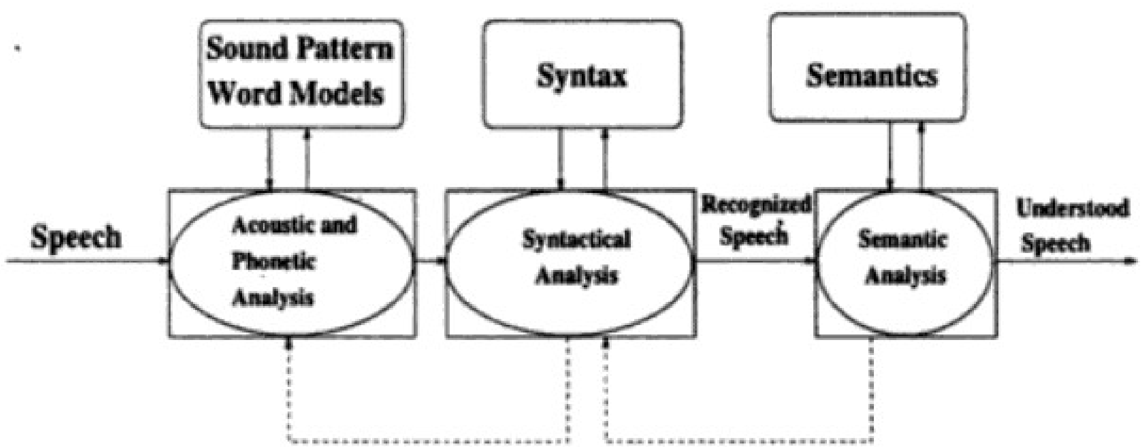


Figure 3.11: *Components of speech recognition and understanding.*

1. **Acoustic and phonetic analysis**
   In this step, basic low level analysis is performed to develop the basic units of speech.

2. **Syntactical analysis**
   The units identified in step 1 go through syntactic analysis. Errors are recognized, unambiguity is eliminated. Now the speech has been correctly recognized.

3. **Semantic analysis**
   Finally, the recognized speech goes through semantic analysis to understand the meaning of what is said in the speech. Sophisticated analysis models, AI and neural nets can be used for this purpose.

Speech transmission

- Speech transmission deals with the efficient coding of speech signal for transmission over networks without losing quality.

example: The data rate of a PCM-coded stereo-audio signal with CD-quality requirements is:

2 * 44100/s * 16 bit / 8 = 176400 bytes/sec

For telephone quality speech transmission, we need

$$1 * \frac{8000}{sec} * 8 \text{ bits} = 64 \text{ Kbit/s}.$$

- We see that speech transmission requires significant network bandwidth.
- *Speech coding refers to a process that reduces the bit rate of a speech file*

Types of speech coding :

1. *Signal form (waveform) coding*
2. *Source coding*

1. *Signal form (waveform) coding*

- Also called waveform coding
- This is the simplest technique for speech coding.
- The waveform coders analyze code and reconstruct original signal, sample-by-sample.
- This kind of coding considers no speech-specific properties and parameters.
- Here, the goal is to achieve the most efficient coding of the audio signal.i.e. reproduce the exact shape of speech signal waveform without considering nature of human speech production and delivery system.

Types of signal form coding.

- Most popular waveform coding techniques are *Difference Pulse Code Modulation*, and *Adaptive Pulse Code Modulation*.

DPCM	ADPCM
• Calculates the difference between the samples then transmits this small difference data instead of entire input sample signal. • Since the the size of the difference data is very less than the actual sample data, the number of bits required for transmission is reduced. • The data rate can be lowered to 56 Kbits/s without loss of quality.	• This is prediction-based approach. • The concept of ADPCM is to use the past behavior of a signal to forecast it in the future. • Apply prediction to original samples, the predictor is adapted from one speech frame to the next, quantize the prediction error. • Allows data rate reduction to 32 Kbits/s. • reduces bit flow without compromising quality. • The ADPCM method can be applied to all wave forms, high-quality audio, images and other modern data.

2. Source coding

- This technique is essentially a compression technique for binary data.
- Binary representation of audio signal is coded again to reduce the size.
- Source coding reduces redundancy (repeated data) in the speech signal and thus results in signal compression, which means that a significantly lower bit rate is achieved than needed by the original speech signal.

For example : consider following binary data for a speech.

111111000000001111111100101

Here, we do not need to transmit all the bits . We can encode the data such that there will be a message about how many 1's or 0's are there. e.g., 15!08!16!02!101

- This technique uses of variable-length codes in order to reduce the number of symbols in a message to the minimum necessary to represent the information in the message
- most probable source symbols are represented by the shortest codewords, and the less probable by longer codewords
- The most widely used methods for ensuring this are Huffman coding and Shannon–Fano coding

End of unit 2

Thank you!