

Course material

Course:

Digital Signal Processing II

Video:

2.3.2.b LPC Speech Coding

Concepts (extracted from automatically generated subtitles):

Filter coefficients. Sinusoidal excitation. Speech signals. Coefficients of the autoregressive filter. Pole structure. Usual autocorrelation matrix. Speech generation mechanism. Adaptive signal processing. Linear prediction. Sequence of segments. Coefficients of the autoregressive model. White noise. Set of coefficients. Model. Optimal least squares solution.



[to video sequence search](#)
(within Digital Signal Processing II.)

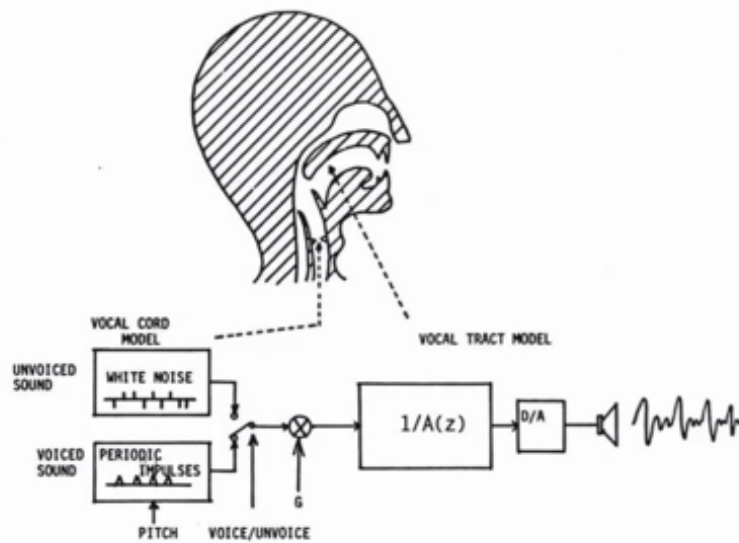


[to video](#)

Center for Digital Education. More educational support material here:

<https://exoset.epfl.ch/booc/>

Example: linear prediction coding of speech



87

Adaptive signal processing is

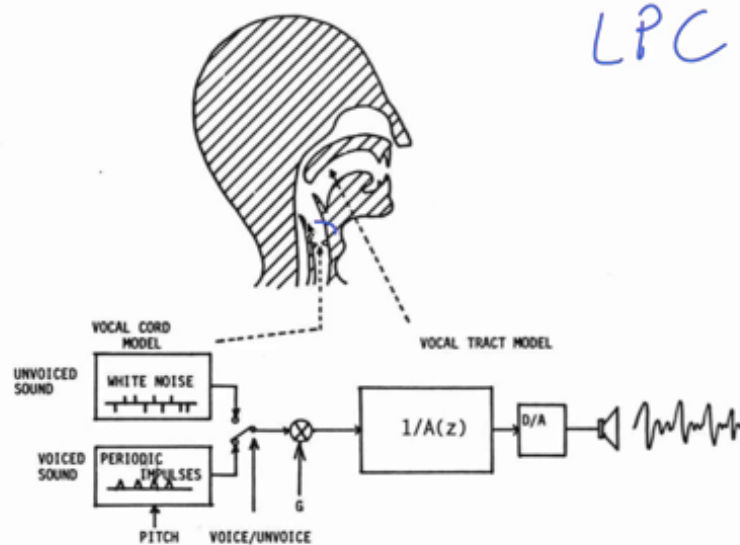
notes

summary

0m 0s



Example: linear prediction coding of speech



87

particularly useful in signal compression. Here we will talk about a classic example which is called linear prediction coding of speech signals. In this model, we assume that the speech generation mechanism is composed of two parts, there's an exciter that can produce

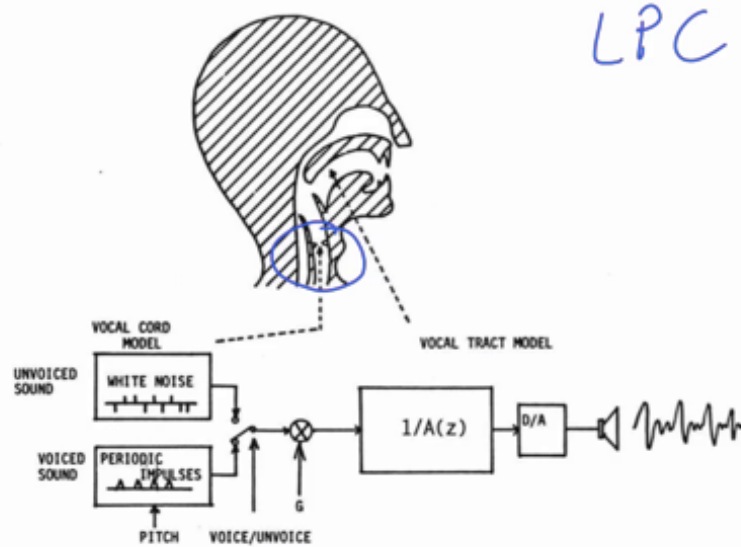
notes

summary

0m 2s



Example: linear prediction coding of speech



87

either a white noise-like signal as in consonants like shh or a sinusoidal excitation as in vowels like aaaa.

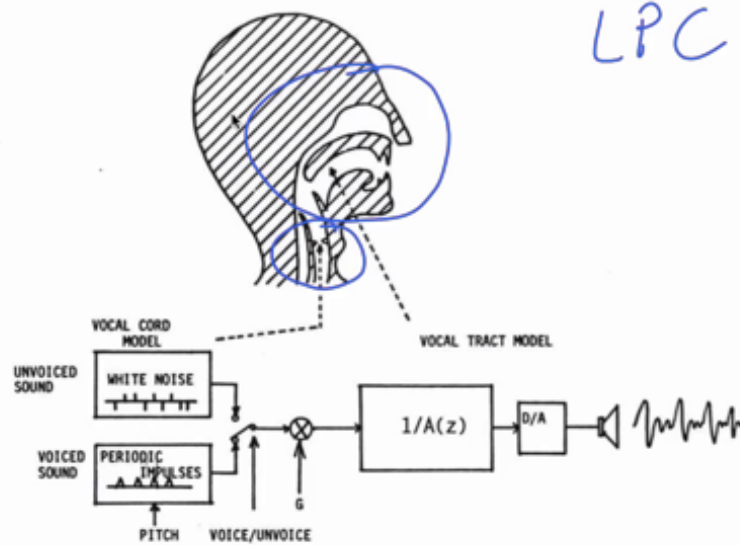
notes

summary

0m 20s



Example: linear prediction coding of speech



87

Then, there's a resonator which is represented by the cavities in the head,

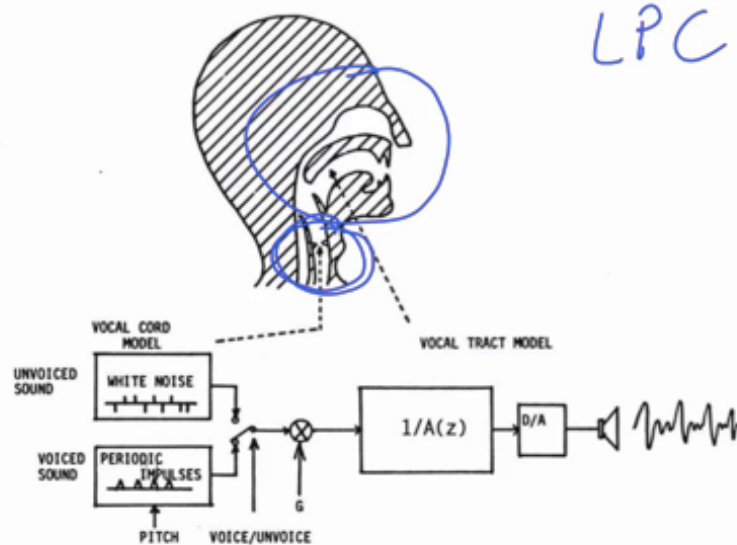
notes

summary

0m 30s



Example: linear prediction coding of speech



87

and we can change the shape of this cavity by changing the shape of our mouth, and therefore change the sound that is produced from the excitations. When you have a resonator, there are some specific modes of the resonator namely, frequency bands, in which the transfer of energy from the excitator to the resonator is particularly good. This structure acts as a filter which is best modeled by an all-pole structure, where each pole takes care of one of these resonance frequencies. A resonator is like the body of a musical instrument like the body of the guitar or a violin and it acts as a filter. It has some resonant frequencies at which the transfer of energy from the source to the resonator is particularly effective. The model for a resonator is an all-pole filter where every pole takes into account the particular resonant frequencies of the resonator. So how does the compression mechanism work?

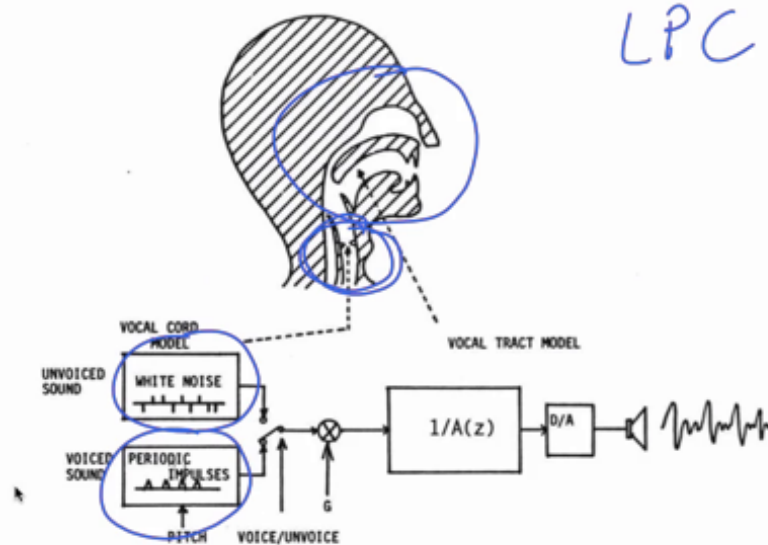
notes

summary

0m 37s



Example: linear prediction coding of speech



87

If we assume that the block diagram that represents the voice production mechanism is like so, so you have two sources either white noise or a sinusoidal impulse that you can switch between,

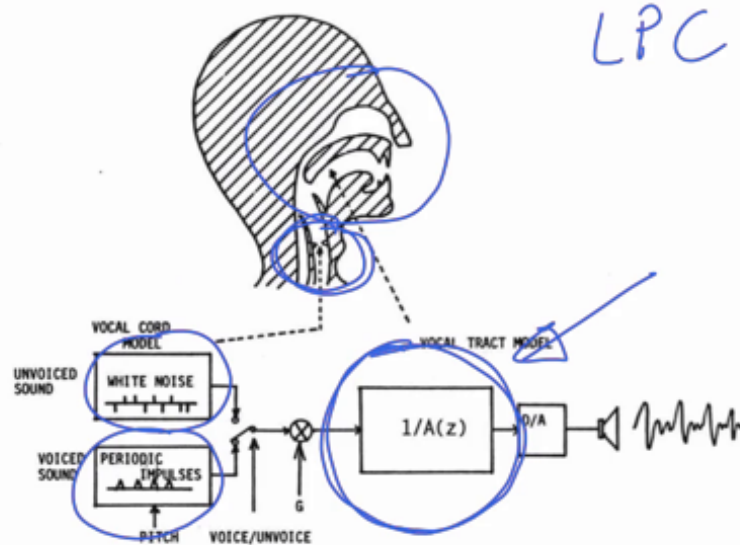
notes

summary

1m 37s



Example: linear prediction coding of speech



87

and then you have a filter that shapes the sound. The idea is to store the speech signal not as a sequence of samples, but as a sequence of segments for which we specify the type of excitation and the filter coefficients. This will be a much more compact representation of a portion of the speech signal.

notes

summary

1m 49s



$$H(z) = \frac{1}{A(z)} = \frac{1}{1 - a_1 z^{-1} - a_2 z^{-2} - \dots - a_N z^{-N}}$$

- ▶ poles model natural resonances of physical systems
- ▶ model is also called autoregressive (output is purely recursive)

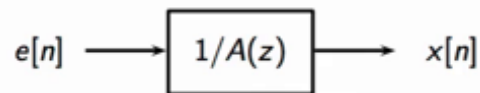
So mathematically, we're going to model our resonator as an all-pole filter like so. This model is also called autoregressive, because as you can see, having only poles mean that it's a purely feedback system.

notes

summary

2m 10s





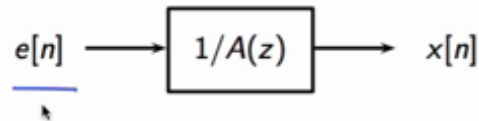
The situation is we have an excitation,

notes

summary

2m 24s





- ▶ $e[n]$: unknown excitation
- ▶ $x[n]$: observable signal
- ▶ can we determine $A(z)$?

89

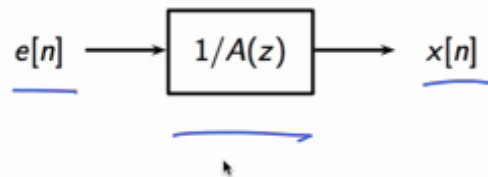
we have the all-pole filter, and then we have the speech signal that we can measure. So this is the only thing we have access to. The excitation is unknown,

notes

summary

2m 25s





- ▶ $e[n]$: unknown excitation
- ▶ $x[n]$: observable signal
- ▶ can we determine $A(z)$?

x of n is the observable signal and A of z is also unknown.

notes

summary

2m 37s



$$\underline{X(z)} = E(z)/A(z)$$

$$E(z) = X(z)A(z)$$

$$e[n] = x[n] - \sum_{k=1}^N a_k x[n-k]$$

So let's see what the relationship between the input and the output is in the z domain.
So the speech signal is the convolution of the excitation with the all-pole filter.

notes

summary

2m 42s



$$X(z) = E(z)/A(z)$$

$$E(z) = X(z)A(z)$$

$$e[n] = x[n] - \sum_{k=1}^N a_k x[n-k]$$

We can write this as being E of z divided by A of z. A of z is the denominator of the transfer function of the resonator. We can now rewrite the excitation as the product of the input times A of z. A of z now is a purely feed-forward filter.

notes

summary

2m 54s



$$X(z) = E(z)/A(z)$$

$$E(z) = X(z)A(z)$$

$$e[n] = x[n] - \sum_{k=1}^N a_k x[n-k]$$

If we go back to the time domain, we can say that the excitation

notes

summary

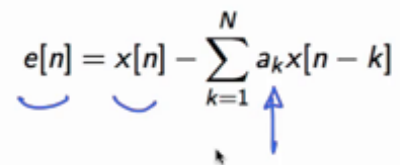
3m 10s



$$X(z) = E(z)/A(z)$$

$$E(z) = X(z)A(z)$$

$$e[n] = x[n] - \sum_{k=1}^N a_k x[n-k]$$



can be written as the current speech sample minus a linear combination of past speech samples where the weights of the linear combination are

notes

summary

3m 18s



91

notes

3m 25s



$$e[n] = d[n] - \sum_{k=0}^{N-1} h[k]x[n-k]$$

minus a linear combination of input samples.

notes

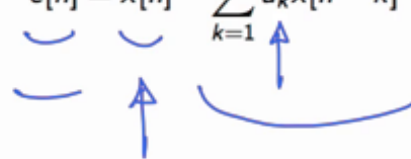
summary

3m 37s



$$X(z) = E(z)/A(z)$$

$$E(z) = X(z)A(z)$$

$$e[n] = x[n] - \sum_{k=1}^N a_k x[n-k]$$


So what this expression is saying, is that the excitation is what remains after you've tried to predict the current speech sample as a linear combination of past speech samples. This makes sense because the excitation does not depend on the filter coefficients. Is what's left after you've taken into account everything that the resonator can predict.

notes

summary

3m 39s



$$e[n] = x[n] - \sum_{k=1}^N a_k x[n-k]$$

(Handwritten blue arrows point to $x[n]$ and $x[n-k]$. A blue bracket is under the entire equation.)

- ▶ we shouldn't be able to predict excitation $e[n]$
- ▶ excitation and prediction should be orthogonal
- ▶ Least Squares solution is *the* solution

So since excitation and prediction should be orthogonal, we can solve this linear prediction problem using Least Squares. Instead of having a desired signal, we have the same signal here and here.

notes

summary

4m 0s



by setting $\partial E [e^2[n]] / \partial a_i$ to zero...

$$\mathbf{R}\hat{\mathbf{a}} = \mathbf{r}$$

$$\begin{bmatrix} r_x[0] & r_x[1] & \dots & r_x[N-1] \\ r_x[1] & r_x[0] & \dots & r_x[N-2] \\ \vdots & \vdots & \ddots & \vdots \\ r_x[N-1] & r_x[N-2] & \dots & r_x[0] \end{bmatrix} \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \\ \vdots \\ \hat{a}_N \end{bmatrix} = \begin{bmatrix} r_x[1] \\ r_x[2] \\ \vdots \\ r_x[N] \end{bmatrix}.$$

But other than that, the math remains the same. So we set the partial derivatives of the mean-square error to zero with respect to the coefficients of the autoregressive model and we obtain an equation that looks like so.

notes

summary

4m 13s



by setting $\partial E [e^2[n]] / \partial a_i$ to zero...

$$\mathbf{R}\hat{\mathbf{a}} = \mathbf{r}$$

$$\begin{bmatrix} r_x[0] & r_x[1] & \dots & r_x[N-1] \\ r_x[1] & r_x[0] & \dots & r_x[N-2] \\ \vdots & \vdots & \ddots & \vdots \\ r_x[N-1] & r_x[N-2] & \dots & r_x[0] \end{bmatrix} \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \\ \vdots \\ \hat{a}_N \end{bmatrix} = \begin{bmatrix} r_x[1] \\ r_x[2] \\ \vdots \\ r_x[N] \end{bmatrix} \cdot \begin{bmatrix} 1 \\ \vdots \\ N \end{bmatrix}$$

R is the usual autocorrelation matrix that we saw before in the Least Squares problem. This duplex matrix, a, is the set of coefficients in the autoregressive model and r is a vector of autocorrelation coefficients where the indices move from one to capital N, N being the number of coefficients in the autoregressive model.

notes

summary

4m 25s



- ▶ segment speech in 20ms chunks (approx. stationary)
- ▶ find the coefficients for an all-pole model
- ▶ inverse filter and find the residual
- ▶ classify the residual excitation as voiced/unvoiced

In practice, linear prediction coding of speech works like so. Speech is not stationary signal per se. When we utter a longer sentence too many things change. But over small segments, we can consider the signal to be stationary. So we split the input into chunks that are approximately 20 milliseconds long, we find the coefficients for

notes

summary

4m 46s



- ▶ segment speech in 20ms chunks (approx. stationary)
- ▶ find the coefficients for an all-pole model
- ▶ inverse filter and find the residual
- ▶ classify the residual excitation as voiced/unvoiced

an autoregressive model for each little segment, we inverse filter the segment with

notes

summary

5m 10s



- ▶ segment speech in 20ms chunks (approx. stationary)
- ▶ find the coefficients for an all-pole model
- ▶ inverse filter and find the residual
- ▶ classify the residual excitation as voiced/unvoiced

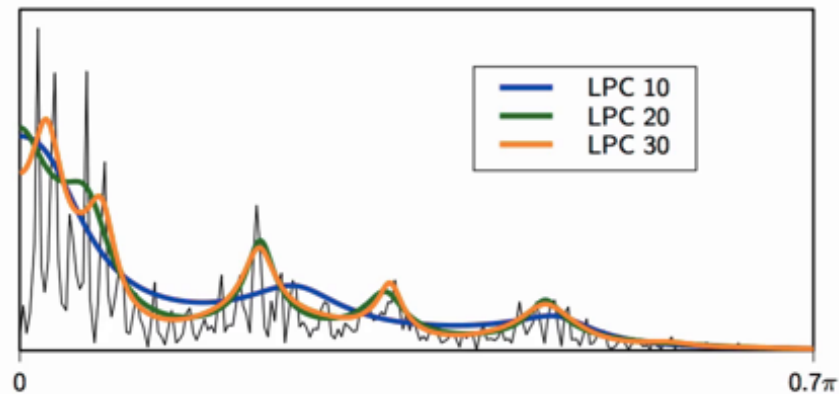
the all-pole filter that we found in order to recover the excitation, then we classify the residual either as white noise or as a periodic excitation as in the case of vowels. Then we send this coefficients over in a compact format.

notes

summary

5m 13s





95

Let's look at an example. For instance, this is a small speech segment, clearly a voiced speech segment because you see all these peaks that correspond to the harmonics of a periodic excitation, and you see a shape here in the spectrum that is given by the resonances in the vocal tract. Now we can try and model these resonances with all-pole filter. If we use 10 coefficients, we get this blue line here that tries to capture this overall envelope of the spectrum. We can increase the order of the LPC model to 20 coefficients and you see that now the model manages to capture smaller details in the shape of the spectrum. If we increase the order of the model even more, the fit is even better. But then of course we try to lose the advantage of sending a compact representation because we have too many coefficients to transmit.

notes

summary

5m 30s



- ▶ normally $N = 20$
- ▶ average bitrate 4Kbit/sec (raw data: 48Kbit/sec)
- ▶ many improvements exist: CELP & Co

So in general, we use 20 LPC coefficients to model speech segments of about 20 milliseconds. With this coding, the average bit rate for a speech signal is about four kilobits per second. This should be compared to the data rate of the raw samples which is about 48 kilobits per seconds. So there's an order of magnitude decrease in bandwidth or storage that is required to store LPC coded speech. There are also many improvements that build on this idea to improve the quality. As a matter of fact, cellular phones use CELP which is a variation of LPC to transmit speech for voice communication. voice communication.

notes

summary

6m 22s

