Sprachverarbeitung – 3.Woche

Andreas Wendemuth

Letzte Woche:

1.Overview Speech Recognition Systems &Architectures

2. Acoustic modeling & feature extraction (1)

3. Feature extraction (2)

4. Klassifikation in HMM Modellen

5. Wortmodellierung (trigramme, tying)

6. search/decoding, lattices,
   wordgraphs, confidence measures

7. acoustic adaptation

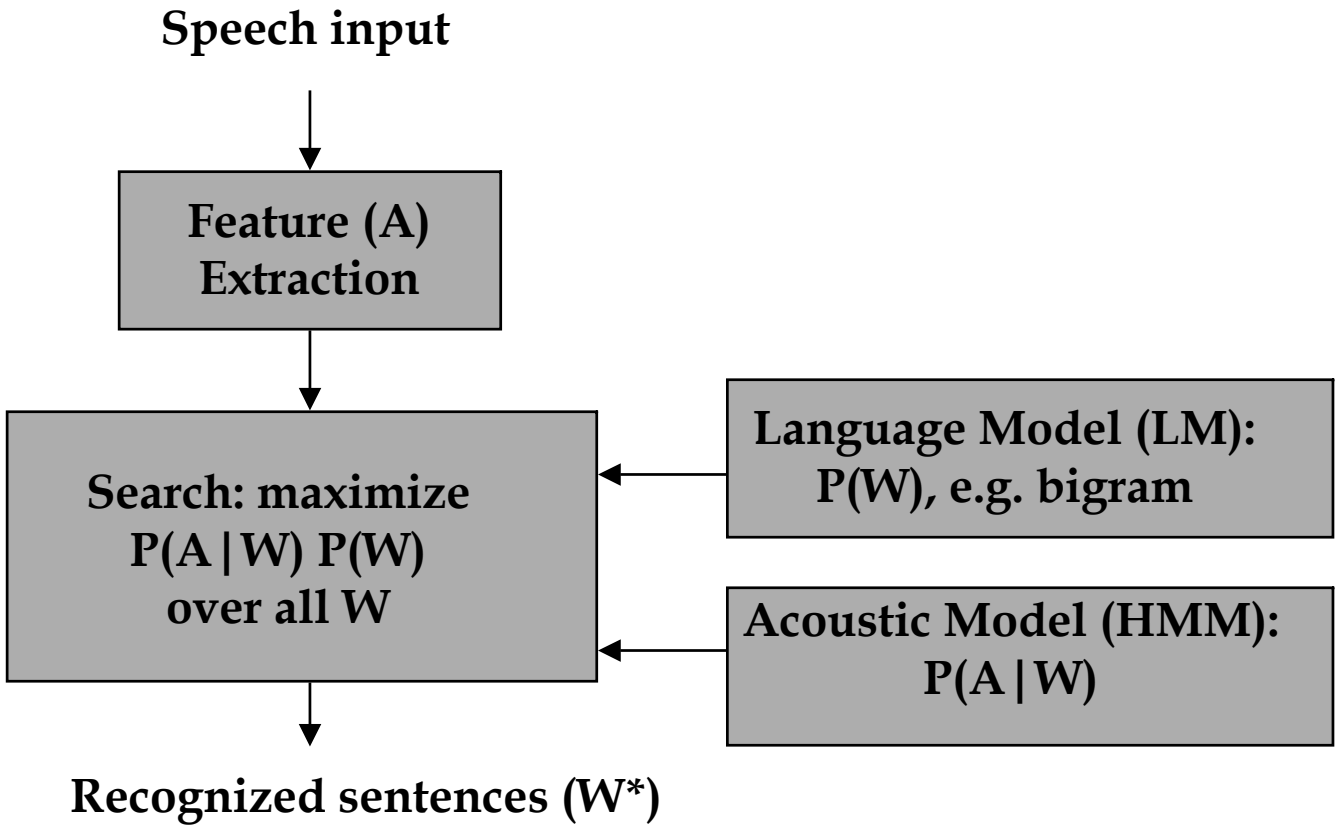
8. language models and grammars, Language model
   adaptation, lexica, phonology

9. speech understanding, dialogue control

10. Design of computer speech recognition systems

SR Architecture revisited

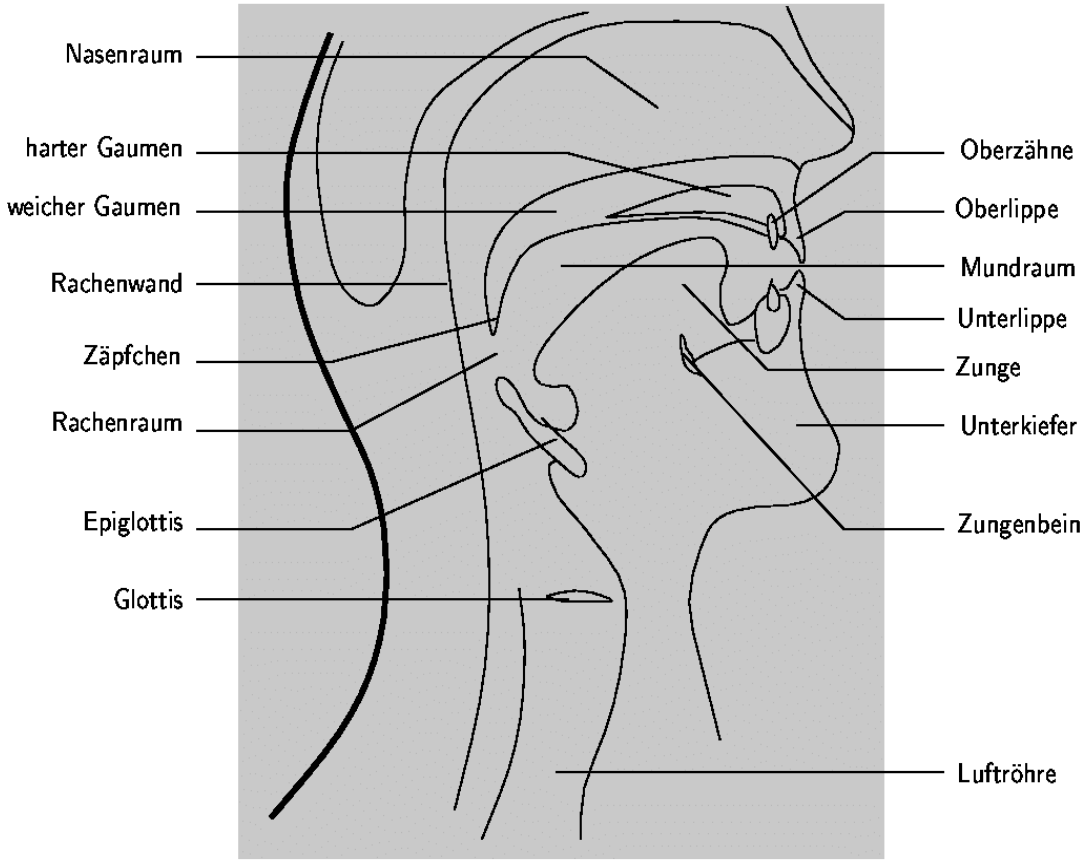
W = a word sequence (e.g. word/ sentence/ whole dictation)
A = an acoustic feature vector sequence (the input for the recognizer)



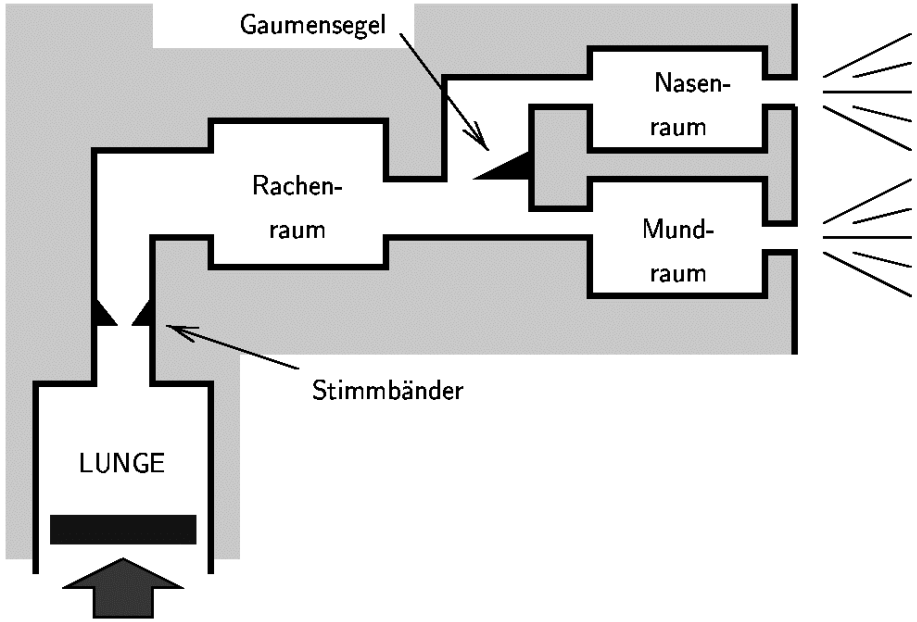
Acoustic modeling & feature extraction

Artikulatorische Phonetik

Die menschlichen Artikulationsorgane



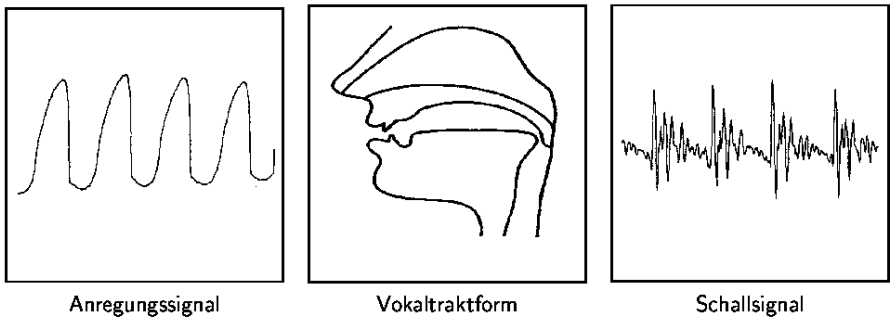
Akustische Sprachproduktion



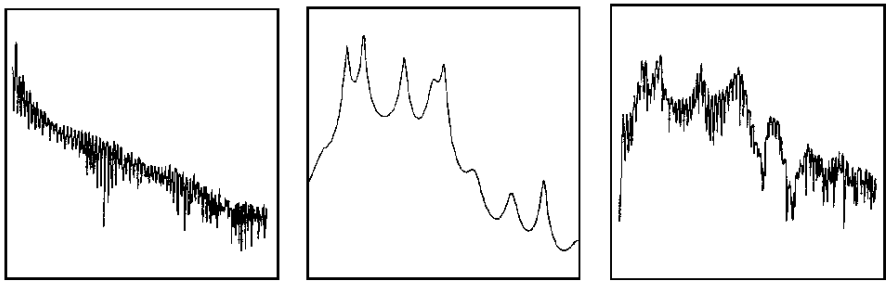
- ***Schallanregung an der Glottis***
  Periodische Stimmbandschwingung
- ***Zeitliche Veränderung der Vokaltraktform***
  Formanten ⇔ Resonanzen des Vokaltrakts
- ***Verluste an den Vokaltraktwänden***
  Elastizität, Wärmeleitung, Flüssigkeitenverformung, Reibungseffekte
- ***Zuschaltung des Nasenraums***
  Antiformanten ⇔ Antiresonanzen des Nasaltrakts
- ***Abstrahlung des Schalls von den Lippen***
  Hochpassfilter

Gesamtübertragungsfunktion

$$F(z) \;=\; \sigma \cdot G(z) \cdot V(z) \cdot R(z)$$



Anregungssignal | Vokaltraktform | Schallsignal



Anregungsspektrum | AR-Modellspektrum | Signalspektrum

Autoregressives Modell (Allpol-System)

$$H(z) \;\approx\; \sigma \;/\; A(z) \qquad \text{mit } A = \sum_{m=0}^{M} a_m z^{-m}$$

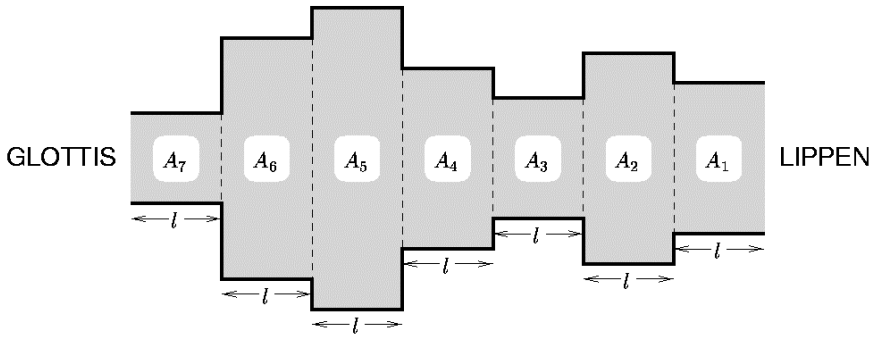
Rationales Modell (ARMA-System)

$$H(z) \;\approx\; B(z) \;/\; A(z) \qquad \text{mit } A = \sum_{m=0}^{M} a_m z^{-m},\; B = \sum_{m=0}^{N} b_m z^{-m}$$

Die Nullstellen von $B(z)$ heißen ***Antiformanten***.

Das Vokaltraktmodell

Verlustfreie akustische Röhre mit gleichlangen Zylinderabschnitten mit den Querschnittflächen $A_1, A_2, \ldots, A_M$



Reflexionskoeffizienten

(Schallfluß vor- und rücklaufender Wellen)

$$k_i = \frac{A_i - A_{i+1}}{A_i + A_{i+1}}, \qquad i = 0, \ldots, M$$

Übertragungsverhalten bei bandbegrenztem Glottissignal:

$$V(z) = \frac{\prod_{i=0}^{M}(1 + k_i)}{1 - \sum_{i=1}^{M} a_i z^{-i}}$$

Diese Woche:

1.Overview Speech Recognition Systems &Architectures
2. Acoustic modeling & feature extraction (1)
3. Feature extraction (2)
4. Klassifikation in HMM Modellen
5. Wortmodellierung (trigramme, tying)
6. search/decoding, lattices,
   wordgraphs, confidence measures
7. acoustic adaptation
8. language models and grammars, Language model
   adaptation, lexica, phonology
9. speech understanding, dialogue control
10. Design of computer speech recognition systems

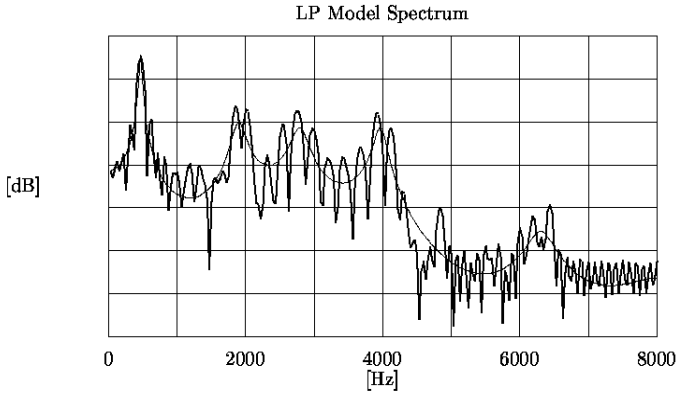
Spektrale Dekonvolution

Akustisches Sprachproduktionsmodell:

$$\begin{aligned} f(t) &= e(t) \star h(t) && (\text{„Zeitbereich“}) \\ F(z) &= E(z) \star H(z) && (\text{„}z\text{-Ebene“}) \end{aligned}$$

Spektrale Signaleigenschaften:

- ***Neigung***
- ***Formantstruktur***
- ***Harmonische Struktur***



Spektrale Glättung:

- *Filterbank* — Bandpassenergien; nichtlineares System
- *Cepstrum* — Tiefpaßgefiltertes Spektrum $\log |F(e^{i\phi})|^2$
- *Lineare Vorhersage* — komplexe Polynomapprox. von $F(z)^{-1}$

Cepstrum

Die Idee der spektralen Dekonvolution

$$\begin{aligned} f_n &= e_n \star h_n \\ \mathrm{FT}\{f_n\} &= \mathrm{FT}\{e_n\} \cdot \mathrm{FT}\{h_n\} \\ \log \mathrm{FT}\{f_n\} &= \log \mathrm{FT}\{e_n\} + \log \mathrm{FT}\{h_n\} \\ \mathrm{FT}^{-1}\{\log \mathrm{FT}\{f_n\}\} &= \mathrm{FT}^{-1}\{\log \mathrm{FT}\{e_n\}\} + \mathrm{FT}^{-1}\{\log \mathrm{FT}\{h_n\}\} \end{aligned}$$

Komplexes und reelles Cepstrum

$$\mathrm{FT}^{-1}\{\log \mathrm{FT}\{f_n\}\} \quad \text{und} \quad \mathrm{FT}^{-1}\{\log |\mathrm{FT}\{f_n\}|\}$$

Berechnung des reellen Cepstrums

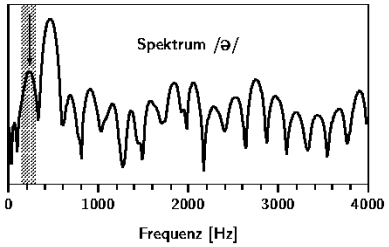
$$c_q^{(m)} = \mathrm{DFT}^{-1}\{\log \mathrm{DFT}\{f_n\}\} = \frac{1}{N} \sum_{\nu=0}^{N-1} \log |F_\nu^{(m)}| e^{i2\pi\nu q/N}, \quad q = 0, \ldots, N-$$

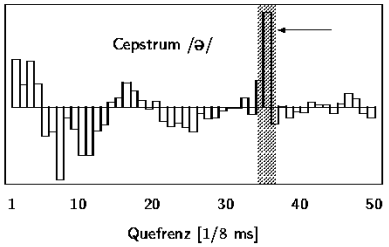
Diskrete Kosinustransformation (DCT)

$$\begin{aligned} c_0^{(m)} &= \sqrt{2/N} \sum_{\nu=0}^{N/2-1} \log |F_\nu^{(m)}| \\ c_q^{(m)} &= \sqrt{4/N} \sum_{\nu=0}^{N/2-1} \log |F_\nu^{(m)}| \cos \frac{\pi q(2\nu+1)}{N} \quad \text{für } q = 1, \ldots, N/2 \end{aligned}$$

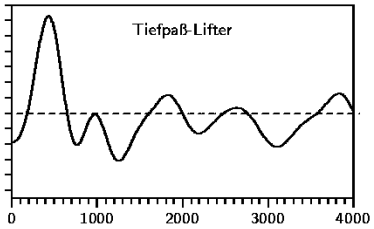
denn Betragsspektrum und Cepstrum sind reell und symmetrisch ...

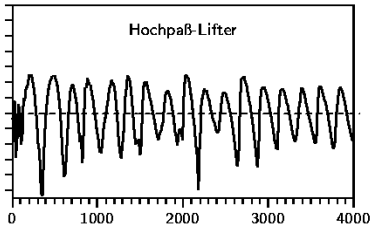
Homomorphe Analyse (Cepstrum)





- Cepstrum $\Leftrightarrow$ Spektrum des logarith. Betragsspektrums
- ***Quefrenz*** = Periodendauer in Sekunden (Einheit ist $^1/_{f_A}$)
- ***Cepstralgipfel*** markiert die Dauer der Grundperiode
- ***Formantstruktur*** = niedrige Cepstralkoeffizienten





- ***Lifterung*** — Rücktransformation in den Spektralbereich

$$\{\hat{C}_\nu^{(m)}\} \;=\; \mathrm{DFT}\{\hat{c}_q^{(m)}\}$$

- Grobstruktur: $\hat{c}_q^{(m)} = 0$ für $q > 20$ (Tiefpaß)
- Feinstruktur: $\hat{c}_q^{(m)} = 0$ für $q < 20$ (Hochpaß)

Z-Transformation (Einschub)

Z transforms

The *z*-transform is defined by:

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n} \qquad (43)$$

The sequence, $x(n)$ is known and z is a complex number. Hence $X(z)$ is just a weighted sum. For example, for the sequence: $x(0) = 1$, $x(1) = 3$, $x(2) = 3$, $x(3) = 1$ and $x(n) = 0$ otherwise

$$X(z) = 1 + 3z^{-1} + 3z^{-2} + z^{-3} \qquad (44)$$

and evaluating this at a particular point, e.g. $z = i/2$

$$\begin{aligned} X(i/2) &= 1 + 3(i/2)^{-1} + 3(i/2)^{-2} + (i/2)^{-3} && (45) \\ &= 1 + 3(-2i) + 3(-4) + (8i) && (46) \\ &= -11 + 2i && (47) \end{aligned}$$

Only defined for values of z where the series converges.

That is, *z*-transform is the general version of the discrete Fourier transform. To obtain the Fourier restrict z to lie on the unit circle $z = e^{i\theta} = \cos\theta + i\sin\theta$.

There are several ways of obtaining the inverse *z* transform:

a) By inspection: if $X(z)$ can be written as a simple polynomial in z then the time domain sequence is the coeffients of the polynomial

b) By expansion: expanding $X(z)$ as a polynomial in z

c) By decomposition: breaking up $X(z)$ into parts whose inverse z transforms are known (e.g. see table 3.1 in [4])

d) By definition: the inverse transform is defined by:

$$x(n) = \frac{1}{2\pi i} \oint_C X(z) z^{n-1} dz \qquad (48)$$

Where C is a closed contour that includes $z = 0$.

The z transform is a linear transform, i.e.

$$y(n) = \alpha x_1(n) + \beta x_2(n) \qquad (49)$$

$$Y(z) = \alpha X_1(z) + \beta X_2(z) \qquad (50)$$

So, if $y(n)$ is the convolution of two signals, $h(n)$ and $x(n)$, i.e.:

$$y(n) = \sum_{k=-\infty}^{\infty} x(k) h(n-k) \qquad (51)$$

then

$$= x(n) * h(n) \qquad (52)$$

$$Y(z) = H(z) X(z) \qquad (53)$$

The linear filters of section 2.2 can now be expressed in terms of *z*-transforms.

The general linear filter is expressed as:

$$Y(z) = H(z)X(z) \qquad (53)$$

where $H(z)$ is called the ``system function" and is the *z*-transform of the unit sample response.

For the FIR filter of order q:

$$y(n) = \sum_{r=1}^{q} b_r x_{n-r} \qquad (54)$$

$$Y(z) = H(z)X(z) \qquad (55)$$

$$H(z) = \sum_{r=0}^{q} b_k z^{-k} \qquad (56)$$

Similarly for the IIR filter:

$$H(z) = \frac{\sum_{r=0}^{q} b_r z^{-r}}{\sum_{k=0}^{p} a_k z^{-k}} \qquad (57)$$

This is useful as $H(z)$ can be factored:

$$H(z) = \frac{A \prod_{r=1}^{q}(1 - c_r z^{-1})}{\prod_{k=1}^{p}(1 - d_k z^{-1})} \qquad (58)$$

From this equation it can be seen that if $(1 - c_r z^{-1}) = 0$ then the filter will have zero response - these are the ``zeros" of the linear system.

Similarly, $(1 - d_k z^{-1}) = 0$ defines the ``poles" of the linear system.
When $q = 0$, as in linear prediction, we have an ``all pole" filter.

For a stable system, all the poles must lie within the unit circle.

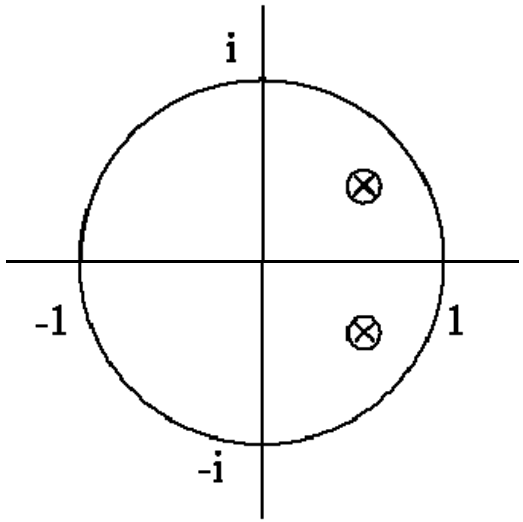


Figure 34: An argand diagram showing a stable pole-pair within the unit circle

An unstable system is one whose output is unbounded in response the unit impulse.
Manipulation of the form of $H(z)$ allows many different implementations. For example, as the coefficients a_i and b_i are real, the poles and zeros occur in complex conjugate pairs. By grouping these together $H(z)$ can be expressed in terms of second order sections:

$$H(z) = A \prod_{k=1}^{P} \frac{1 + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}} \qquad (59)$$

This ``cascade form'' is illustrated in figure 35.

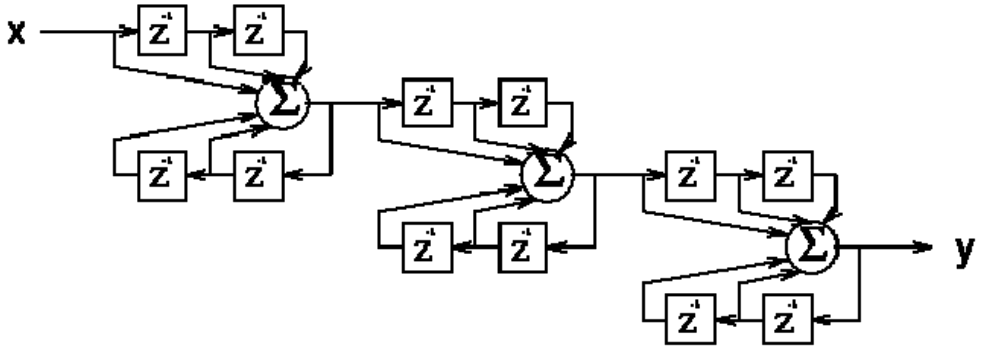


Figure 35: The cascade form for a linear filter

It is also possible to expand $H(z)$ in terms of partial fractions:

$$H(z) = \sum_{k=1}^{p} \frac{c_{0k} + c_{1k} z^{-1}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}} \qquad (60)$$

This ``parallel form" is illustrated in figure 36.

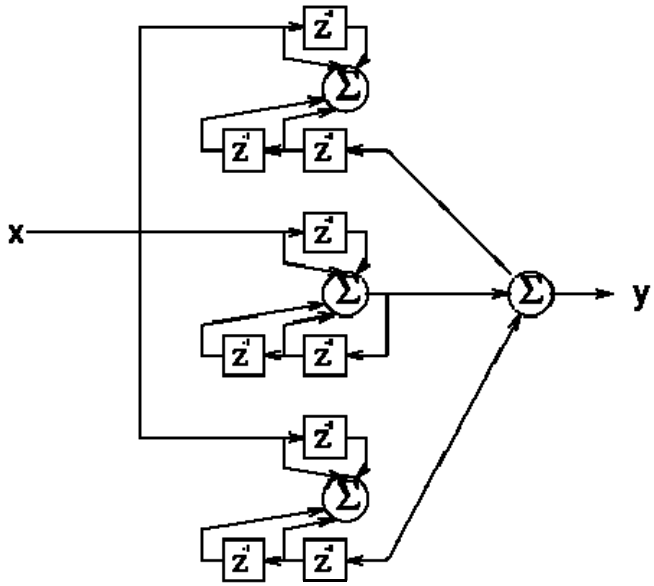


Figure 36: The parallel form for a linear filter
Both forms are popular in speech synthesis - indeed the Klatt synthesiser has both a parallel and a cascade path (for ease of specifying the coefficients I assume).

Finite Impulse Response (FIR) Filters

A Finite Impulse Response (FIR) filter produces an output, y(n), that is the weighted sum of the current and past inputs, x(n).

$$y_n = b_0 x_n + b_1 x_{n-1} + b_2 x_{n-2} + \ldots + b_q x_{n-q} \quad (1)$$

$$= \sum_{j=0}^{q} b_j x_{n-j} \quad (2)$$

This is shown in figure 4 with z^{-1} representing a unit delay.

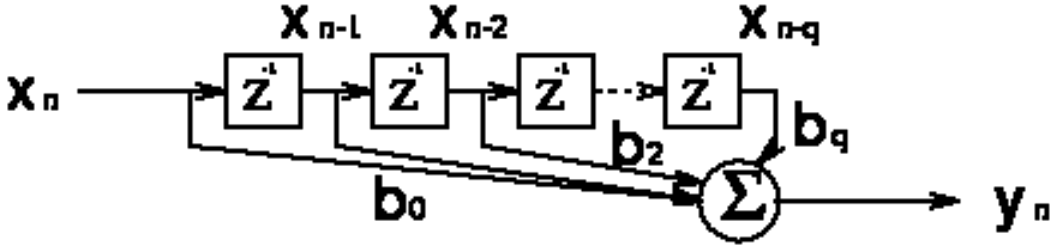


Figure 4: A FIR filter

Infinite Impulse Response (IIR) Filters

An Infinite Impulse Response (IIR) filter produces an output, y(n), that is the weighted sum of the current and past inputs, x(n), and past outputs. The Linear Predictive model is a specical case of an IIR filter and shown in figure 7.

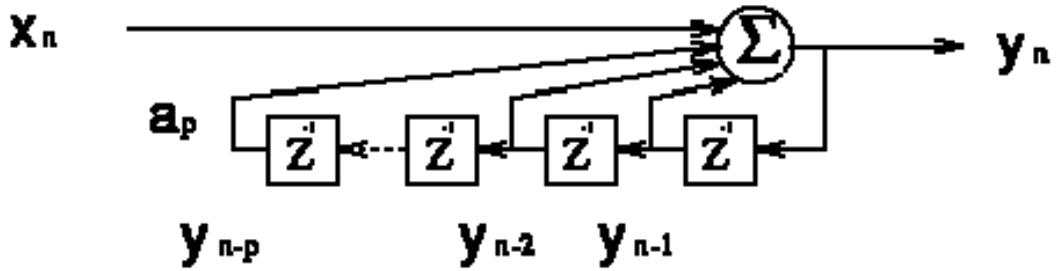


Figure 7: An IIR filter

The general IIR filter (figure 8) is given by:

$$y_n = \sum_{i=1}^{p} a_i y_{n-i} + \sum_{j=0}^{q} b_j x_{n-j} \qquad (11)$$

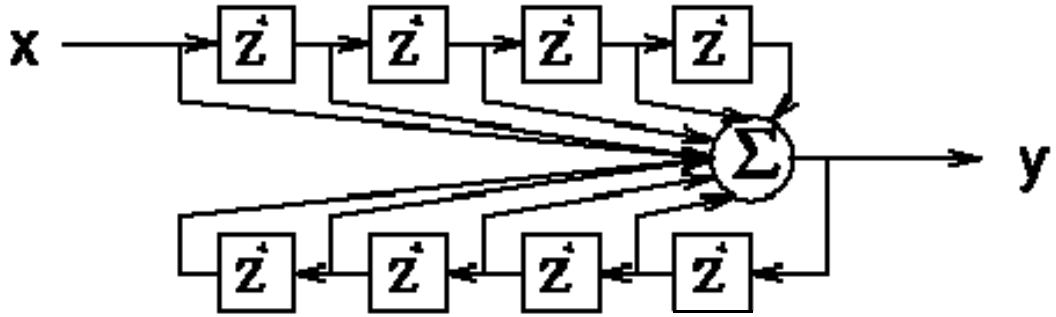


Figure 8: The general linear filter

If p = 0 then the system represents a finite impulse response (FIR) filter. If p is not zero, then the system is an infinite impulse response (IIR) filter.

An example is the two pole resonator with center frequency ω and bandwidth related to r is:

$$y_n = 2r\cos(\omega T)y_{n-1} - r^2 y_{n-2} + x_n - \cos(\omega T)x_{n-1} \qquad (12)$$

Common types of IIR filter:

Type	Characteristics
Butterworth	maximally flat amplitude
Bessel	maximally flat group delay
Chebyshev	equiripple in passband or stop-band
Elliptic	equiripple in passband and stop-band

Lineare Vorhersageformel, Ordnung p

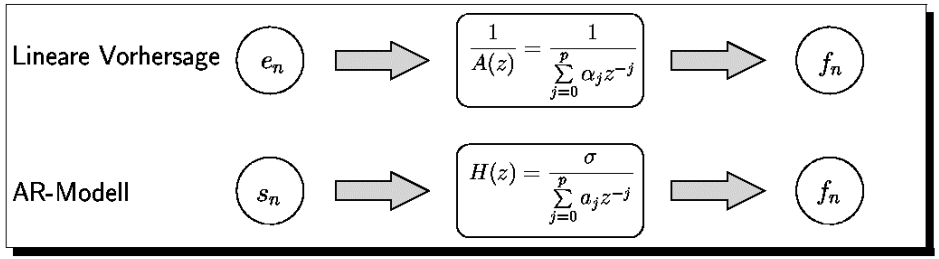
$$\hat{f}_n \;=\; -\sum_{j=1}^{p} \alpha_j f_{n-j}$$

Vorhersagefehler mit den Prädiktionskoeffizienten α_j

$$e_n \;=\; f_n - \hat{f}_n \;=\; \sum_{j=0}^{p} \alpha_j f_{n-j} \qquad (\alpha_0 = 1)$$

Nach z-Transformation ergeben sich die Systemgleichungen

$$E(z) \;=\; F(z) \cdot A(z) \quad \text{und} \quad F(z) \;=\; E(z) \cdot \frac{1}{A(z)}\,,$$



Wenn die Vorhersagekoeffizienten α_j mit den Modellparametern a_j übereinstimmen:

$$e_n \;=\; \sigma s_n \qquad \text{(Fehler = Verstärkung} \times \text{Anregung)}$$

Bestimmung der Vorhersagekoeffizienten:

Minimiere den akkumulierten **quadratischen Fehler**

$$\varepsilon \;=\; \sum_{n=n_0}^{n_1} e_n^2 \;=\; \sum_{n=n_0}^{n_1} \left(\sum_{j=0}^{p} \alpha_j f_{n-j} \right)^2$$

Kompaktere Schreibweise (quadratische Form)
durch geeignete $p \times p$-Matrix $\mathbf{\Phi}$:

$$\varepsilon \;=\; \sum_{j=0}^{p} \sum_{k=0}^{p} \alpha_j \phi_{jk} \alpha_k \quad \text{mit} \quad \phi_{jk} \;=\; \sum_{n=n_0}^{n_1} f_{n-j} f_{n-k}$$

Nullsetzen der partiellen Ableitungen:

$$\partial \varepsilon / \partial \alpha_k \;=\; 2 \sum_{j=0}^{p} \alpha_j \phi_{jk}$$

⇨ lineares Gleichungssystem in den Variablen $\alpha_1, \ldots, \alpha_p$

$$\sum_{j=1}^{p} \alpha_j \phi_{jk} \;=\; -\phi_{0k} \;, \quad k = 1, \ldots, p$$

Die **Kovarianzmethode**
summiert in den Grenzen $n_0 = m + p$, $n_1 = m + N - 1$

$$\phi_{jk}^{(m)} = \sum_{n=m+p}^{m+N-1} f_{n-j} f_{n-k} = \sum_{n=p}^{N-1} f_{m+n-j} f_{m+n-k}$$

Cholesky-Zerlegung der symmetrischen Matrix $\mathbf{\Phi}^{(m)}$

⇨ $(p^3 + 3p^2 - 4p)/6$ Punktoperationen je Zeitfenster

Die **Autokorrelationsmethode**
summiert das „ausgestanzte“ Kurzzeitsignal:

$$\phi_{jk}^{(m)} = r_{|j-k|}^{(m)} \quad \text{(Kurzzeitautokorrelationskoeffizienten)}$$

Die **Durbinrekursion** erfordert nur $O(p^2)$ Punktoperationen:

$$\begin{aligned} k_n &= \frac{1}{\varepsilon_{n-1}} \cdot \sum_{j=0}^{n-1} \alpha_j^{(n-1)} r_{|n-j|} \\ \varepsilon_n &= \varepsilon_{n-1} \cdot (1 - k_n^2) \\ \alpha_j^{(n)} &= \begin{cases} 1 & j = 0 \\ \alpha_j^{(n-1)} - k_n \alpha_{n-j}^{(n-1)} & 1 \le j \le n \\ 0 & j = n+1 \end{cases} \end{aligned}$$

Initialisierung $\varepsilon_0 = r_0$, Vorhersagekoeffizienten $\left\{\alpha_1^{(n)}, \ldots, \alpha_n^{(n)}\right\}$, quadratischer Fehler ε_n

Das Modellspektrum

Verknüpfen von Vorhersage- und Produktionsmodell durch die idealisierte Annahme

$$H(z) \;=\; \sigma / A(z)$$

Querschnittflächen A_j aus partieller Korrelation (PARCOR) k_n:

$$A_{j+1} \;=\; \frac{1-k_j}{1+k_j} \cdot A_j$$

Schätzung des **Vokaltraktfrequenzgangs**

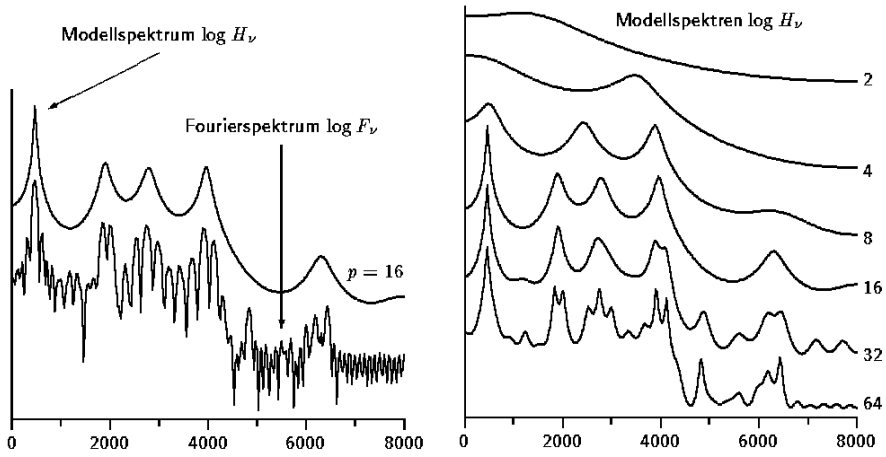
$$H_\nu \;=\; H(e^{2\pi i\nu/N}) \;=\; \frac{\sigma}{A(e^{2\pi i\nu/N})} \;=\; \frac{\sigma}{A_\nu}$$

Funktionswerte A_ν von $A(z)$ auf dem Einheitskreis durch

$$\mathrm{DFT}\{1, \alpha_1, \ldots, \alpha_p, \underbrace{0, \ldots, 0}_{N-p-1 \text{ Nullen}}\}$$

Verstärkungsfaktor aus Autokorrelationsfunktion

$$\sigma^2 \;=\; \sum_{j=0}^{p} \alpha_j r_j$$



Lattice Filter Implementation

Direct implementation of the IIR filter can lead to instabilities if a_i is quantised. The filter is stable provided $0 < k_i < 1$ - hence k_i can be quantised, the result is guaranteed to be stable.

We can either convert back from k_i to a_i or implement the IIR filter as a lattice and use the values directly - useful if working on a limited precision DSP chip (e.g. a GSM phone).

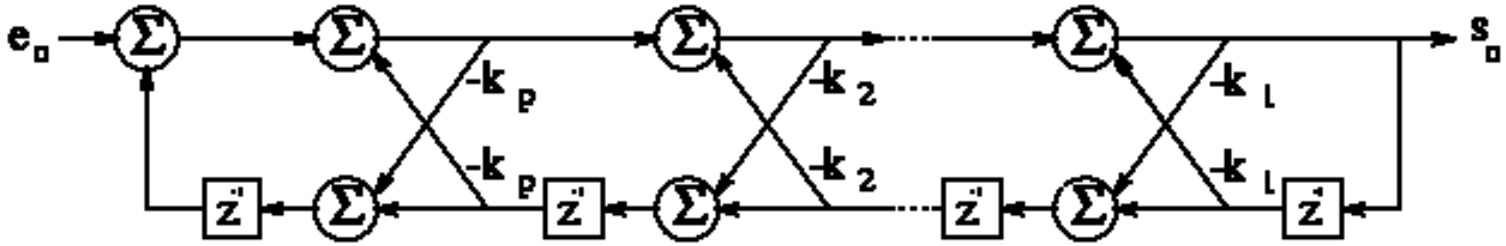


Figure 40: The lattice filter

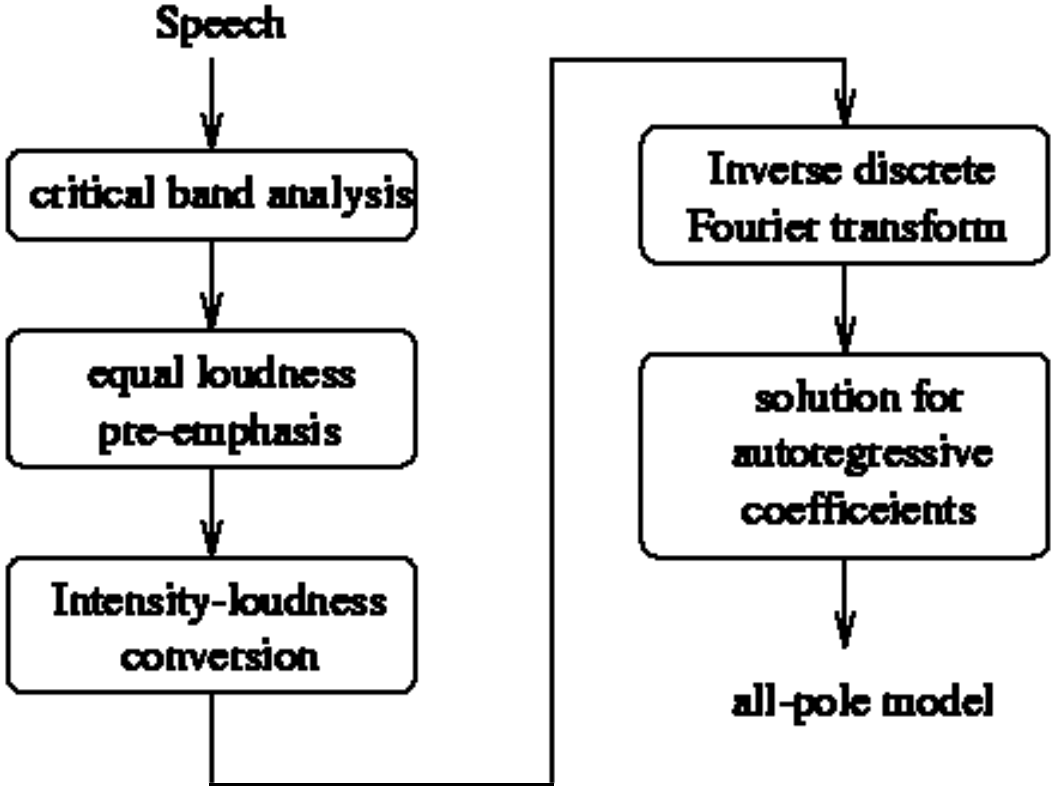
This is analogous with the lossless tube model:

each filter section is one section of the tube
The forward wave is partially reflected backwards
The backward wave is partially reflected forwards

Hence the terminology of k_i

Perceptual Linear Prediction (PLP)

A combination of DFT and LP techniques is perceptual linear prediction (PLP).



Pre-Emphasis

The LP filter so far presented attempts to fit an all-pole model using the least-mean-squares distance metric.

The lower formants contain more energy and therefore are preferentially modeled with respect to the higher formants

A pre-emphasis filter,

$$s'_n = s_n - \alpha_1 s_{n-1} \qquad (82)$$

is often used to boost the higher frequencies. Typically , $0.96 \leq \alpha_1 \leq 0.99$ **or the optimal pre-emphasis** $\alpha_1 = r_1/r_0$ **is used.**

If reconstructing the speech the inverse filter should be used:

$$s_n = s'_n + \alpha_1 s_{n-1} \qquad (83)$$

Intensity-Loudness

Perceived loudness, $I(\omega)$, is approximately the cube root of the intensity,

$$L(\omega) = I(\omega)^{1/3} \qquad (113)$$

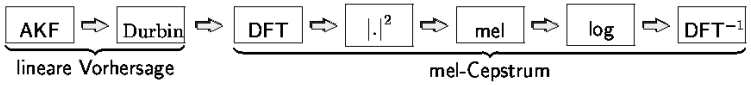
Not true for very loud or very quiet sounds

A reasonable approximation for speech

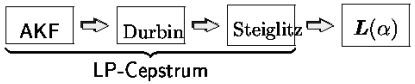
Gehörrichtige Frequenzverzerrung

Integration der Mel-Skalierung in die Berechnung der Vorhersagemerkmale

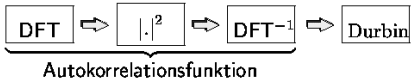
LP-mel-Cepstrum:



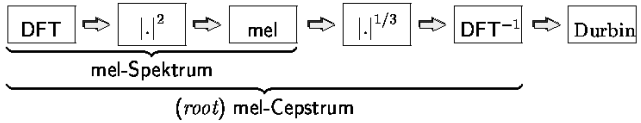
Bilineartransformiertes LP-Cepstrum:



Lineare Vorhersage:



Perzeptuelle lineare Vorhersage (PLP):



Bemerkung 3.4

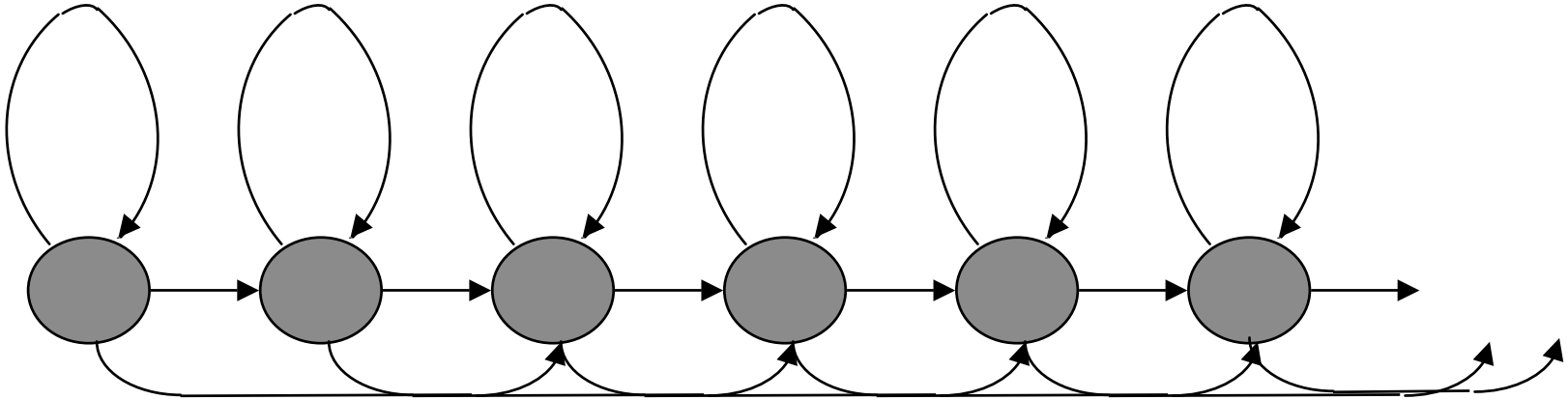
- $L(\alpha)$ ist eine geeignete ***Bilineartransformation*** zur Frequenzverzerrung im Cepstralbereich
- PLP nach dem Satz von Wiener & Khintchine:

$$\mathrm{FT}\{r_n\} = |\mathrm{FT}\{f_n\}|^2 \quad \text{bzw.} \quad \{r_n\} = \mathrm{FT}^{-1}\{|\mathrm{FT}\{f_n\}|^2\}$$

Nächste Woche:

1.Overview Speech Recognition Systems &Architectures
2. Acoustic modeling & feature extraction (1)
3. Feature extraction (2)
4. Klassifikation in HMM Modellen
5. Wortmodellierung (trigramme, tying)
6. search/decoding, lattices,
   wordgraphs, confidence measures
7. acoustic adaptation
8. language models and grammars, Language model
   adaptation, lexica, phonology
9. speech understanding, dialogue control
10. Design of computer speech recognition systems

Acoustic Modeling



Hidden Markov Model (HMM)

each **state** carries a probability density function