Predictive coding of media signals

Prof.dr.sc. Davor Petrinović,

Linear prediction

prof.dr.sc. Davor Petrinović

Process prediction

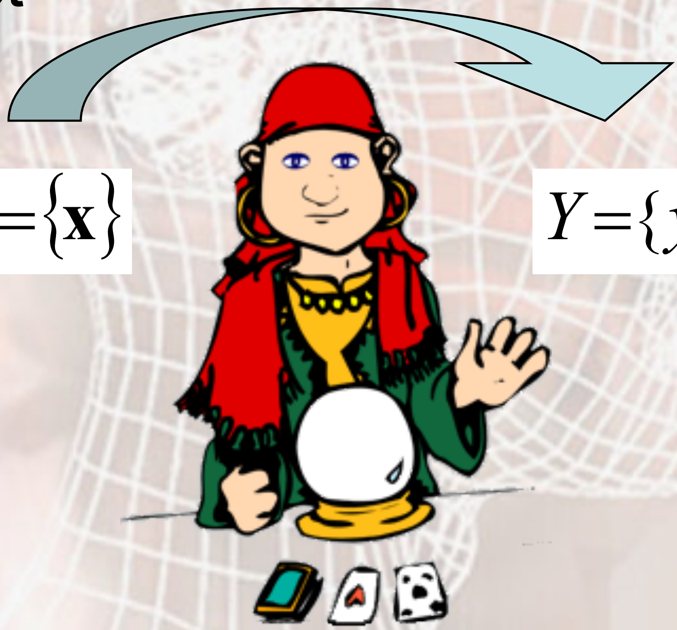
- Prediction task:
  - Based on the known vector process ***X***, it is necessary to predict the scalar process *Y*.
  - The basic assumption is that there are certain statistical dependencies between the individual components of process ***X*** and target process *Y*.
  - The predictor must be able to identify and model such statistical dependencies.
  - With the use of predictors, the description of process *Y* will be much simpler, because if we know process ***X***, most of the information contained in process *Y* can be "guessed" by prediction from process ***X***, and only possible differences in relation to the actual value must be coded!

Process prediction

– Let process ***X*** be a vector process of dimension *p*, ***X***={***x***}, and *Y* be the target scalar process *Y*={*y*} we want to predict

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}$$

$$\mathbf{X} = \{\mathbf{x}\}$$



$$Y = \{y\}$$

– Unfortunately, in practice the prediction is always approximate, so we denote it with a label : $\widetilde{Y} = \{\widetilde{y}\}$

Prediction error

- The goal of a good predictor is to make the process prediction $\tilde{Y}=\{\tilde{y}\}$ as close as possible to the target process $Y=\{y\}$ i.e., to minimize the prediction error process $E=\{e\}$ where: $e=y-\tilde{y}$
- In this sense, it is necessary to introduce a measure of "similarity", and in practice for this purpose ordinary Euclidean distances are most often used, i.e., by using variance of process *E*:

$$\sigma_e^2=E[e^2]=E[(y-\tilde{y})^2]$$

- ... assuming that the expectation of prediction error is zero, $E[e]=0$ (unbiased prediction).

Predictor

- The predictor can be realized in a variety of ways, depending on the type of statistical dependencies we are trying to exploit.
- Most generally, ... the predictor can be constructed from a **joint (or conditional) probability density function** of the original vector process ***X*** and the target process *Y*.
- In practice, a **linear predictor** is most commonly used that exploits the linear dependencies between the source and target processes described by **correlations**.

Linear predictor

- The linear predictor achieves the prediction procedure by simple scalar multiplication of the original process ***X*** with the predictor α:

$$\tilde{y} = \begin{bmatrix} \quad \boldsymbol{\alpha}^{\mathsf{T}} \quad \end{bmatrix} \cdot \begin{bmatrix} \\ \mathbf{x} \\ \\ \end{bmatrix}$$

  - Predictor α is selected to minimize the variance of the prediction error:

$$\overline{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}}(\sigma_e^2(\boldsymbol{\alpha})) \qquad \sigma_e^2(\boldsymbol{\alpha}) = E[(y - \boldsymbol{\alpha}^{\mathsf{T}}\mathbf{x})^2]$$

Optimal linear predictor

- It can be shown that the predictor that minimizes $\sigma_e^2(\alpha)$ must satisfy the following condition:

$$E\left[\begin{bmatrix} \\ \mathbf{x} \\ \\ \end{bmatrix} \cdot e\right] = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\begin{aligned} e &= y - \overline{\boldsymbol{\alpha}}^{\mathsf{T}} \mathbf{x} \\ &= y - \mathbf{x}^{\mathsf{T}} \overline{\boldsymbol{\alpha}} \end{aligned}$$

  - The described condition requires that the prediction error *e* be linearly independent with each component of the original process ***X***, so it is also called the orthogonality condition.

Optimal linear predictor

- By substituting the expression for the prediction error, we get:

$$E[\mathbf{x}\cdot(y-\mathbf{x}^{\mathsf{T}}\overline{\boldsymbol{\alpha}})]=E[\mathbf{x}\cdot y]-E[\mathbf{x}\cdot\mathbf{x}^{\mathsf{T}}\overline{\boldsymbol{\alpha}}]=\begin{bmatrix}0\\ \vdots\\ 0\end{bmatrix}$$

  - and since the predictor can be drawn in front of the expectation operator, we obtain the following expression:

$$\boldsymbol{\Phi}_{xx}\cdot\overline{\boldsymbol{\alpha}}=\boldsymbol{\Psi}_{xy}$$

$$\boldsymbol{\Phi}_{xx}=E[\mathbf{x}\cdot\mathbf{x}^{\mathsf{T}}]=E[\begin{bmatrix}\\ \mathbf{x}\\ \\ \end{bmatrix}\cdot[\quad\mathbf{x}^{\mathsf{T}}\quad]] \qquad \boldsymbol{\Psi}_{xy}=E[\mathbf{x}\cdot y]=E[\begin{bmatrix}\\ \mathbf{x}\\ \\ \end{bmatrix}\cdot y]$$

Optimal linear predictor

- Thus, the optimal predictor is found by solving a linear system of equations.
  - The system matrix $\mathbf{\Phi}_{xx}$ is called the covariance matrix because it describes the correlation properties between all pairs of components of the original process ***X***;
  - It is a symmetric matrix of dimension $p \times p$.
  - The column $\mathbf{\Psi}_{xy}$ describes the correlations between the target process Y and the individual components of the source process ***X*** and has dimension $p \times 1$.
  - We find the solution as: $\overline{\boldsymbol{\alpha}} = \mathbf{\Phi}_{xx}^{-1} \mathbf{\Psi}_{xy}$

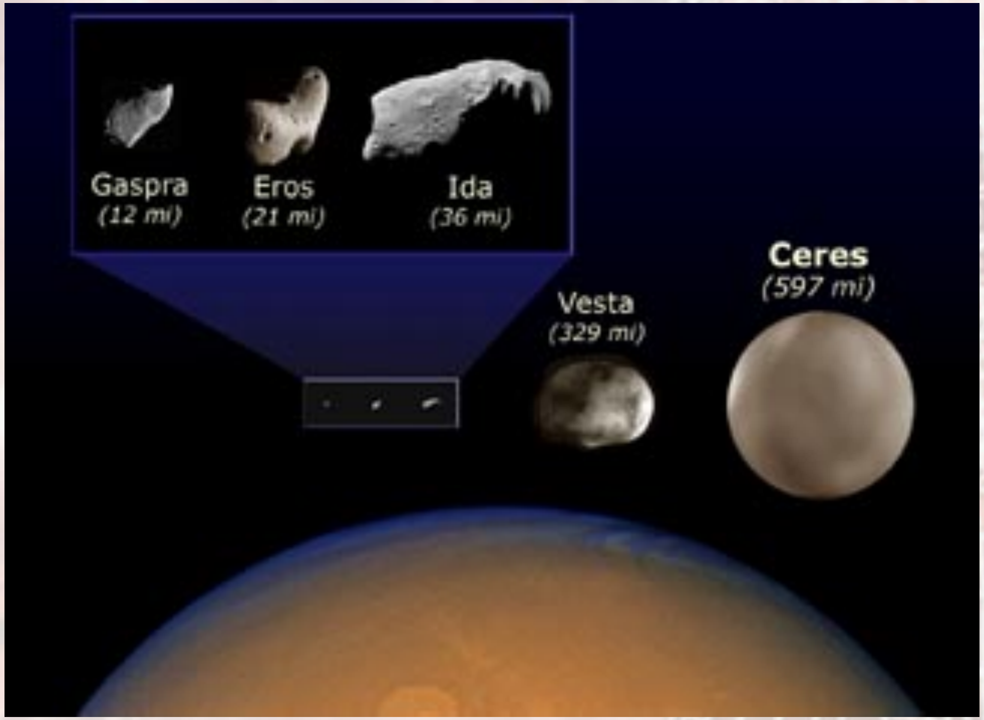
Linear prediction

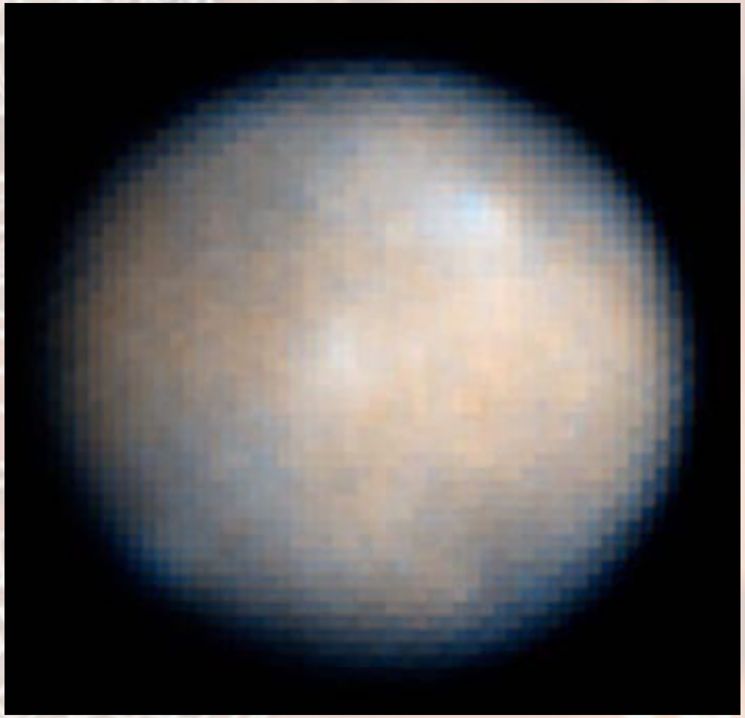
- Linear prediction is closely related to **regression analysis**, which in its earliest version derives from the **least squares method**.
- The least squares method was developed by the mathematicians **Legendre** (1805) and **Gauss** (1809) for the analysis of the orbits of celestial bodies; planets and their natural satellites.

Least squares method

- Adrien-Marie Legendre Carl Friedrich Gauss

Least squares method





- Gauss applied the least squares procedure to the estimation of the path of the dwarf planet Ceres
- *Theoria motus corporum coelestium in sectionibus conicis solem ambientum, 1801 (published 1809)*

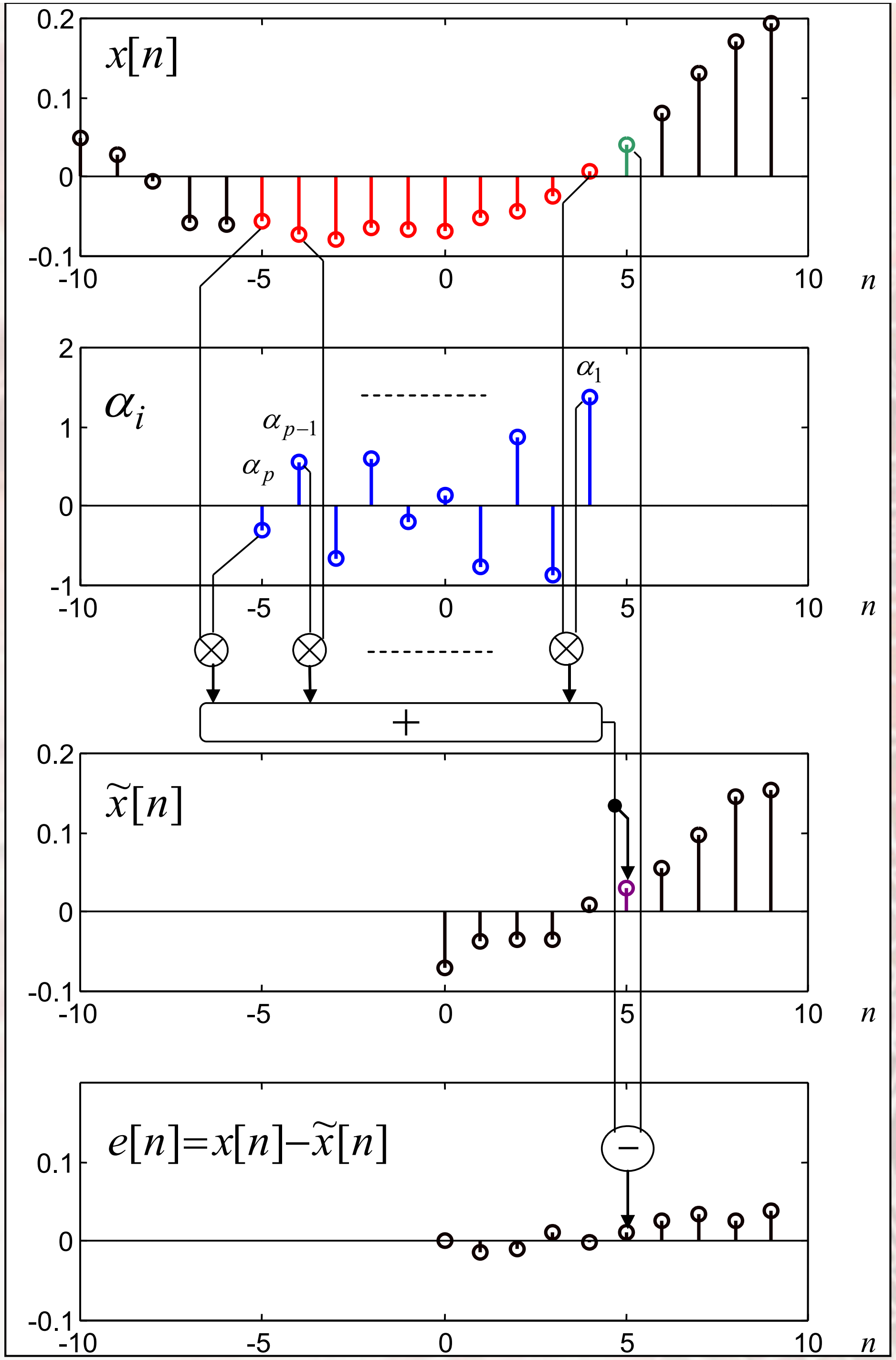
Modeling of time correlations

- The linear prediction procedure can be used to model **signal time correlations**.
- Assumption ... the current signal sample $x[n]$ is correlated with the previous samples $x[n\text{-}1]$, $x[n\text{-}2]$, ... $x[n\text{-}p]$, and can thus be predicted from them.
- The source process $\boldsymbol{X}$ and the target process Y are formed from samples of the same signal as :

$$\mathbf{x} = \begin{bmatrix} x[n-1] \\ x[n-2] \\ \vdots \\ x[n-p] \end{bmatrix}$$

$$\mathbf{X} = \{\mathbf{x}\}$$

$$Y = \{y\} = \{x[n]\}$$



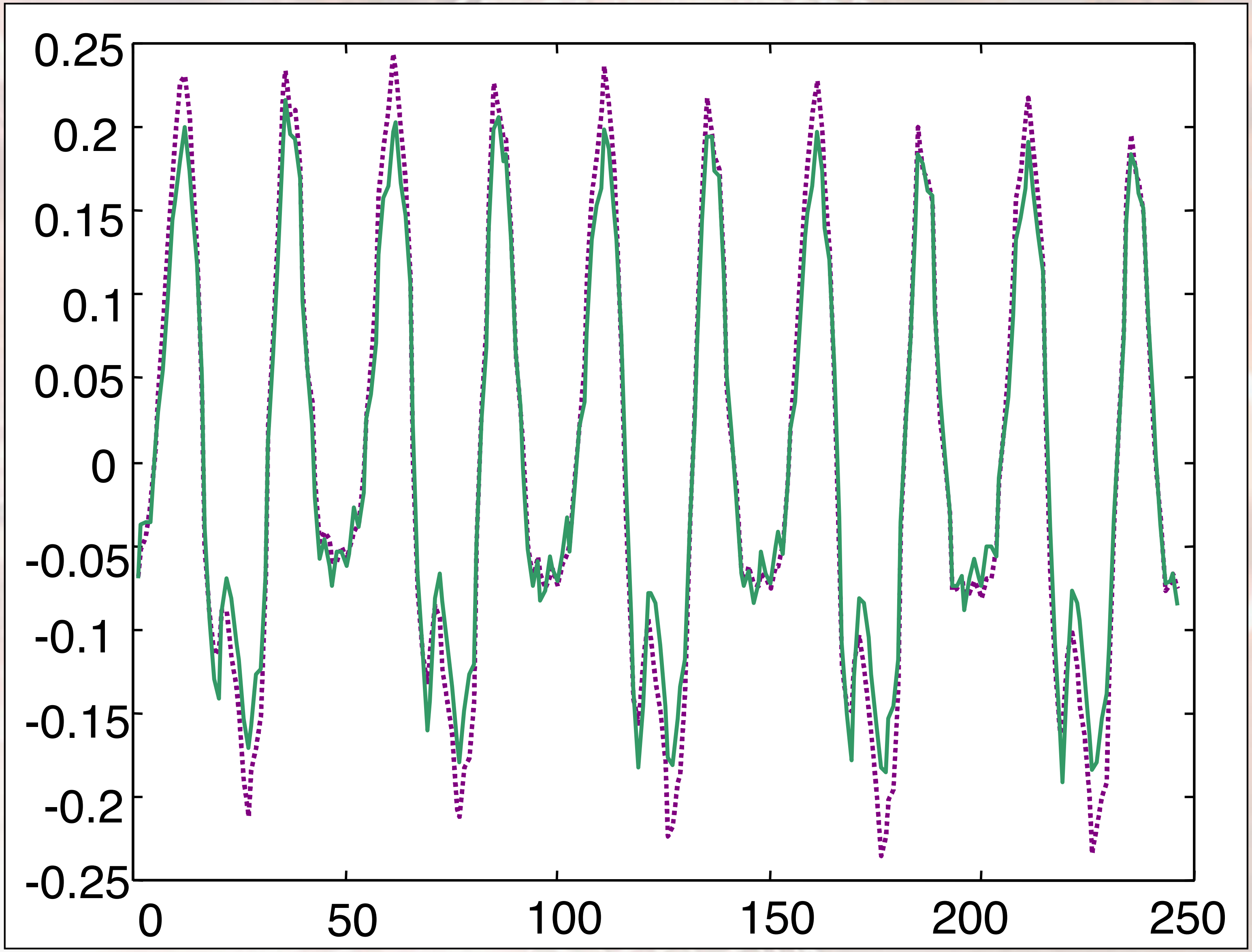
- signal
  - original process .. ***x***
  - target process ... *y*

- predictor coefficients α

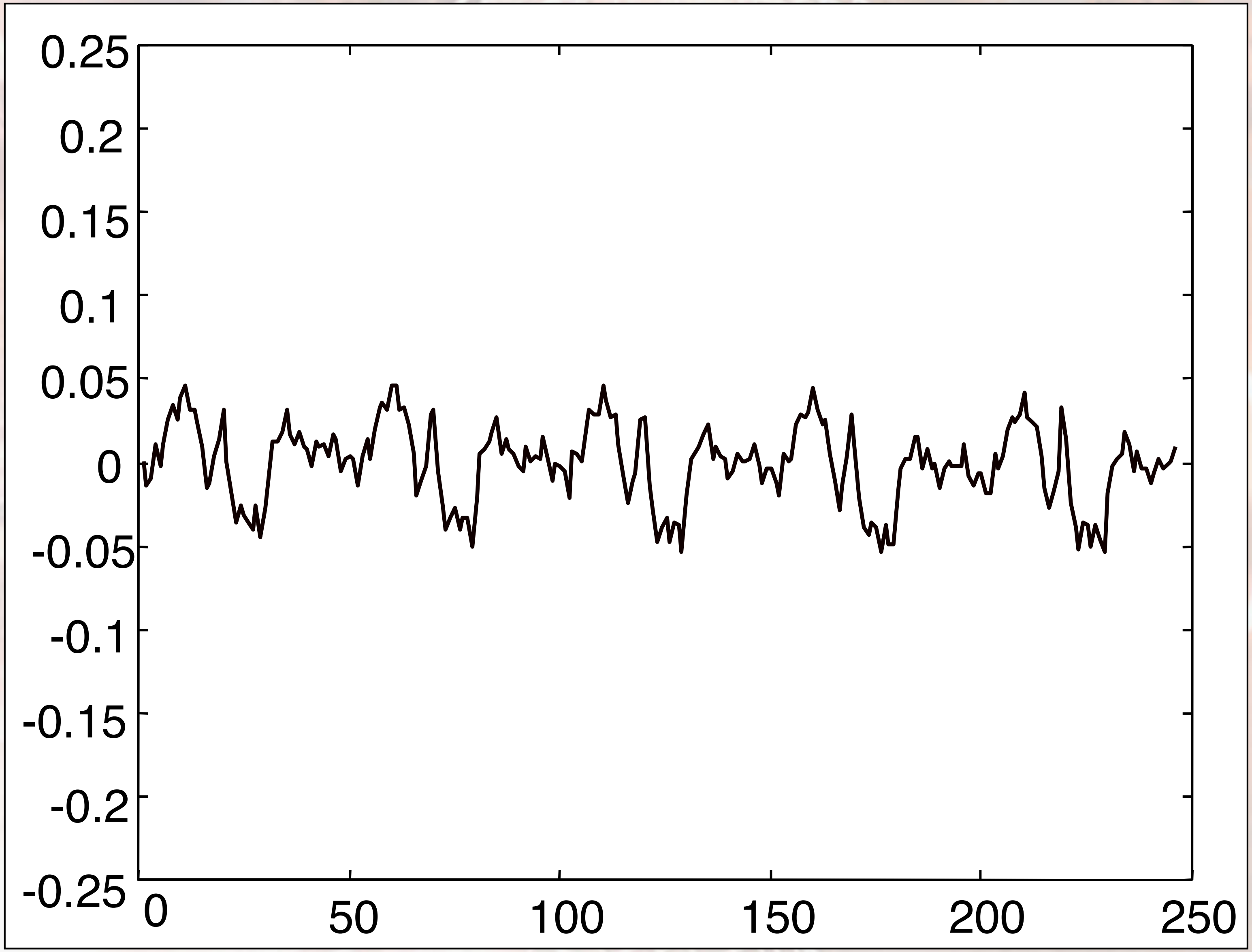
- prediction

- prediction error

Example of a signal and its prediction



Prediction error for the same example



Linear prediction of time-correlated signals

- For the described case of modeling time correlations, the matrix of the system of equations is:

$$\boldsymbol{\Phi}_{xx} = E[\mathbf{x}\cdot\mathbf{x}^{\mathsf{T}}] = E\left[\begin{bmatrix} x[n-1] \\ x[n-2] \\ \vdots \\ x[n-p] \end{bmatrix} \cdot [x[n-1]\ x[n-2]\ \cdots\ x[n-p]] \right]$$

  – assuming the ergodicity of process ***X*** the expectation of this outer product is calculated by averaging over all time samples (i.e., as time average over index *n*)

Linear prediction of time-correlated signals

- Analogously, the column $\mathbf{\Psi}_{xy}$ to the right of the matrix equation is determined as:

$$\mathbf{\Psi}_{xy} = E[\mathbf{x} \cdot y] = E\left[\begin{bmatrix} x[n-1] \\ x[n-2] \\ \vdots \\ x[n-p] \end{bmatrix} \cdot x[n] \right]$$

Linear prediction of time-correlated signals

- In practice, the signal *x*[*n*] for which the optimal predictor is calculated will be of finite duration *N*.
- Then, assuming the stationarity of the signal, the expectation operator can be replaced by simple averaging over all available samples of the final realization of this process :

$$\mathbf{\Phi}_{xx}(i,k)=E[x[n-i]\cdot x[n-k]]\cong \phi(i,k)=\frac{1}{N}\sum_{n} x[n-i]\cdot x[n-k]$$

$$\mathbf{\Psi}_{xy}(i)=E[x[n-i]\cdot x[n]]\cong \phi(i,0)=\frac{1}{N}\sum_{n} x[n-i]\cdot x[n]$$

Linear prediction of time-correlated signals

- This leads to a system of equations of form :

$$\begin{bmatrix} \phi(1,1) & \phi(1,2) & \ldots & \phi(1,p) \\ \phi(2,1) & \phi(2,2) & \ldots & \phi(2,p) \\ \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ \phi(p,1) & \phi(p,2) & \ldots & \phi(p,p) \end{bmatrix} \begin{bmatrix} \overline{\alpha}_1 \\ \overline{\alpha}_2 \\ \ldots \\ \ldots \\ \overline{\alpha}_p \end{bmatrix} = \begin{bmatrix} \phi(1,0) \\ \phi(2,0) \\ \ldots \\ \ldots \\ \phi(p,0) \end{bmatrix}$$

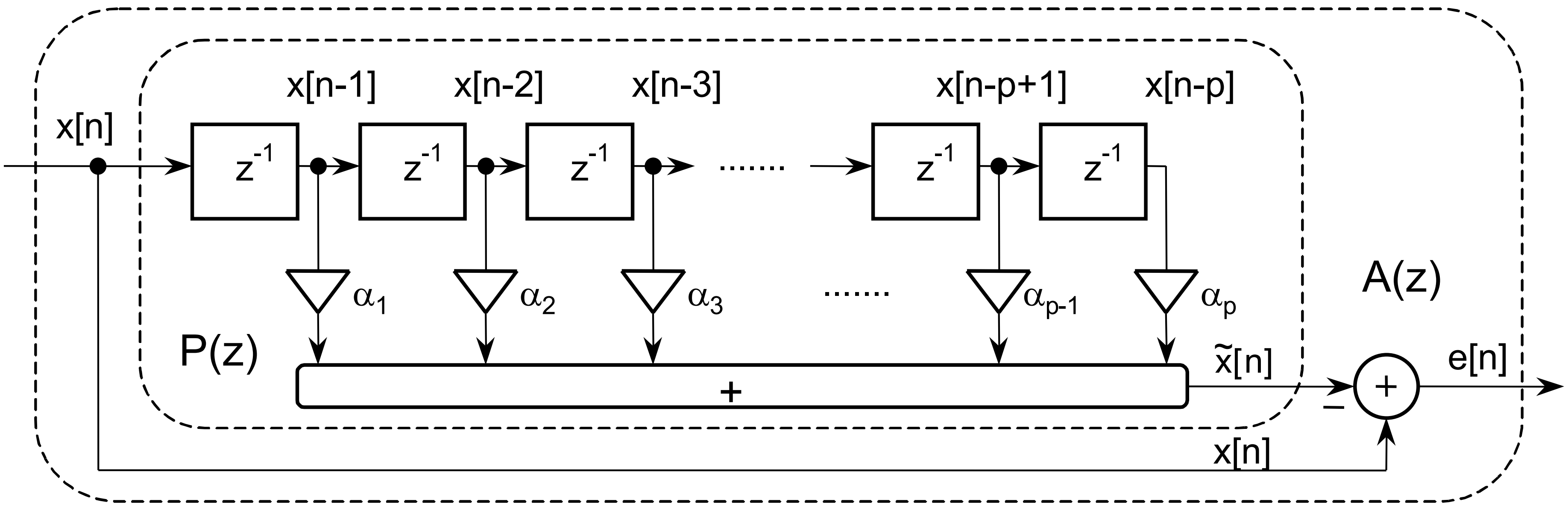
$$\phi(i,k) = \frac{1}{N} \sum_n x[n-i] \cdot x[n-k]$$

  – factor 1/*N* can be omitted due to cancelation!

Properties of a linear predictor

- A predictor with *p* coefficients can model correlations up to the *p* samples backwards
  - The number *p* is called the **prediction order**.
  - Increasing the order monotonically reduces the prediction error variance $\sigma_e^2(\overline{\boldsymbol{\alpha}})$
- The linear predictor has a very simple implementation as a digital filter with a finite impulse response of order *p*, with a transfer function denoted by *P*(*z*) .
  - The prediction calculation requires *p* multiplication and *p*-1 addition for one prediction output sample $\tilde{x}[n]$

Implementation of a linear predictor



- Transfer function of predictor $P(z)$

$$e[n] = x[n] - \tilde{x}[n] = x[n] - \sum_{k=1}^{p} \alpha_k x[n-k]$$

$$P(z) = \sum_{k=1}^{p} \alpha_k z^{-k}$$

$$A(z) = \frac{E(z)}{X(z)} = 1 - \sum_{k=1}^{p} \alpha_k z^{-k} = 1 - P(z)$$

Reconstruction of the output signal from prediction error

- Under certain conditions, it is possible to "invert" the prediction procedure by reconstructing the original signal from the prediction error signal.
- This is done by digital filtering of *e*[*n*] with a recursive filter *H*(*z*) which has an infinite impulse response *h*[*n*] and a transfer function:

$$H(z) = \frac{1}{A(z)} = \frac{X(z)}{E(z)} = \frac{1}{1-P(z)} = \frac{1}{1-\sum_{k=1}^{p}\alpha_k z^{-k}}$$

  - ... so, *A*(*z*) is called an inverse filter.

Reconstruction of the output signal from prediction error

- Filter $A(z)$ and $H(z)$ are inverse to each other, so the zeros of filter $A(z)$ represent the poles of filter $H(z)$.
- Filter $H(z)$ will be stable, and reconstruction is possible only if all of its poles are within the unit circle in the z-plane, which requires that filter $A(z)$ has all of its zero with magnitude less than unity.
- This can be achieved by applying appropriate predictor calculation procedures, and one of the most popular is the **autocorrelation method**.

System of equations for autocorrelation method

- $R(j)$ is the autocorrelation of the finite signal $x[n]$ with samples at indices n=0 to N-1 for lag j

$$R(j) = \sum_{m=0}^{N-1-j} x[m] \cdot x[m+j]$$

$$\begin{bmatrix} R(0) & R(1) & R(2) & .... & R(p-1) \\ R(1) & R(0) & R(1) & .... & R(p-2) \\ R(2) & R(1) & R(0) & .... & R(p-3) \\ .... & .... & .... & .... & .... \\ .... & .... & .... & .... & .... \\ R(p-1) & R(p-2) & R(p-3) & .... & R(0) \end{bmatrix} \begin{bmatrix} \overline{\alpha}_1 \\ \overline{\alpha}_2 \\ \overline{\alpha}_3 \\ ... \\ ... \\ \overline{\alpha}_p \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ R(3) \\ .... \\ .... \\ R(p) \end{bmatrix}$$

Linear Prediction Coding, LPC

- Due to its high prediction efficiency on a number of natural processes and signals, the linear prediction procedure finds frequent application in media signal coding procedures.
- Source coding procedures that use linear prediction are collectively called **Linear Prediction Coding, LPC**.
- One of the first applications of LPC was in the field of **speech coding**, but it was later extended to other types of media signals as well.

Types of predictive coding

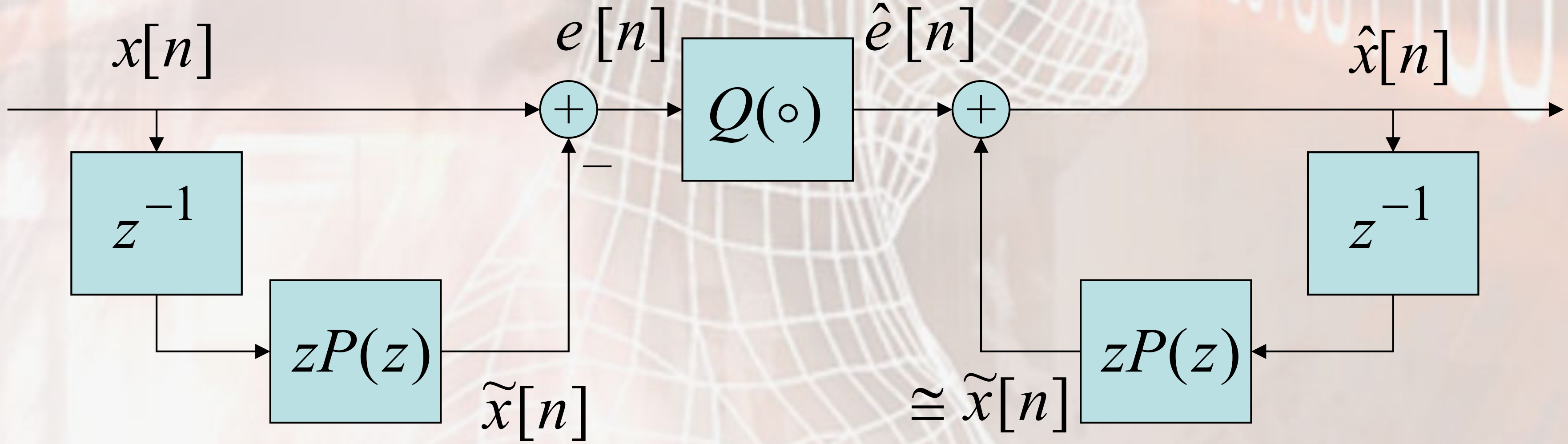
- Predictive signal coding can be divided into two basic groups:
  - ***open-loop prediction*** and
  - ***closed-loop prediction***.
- The fundamental difference is in the original process ***X*** from which the prediction is calculated:
  - In open loop coding, the source process used by the encoder and the decoder is similar but not the same, while
  - in closed loop predictive coding the same source process ***X*** is available and used on both sides, encoder and decoder.

Open loop Predictive coding

- The encoder performs prediction based on the delayed samples of the original signal and calculates the prediction error signal.
- The quantized prediction error signal is transmitted to the decoder.
- The decoder performs the inverse procedure on quantized prediction error samples :
  - calculates an approximate prediction based on previous reconstructed samples;
  - it adds that prediction to the quantized prediction error to calculate the current reconstructed output sample.

Open loop Predictive coding

- Sources of reconstruction errors :
  - quantization error: $e[n]-\hat{e}[n]$, but also
  - the difference between the predictions on the encoder and the decoder side (different source!).
- Reducing the quantization error automatically reduces the second error as well!



Closed loop Predictive coding

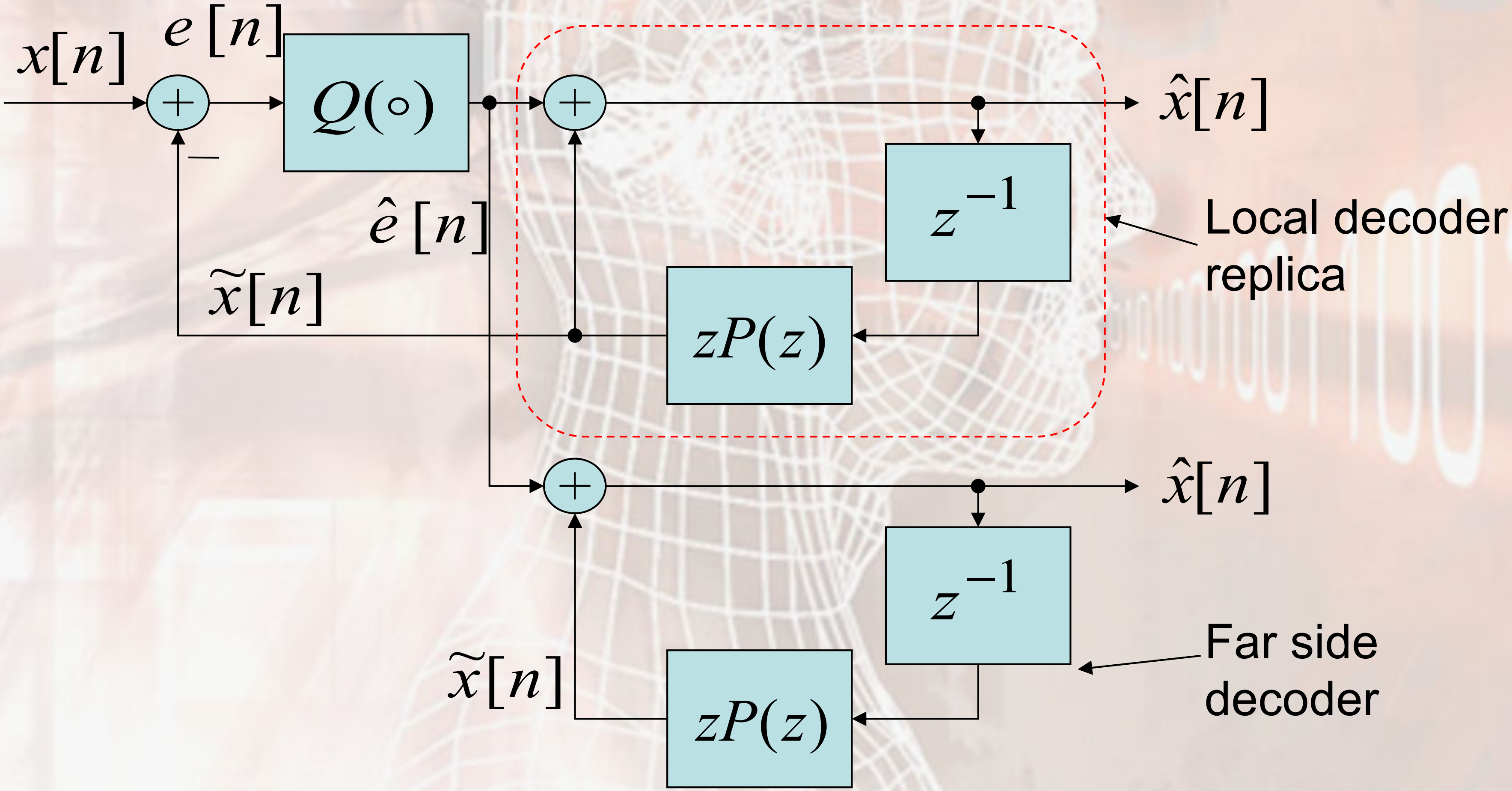
- In **closed loop** coding, *p* of the last reconstructed samples are used as the original prediction source instead of the delayed samples of the unquantized input signal.
- For this purpose, the encoder has a "local replica" of the decoder, which generates identical reconstructed signal samples at the output, as well as an actual decoder on the far side.
- In this way, previously reconstructed samples that are available on both sides represent a common prediction source process ***X*** based on which the current input unquantized sample $x[n]$ is predicted.

Closed loop Predictive coding

- The encoder calculates the difference between the current input sample and the prediction.
- It forwards the quantized prediction error to the decoder.
- The encoder and decoder add this quantized value to the locally calculated prediction and thus obtain a reconstructed (quantized) value of the original sample which they use as the decoder output (signal reconstruction) but also as the predictor source for the next sample on both sides.

Closed loop Predictive coding

- The encoder and decoder use the same prediction!



Predictor adaptation

- The media signal predicted by the predictor typically has **time-varying correlation properties**.
- Therefore, it is necessary to periodically adjust (**adapt**) the predictor to the signal properties.
- Unfortunately, this information must be reported to the decoder such that it also has an up-to-date replica of the predictor at all times, in order to be able to unambiguously decode the received message.
- Sending this side information "consumes" the communication channel and reduces efficiency.

Predictor adaptation

- The described method of predictor adaptation is called **forward-adaptive prediction** because additional information about the predictor is sent directly to the decoder as side information.
- The predictor in such an adaptation is typically calculated to maximize the efficiency of predictive coding of the samples that follow (future samples).
- The alternative is **backward-adaptive prediction** which avoids the need to send a predictor to the decoder.

Backward adaptation

- In the case of backward adaptation, a similar principle is used as in the case of predictive coding in a closed loop :
  – adaptation is carried out from samples available on both sides, i.e., from previously reconstructed output samples.
- Conceptual problem, ...
  – The predictor is optimized for samples that have already been sent, not for those yet to come, consequently
  – the procedure only makes sense if the correlation properties of the signal change slowly enough.
  – Additionally, the predictor is calculated from quantized output samples, so quantization error also affects the efficiency (optimality) of a particular predictor ... it does not work for low data rates!

What have we learned?

- signal prediction, source and target process, prediction error
- predictor, method of determination
- linear predictor
- optimal solution, orthogonality condition
- least squares method
- temporal correlation modeling
- linear prediction of time-correlated signals
- properties and performance of predictors
- reconstruction of signal from prediction error
- autocorrelation method for determining predictors
- linear prediction coding
- predictive coding in open and closed loop
- predictor adaptation

Basics of speech coding

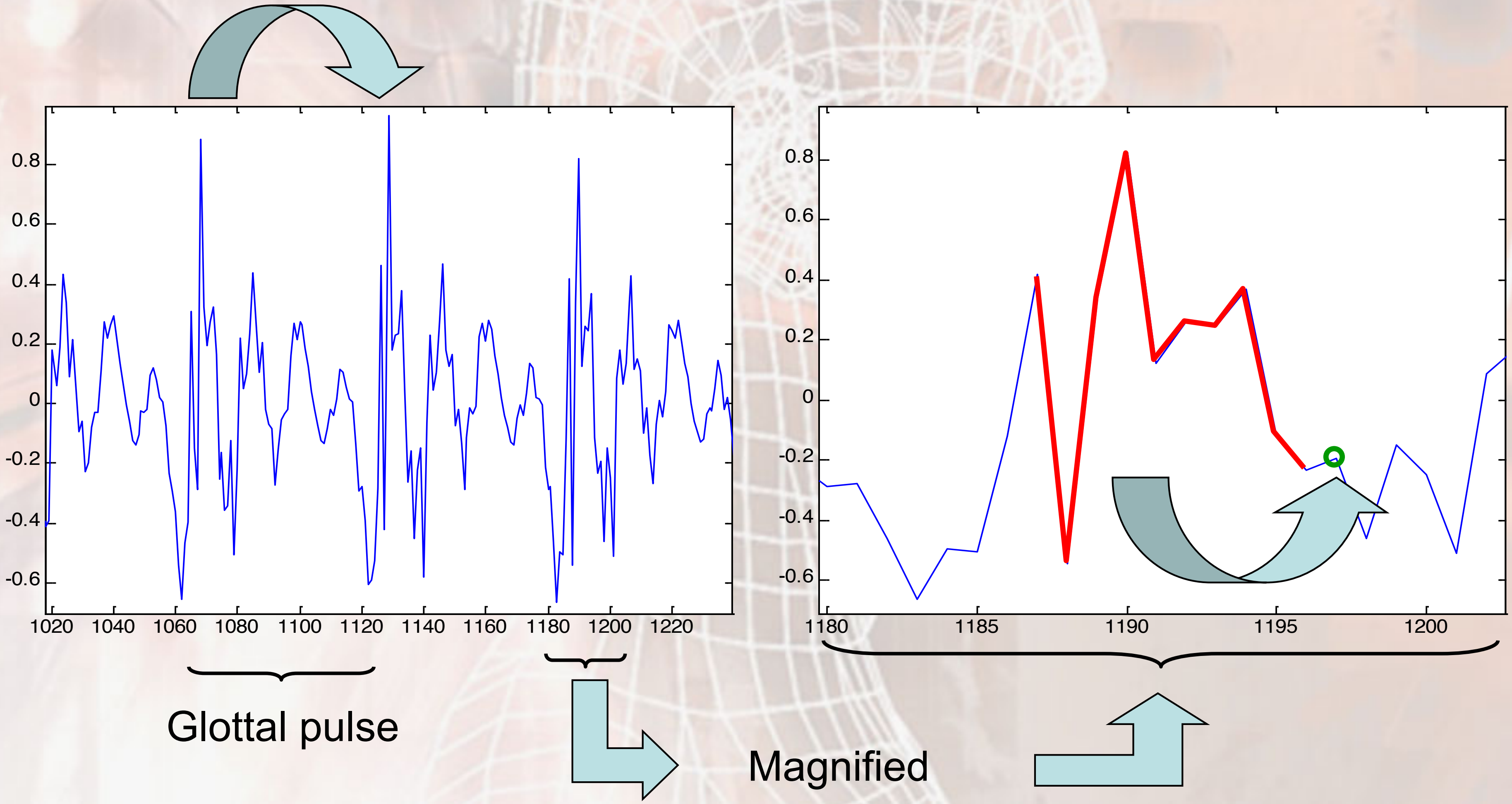
prof.dr.sc. Davor Petrinović

Speech coding

- Most speech coding procedures used today are based on linear prediction.
- The LPC model is extremely close to the physical process of speech formation, so the application of LPC coding to speech gives extremely good results!
- There are two types of correlations in speech
  - temporal **short-term correlations** related to **vocal tract** activity
  - temporal **long-term correlations** related to the process of **excitation of the vocal tract** on the vocal cords.
- LPC has been used successfully for both.

Correlations in speech

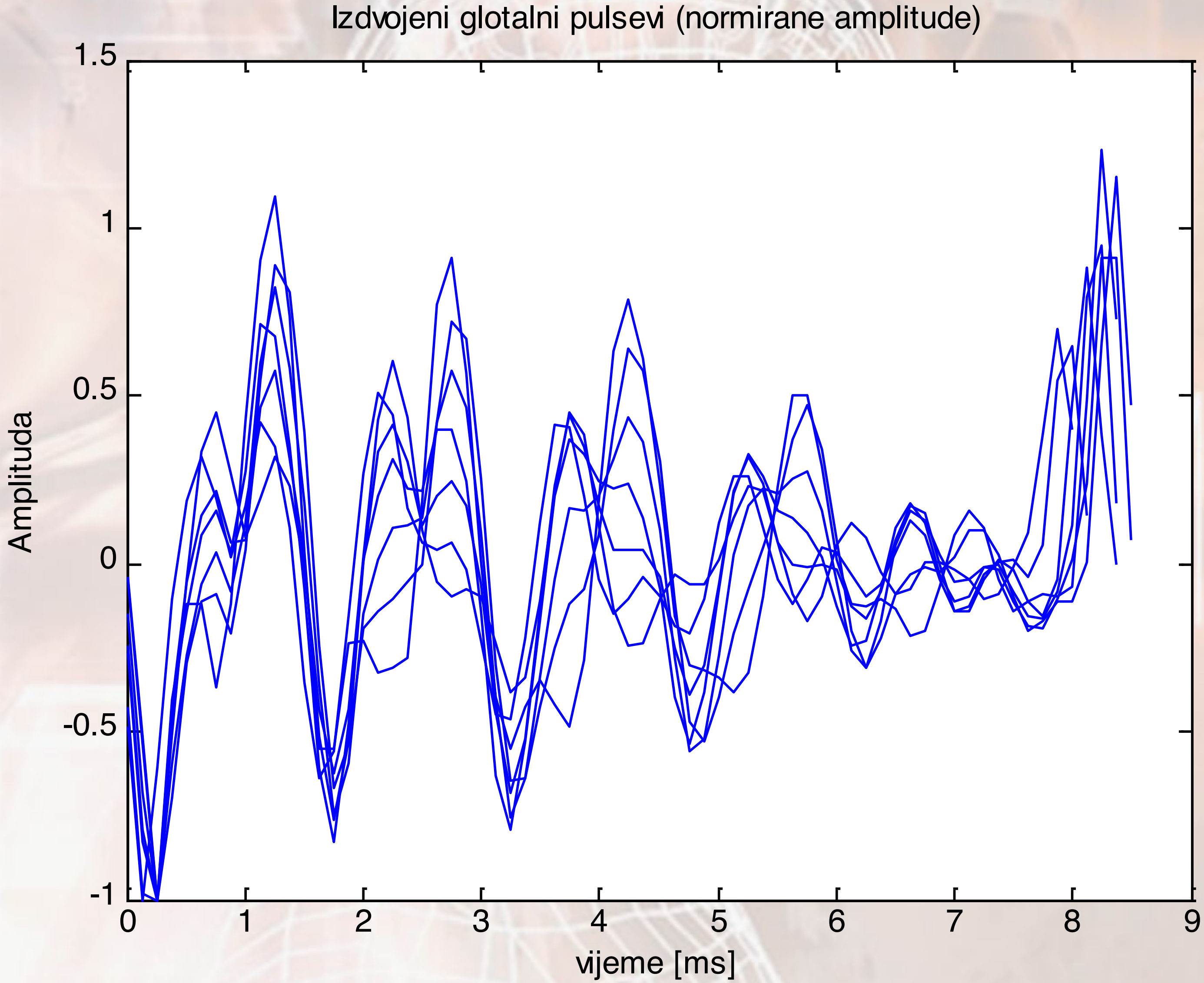
- long-term correlation: short-term correlation:



Long-term correlations

- **Long-term correlations** are manifested as similarities between adjacent periods (glottal pulses) of the speech signal.
- Each period corresponds to one opening of the vocal cords, when a new pulse of air current from the lungs appears at the entrance of the vocal tract.
- The **pitch period** depends on the pitch of the voice and ranges from *PP*=3ms for children's voices, up to *PP*=12-15ms for deep male voices.
- The more similar the neighboring periods are and the more time-constant their period is, the long-term prediction becomes more successful.

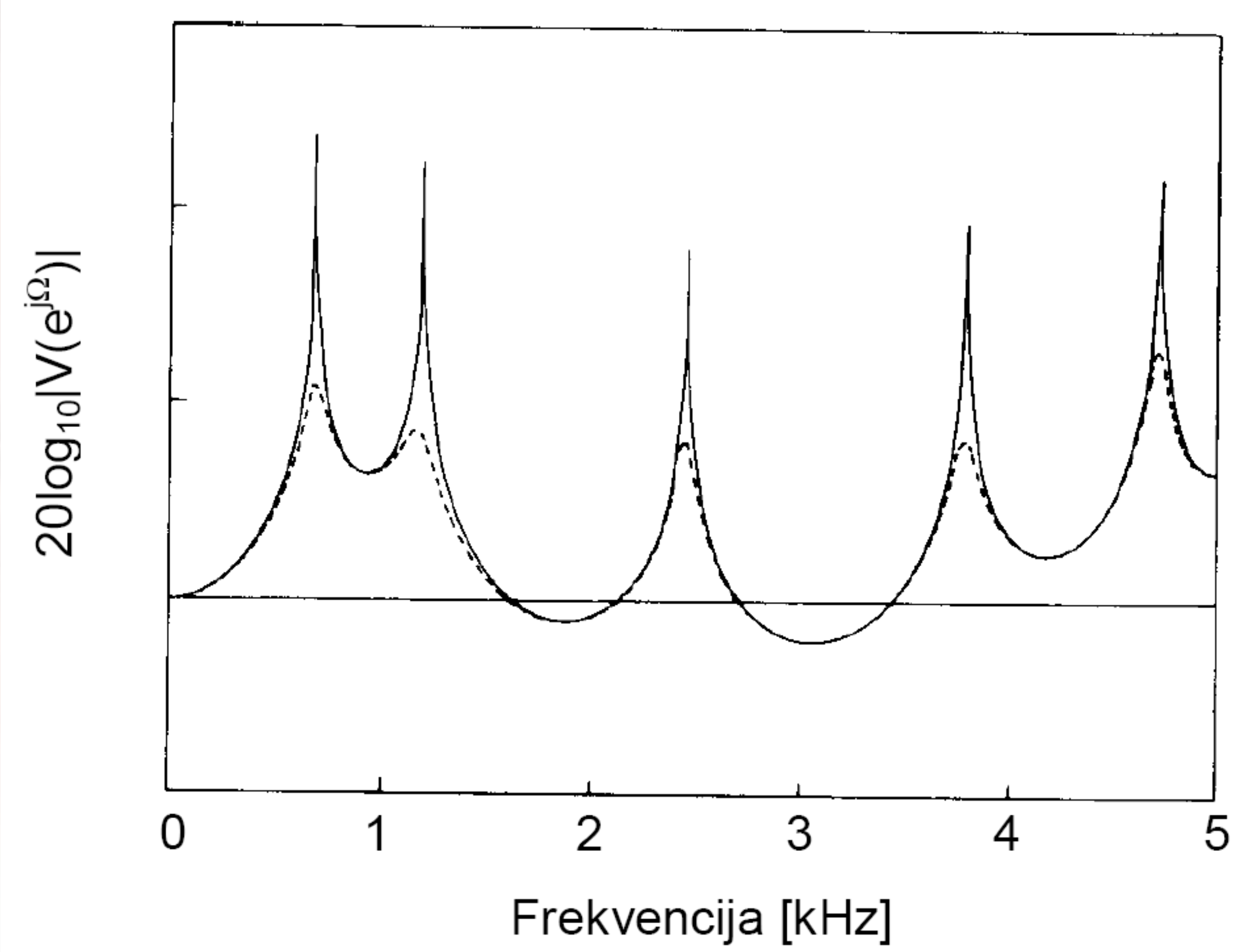
Similarity of adjacent periods in speech



Short-term correlations

- **Short-term** correlations are related to changes introduced by the vocal tract on the signal coming from the vocal cords and extend over time periods within 1ms.
  - The vocal tract acts as a series of **selective filters** that amplify individual spectral regions of the excitation signal.
  - The spectral positions and widths of these filters depend on the **phoneme** being spoken.
  - E.g., for the **vowel** “a” the neighborhoods of the following frequencies are significantly amplified: 800Hz, 1200Hz, 2500Hz, 3800Hz.
  - Such resonant characteristics in speech are called **formants**.

Formants of vowel “a”



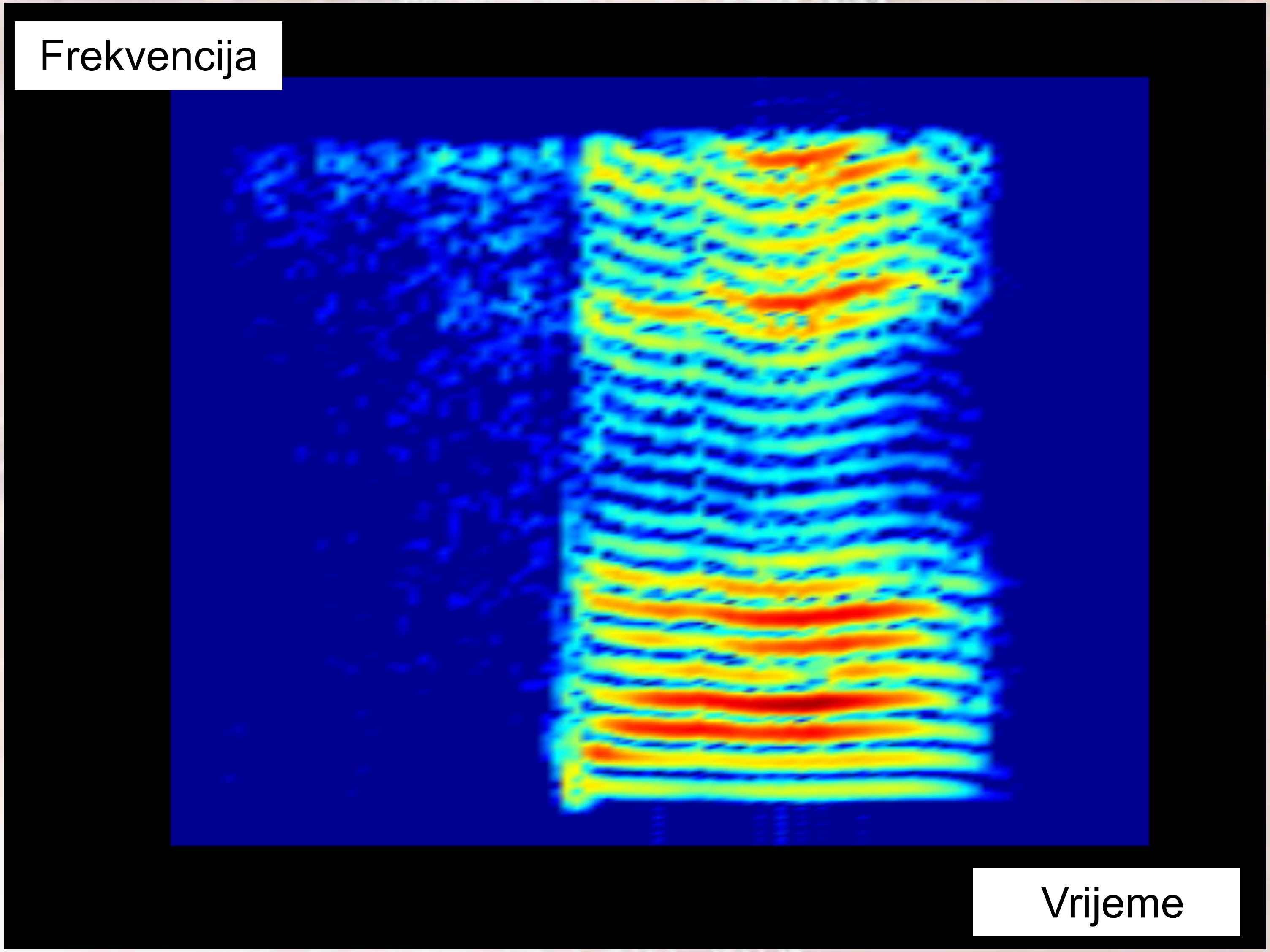
Spectrogram of speech

- Used to visualize **correlations** in a speech signal.
- It shows the change of the spectrum, i.e., the magnitude of the Fourier transform of speech over time and with frequency.
- Log-magnitude (intensity of the spectrum) is represented by the third dimension using pseudo color (or shade of gray).
- Depending on the frequency selectivity, there are two types of spectrograms:
  - **narrowband spectrogram** and
  - **wideband spectrogram**.

Narrowband speech spectrogram

- The frequency resolution of the analysis was adjusted to reflect the periodicity of the signals related to **long-term correlations** in the spectrogram.
- Each individual time window of the Fourier analysis must enclose more than two glottal pulses.
- Long-term correlations are manifested as **horizontal stripes** in the narrowband spectrogram.
- The track spacing is equal to the fundamental frequency of vocal cord vibrations (reciprocal of the pitch period *PP*)
- The vertical deviations of these lines describe the change in the fundamental frequency of the voice during natural pronunciation.

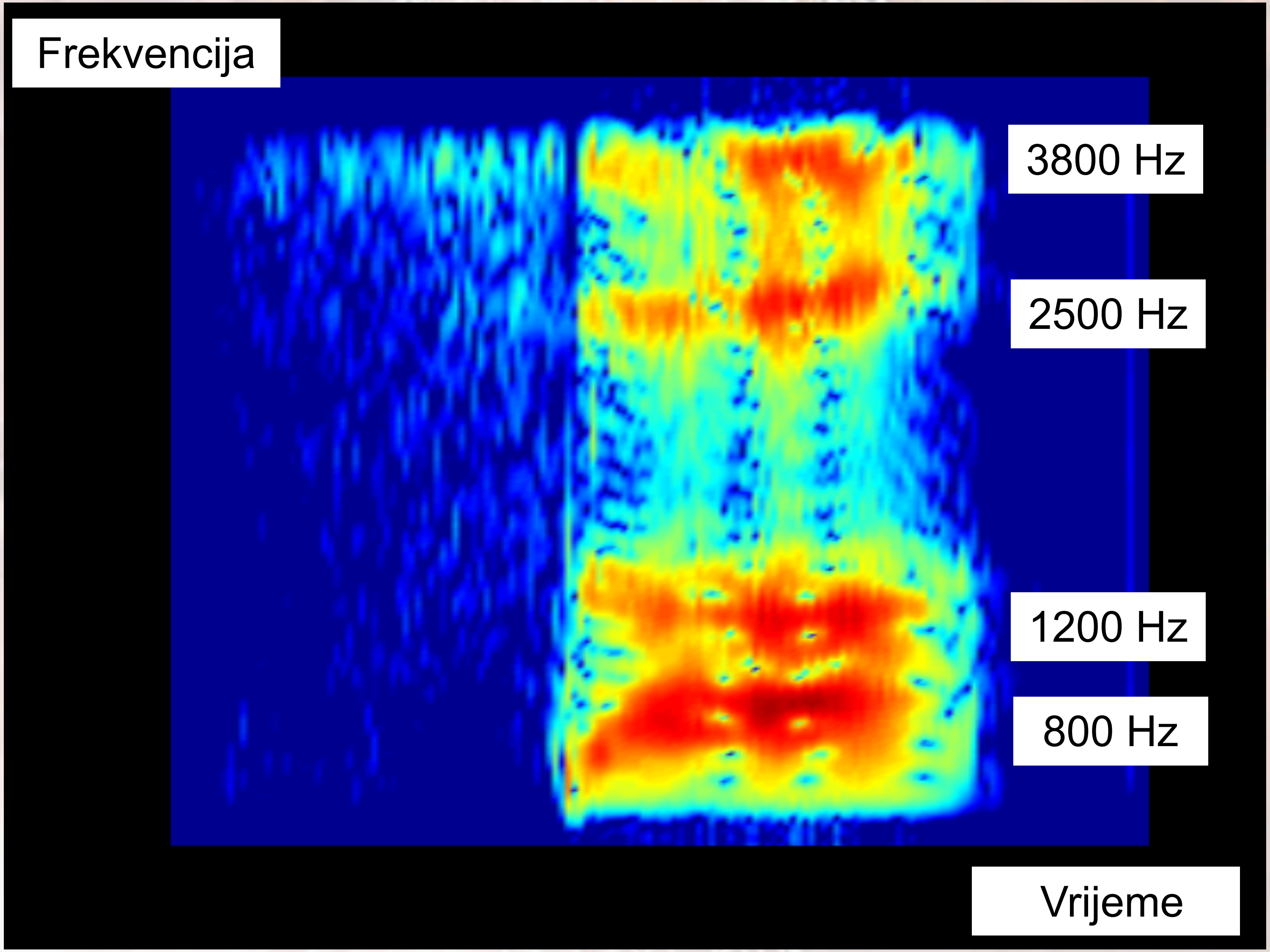
Narrowband spectrogram of “sa”



Wideband speech spectrogram

- Wideband spectrogram is used to visualize **short-term correlations** in the speech signal.
- Rapid changes along the frequency axis that are related to the periodic excitation signal (vocal folds vibration) have been removed, because the frequency resolution of the analysis is insufficient to separate individual harmonics (spectral lines).
- It shows the spectral peaks that correspond to the **temporal evolution of the format structure** during the pronunciation of an individual voice.

Wideband spectrogram of “sa”



The beginnings of speech modeling and coding

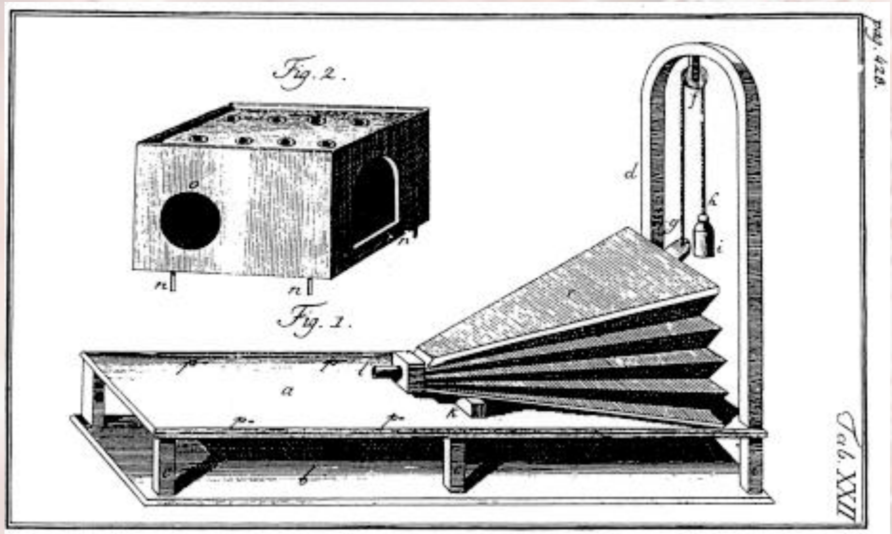
- Research in the field of speech began very early.
- The theory of speech modeling and coding is the result of research by physicists and engineers who have been analyzing and proposing various **physical models of the process of speech formation** in the vocal tract since the 18th century.
- They also built various mechanical and electrical devices that tried to simulate these processes ... i.e., to synthesize speech.

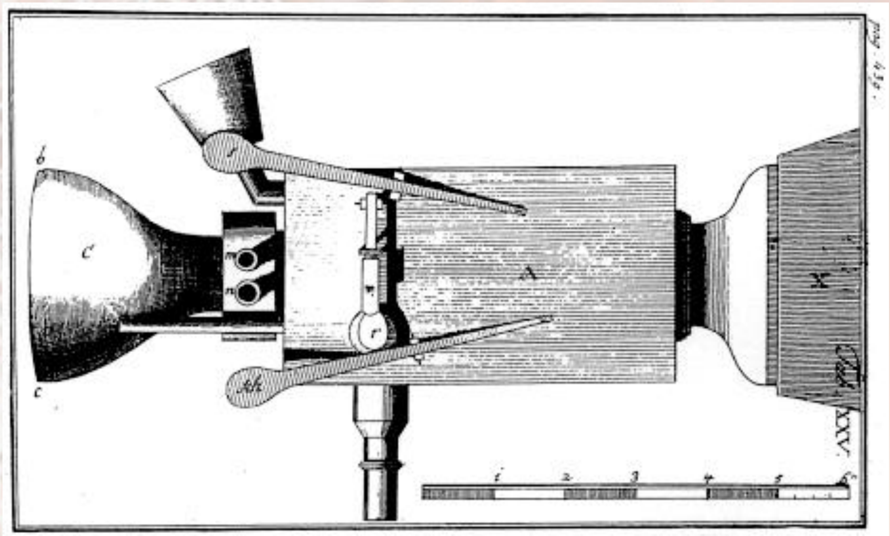
Mechanical speech synthesizer

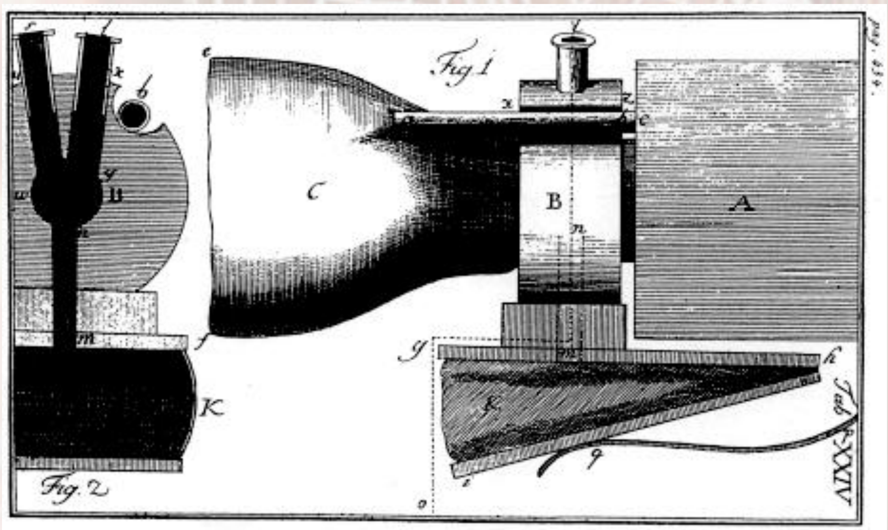
- **Wolfgang von Kempelen**
- **1734-1804**

- **In 1791.** he built a mechanical device that could “speak” under the control of the operator.
- He describes it in the book : *Mechanismus der menschlichen Sprache nebst Beschreibung einer sprechenden Maschine*

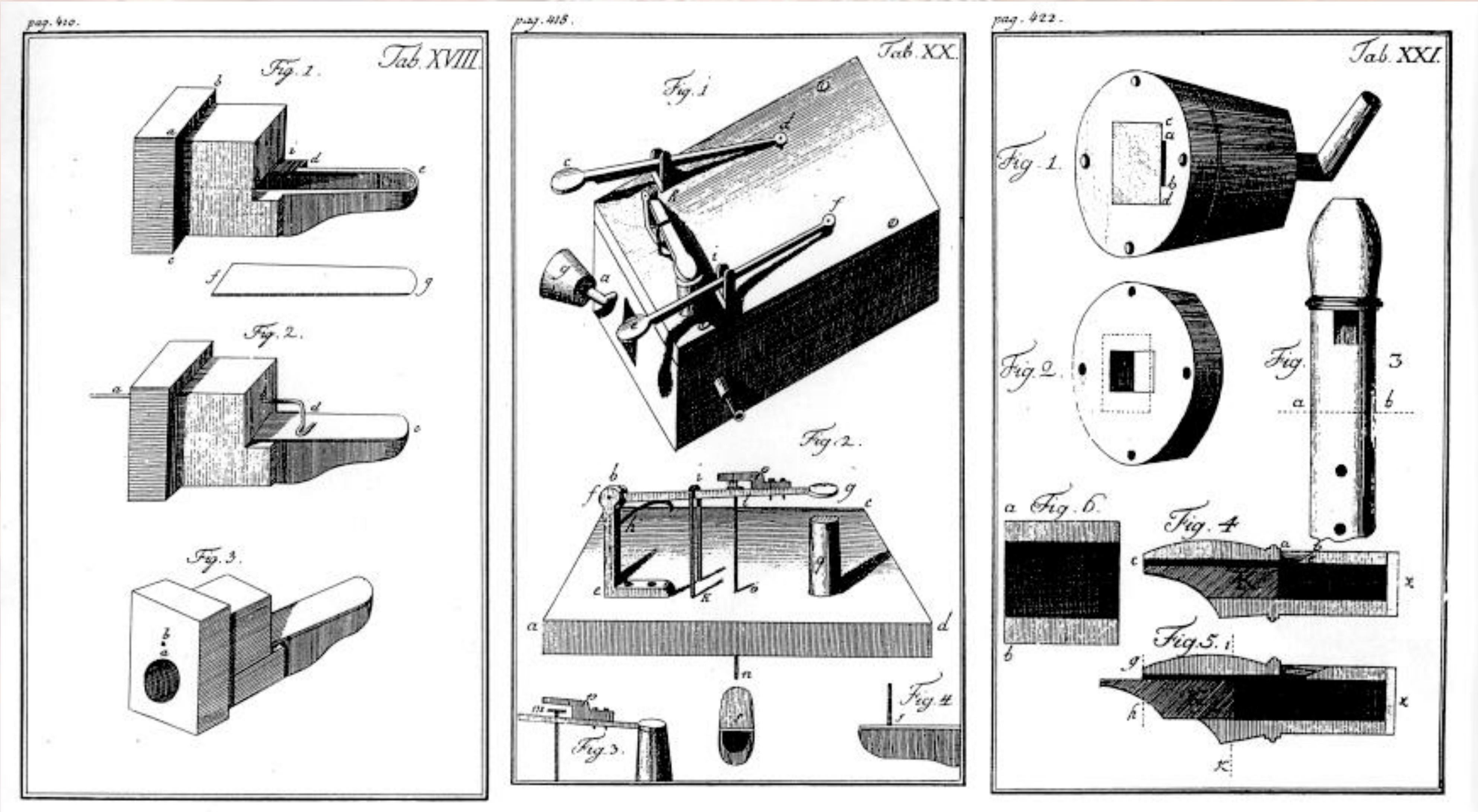
Kempelen's speaking machine







Kempelen's speaking machine



Kempelen's speaking machine

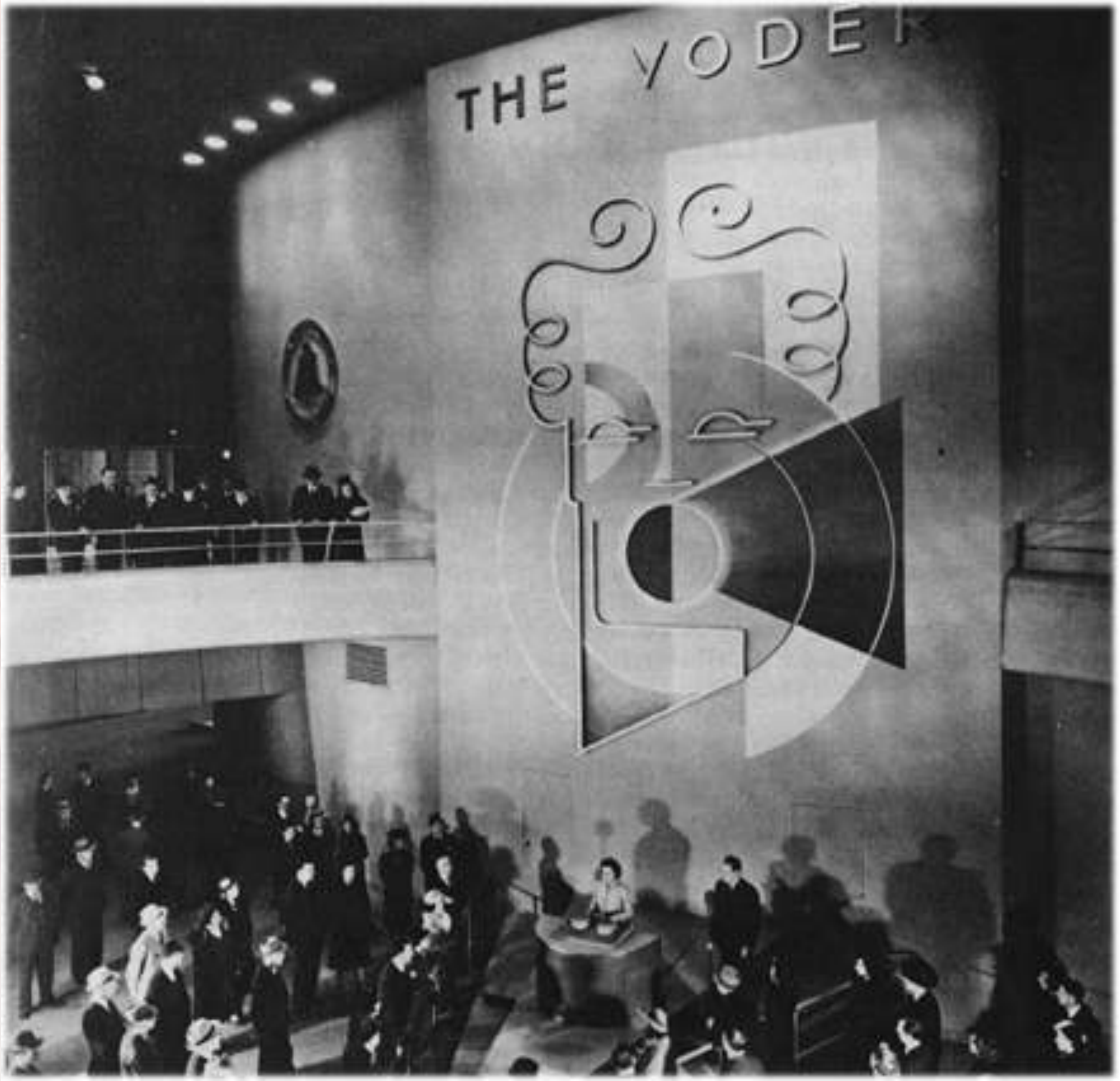
- Exhibited in: *Deutsches Museum von Meisterwerken der Naturwissenschaft und Technik in München*

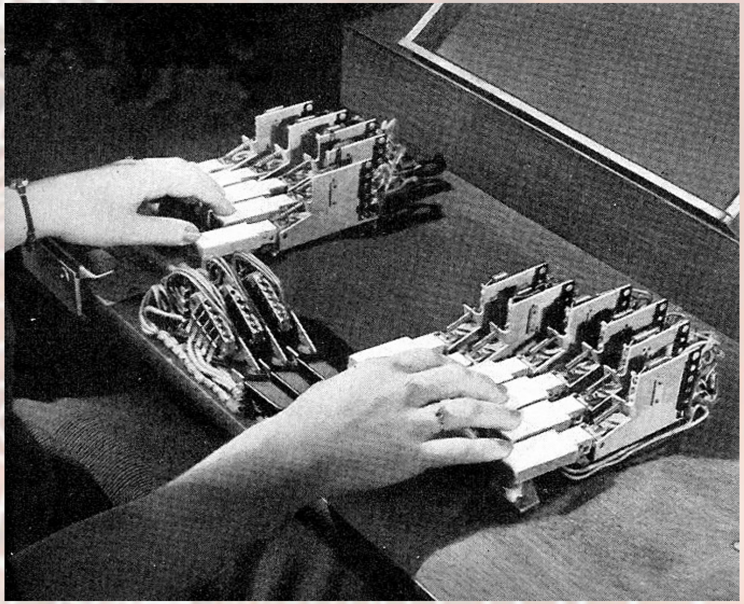
The first electric synthesizer, VODER

- **Homer Dudley**

- In 1939, at the World Fair in New York, he presented an all-electric speech synthesizer developed at the Bell Telephone Laboratory called **VODER** (**V**oice **O**perated re**C**or**DER**).
- One of the greatest "miracles" of that time.

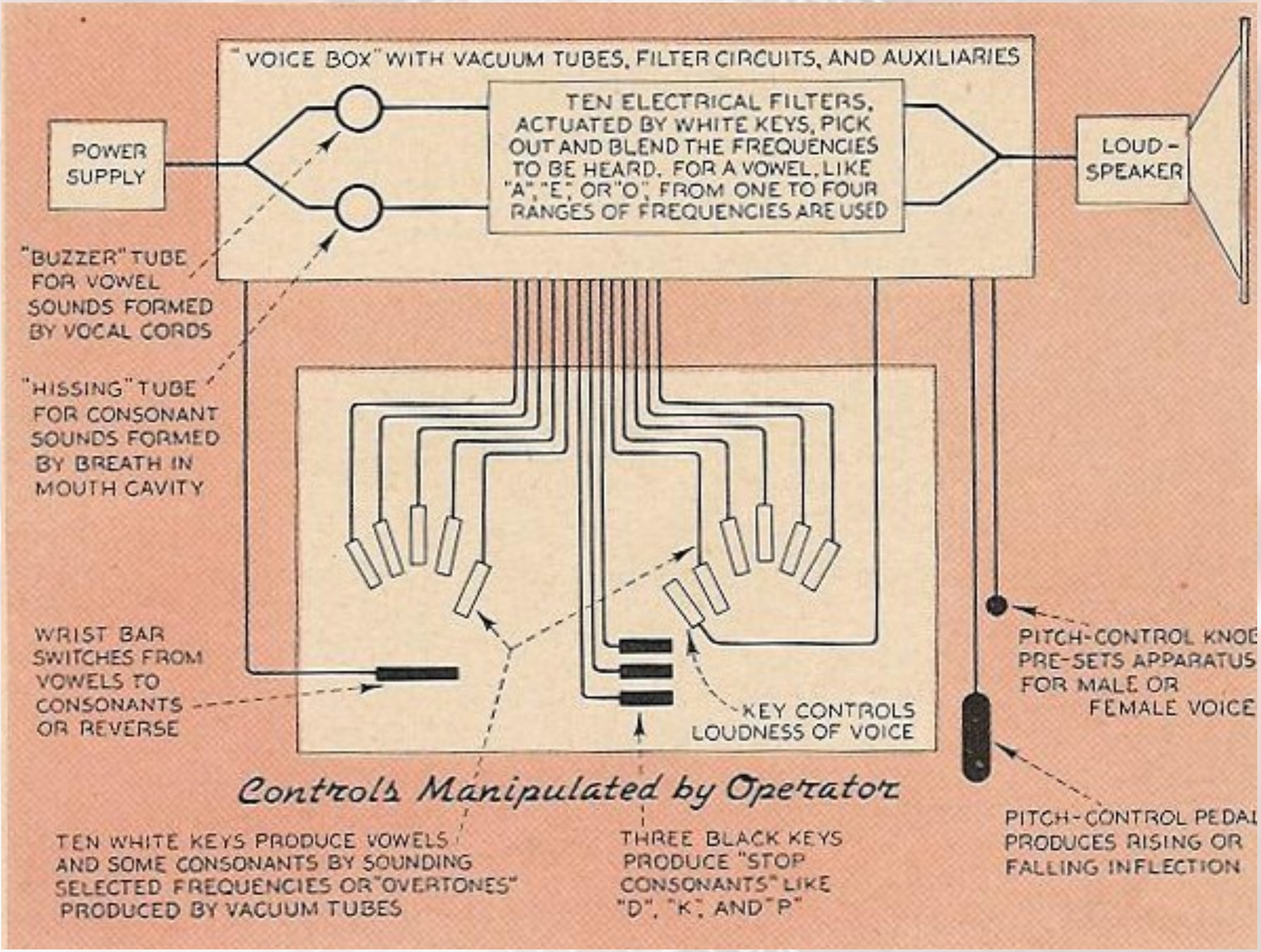
The first electric synthesizer, VODER



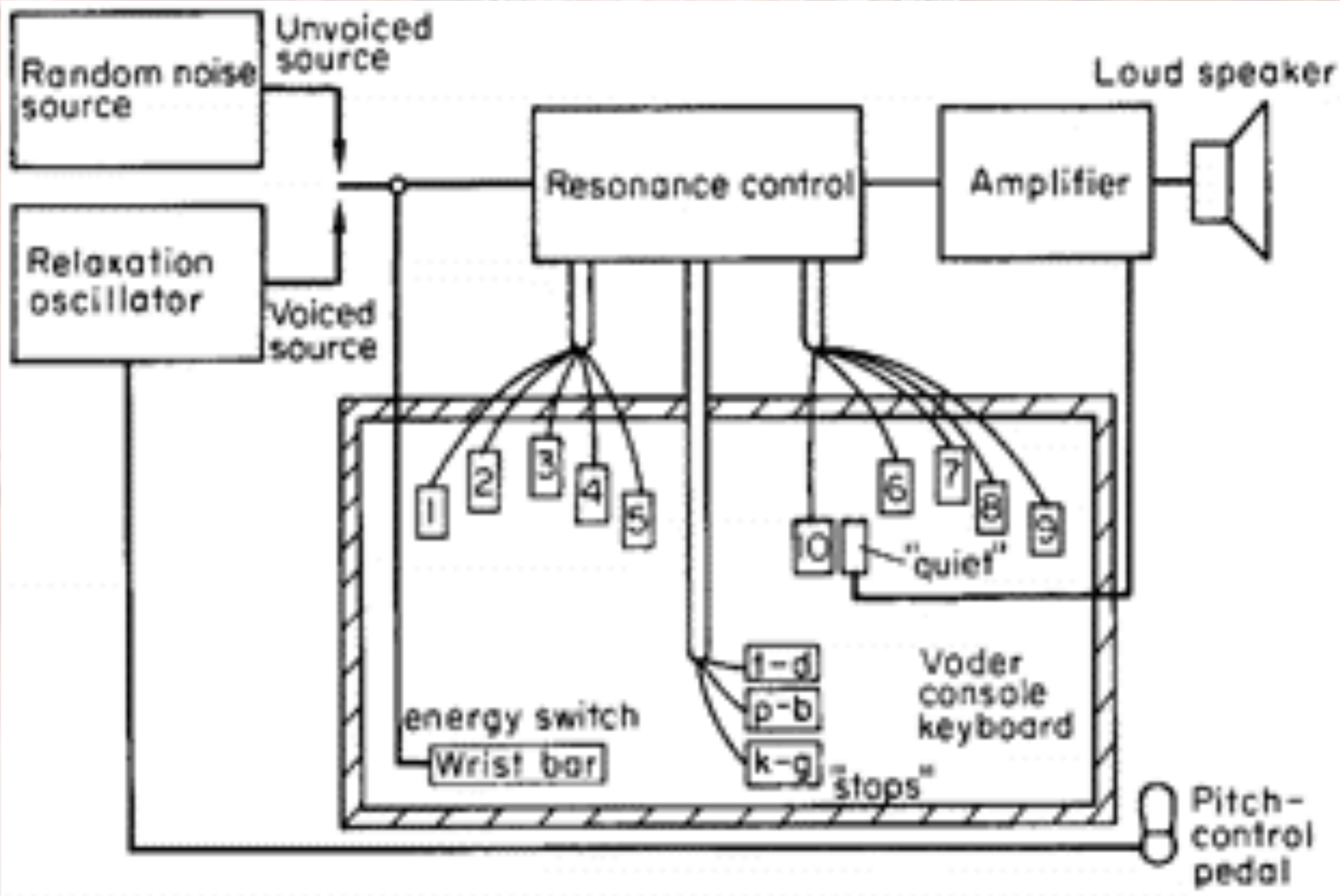


- Keyboard control
- Presentation at the 1939 fair

The first electric synthesizer, VODER



The first electric synthesizer, VODER

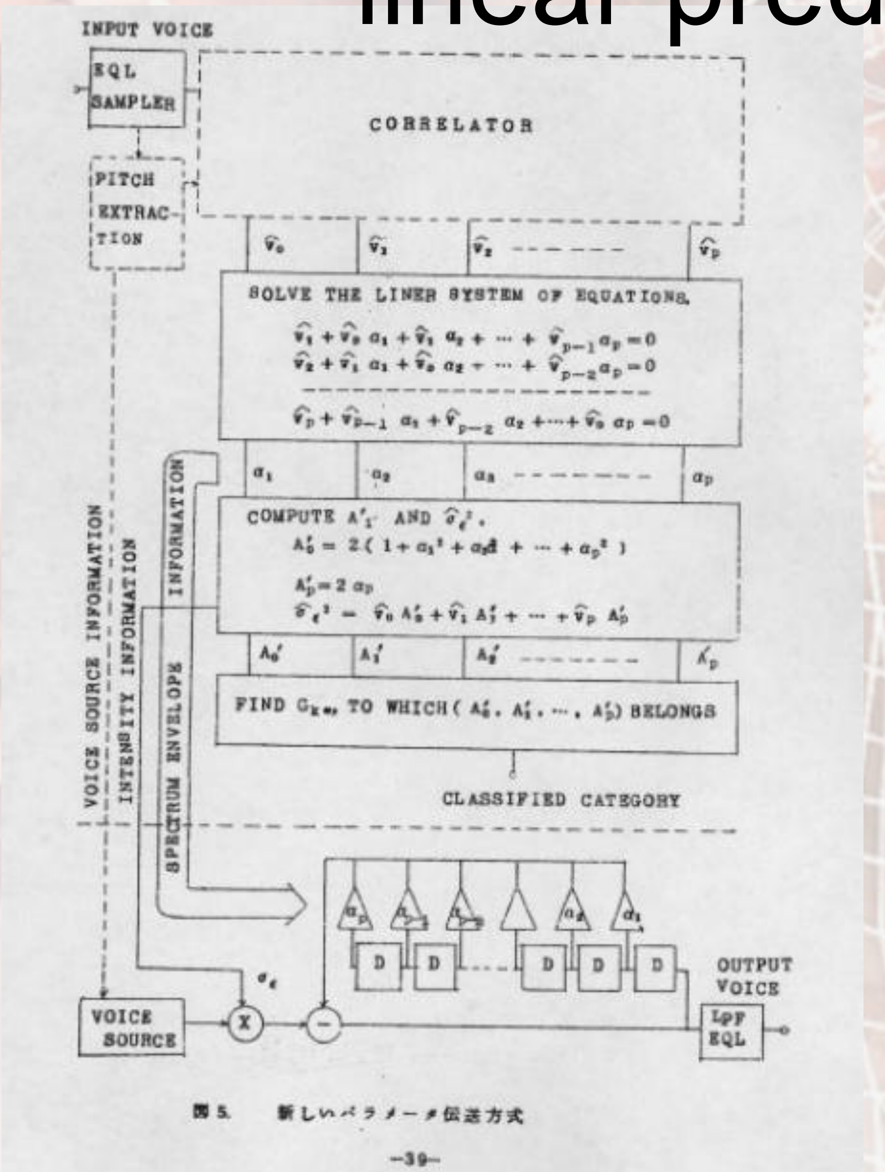


Beginnings of the application of linear prediction for speech

- **F.Itakura** i **S.Saito** working for NTT, Japan propose in 1966 speech coding procedure based on linear prediction.

- Fumitada Itakura

Beginnings of the application of linear prediction for speech

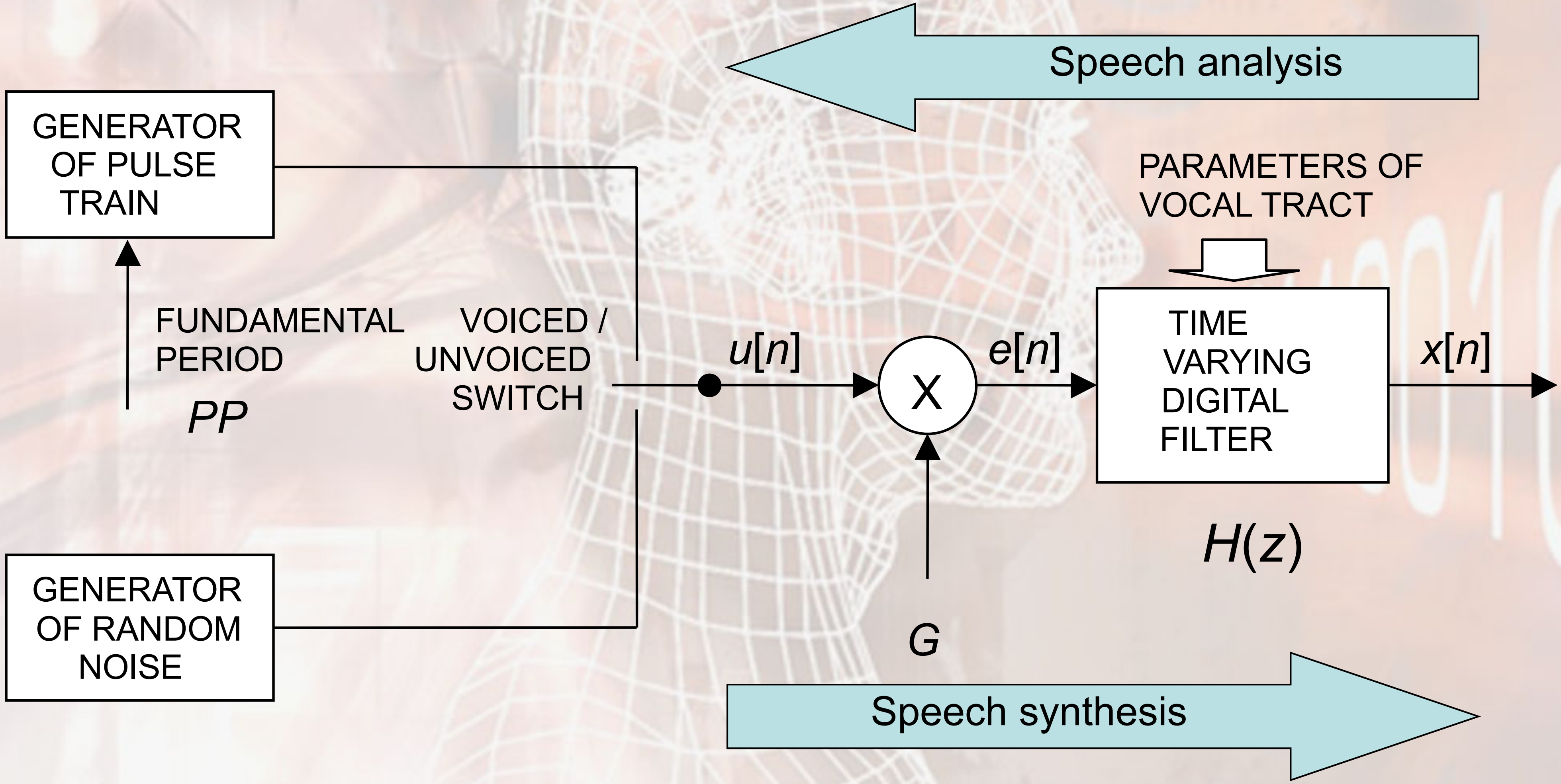


- Figure from work :
  - S. Saito and F. Itakura, “**The theoretical consideration of statistically optimum methods for speech spectral density**,” Report No. 3107, Electrical Communication Laboratory, NTT, Tokyo, December 1966.

LPC VOCODER

- Speech can be successfully "imitated" by passing a time-discrete excitation signal through a digital filter with several resonant characteristics.
- The excitation signal is:
  – a periodic series of pulses that simulates the vocal cords vibrations, or
  – a noise-like signal that simulates the excitation of the vocal tract for unvoiced sounds.
- The parameters of the model are:
  – filter properties (positions and widths of formants),
  – amplitude and type of excitation signal (voiced / unvoiced),
  – period of impulse train for voiced sounds.

LPC VOCODER


Speech analysis
GENERATOR OF PULSE TRAIN
PARAMETERS OF VOCAL TRACT
FUNDAMENTAL PERIOD
PP
VOICED / UNVOICED SWITCH
u[n]
X
e[n]
TIME VARYING DIGITAL FILTER
x[n]
H(z)
GENERATOR OF RANDOM NOISE
G
Speech synthesis

LPC VOCODER

- Vocoder is not trying to accurately represent the waveform of speech!
- The only requirement is that the speech synthesized by the described model **sounds as similar as possible** to the input speech signal.
- Since the speech samples themselves are not coded at all, but only the parameters of this model are used for synthesis on the decoder side, Vocoder belongs to the class of **parametric source coders**.

LPC VOCODER

- The $H(z)$ system "mimics" the vocal tract, i.e., short-term correlations in the speech signal.
- Speech evolves naturally in time, so correlations change over time as well (from sound to sound).
- The $H(z)$ filter must be adaptive and continuously track the correlation changes in the signal.
- The remaining model parameters: (voiced / unvoiced switch, gain G, and pitch period PP) also must be updated periodically according to changes in input speech.

Analysis and synthesis in the parametric speech coder

- The process of determining the "best" values of model parameters based on a short segment of the input speech signal is called **speech analysis**, i.e., the process of **estimating model parameters**.
- **Synthesis** is the reverse procedure where a short segment of a speech signal is formed by passing a selected excitation through a filter with chosen parameters.
- The analysis is performed within the encoder, which encodes the quantized parameters of the model and send them to the decoder, which then performs the described synthesis procedure.

Application of linear prediction for estimating Vocoder parameters

- Cookbook ...:
  - Apply the linear prediction analysis on a short segment of the speech signal (20-30ms) and find the predictor *P*(*z*) of order *p* described by the coefficients: $\overline{\alpha}_1, \overline{\alpha}_2, \ldots \overline{\alpha}_p$ which minimizes the energy of the prediction error *e*[*n*].
  - Use this predictor to calculate the prediction error signal *e*[*n*] and determine its variance: $\sigma_e^2(\overline{\boldsymbol{\alpha}})$
  - As the model gain factor *G*, select the second root of this variance: ... $G = \sigma_e(\overline{\boldsymbol{\alpha}})$

Application of linear prediction for estimating Vocoder parameters

- Cookbook ... continued:
  - Investigate whether there are periodicities in the prediction error signal, and if so, determine their pitch period *PP*.
  - The position of the voiced/unvoiced switch is set depending on the identified periodicity.
  - For this purpose, the **autocorrelation** of the prediction error signal is most often calculated, and a time-lag is sought with which a good match of the signal with its delayed replica is achieved.

Application of linear prediction for estimating Vocoder parameters

- Cookbook ... continued:
  - The estimated parameters of the model are quantized and coded in a convenient way and inserted into the output coded message.
  - The whole process is repeated on a new segment of speech, with a “focus point” advance of typically 10 or 20 ms.
  - The frequency of analysis is the reciprocal value of this time shift and is commonly referred to as **analysis frame rate** (typically in the range of 50 to 100 frames per second).

Decoding procedure in Vocoder

- Decoder cookbook:
  - Decode the received message and reconstruct all model parameters.
  - Depending on the voicing information (periodicity) form a segment of unit variance excitation signal consisting of:
    - the regular sequence of unit pulses at the spacing of PP samples, or random white noise.
  - Multiply this signal by the model gain factor G, which gives the “synthetic” signal $e[n]$ at the filter input $H(z)$.

Decoding procedure in Vocoder

- Decoder cookbook… continued:
  - This synthetic excitation signal is passed through a recursive digital filter *H*(*z*) modeling the vocal tract, with quantized parameters determined in the analysis procedure:

$$\hat{\alpha}_k = Q(\overline{\alpha}_k)$$

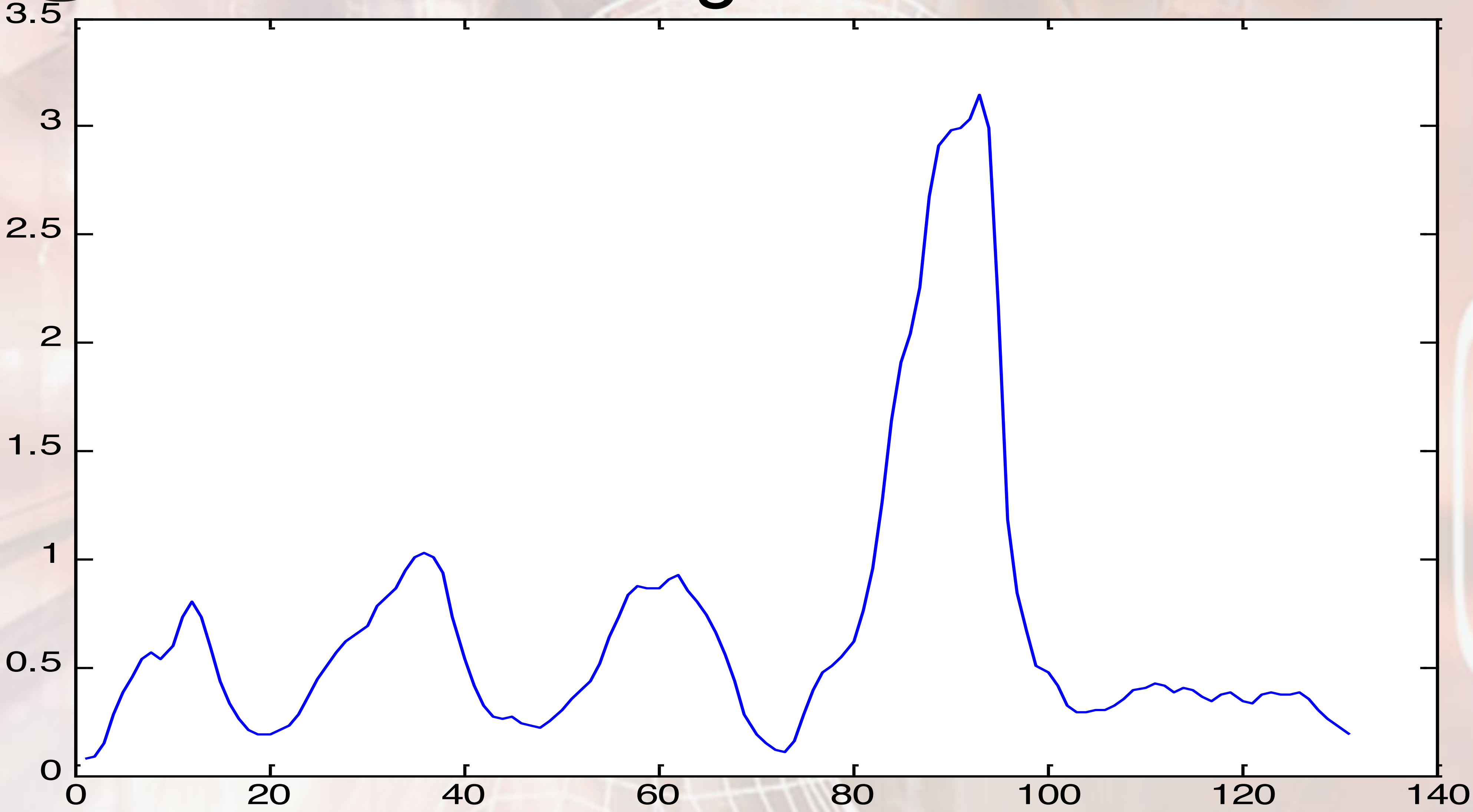
$$H(z) = \frac{1}{1 - P(z)} = \frac{1}{1 - \sum_{k=1}^{p} \hat{\alpha}_k z^{-k}}$$

  - Send the synthesized signal segment to the decoder output and repeat the procedure for the new frame.

Vocoder simulation in Matlab

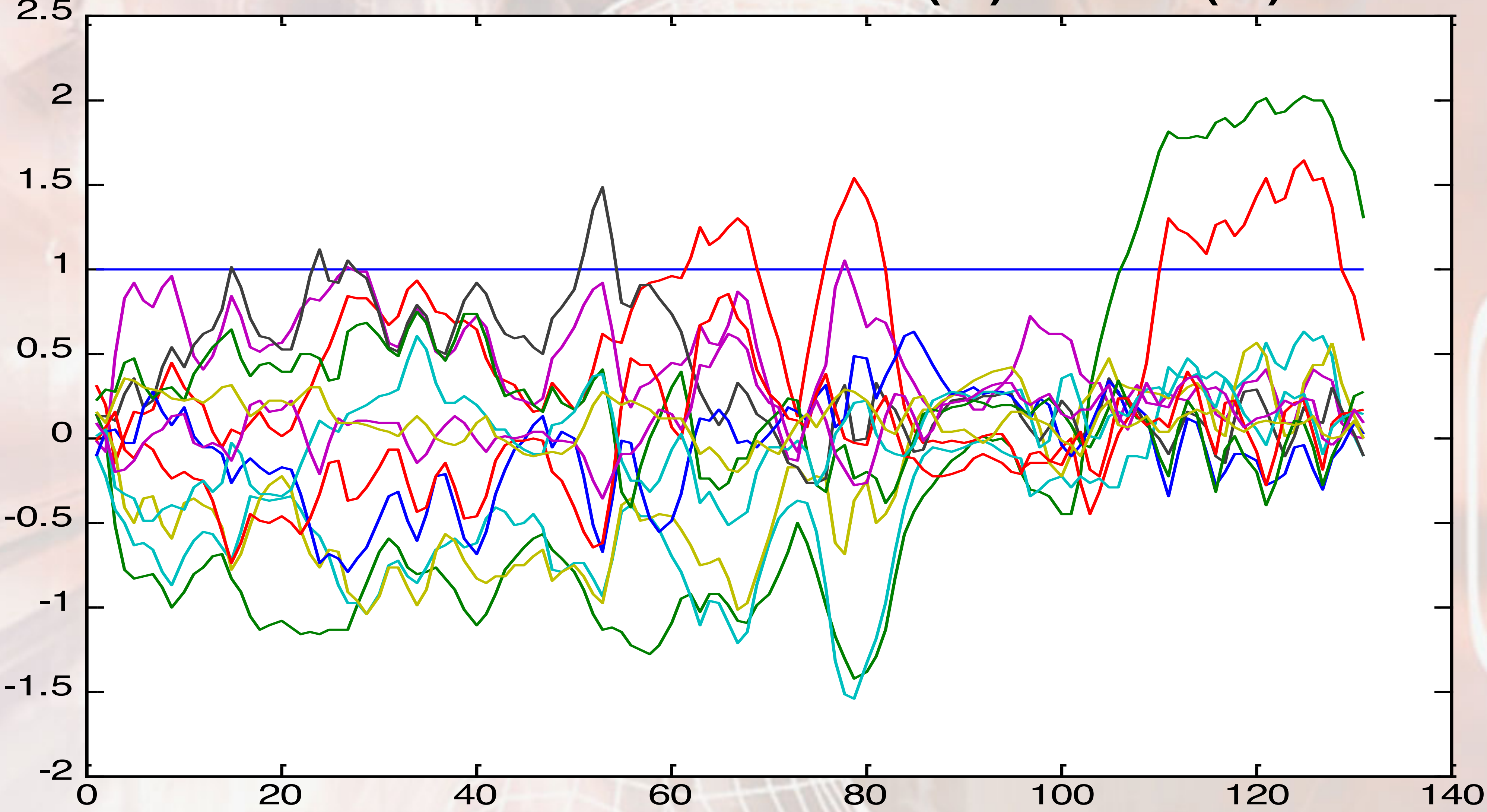
- Program used in Laboratory exercises in Digital Speech Processing:
  - sampling frequency f_s=11050 Hz,
  - analysis window width 46ms,
  - frame advance of 10ms (100 frames per sec),
  - predictor order p=12.
- The program performs analysis and synthesis, but without quantizing the model parameters - we investigate the limitations of the vocoder model itself !

Model gain term G



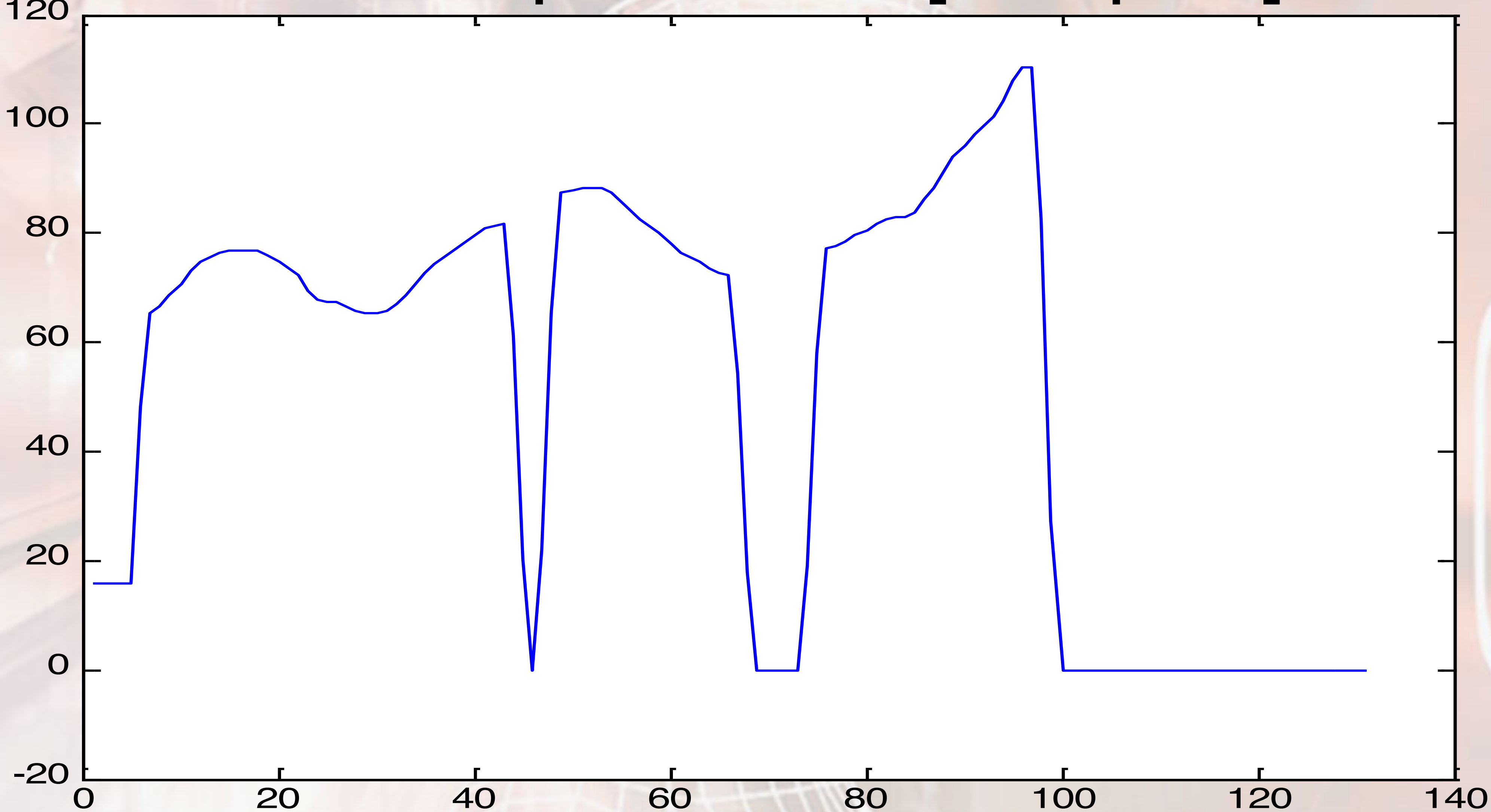
- O voje moj gla s

Coefficients of $A(z)=1-P(z)$



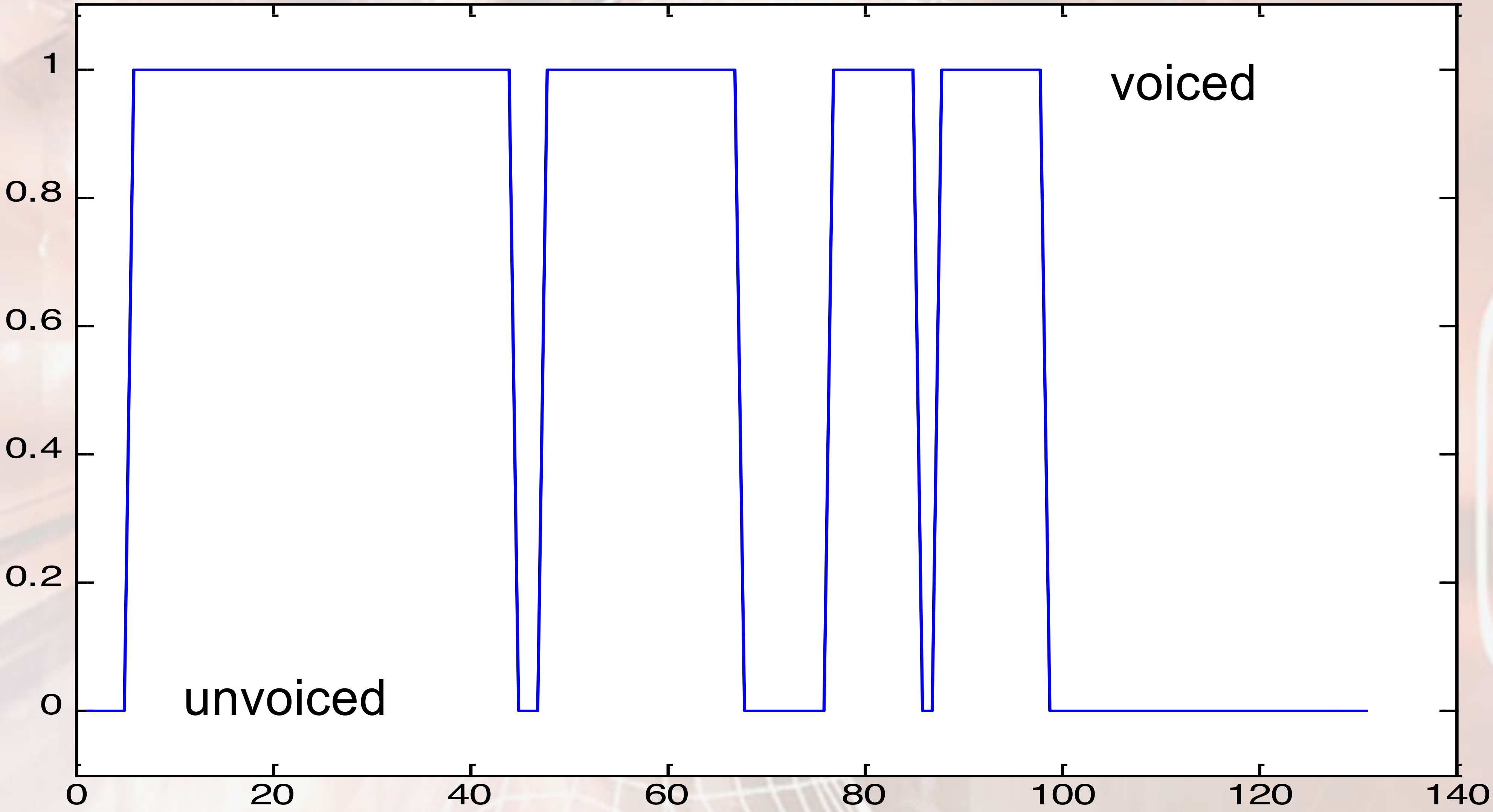
- O voje moj gla s

Pitch period *PP* [samples]



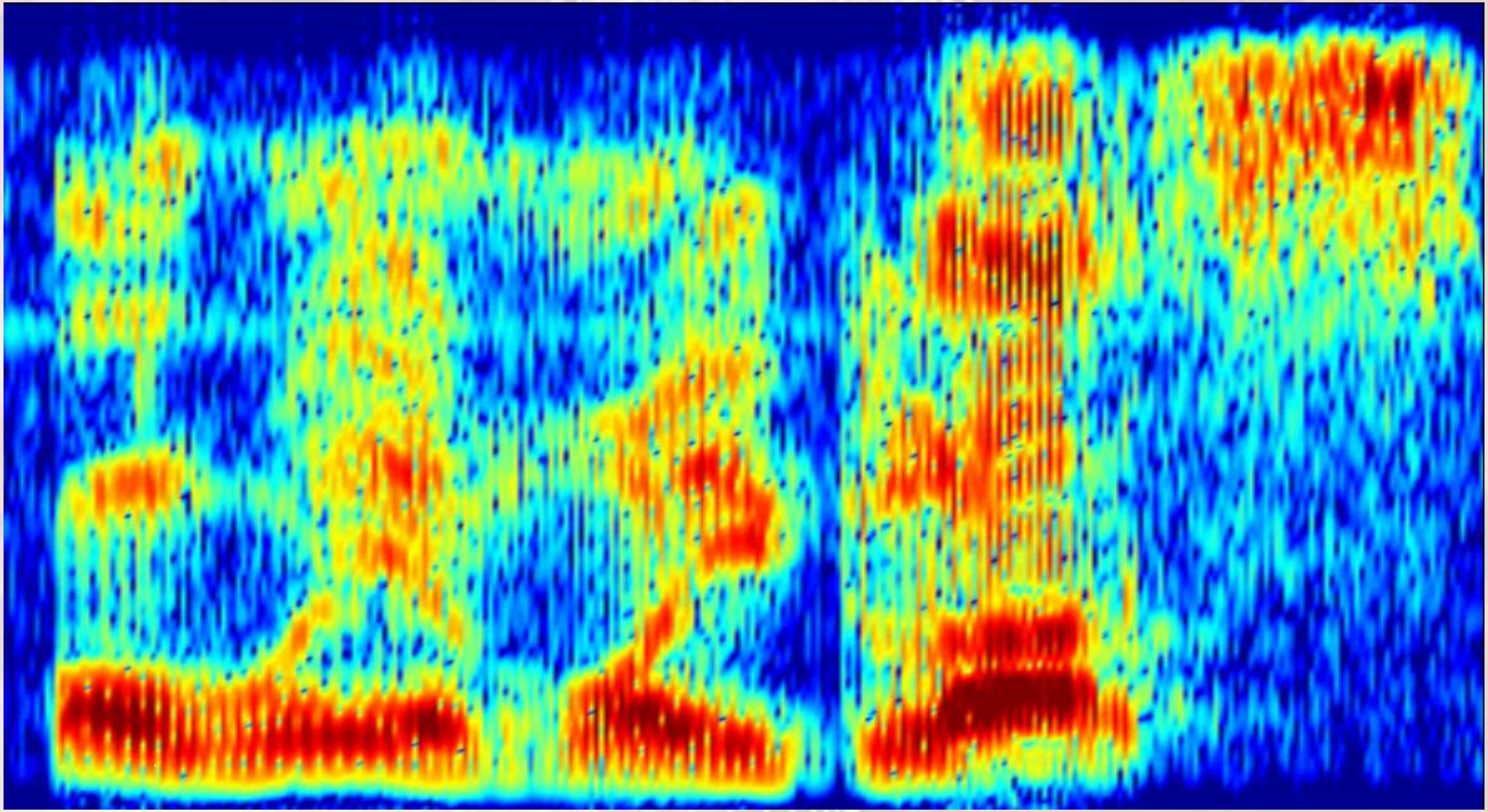
- O v o j e m o j g l a s

Voiced / Unvoiced



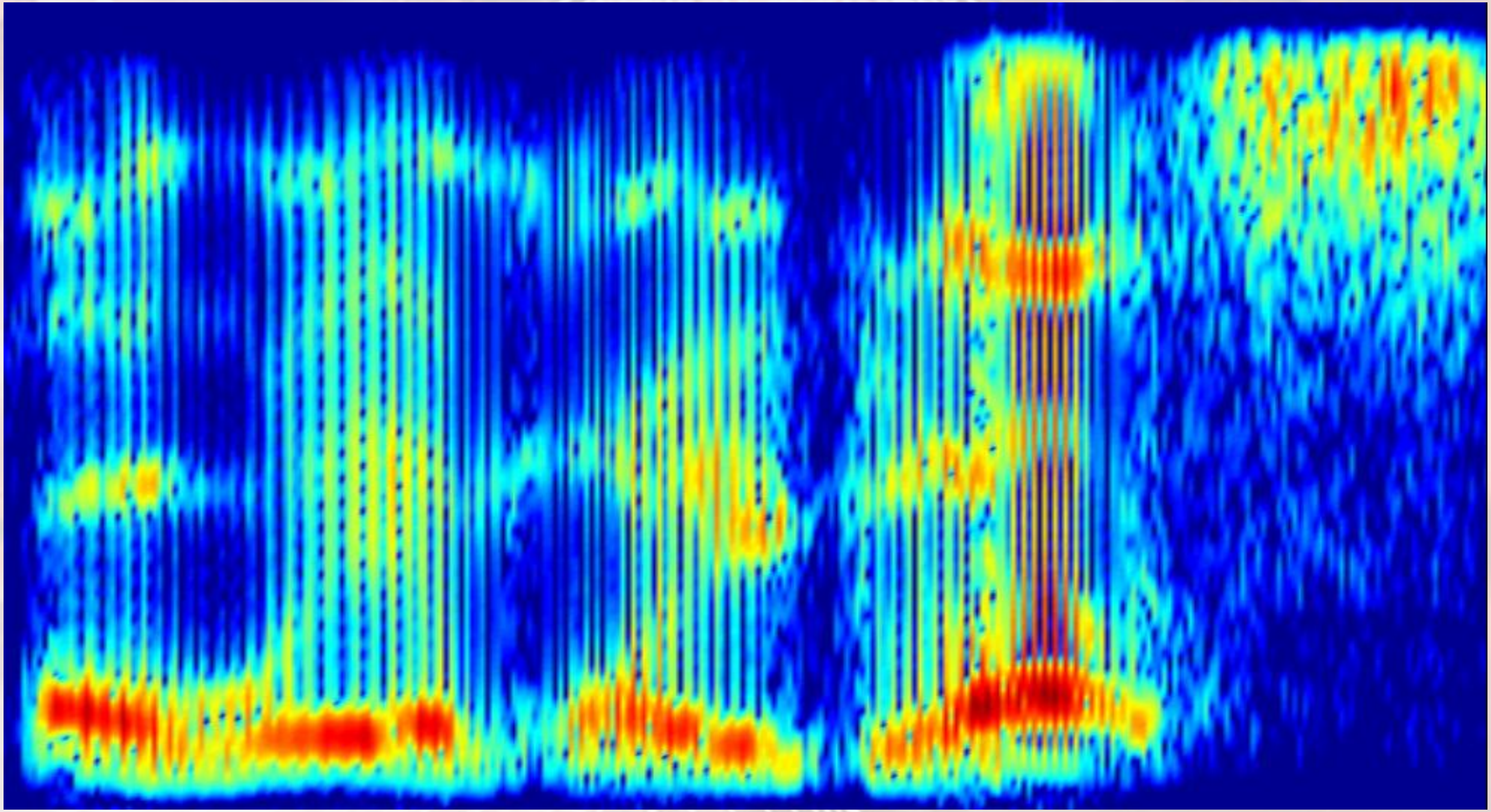
- O v o j e m o j g l a s

Wideband spec. of input speech



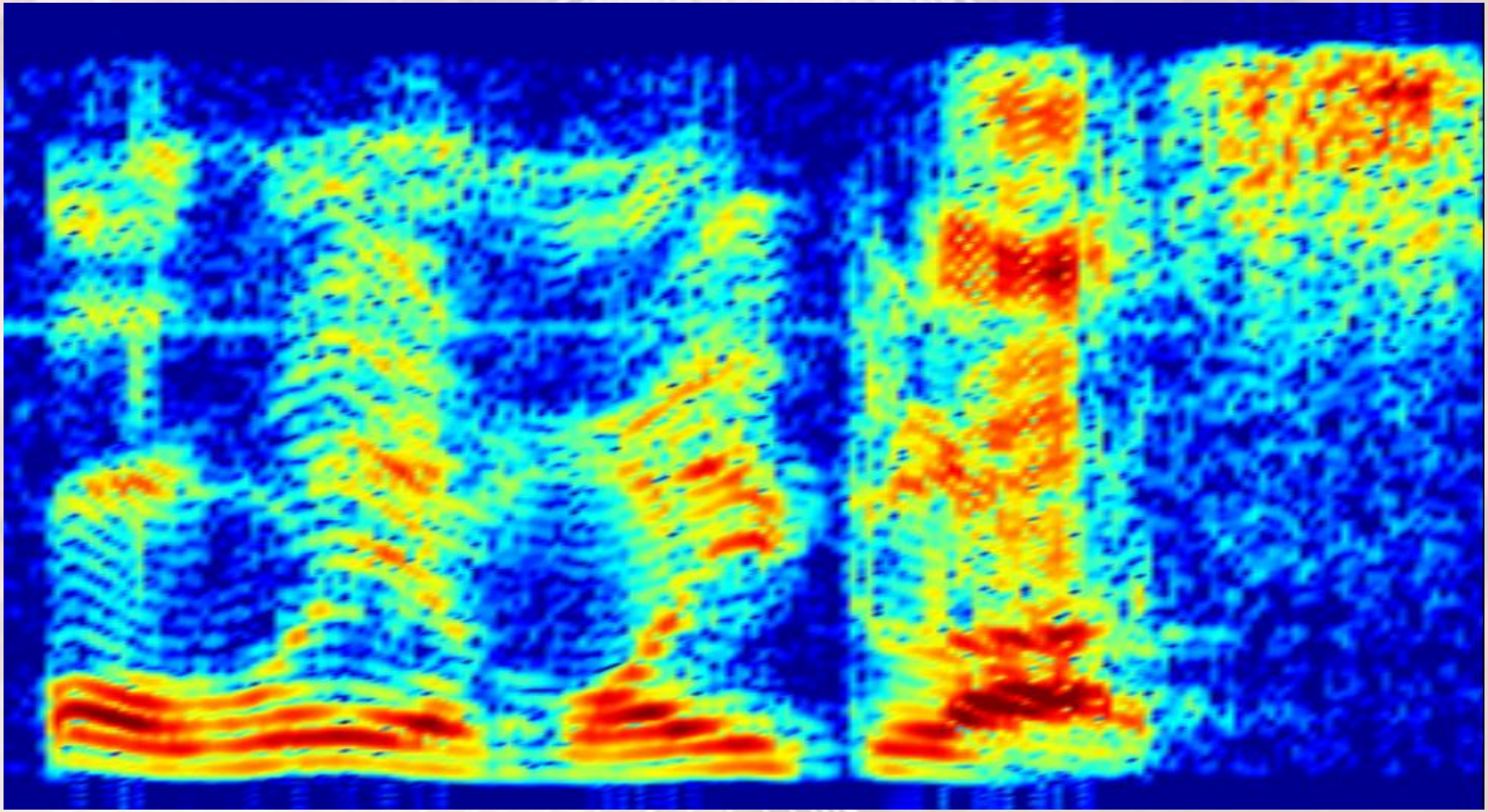
- O v oje m o j gl a s

Wideband spec. of LPC vocoder



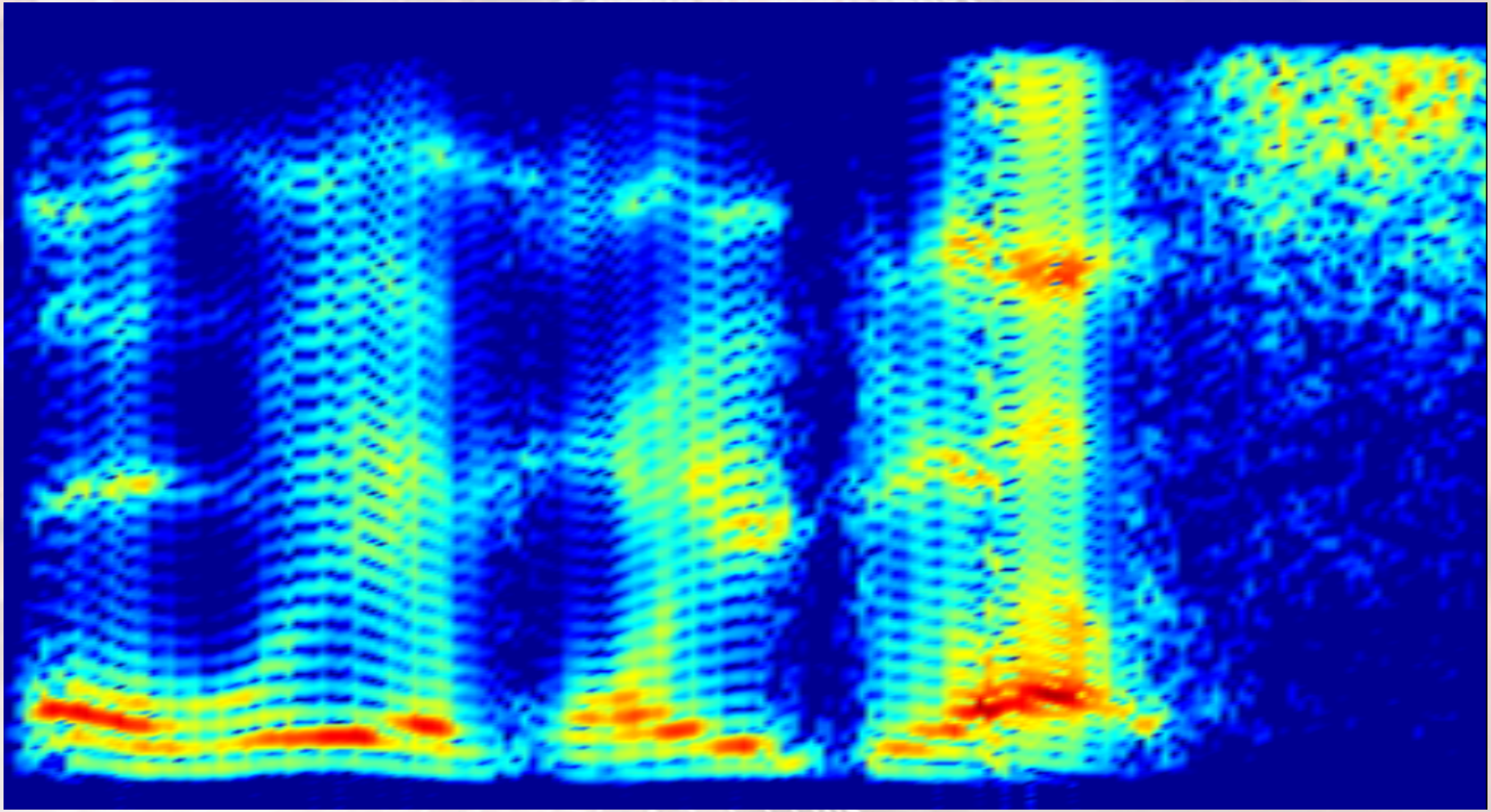
- O v oje m o j gl a s

Narrowband spec. of input speech



• O v oje m o j gl a s

Narrowband spec. of LPC voc.



- O v oje m o j gl a s

Vocoder simulation in Matlab

- Speech occasionally sounds synthetic, but it is understandable!
- The parametric model allows various manipulations of this signal:
  - we repeat each set of parameters twice and get a signal twice as long
  - we reject every other set of parameters and get a signal twice as short
  - we falsely double the pitch period *PP*
  - we falsely halve the pitch period *PP*

LPC-10 (FS-1015)

- One of the first standards for speech coding was based on the Vocoder model.
- A standard developed for the US DoD, later adopted by NATO, for security communication purposes.
- It uses a 10-order LPC predictor (... hence the name).
- Input speech is sampled with f_s=8kHz.
- The data rate of the input signal of nominal accuracy is R=64kbit/s (8 bit per sample).
- Output (coded) data rate is R=2.4kbit/s,
- ... achieves 27x compression.

LPC-10 (FS-1015)

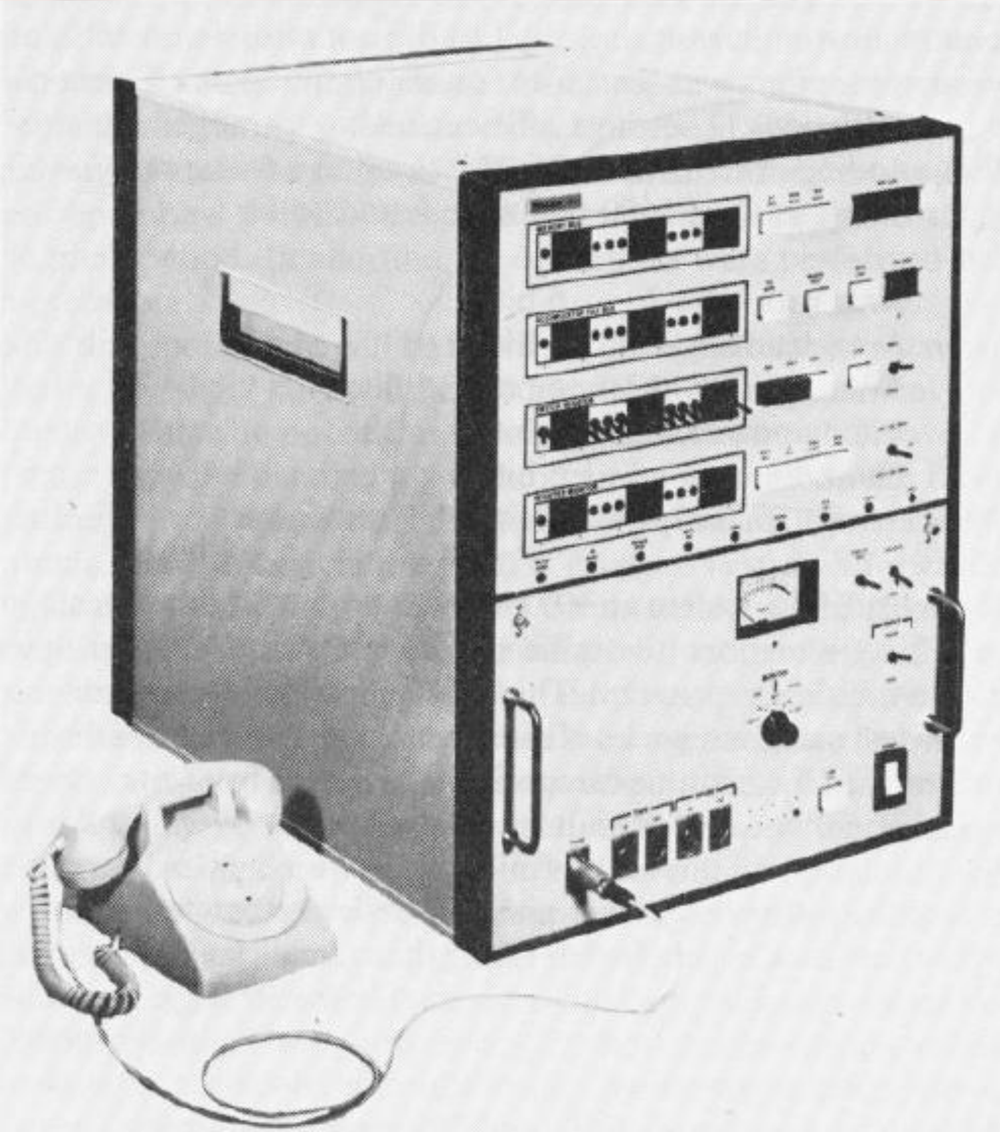
- Structure of the analysis / synthesis frame in LPC-10 with ***bit allocation***:

54 bits/frame Pitch + U/V->7bits G->5bits K1 a K4->5bits K5 a K8-> 4bits K9->3bits K10->2bits	Fs= 8000 samples/sec 54bits/frame 180 samples/frame (22.5 ms/frame) 54*8000/180=2400bits/sec

LPC-10 (FS-1015)

- Fortran and C source code for LPC-10 encoder available on web.
- e.g.,
  - http://www.speech.cs.cmu.edu/comp.speech/Section3/Software/celp-3.2a.html
- Speech quality is quite poor, but even today it is used in the category of encoders with very low output data rate ...
- newer revisions work with a rate of only 1.2kbit/s

LPC encoder for real-time operation



- Philco-Ford, LONGBRAKE II, 1974
- Sold only 4 copies (Navy, NSA)
- Weighs 120kg
- Uses **P**hilco-**F**ord **S**ignal **P**rocessor, **PFSP**

Vocoder

- Vocoder used in 1973 by Kraftwerk on the album “Ralf und Florian”

Secure communication telephone STU-III

- Based on the newer generation FS1016 CELP encoder

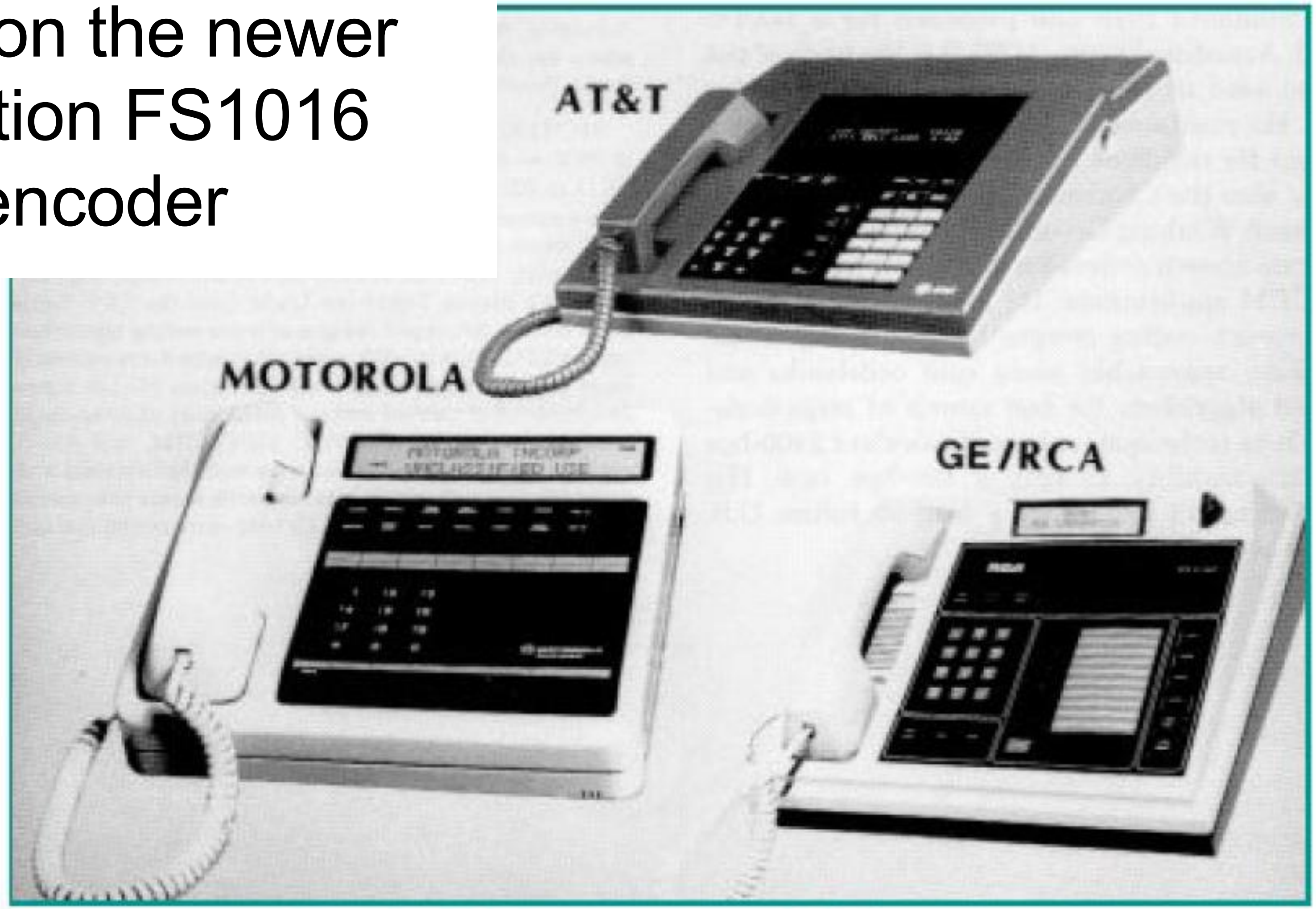


Fig. 4. The STU-III secure voice terminal family, circa 1986

What have we learned?

- correlations in speech and their sources
- formants in speech
- spectrogram; wideband, narrowband
- mechanical speech synthesizer
- electric speech synthesizer, Voder
- parametric coding, Vocoder
- analysis and synthesis in parametric encoders
- application of LPC in Vocoder
- example of Vocoder
- LPC-10 speech coding standard
- applications and actual devices