Chair of Computer Science 6
RWTH Aachen University
Prof. Dr.-Ing. H. Ney

Aachen, 08. 12. 2009

M. Nußbaum-Thom,
R. Schlüter

# Automatic Speech Recognition

## 6. Exercise

Submission of the solutions: 16. 12. 2009 at the beginning of the lecture

---

For the calculation, implementation and plotting of the following exercises you will need a mathematical toolkit. The public domain toolkit Octave is available at `http://www.octave.org`.

**Task 6.1** The signal from Task 1 is sampled at the frequency $f_S$ determined according to the sampling theorem. We apply a DFT of length $N$ where

- $N$ is a power of two and
- the distance between the position of two DFT coefficients is maximum $5$Hz.

Determine the minimal value for $N$. What range of sampling frequencies ($f_{min} \leq f_S \leq f_{max}$) do the constraints allow at the chosen value of $N$.

(2 P.)

**Task 6.2** Given the signal

$$s(t) = A_0 \cos(\omega_0 t) + A_1 \cos(\omega_1 t).$$

(a) Give a formula for the sampled discrete signal $s[n]$. (1 P.)

(b) Give a formula for the signal $v[n]$ which is generated by applying a window function $w[n]$ to $s[n]$. Determine the corresponding Fourier transform $V(e^{i\omega})$. (1 P.)

(c) Implement the Discrete Fourier Transform in a mathematical toolkit. The function has to support arbitrary signal length resp. arbitrary window length: `spectrum = dft(signal)`. (1 P.)

Following parameters are given further on in this exercise:

$$1/T_S = 10\text{kHz} \quad A_0 = 1 \quad A_1 = 0.75 \quad \omega_0 = \frac{2\pi}{14}10\text{kHz} \quad \omega_1 = \frac{4\pi}{15}10\text{kHz}$$

Use a rectangular window function of the length 64.

(d) Plot $v[n]$. (1 P.)

(e) Perform the DFT with length $N = 64$ and plot the real, the imaginary part and the amplitude of the spectrum. (1 P.)

(f) Interpret the curve of $|V[n]|$. (1 P.)

(g) Perform the DFT with length $N = 1024$ and plot the amplitude spectrum of $V[n]$ again. (Hint: use zero-padding.) (1 P.)

**Task 6.3**

For the convolution of two general signals $x_1$ and $x_2$

$$x_1[n] = \begin{cases} \neq 0 & \text{if } 0 \leq n \leq N \\ 0 & \text{else} \end{cases} \qquad x_2[n] = \begin{cases} \neq 0 & \text{if } 0 \leq n \leq N \\ 0 & \text{else.} \end{cases}$$

sketch the number of necessary operations as a function of $N$ for

- direct implementation, (0.5 P.)
- implementation with DFT and (0.5 P.)
- implementation with FFT. (0.5 P.)

Starting from which $N$ is the implementation with FFT faster than the other implementations?