



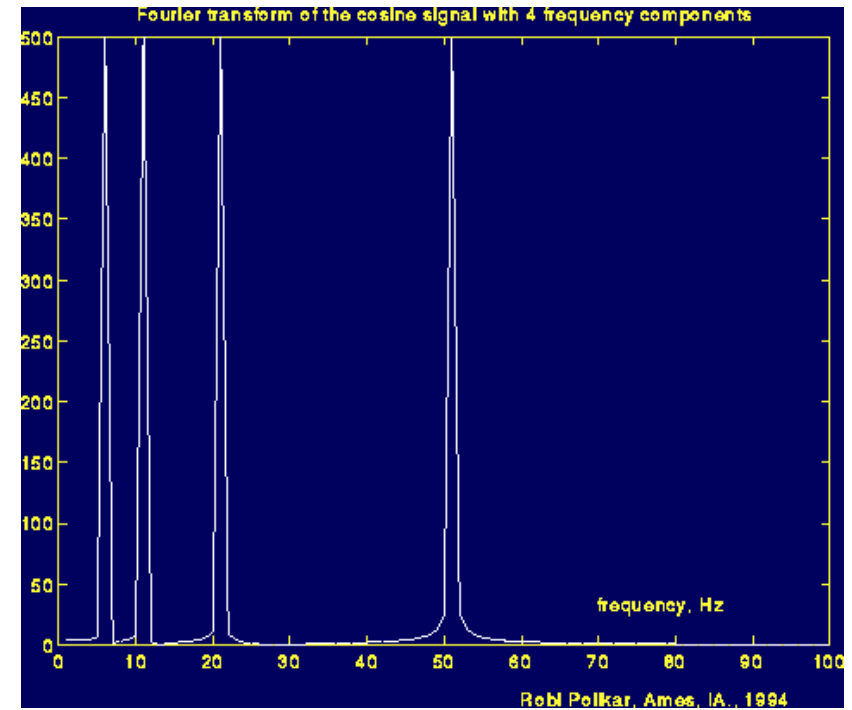
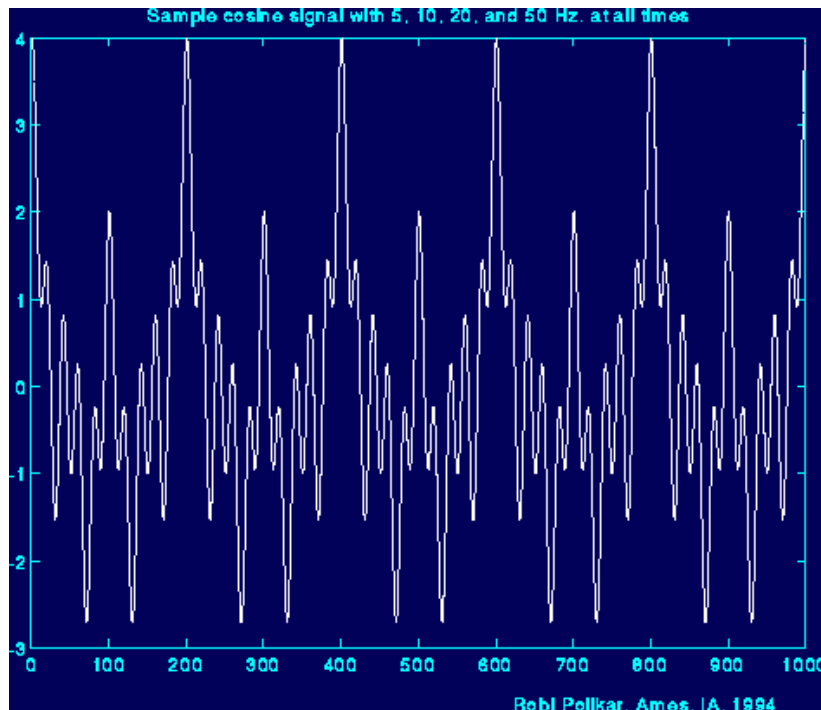
17.01.2019

Wavelets



17.01.2019

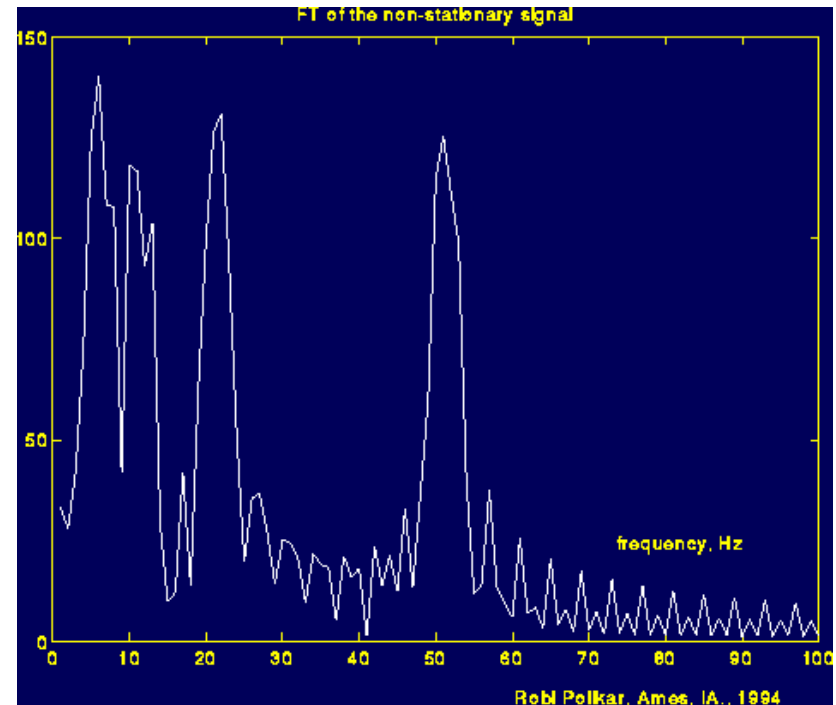
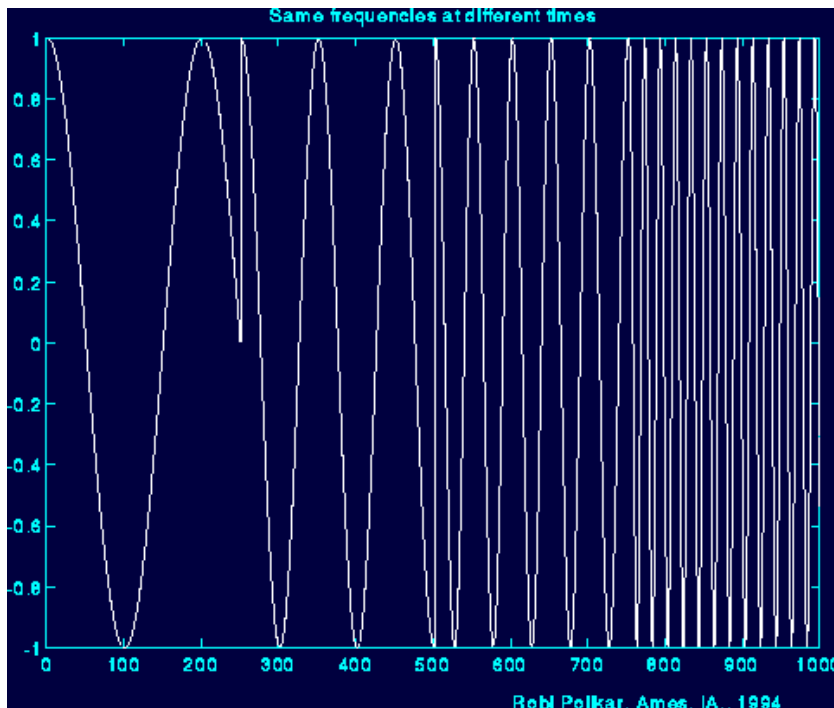
Fouriertransformation





17.01.2019

Fouriertransformation





Kurzzeit-Fouriertransformation

- Multiplikation mit Fensterfunktion und anschließend Fouriertransformation

$$F(\tau, \nu) = \int_{-\infty}^{\infty} f(t)w(t - \tau)e^{-2\pi i \nu t} dt$$

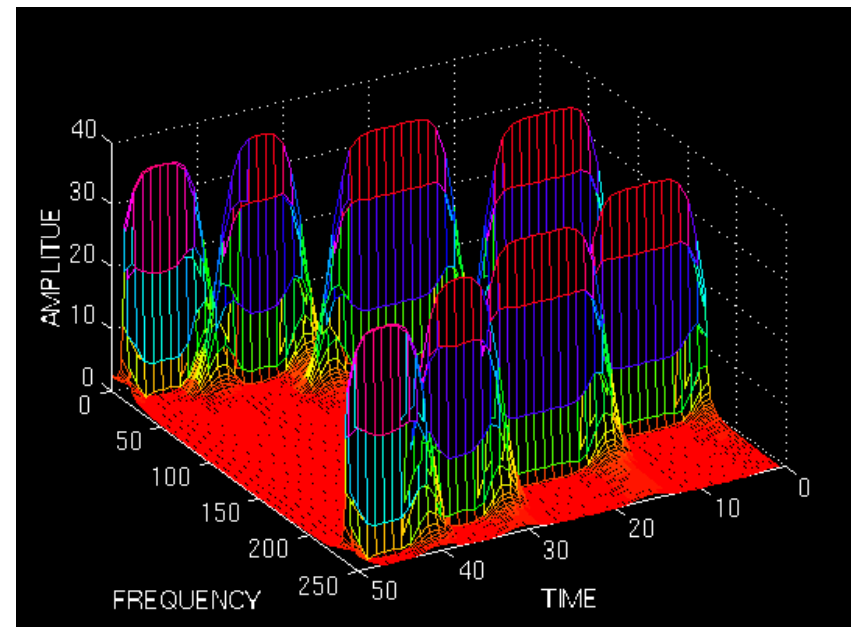
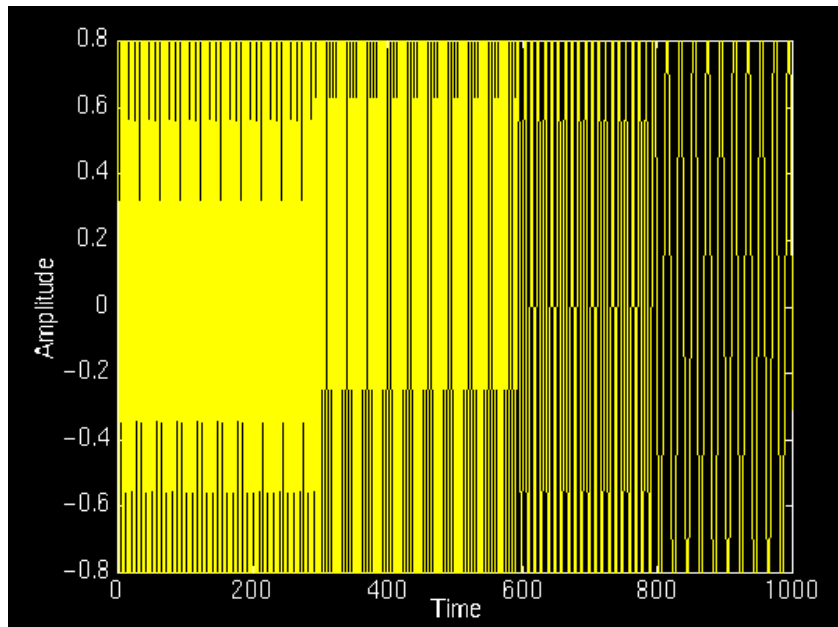
- Basisfunktionen:

$$g(\tau, \nu) = w(t - \tau)e^{-2\pi i \nu t}$$



17.01.2019

Kurzzeit-Fouriertransformation





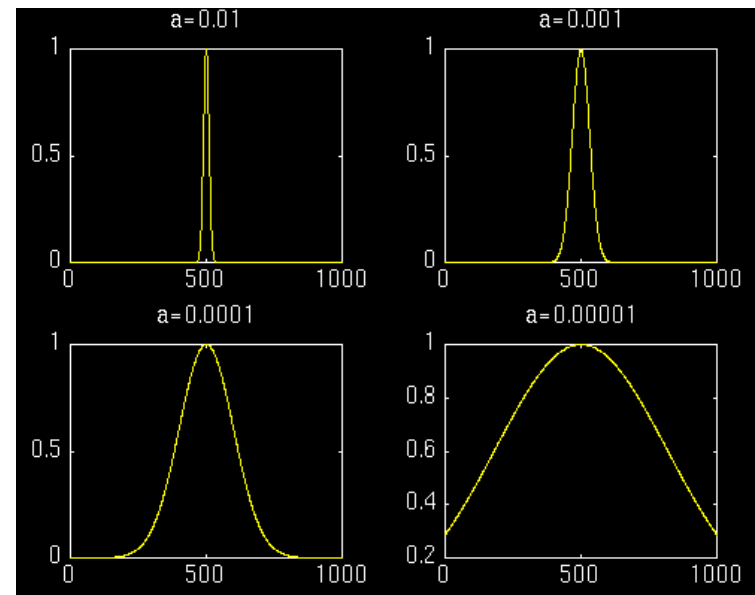
17.01.2019

Kompromiß:

- Zeitliche (örtliche) Auflösung gegen
- Frequenzauflösung

Verschieden breite
Fensterfunktionen

$w(t)$:





17.01.2019

Heisenberg'sche Unschärferelation:
Ortsauflösung $\leftrightarrow$ Frequenzauflösung

$$\begin{aligned}\Delta p \Delta x &\geq \frac{h}{2\pi} \\ \Delta E \Delta t &\geq \frac{h}{4\pi} \\ \Delta \nu \Delta t &\geq \frac{1}{4\pi}\end{aligned}$$

Griechisch ν entspricht Frequenz u



17.01.2019

Eine ähnliche Unschärferelation erhalten wir, wenn wir die Fouriertransformierte der Gaußglocke betrachten:

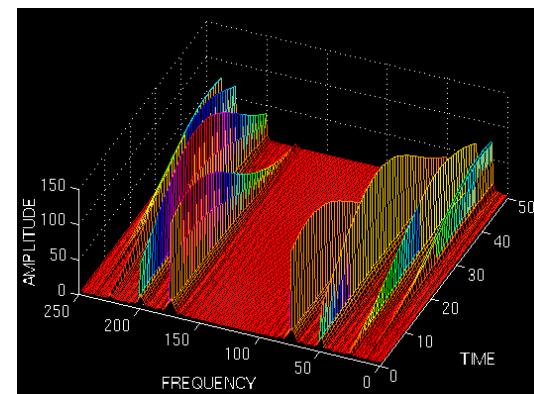
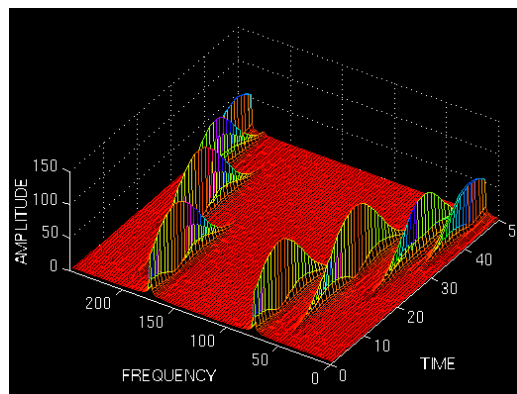
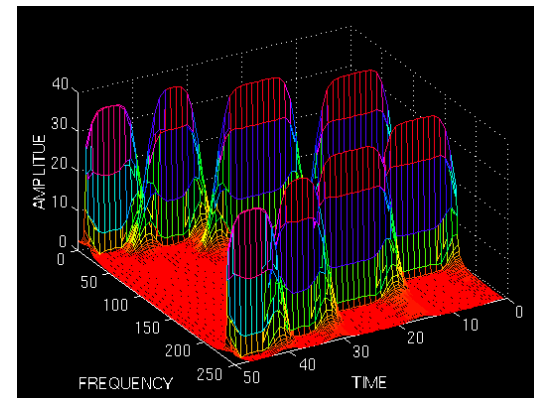
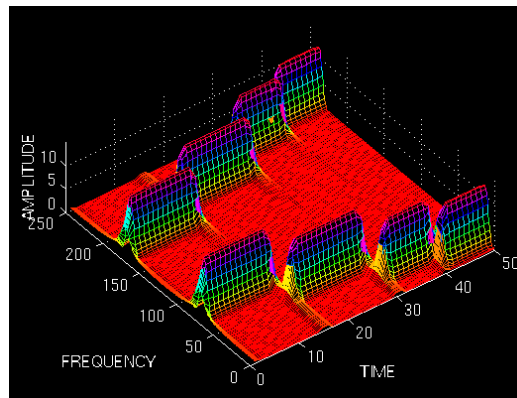
$$\Psi\left(\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{t^2}{2\sigma^2}}\right) = e^{-\frac{\sigma^2\omega^2}{2}} = e^{-\frac{(2\pi\sigma)^2 u^2}{2}} = e^{-\frac{u^2}{2\sigma'^2}}$$
$$\sigma' = \frac{1}{2\pi\sigma} ; \sigma\sigma' = \frac{1}{2\pi}$$

Grob gesagt, entspricht σ der Zeitunschärfe Δt und σ' der Frequenzunschärfe Δu . Der andere Vorfaktor ergibt sich durch eine andere Definition der Breite



17.01.2019

Zeitliche Auflösung <-> Frequenzauflösung

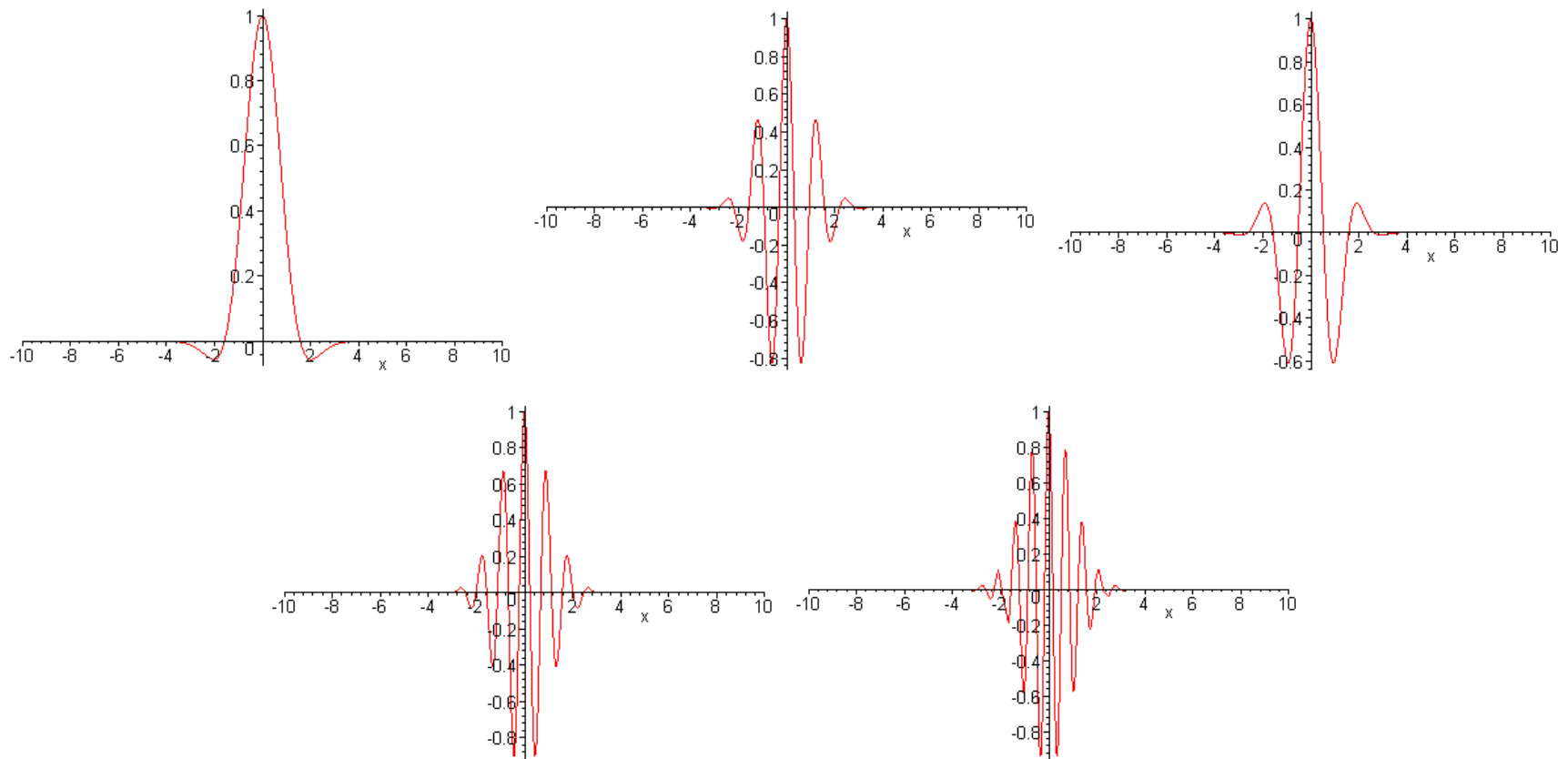




Kontinuierliche Wavelettransformation

17.01.2019

Konstantes Fenster





17.01.2019

Problem:

- konstantes Fenster enthält
 - Wenige Wellenzüge bei tiefen Frequenzen
 - Viele Wellenzüge bei hohen Frequenzen

Lösung:

- Fenstergröße variabel
 - Kleines Fenster bei hohen Frequenzen
 - großes Fenster bei tiefen Frequenzen
- Relative Frequenzauflösung konstant

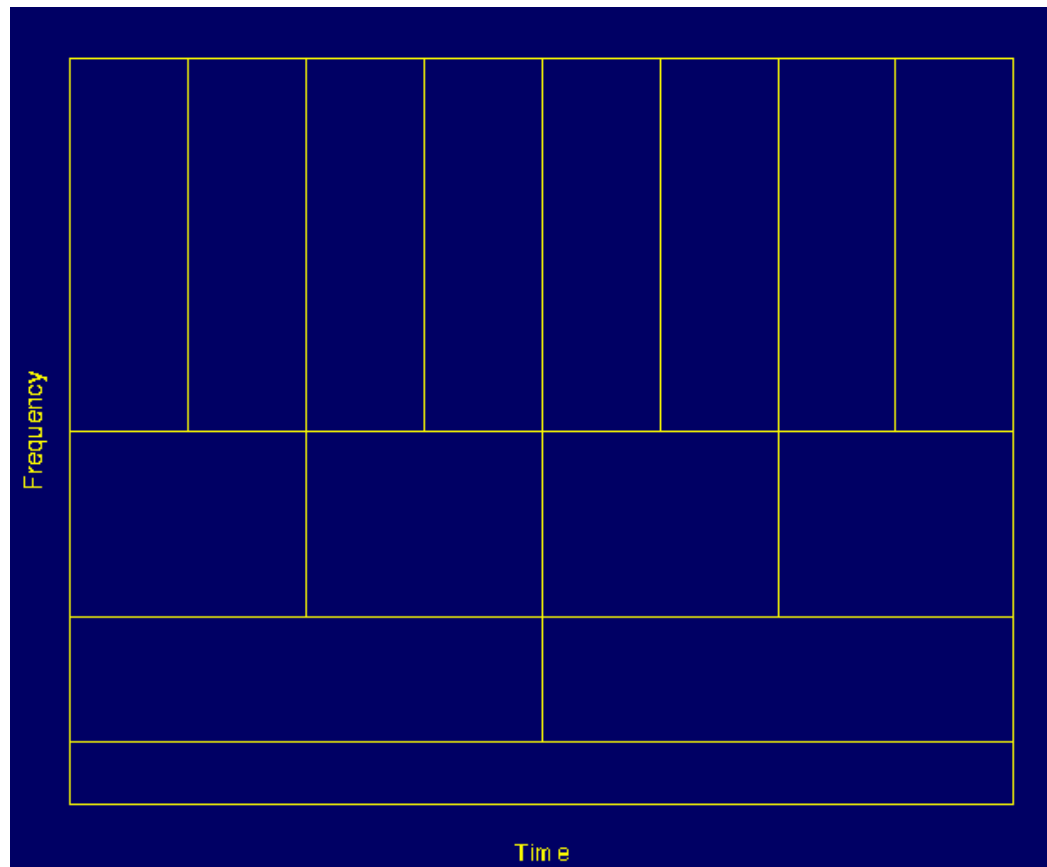
$$\frac{\Delta \nu}{\nu} = \text{const}$$



Kontinuierliche Wavelettransformation

17.01.2019

Frequenzauflösung $\leftrightarrow$ Zeitauflösung





Kontinuierliche Wavelettransformation

17.01.2019

Definition

$$F_w(s, \tau) = \int_{-\infty}^{\infty} f(t) \psi_{s, \tau}(t) dt$$

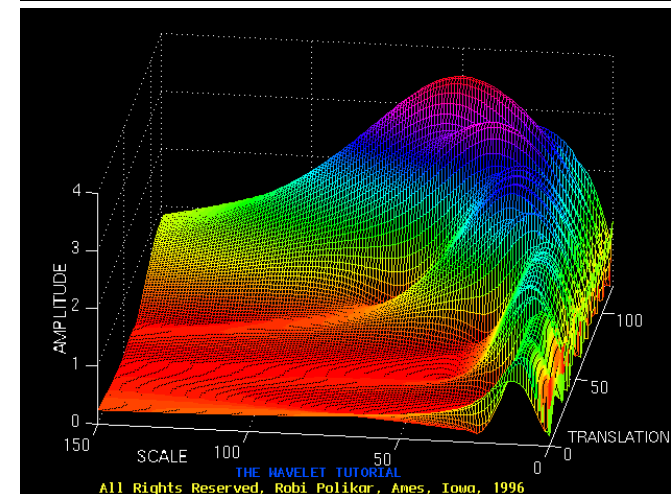
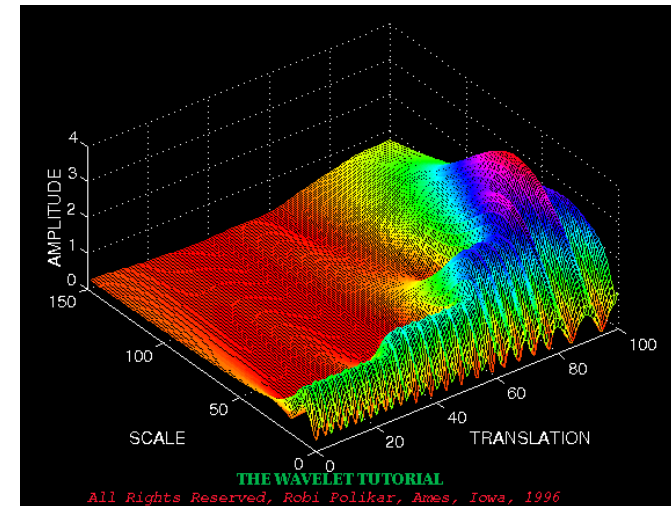
$$= \int_{-\infty}^{\infty} f(t) s^{-1/2} \psi\left(\frac{t - \tau}{s}\right) dt$$

- Scale s spielt Rolle der Frequenz ν (bzw. $1/\nu$)
- Analysefunktion: Wavelet

$$\psi_{s, \tau}(t) = s^{-1/2} \psi\left(\frac{t - \tau}{s}\right)$$

Kontinuierliche Wavelettransformation

17.01.2019





Kontinuierliche Wavelettransformation

17.01.2019

Kontinuierliche Wavelettransformation:

- Erstaunlich aber wahr:
Synthese mit Wavelets als Basisfunktionen:

$$f(t) = c \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F_w(s, \tau) \psi_{s, \tau}(t) ds d\tau$$

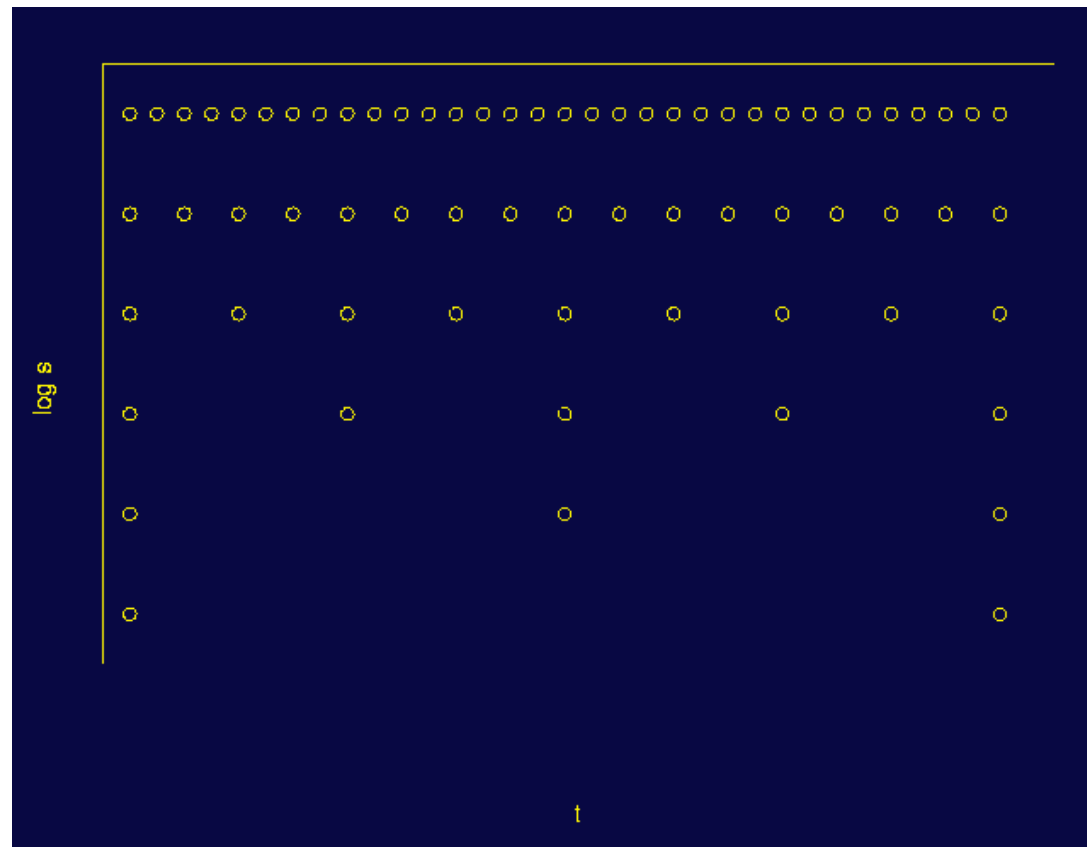
- Basisfunktionen sind nicht orthogonal
- Darstellung redundant
- Sampling bei Erhaltung der kompletten Information u. U. möglich!



Diskrete Wavelettransformation

17.01.2019

Sampling der Translation/Scale-Ebene





Diskrete Wavelettransformation

17.01.2019

Durch das Sampling erhalten wir diskrete Wavelets.

Aus:

$$\psi_{s,\tau}(t) = s^{-1/2} \psi\left(\frac{t-\tau}{s}\right)$$

wird:

$$\psi_{j,k}(t) = s_0^{-j/2} \psi(s_0^{-j}t - k\tau_0)$$



Diskrete Wavelettransformation

17.01.2019

Diskrete Wavelets

$$\psi_{j,k}(t) = s_0^{-j/2} \psi(s_0^{-j}t - k\tau_0)$$

Darstellung von $f(x)$ als Summe bei orthogonalen Wavelets:

$$f(t) = \sum_{j,k} F_w(j,k) \psi_{j,k}(t)$$



Diskrete Wavelettransformation

17.01.2019

Es gibt auch sog. biorthogonale Wavelets, bei denen die Analysefunktion nicht mit dem Wavelet selbst übereinstimmt

$$F_w(s, \tau) = \int_{-\infty}^{\infty} f(t) \tilde{\psi}_{s, \tau}(t) dt$$

aber die die Bedingung

$$\psi_{j,k} \tilde{\psi}_{l,m} = \delta_{jl} \delta_{km}$$

erfüllen.



Diskrete Wavelettransformation

17.01.2019

Von der kontinuierlichen zur diskreten WT:

Die Dyadische Wavelettransformation

- Sampling der Frequenz- bzw. Skalen-achse in Zweierpotenzen
- Gegeben: Wavelet mit der Eigenschaft:

$$\int \psi(t) dt = 0$$



Diskrete Wavelettransformation

17.01.2019

Wir definieren

$$\psi_i(t) = \frac{1}{2^{\frac{i}{2}}} \psi\left(\frac{t}{2^i}\right)$$

Die kontinuierliche Wavelettransformation mit scale 2^i ist dann gegeben durch

$$F_W(2^i, t) = f(t) \bullet \psi_i(t) = \int_{-\infty}^{\infty} f(\tau) \psi_i(t - \tau) d\tau$$



Diskrete Wavelettransformation

17.01.2019

Die dyadische Wavelettransformation ist dann die Sequenz von Funktionen

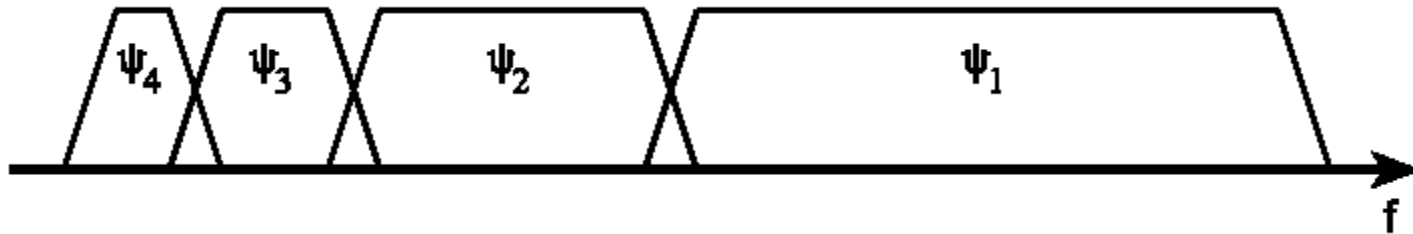
$$WF_{dyadic} [f(\cdot)] = F_W(2^i, t)$$

Werden diese Funktionen geeignet in der Zeit abgetastet, erhalten wir wieder eine diskrete Wavelettransformation

Diskrete Wavelettransformation

17.01.2019

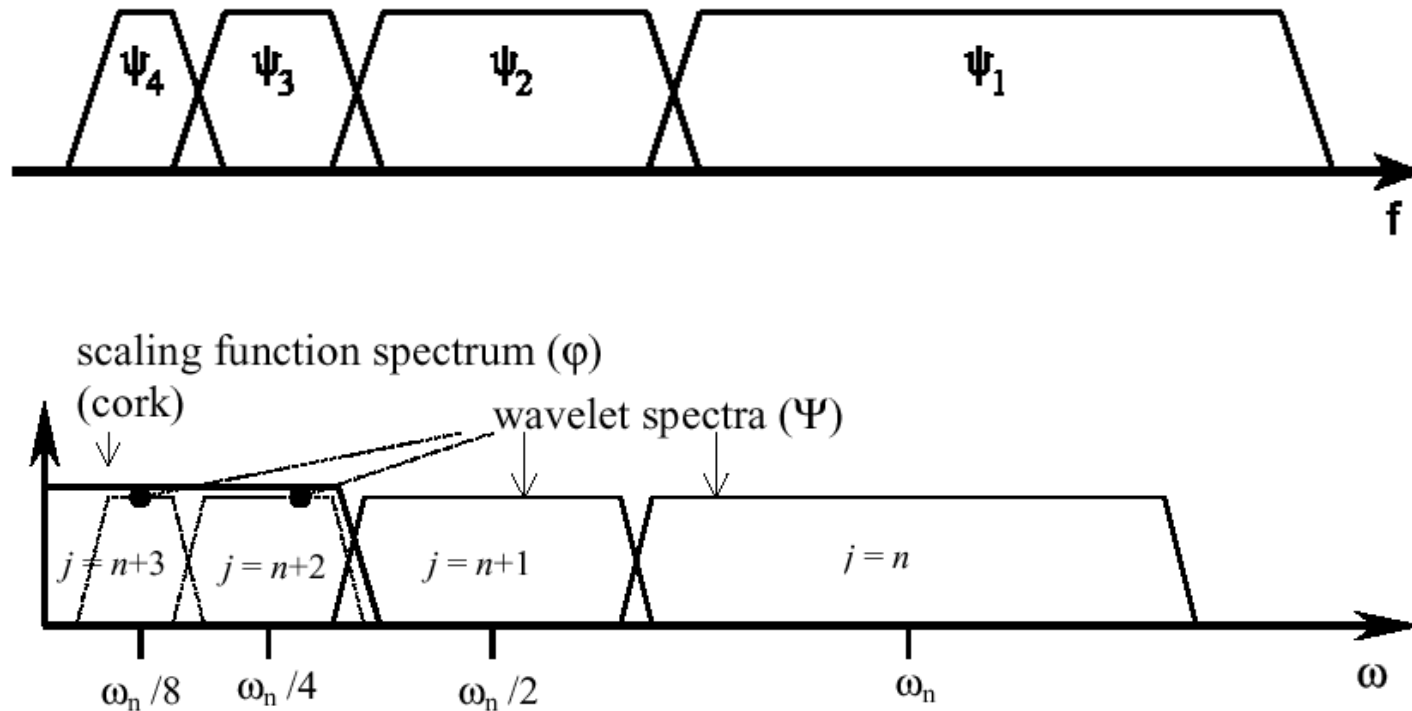
Alternative Darstellung durch Filter:



Diskrete Wavelettransformation

17.01.2019

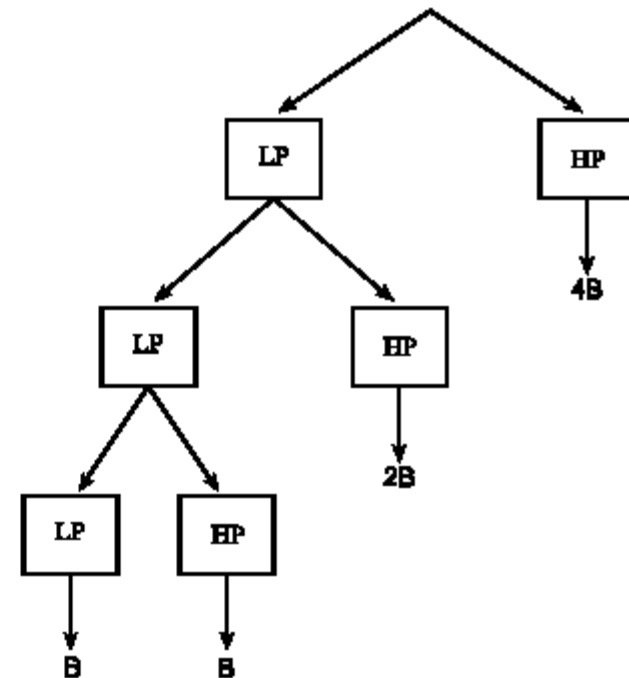
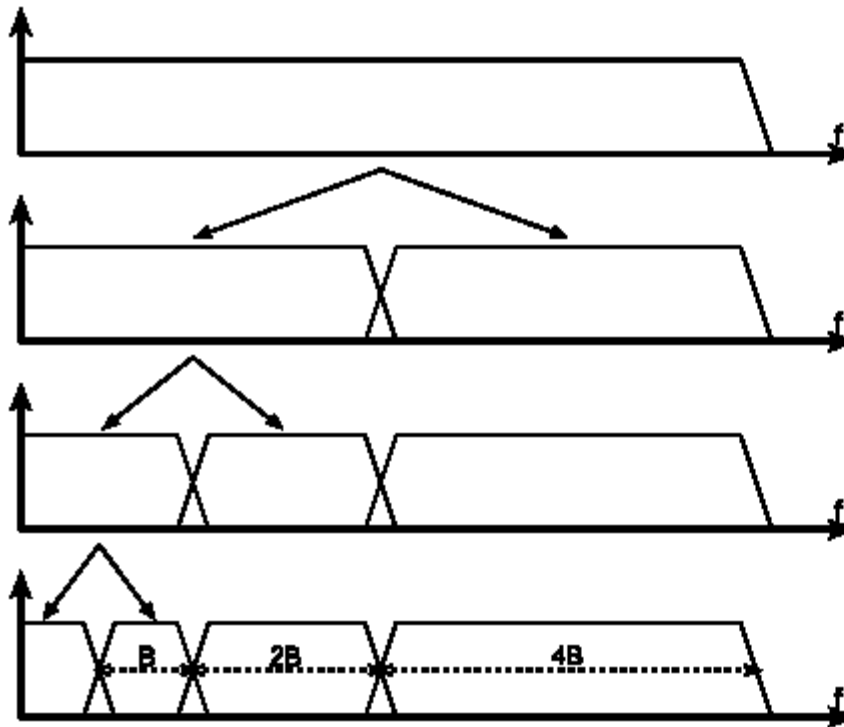
Ersatz der Bandpaßfilter (Wavelets) durch Tiefpaßfilter



Diskrete Wavelettransformation

17.01.2019

Ersatz der Bandpaßfilter (Wavelets) durch Tiefpaß- und Hochpaßfilter

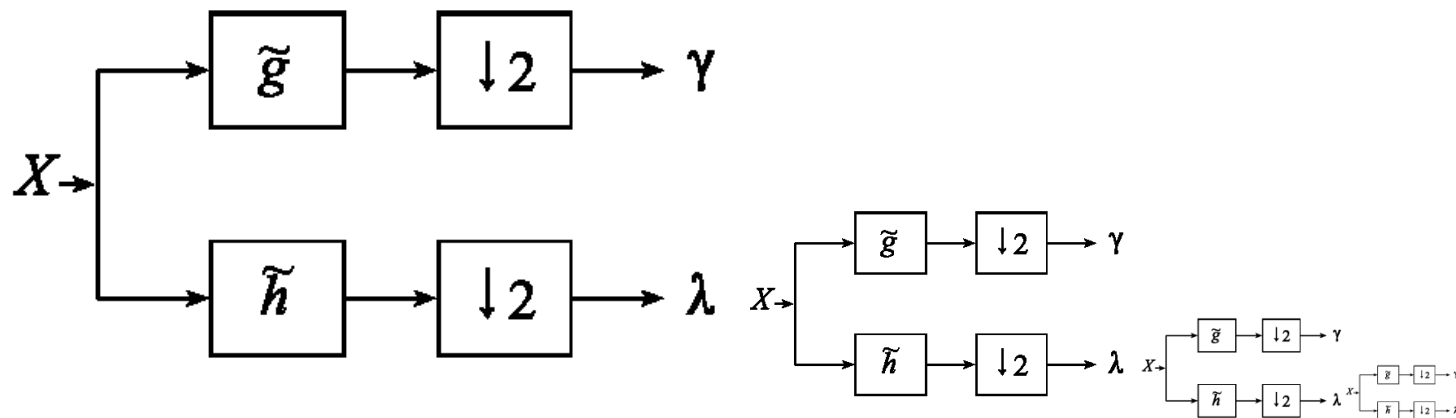




Lifting

17.01.2019

- Wavelettransformation durch rekursive Anwendung von Filtern und Downsampling:

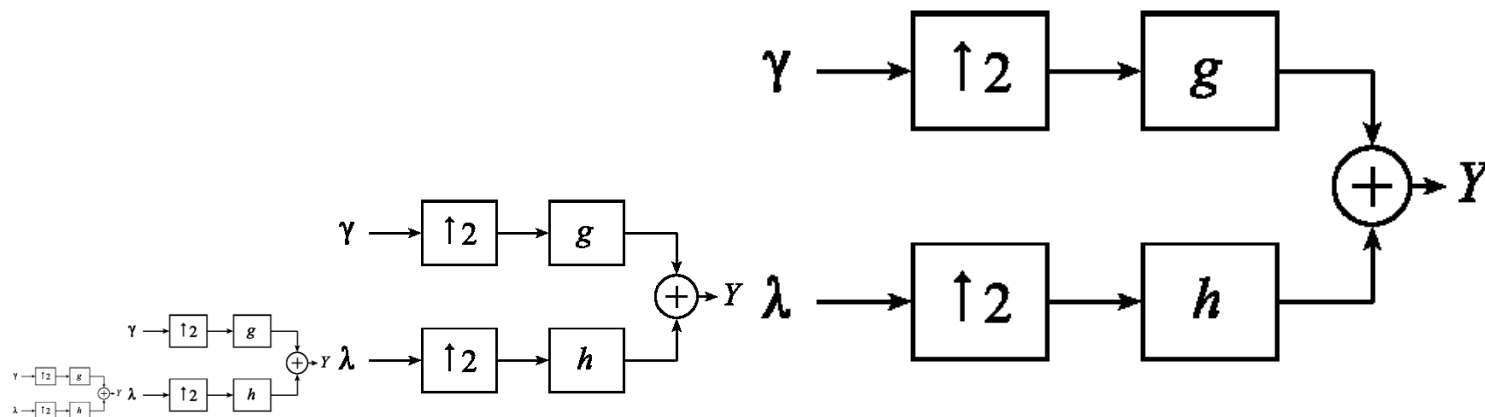




Lifting

17.01.2019

- Rekonstruktion durch Upsampling und Filtern:

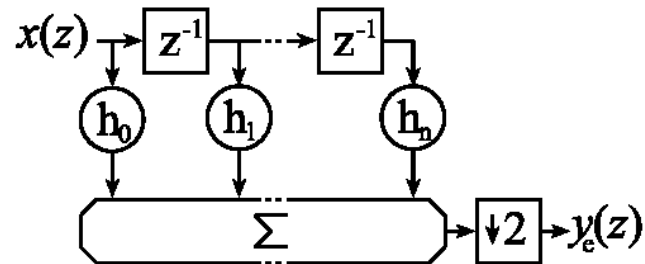




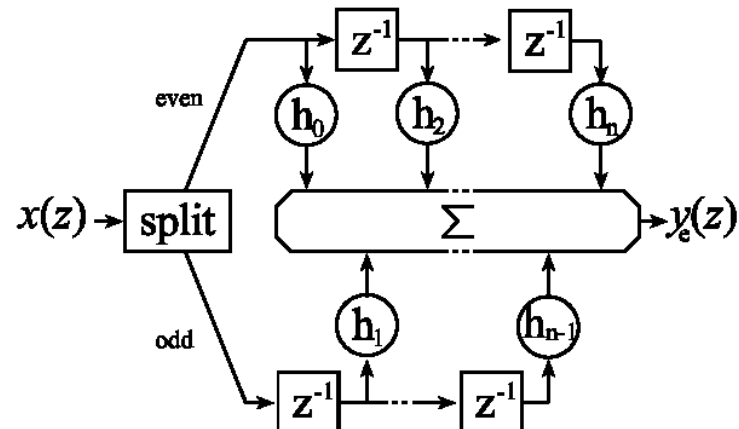
Lifting

17.01.2019

Standardfilter mit Downsampling



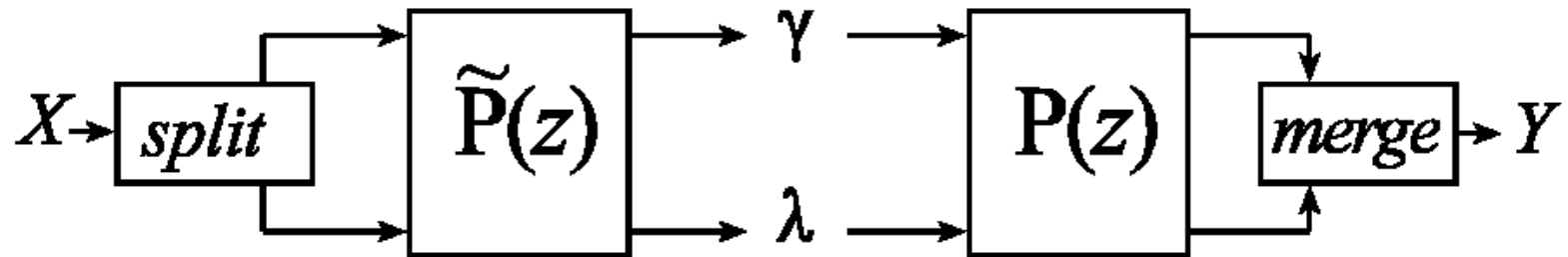
Effizientere Implementierung



Lifting

17.01.2019

Gleiches Prinzip bei Rekonstruktion
Filter- und Rekonstruktionsstufe:



Darstellung durch Matrix :

$$\begin{pmatrix} \lambda(z) \\ \gamma(z) \end{pmatrix} = \tilde{\mathbf{P}}(z) \begin{pmatrix} x_e(z) \\ z^{-1}x_o(z) \end{pmatrix}$$



Lifting

17.01.2019

Lifting:

- Aus $P(z)$ wird neues Filterpaar erzeugt:

$$g^{new}(z) = g(z) + h(z)s(z^2), \quad (16)$$

where $s(z^2)$ is a Laurent polynomial. This can be seen very easily if we write g^{new} in polyphase form (see also the footnote with equation (12)) and assemble the new polyphase matrix as

$$\mathbf{P}^{new}(z) = \begin{pmatrix} h_e(z) & h_o(z)s(z) + g_e(z) \\ h_o(z) & h_o(z)s(z) + g_o(z) \end{pmatrix} = \mathbf{P}(z) \begin{pmatrix} 1 & s(z) \\ 0 & 1 \end{pmatrix}. \quad (17)$$

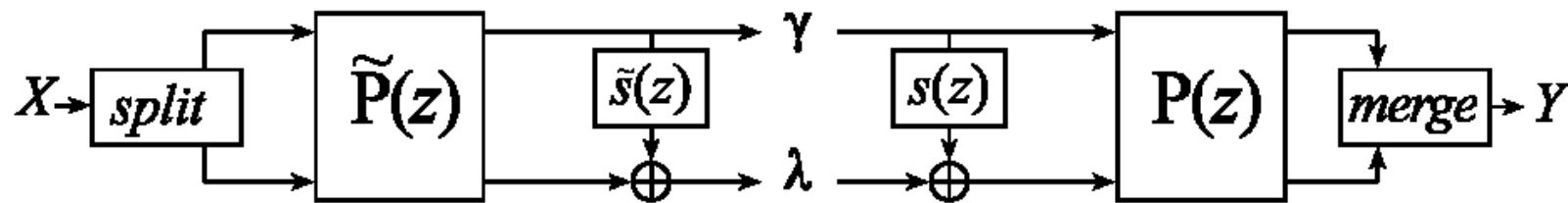
As can be easily verified the determinant of the new polyphase matrix also equals 1, which proofs (16).

Similarly, we can apply the lifting theorem to create the filter $\tilde{h}^{new}(z)$ complementary to $\tilde{g}(z)$ (recall the identities from (11))

$$\tilde{h}^{new}(z) = \tilde{h}(z) + \tilde{g}(z)\tilde{s}(z^2), \quad (18)$$

with the new dual polyphase matrix given by

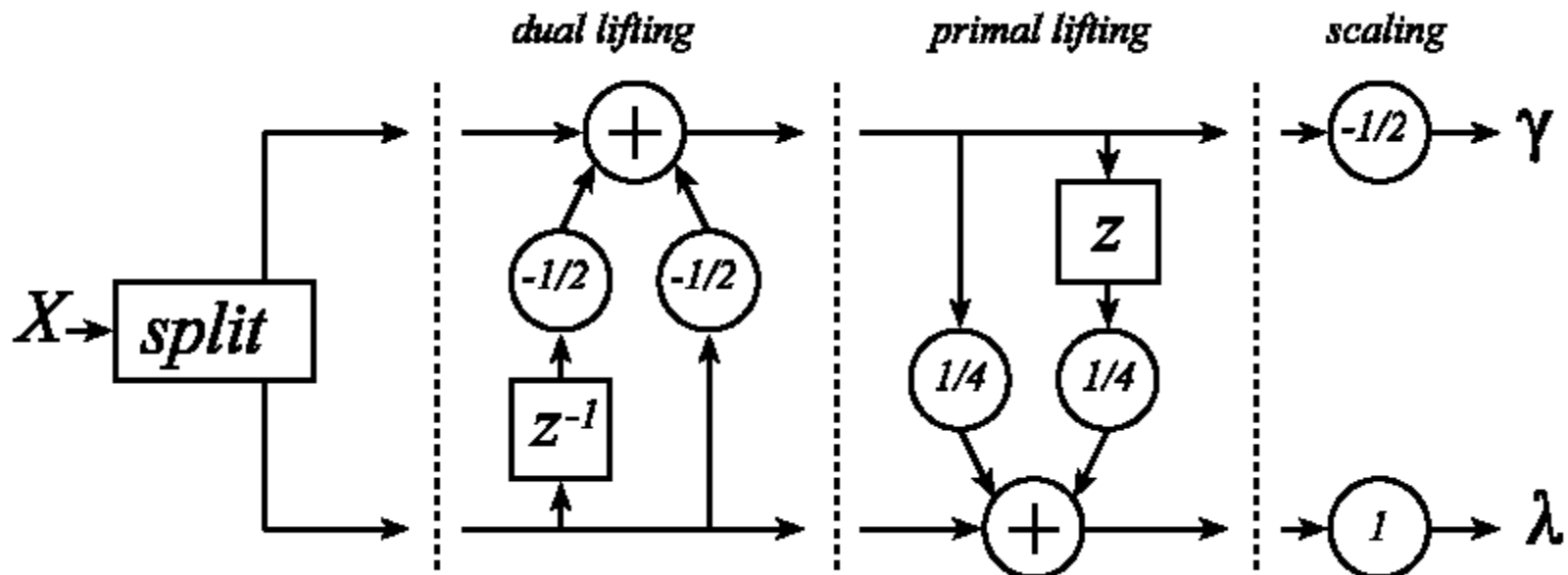
$$\tilde{\mathbf{P}}^{new}(z) = \begin{pmatrix} \tilde{h}_e(z) + \tilde{g}_e(z)\tilde{s}(z) & \tilde{h}_o(z) + \tilde{g}_o(z)\tilde{s}(z) \\ \tilde{g}_e(z) & \tilde{g}_o(z) \end{pmatrix} = \begin{pmatrix} 1 & \tilde{s}(z) \\ 0 & 1 \end{pmatrix} \tilde{\mathbf{P}}(z). \quad (19)$$



Lifting

Lifting: Zusammenhang mit Prediction

Auch Komplizierte Filter können komplett mit Lifting erzeugt werden.





Lifting

17.01.2019

Eigenschaften:

- Leicht zu invertieren
- Schnell
- Verwendung von Integerarithmetik möglich
 - Es kann beliebig gerundet werden
 - Trotzdem exakt rekonstruierbar
- Daher:

Fast Lifting Wavelet Transform



Wavelettransformation für Bilder

17.01.2019

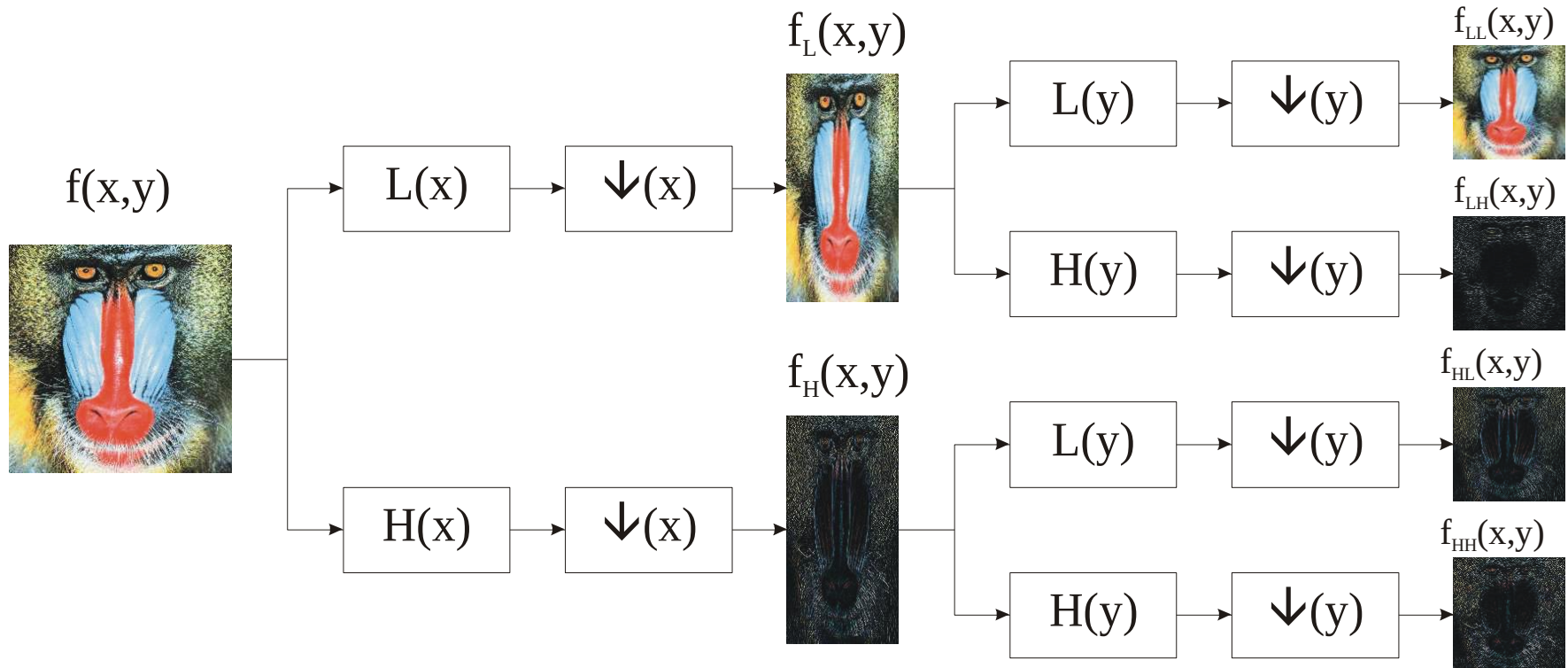
2D-Wavelettransformation

- Hintereinanderausführen von zwei 1D-Zerlegungen
- Z.B. zuerst Zeilen, dann Spalten
- Rekursive Weiterzerlegung der tieffrequenten Teilbilder

Wavelettransformation für Bilder

17.01.2019

Zerlegung in tieffrequentes Teilbild und drei hochfrequente Teilbilder





Literatur

17.01.2019

Bernd Jähne, Digitale Bildverarbeitung, Springer 1997

Tony Lindeberg: Scale-space theory: A framework for handling image structures at multiple scales

In: Proc. CERN School of Computing, Egmond aan Zee, The Netherlands, Sep. 8--21, 1996.
<ftp://ftp.nada.kth.se/CVAP/reports/lin-cern-summ-school-96.pdf>

Wim Sweldens, Peter Schröder: Building your own wavelets at home

In „Wavelets in Computer Graphics“, ACM SIGGRAPH Course Notes, 1996.
<http://www.multires.caltech.edu/teaching/courses/waveletcourse/athome.pdf>

Robi Polikar: THE WAVELET TUTORIAL

<http://users.rowan.edu/~polikar/WAVELETS/WTtutorial.html>