

---

# **MPEG-4 ALS**

## **Verlustlose Kompression von Audiosignalen**

---

**Tilman Liebchen**

Technische Universität Berlin

Fachgebiet Nachrichtenübertragung

(Prof. Thomas Sikora)

**Gesellschaft für Informatik, Dresden, 26.1.2006**

# Übersicht

---

- **Einführung**
  - MPEG-Standards
- **Audiocodierung**
  - PCM, verlustbehaftete und verlustlose Codierung
  - Psychoakustische Grundlagen
  - MP3, AAC, etc.
- **Verlustlose Audiocodierung**
  - Motivation und Grundlagen
  - Anwendungen
- **MPEG-4 Audio Lossless Coding (ALS)**
  - Übersicht zur Standardisierung
  - Funktionsweise und Tools
  - Kompressions-Ergebnisse
- **Zusammenfassung & Ausblick**

# MPEG-Standards

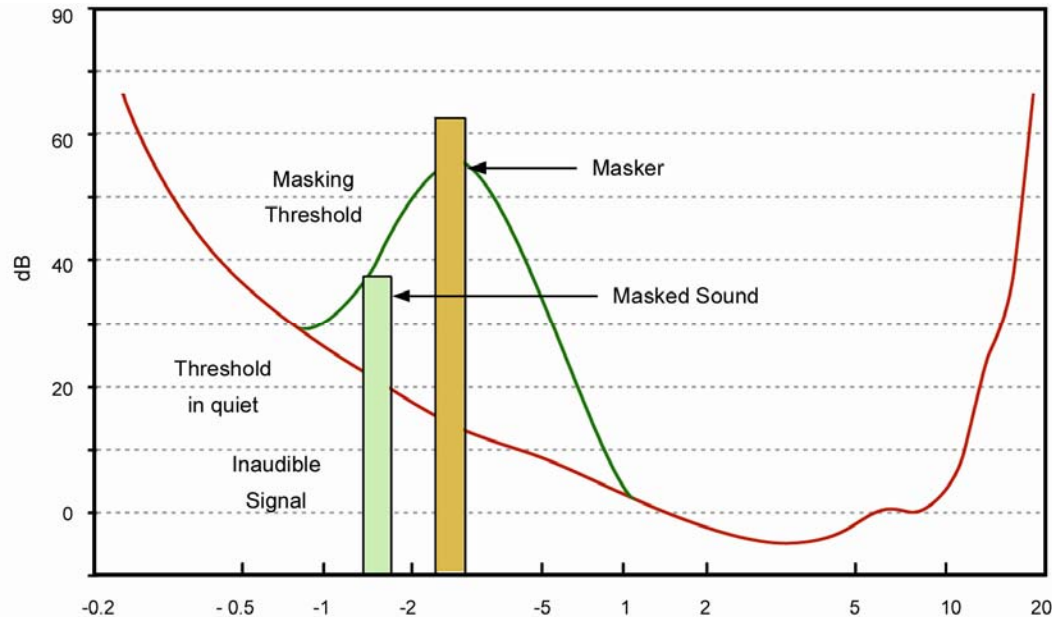


- **MPEG – Moving Picture Experts Group (1988)**
  - Arbeitsgruppe der ISO/IEC für Audio-/Videocodierung
- **MPEG-1 (1992)**
  - u.a. Video-CD, CD-I, MPEG-1 Audio Layer III ist “MP3”
- **MPEG-2 (1994)**
  - u.a. DVD, DVB (Digitales Fernsehen), DAB (Digitales Radio)
- **MPEG-4 (seit 1998)**
  - Multimedia-Standard, objektorientiert, skalierbar
  - u.a. AAC (MP3-Nachfolger), MPEG-4 Video, AVC (H.264)
- **MPEG-7 (2001)**
  - Multimedia Content Description (Metadaten – “Bits about the bits”)
- **MPEG-21**
  - “Framework for multimedia delivery and consumption”

- **CD-Audio (PCM, unkomprimiert)**
  - 44,1 kHz Abtastfrequenz, 16 Bit Auflösung, Stereo (2 Kanäle)
  - Datenrate  $R = 44100 \times 16 \times 2 \approx 1,4 \text{ Mbit/s}$
- **Verlustbehaftete Kompression (“Lossy Coding”)**
  - MP3, AAC, WMA, OggVorbis
  - Hohe Kompression  $> 1:10$ , typische Datenrate  $R = 128 \text{ kbit/s}$
  - Ausnutzung von psychoakustischen Effekten (“Perceptual Coding”)
  - Originalsignal wird nicht erhalten (daher “verlustbehaftet”)
- **Verlustlose Kompression (“Lossless Coding”)**
  - Bit-exakte Rekonstruktion des Originals (äquivalent zu PCM)
  - Geringere Kompression (typisch 2:1)

# Psychoakustik (1)

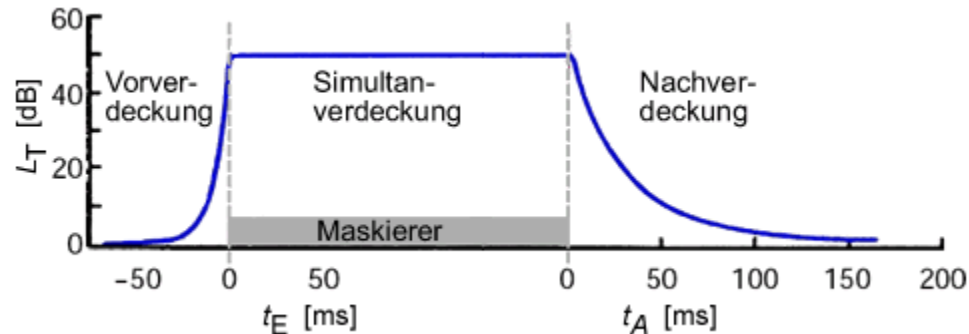
- Verdeckung (Masking): Simultanverdeckung



- Threshold in quiet: Ruhehörschwelle
- Masking Threshold: Mithörschwelle

# Psychoakustik (2)

- **Vor- und Nachverdeckung**



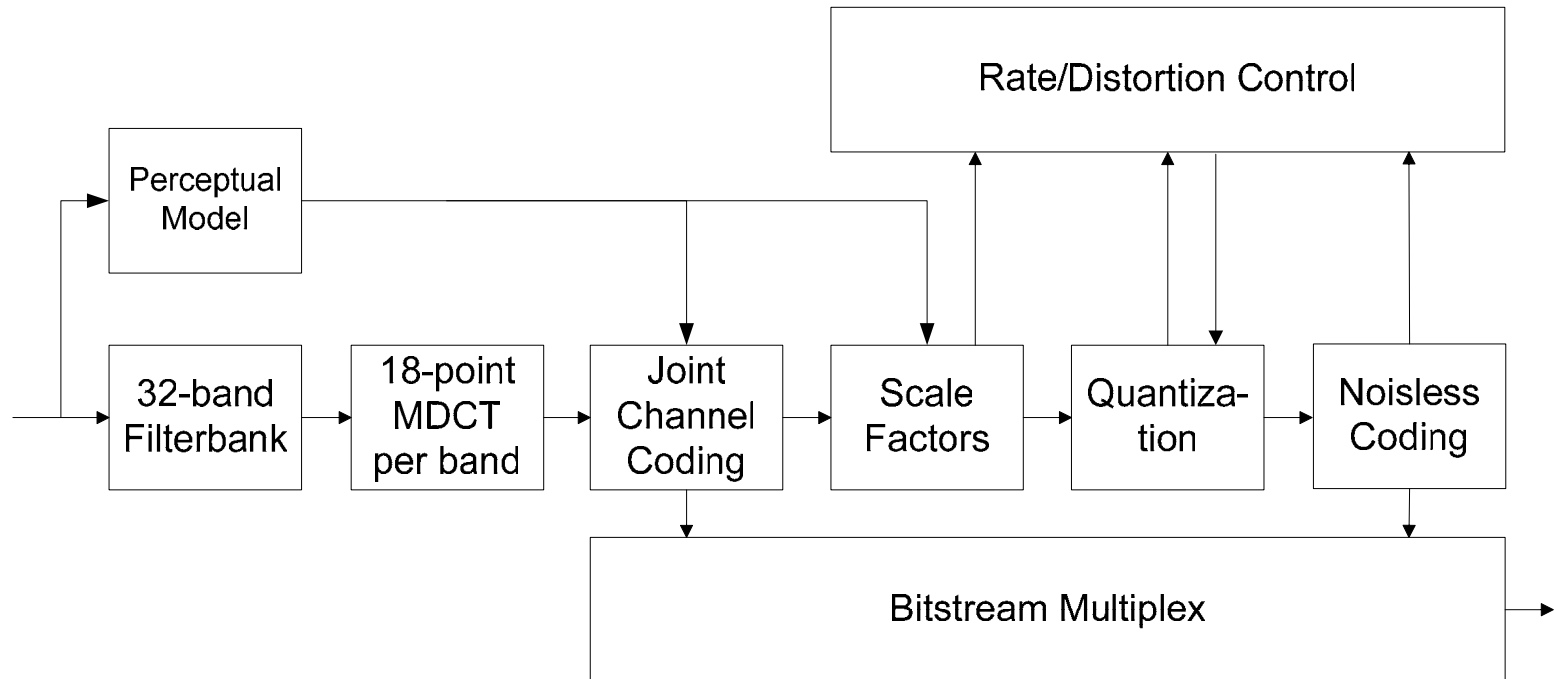
- **Ausnutzung zur Datenkompression (“Perceptual Coding”)**

- Die nicht hörbaren (verdeckten) Anteile werden nicht übertragen
- Die hörbaren Anteile werden so ungenau wie möglich beschrieben, so dass der Fehler (Rauschen) gerade noch unter der Mithörschwelle liegt
- Die verbleibende Datenmenge wird zusätzlich komprimiert

- **Problem: Das Originalsignal lässt sich nicht wiedergewinnen**

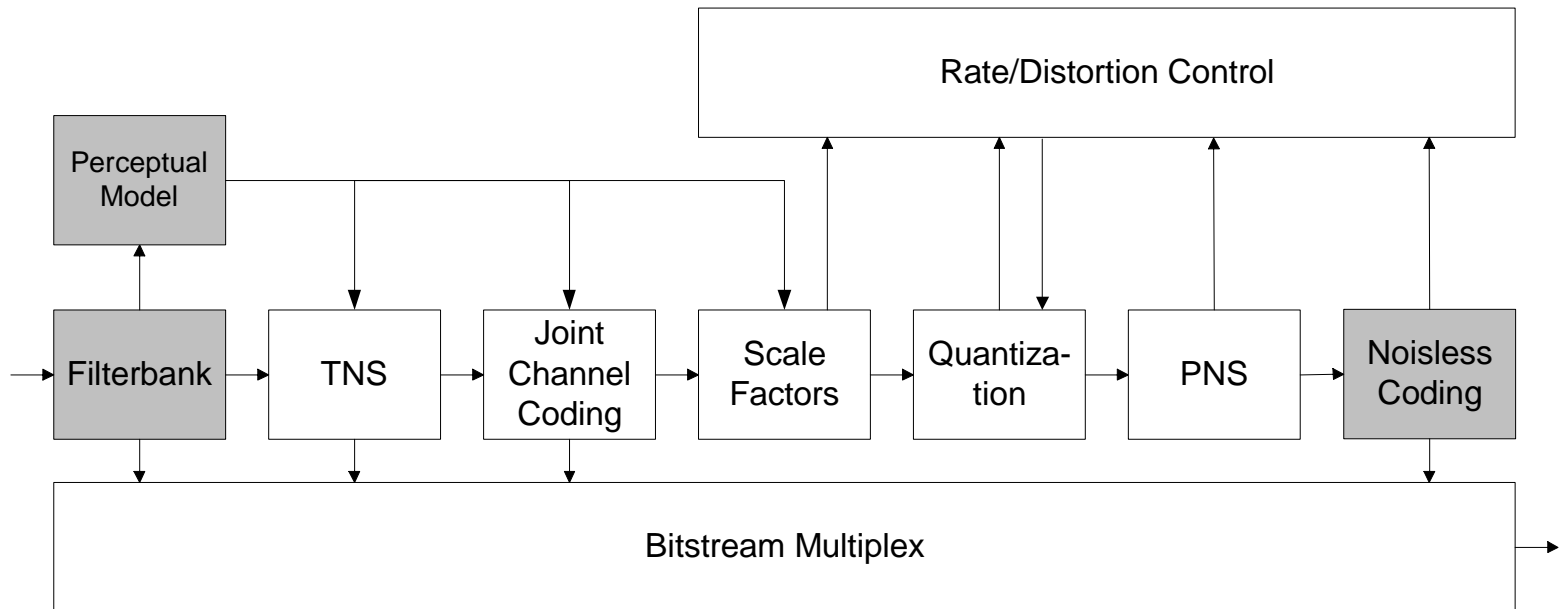
# MP3 (MPEG-1 Audio Layer III)

- **MP3-Encoder**



# MPEG Advanced Audio Coding (AAC)

- AAC-Encoder



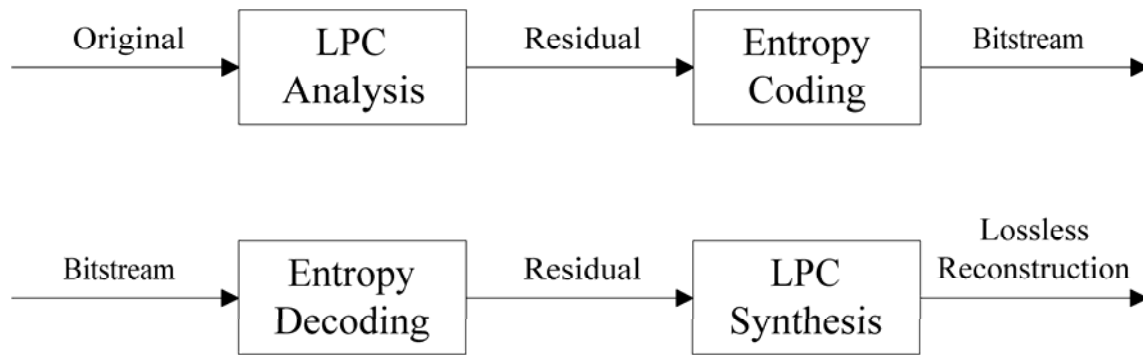
# Verlustlose Audiocodierung

- **Motivation**

- Audio-Kompression mit perfekter (bit-exakter) Rekonstruktion
- Universelle Kompressionsverfahren versagen bei Audiosignalen

- **Typische Vorgehensweise**

- *Dekorrelation* des Audiosignals (Prädiktion / Transformation)
- *Entropiecodierung* der dekorrelierten Samples



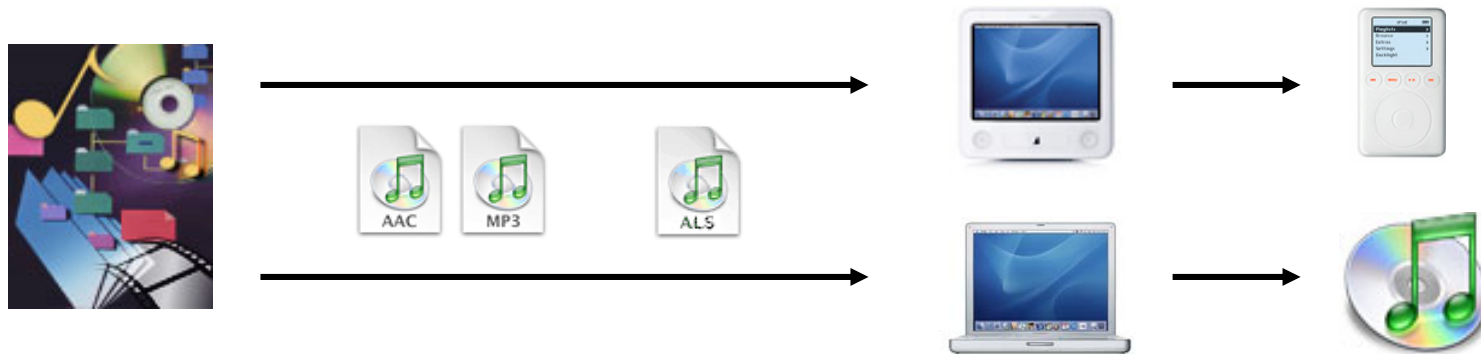
# Anwendungen

---

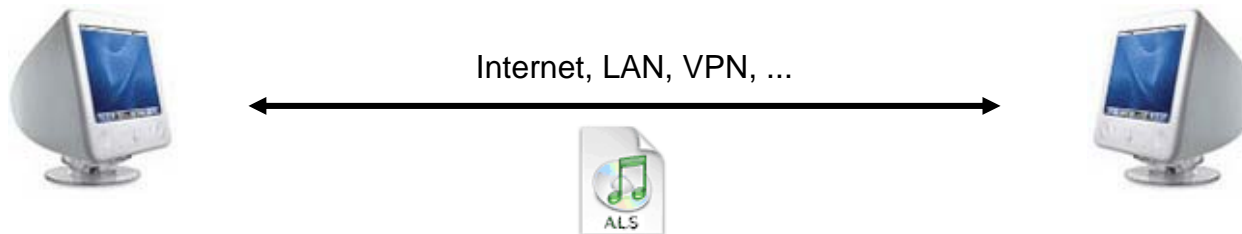
- **Übertragung von Audiodaten**
  - Internet, LAN, VPN
- **Archivierung**
  - Rundfunk, Studios, Plattenfirmen, Bibliotheken
- **Studiobetrieb**
  - Aufnahme, Speicherung, Übertragung, “Verteiltes Arbeiten”
- **Hochauflösende Disc-Formate**
  - DVD-Audio, Blu-ray Disc, HD-DVD
- **Musikvertrieb über das Internet**
  - Internet-Musikshops (iTunes etc.)
  - Kompressionfaktor von 2:1 spart ca. 30 Minuten beim Download einer CD über 1.5 Mbit/s DSL
- **Mobile Abspielgeräte**
  - iPod etc., PDA, Smartphones

# Ausgewählte Beispiele

- **Internet-Musikshop**



- **Übertragung von Audiodateien**



# MPEG-4 ALS – Übersicht

---

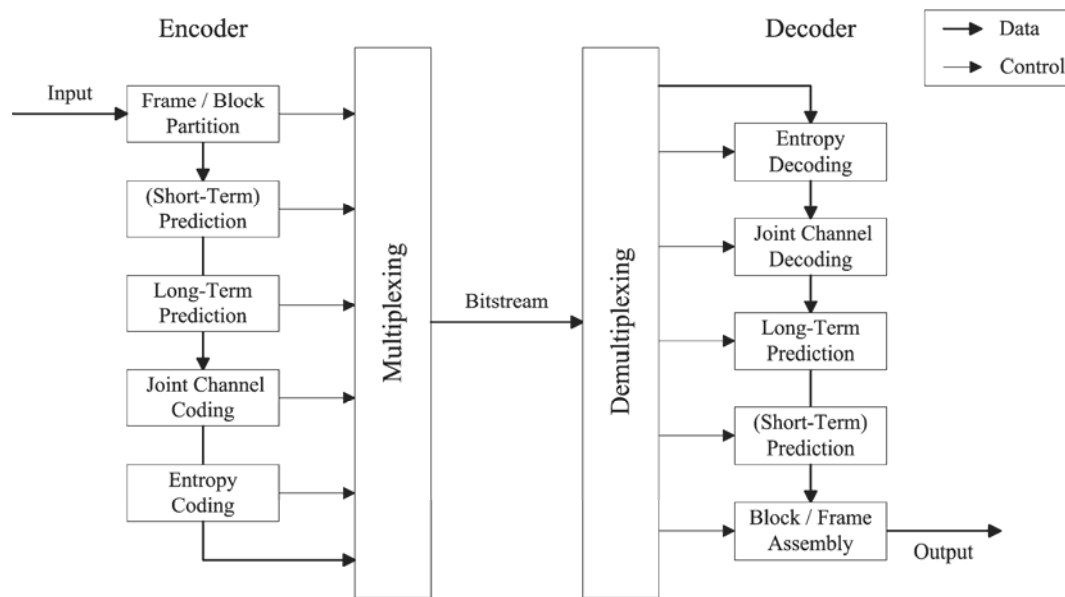
- **Der MPEG-4 ALS Standard**

- ISO/IEC 14496-3:2005/AMD 2 (Erweiterung von MPEG-4 Audio)
- Audio-Kompression mit verlustloser (bit-exakter) Rekonstruktion
- Basierend auf Technologie der Technischen Universität Berlin, Erweiterungen u.a. von RealNetworks und NTT
- Engültige Spezifikation wurde im Juli 2005 fertiggestellt

- **Eigenschaften von MPEG-4 ALS**

- Unterstützt Auflösungen bis 32-bit (inkl. IEEE-Fließkomma) bei beliebigen Abtastfrequenzen (16/44.1, 16/48, 24/48, 24/96, 24/192, ...)
- Mehrkanal mit bis zu 65536 Kanälen (inkl. 5.1 Surround)
- Schneller Zugriff (Random Access) auf die komprimierten Daten
- Speicherung als MP4-Datei (Multiplex mit Video und Metadaten)
- Auch geeignet für medizinische (z.B. EEG) und seismische Daten

# Der MPEG-4 ALS Codec



- **Frame / Block Partition** teilt den Audio-Datenstrom in Frames und weiter in Blöcke unterschiedlicher Länge (Block Length Switching)
- **(Short-Term) Prediction** berechnet das Restfehlersignal
- **Long-Term Prediction** entfernt zusätzlich periodische Signalanteile
- **Joint Channel Coding** nutzt Abhängigkeiten zwischen Kanälen aus
- **Entropy Coding** komprimiert das verbleibende Restfehlersignal

- **Prädiktion mit FIR-Filtern**

- Der aktuelle Abtastwert  $x(n)$  wird aus vorherigen Werten geschätzt:

$$\hat{x}(n) = \sum_{k=1}^K h_k \cdot x(n-k)$$

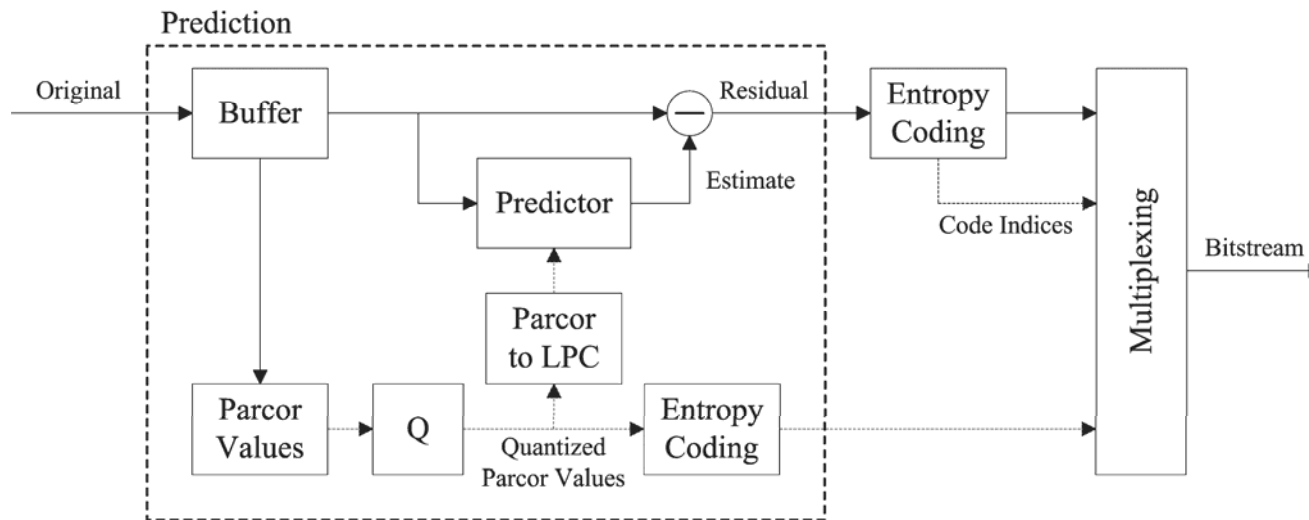
- Das Restfehlersignal (Residual) ist dann  $e(n) = x(n) - \hat{x}(n)$
- Die Varianz des Restfehlers  $e(n)$  ist kleiner als die des Originals  $x(n)$ , so dass die benötigte Bitrate ebenfalls geringer ist

- **Methoden zur Adaption der Prädiktor-Koeffizienten**

- Vorwärts-Adaption: Koeffizienten werden als Nebeninformation übertragen; einfacher Decoder, da Adaption nur im Encoder
- Rückwärts-Adaption: Encoder und Decoder führen eine identische Adaption der Koeffizienten durch; hohe Komplexität des Decoders
- MPEG-4 ALS definiert sowohl Vorwärts- als auch Rückwärts-Adaption (im folgenden wird nur die Vorwärts-Adaption beschrieben)

# Vorwärts-adaptive Prädiktion (1)

- **MPEG-4 ALS Encoder**



- **Buffer** – Speichert einen Block mit Audio-Abtastwerten
- **Parcor Values** – Berechnet die optimalen Parcor-Koeffizienten
- **Q** – Quantisierung der Parcor-Koeffizienten
- **Parcor to LPC** – Konvertierung in LPC-Koeffizienten
- **Predictor** – Berechnet das Restfehlersignal (Residual)

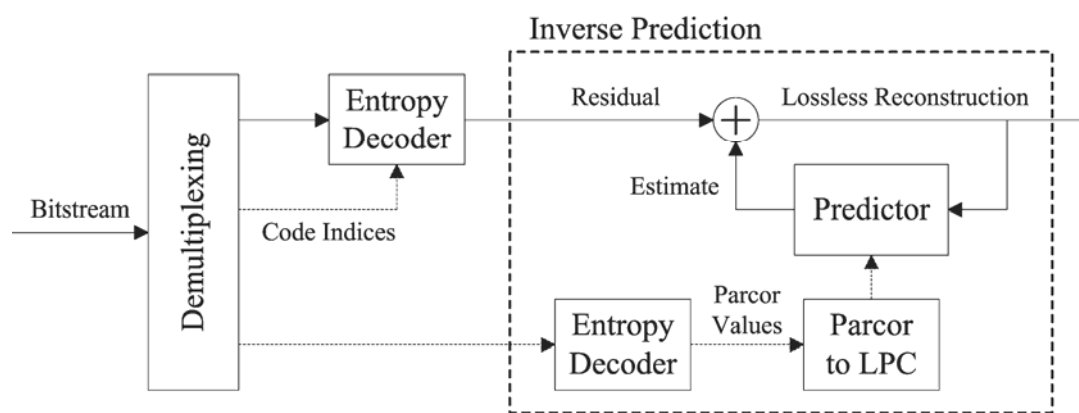
# Vorwärts-adaptive Prädiktion (2)

---

- **Berechnung der Prädiktor-Koeffizienten**
  - Die optimalen Prädiktor-Koeffizienten  $h_k$  werden typischerweise mittels der Autokorrelations-Methode bestimmt
  - Eine effiziente Realisierung ist der Levinson-Durbin-Algorithmus
  - Der Algorithmus berechnet zusätzlich die Parcor-Koeffizienten  $r_k$
- **Adaptation der Prädiktor-Ordnung**
  - Eine höhere Ordnung  $K$  verringert zwar die Varianz und damit die Bitrate des Restfehlersignals, durch die größere Anzahl Koeffizienten wächst allerdings die Bitrate zu deren Übertragung
  - Eine adaptive Wahl der Prädiktor-Ordnung  $K$  durch den Encoder minimiert die Gesamt-Bitrate (Restfehler + Koeffizienten)
  - Die maximale Prädiktor-Ordnung ist  $K_{\max} = 1023$
  - Für die mittlere Ordnung gilt typischerweise  $K_{\text{avg}} \ll K_{\max}$

# Vorwärts-adaptive Prädiktion (3)

- **MPEG-4 ALS Decoder**



- Restfehlersignal und Parcor-Koeffizienten werden decodiert
- Parcor-Koeffizienten werden in LPC-Koeffizienten konvertiert
- Der inverse Prädiktor rekonstruiert das Originalsignal
- Der Decoder ist deutlich einfacher als der Encoder (da keine Adaption)
- Die Komplexität hängt von der mittleren Prädiktor-Ordnung  $K_{\text{avg}}$  ab

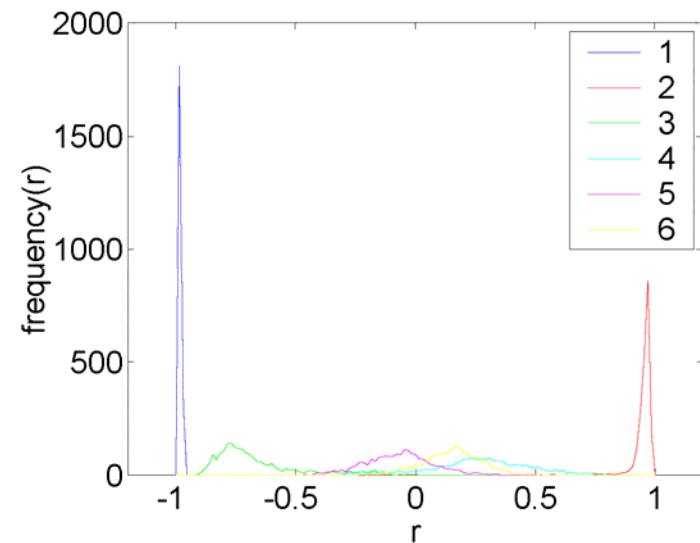
# Übertragung der Prädiktor-Koeffizienten (1)

- **Problemstellung**

- Effiziente aber ausreichend genaue Quantisierung notwendig
- Direkte Quantisierung der LPC-Koeffizienten  $h_k$  nicht effizient, da kleine Fehler zu großen spektralen Abweichungen führen können

- **Parcor-Koeffizienten**

- Quantisierung der Parcor- (Reflektions-) Koeffizienten  $r_k$
- Werte der Koeffizienten liegen im Intervall  $[-1,+1]$
- Koeffizienten robust, außer wenn Betrag nahe Eins
- Koeffizienten  $r_1$  und  $r_2$  liegen fast immer nahe  $-1$  und  $+1$



# Übertragung der Prädiktor-Koeffizienten (2)

- **Kompondierung und Quantisierung**

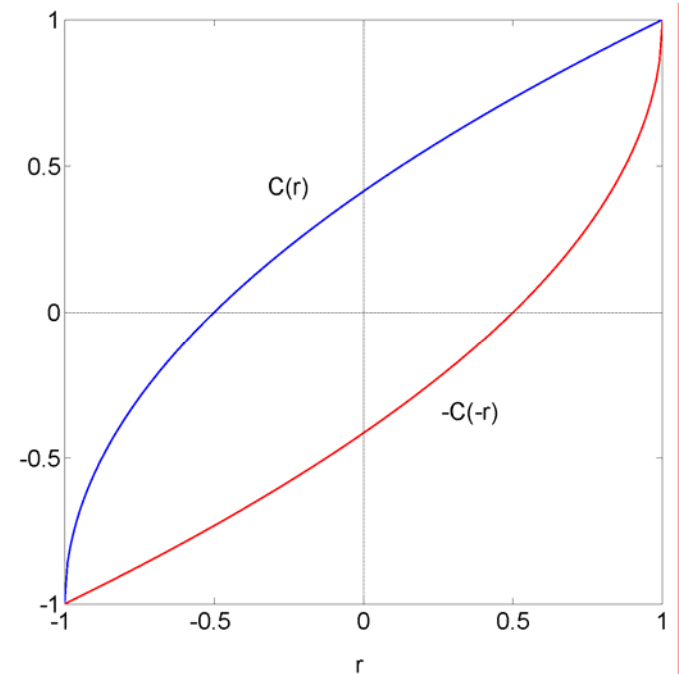
- Nichtlinearer Komponder für die ersten beiden Koeffizienten, führt zu besserer Auflösung bei  $-1$  and  $+1$
- Anschließend lineare Quantisierung aller Koeffizienten mit 7 Bits

- **Entropiecodierung**

- Rice-Codes (entsprechend Statistik)
- Durchschnittlich 4 Bits/Koeffizient

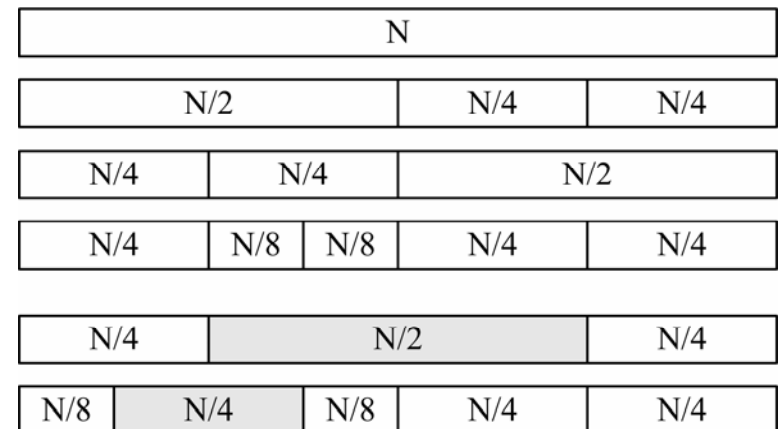
- **Verlustlose Umsetzung**

- Die quantisierten Parcor-Koeffizienten müssen für das Prädiktionsfilter in LPC-Koeffizienten konvertiert werden
- Identische Konvertierung (Integer-Arithmetik) in Encoder und Decoder



# Blocklängen-Umschaltung

- **Ansatz**
  - Ein Satz Prädiktor-Koeffizienten für jeden Block von Abtastwerten
  - Umschaltung der Blocklänge ermöglicht Anpassung sowohl an transiente als auch an stationäre Abschnitte
- **Realisierung**
  - Hierarchische Unterteilung eines Frames in bis zu 32 Blöcke
  - Stationäre Abschnitte: Lange Blöcke, hohe Prädiktor-Ordnung
  - Transiente Abschnitte: Kurze Blöcke, kleine Prädiktor-Ordnung
  - Unterteilung wird als Nebeninfo übertragen ( $\leq 32$  Bits/Frame)



Beispiele für Blocklängen-Umschaltung:  
Die Unterteilungen mit grauen Blöcken  
sind nicht möglich (da nicht hierarchisch)

# Random Access

---

- **Anforderungen**

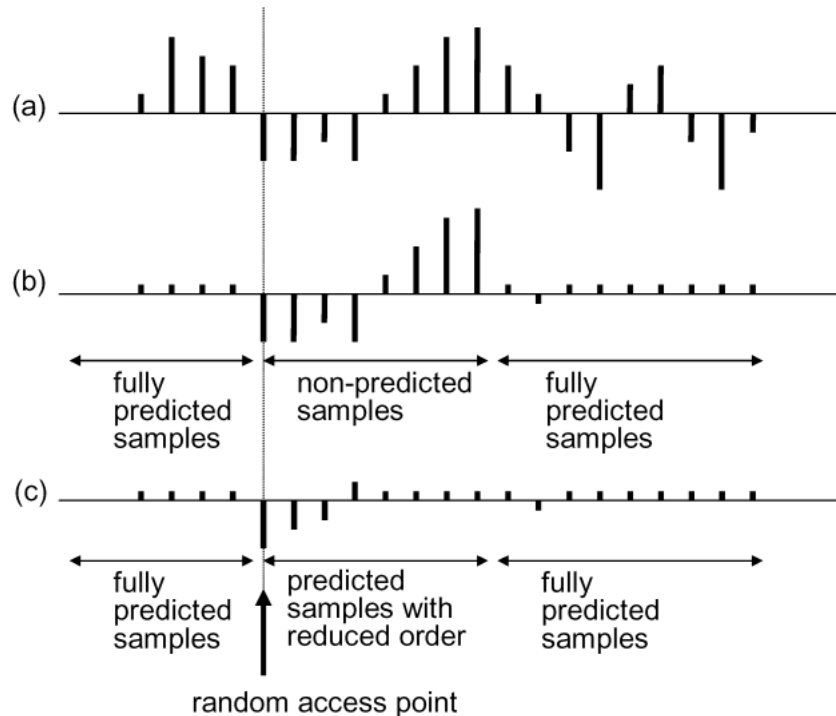
- Direkter Zugriff auf beliebige Abschnitte des codierten Audiosignals, ohne Decodierung vorhergehender Abschnitte
- Mindestauflösung beträgt 0,5 Sekunden

- **Umsetzung**

- Erzeugung von *Random Access (RA) Frames* (z.B. alle 100 ms)
- RA Frame enthält Abstand (in Bytes) zum nächsten RA Frame
- *Progressive Prädiktion* (d.h. Prädiktion mit reduzierter Ordnung) der ersten  $K$  (= Prädiktorordnung) Samples
- Ermöglicht quasi gleiche Kompression wie ohne Random Access bei gleichem Aufwand

# Random Access – Progressive Prädiktion

- **Prädiktion in Random Access Frames**

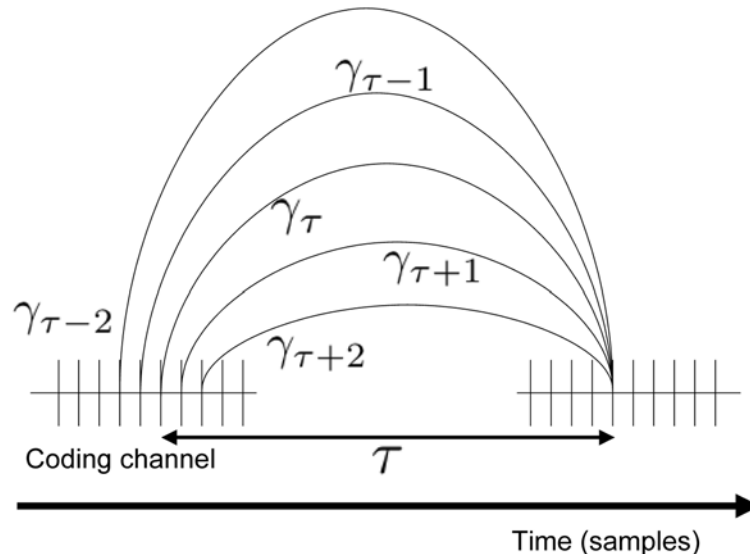


- (a) Originalsignal, (b) Restfehlersignal bei herkömmlicher Prädiktion, (c) Restfehlersignal bei progressiver Prädiktion

# Langzeit-Prädiktion

- **Long-Term Prediction (LTP)**

- Ausnutzung periodischer Anteile im Restfehlersignal
- Zusätzlicher Prädiktor mit 5 Koeffizienten
- Adaptive Bestimmung der Koeffizienten  $\gamma$  und der Verzögerung  $\tau$
- Vorteilhaft in Verbindung mit dem normalen (Kurzzeit-) Prädiktor, insbesondere bei niedrigen Prädiktor-Ordnungen



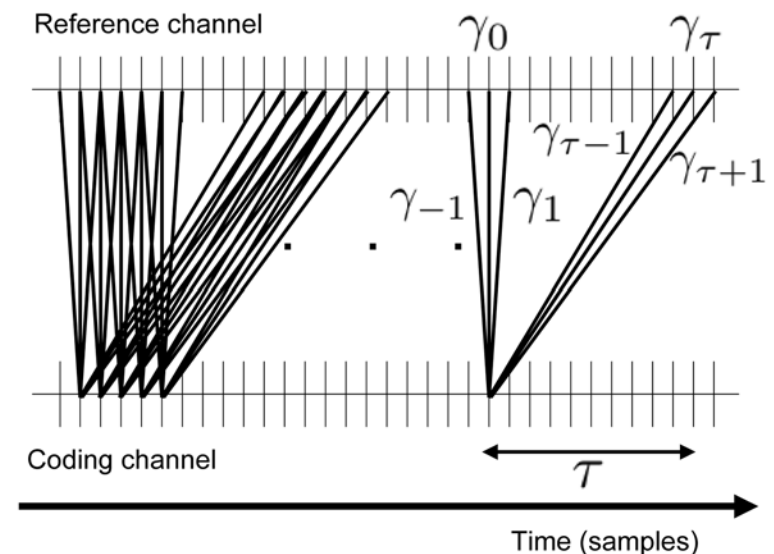
# Codierung von mehreren Kanälen

- **Joint Stereo (Differenz-Codierung)**

- Differenz-Codierung von Kanalpaaren (z.B. L/R oder Ls/Rs)
- Übertragung des Differenzsignals  $D = R - L$  anstelle von L oder R
- Vorteilhaft, wenn beide Kanäle sehr ähnlich sind

- **Multi-Channel Coding (MCC)**

- Gewichtete Differenz-Codierung zwischen beliebigen Kanälen
- Angewendet auf Restfehler, d.h. nach der Prädiktion
- Kreuz-Prädiktion mit 3 oder 6 Koeffizienten  $\gamma$
- Adaptive Umschaltung zwischen MCC und Joint Stereo möglich



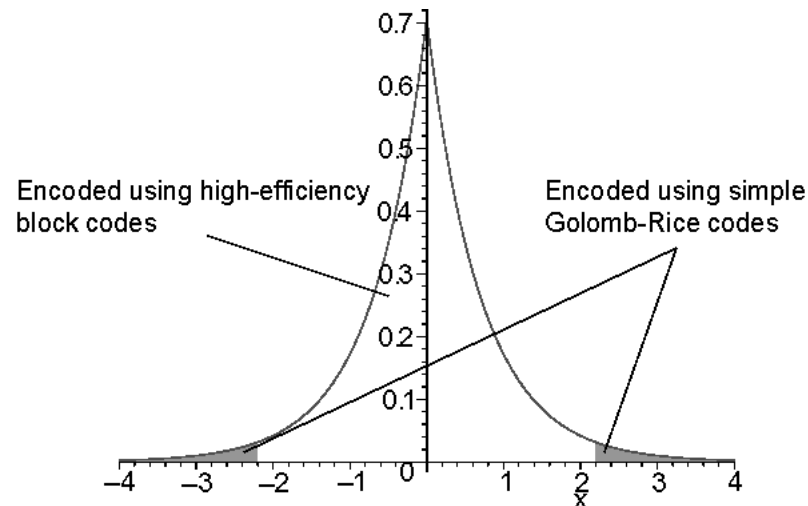
# Entropiecodierung des Restfehlersignals

---

- **Ziel und Prinzip der Entropiecodierung**
  - Effiziente Codierung des Restfehlersignals nach der Prädiktion (einschließlich LTP, Joint Stereo, MCC)
  - Ausnutzung der ungleichförmigen Häufigkeitsverteilung
- **Rice-Codierung**
  - Einfache Codierung basierend auf Exponentialverteilungen unterschiedlicher Standardabweichung  $\sigma$
  - Codes werden durch jeweils einen einzigen Parameter (in Abh. von  $\sigma$ ) eindeutig definiert
- **BGMC (Block Gilbert Moore Codes)**
  - Arithmetische Codierung der MSBs (grobe Form der Verteilung)
  - PCM-Codierung der LSBs (quasi gleichverteilt)

# Entropiecodierung – BGMC

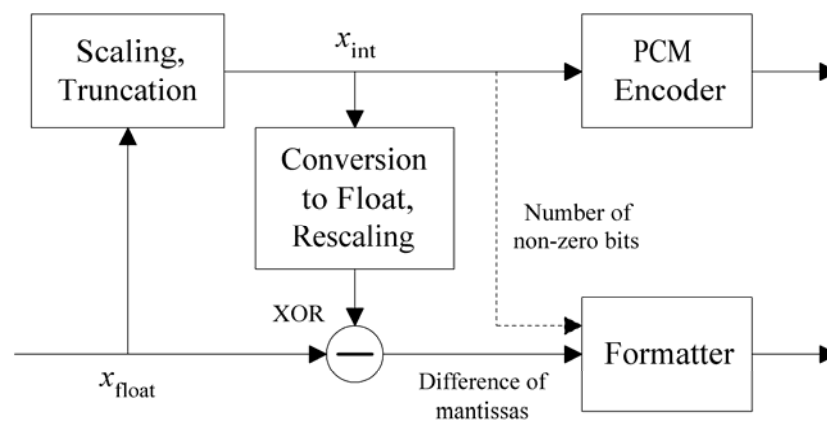
- **Aufteilung der Verteilungsfunktion**
  - Zentrale Region (MSBs: Arithmetische Codierung; LSBs: direkt)
  - Endbereiche (Rice-Codes)



- **Verbesserte Codiereffizienz gegenüber Rice-Codierung**
- **Komplexität etwa doppelt so hoch**

# Codierung von Fließkomma-Daten

- **IEEE 32-bit Fließkomma-Format (Typ *float*)**
  - 1 bit Vorzeichen  $s$ , 8 bit Exponent  $e$ , 23 bit Mantisse  $1.m$
  - Wert  $x_{\text{float}} = (-1)^s \cdot 2^{e-127} \cdot 1.m$
- **Aufteilung in Integer- und “Nachkomma”-Anteil**



- Integer-Anteil wird mit normalem PCM-Encoder codiert
- Zusätzlich Codierung des “Nachkomma”-Anteils

# Kompressions-Ergebnisse

- **Mittlere Kompression für verschiedene Audio-Formate**
  - Verbleibende Datenmenge in Prozent (%), kleinere Werte sind besser

<b>Format (Stereo)</b>	<b>FLAC</b>	<b>ALS</b> (low)	<b>ALS</b> (medium)	<b>ALS</b> (maximum)
48 kHz / 16-bit	48.6	46.5	45.3	44.5
48 kHz / 24-bit	68.4	63.9	63.2	62.7
96 kHz / 24-bit	56.7	47.4	46.3	46.1
192 kHz / 24-bit	-	38.4	37.6	37.5
<b>Total</b>	-	<b>48.9</b>	<b>48.1</b>	<b>47.7</b>

- ALS low: Prädiktor-Ordnung  $K \leq 15$ , Rice Coding, kein LTP, kein MCC
- ALS medium: Prädiktor-Ordnung  $K \leq 30$ , BGMC
- ALS maximum: Prädiktor-Ordnung  $K \leq 1023$ , BGMC

# Komplexität (Referenz-Decoder)

- **Mittlere CPU-Last in Prozent (%), 1.7 GHz Pentium-M**

<b>Format (Stereo)</b>	<b>ALS (low)</b>	<b>ALS (medium)</b>	<b>ALS (maximum)</b>
48 kHz / 16-bit	1.6	4.7	18.2
48 kHz / 24-bit	1.8	5.6	19.1
96 kHz / 24-bit	3.6	11.6	23.1
192 kHz / 24-bit	6.7	19.4	24.4

- ALS low:  $K \leq 15$ , Rice Coding, kein LTP, kein MCC
  - ALS medium:  $K \leq 30$ , BGMC
  - ALS maximum:  $K \leq 1023$ , BGMC
- **Echtzeit-Decodierung auch auf einfachen Geräten möglich**

# Zusammenfassung und Ausblick

---

- **Zusammenfassung**

- MPEG-4 Audio Lossless Coding (ALS) ist ein standardisiertes, äußerst effizientes verlustloses Audiocodierverfahren
- Original-Audioqualität bei reduzierter Bitrate gegenüber PCM, daher auch für professionelle Anwendungen geeignet
- Standardisierung auf Grundlage eines Codecs der TU Berlin, mit Erweiterungen u.a. von RealNetworks und NTT
- Flexible Encoder-Einstellungen, geringe Komplexität des Decoders

- **Ausblick**

- Der Standard wurde im Dezember 2005 endgültig verabschiedet und wird demnächst veröffentlicht ([www.iso.ch](http://www.iso.ch))
- Als weltweiter Standard gewährleistet MPEG-4 ALS die Kompatibilität zwischen unterschiedlicher Hardware und Software
- Garantiert langfristige herstellerunabhängige Unterstützung

# Mehr Informationen zu MPEG-4 ALS

---

- **MPEG-4 ALS Website:**

<http://www.nue.tu-berlin.de/mp4als/>

- **MPEG Website:**

<http://www.chiariglione.org/mpeg/>

- **Noch Fragen...?**