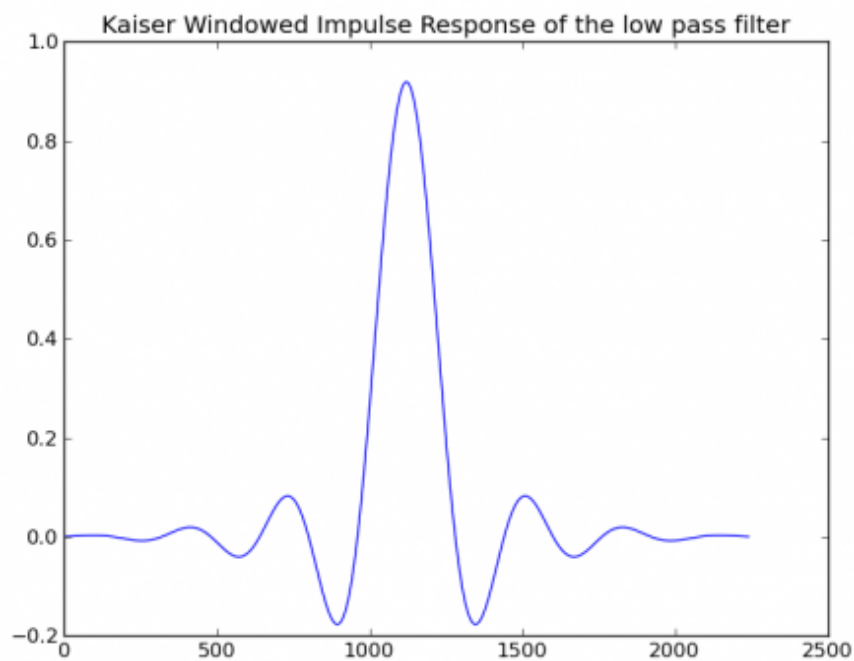


# Asynchronous Sample Rate Conversion

## ASRC

Prof. Dr. Christian Münker



sinc-Funktion

26. September 2014

[Christian.Muenker@hm.edu](mailto:Christian.Muenker@hm.edu)

## Inhaltsverzeichnis

<b>Inhaltsverzeichnis</b>	<b>2</b>
<b>1 Worum geht's?</b>	<b>2</b>
<b>2 Interpolationsverfahren</b>	<b>3</b>
2.1 Zero-Order Hold Interpolation . . . . .	3
2.2 Lineare Interpolation . . . . .	3
2.3 Quadratische Interpolation . . . . .	3
2.4 Spline-Interpolation . . . . .	3
<b>3 Simulation und Modellierung</b>	<b>3</b>
<b>4 Literatur</b>	<b>3</b>

## 1 Worum geht's?

Das Prinzip der Interpolation ist jedem bekannt, der in einer Tabelle einmal versucht hat „zwischen den Zeilen“ zu lesen, also Zwischenwerte aus den gegebenen Tabellenwerten abzuleiten: Interpolation erzeugt eine kontinuierliche Funktion aus einer Menge von diskreten Stützstellen. Sie ist an jedem Punkt auswertbar und stimmt an den Stützstellen mit diesen (exakt oder näherungsweise) überein. Man kann einen Interpolator also auch als Filter mit zeitdiskreter Eingangsfolge und zeitkontinuierlichem Ausgangssignal betrachten.

Das ist in der digitalen Signalverarbeitung in vielen Anwendungen von großem Wert. Die Buchstaben dieses Textes liegen auf dem Rechner beispielsweise nur in Form einiger diskreter Punkte vor. Mit diesen an sich kann noch nichts dargestellt werden - erst durch Spline-Interpolation bekommen diese ihre Form. Der große Vorteil speichern ist offensichtlich: Die Lage von nur 20 Punkte im Raum zu speichern ist wesentlich effizienter als ein Bild für jede erdenkliche Auflösung. Interpolation, also die Problemstellung aus einer Reihe von Zahlenwerten verlässliche Zwischenwerte zu ermitteln, hat eine lange Geschichte. [Mei02] beschreibt z.B. die Geschichte der Interpolation von der Antike bis in das 20. Jahrhundert und ordnet sie so in einen größeren Kontext ein.

Mit dem Siegeszug der digitalen Signalverarbeitung tat sich ein riesiges neues Einsatzfeld für die Interpolation auf: Die Abtastrate von Mess- und Audiodaten musste geändert werden, die Auflösung von digitalen Bildern konvertiert werden. Julius O. Smith's „Digital Audio Resampling Home Page“ (<https://ccrma.stanford.edu/~jos/resample/resample.html>, auch als pdf)

Der Schwerpunkt dieses Artikels liegt auf der synchronen und asynchronen Abtastratenwandlung und deren Implementierung in Soft- und Hardware.

[Lok05] unterscheidet zwischen „arbitrary sample rate conversion“ und „asynchronous arbitrary sample rate conversion“

ASRC ist Literatur:

JOS

Uri Nieto Audio Resampling in Python, <http://urinieto.com/2011/05/audio-resampling-in-python/>

## 2 Interpolationsverfahren

### 2.1 Zero-Order Hold Interpolation

### 2.2 Lineare Interpolation

### 2.3 Quadratische Interpolation

### 2.4 Spline-Interpolation

## 3 Simulation und Modellierung

### Wichtige Begriffe

**Sample Rate** (Abtastrate) die Rate, mit der ein zeitdiskretes Signal vorliegt

**Upsampling** Erhöhen der Abtastrate um einen ganzzahligen Faktor, die fehlenden Samples werden durch Null gefüllt (zero-stuffing) oder durch Wiederholen des letzten Werts (Zero-Order Hold). Dabei entstehen Kopien des ehemaligen Basisbands bis zur neuen  $\rightarrow$  Nyquistfrequenz, sog.  $\rightarrow$  Images, die durch Anti-Image Filter entfernt werden müssen.

**Downsampling** Verringerung der Abtastrate um einen ganzzahligen Faktor durch Weglassen von Abtastwerten. Ohne vorherige Bandbreitenbegrenzung (Anti-Alias Filterung) besteht die Gefahr von  $\rightarrow$  Aliasing.

#### Resampling

**Synchron** wenn sich das Verhältnis von zwei Abtastraten durch ein einfaches Bruch ausdrücken lässt, z.B. beim Resampling von 44.1kHz to 48kHz,  $44.1/48 = 147 / 160$ .

**Asynchron** wenn die zwei Abtastraten kein einfaches rationales Verhältnis haben

## 4 Literatur

Da das Konzept der Interpolation in den verschiedensten Zusammenhängen verwendet wird, ist die mathematische Behandlung auch dementsprechend verschieden. Zwischen den folgenden Konzepten sollten meiner Meinung nach unterschieden werden:

**Fractional-Delay-Filter:** Ein Interpolator, aus dessen kontinuierlichem Ausgangssignal nur ein Wert entnommen wird



**Interpolationsfilter:** Filter zum Entfernen der Images bei Aufwärtstastung um einen ganzzahligen Faktor

**Effiziente Strukturen:** Polyphasen-Filter, Farrow-Filter

**Abtastratenwandler:** Systeme bestehend aus Auf- bzw. Abwärtstastern, Filtern und/oder Interpolatoren

## Übersichtswerke

Die beste und fundierteste Arbeit fand ich in [Eva00]. In der Dissertation wird ein mathematisches Modell für Abtastratenwandler vorgestellt und der Entwurf von Interpolatoren behandelt. Die Herangehensweise ist dabei so mathematisch wie es die Thematik erfordert. Daher ist die Arbeit für einen Einstieg eher nicht geeignet. Ich könnte erst jetzt, zum Ende meiner Beschäftigung, damit arbeiten.

Steven Smith, bekannt für seine frei im Internet verfügbaren Bücher, hält seit kurzem (2011) eine Vorlesung zur Thematik. Das Skript findet sich in [Smi11]. Ein weiteres gutes Skript von einem anderen Autor findet sich in [DB]. Unter [LBR04] findet sich eine interessante Gegenüberstellung unterschiedlicher Modelle von Interpolationsfiltern. In [SR73] gibt Ronald Schafer schließlich eine gute Übersicht über Interpolation in der Signalverarbeitung, Abtastratenkonvertierung, Lagrange Interpolatoren und über den Entwurf von FIR Interpolationsfiltern.

## Interpolationstheorie

Zum Verständnis der Interpolatoren ein umfassenderes Verständnis nötig war, als in meinem Studium vermittelt wurde. Ich machte mich daher auf, allgemein-gültigere Definitionen der Interpolation zu finden. In [Pin] geht es um die Anfänge der Approximationstheorie nach Weierstraß. In [Fom00] fand ich zum ersten Mal eine funktionalanalytische Beschreibung der Interpolation mithilfe von Basisfunktionen - eine Herangehensweise, die zum Verständnis der Interpolation mittels B-Splines wichtig ist.

## Interpolatoren

Zum Entwurf von Lagrange Interpolatoren erfährt man etwas in [SR73] und [Ye03]. Eine gute Übersicht über B-Splines gibt Michael Unser in seinen Papers, z.B. zum Einstieg in das Thema in [Uns99], oder in [UAE93a] und [UAE93b]. Der Zusammenhang zwischen kardinalen B-Splines und deren Konvergenz gegen den idealen bandbegrenzten Interpolator wird in [AA92] behandelt. Auch die bereits erwähnte Dissertation [Eva00] sowie ein daraus abgeleitete Journal Paper [Eva03] behandeln den Entwurf von Interpolatoren.



## Implementierung von Interpolatoren

Zur Implementierung eines Interpolators auf einem FPGA ist eine Abbildung der Theorie auf eine effiziente Hardware-Struktur nötig. In seinem Buch behandelt Fredric Harris einige Aspekte der Implementierung von Polyphasen- und Farrow-Filtern in Hardware [Har04]. In [?] entwickelt Steven Smith eine an der Praxis orientierte Methode, ideale bandbegrenzte Interpolation zu approximieren und mit Lookup-Tables zu implementieren. [Far88] ist das berühmte Paper von C. Farrow, in dem er das Farrow-Filter zur effizienten Implementierung von polynomialen Filtern vorstellt. In [NCD09] wird ein Fractional-Delay-Filter auf einem FPGA in einer aufwandsarmen Struktur realisiert. In [UAE93b] behandelt Michael Unser die Implementierung von B-Spline Interpolatoren. In [FLL<sup>+</sup>93a] und [FLL<sup>+</sup>93b] wird eine Struktur von Lagrange Interpolatoren vorgestellt, die ohne Multiplizierer auskommt.

## Anwendungen von Interpolatoren

Interpolatoren kommen nicht nur in Abtastratenwandlern zum Einsatz, sondern auch in der Bildverarbeitung [UTY95], und Medizintechnik [LGS01]. Sie werden auch dazu verwendet, um mit einer kontinuierlichen Form eines Signals zu rechnen [DB]. In [Nie01] findet sich ein sehr interessanter Vergleich verschiedener Polynom-Interpolatoren zum Resampling von überabgetasteten (oversampled) Audio Signalen.

## Entwurf von Abtastratenwandlern

Ein Interpolator ist ein Teil-System eines Abtastratenwandlers. Er ist im einfachsten Falle ein Zero-Order-Hold. In [Lok05] gibt Ivar Løkken einen einfachen Einstieg in den Entwurf von mehrstufigen Wandlern mit Zero-Order-Hold, und in Grundzügen auch, mit First-Order-Hold Interpolatoren. In [CR81] geht es um Ähnliches. Unter [Inf] finden sich interessante Tests verschiedener Wandler.

## FFT-basierte Spektralanalyse, Fensterung

Die Bestimmung des SNRs der von mir in Python modellierten Wandler machte es notwendig, mich im Detail mit FFT basierter Spektralanalyse, und der Fensterung von Signalen zu beschäftigen. [GH02] gibt zu beiden Themen eine gute Übersicht. Das Paper von Fredric Harris unter [Har78] ist das Standardwerk zum Einsatz von Fenstern. Unter [JH07] findet sich eine ausführliche Erklärung des Zusammenhangs zwischen Leistungsdichte-Spektrum und Leistungs-Spektrum im Spectrum-Scope Block von Simulink, und in [Nar98] wird die Fourier-Transformierte einer Funktion durch die DFT approximiert.



## Literatur

- [AA92] M. E. A. Aldroubi, M. Unser, *Cardinal spline filters: Stability and convergence to ideal sinc interpolator*, Signal Processing **28** (1992), 127–138.
- [CR81] R. E. Crochiere and L. R. Rabiner, *Interpolation and decimation of digital signals—a tutorial review*, Proceedings of the IEEE **69** (1981), no. 3, 300–331.
- [DB] J. V. D. Babic, *Polynomial-based interpolation filters for dsp applications*, Lecture Notes, <http://www.cs.tut.fi/kurssit/TLT-5806/Interpol.pdf>.
- [Eva00] G. Evangelista, *Zum entwurf digitaler systeme zur asynchronen abtastratenumsetzung*, Ph.D. thesis, Ruhr-Universität Bochum, Dez. 2000.
- [Eva03] ———, *Design of Digital Systems for Arbitrary Sampling Rate Conversion*, EURASIP J. Signal Processing **83** (2003), no. 2, 377–387.
- [Far88] C. W. Farrow, *A continuously variable digital delay element*, Proc. IEEE Int Circuits and Systems Symp, 1988, pp. 2641–2645.
- [FLL<sup>+</sup>93a] F. Francesconi, G. Lazzari, V. Liberali, F. Maloberti, and G. Torelli, *Multiplier-free lagrange interpolators for oversampled d/a converters*, Proc. IEEE Int Circuits and Systems, ISCAS '93 Symp, 1993, pp. 219–222.
- [FLL<sup>+</sup>93b] ———, *A novel interpolator architecture for sigma-delta dacs*, Proc. [4th] European Conf. with the European Event in ASIC Design Design Automation, 1993, pp. 249–253.
- [Fom00] S. Fomel, *On the general theory of data interpolation*, Stanford Exploration Project **Report SERGEY** (2000), 101–116, <http://sepwww.stanford.edu/oldsep/sergey/sepsergey/genint.pdf>.
- [GH02] R. S. G. Heinzl, A. Rüdiger, *Spectrum and spectral density estimation by the discrete fourier transform (dft), including a comprehensive list of window functions and some new flat-top windows*, Max-Planck-Institut für Gravitationsphysik (2002), [http://www.rssd.esa.int/SP/LISAPATHFINDER/docs/Data\\_Analysis/GH\\_FFT.pdf](http://www.rssd.esa.int/SP/LISAPATHFINDER/docs/Data_Analysis/GH_FFT.pdf).
- [Har78] F. J. Harris, *On the use of windows for harmonic analysis with the discrete fourier transform*, Proceedings of the IEEE **66** (1978), no. 1, 51–83.
- [Har04] F. J. Harris, *Multirate signal processing for communication systems*, Prentice Hall, 2004.
- [Inf] Infinitewave, *Test of sampling rate converters*, <http://src.infinitewave.ca/>.
- [JH07] F. Q. Josef Hoffmann, *Signalverarbeitung mit matlab und simulink*, Oldenbourg, 2007, Abschnitt 7.4 Spektrale Leistungsdichte und Power Spectrum.
- [LBR04] V. Lehtinen, D. Babic, and M. Renfors, *Comparison of continuous-and discrete-time modelling of polynomial-based interpolation filters*, Proc. 6th Nordic Signal Processing Symp. NORSIG 2004, 2004, pp. 49–52.



- [LGS01] T. M. Lehmann, C. Gonner, and K. Spitzer, *Addendum: B-spline interpolation in medical image processing*, IEEE Transactions on Medical Imaging **20** (2001), no. 7, 660–665.
- [Lok05] I. Lokken, *The ups and downs of arbitrary sample rate conversion*, April 2005.
- [Mei02] E. Meijering, *A chronology of interpolation: from ancient astronomy to modern signal and image processing*, Proceedings of the IEEE **90** (2002), no. 3, 319–342.
- [Nar98] F. J. Narcowich, *The dft approximation to the fourier transform*, Department of Mathematics (1998), [http://www.math.tamu.edu/~fnarc/psfiles/fft\\_ft.ps](http://www.math.tamu.edu/~fnarc/psfiles/fft_ft.ps).
- [NCD09] U. Nithirochananont, S. Chivapreecha, and K. Dejhan, *An fpga-based implementation of variable fractional delay filter*, Proc. 5th Int. Colloquium Signal Processing & Its Applications CSPA 2009, 2009, pp. 104–107.
- [Nie01] O. Niemitalo, *Polynomial interpolators for high-quality resampling of oversampled audio*.
- [Pin] A. Pinkus, *Weierstrass and approximation theory*, <http://www.math.umanitoba.ca/HAT/fpapers/wap.pdf>.
- [Smi11] J. O. Smith, *Interpolated delay lines, ideal bandlimited interpolation, and fractional delay filter design*, Lecture Notes, February 2011, <https://ccrma.stanford.edu/~jos/Interpolation/Interpolation.pdf>.
- [SR73] R. W. Schafer and L. R. Rabiner, *A digital signal processing approach to interpolation*, Proceedings of the IEEE **61** (1973), no. 6, 692–702.
- [UAE93a] M. Unser, A. Aldroubi, and M. Eden, *B-spline signal processing. i. theory*, IEEE Transactions on Signal Processing **41** (1993), no. 2, 821–833.
- [UAE93b] ———, *B-spline signal processing. ii. efficiency design and applications*, IEEE Transactions on Signal Processing **41** (1993), no. 2, 834–848.
- [Uns99] M. Unser, *Splines: a perfect fit for signal and image processing*, IEEE Signal Processing Magazine **16** (1999), no. 6, 22–38.
- [UTY95] M. Unser, P. Thevenaz, and L. Yaroslavsky, *Convolution-based interpolation for fast, high-quality rotation of images*, IEEE Transactions on Image Processing **4** (1995), no. 10, 1371–1381.
- [Ye03] Z. Ye, *Linear phase lagrange interpolation filter using odd number of basepoints*, Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '03), vol. 6, 2003.

