

Schnelle Gegenbauertransformation

Michael Quellmalz

10. Juni 2013

Dieser Vortrag basiert auf dem Artikel „The expansion in Gegenbauer polynomials: A simple method for the fast computation of the Gegenbauer coefficients“ von Enrico De Michaeli und Giovanni Alberto Viano. [1]

1 Berechnung der Gegenbauerkoeffizienten

Die Gegenbauer-Polynome $C_n^\lambda(x)$ sind orthogonale Polynome auf dem Intervall $[-1, 1]$ bezüglich der Gewichtsfunktionen

$$w^\lambda(x) = (1 - x^2)^{\lambda - \frac{1}{2}},$$

sodass

$$\langle C_n^\lambda, C_m^\lambda \rangle_{w^\lambda} = \int_{-1}^1 C_n^\lambda(x) C_m^\lambda(x) w^\lambda(x) dx = \delta_{mn} h_n^\lambda$$

mit den durch

$$h_n^\lambda = \frac{2^{1-2\lambda} \pi \Gamma(n + 2\lambda)}{(n + \lambda) n! \Gamma(\lambda)^2}$$

gegebenen Normierungsfaktoren gilt. Dabei ist Γ die Gamma-Funktion

$$\Gamma(\lambda) = \int_0^\infty t^{\lambda-1} e^{-t} dt.$$

Für $\lambda = \frac{1}{2}$ ist $w^{\frac{1}{2}} \equiv 1$, also entsprechen dann die $C_n^{\frac{1}{2}}$ den Legendrepoly-nomen P_n .

Im Weiteren benötigen wir die folgende Integraldarstellung, die für $\lambda > 0$ gilt:

$$C_n^\lambda(x) = \frac{(n + \lambda) h_n^\lambda}{\pi} \int_0^\pi \left(x + i\sqrt{1 - x^2} \cos(y) \right)^n \sin(y)^{2\lambda-1} dy \quad (1.1)$$

Zunächst betrachten wir die Berechnung der Gegenbauer-Koeffizienten $f_n^\lambda \in \mathbb{C}$ einer Funktion

$$f \in L_w^2(-1, 1) := \left\{ f : (-1, 1) \rightarrow \mathbb{C} : \int_{-1}^1 |f(x)|^2 w(x) dx < \infty \right\}$$

also

$$f_n^\lambda = \frac{1}{h_n^\lambda} \int_{-1}^1 f(x) C_n^\lambda(x) (1-x^2)^{\lambda-\frac{1}{2}} dx \quad (1.2)$$

damit gilt dann in $L_w^2(-1, 1)$

$$f = \sum_{n=0}^{\infty} f_n^\lambda C_n^\lambda. \quad (1.3)$$

Lemma 1. *Es gilt für $\lambda > 0$*

$$C_n^\lambda(\cos u) = \frac{(n+\lambda) h_n^\lambda}{\pi} \frac{e^{-i\pi\lambda}}{\sin(u)^{2\lambda-1}} \int_u^{2\pi-u} e^{i(n+\lambda)t} (2\cos(u) - 2\cos(t))^{\lambda-1} dt. \quad (1.4)$$

Beweis. Aus (1.1) folgt mit $u = \arccos x$, $0 \leq u \leq \pi$

$$C_n^\lambda(\cos u) = \frac{(n+\lambda) h_n^\lambda}{\pi} \int_0^\pi (\cos u + i \sin(u) \cos(y))^n \sin(y)^{2\lambda-1} dy$$

Wir führen jetzt die (komplexen) Substitution $y \mapsto z$ durch mit

$$z = (\cos u + i \sin(u) \cos(y))$$

sodass

$$dz = -\sin(u) \sin(y) dy$$

ist. Der Integrationsweg von z führt dann also in der komplexen Ebene entlang einer Gerade parallel zur imaginären Achse von $\cos u + i \sin(u) = e^{iu}$ nach $\cos u - i \sin(u) = e^{-iu}$. Dank des cauchyschen Integralsatzes wählen wir nun den Integrationsweg mit gleichem Ergebnis entlang des Kreissegments $\{e^{is} : u \geq s \geq -u\}$ und nehmen jetzt t mit $z = e^{it}$ ($u \geq t \geq -u$) als neue Integrationsvariable mit der Zuweisung

$$(\cos u + i \sin(u) \cos(y)) \mapsto e^{it}$$

und damit auch

$$-\sin(u) \sin(y) dy \mapsto e^{it} dt.$$

Weiterhin folgt mithilfe von

$$\begin{aligned}
2e^{it}(\cos t - \cos u) &= e^{it}(e^{it} + e^{-it} - e^{iu} - e^{-iu}) \\
&= e^{2it} + 1 - e^{iu+it} - e^{-iu+it} \\
&= (e^{it} - e^{iu})(e^{it} - e^{-iu}) \\
\leftrightarrow & (i \sin(u) \cos(y) - i \sin(u))(i \sin(u) \cos(y) + i \sin(u)) \\
&= -\sin(u)^2 \cos(y)^2 + \sin(u)^2 \\
&= (\sin(u) \sin(y))^2
\end{aligned}$$

nun

$$\begin{aligned}
C_n^\lambda(\cos u) &= -\frac{(n+\lambda)h_n^\lambda}{\pi} \int_u^{2\pi-u} e^{int} \frac{(2e^{it}(\cos t - \cos u))^{\lambda-1}}{\sin(u)^{2\lambda-1}} dt \\
&= -\frac{(n+\lambda)h_n^\lambda}{\pi \sin(u)^{2\lambda-1}} \int_u^{2\pi-u} e^{i(n+\lambda)t} (2(\cos t - \cos u))^{\lambda-1} dt.
\end{aligned}$$

Wegen $(2(\cos t - \cos u))^{\lambda-1} = e^{-i\pi(\lambda-1)} (2(\cos u - \cos t))^{\lambda-1}$ folgt schließlich

$$C_n^\lambda(\cos u) = \frac{(n+\lambda)h_n^\lambda}{\pi} \frac{e^{-i\pi\lambda}}{\sin(u)^{2\lambda-1}} \int_u^{2\pi-u} e^{i(n+\lambda)t} (2\cos(u) - 2\cos(t))^{\lambda-1} dt.$$

□

Theorem 2. Für die Gegenbauerkoeffizienten f_n^λ ($\lambda > 0$) einer Funktion $f \in L_\omega^2(-1, 1)$ gilt die Darstellung

$$f_n^\lambda = (n+\lambda) \int_{-\pi}^{\pi} \phi(t) e^{int} dt \tag{1.5}$$

mit

$$\phi(t) = \frac{2^{\lambda-1} e^{i\lambda(t-\pi \operatorname{sgn}(t))}}{\pi\lambda} \int_0^{(1-\cos(t))^\lambda} f\left(x^{\frac{1}{\lambda}} + \cos(t)\right) dx, \tag{1.6}$$

wobei sgn die Vorzeichenfunktion (Signumfunktion) ist.

Beweis. Mit der Substitution $x = \cos u$, $dx = -\sin(u) du$ in Gleichung (1.2) und unter Verwendung von (1.4) folgt

$$f_n^\lambda = \frac{1}{h_n^\lambda} \int_0^\pi f(\cos u) C_n^\lambda(\cos u) \sin(u)^{2\lambda} du.$$

Mit der Definition $c_n^\lambda := \frac{f_n^\lambda}{n+\lambda}$ gilt

$$c_n^\lambda = \frac{e^{-i\pi\lambda}}{\pi} \int_0^\pi f(\cos u) \sin(u) \int_u^{2\pi-u} e^{i(n+\lambda)t} (2\cos(u) - 2\cos(t))^{\lambda-1} dt du$$

Die Vertauschung der Integrationsreihenfolge liefert die beiden Integrale

$$\int_0^\pi e^{i(n+\lambda)t} \int_0^t f(\cos u) \sin(u) (2\cos(u) - 2\cos(t))^{\lambda-1} du dt$$

sowie

$$\int_\pi^{2\pi} e^{i(n+\lambda)t} \int_0^{2\pi-t} f(\cos u) \sin(u) (2\cos(u) - 2\cos(t))^{\lambda-1} du dt.$$

Mit der Substitution $(t, u) \mapsto (t - 2\pi, -u)$ im zweiten Integral ist dieses gleich

$$e^{2\pi i\lambda} \int_{-\pi}^0 e^{i(n+\lambda)t} \int_0^t f(\cos u) \sin(u) (2\cos(u) - 2\cos(t))^{\lambda-1} du dt$$

und damit lautet der gesamte Ausdruck

$$\begin{aligned} & c_n^\lambda \\ &= \frac{e^{-i\pi\lambda}}{\pi} \int_0^\pi e^{i(n+\lambda)t} \int_0^t f(\cos u) \sin(u) (2\cos(u) - 2\cos(t))^{\lambda-1} du dt \\ & \quad + \frac{e^{i\pi\lambda}}{\pi} \int_{-\pi}^0 e^{i(n+\lambda)t} \int_0^t f(\cos u) \sin(u) (2\cos(u) - 2\cos(t))^{\lambda-1} du dt \\ &= \int_{-\pi}^\pi \frac{e^{int}}{\pi} \left(e^{i\lambda(t-\pi\operatorname{sgn}(t))} \int_0^t f(\cos u) \sin(u) (2\cos(u) - 2\cos(t))^{\lambda-1} du \right) dt \\ &= \left| \begin{array}{l} x(u) = (\cos u - \cos t)^\lambda \\ dx = \lambda(\cos u - \cos t)^{\lambda-1} \sin(u) du \end{array} \right| \\ &= \int_{-\pi}^\pi e^{int} \left(\frac{2^{\lambda-1} e^{i\lambda(t-\pi\operatorname{sgn}(t))}}{\pi\lambda} \int_0^{(1-\cos t)^\lambda} f\left(x^{\frac{1}{\lambda}} + \cos t\right) dx \right) dt. \end{aligned}$$

□

Mit der Substitution $y = x^{\frac{1}{\lambda}} + \cos(t)$, $dx = \lambda y^{\lambda-1} dy$ folgt die alternative Darstellung

$$\int_0^{(1-\cos(t))^\lambda} f\left(x^{\frac{1}{\lambda}} + \cos(t)\right) dx = \lambda \int_{\cos(t)}^1 f(y) (y - \cos(t))^{\lambda-1} dy.$$

Korollar 3. Für die Legendrepolynome ergibt sich

$$f_n^{1/2} = \left(n + \frac{1}{2}\right) \int_{-\pi}^{\pi} \phi(t) e^{int} dt$$

mit

$$\phi(t) = \frac{\sqrt{2} e^{i\frac{1}{2}(t-\pi \operatorname{sgn}(t))}}{\pi} \int_0^{\sqrt{1-\cos(t)}} f(x^2 + \cos(t)) dx.$$

Wir wollen nun Theorem 2 für einen schnellen Algorithmus zur Berechnung der ersten N Gegenbauerkoeffizienten f_k^λ , $j = 0, \dots, N-1$, einer Funktion f anwenden: Für die Berechnung des Integrals in Gleichung (1.5) verwenden wir die Rechteckregel mit den Quadraturknoten

$$t_j = \frac{2\pi j}{N} - \pi, \quad j = 0, \dots, N-1.$$

Wir machen also die Näherung

$$\int_{-\pi}^{\pi} \phi(t) e^{ikt} dt \approx \frac{2\pi}{N} \sum_{j=0}^{N-1} \phi(t_j) e^{ikt_j}, \quad k = 0, \dots, N-1.$$

Die Summe kann nun mit einer inversen schnellen Fouriertransformation (IFFT) berechnet werden. In (1.6) wird das Integral durch eine Quadraturformel ersetzt. Es ergibt sich somit folgender Algorithmus für die Berechnung der ersten N Gegenbauerkoeffizienten einer gegebenen Funktion f .

Algorithmus 1

Eingabe: Eine Funktion $f : [-1, 1] \rightarrow \mathbb{C}$.

1. Berechne die Funktion $\phi(t_j)$ an den äquidistanten Punkten $t_j = \frac{2\pi j}{N} - \pi$, $j = 0, \dots, N-1$ mittels numerischer Integration von

$$\phi(t) = \frac{2^{\lambda-1} e^{i\lambda(t-\pi \operatorname{sgn}(t))}}{\pi \lambda} \int_0^{(1-\cos(t))^\lambda} f\left(x^{\frac{1}{\lambda}} + \cos(t)\right) dx.$$

2. Berechne

$$c_k^\lambda = \frac{2\pi}{N} \sum_{j=0}^{N-1} \phi(t_j) e^{ijk}, \quad k = 0, \dots, N-1$$

mittels einer FFT (Fast Fourier Transform) des Vektors $[\phi(t_j)]_{j=0}^{N-1}$.

3. Berechne $\tilde{f}_k^\lambda = (k + \lambda) c_k^\lambda$.

Ausgabe: Näherung \tilde{f}_k^λ , $j = 0, \dots, N-1$ der ersten N Gegenbauerkoeffizienten von f .

Für den ersten Schritt muss das Integral in (1.6) numerisch berechnet werden, was verschiedene Methoden zulässt, abhängig beispielsweise davon, ob die Funktion explizit, also als Formel, bekannt ist, oder nur die Funktionswerte an bestimmten Stellen zur Verfügung stehen, wobei wir uns im Folgenden auf Ersteres konzentrieren. Wird im ersten Schritt eine ν -Punkte-Quadraturformel auf κ Subintervallen ausgeführt, so werden dafür $\nu\kappa$ Schritte benötigt. Die FFT im zweiten Schritt benötigt $\mathcal{O}(N \log N)$ Schritte. Der Algorithmus hat insgesamt eine Komplexität von $\mathcal{O}(N(\log(N) + \nu\kappa))$. Ist f beispielsweise 2ν -mal stetig differenzierbar, so ergibt sich für eine ν -Punkt-Gauss-Quadratur auf κ Subintervallen ein Fehler von

$$R_{\nu,\kappa}^\lambda(t) = \frac{B(\nu) \cdot (1 - \cos(t))^{\lambda(2\nu+1)}}{\kappa^{2\nu}} \left[\frac{d^{2\nu}}{dx^{2\nu}} f \left(x^{\frac{1}{\lambda}} + \cos(t) \right) \right]_{x=\xi}$$

mit einer Zwischenwertstelle $\xi \in (0, (1 - \cos(t))^\lambda)$ und einer beschränkten Funktion $B(\nu)$, die noch von den für die Quadratur benutzten, orthogonalen Polynomen abhängt. Der Fehler der Quadratur hängt also im Wesentlichen nur von der Funktion f , beziehungsweise deren Regularität, nicht aber von N ab, und er sinkt (im Allgemeinen) mit steigender Anzahl der Punktauswertungen, sodass ν und κ nur in Abhängigkeit von f gewählt werden können, womit sich eine Komplexität von $\mathcal{O}(N \log(N))$ ergibt. Für die Implementierung in [2] wurde dazu in Schritt 1 eine adaptive Gauss-Kronrod-Methode gewählt.

2 Berechnung der Funktionswerte

Wir wollen nun den umgekehrten Weg gehen, also aus gegebenen Gegenbauerkoeffizienten $f_k^\lambda \in \mathbb{R}$ die Funktionswerte

$$f_{N+1}(x_j) = \sum_{k=0}^N f_k^\lambda C_k^\lambda(x_j)$$

an gewissen, vorgegebenen Punkten $x_j \in [-1, 1]$, $j = 0, \dots, N-1$ berechnen.

Theorem 4. Die Gegenbauerentwicklung

$$\sum_{k=0}^N f_k^\lambda C_k^\lambda$$

einer Funktion $f \in C[-1, 1]$ (f stetig) konvergiere gleichmäßig gegen f auf $[-1, 1]$ für $N \rightarrow \infty$. Dann gilt für $\lambda > 0$ und $u \in [0, \pi]$ die Darstellung

$$f(\cos u) = \frac{1}{\sin(u)^{2\lambda-1}} \int_u^\pi \psi^\lambda(t) \cdot (2 \cos u - 2 \cos t)^{\lambda-1} dt \quad (2.1)$$

mit

$$\psi^\lambda(t) = \frac{2^{2-2\lambda}}{\Gamma(\lambda)^2} \sum_{n=0}^{\infty} c_n^\lambda \cos((n+\lambda)t - \lambda\pi), \quad t \in [0, \pi] \quad (2.2)$$

und

$$c_n^\lambda = \frac{\Gamma(n+2\lambda)}{\Gamma(n+1)} f_n^\lambda.$$

Beweis. Aus (1.4) folgt

$$\begin{aligned} & C_n^\lambda(\cos u) \\ &= \frac{(n+\lambda) h_n^\lambda}{\pi \cdot \sin(u)^{2\lambda-1}} \int_u^{2\pi-u} e^{i((n+\lambda)t - \lambda\pi)} (2 \cos(u) - 2 \cos(t))^{\lambda-1} dt \\ &= \frac{(n+\lambda) h_n^\lambda}{\pi \cdot \sin(u)^{2\lambda-1}} \left(\int_u^\pi + \int_\pi^{2\pi-u} \right) e^{i((n+\lambda)t - \lambda\pi)} (2 \cos(u) - 2 \cos(t))^{\lambda-1} dt \end{aligned}$$

Mit der Substitution $t \mapsto 2\pi - t$ im rechten Integral ist der gesamte Ausdruck

$$\begin{aligned} &= \frac{(n+\lambda) h_n^\lambda}{\pi \cdot \sin(u)^{2\lambda-1}} \int_u^\pi e^{i((n+\lambda)t - \lambda\pi)} (2 \cos(u) - 2 \cos(t))^{\lambda-1} dt \\ &\quad + \frac{(n+\lambda) h_n^\lambda}{\pi \cdot \sin(u)^{2\lambda-1}} \int_u^\pi e^{-i((n+\lambda)t - \lambda\pi)} (2 \cos(u) - 2 \cos(t))^{\lambda-1} dt \\ &= \frac{2(n+\lambda) h_n^\lambda}{\pi \cdot \sin(u)^{2\lambda-1}} \int_u^\pi \cos((n+\lambda)t - \lambda\pi) (2 \cos(u) - 2 \cos(t))^{\lambda-1} dt. \end{aligned}$$

Dies eingesetzt in Gleichung (1.3) liefert dank der vorausgesetzten gleichmäßigen Konvergenz

$$f(\cos u) = \sum_{n=0}^{\infty} f_n^\lambda \frac{2(n+\lambda) h_n^\lambda}{\pi \cdot \sin(u)^{2\lambda-1}} \int_u^\pi \cos((n+\lambda)t - \lambda\pi) (2 \cos(u) - 2 \cos(t))^{\lambda-1} dt.$$

Wieder durch die gleichmäßige Konvergenz der Reihe können Summation und Integral vertauscht werden.

$$f(\cos u) = \int_u^\pi \left(\sum_{n=0}^{\infty} 2(n+\lambda) h_n^\lambda f_n^\lambda \cos((n+\lambda)t - \lambda\pi) \right) \frac{(2\cos(u) - 2\cos(t))^{\lambda-1}}{\pi \cdot \sin(u)^{2\lambda-1}} dt$$

Unter Hinzunahme der Formel für h_n^λ folgt nun, dass $f(\cos u)$ gleich

$$\int_u^\pi \left(\sum_{n=0}^{\infty} \frac{2^{2-2\lambda} \Gamma(n+2\lambda) f_n^\lambda \cos((n+\lambda)t - \lambda\pi)}{n! \Gamma(\lambda)^2} \right) \frac{(2\cos(u) - 2\cos(t))^{\lambda-1}}{\sin(u)^{2\lambda-1}} dt$$

ist, also gilt die Behauptung. \square

Um einen Algorithmus zu erhalten, betrachten wir nur die ersten N Gegenbauerkoeffizienten $f_k^\lambda \in \mathbb{R}$, $k = 0, \dots, N-1$, die restlichen werden Null gesetzt. Für die Summe

$$\sum_{k=0}^N c_k^\lambda \cos((k+\lambda)t_l - \lambda\pi)$$

aus Gleichung (2.2), ausgewertet an den Stellen $t_l = \frac{l\pi}{N}$, $l = 0, \dots, N-1$, ergibt sich

$$\begin{aligned} & \sum_{k=0}^{N-1} c_k^\lambda \cos\left((k+\lambda) \frac{2l\pi}{N} - \lambda\pi\right) \\ &= \frac{1}{2} \sum_{k=0}^{N-1} c_k^\lambda e^{i\pi((k+\lambda)\frac{2l}{N} - \lambda)} + \frac{1}{2} \sum_{k=0}^{N-1} c_k^\lambda e^{-i\pi((k+\lambda)\frac{2l}{N} - \lambda)} \\ &= \frac{1}{2} e^{i\pi\lambda(\frac{2l}{N} - 1)} \sum_{k=0}^{N-1} c_k^\lambda e^{i\pi k \frac{2l}{N}} + \frac{1}{2} e^{-i\pi\lambda(\frac{2l}{N} - 1)} \sum_{k=0}^{N-1} c_k^\lambda e^{-i\pi k \frac{2l}{N}} \\ &= \frac{1}{2} \left(\cos\left(\pi\lambda\left(\frac{2l}{N} - 1\right)\right) + i \sin\left(\pi\lambda\left(\frac{2l}{N} - 1\right)\right) \right) \sum_{k=0}^{N-1} c_k^\lambda e^{i\pi k \frac{l}{N}} \\ & \quad + \frac{1}{2} \left(\cos\left(\pi\lambda\left(\frac{2l}{N} - 1\right)\right) - i \sin\left(\pi\lambda\left(\frac{2l}{N} - 1\right)\right) \right) \sum_{k=0}^{N-1} c_k^\lambda e^{-i\pi k \frac{2l}{N}} \\ &= \frac{1}{2} \cos\left(\pi\lambda\left(\frac{2l}{N} - 1\right)\right) \left(\sum_{k=0}^{N-1} c_k^\lambda e^{i\pi k \frac{2l}{N}} + \sum_{k=0}^{N-1} c_k^\lambda e^{-i\pi k \frac{2l}{N}} \right) \\ & \quad - \frac{1}{2} i \sin\left(\pi\lambda\left(\frac{2l}{N} - 1\right)\right) \left(\sum_{k=0}^{N-1} c_k^\lambda e^{-i\pi k \frac{2l}{N}} - \sum_{k=0}^{N-1} c_k^\lambda e^{i\pi k \frac{2l}{N}} \right), \end{aligned}$$

und das ist schließlich gleich

$$\cos\left(\pi\lambda\left(\frac{2l}{N}-1\right)\right)\operatorname{Re}\left(\sum_{k=0}^{N-1}c_k^\lambda e^{i\pi k\frac{2l}{N}}\right)-\sin\left(\pi\lambda\left(\frac{l}{N}-1\right)\right)\operatorname{Im}\left(\sum_{k=0}^{N-1}c_k^\lambda e^{i\pi k\frac{2l}{N}}\right).$$

Damit haben wir die Berechnung von $\psi^{(\lambda)}(t_l)$, $l = 0, \dots, N-1$, also auf die diskrete Fouriertransformation

$$\sum_{k=0}^{N-1}c_k^\lambda e^{i\pi k\frac{2l}{N}}$$

zurückgeführt, die wieder in $\mathcal{O}(N \log N)$ Schritten durchgeführt werden kann.

Es ergibt sich folgender Algorithmus:

Algorithmus 2

Eingabe: Gegenbauerkoeffizienten $f_k^\lambda \in \mathbb{R}$, $k = 0, \dots, N-1$.

1. Berechne $c_k^\lambda = \frac{\Gamma(n+2\lambda)}{\Gamma(n+1)} f_k^\lambda$, $k = 0, \dots, N-1$.
2. Berechne mittels schneller (inverser) Fouriertransformation

$$\tilde{\psi}_N^\lambda(t_l) = \sum_{k=0}^{N-1}c_k^\lambda e^{i\pi k\frac{2l}{N}}$$

an $t_l = \frac{l\pi}{N}$, $l = 0, \dots, N-1$.

3. Berechne

$$\psi_N^\lambda(t_l) = \frac{2^{2-2\lambda}}{\Gamma(\lambda)^2} \cos(\lambda(t_l - \pi)) \operatorname{Re}\left(\tilde{\psi}_N^\lambda(t_l)\right) - \sin(\lambda(t_l - \pi)) \operatorname{Im}\left(\tilde{\psi}_N^\lambda(t_l)\right)$$

4. Berechne die mittels einer geeigneten Quadraturformel

$$f_N(\cos u_j) = \frac{1}{\sin(u_j)^{2\lambda-1}} \int_{u_j}^{\pi} \psi_N^\lambda(t) \cdot (2(\cos u_j - \cos t))^{\lambda-1} dt \quad (2.3)$$

an den Stellen $u_j \in [0, \pi]$, $j = 0, \dots, N-1$.

Ausgabe: Näherung der Funktionswerte $f_N(\cos u_j)$ an den Stellen $u_j \in [0, \pi]$, $j = 0, \dots, N-1$.

Für $\lambda \in (0, 1)$, also insbesondere für die Legendrepolynome, weist der Integrand eine schwache Singularität auf, was bei der Quadraturmethode berücksichtigt werden muss. Weiterhin steht die Funktion ψ^λ nur an bestimmten Stellen durch die FFT zur Verfügung. In Gleichung (2.3) muss also eine Quadraturmethode genutzt werden, die auf bekannte Stützstellen setzt, beispielsweise die Newton-Cotes-Quadratur. Auch bei diesem Verfahren ist der $\mathcal{O}(N \log N)$ -Teil der Komplexität durch die FFT wieder dominierend.

3 Numerische Tests

Die Algorithmen wurden in [2] implementiert. Für die Berechnung der Gegenbauerkoeffizienten (Algorithmus 1) wurde im ersten Schritt für das Integral das adaptive Gauss-Kronrod-Verfahren `gsl_integration_qag` und für die Fouriertransformation im zweiten Schritt `gsl_fft_complex_radix2_backward`, jeweils aus der GNU Scientific Library (GSL), verwendet. Nachfolgend die Resultate eines Tests mit dem Legendrepolynom $f = P_8 = C_8^{\frac{1}{2}}$ als Testfunktion und $\lambda = \frac{1}{2}$:

N	Rechenzeit (sec)	RMSE
32	0.011000	1.369145e-15
64	0.013000	1.189136e-15
128	0.022000	2.478051e-15
256	0.042000	4.050946e-15
512	0.085000	5.088515e-15
1024	0.169000	8.843815e-15
2048	0.340000	2.304247e-14
4096	0.677000	4.427718e-14
8192	1.426000	4.742550e-14
16384	2.826000	7.550816e-14
32768	5.479000	2.294358e-13
65536	10.978000	5.508943e-13

Tabelle 1: Berechnung der Gegenbauerkoeffizienten

In Tabelle 1 ist die Berechnungszeit und die Wurzel aus dem mittleren quadratischen Fehler (RMSE, root mean square error) in Abhängigkeit von der Anzahl N der berechneten Legendrekoeffizienten zu sehen. Dabei ist der RMSE definiert als

$$\sqrt{\frac{1}{N} \sum_{k=0}^{N-1} |\tilde{f}_k^\lambda - f_k^\lambda|^2}$$

definiert, wobei f_k^λ die tatsächlichen Legendrekoeffizienten und \tilde{f}_k^λ die hier berechnete Näherung ist. Man erkennt hier den annähernd linearen Zusammenhang zwischen der Berechnungszeit und N . Diese Ergebnisse sind auch in Abbildung 3.1 und Abbildung 3.2 dargestellt.

Der Algorithmus 2 zur Umkehrung ist auch implementiert worden. Hier werden aber zur Berechnung von u_j in Gleichung (2.3) die Newton-Cotes-Quadratur (standardmäßig die Regel von Boole mit vier Stützstellen) jeweils an den Stellen t_l , $l = j, \dots, N - 1$ genutzt, was also $4 \sum_{j=0}^{N-1} (N - j - 1) = \mathcal{O}(N^2)$ Punktauswertungen. Werden also aus N Gegenbauerkoeffizienten N Funktionswerte berechnet, ergibt sich eine Komplexität von $\mathcal{O}(N^2)$ (für diese Implementierung). Dies kann man auch in Tabelle 2 sehen, in der die Berechnungszeit für N

Funktionswerte aus N Koeffizienten berechnet wurde. Dabei wurden wie oben die Koeffizienten für das Legendrepolynom P_8 und $\lambda = \frac{1}{2}$ verwendet.

N	Rechenzeit (sec)
64	0.003
128	0.006
256	0.024
512	0.081
1024	0.315
2048	1.230
4096	4.888
8192	19.536
16384	79.696

Tabelle 2: Berechnung der Funktionswerte

Literatur

- [1] Enrico De Michaeli, Giovanni Alberto Viano (2013): „The expansion in Gegenbauer polynomials: A simple method for the fast computation of the Gegenbauer coefficients“. Journal of Computational Physics, Volume 239, Seiten 112-122. <http://dx.doi.org/10.1016/j.jcp.2011.03.031>
- [2] Enrico De Micheli, Giovanni Alberto Viano (2012, 2013): Software Package Gegenbauer-Transform http://users.ge.ibf.cnr.it/demicheli/gegenbauer_transform.tgz

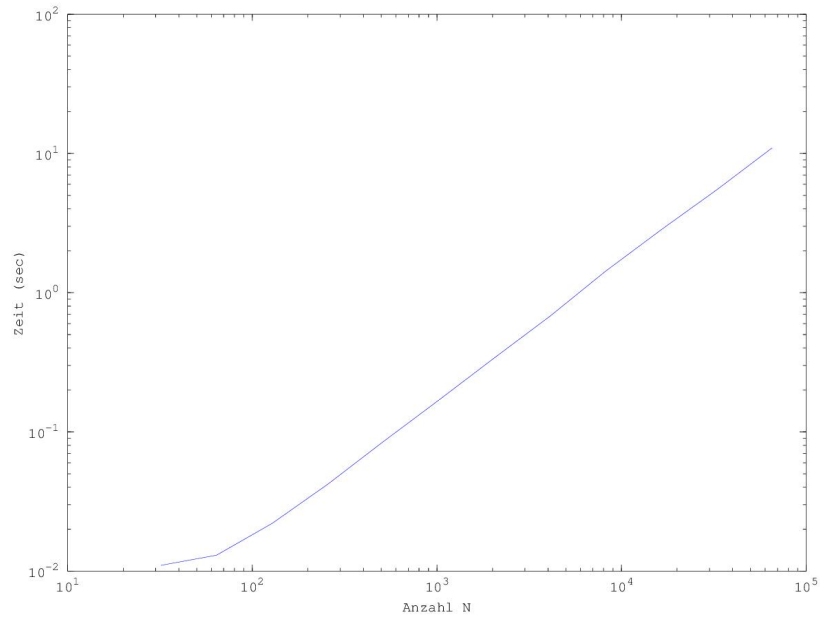


Abbildung 3.1: Zeitkomplexität der Koeffizientenberechnung

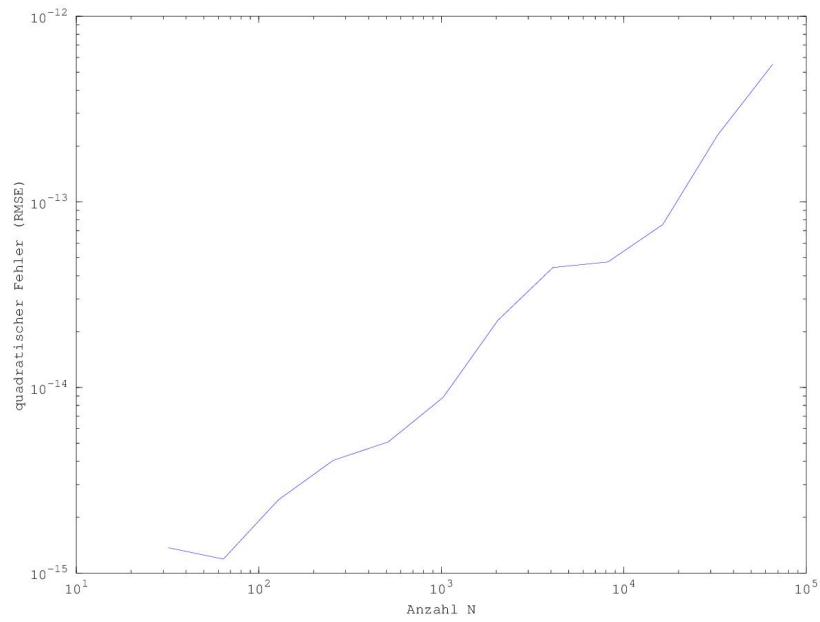


Abbildung 3.2: Berechnungsfehler der Koeffizientenberechnung