

# **Numerieke Modelling en Benadering: Chebyshev veeltermen**

Sander Prenen, r0701014

15 april 2020

# Inhoudsopgave

<b>1</b>	<b>Inleiding</b>	<b>2</b>
<b>2</b>	<b>Continue kleinste kwadratenbenadering met Chebyshev veeltermen</b>	<b>2</b>
2.1	Beste benadering in $V_n$ . . . . .	2
2.1.1	Beste benadering . . . . .	2
2.2	Evalueren van de Chebyshev veeltermen . . . . .	3
2.3	Bepalen van de coëfficiënten . . . . .	3
2.4	Benaderen van een functie . . . . .	4
<b>3</b>	<b>Interpolatie in Chebyshev knooppunten</b>	<b>4</b>
3.1	Interpoleren van een functie . . . . .	6
3.1.1	Cosinus functie interpoleren . . . . .	6
3.1.2	Runge functie interpoleren . . . . .	8
3.1.3	Grootte van de fouten en conditie van de matrix $M$ . . . . .	10

# 1 Inleiding

In dit practicum worden de benaderingseigenschappen van Chebyshev veeltermen van de eerste soort bekeken. Deze veeltermen blijken uiterst geschikt voor het benaderen van eindige, reële functies met behulp van de continue kleinste kwadratenbenadering. Ook zijn hun nulpunten vaak een uitstekende keuze voor de abscissa voor veelterminterpolatie. In sectie 2 wordt dieper ingegaan op de benaderingseigenschappen, terwijl in sectie 3 de interpolatie besproken wordt.

## 2 Continue kleinste kwadratenbenadering met Chebyshev veeltermen

In deze sectie wordt geprobeerd continue functies op eindige, reële intervallen te benaderen aan de hand van Chebyshev-veeltermen van de eerste soort. Dit zijn veeltermen die voldoen aan de volgende voorwaarde:  $T_k(x) = \cos(k \cdot \arccos(x))$  voor  $x \in [-1, 1]$  en  $k = 0, 1, 2, \dots$ . De veeltermen vormen een basis voor de ruimte  $V_n$ , een deelvectorruimte van  $C([-1, 1])$ .

### 2.1 Beste benadering in $V_n$

Indien de basisvectoren  $T_k(x)$  orthogonale vectoren zijn, geldt volgende uitdrukking voor de beste benadering  $y_n(x)$  voor een functie  $f(x) \in C([-1, 1])$ :

$$y_n(x) = \sum_{k=0}^n a_k T_k(x) \quad \text{met} \quad a_k = \frac{(f, T_k)}{(T_k, T_k)} \quad (1)$$

Met als scalair product:

$$(f, g) = \int_{-1}^1 \frac{f(x)g(x)}{\sqrt{1-x^2}} dx$$

Om deze formule te kunnen gebruiken, moet dus worden aangetoond dat de basisvectoren orthogonaal zijn. Dit wil zeggen dat alle onderlinge scalaire producten nul zijn, behalve dat van een basisvector met zichzelf. Volgens bladzijde 69 in de cursus 'Numerieke modellering en benadering' van professor Vandewalle zijn de veeltermen wel degelijk orthogonaal en mag de bovenstaande uitdrukking gebruikt worden. De resultaten van de scalaire producten worden hier nog kort neergeschreven:

$$(T_k, T_j) = \begin{cases} 0, & j \neq k \\ \frac{\pi}{2}, & j = k \neq 0 \\ \pi, & j = k = 0 \end{cases} \quad (2)$$

#### 2.1.1 Beste benadering

Omdat de basisvectoren orthogonaal zijn, kan formule (1) gebruikt worden. In deze sectie worden de coëfficiënten  $a_k$  bepaald.

$$\begin{aligned} a_k &= \frac{(f, T_k)}{(T_k, T_k)} = \frac{1}{(T_k, T_k)} \int_{-1}^1 \frac{f(x)T_k(x)}{\sqrt{1-x^2}} dx = \frac{1}{(T_k, T_k)} \int_{-1}^1 \frac{f(x) \cos(k \arccos(x))}{\sqrt{1-x^2}} dx \\ &= \frac{1}{(T_k, T_k)} \int_{\pi}^0 \frac{f(\cos \theta) \cos(k\theta) (-\sin(\theta))}{\sqrt{1-\cos^2(\theta)}} d\theta = \frac{1}{(T_k, T_k)} \int_0^{\pi} f(\cos \theta) \cos(k\theta) d\theta \end{aligned}$$

Dit geeft dus volgende uitdrukking voor  $a_k$ , gebruikmakend van de uitdrukkingen in vergelijking (2):

$$a_k = \begin{cases} \frac{1}{\pi} \int_0^\pi f(\cos \theta) d\theta, & k = 0 \\ \frac{2}{\pi} \int_0^\pi f(\cos \theta) \cos(k\theta) d\theta, & k > 0 \end{cases} \quad (3)$$

## 2.2 Evalueren van de Chebyshev veeltermen

Om de beste benadering numeriek te bepalen, moet vergelijking (1) geëvalueerd worden. De functie `evalCheb` geeft een vector  $v = (f_1, f_2, \dots, f_N) \in \mathbb{R}^N$  terug. Deze vector wordt bekomen uit inputvectoren  $a = (a_0, a_1, \dots, a_n) \in \mathbb{R}^{n+1}$  en  $x = (x_1, x_2, \dots, x_N) \in \mathbb{R}^N$  op de volgende manier:

$$f_i = y_n(x_i) = a_0 T_0(x_i) + a_1 T_1(x_i) + \dots + a_n T_n(x_i)$$

In Listing 1 wordt de MATLAB code voor deze berekening weergegeven.

```
1 function v = evalCheb(a,x)
2
3 % Lengte van de vectoren bepalen
4 n = length(a);
5
6 % Vector v invullen
7 j = (1:n);
8 v = sum(a .* cos((j-1) .* acos(x')), 2)';
```

Listing 1: evalCheb.m

## 2.3 Bepalen van de coëfficiënten

Voor het bepalen van de coëfficiënten  $a_k$  kan vergelijking (3) gebruikt worden. Deze vergelijking kan met behulp van de trapeziumregel voor numerieke integratie met een discretizatie bestaande uit  $n$  intervallen, benaderd worden door:

$$a_k \approx \frac{1}{n} f(1) + \frac{2}{n} \sum_{l=1}^{n-1} f\left(\cos\left(\frac{l\pi}{n}\right)\right) \cos\left(k \frac{l\pi}{n}\right) + \frac{(-1)^k}{n} f(-1), \quad 0 < k \leq n$$

Hieruit volgt dat de coëfficiënten  $a_k$  kunnen berekend worden als de discrete cosinus transformatie (DCT) van de rij:

$$f(z_{0,n}), f(z_{1,n}), \dots, f(z_{n,n}) \quad (4)$$

Hierin is  $z_{l,n} = \cos(l\pi/n)$ . Deze punten zijn de extrema van  $T_k(x)$  aangezien dit de nulpunten zijn van de eerste afgeleide van  $T_k(x)$ .

$$\frac{dT_k(x)}{dx} = \frac{d}{dx} \cos(k \arccos(x)) = -\sin(k \arccos(x)) \cdot k \cdot \frac{1}{\sqrt{1-x^2}}$$

De afgeleide wordt nul als  $k = 0$  of  $\sin(k \arccos(x)) = 0$ . In het tweede geval herleidt dit zich tot:

$$k \arccos(x) = l\pi \Leftrightarrow x = \cos\left(\frac{l\pi}{k}\right)$$

In de functie `approxCheby` wordt de DCT van de rij in vergelijking (4) bepaald. De code van deze functie kan gevonden worden in Listing 2.

```

1 function a = approxCheby(f,n)
2
3 % z vector maken
4 i = (1:n+1);
5 z = cos((i-1) * pi / n);
6
7 % f evalueren
8 v = f(z);
9
10 % DCT berekenen
11 w = fliplr(v(2:end-1));
12 v_even = [v w];
13 V = fft(v_even)/n;
14 V = real(V(1:n+1));
15 a = [V(1)/2 V(2:end)];

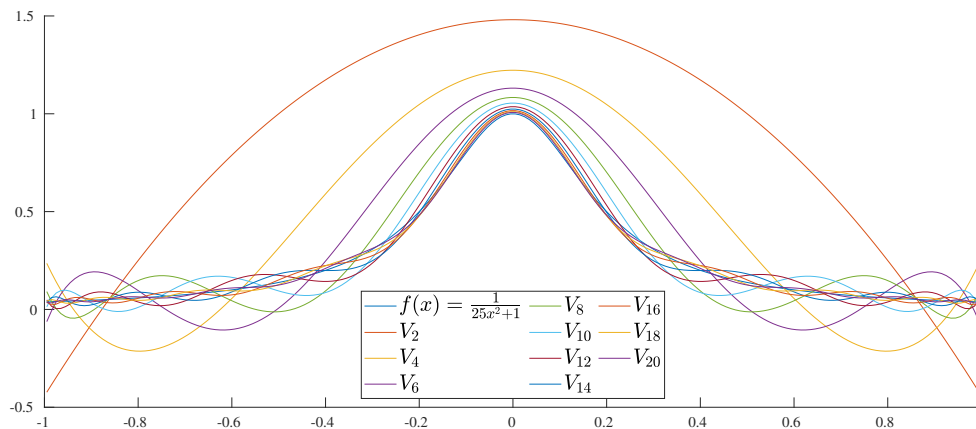
```

Listing 2: approxCheby.m

## 2.4 Benaderen van een functie

De algoritmes die in Listing 1 en 2 te zien zijn, kunnen gebruikt worden voor het benaderen van een willekeurige functie in  $C([-1, 1])$ . Om een functie te benaderen worden de coëfficiënten  $a_k$  bepaald met `approxCheby`. Deze coëfficiënten worden meegegeven aan `evalCheb`. De punten waarin de benadering geëvalueerd moet worden, worden meegegeven in de  $x$  vector. In dit geval wordt gekozen voor 200 equidistante punten tussen  $-1$  en  $1$ .

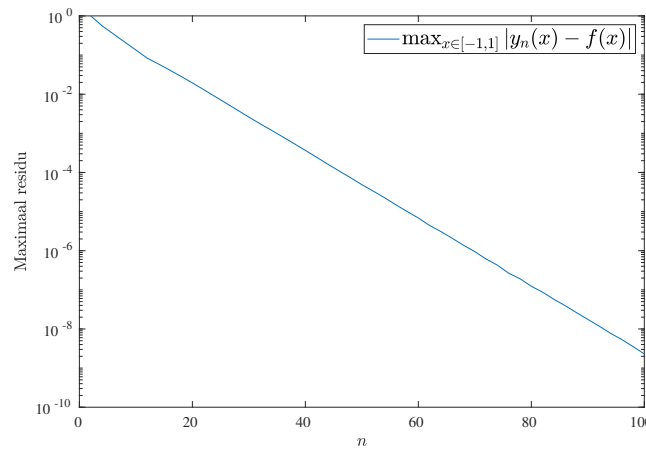
In figuur 1 worden de beste benaderingen voor de Runge functie in ruimtes  $V_2$  tot  $V_{20}$  weergegeven. De Runge functie heeft als voorschrift  $f(x) = \frac{1}{25x^2+1}$ . Figuur 2 toont de maximale afwijking  $d(y_n, f) = \max_{x \in [-1, 1]} |y_n(x) - f(x)|$ , in functie van de graad  $n$  van de benaderende veelterm. Deze fout daalt exponentieel met stijgende  $n$ . Merk op dat deze afstand niet de geïnduceerde afstand van het scalair product is.



Figuur 1: De beste benaderingen van de Runge functie in de ruimtes  $V_2$  tot  $V_{20}$ .

## 3 Interpolatie in Chebyshev knooppunten

Veeltermen van graad  $n - 1$  kunnen een functie interpoleren in  $n$  interpolatiepunten. De interpolatiepunten  $x_k$ , voor  $k = 1, 2, \dots, n$ , kunnen vrij gekozen worden. Er wordt gezocht naar een veelterm  $g(x)$  van graad  $n - 1$  die de functie  $f(x)$  interpoleert, zodanig dat de volgende



Figuur 2: De maximale fout voor de benadering van de runge functie in de ruimtes  $V_2$  tot  $V_{100}$ .

voorwaarde voldaan is:

$$g(x_i) = f(x_i), \quad i = 1, \dots, n$$

De veelterm  $g(x)$  kan worden voorgesteld in de vorm  $g(x) = \sum_{k=0}^{n-1} c_k \psi_k(x)$ . Hierin zijn  $\psi_k(x)$  de basisfuncties, in dit geval dus  $T_k(x)$ . De interpolatievoorwaarden leiden tot volgend lineair stelsel:

$$Mc = B \tag{5}$$

De oplossing van dit stelsel is de vector  $c$  met de coëfficiënten  $c_k$ . De elementen  $M_{ij}$  van de matrix  $M$  op rij  $i$  en kolom  $j$  worden gegeven door:

$$M_{ij} = \psi_j(x_i) = T_j(x_i)$$

en de elementen  $B_i$  van de vector  $B$  worden gegeven door:

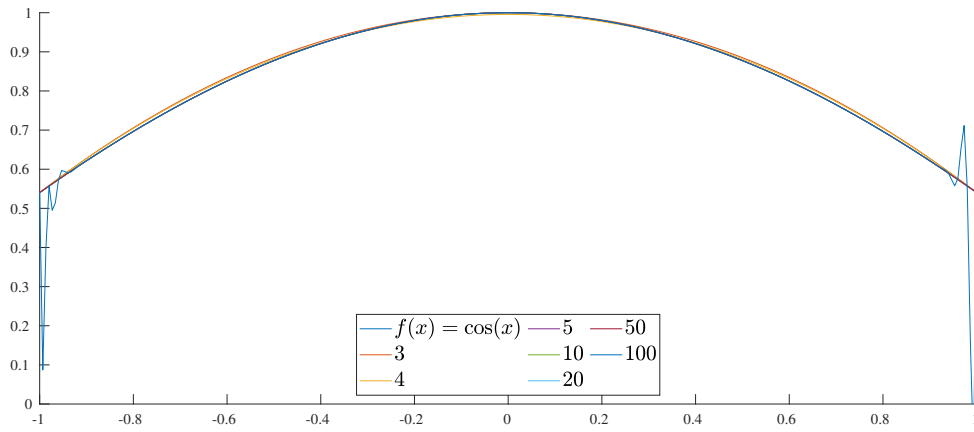
$$B_i = f(x_i).$$

In de functie `interpolate` worden de coëfficiënten  $c_k$  en het conditiegetal van de matrix  $M$  berekend. Als invoer verwacht de functie de interpolatiepunten  $x_k$  en de te benaderen functie  $f(x)$ . De code van deze functie is terug te vinden in Listing 3.

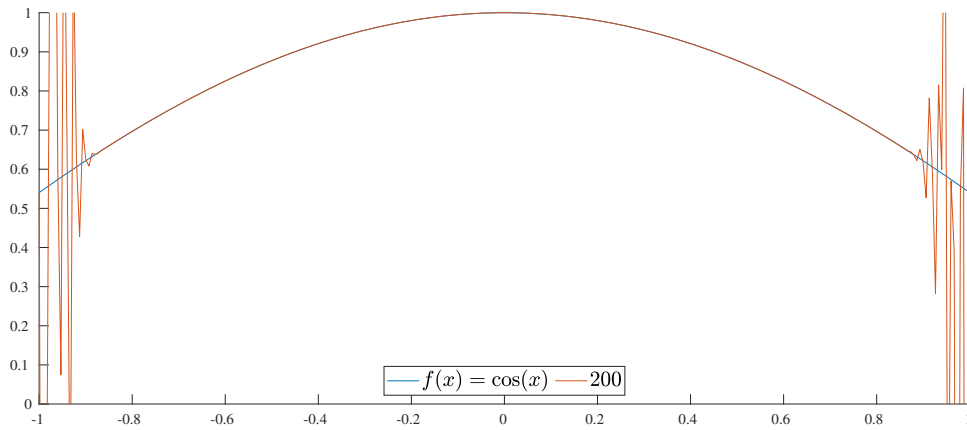
```

1 function [c, kappa] = interpolate(x,f)
2
3 % n bepalen
4 n = length(x);
5
6 % B vector opstellen
7 B = f(x)';
8
9 % M opstellen
10 j = (1:n);
11 M = cos((j-1) .* acos(x'))';
12
13 % Conditiegetal
14 kappa = cond(M);
15
16 % Stelsel oplossen
17 c = M \ B;
```

Listing 3: interpolate.m



Figuur 3: Interpolatie van de functie  $f(x) = \cos(x)$  met een lage  $n$  en equidistante interpolatiepunten. De cijfers in de legende zijn de verschillende  $n$ -waarden.



Figuur 4: Interpolatie van de functie  $f(x) = \cos(x)$  met een hoge  $n$  en equidistante interpolatiepunten. De cijfers in de legende zijn de verschillende  $n$ -waarden.

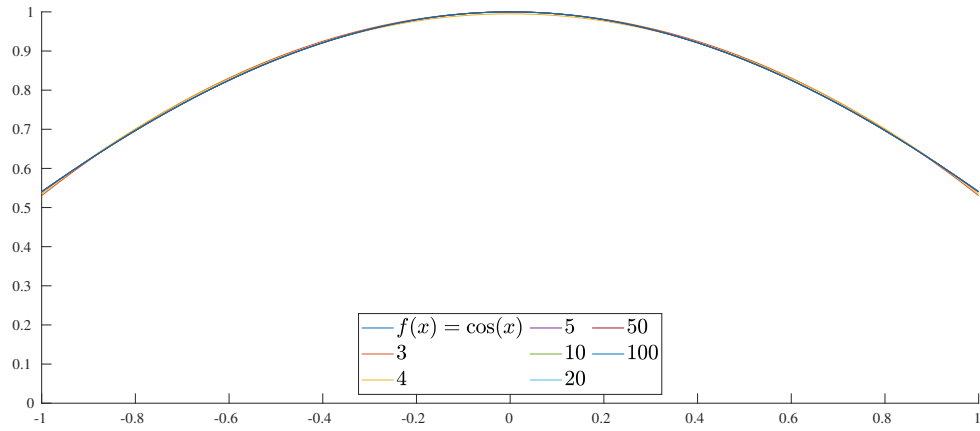
### 3.1 Interpoleren van een functie

#### 3.1.1 Cosinus functie interpoleren

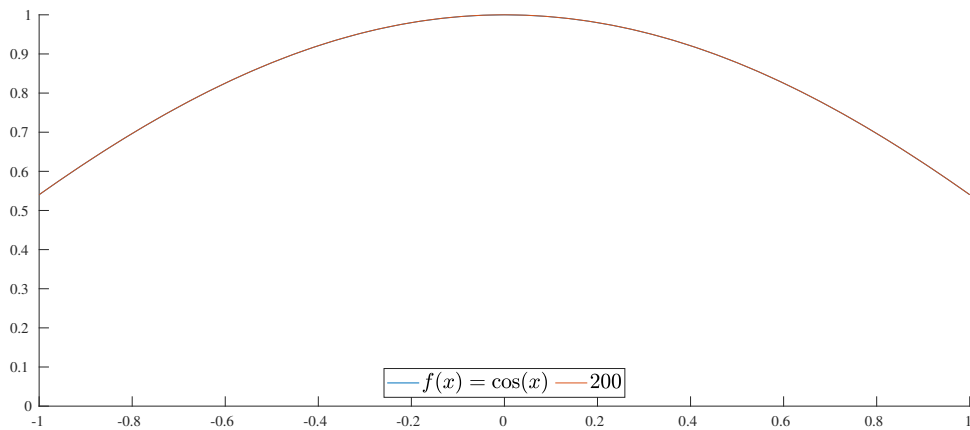
De functie `interpolate` wordt gebruikt om de functie  $f(x) = \cos(x)$  te benaderen. De coëfficiënten  $c_k$  worden meegegeven aan de functie `evalCheb` om de interpolant  $g(x)$  te kunnen plotten. Voor de interpolatiepunten  $x_k$  worden eerst equidistante punten gekozen. De resultaten worden voor verschillende  $n$ -waarden getoond in figuren 3 en 4. Naarmate  $n$  stijgt vertoont de benadering aan de randen meer afwijking ten op zichte van  $f(x)$ . Bij een te hoge  $n$  zoals in figuur 4 is de benadering aan de rand heel slecht doordat de gebruikte methode niet stabiel is bij equidistante interpolatiepunten. Echter wordt de benadering in het midden steeds beter. Door de interpolatiepunten zo te kiezen dat ze de nulpunten zijn van de  $k$ -de Chebyshev veelterm, wordt de methode wel stabiel. Deze nulpunten zijn van de vorm:

$$x_i = \cos\left(\frac{\pi(2i-1)}{2k}\right), \quad i = 1, \dots, k$$

Deze punten worden ook wel de Chebyshev knooppunten genoemd. In figuren 5 en 6 zijn de interpolatieveeltermen met de Chebyshev knooppunten te zien. De fouten aan de randen zijn veel kleiner en zelfs voor hoge  $n$  waarden is de methode stabiel. Dit kan gezien worden in figuur 6.

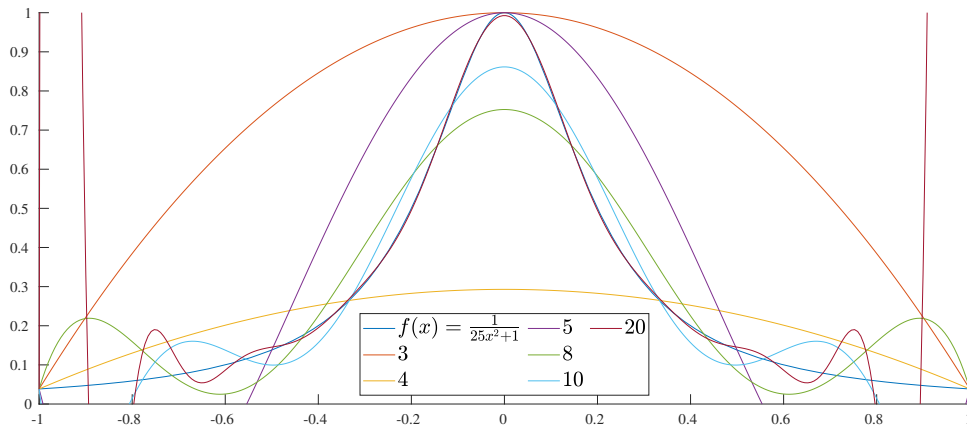


Figuur 5: Interpolatie van de functie  $f(x) = \cos(x)$  met een lage  $n$  en Chebyshev knooppunten. De cijfers in de legende zijn de verschillende  $n$ -waarden.

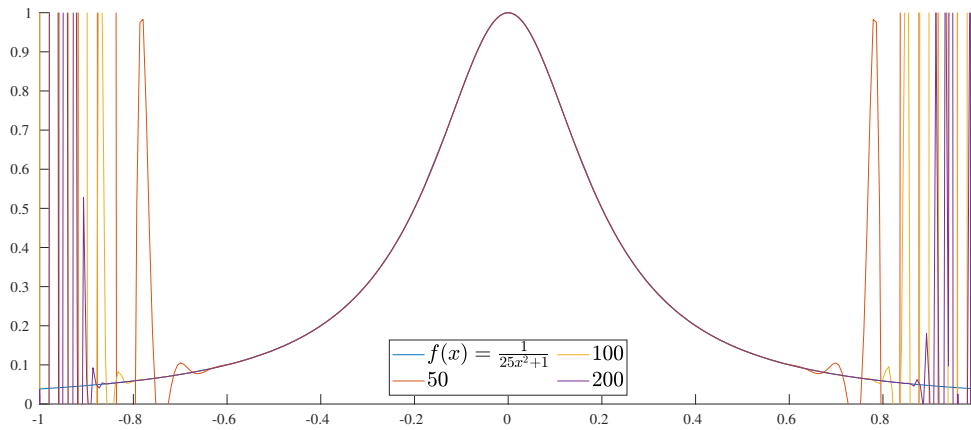


Figuur 6: Interpolatie van de functie  $f(x) = \cos(x)$  met een hoge  $n$  en Chebyshev knooppunten. De cijfers in de legende zijn de verschillende  $n$ -waarden.





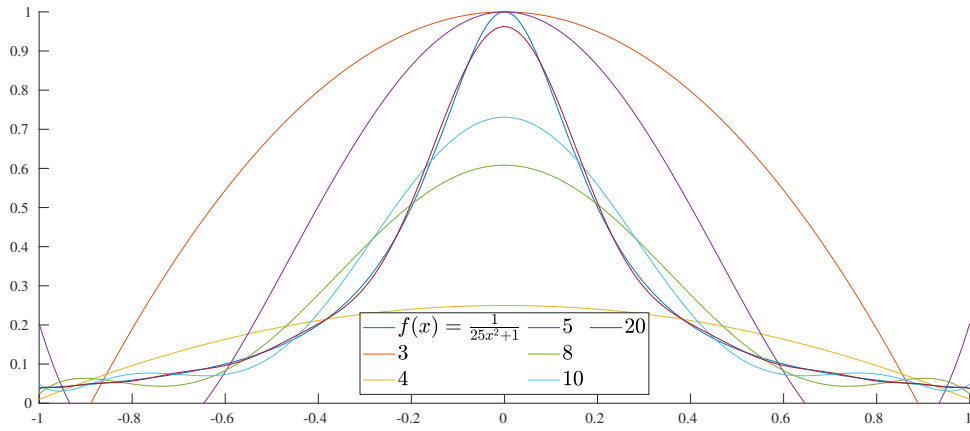
Figuur 7: Interpolatie van de functie  $f(x) = \frac{1}{25x^2+1}$  met een lage  $n$  en equidistante interpolatiepunten. De cijfers in de legende zijn de verschillende  $n$ -waarden.



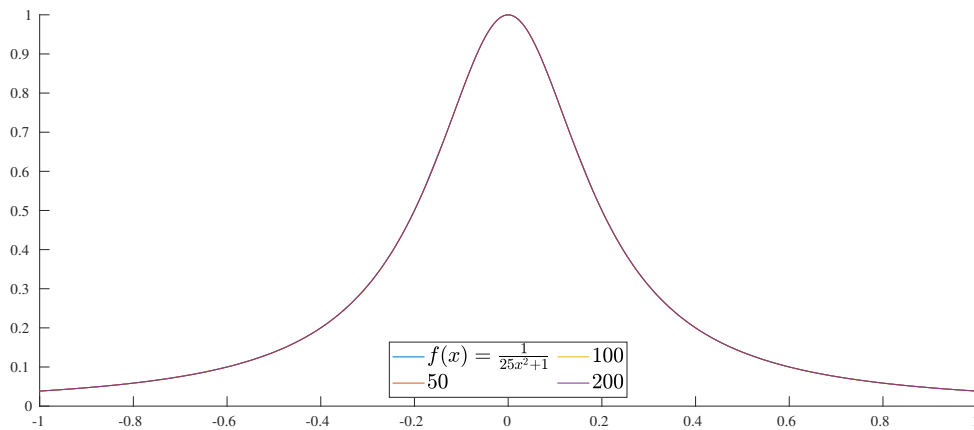
Figuur 8: Interpolatie van de functie  $f(x) = \frac{1}{25x^2+1}$  met een hoge  $n$  en equidistante interpolatiepunten. De cijfers in de legende zijn de verschillende  $n$ -waarden.

### 3.1.2 Runge functie interpoleren

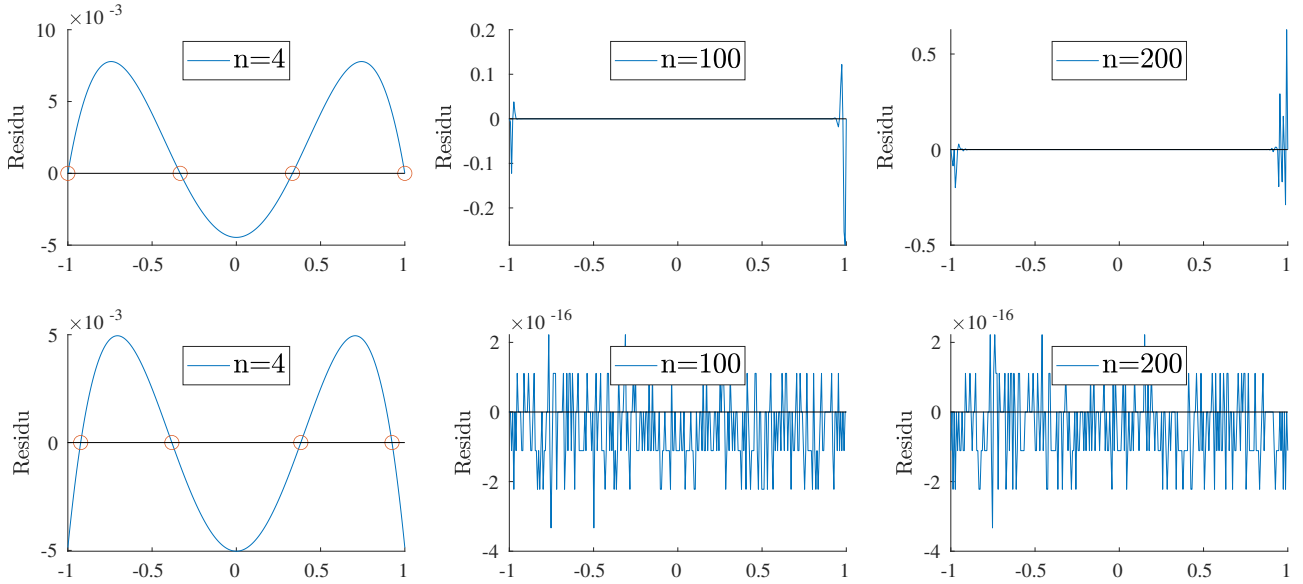
Met eenzelfde werkwijze als in sectie 3.1.1 wordt nu de Runge functie  $f(x) = \frac{1}{25x^2+1}$  geïnterpoleerd. Eerst worden equidistante interpolatiepunten  $x_k$  gebruikt. Deze resultaten zijn te zien in figuren 7 en 8. De fout aan de randen wordt groter bij een stijgende waarde van  $n$ , maar de benadering in het midden van het interval wordt beter. Opnieuw kunnen als interpolatiepunten ook de Chebyshev knooppunten gebruikt worden. Hierdoor wordt de methode stabiel en verdwijnen de fouten aan de randen. De resultaten van deze interpolaties kunnen in figuren 9 en 10 bekeken worden.



Figuur 9: Interpolatie van de functie  $f(x) = \frac{1}{25x^2+1}$  met een lage  $n$  en Chebyshev knooppunten. De cijfers in de legende zijn de verschillende  $n$ -waarden.



Figuur 10: Interpolatie van de functie  $f(x) = \frac{1}{25x^2+1}$  met een hoge  $n$  en Chebyshev knooppunten. De cijfers in de legende zijn de verschillende  $n$ -waarden.

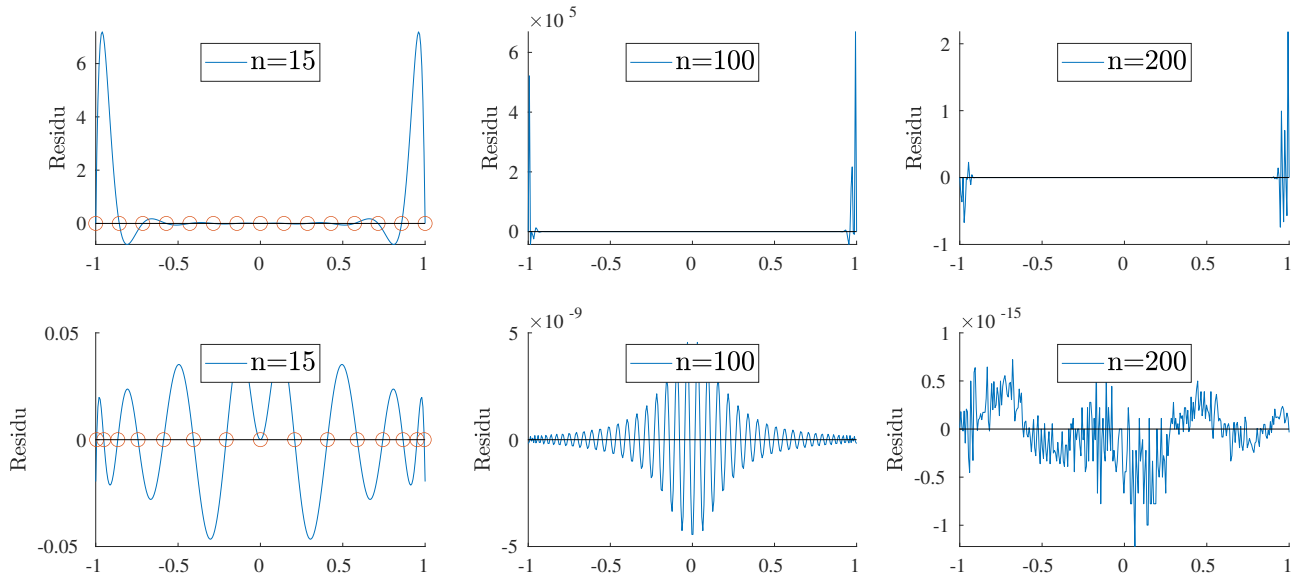


Figuur 11: Fouten over het hele interval  $[-1, 1]$  voor verschillende waarden van  $n$  voor de interpolatie van de cosinus functie. De bovenste rij figuren geeft de fout weer voor equidistante interpolatiepunten, terwijl de onderste rij figuren de fout voor Chebyshev-interpolatiepunten bevat. De interpolatiepunten zijn aangeduid in de eerste plot van elke reeks. Ze zijn weggelaten bij de andere plots om ze niet te onoverzichtelijk te maken.

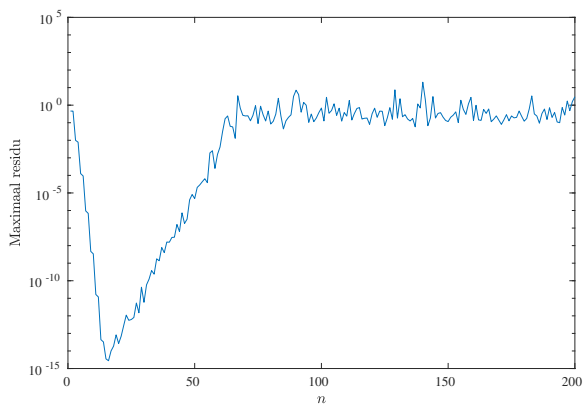
### 3.1.3 Grootte van de fouten en conditie van de matrix $M$

In figuur 11 en figuur 12 wordt de fout over het hele interval  $[-1, 1]$  voor verschillende waarden van  $n$  geplott. Er kan duidelijk gezien worden dat voor equidistante interpolatiepunten de fouten zich vooral voordoen aan de randen en dat deze fout groter wordt met stijgende  $n$ . Voor de Chebyshev-interpolatiepunten is de fout in het midden groter in vergelijking met de randen. Dit is te wijten aan de ligging van de interpolatiepunten. Er liggen meer punten bij de randen waardoor de benadering er beter is. Dit effect is goed te zien op de vijfde plot van figuur 12. In figuren 13 en 14 wordt de maximale afwijking  $d(g, f) = \max_{x \in [-1, 1]} |g(x) - f(x)|$ , in functie van het aantal interpolatiepunten geschetst. Voor equidistante interpolatiepunten daalt de fout eerst, maar door de instabiliteit van de methode zal vanaf een grotere  $n$  de fout weer stijgen. Als de Chebyshev knooppunten als interpolatiepunten worden gekozen dan is de fout monotoon dalend tot de machineprecisie. De zaagtandstructuur die te zien is in figuur 14b is te wijten aan de vorm van de runge functie. Bij een oneven aantal interpolatiepunten, ligt er een interpolatiepunt in het midden van het interval. Hierdoor zal de piek van de runge functie beter benaderd worden en is de fout bijgevolg kleiner.

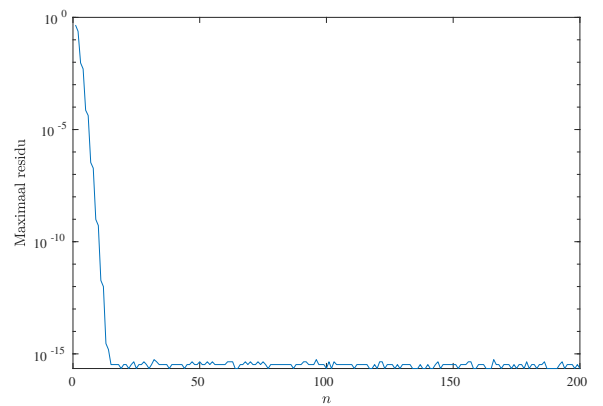
Dat de gebruikte methode instabiel is, kan ook afgeleid worden uit het conditiegetal  $\kappa$  van de matrix  $M$  in vergelijking (5). Dit getal geeft weer hoe het resultaat van een bewerking verandert met de invoer. Hoe hoger het conditiegetal, hoe groter de afwijking op het resultaat zal zijn bij een kleine afwijking van de invoer. In figuur 15 is te zien hoe het conditiegetal  $\kappa$  van de matrix  $M$  evolueert in functie van het aantal interpolatiepunten  $n$ . Bij een equidistante verdeling van de interpolatiepunten, neemt  $\kappa$  al snel een heel grote waarde aan, hetgeen de instabiliteit van de methode verklaart. Indien de nulpunten van de  $k$ -de Chebyshev veelterm gebruikt worden om te interpoleren, blijft  $\kappa$  een constante, lage waarde onafhankelijk van  $n$ . Dit verklaart waarom met deze interpolatiepunten de methode wel stabiel is.



Figuur 12: Fouten over het hele interval  $[-1, 1]$  voor verschillende waarden van  $n$  voor de interpolatie van de Runge functie. De bovenste rij figuren geeft de fout weer voor equidistante interpolatiepunten, terwijl de onderste rij figuren de fout voor Chebyshev-interpolatiepunten bevat. De interpolatiepunten zijn aangeduid in de eerste plot van elke reeks. Ze zijn weggelaten bij de andere plots om ze niet te onoverzichtelijk te maken.

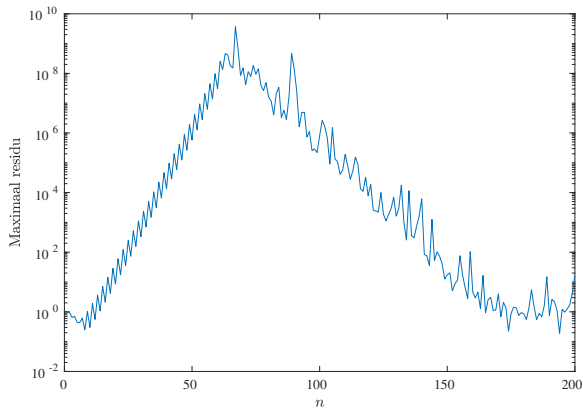


(a) Cosinus interpolatie met equidistante interpolatiepunten

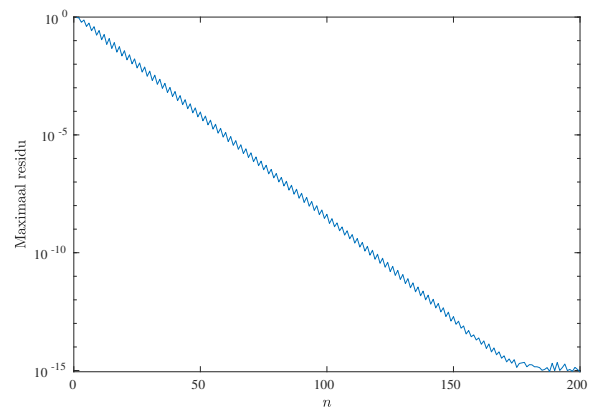


(b) Cosinus interpolatie met Chebyshev knooppunten

Figuur 13: Fout bij interpolatie van de cosinus in functie van het aantal interpolatiepunten  $n$

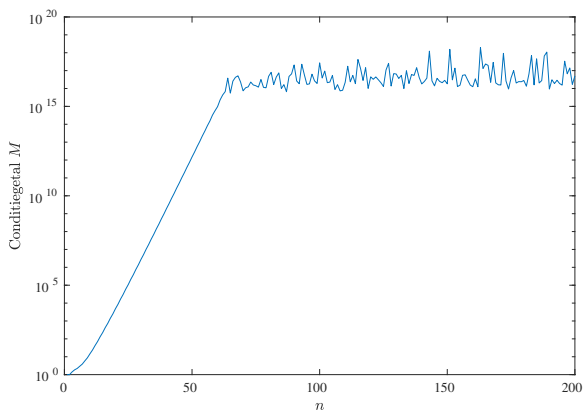


(a) Runge functie interpolatie met equidistante interpolatiepunten

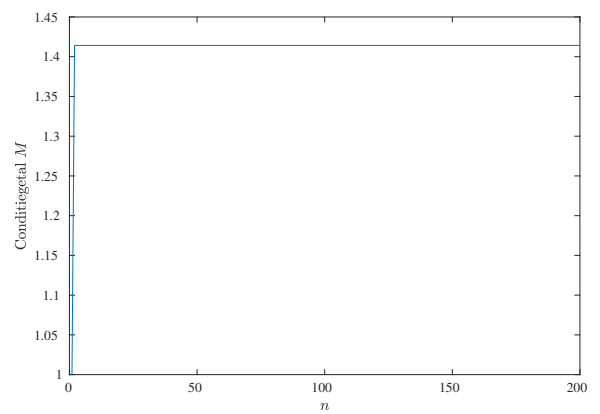


(b) Runge functie interpolatie met Chebyshev knooppunten

Figuur 14: Fout bij interpolatie van de runge functie in functie van het aantal interpolatiepunten  $n$ .



(a) Conditiegetal bij equidistanteinterpolatiepunten



(b) Conditiegetal bij interpolatie met Chebyshev knooppunten

Figuur 15: Conditiegetal  $\kappa$  van matrix  $M$  in functie van het aantal interpolatiepunten  $n$