

Knooppunten Vandermonde-matrices en spline-interpolatie lissajousfiguur

Robbe Vidts (r0836496) - Ferre Van der Vieren (r0851455)
Numerieke wiskunde [G0N90B]

14 september 2022

1 Introductie

Het oplossen van het stelsel $Vx = b$ met V een Vandermonde-matrix en x, b kolomvectoren is een veelvoorkomend probleem bij allerlei wiskundige toepassingen, zoals onder andere interpolatie of Reed-solomon codes. Vandermonde-matrices zijn van de vorm

$$V = \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^{n-1} \end{bmatrix}$$

met $x \in \mathbb{R}^n$. Er zijn echter enkele problemen bij het oplossen van dergelijke stelsels. We bespreken deze kort.

Opdracht 1. Een Vandermonde-matrix is logischerwijs vol, wat het oplossen ervan intensief maakt. We kunnen een stelsel van de vorm $Ax = b$ met $A \in \mathbb{R}^{n \times n}$ algemeen oplossen door de methode van Gauss: eerst herleiden we A tot een bovendriehoeksmatrix (eliminatiefase), waarna het driehoekig stelsel uitgewerkt kan worden door substitutie.

Het terugbrengen van de matrix tot bovendriehoeksvorm kan gerealiseerd worden met behulp van Algoritme 3.2 (uit het handboek Inleiding tot numerieke wiskunde). Hierbij veroorzaakt de binnenste lus, voor elke iteratie van de buitenste, $(n-1)((n-i+1)(n-i))$ operaties, en de buitenste lus wordt uitgevoerd voor $i = 1, 2, \dots, n-1$, waardoor het totale aantal operaties gelijk is aan

$$\sum_{i=1}^n (n-1)((n-i+1)(n-i)) \quad (1)$$

Dit resulteert in een complexiteit van $\mathcal{O}(n^3)$. Het resulterende bovendriehoekig stelsel kan dan opgelost worden met achterwaartse substitutie, waarbij men simpelweg zal itereren voor $i = n, n-1, \dots, 1$ waarbij alle x_n, x_{n-1}, \dots, x_1 berekend worden. Dit heeft slechts een kwadratische tijdscomplexiteit. Het algemeen stelsel $Ax = b$ is dus op te lossen met $\mathcal{O}(n^3)$ bewerkingen; we houden hierbij echter geen rekening met de gekende structuur van een Vandermonde-matrix.

Een efficiëntere methode bestaat erin gebruik te maken van interpolerende veeltermen. Het oplossen van het stelsel $Vx = b$ met V een Vandermonde-matrix en v_1, \dots, v_n de knooppunten van V is namelijk equivalent met het vinden van de interpolerende veelterm $y_n(v) = x_0 + x_1v + \dots + x_nv^{n-1}$ waarvoor $y_n(v_i) = b_i$ met $i = 0, 1, \dots, n$. Dit coëfficiëntenprobleem kan opgelost worden met interpolatie volgens Lagrange, waarbij de veelterm van graad n geschreven wordt als

$$y_n(v) = l_0(v)b_0 + l_1(v)b_1 + \dots + l_n(v)b_n \quad (2)$$

zodat $l_i(v_j) = \delta_{ij}$. Deze coëfficiënten $l_i(v_j)$ kunnen berekend worden met de formule

$$l_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{(x - x_j)}{(x_i - x_j)} \quad (3)$$

We kunnen dan de veelterm uit (3) rangschikken volgens machten van x , en zo de coëfficiënten vinden van de originele interpolerende veelterm, wat het resultaat van het stelsel $Vx = b$ oplevert (de coëfficiënten x_i). Deze methode vergt $\mathcal{O}(n^2)$ operaties. Dit is dus een significante verbetering over het gebruiken van de algemenere methode van Gauss.

Opdracht 2. Een ander probleem bij het oplossen van dergelijke stelsels is dat de Vandermonde-matrix (vaak) slecht geconditioneerd is. Indien een probleem slecht-geconditioneerd is, zal in het algemeen de oplossing van het probleem (de output) significant wijzigen wanneer de gegevens (de input) wijzigt. Dit is ongewenst gedrag omdat minimale fouten in de gegevens (bv. afgeronde machinegetallen) een sterke invloed zullen hebben op het resultaat. Op die manier kan de bekomen output sterk afwijken van het gewenste resultaat.

We gebruiken vaak experimentele gegevens in het geval van interpolerende veeltermen. Stel dat we n experimentele punten (x_i, f_i) hebben verzameld, en dat we uit de theorie weten dat de punten een lineair verband moeten weergeven, dus dat $f = cx$ over heel het domein en met c een constante. Uit het experiment blijkt echter dat de gevonden punten niet exact op een rechte lijn liggen, de punten vertonen dus een fout. Stel dat het interpolatieprobleem slecht geconditioneerd is, dan zal de resulterende interpolatieveelterm niet lijken op de veelterm die we verwachten volgens de theorie, in dit voorbeeld een rechte. De veelterm zal bijvoorbeeld sterk fluctueren rond de verwachte rechte doordat deze van de n -de graad is, in plaats van de 1-ste graad.

Opdracht 3. Stel V een Vandermonde-matrix en beschouw de j -de Lagrange veelterm $l_j(x)$ op knooppunten x_1, \dots, x_n met $l_j(x) = \sum_{i=1}^n u_{ij} x^{i-1}$, dan is de matrix W met $W_{ij} = u_{ij}$ de inverse van V .

Bewijs. We weten dat voor de j -de Lagrange veelterm $l_j(x)$ voor i en $j = 0, 1, \dots, n$ geldt:

$$l_j(x_i) = \delta_{ij} = \begin{cases} 1, & \text{als } i = j \\ 0, & \text{als } i \neq j \end{cases}$$

met δ_{ij} de Kronecker-delta. We kunnen nu de Lagrange veelterm uitschrijven en bekomen zo:

$$l_j(x_i) = \sum_{i=1}^n u_{ij} x_i^{i-1} = u_{1j} + u_{2j} x_i + \dots + u_{nj} x_i^{n-1} = \delta_{ij}$$

Dit kan geschreven worden als matrix-product van een $(1 \times n)$ -matrix met een $(n \times 1)$ -matrix:

$$[1 \quad x_i \quad \dots \quad x_i^{n-1}] \times [u_{1j} \quad u_{2j} \quad \dots \quad u_{nj}]^T$$

De eerste matrix kunnen we uitbreiden door de 0-tot- $(n-1)$ ste machten van de overgebleven punten onder elkaar te schrijven. We bekomen dan:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^{n-1} \end{bmatrix} \times [u_{1j} \quad u_{2j} \quad \dots \quad u_{nj}]^T = [\delta_{1j} \quad \delta_{2j} \quad \dots \quad \delta_{nj}]$$

Merk op dat dit precies de Vandermonde matrix V is. De tweede matrix kunnen we uitbreiden door alle coëfficiënten van de Lagrange veelterm voor $j = 1, \dots, n$ naast elkaar te schrijven. Dit levert:

$$V \times \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ u_{21} & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \dots & u_{nn} \end{bmatrix} = \begin{bmatrix} \delta_{11} & \delta_{12} & \dots & \delta_{1n} \\ \delta_{21} & \delta_{22} & \dots & \delta_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{n1} & \delta_{n2} & \dots & \delta_{nn} \end{bmatrix}$$

De tweede matrix, die bestaat uit de coëfficiënten van de Lagrange veeltermen, vormt matrix W . Als resultaat krijgen we een $(n \times n)$ -matrix die volledig bestaat uit Kronecker-delta's. Omdat $i = j$ op de hoofd-diagonaal en $i \neq j$ elders bekomen we als het ware de eenheidsmatrix I_n . De totale bewerking geeft ons dus $V \times W = I_n$. Wanneer het product van twee vierkante matrices A en B de eenheidsmatrix is, dan geldt uit de definitie van inverse matrices dat B de inverse is van A (en omgekeerd), met andere woorden dat $B = A^{-1}$. In dit geval is de matrix W dus de inverse van de Vandermondematrix V , dus $W = V^{-1}$. \square

2 Knooppunten in \mathbb{R}

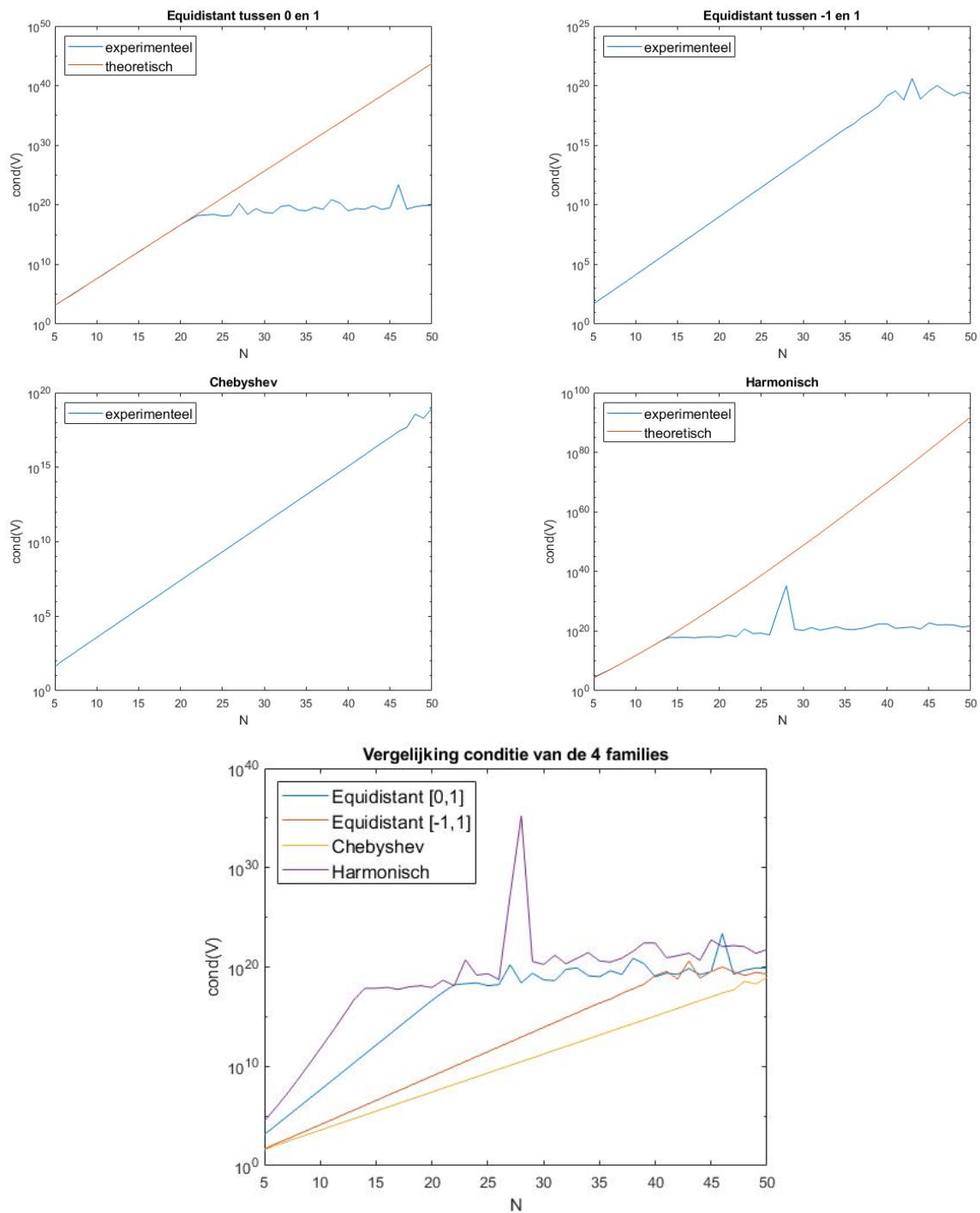
Opdracht 4. Indien voor alle knooppunten van een Vandermonde-matrix V geldt dat $0 \leq x_1, \dots, x_n \leq 1$, zal $\|V^{-1}\|_1$ zijn bovengrens bereiken. De 1-norm van een algemene matrix A is gelijk aan $\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$. Voor V is deze uitdrukking gelijk aan n , aangezien dat $0 \leq x_1, \dots, x_n < 1$, waardoor de machten van een knooppunt x_i ook steeds positief en kleiner dan 1 zullen zijn. De som van de elementen van de eerste kolom van V is dus de grootste mogelijke som van kolomelementen, waardoor $\|V\|_1 = \sum_{i=1}^n |v_{i1}| = \sum_{i=1}^n 1 = n$. We weten dat het conditiegetal van een niet-singuliere matrix A gelijk is aan $K(A) = \|A\|_1 \|A^{-1}\|_1$, en aangezien V met verschillende knooppunten niet-singulier is ($\det(V) \neq 0$), geldt er vanwege het voorgaande

$$K(V) = \|V\|_1 \|V^{-1}\|_1 = n \left(\max_j \prod_{i \neq j} \frac{1 + |x_i|}{|x_i - x_j|} \right) \quad (4)$$

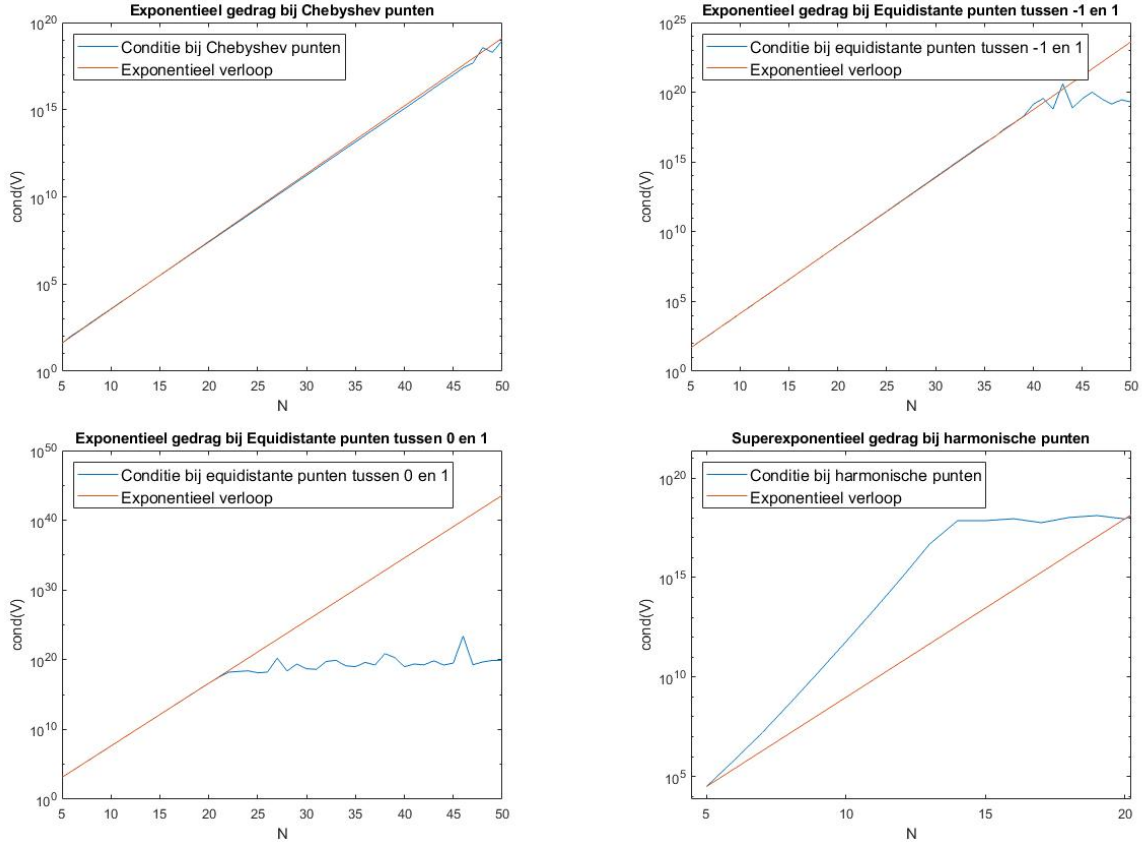
Opdracht 5. De twee families uit opdracht 4 waarvoor de conditie theoretisch bepaald kan worden zijn de families van equidistante punten tussen 0 en 1 en de familie van harmonische punten. Deze punten zijn namelijk allen positief zodat we uitdrukking (4), die we eerder afgeleid hebben, kunnen gebruiken. Deze zullen we vergelijken met de experimentele waarden berekend met Matlab. De resultaten zijn te zien in [Figuur 1]. Merk op dat de gegevens geplot zijn op een logaritmische schaal. De oranje grafiek geeft de theoretische conditie weer (indien van toepassing) en de blauwe de experimentele, telkens voor de waarden $N = 5, \dots, 50$. Het is duidelijk te zien dat de theoretische waarden een perfect exponentieel verband weergeven, en dat de experimentele condities voor kleinere N -waarden de theoretische conditie goed weerspiegelen, maar ze voor grotere N afwijken.

We vergelijken nu de conditie van de vier families door deze te plotten naast elkaar. We zien dat alle families een exponentieel verband tonen totdat ze plots stagneren rond dezelfde waarde, ongeveer 10^{20} . Dit is te verklaren. Voor grote waarden van N worden heel grote machten berekend van de knooppunten, waardoor machten van waarden gelegen in $[-1, 1]$ naderen naar 0 en machten buiten dat interval (met uitzondering van -1 en 1 zelf) naar oneindig. Matlab zal in die gevallen werken met een getal dicht bij 0 (maar niet gelijk aan 0) of een zeer groot getal. De precisie is echter beperkt waardoor getallen kleiner dan deze ondergrens of groter dan de bovengrens steeds tot dezelfde waarden afgerond worden. Als gevolg zullen grootteordes binnen de bewerkingen die matlab uitvoert vanaf een bepaalde N ongeveer dezelfde zijn, waardoor de resultaten niet meer sterk zullen variëren en in dit geval dus niet meer stijgen.

De conditie van de Vandermonde-matrix bij de harmonische knooppunten vertoont licht superexponentieel gedrag. We kunnen dit zien op [Figuur 2]. De oranje rechte geeft een exponentieel verloop weer van vorm be^{ax} , waarvan de parameters a en b te vinden zijn in [Tabel 1], en is geconstrueerd zodat het begin van beide curven gelijk lopen. Als men goed kijkt ziet men inderdaad dat de blauwe curve geen rechte lijn vormt, vooral niet in het begin. Op een logaritmische schaal betekent dit dat het verloop superexponentieel is.



Figuur 1: $\text{cond}(V)$ voor de verschillende families



Figuur 2: exponentieel gedrag $\text{cond}(V)$ voor de verschillende families

De andere functies gedragen zich (asymptotisch) exponentieel want ze vertonen een lineair verband op een logaritmische schaal. Ze hebben de vorm be^{ax} met als gevolg dat we parameters a en b kunnen schatten om er een benaderende exponentiële functie naast te plotten. We gaan als volgt te werk. We nemen 2 punten (x_1, y_1) en (x_2, y_2) op de functie voor de conditie van de familie. Daardoor willen we een exponentieel verloop tekenen, waardoor het volgende moet gelden:

$$\begin{cases} be^{ax_1} = y_1 \\ be^{ax_2} = y_2 \end{cases}$$

Hier de waarden voor a en b uithalen is moeilijk, waardoor we van beide kanten het natuurlijk logaritme nemen.

$$\begin{cases} \ln(be^{ax_1}) = \ln(y_1) \\ \ln(be^{ax_2}) = \ln(y_2) \end{cases} \Leftrightarrow \begin{cases} \ln(b) + ax_1 = \ln(y_1) \\ \ln(b) + ax_2 = \ln(y_2) \end{cases}$$

Dit stelsel kunnen we oplossen als een lineair stelsel met 2 vergelijkingen en 2 onbekenden. In MatLab stellen we bijhorende matrices op zodat we met behulp van de backslash-operator a en $\ln(b)$ kunnen vinden. Om ook b te vinden plaatsen we $\ln(b)$ in de exponent zodat $b = e^{\ln(b)}$. De waarden die we bekomen voor a en b voor de overige families zijn hieronder weergegeven.

Parameters	Harmonisch	Equidistant [0, 1]	Equidistant [-1, 1]	Chebyshev
a	3.0616	2.0689	1.1221	0.89518
b	0.0069053	0.045028	0.18292	0.46658

Tabel 1: a en b voor de verschillende families (voor oranje grafiek be^{ax} in [Figuur 2])

Opdracht 6. Beschouw nu het stelsel $Vx = b$ met knooppunten v_1, \dots, v_n met $b = [1 \dots 1]^T$. We zoeken dus de vector $x = [x_1, \dots, x_n]$ waarvoor geldt dat

$$\begin{cases} x_1 + v_1 x_2 + v_1^2 x_3 + \dots + v_1^{n-1} x_n = 1 \\ x_1 + v_2 x_2 + v_2^2 x_3 + \dots + v_2^{n-1} x_n = 1 \\ \dots \\ x_1 + v_n x_2 + v_n^2 x_3 + \dots + v_n^{n-1} x_n = 1 \end{cases}$$

Dit is een stelsel met n onbekenden x_1, \dots, x_n en n onafhankelijke vergelijkingen (aangezien $\det(V) \neq 0$). Het heeft dus maar één oplossing. We vinden de evidente oplossing $x = (1, 0, 0, \dots, 0)^T$, en kunnen dus ook concluderen dat er geen andere oplossingen mogelijk zijn.

Stel nu $n = 7$ en beschouw de familie van Harmonische punten $\{1/v | v = 1, \dots, n\}$. We brengen een perturbatie δb aan op b , en lossen het stelsel $V\tilde{x} = (b + \delta b)$ op. Onderstaande tabel toont de bekomen voorwaartse fouten en bijhorende bovengrenzen voor vijf verschillende perturbaties. Daaronder is de gebruikte code weergegeven.

Perturbatie	Relatieve fout	Bovengrens
1	5.089e-06	1.3312e-05
2	9.7313e-07	7.643e-06
3	2.8837e-08	1.3252e-05
4	3.063e-06	9.3121e-06
5	4.0598e-06	1.8759e-05

Tabel 2: voorwaartse fouten en bijhorende bovengrenzen voor perturbaties

```

% --- OPDRACHT 6 --- %

%%% we kiezen de harmonische punten %%%

mean = 0;
dev = 10^(-12);
b = ones([7,1]);          % n = 7
harm = zeros(1,7);
for v = 1:7
    harm(v) = 1/v;
end
V = Vandermonde(harm);
conditie = cond(V, 1); % conditiegetal (norm = 1)

% voorwaartse relatieve fouten en bijhorende bovengrenzen voor tabel
relatieve_fouten = zeros(1, 5);
bovengrenzen = zeros(1, 5);

for i = 1:5
    perturbatie = normrnd(mean, dev, [1, 7]).';

    x_fault = V \ (b + perturbatie.*b);
    x_nofault = V \ (b);

    % we blijven werken met norm 1
    norm_delta = norm(perturbatie, 1);
    norm_b = norm(b, 1);

    bovengrenzen(i) = conditie*(norm_delta/norm_b);
    relatieve_fouten(i) = norm((x_fault - x_nofault),1) / norm(x_nofault, 1);
end

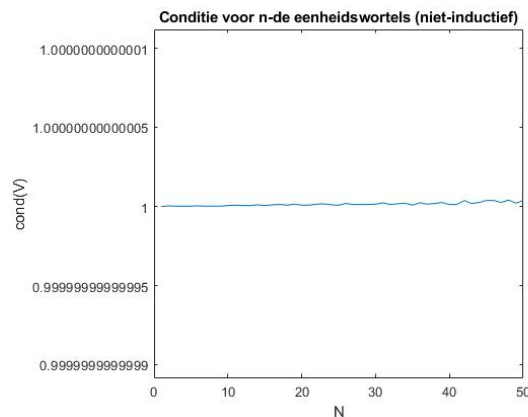
% tabel opstellen
T = table([1:5].', relatieve_fouten.', bovengrenzen. ');
T.Properties.VariableNames = ["Perturbatie", "Relatieve fout", "Bovengrens"];
disp(T);

```

Figuur 3: Matlab code voor [Tabel 2] te berekenen

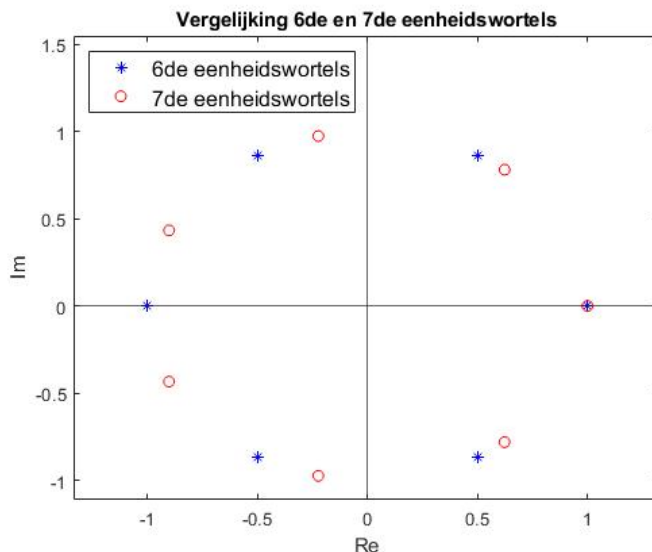
3 Knooppunten in \mathbb{C}

Opdracht 7. De Vandermonde-matrices met dimensie $n = 1 \dots 50$ en n de eenheidswortels als knooppunten hebben inderdaad een uitstekende conditie, wat duidelijk te zien is in [Figuur 4]. De conditie van V is - zoals te zien is - gelijk aan 1 voor alle $n = 1 \dots 50$.



Figuur 4: conditie voor n -de eenheidswortels (niet-inductief)

Opdracht 8. De 6de en 7de eenheidswortels staan geplot in [Figuur 5], respectievelijk met blauwe sterren en rode bollen. Alle knooppunten behalve punt $1 + 0i$ verschillen. In de volgende paragraaf verklaren we de ligging van deze n de eenheidswortels, en geven we ook een methode om hun volgorde visueel te voorspellen.

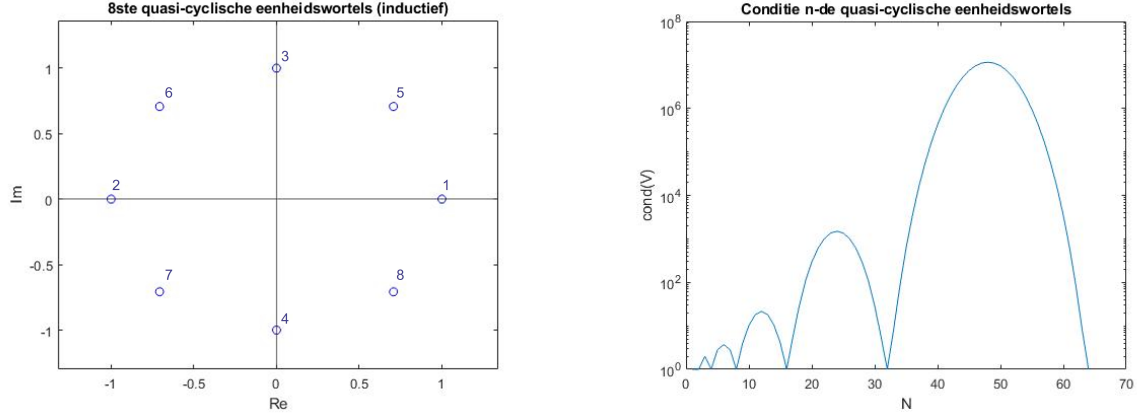


Figuur 5: ligging 6de en 7de eenheidswortels

Opdracht 9. We zullen nu de n de eenheidswortels omvormen tot een inductieve familie, beginnend met quasi-cyclische knooppunten. Het resultaat voor de eerste acht eenheidswortels is te zien in [Figuur 6]. De eerste vier eenheidswortels liggen op de reële en imaginaire as met als waarden $1 + 0i$, $-1 + 0i$, $0 + 1i$, $0 - 1i$. De overgebleven 8ste eenheidswortels liggen telkens tussen twee van de vier eerdere punten, op een cirkel die alle acht punten snijdt. We kunnen als volgt de volgorde van de eenheidswortels visueel voorspellen. De eerste eenheidswortel ligt logischerwijs vast en is gelijk aan $1 + 0i$. Neem de cirkel met straal 1 en middelpunt $0 + 0i$. Stel dan k gelijk aan 0. De volgende (2^k) de punten liggen op deze cirkel telkens tussen twee eerdere eenheidswortels. De volgorde wordt bepaald door de cirkel in tegenwijzerszin te overlopen startend uit punt $1 + 0i$. De $(2^{(k+1)})$ de eenheidswortels zijn dan gevonden. Dan wordt k verhoogd met 1, enzovoort.

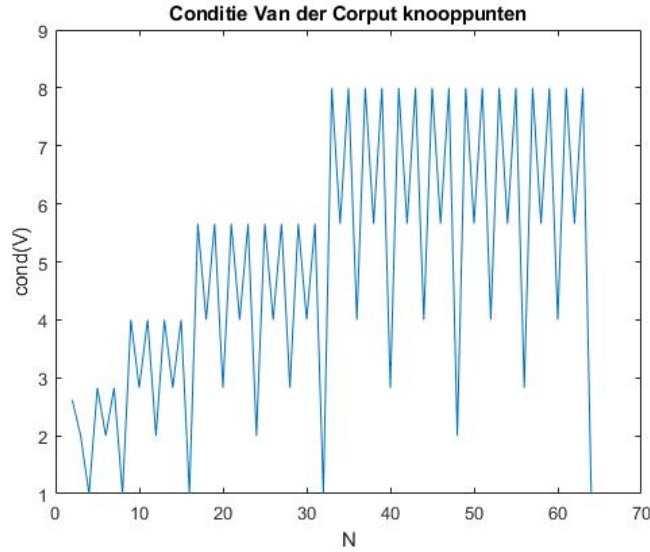
We illustreren dit tot en met $k = 3$. De eerste eenheidswortel ligt op $1 + 0i$ en we starten met $k = 0$, dus zoeken $2^k = 1$ ander punten. De twee eenheidswortels waartussen dit punt moet liggen op de cirkel is voor dit geval twee keer dezelfde, namelijk de eerste eenheidswortel $1 + 0i$. Het punt wordt dus $-1 + 0i$, zodat we de $(2^{(k+1)} = 2)$ de eenheidswortels hebben gevonden. We verhogen nu k met 1. We zoeken dus $2^k = 2$ andere punten. Deze zijn gelegen tussen twee eerdere eenheidswortels op de cirkel. In dit geval zijn dit $1 + 0i$ en $-1 + 0i$, zodat de twee nieuwe punten op $0 + 1i$ en $0 - 1i$ verschijnen, respectievelijk. De $(2^{(k+1)} = 4)$ de eenheidswortels zijn geconstrueerd en we kunnen k opnieuw verhogen om de 8ste eenheidswortels te vinden door vier extra punten op de cirkel te positioneren tussen de 4de eenheidswortels.

We zien ook in [Figuur 6] dat de conditiegetallen van de geassocieerde Vandermonde-matrices bij quasi-cyclische knooppunten steeds schommelen; een lokaal minimum wordt bereikt in machten van twee (bv. bij $n = 16$), en een maximum middenin opeenvolgende machten van twee (bv. bij $n = 24$). Dit is logisch, want voor machten van 2 vormen de punten in het imaginair vlak een puntspiegeling. Er is dus het meest mogelijke symmetrie, waardoor de conditie het laagst is. De conditie is onacceptabel groot voor deze waarden. We kunnen echter beter doen aan de hand van de Van der Corput knooppunten.



Figuur 6: ligging en cond(V) voor quasi-cyclische eenheidswortels

Opdracht 10. We plotten opnieuw de conditiegetallen voor de geassocieerde Vandermonde-matrices voor $n = 1, \dots, 64$, maar nu voor de Van der Corput knooppunten in [Figuur 7]. We merken een minder grotere stijging in conditie als in [Figuur 6], maar nu wel een schommeling tussen even en oneven waarden. Dit is dan ook logisch, aangezien voor even waarden er meer symmetrie is, en zoals al eerder vermeld zal daardoor de conditie lager zijn.



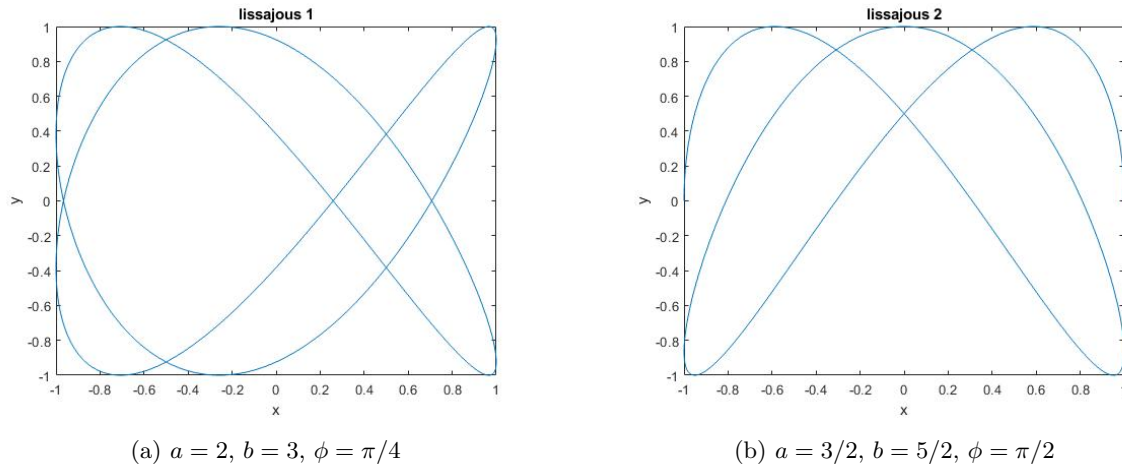
Figuur 7: cond(V) voor Van der Corput knooppunten

4 Spline-interpolatie

Opdracht 12. We definiëren een lissajousfiguur door de parametrisatie

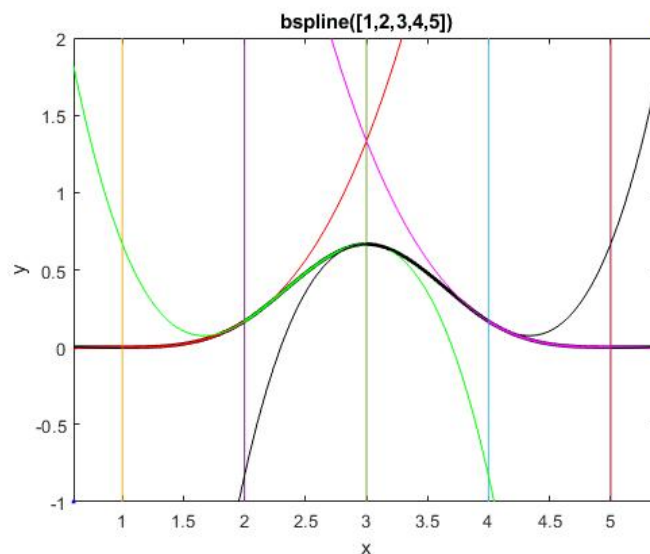
$$\delta : [0, 2\pi] \rightarrow \mathbb{R}^2 : \theta \rightarrow \begin{pmatrix} \sin(a\theta + \phi) \\ \sin(b\theta) \end{pmatrix} \quad (5)$$

met a, b en ϕ parameters. Hieronder zijn twee lissajousfiguren met verschillende parameters weergegeven.



Figuur 8: lissajousfiguren

Opdracht 13. We geven nu de functie `bspline([1,2,3,4,5])` in (in Matlab). Dit resulteert in [Figuur 9]. De bspline is van orde vier, aangezien we werken met vijf knooppunten, en bestaat dus uit onderdelen van vier derdegraadsfuncties (groen, rood, zwart en paars). De verticale lijnen markeren de verschillende knooppunten: in dit specifiek voorbeeld zijn de x-waarden van de knooppunten 1,2,3,4 en 5. De gevonden bspline gaat in elk van deze knooppunten op een vloeiende wijze over in een andere derdegraadsfunctie, wat duidelijk te zien is op de figuur.



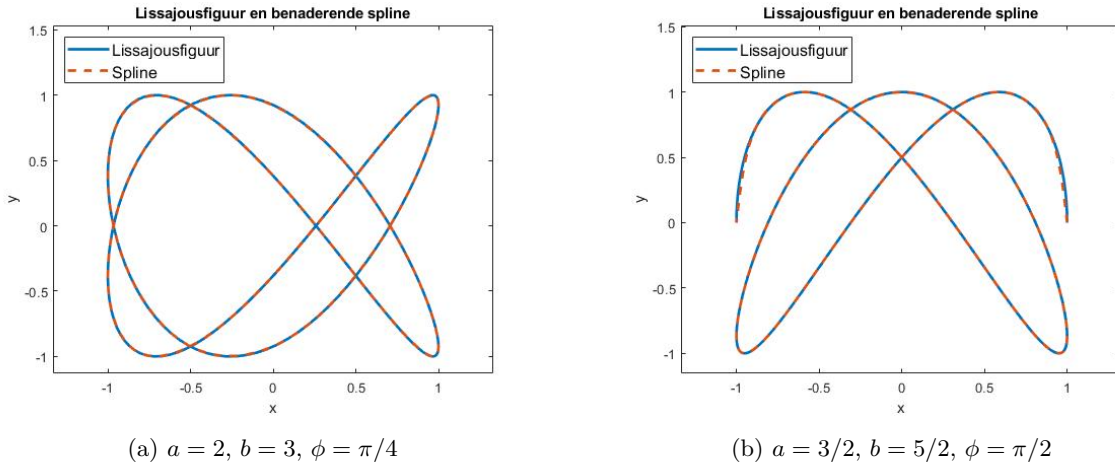
Figuur 9: `bspline([1,2,3,4,5])`

Opdracht 14. We implementeren nu de functies `cubicsplinematrix(t, tbefore, tafter, periodic)` en `cubicsplinesolve(t, tbefore, tafter, periodic, f, k)`. De parameter `periodic` voegt twee extra (rand)voorwaarden toe aan de kubische b-spline functie $s(x)$: $s^{(2)}(t_0) = s^{(2)}(t_n) = 0$ indien we werken met een natuurlijke kubische spline-functie en $s^{(j)}(t_0) = s^{(j)}(t_n)$ voor $j = 1, 2$ voor een periodieke spline-functie. We beschrijven deze periodieke randvoorwaarden als $s^{(j)}(t_0) - s^{(j)}(t_n) = 0$ voor $j = 1, 2$ om de implementatie in Matlab te vereenvoudigen. Op die manier blijft de vector $[0; f_0; f_1; \dots; f_n; 0]$ dezelfde vorm behouden in het stelsel

voor zowel natuurlijke als periodieke voorwaarden (eerste en laatste element is een 0).

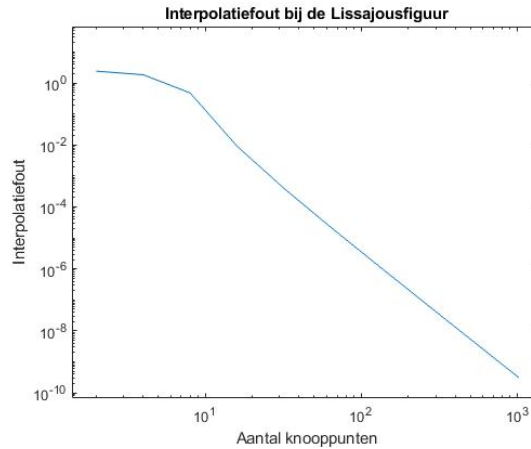
Stel dat t een vector is met $n + 1$ knooppunten, dan is de matrix A van de functie *cubicsplinematrix* van dimensie $n + 3$. De twee randvoorwaarden en de $n + 1$ interpolatievoorwaarden $s(x_i) = f_i$ voor de knooppunten $(x_i, f_i), i = 0, \dots, n$ in de vector t leiden namelijk tot $n + 3$ voorwaarden in de matrix A . Kubische splines zijn per definitie van graad $k = 3$, en dus hebben we in totaal $2 * 3 = 6$ bijkomende knooppunten $t_{-3}, t_{-2}, t_{-1}, t_{n+1}, t_{n+2}, t_{n+3}$ nodig. De parameters *tbefore* en *tafter* bevatten dus elk drie knooppunten.

Opdracht 15. We interpoleren nu de twee lissajousfiguren uit [Figuur 8] met 20 equidistante interpolatiepunten en $k = 200$. [Figuur 10] toont de resultaten van deze interpolatie. De lissajousfiguur wordt gegeven met de volle lijn, en de benaderende spline met de streepjeslijn. Zoals te zien is op de figuur benadert de spline de lissajousfiguur heel accuraat.



Figuur 10: benaderende splines van lissajousfiguren

Opdracht 16. We beschouwen nu enkel de lissajousfiguur met $a = 2, b = 3, \phi = \pi/4$ en interpoleren met 2^r interpolatiepunten. De bekomen spline is dan $s(\theta) := (s_x(\theta), s_y(\theta))^T$. We plotten nu de interpolatiefout $\|s - \delta\|_{\inf}$ ten opzichte van het aantal knooppunten 2^r in [Figuur 11].



Figuur 11: interpolatiefout bij lissajousfiguur

Als de afstand tussen opeenvolgende punten halveert, zal het aantal knooppunten verdubbelen. De stapgrootte is dus steeds twee in dit geval, want bij elke stap wordt er verdubbeld. Bij een laag aantal punten ($2^1, 2^2$) zal de interpolatiefout afnemen met een kleine factor. Maar vanaf ongeveer acht punten zien we een sterkere en consistente afname; de interpolatiefout neemt dan telkens af met een factor van ongeveer vijftig. Dit is dan ook de maximale daling die we waarnemen. Bij veel grotere waarden van het aantal knooppunten (bv. $n = 2^9$) merken we dat de afname minder dan vijftig wordt.

We kunnen daarom stellen dat de orde van de methode gelijk is aan vijf, immers $2^5 \leq 50 \leq 2^6$ (stapgrootte is gelijk aan twee) en 2^5 vormt dus de grootste ondergrens van macht twee voor de factor ($= 50$) waarmee steeds verkleind wordt per stapgrootte. Dit komt overeen met wat we verwachten uit de theorie. We gebruiken definitie 5.3 (uit het boek Inleiding tot de numerieke wiskunde) om ons antwoord te staven. In dit geval is de stapgrootte $h = 2$ en dus geldt

$$\max_{1 \leq k \leq n} |d_k| \leq O(2^{p+1})$$

Zoals we reeds besproken hebben is $2^5 \leq 50 \leq 2^6$, wat resulteert in

$$\max_{1 \leq k \leq n} |d_k| \leq O(2^6)$$

waardoor volgens de theorie $p = 5$, hetgeen klopt met onze interpretatie van [Figuur 11].