Practicum Numerieke Wiskunde

Bachelor Informatica-Wiskunde

Academiejaar 2021-2022

Introductie

Wanneer we een interpolerende veelterm zoeken aan de hand van de monomiaalbasis reduceert het probleem tot het oplossen van een stelsel $Ax = b$, met A een Vandermonde-matrix. Ook bij andere numerieke problemen durft de Vandermonde wel eens opduiken, bijvoorbeeld bij numerieke integratie (Gauss-kwadratuur). Een verdere analyse van het gedrag van deze matrices kan dus zeker interessant zijn. Een dergelijk stelsel oplossen is problematisch om twee redenen:

- De matrix is vol (zeer weinig elementen zijn nul). Dit stelsel oplossen vergt veel werk.
- De matrix is bovendien (vaak) slecht geconditioneerd.

Opdracht 1. (theorievraag) Betreffende het eerste punt, hoeveel werk is nodig om een dergelijk stelsel op te lossen? In het geval van interpolerende veeltermen, hoe wordt dit probleem omzeild?

Opdracht 2. (theorievraag) Betreffende het tweede punt, leg uit wat het betekent dat een probleem slecht-geconditioneerd is. Kopieer hier niet woord voor woord de definitie uit de cursus, maar toon dat je het concept "conditie" hebt begrepen. In het geval van interpolerende veeltermen, willen we vaak interpoleren door experimentele gegevens. Waarom is dit relevant?

In het eerste deel van dit practicum zullen we focussen op het tweede punt. Wat zijn enkele eigenschappen van zo'n Vandermonde-matrix? En kunnen we deze theoretische eigenschappen experimenteel (i.e. numeriek) nagaan? We herhalen de verschillende definities. Een matrix A wordt soms genoteerd als $[a_{ij}]$, waarbij a_{ij} (eventueel $a_{i,j}$) het element van A voorstelt op positie (i, j) (i.e. rij i, kolom j). Gegeven een vector $x \in \mathbb{R}^n$. We veronderstellen $x_i \neq x_j$, wanneer $i \neq j$. De bijhorende Vandermonde-matrix wordt gegeven door

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

De conditie van een niet-singuliere matrix A wordt gegeven door

$$\text{cond}(A) = \|A\| \cdot \|A^{-1}\|,$$

waarbij $\|\cdot\|$ een gepaste matrix-norm is. In het eerste deel van dit practicum werken we met de 1-norm

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^{n} |a_{ij}|.$$

We schrijven cond_1 om verwarring tegen te gaan. Een uitdrukking voor de inverse van een Vandermonde-matrix wordt gevonden als volgt. Beschouw de j-de Lagrange veelterm $\ell_j(x)$[1] op knooppunten $x_1 \dots x_n$. Herschrijf deze als

$$\ell_j(x) = \sum_{i=1}^{n} u_{ij} x^{i-1}.$$

[1] In Latex via `\ell`

Definieer de matrix W, met als elementen

$$W_{ij} = u_{ij}.$$

Opdracht 3. Toon aan dat $V^{-1} = W$. Jullie bewijs zal zowel op correctheid als vorm worden gequoteerd. Herinner onder andere:

- een bewijs bevat volzinnen die de formules en berekeningen introduceren en/of uitleggen;
- nieuwe notatie dient geïntroduceerd te worden;
- een bewijs bevat geen magische stappen. Elke stap is "evident" of wordt ondersteund door een verwijzing naar de cursus, of een verwijzing naar/citatie van een stelling/resultaat/definitie.

Knooppunten in $\mathbb{R}$

Een Vandermonde-matrix hangt natuurlijk af van de keuze van de knooppunten. De volgende stelling mogen jullie als gegeven beschouwen. Het bewijs maakt gebruik van het feit dat we een uitdrukking gevonden hebben voor de inverse van de Vandermonde-matrix.

Stelling 1. *Gegeven een Vandermonde-matrix V gedefinieerd op de knooppunten $x_1 \dots x_n$, dan geldt het volgende resultaat*

$$\max_j \prod_{i \neq j} \frac{\max(1, |x_i|)}{|x_i - x_j|} < \|V^{-1}\|_1 \leq \max_j \prod_{i \neq j} \frac{1 + |x_i|}{|x_i - x_j|}. \tag{1}$$

De bovengrens wordt bereikt indien alle knooppunten positief zijn.

Opdracht 4. Veronderstel dat de knooppunten positief zijn, en kleiner dan 1. Geef een uitdrukking voor het conditiegetal van de bijhorende Vandermonde-matrix. Je formule mag uitdrukkingen uit (1) bevatten.

Een *familie van knooppunten* is een reeks verzamelingen $\{A_n\}_{n \in \mathbb{N}}$ zodat elke A_n exact n knooppunten bevat. De verzameling A_n zijn de knooppunten van dimensie n. We zijn geïnteresseerd wat er gebeurt met de eigenschappen van deze knooppunten wanneer n groot wordt. We beschouwen nu vier families:

- Equidistante punten in het interval $[0, 1]$;
- Equidistante punten in het interval $[-1, 1]$;
- Chebyshev punten $\{\cos((2\nu - 1)\pi/2n) \mid \nu = 1 \dots n\}$;
- Harmonische punten $\{1/\nu \mid \nu = 1 \dots n\}$.

Opdracht 5. Schrijf een Matlab-functie die, gegeven een vector met n verschillende knooppunten, de bijhorende Vandermonde-matrix opstelt. De functie-header ziet er uit als volgt

```
function V = Vandermonde(x)
% x is een verticale vector met n verschillende elementen
% V is de bijhorende nxn Vandermonde-matrix
```

Hou rekening met de efficiëntie van je algoritme. Vervolgens gaan we de conditie (t.o.v. de 1-norm) na voor de verschillende families. We zijn geïnteresseerd wat er gebeurt met de conditie indien n groot wordt. Schrijf een Matlab-script `CondVandReal`. Hierin doe je het volgende:

- Bereken de conditie van de Vandermonde-matrices voor $n = 5 \dots 50$, aan de hand van het matlab commando `cond`. Verifieer zeker op de help-pagina[2] welke opties je kan meegeven aan deze functie!

[2] `https://www.mathworks.com/help/matlab/ref/cond.html`

- Voor de families die in het geval van Opdracht 4 vallen, maak een figuur waarin je de theoretische conditie plot naast de experimentele waarde die je hierboven hebt bekomen.
- Maak een figuur waarin je de resultaten voor de vier verschillende families samen plot, ten opzichte van de waarde n op de x-as.
- De plots vertonen een consistent stijgend gedrag waarna ze plots[3] stagneren. Hoe komt dit? Zorg dat je antwoord voldoende nauwkeurig is.
- De harmonische knooppunten leiden tot licht superexponentieel gedrag. Visueel is dit moeilijk te zien. Ga dit na door er een exponentieel groeiende curve naast te plotten.
- De andere drie functies gedragen zich (asymptotisch) exponentieel (i.e. ze zijn van de vorm $\sim be^{an}$). Hoe kan je dit zien op je figuur? Waar komt de waarde a mee overeen in de functies in de figuur? Geef een schatting van a voor de drie exponentiële functies. Leg ook uit hoe je hieraan komt. Eénmaal een schatting voor a gevonden, zoek dan ook een schatting voor b. Neem je resultaten op in een tabel.

Vergeet niet om de gevraagde figuren op te nemen in je verslag.

Opdracht 6. Uiteindelijk gaan we ook nog even de conditie na door enkele stelsels op te lossen. We beschouwen $b = [1 \dots 1]^\top$ (een vector van lengte n, met een 1 op elke positie). Neem V eender welke Vandermonde-matrix van dimensie n. Wat is de exacte oplossing van het stelsel $Vx = b$? Waarom? Neem $n = 7$, en kies, uit de vier bovenstaande families, je favoriet[4]. We brengen een perturbatie δb aan op de vector b. Begin een nieuwe sectie[5] in je script `CondVandReal`.

- Voor de perturbatie kiezen we een normaal verdeelde vector met gemiddelde 0, en standaarddeviatie 10^{-12}. Tip: bekijk de help-pagina van `randrnd()`.
- Gebruik de backslash operator om de oplossing van het stelsel $V\tilde{x} = (b + \delta b)$ te vinden. Je mag er (voor deze opdracht) van uitgaan dat de backslash operator je de exacte oplossing van dit stelsel geeft. Ga na dat je door middel van het conditiegetal inderdaad een bovengrens bekomt op de voorwaartse relatieve fout. Je mag aannemen dat je geen fout maakt bij de constructie van V. Doe dit experiment voor vijf verschillende perturbaties. Neem de bekomen voorwaartse fouten en bijhorende bovengrenzen op in een tabel. Print de gebruikte code in je verslag[6].

Knooppunten in $\mathbb{C}$

Men kan aantonen dat reële knooppunten altijd leiden tot slecht-geconditioneerde Vandermonde-matrices. Ter illustratie vermelden we het volgende resultaat.

Stelling 2. *Neem n verschillende positieve reële knooppunten, en beschouw de bijhorende Vandermonde-matrix V. Er geldt dat*

$$cond_1(V) > 2^{n-1}.$$

We zijn natuurlijk niet verplicht om knooppunten te nemen die zuiver reëel zijn. Zowaar, indien we complexe knooppunten toelaten, dan zijn er keuzes die wel leiden tot een goed-geconditioneerde Vandermonde-matrix. We beschouwen de familie knooppunten betaande uit de $n$de eenheidswortels, namelijk

$$\{e^{2\pi\imath(\nu-1)/n} \mid \nu = 1 \dots n\}.$$

In deze sectie zullen we werken met de 2-norm[7], dit betekent dat het beschouwde conditiegetal gegeven wordt door

$$\mathrm{cond}_2(A) = \|A\|_2 \cdot \|A^{-1}\|_2.$$

[3] Pun intended

[4] Dit is geen (strik)vraag. Kies die punten die jou het meest in bekoring brengen.

[5] Door middel van `%%`

[6] Door middel van een figuur, of gebruik `\verb|.|`

[7] Deze keuze is pragmatisch. De 2-norm blijkt de meest natuurlijke keuze voor bovenstaande knooppunten.

Sla voor dit onderdeel je code op in een script met de naam `CondVandImag`. Begin een nieuwe sectie voor elke vraag.

Opdracht 7. Ga via Matlab na dat de bijhorende Vandermonde-matrices van dimensie $n = 1 \ldots 50$ inderdaad een uitstekende conditie hebben.

De goede conditie, is een fijn resultaat. Echter de voorgestelde knooppunten zijn niet ideaal, in de volgende zin. Indien we overstappen van $n$de eenheidswortels naar $(n+1)$de eenheidswortels (lees: we verhogen het aantal knooppunten met één), dan veranderen zo goed als alle knooppunten!

Opdracht 8. Ga dit na door de 6de eenheidswortels te plotten in het blauw, en de 7de eenheidswortels in het rood.

Gegeven een familie knooppunten. Herinner dat A_n de verzameling knooppunten is van dimensie n. In een ideale wereld zal $A_{n-1} \subset A_n$. Zo'n familie noemen we *inductief.* Merk op dat in dit geval $A_n \backslash A_{n-1}$ exact één element bevat. We geven dit knooppunt de naam a_n, en bekomen een rij van getallen (knooppunten) $\{a_n\}_{n \in \mathbb{N}}$, met de eigenschap dat A_n precies de eerste n elementen van deze rij bevat. Noem deze de *inducerende rij.* Denken over een inductieve familie, is dus hetzelfde als denken over een rij knooppunten.

We zullen, op twee verschillende manieren, de familie $n$de eenheidswortels omvormen tot een inductieve familie. Onze eerste poging is naïef van aard en de gemaakte keuzes zullen leiden tot hoge conditiegetallen. Vervolgens stellen we een alternatieve inductieve familie knooppunten voor, waarbij het conditiegetal wel groeit, maar heel traag. Starten doen we met quasi-cyclische knooppunten. We definiëren de inductieve familie via de inducerende rij. Merk eerst op dat alle $n$de eenheidswortels ook $2n$de eenheidswortels zijn (i.e. als $x^n = 1$, dan is ook $x^{2n} = 1$). We zullen ervoor zorgen dat voor elke k, de eerste 2^k elementen van de rij, exact de 2^kde eenheidswortels zijn. Stel $a_1 = 1$. Definieer vervolgens, voor $2^k < n \leq 2^{k+1}$, het $n$de element van de inducerende rij als

$$a_n := e^{2\pi\imath(2(n-2^k)-1)/(2^{k+1})}. \tag{2}$$

Opdracht 9. Maak een figuur met de eerste acht quasi-cyclische knooppunten (i.e. $a_1 \ldots a_8$). Naast elk punt, zet je de bijhorende index.

- Waar liggen de 4de eenheidswortels? Deze zijn allen ook 8ste eenheidswortels. Waar liggen de vier overgebleven 8ste eenheidswortels ten opzichte hiervan? Hoe kan je hun volgorde makkelijk visueel beschrijven?

Maak vervolgens een andere figuur die de conditiegetallen bevat van de geassocieerde Vandermonde-matrices voor $n = 1 \ldots 64$. Je kan je oplossing verifiëren als volgt: de conditie is minimaal in machten van twee, en bereikt een lokaal maximum middenin opeenvolgende machten van twee.

Uit de figuur blijkt dat tussen opeenvolgende machten van twee, de conditie zeer groot wordt. We kunnen echter beter doen aan de hand van de *Van der Corput knooppunten.* We definiëren deze opnieuw aan de hand van de inducerende rij. Schrijf een geheel getal $n \geq 0$ in binaire vorm

$$n = \sum_{j=0}^{\infty} n_j 2^j, \text{ met } n_j \in \{0, 1\}.$$

Vervolgens definiëren we

$$\tilde{n} = \sum_{j=0}^{\infty} n_j 2^{-j-1}.$$

De rij wordt dan gedefinieerd door

$$a_1 = 1 \text{ en } a_{n+1} := e^{2\pi\imath\tilde{n}}.$$

Deze volgorde zorgt er voor dat de eenheidswortels op een meer uniforme manier worden toegevoegd.

Opdracht 10. Plot opnieuw de conditiegetallen voor de geassocieerde Vandermonde-matrices voor $n = 1 \ldots 64$.

Opdracht 11. (**extra**[8]) Geef een bewijs dat de Van der Corput knooppunten van dimensie 2^k gewoon de 2^kde eenheidswortels zijn.

Opmerking: beschouw een rij van complexe getallen $\{a_n\}_{n=1}^{\infty}$ zodat $|a_n| = 1$, en zodat voor elke $k > 0$ de eerste 2^k elementen uit de rij exact de 2^kde eenheidswortels zijn. De Van der Corput rij is van deze vorm. Definieer V_n de Vandermonde-matrix op de eerste n elementen van deze rij. We trachten de conditie van de Vandermonde-matrices zo klein mogelijk te houden. Dit betekent dat we een rij zoeken zodat voor elke $k > 0$

$$\max_{2^k < n < 2^{k+1}} \mathrm{cond}_2(V_n)$$

minimaal is. De Van der Corput rij tracht dit probleem op te lossen. Het is echter niet geweten of deze ook optimaal is.

Spline-interpolatie

In dit tweede deel van het practicum[9], zullen we een lissajousfiguur benaderen aan de hand van kubische b-splines. Een lissajousfiguur wordt gegeven door een parametrisatie $\gamma : [0, 2\pi] \to \mathbb{R}^2$:

$$\theta \mapsto \begin{pmatrix} \sin(a \cdot \theta + \phi) \\ \sin(b \cdot \theta) \end{pmatrix}. \tag{3}$$

Opdracht 12. Maak een functie die gegeven a, b, ϕ en een geheel getal k, de bovenstaande parametrisatie evalueert in k equidistante punten (equidistant in $[0, 2\pi]$). De functieheader ziet eruit als

```
function [gammax,gammay] = lissajous(a, b, phi, k)
% a, b en phi definiëren de parametrisatie van de lissajousfiguur
% gammax en gammay zijn vectoren van lengte k, die de x en y waarden geven voor theta
% in linspace(0,2pi,k)
```

Experimenteer voor enkele waarden van a, b en ϕ. Neem vervolgens de figuren voor

- $a = 2$, $b = 3$, $\phi = \pi/4$; en
- $a = 3/2$, $b = 5/2$, $\phi = \pi/2$;

op in je verslag.

Vervolgens gaan we over tot kubische b-splines. Eerst implementeren we de nodige functies om het interpolatieprobleem op te lossen. Je mag de volgende Matlab-functies gebruiken:

- `pp = bspline(v)` geeft de b-spline terug gedefinieerd op de knooppunten in vector `v`. Merk op dat `pp` hier teruggegeven wordt als "structure"[10].
- `ppval(pp,t)` evalueert de de spline `pp` (gegeven als structure) in het punt `t`. Of indien `t` een vector is, in alle punten van `t`.
- `fnder(pp,n)` leidt de spline `pp` (opnieuw een structure) `n` keer af.

Opdracht 13. Indien `bspline` wordt opgeroepen zonder toekenning, dan geeft deze een plot terug van de b-spline op de gegeven knooppunten. Geef `bspline([1,2,3,4,5])` in, in Matlab. Van welke orde is de gegeven spline? Er staan nog andere functies/krommen op deze plot. Leg uit wat ze voorstellen. Neem de figuur op in je verslag.

[8] Dit is een (iets uitdagendere) extra opgave. Als je deze niet maakt, kan je nog steeds het maximum van de punten halen. De maximale score is steeds 3.

[9] Notatie gebaseerd op `https://p.cygnus.cc.kuleuven.be/bbcswebdav/pid-27211404-dt-content-rid-278024442_2/courses/B-KUL-G0N90a-2021/splines.pdf`

[10] Je hoeft niet te begrijpen wat deze precies doen om de oefening op te lossen, maar even de help-pagina hierover bekijken kan nooit kwaad.

Opdracht 14. Vervolgens implementeren we twee functies. De eerste stelt het op te lossen stelsel op (meer precies geeft deze de matrix[11] van het stelsel terug). Gebruik volgende header.

```
  function A = cubicsplinematrix(t, tbefore, tafter, periodic)
% t is een vector met knooppunten t_0 ... t_n. Deze zijn geordend en allen
% verschillend
% tbefore is een (geordende) vector met extra knooppunten die we vooraan toevoegen
% tafter is een (geordende) vector met extra knooppunten die we achteraan toevoegen
% periodic is een logische constante. Een periodic = 1 komt overeen met periodische
% voorwaarden. Een periodic = 0 komt overeen met natuurlijke voorwaarden.
% A is de matrix die de interpolatievoorwaarden beschrijft.
```

Een noot over `periodic`. In de les zagen we dat we voor elk knooppunt in `t` een interpolatievoorwaarde krijgen. Dit volstaat echter niet om de interpolerende spline uniek te bepalen. Hierom voegen we extra (rand)voorwaarden toe. In de les zagen we twee mogelijkheden, periodiek en natuurlijk. Vooraleer je begint te implementeren, geef een antwoord op volgende vragen: als `t` een vector is met $n+1$ knooppunten, wat zijn dan de dimensies van A? Hoeveel knooppunten bevatten `tbefore` en `tafter`?

Nu we het stelsel kunnen opstellen implementeren we een functie die het interpolatieprobleem oplost.

```
  function [c,y] = cubicsplinesolve(t, tbefore, tafter, periodic, f, k)
% t is een vector met knooppunten t_0 ... t_n. Deze zijn geordend en allen
% verschillend
% tbefore is een (geordende) vector met extra knooppunten die we vooraan toevoegen
% tafter is een (geordende) vector met extra knooppunten die we achteraan toevoegen
% periodic is een logische constante. Een periodic = 1 komt overeen met periodische
% voorwaarden. Een periodic = 0 komt overeen met natuurlijke voorwaarden.
% f is een vector met de interpolatiewaarden geassocieerd aan de knooppunten in t
% c is de oplossing van het stelsel Ac = f. Het geeft de coëfficiënten van de
% interpolerende spline s ten opzichte van de b-spline-basis
% y is een vector die de gevonden interpolerende spline s evalueert in k
% equidistante punten tussen t_0 en t_n.
```

Voor het oplossen van het lineaire stelsel, mogen jullie uiteraard het backslash commando in Matlab gebruiken.

Nu over naar de toepassing.

Opdracht 15. In een script `cubespline.m` doe het volgende. Interpoleer de twee lissajousfiguren die we in het begin van deze sectie hebben beschouwd. Gebruik 20 equidistante interpolatiepunten. Kies gepaste waarden voor `tbefore` en `tafter` (de enige voorwaarde zijnde dat deze respectievelijk voor en achter de knooppunten in `t` liggen). Neem k gelijk aan 200. Kies de gepaste randvoorwaarden. Maak telkens een figuur waarin de lissajousfiguur gegeven wordt met een volle lijn, en de spline die haar benadert met een streepjeslijn. Vergeet niet, je interpoleert hier een parametrisatie van een kromme in het vlak. je moet dus telkens twee functies interpoleren, namelijk de x-coördinaten $\gamma_1(\theta)$ en y-coördinaten $\gamma_2(\theta)$.

Als laatste gaan we na hoe de interpolatiefout zich gedraagd ten opzichte van het aantal knooppunten. We definiëren eerst een functienorm voor een functie $g : [0, 2\pi] \to \mathbb{R}^2$ als

$$\|g\|_\infty = \sup_{\theta \in [0,2\pi]} \|g(\theta)\|_2.$$

[11] Zie pagina 9 in de nota's over splines.

In praktijk berekenen we deze door een maximum te nemen over een groot aantal punten in het interval $[0, 2\pi]$. I.e. voor een gepaste keuze van $m > 0$ wordt deze norm in praktijk

$$\|g\|_\infty = \max_{\theta \in V} \|g(\theta)\|_2,$$

met $V = \{2k\pi/m \mid 0 \leq k \leq m\}$. Je mag voor de volgende opdracht m gelijk nemen aan 2^{13}.

Opdracht 16. Vervolgens, enkel voor $a = 2$, $b = 3$, $\phi = \pi/4$ in 3 lossen we opnieuw het interpolatieprobleem op, maar nu met 2^r interpolatiepunten ($r = 1 \dots 10$). Als $s_x(\theta)$ de spline is voor de x-waarden, en $s_y(\theta)$ de spline voor de y-waarden, dan zal $s(\theta) := (s_x(\theta), s_y(\theta))^\top$. Beschouw de interpolatiefout $\|s - \gamma\|_\infty$. Maak een figuur met op de x-as het aantal knooppunten, en op de y-as de interpolatiefout. Beantwoord volgende vragen:

- Als de afstand tussen opeenvolgende knooppunten halveert, hoeveel kleiner wordt de fout dan?
- Wat zegt dit over de orde van de methode?
- Komt dit overeen met wat je verwacht uit de theorie? Citeer de relevante stelling en leg kort uit waarom deze van toepassing is.

Praktisch

Groepen

Je werkt in groepjes van twee personen. Ten laatste op **maandag 11 april** vermeld je per groepje de twee namen en de twee studentennummers van de leden van het groepje op het forum[12] '*Groepjes Practicum*' op Toledo. Er is een forum '*Practicum: vind groepslid*' aangemaakt. Deze kan helpen om een partner te vinden.

Tijdsbesteding

Dit practicum heeft een belasting van ongeveer 16 uur per student. Beschouw dit getal als een richtlijn voor je tijdsbesteding. Het is niet de bedoeling om hier ver over te gaan, maar we willen er wel op wijzen dat deze richtlijn ervan uitgaat dat je bij bent met de oefenzittingen (vooral die in Matlab) en de nodige leerstof. **Begin zeker tijdig aan het practicum!** Tip: de onderdelen "Knooppunten in $\mathbb{R}$", "Knooppunten in $\mathbb{C}$" en een deel van de "Introductie", kan je reeds maken (en schrijven) wanneer het onderdeel stelsels in de les wordt afgerond.

Begeleide oefenzitting

Volgens de voorlopige planning zal je rond oefenzitting 10 (afhankelijk van je groep) aan het practicum kunnen werken en kan je ook vragen stellen aan de assistent. Deze oefenzitting hoort niet bij de belasting. Je mag je vragen ook sturen naar **het e-mailadres tom.kaiser@kuleuven.be met als onderwerp "Vragen practicum"** zodat ik deze gemakkelijk kan onderscheiden van andere mails. Gelieve zo duidelijk mogelijk je vraag te stellen (evt. met Matlab-script) en probeer je vragen wat te groeperen. Voor vragen die de laatste dagen voor de deadline nog gesteld worden, kan ik niet garanderen dat deze nog beantwoord worden.

Evaluatie

Het practicum telt mee voor 3 van de 20 punten van het eindexamen. De evaluatie van het practicum is gebaseerd op de ingediende code en het verslag.

[12] Zie Toledo > Numerieke Wiskunde[G0N90a] > Communicatie > Discussieruimte > Groepjes practicum

Code

De code moet geupload worden op Toledo door iemand van het groepje ten laatste op **vrijdag 6 mei**. Zet al de code in één zip-bestand met jullie achternamen in de bestandsnaam, bv. `codeDeSmetVanIeper.zip`. Het zip-bestand moet minstens de volgende bestanden bevatten:

- `Vandermonde.m`
- `CondVandReal.m`
- `CondVandImag.m`
- `lissajous.m`
- `cubicsplinematrix.m`
- `cubicsplinesolve.m`
- `cubespline.m`

Vergeet mogelijke hulpfuncties niet toe te voegen die jullie geschreven hebben. Dit betekent dat jullie code moet werken met de versie die op Toledo staat.

Verslag

Schrijf een duidelijk gestructureerd en bondig verslag van maximaal 10 pagina's waarin zeker het volgende staat:

- Alle figuren en tabellen. Gebruik doordachte figuren en tabellen om je bevindingen te verduidelijken. Gebruik logaritmische grafieken waar nodig. Kies de schaal van je figuren oordeelkundig, vooral als twee verschillende figuren vergeleken moeten worden. Voeg de gepaste legendes toe, en benoem je assen. Je hoeft geen uitgebreid onderschrift te voorzien, als je in de tekst duidelijk naar de figuur verwijst.
- Antwoorden op alle opgaves en vragen die gesteld worden.
- Indien je dit nodig acht, kan je in het verslag bepaalde ontwerpkeuzes voor je algoritmes verduidelijken.
- Tijdsbesteding aan de verschillende onderdelen van het practicum:
  - Schrijven van de functies;
  - Schrijven van de scripts en het maken van de figuren;
  - Debuggen;
  - Schrijven van het verslag.

  Indien jullie het werk verdeeld hebben en de tijdsschattingen voor beide groepsleden verschillen, geef dit dan ook duidelijk aan.

Je uploadt je verslag op Toledo als een pdf-bestand ten laatste op **maandag 9 mei**. Geef het pdf-bestand een analoge naam als je code, bv. `verslagDeSmetVanIeper.pdf`. Zich niet houden aan bovenstaande instructies zal leiden tot penalisatie in de quotering.

Overzicht deadlines

Wat	Waar	Deadline
Verdelen in groepjes	Forum	maandag 11 april
Uploaden code	Toledo	vrijdag 6 mei
Uploaden verslag	Toledo	maandag 9 mei

Veel succes!
Tom en Marc

Appendix

Aangezien jullie nog niet vaak verslagen hebben geschreven, enkele algemene tips in willekeurige volgorde:

- Antwoord gericht op de vraag. Niet relevante "bla-bla" lezen we toch niet.
- Aangezien dit practicum een opeenvolging is van opdrachten, mag je verslag een opeenvolging zijn van antwoorden. Je hoeft geen roman te schrijven. Zorg gewoon dat het er netjes uitziet.
- Gebruik bij voorkeur korte zinnen (i.e. geen 5 geneste bijzinnen). Dit verhoogt de leesbaarheid.
- Indien je resultaten gebruikt die niet standaard zijn, dan refereer je daar naar, of bewijs je die.
- Foto's van tekeningen[13] mogen worden toegevoegd, maar zorg dat deze verzorgd zijn.
- Voorblad en inhoudstafel zijn overbodig. Je schrijft een verslag, geen boek.
- Geen foto's van berekeningen!
- Gebruik bij voorkeur Latex, maar Word is toegelaten (hier wordt geen onderscheid gemaakt bij de quotering). Echter, je zal tijdens je opleiding niet onder het gebruik van Latex uitkomen, so you can as well start early.
- Zorg dat de notatie duidelijk is, gebruik deze consistent.
- Structureer je code. Bv. hetgeen binnen een for-loop staat geef je een indent. Gebruik voldoende spaties om de leesbaarheid te vergroten.
- Geef de constanten en variabelen overzichtelijke namen.

[13] Waarschijnlijk niet van toepassing in dit practicum