Een onderzoek aan de Boundary Element Methode voor nauwkeurige localisatie van hersenactiviteit

Raymond van Gent
Juni 1998

Technische Universiteit Delft
Faculteit der Informatietechnologie en Systemen
Subfaculteit Technische Wiskunde

MEG centrum KNAW
Academisch Ziekenhuis Vrije Universiteit
Polykliniek receptie C
De Boelelaan 1117
1081 HV Amsterdam

Afstudeercommissie:
prof. dr. ir. P. Wesseling (Technische Universiteit Delft)
dr. B. W. van Dijk (MEG centrum KNAW)
dr. J. C. de Munck (MEG centrum KNAW)
ir. A. Segal (Technische Universiteit Delft)
dr. ir. C. Vuik (Technische Universiteit Delft)

Voorwoord

Via deze weg wil ik het MEG centrum KNAW bedanken voor de mogelijkheid, die zij hebben geboden om dit afstudeerwerk te realiseren. In het bijzonder gaat mijn dank uit naar dr. J.C. de Munck voor de begeleiding op het stageadres.
Verder gaat mijn dank uit naar ir. A. Segal en dr.ir. C. Vuik voor het begeleiden van mijn afstudeerstage vanuit de universiteit en voor het bestuderen en corrigeren van de verschillende versies van dit verslag
Tenslotte gaat mijn dank uit naar de leden van mijn afstudeercommissie, prof.dr.ir.P. Wesseling, dr.B.W. van Dijk, dr.J.C. de Munck, ir.A. Segal en dr.ir.C. Vuik.

Raymond van Gent, juni 1998

Inhoudsopgave

Hoofdstuk 1

Introductie

Deze afstudeeropdracht is uitgevoerd bij het MEG centrum KNAW. Het MEG centrum KNAW is een instituut van de Koninklijke Nederlandse Akademie van Wetenschappen, gehuisvest in de polikliniek van het Academisch Ziekenhuis van de Vrije Universiteit in Amsterdam.

Het MEG centrum richt zich op toepassing en verbetering van de techniek van het Magneto-encefalogram en dat betekent letterlijk "magnetisch hersenschrift". MEG is een nieuwe methode waarmee de werking van de hersenen kan worden bestudeerd. Dit gebeurt met behulp van magnetische signalen die door de hersencellen worden gemaakt als ze actief zijn. Als de hersencellen actief zijn ontstaan er potentiaalverschillen op de hoofdhuid en een magnetisch signaal buiten het hoofd. De meting van het magnetisch signaal wordt MEG genoemd.

Meerkanaals MEG/EEG (EEG staat voor Electro-encefalogram) signalen kunnen worden gebruikt om de onderliggende activiteit te lokaliseren (in ons geval dus de hersenactiviteit). Hiertoe wordt een mathematisch model opgesteld dat het verband geeft tussen electrische bronnen en de gegeven signalen. Wanneer de geometrie van de geleider bekend is en wanneer de bron door een beperkt aantal parameters beschreven wordt (bv. een stroomdipool) dan is het bronlocatie probleem teruggebracht tot een parameterschattingsprobleem: bepaal die set bronparameters die de metingen zo goed mogelijk beschrijven. De nauwkeurigheid waarmee bronnen in de hersenen gelocaliseerd kunnen worden wordt in belangrijke mate bepaald door de nauwkeurigheid van het gebruikte geleidingsmodel. Het eenvoudigste model van het hoofd bestaat uit een set van concentrische bolschillen, met ieder hun eigen geleidbaarheid. Deze schillen stellen dan voor: de hersenen, de cerebrospinale vloeistoflaag, de schedel en de huid. Het bolschillenmodel is analytisch oplosbaar en geeft een snel algoritme, maar een beperkte nauwkeurigheid.

Met de zogenaamde Boundary Element Methode (BEM) kan een realistisch gevormd hoofd worden gemodelleerd dat bestaat uit homogene compartimenten van constante geleidbaarheid. Hiertoe wordt een randintegraalvergelijking opgesteld voor de potentiaal op de scheidingsoppervlakken. Deze vergelijking kan worden opgelost door een driehoeksbelegging te maken op de scheidingsoppervlakken. Op elk van de driehoekjes wordt de potentiaal constant genomen, of de potentiaal kan een lineair of kwadratisch verloop hebben. Hiermee wordt de integraalvergelijking gediscretiseerd en krijgen we een systeem van lineaire vergelijkingen dat numeriek oplosbaar is. Bij het localiseren van electrische bronnen uit MEG/EEG signalen wordt voor een groot aantal combinaties van bronnen de potentiaal/het magnetisch veld uitgerekend. Daarom is het toepassen van realistische modellen een afweging van snelheid en nauwkeurigheid. Hoe meer driehoekjes en groter de orde van de interpolatie over de driehoekjes, hoe nauwkeuriger het model, maar hoe langzamer het algoritme. Met behulp van simulaties willen we er achter komen waar het optimum in de praktijk ligt.

Mijn afstudeeropdracht was voor het oplossen van het probleem een snel en een zo nauwkeurig mogelijk model te krijgen. Er zijn dan enkele vragen die nader onderzocht moeten worden. Hoe fijn moet de driehoeksbelegging worden gekozen willen we een bepaalde nauwkeurigheid verkrijgen? Is het mogelijk cq wenselijk om een locale verfijning toe te passen? Heeft het zin om hogere orde interpolaties op de driehoeken te gebruiken of neemt dan alleen de grootte van het systeem van vergelijkingen toe? Kan met een grovere driehoeksbelegging worden volstaan als we in plaats van platte driehoekjes gekromde driehoekjes nemen en wegen de extra kosten hier tegenop?

Hoofdstuk 2

Probleemstelling

2.1 De fysische achtergronden

2.1.1 De Methode

Meerkanaals MEG/EEG signalen kunnen worden gebruikt om de onderliggende hersenactiviteit te lokaliseren. Dat gebeurt bij de MEG met een apparaat met meetpunten, die de magnetisch signalen meten, op een deel van een bolschil die over het hoofd heen wordt geschoven. Bij het maken van een EEG daarintegen moeten de meetpunten op het hoofd aangebracht worden die de potentiaal aldaar meten.

Het maken van een MEG/EEG wordt onder andere gedaan bij epilepsie patienten en mensen met de ziekte van Alzheimer. Om deze mensen in de toekomst te kunnen genezen is het van belang om de werking van de hersenen beter te bestuderen. Dit wordt gedaan door de patienten een opdracht te geven waarbij men de hersenen moet gebruiken. Hierdoor ontstaan er signalen die gemeten worden met de MEG/EEG.
Het is van groot belang dat de metingen een zo nauwkeurig mogelijke beschrijving van de locatie van de signalen leveren.

2.1.2 Fysische probleemstelling

Bij het genereren van een MEG wordt het magnetisch veld bepaald dat afgegeven wordt door de aanwezigheid van hersenactiviteit. Met het gemeten magnetische veld hopen we de locatie en sterkte van de bronnen te kunnen bepalen, die voor deze activiteit zorgen.

Figuur 2.1: MEG systeem

De vergelijkingen die dit fysisch probleem beschrijven zijn de vergelijkingen van *Maxwell* die gereduceerd zijn tot:

$$\mathrm{rot}(\mu_0^{-1}\mathbf{B}) = \sigma\mathbf{E} + \mathbf{J_i}, \tag{2.1}$$

$$\mathrm{rot}\mathbf{E} = \mathbf{0}. \tag{2.2}$$

Hier is $\mathbf{B}$ de magnetische inductie met S.I. eenheid $Nsm^{-1}C^{-1}$, $\mathbf{E}$ de electrische veldsterkte in NC^{-1}, $\mathbf{J_i}$ de stroomdichtheid van de i^e bron, σ de conductiviteit in $C^2N^{-1}m^{-2}s^{-1}$ en μ_0 de magnetische permeabiliteit van vacuüm in Ns^2C^{-2}.

Uit de laatste vergelijking (2.2) volgt dat er een functie moet zijn waarvoor de rotatie nul is. Dit geldt als deze functie een gradient van een scalair is. We kiezen daarom

$$\mathbf{E} = -\nabla\psi. \tag{2.3}$$

ψ wordt de potentiaal genoemd met als dimensie $NC^{-1}m^{-1} = Volt$ en dit wordt in de fysica ook wel de spanning genoemd.
Verder geldt dat de totale stroomdichtheid $\mathbf{J}$ bepaald wordt door de volume stroom, die ontstaat doordat er electromagnetische krachten op de vrije ladingen werken, en door de stroom afkomstig van de bronnen en geldt dat:

$$\mathbf{J} = \sigma\mathbf{E} + \mathbf{J_i}. \tag{2.4}$$

met $\mathbf{J}$ in $Cm^{-2}s^{-1}$.
Voor het oplossen van $\mathbf{B}$ gebruiken we de wet van *Biot-Savart* (zie [3], pagina 83-86 en [9]) en krijgen dan:

$$\mathbf{B}(\mathbf{x}) = \frac{\mu_0}{4\pi}\int_V \mathbf{J}(\mathbf{x_0}) \times \frac{\mathbf{x}-\mathbf{x_0}}{|\mathbf{x}-\mathbf{x_0}|^3} dV_0 \tag{2.5}$$

met $\mathbf{x_0}$ als integratievariabele en $\mathbf{x}$ het punt waar we $\mathbf{B}$ willen weten.
Om dit zo nauwkeurig mogelijk te kunnen uitvoeren is het van belang dat we een nauwkeurige weergave hebben van het oppervlak waar de activiteiten plaats vinden. Merk op dat $\mathbf{J}$ en dus $\mathbf{B}$ afhangt van $\sigma\nabla\psi$.

Randvoorwaarden

Om een stelsel vergelijkingen eenduidig op te lossen zijn randvoorwaarden nodig. Bovendien hebben we in ons probleem te maken met verschillende schillen met ieder zijn eigen geleiding en daarom zijn er ook aansluitvoorwaarden nodig.
We bekijken een geleidend medium met conductiviteit $\sigma \neq 0$. Als er een electrisch veld ontstaat, treedt er verplaatsing van vrije lading op.
Op de grens tussen twee lagen hebben we behoud van lading zodat er evenveel lading een volume in stroomt als er uit stroomt . Deze eigenschappen, samen met het feit dat het medium *isotroop* en *homogeen* is, leveren de benodigde voorwaarden aan de scheidingsoppervlakken.
De eerste scheidingsvoorwaarde volgt uit het feit dat $\sigma\mathbf{E}_n$ continu is en dus dat

$$\sigma\frac{\partial\psi}{\partial n} \quad \text{continu}$$

is.
Verder geldt op het scheidingsoppervlak dat de potentiaal daar continu verloopt.

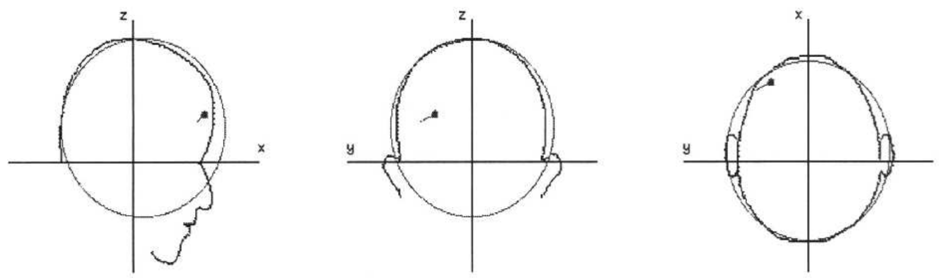
De randvoorwaarde aan de buitenste rand is afgeleid uit de eerste scheidingsvoorwaarde en het feit dat buiten het volume geen electrisch veld is. We krijgen dan:

$$\sigma \frac{\partial \psi}{\partial n} = 0.$$

(Voor meer informatie zie [10] en [4].)

2.1.3 Modellering van het hoofd

De nauwkeurigheid waarmee bronnen in de hersenen gelocaliseerd kunnen worden, wordt in belangrijke mate bepaald door de geometrie van het geleidingsmodel (het hoofd). Het hoofd bestaat uit verschillende compartimenten en het eenvoudigste model daarvan bestaat uit een set concentrische bolschillen, met ieder zijn eigen geleidbaarheid. Deze schillen stellen dan voor: de hersenen, de celobrospinale vloeistoflaag, de schedel en de hoofdhuid.
Het bolschillen model is analytisch oplosbaar (zie paragraaf 3.1), maar geeft een beperkte nauwkeurigheid, omdat het menselijk hoofd nu eenmaal niet bolvormig is. Het bolschillen model is dus niet erg realistisch.



Figuur 2.2: Bolschillen vs. Realistisch

We moeten dus een betere geometrische beschrijving van het hoofd maken om een oplossing te krijgen die nauwkeuriger en dus realistischer is. Met dit realistische model kunnen we dan op een nauwkeurige manier de locatie van de bron bepalen uit de metingen, maar is het mathematische model niet meer analytisch oplosbaar.

2.1.4 Het inverse probleem

Wat we in ons probleem uiteindelijk willen weten is de locatie van de bronnen. Dit probleem wordt ook wel een inverse probleem genoemd. In ons geval komt dat doordat we de potentiaal meten en dus, net als de geometrie, als bekend beschouwen echter de bron, welke een parameter in de differentiaalvergelijking is, is onbekend. Het inverse probleem is over het algemeen moeilijker oplosbaar dan het voorwaartse probleem. Een ander probleem dat bij het inverse probleem ontstaat is dat de oplossing niet eenduidig hoeft te zijn. Kijken we naar het magnetisch inverse probleem, dan blijkt dat dit geen unieke oplossing heeft. Dit komt door het feit dat er zogenaamde magnetisch stille bronnen zijn, die geen magnetisch veld buiten de geleider V produceren. Zo'n stroombron kan altijd aan de oplossing worden toegevoegd zonder het veld buiten V te beinvloeden (zie [9]).

Wat we wel willen weten zijn de locatie, oriëtatie en de sterkte van de onderliggende bronnen. Deze bepalen de grootte van de potentiaal. Om de waarde van die bronnen nu te bepalen moeten we die waarde vinden waarvoor geldt dat de bijbehorende potentiaal het minst van de meetwaarde afwijkt.
Bij zowel het voorwaartse probleem, waar we de potentiaal willen weten, als bij het inverse probleem is de geometrie van het oppervlak heel belangrijk. Is het oppervlak niet realistisch genoeg dan kan het voorkomen dat we de bronnen op een hele andere plaats berekenen dan waar ze in werkelijkheid zitten. Het is daarom dan ook belangrijk dat aan een bepaalde nauwkeurigheid, van de geometrische beschrijving, wordt voldaan.
Daar de belegging van de geometrie waarschijnlijk invloed heeft op de locatie van de bronnen, willen we onderzoeken hoe we een zo nauwkeurig mogelijke discretisatie van de geometrie van het hoofd krijgen. Daartoe moeten we alleen eerst naar het voorwaartse probleem kijken en de waarde van de bronnen als bekend beschouwen. Als we dan een goede oplossing voor de, gegeven, bronnen gevonden hebben, kunnen we weer terug naar het inverse probleem (voor het vinden van deze oplossing gebruiken we een algoritme dat al aanwezig is). Het inverse probleem laten we voorlopig ook even voor wat het is en in het vervolg zullen we het voorwaartse probleem beschouwen.

2.2 Mathematische probleemstelling

2.2.1 Eén compartiment

Beschouwen we eerst het probleem waarbij we te maken hebben met één compartiment dan kan het fysisch probleem omgeschreven worden in het volgende mathematische probleem:

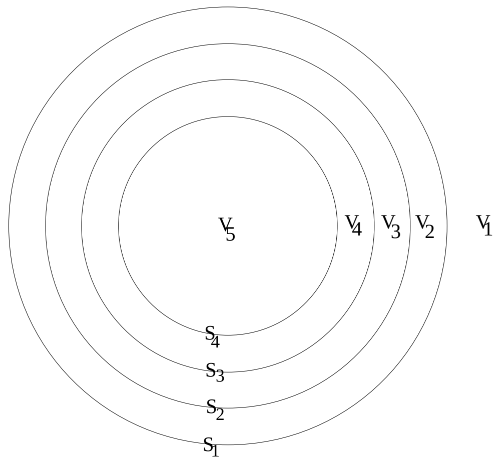
$$\mathrm{div}(\sigma \nabla \psi) = v \qquad \mathbf{x} \in V \tag{2.6}$$

$$(\sigma \nabla \psi) \cdot n \quad = \quad 0 \qquad\qquad \mathbf{x} \in S,$$

met S het oppervlak van het compartiment en V het volume ervan.

2.2.2 Meerdere compartimenten

Willen we naar een meer realistisch model kijken dan hebben we te maken met meerdere compartimenten en hebben we de aansluitvoorwaarden nodig voor het oplossen van ons model. Stel dat we kijken naar de geneste volumes V_i (i=1..5) die gescheiden zijn door de oppervlakken S_i (i=1..4). V_1 is het volume buiten het hoofd is, V_5 het volume van de hersenen.



Figuur 2.3: Meerdere Compartimenten, genest

Noem de potentiaal in volume V_i, ψ_i. Het model gaat dan over in:

$$\begin{aligned}
\operatorname{div}(\sigma_i \nabla \psi_i) &= v_i && \mathbf{x} \in V_i && i = 1..5 \qquad (2.7)\\
\psi_i &= \psi_{i+1} && \mathbf{x} \in S_i && i = 2..4\\
(\sigma_i \nabla \psi_i) \cdot n &= -\sigma_{i+1} \nabla \psi_{i+1} \cdot n && \mathbf{x} \in S_i && i = 2..4\\
(\sigma_1 \nabla \psi_1) \cdot n &= 0 && \mathbf{x} \in S_1
\end{aligned}$$

Met S_i het scheidingvlak tussen volume V_i en volume V_{i+1} en n de naar buiten gerichte normaal.

2.2.3 Bronterm v

Om het model op te lossen moeten we ook weten wat v is. In ons geval beschouwen we de hersenen. Het wordt over het algemeen geaccepteerd dat de bronstroom in de hersenen zich als een dipool gedraagt (zie [4]).
Als de bron wordt beschreven met een stroomdipool met moment $\mathbf{M}$ en op de plaats $\mathbf{x_{dip}}$ zit, wordt de opgewekte stroom

$$\mathbf{J_i}(\mathbf{x}) = \mathbf{M}\delta(\mathbf{x_{dip}} - \mathbf{x}) = \mathbf{M}\delta(x_{dip} - x)\delta(y_{dip} - y)\delta(z_{dip} - z) \quad (2.8)$$

(met δ de *Dirac*-delta) en daardoor hebben we

$$v = \mathrm{div}\mathbf{J_i} = \mathbf{M} \cdot \nabla\delta(\mathbf{x_{dip}} - \mathbf{x}) = -\mathbf{M} \cdot \nabla_{dip}\delta(\mathbf{x_{dip}} - \mathbf{x}). \quad (2.9)$$

Hierbij is ∇_{dip} de gradient met betrekking tot het bronpunt. Dit doen we, omdat we hiermee de potentiaal verdeling, als gevolg van een dipool, kunnen bepalen.

Hoofdstuk 3

Oplossingen van het probleem

In dit hoofdstuk zullen we het probleem beschreven in hoofdstuk 2 oplossen. Dit kan in het geval we naar het bolschillenmodel kijken zowel analytisch als met behulp van de Boundary Element Methode (BEM). Hebben we echter te maken met een set willekeurig geneste oppervlakken dan kunnen we hoogst waarschijnlijk alleen de BEM toepassen.
In dit hoofdstuk zullen we daarom zowel de analytische oplossing voor het bolschillen model als de BEM bespreken. Eveneens zal het model in een integraalvergelijking worden geschreven die nodig is voor het toepassen van de BEM.

3.1 Analytische oplossing voor het bolschillenmodel

Omdat we een bol beschouwen gaan we het model (2.6) omschrijven naar bolcoördinaten. Voor een dipool met sterkte $\mathbf{M}$ op de plaats $(r_0, \theta_0, \varphi_0)$ wordt de potentiaal gegeven door de volgende uitdrukking:

$$\psi_{dipool}(r,\theta,\varphi) = \sum_{n=1}^{\infty}\sum_{m=0}^{n}\sum_{\alpha=0}^{1} \frac{Y_{nm\alpha}(\theta,\varphi)}{\|Y_{nm\alpha}\|^2}\mathbf{M}\cdot\nabla_0 R_n(r_0,r) Y_{nm\alpha}(\theta_0,\varphi_0). \quad (3.1)$$

Met:

$$\begin{aligned} Y_{nm\alpha} &= P_{nm}(\cos\theta) X_m(\varphi), && (3.2)\\ X_m(\varphi) &= \cos m\varphi \qquad \alpha = 0, \\ &= \sin m\varphi \qquad \alpha = 1. \end{aligned}$$

Hier is $P_{nm}(\cos\theta)$ de geassocieerde *Legendre* functie (zie Bijlage A). Verder zijn $Y_{nm\alpha}$ niet-genormaliseerde bolfuncties van de n^{de} orde met graad m, de functie is even als $\alpha = 0$ en oneven als $\alpha = 1$. De bolfuncties zijn de eigenfuncties van het θ en φ gedeelte van de *Laplace* operator. (Voor meer hierover zie [10] en [4].)

3.2 Integraalvergelijking

Voor de afleiding van de integraalvergelijking zullen we gebruik maken van de tweede Greense identiteit:

$$\int_V (\psi\nabla^2\phi - \phi\nabla^2\psi)dV = \int_S (\psi\nabla\phi - \phi\nabla\psi)\cdot d\mathbf{S}. \tag{3.3}$$

In deze vergelijking zullen we gebruik maken van de gegevens uit het model en het veld van een puntbron (zie [10]).

3.2.1 Eén compartiment

Vullen we de gegevens uit het mathematische model (2.6) in vergelijking (3.3) in, dan resulteert dat, voor punten op de rand, in een integraalvergelijking van de onderstaande vorm:

$$\Omega(\mathbf{x})\psi(\mathbf{x}) = \psi_m(\mathbf{x}) + \int_S K(\mathbf{x},\mathbf{x_0})\psi(\mathbf{x_0})\cdot d\mathbf{S} \tag{3.4}$$

met

$$K(\mathbf{x},\mathbf{x_0}) = -\nabla\frac{1}{R} = -\nabla\frac{1}{|\mathbf{x}-\mathbf{x_0}|} \tag{3.5}$$

en

$$\psi_m(\mathbf{x}) = -\Big[\int_V \frac{1}{R}\nabla^2\psi dV - \int_S \frac{1}{R}\nabla\psi\cdot d\mathbf{S}\Big] \tag{3.6}$$

waarbij $\nabla^2\psi$ en $\nabla\psi$ gegeven zijn door het model (2.6). Verder is $\Omega(\mathbf{x})$ de ruimtehoek in punt $\mathbf{x}$ en weten we dat σ stuksgewijs constant is, constant per compartiment, en dus wordt (3.6) met (2.6):

$$\psi_m(\mathbf{x}) = -\int_V \frac{v}{R\sigma}dV = -\frac{1}{\sigma}\int_V \frac{v}{R}dV \tag{3.7}$$

Deze vergelijking kunnen we vereenvoudigen, als we aannemen dat we te maken hebben met een dipoolbron (zie paragraaf 2.2.3) in volume V, tot:

$$\psi_m(\mathbf{x}) = \frac{1}{\sigma}\mathbf{M}\cdot\frac{(\mathbf{x}-\mathbf{x_{dip}})}{|\mathbf{x}-\mathbf{x_{dip}}|^3} \tag{3.8}$$

met $\mathbf{M}$ de sterkte van de dipool en $\mathbf{x_{dip}}$ zijn locatie in V.
We hebben nu dus de integraalvergelijking afgeleid voor het geval dat we met één compartiment te maken hebben.

3.2.2 Meerdere compartimenten

De afleiding van de integraalvergelijking voor meerdere compartimenten gaat op vergelijkbare wijze en is uitgewerkt in [10]. Het resultaat dat we dan krijgen is

$$\begin{aligned}(\Omega^+\sigma_k^+ + \Omega^-\sigma_k^-)\psi(\mathbf{x}) &= -\sum_{j=1}^{5}\sigma_j \int_{V_j} \frac{1}{R}\nabla^2\psi dV + \\ &+ \sum_{j=1}^{4}(\sigma_j^- - \sigma_j^+)\int_{S_j} \psi_j(\mathbf{x_0})K_j(\mathbf{x},\mathbf{x_0})\cdot d\mathbf{S_j}\end{aligned} \tag{3.9}$$

met Ω^+ de ruimtehoek van boven het oppervlak gezien, Ω^- de ruimtehoek van onder het oppervlak gezien en $K_j(\mathbf{x},\mathbf{x_0})$ als

$$K_j(\mathbf{x},\mathbf{x_0}) = -\frac{\partial}{\partial n}\frac{1}{|\mathbf{x}-\mathbf{x_0}|}, \qquad \mathbf{x} \in \mathbf{S_j}. \tag{3.10}$$

Hierbij is $\mathbf{n}$ de naar buiten gerichte normaal van oppervlak S_j en geldt de vergelijking voor $k = 1....4$. We hebben nu dus een stelsel integraalvergelijkingen die we moeten gaan oplossen.
In vergelijking (3.9) moet een volume integraal worden opgelost, maar omdat we uit het model (2.7) al weten dat $\sigma_j\nabla^2\psi_j = v_j$ geldt dat de volume integraal als een bekende functie. Deze functie kunnen we, afhankelijk van v_j, dan wel analytisch, dan wel numeriek bepalen. Hier hebben we aangenomen dat we met een dipoolbron te maken hebben en dan is de volume integraal analytisch oplosbaar.
Het hoeft niet zo te zijn dat de compartimenten, net zoals in figuur (2.3), bolschillen zijn. Het is zelfs zo dat de volumes volstrekt willekeurig kunnen liggen. In ons geval is het zo dat de schillen wel genest zijn, maar niet bolvormig.

3.3 Boundary Element Methode

3.3.1 Wat is de Boundary Element Methode (BEM)?

Naast de Eindige Elementen Methode(EEM), de Eindige Differentie Methode(EDM) en de Eindige Volume Methode(EVM) is de Boundary Element Methode(BEM) een veel gebruikte numerieke methode om partiële differentiaalvergelijkingen op te lossen.
De Boundary Element Methode wordt gebruikt voor het oplossen van integraalvergelijkingen. Daartoe wordt een oppervlakte integraal gediscretiseerd. Dat doen we door allereerst het oppervlak in elementen te verdelen op dezelfde wijze als gebruikelijk in de EEM. Wat we in ons geval willen weten is wat de potentiaal op de hoofdhuid is, dat wil zeggen op de rand van het hoofd. Die potentiaal is ontstaan door de aanwezigheid van een bron in de hersenen (de hersenactiviteit). Er is dus een situatie ontstaan waarbij we door de aanwezigheid van een bron in een volume, het resultaat op de rand, van dat volume, willen weten. We lossen dit probleem op met de Boundary Element Methode.

3.3.2 Randintegraal vergelijking

Als we de BEM willen gebruiken hebben we allereerst een integraalvergelijking nodig over het oppervlak. Deze hebben we al afgeleid in paragraaf 3.2.2. Deze integraalvergelijking was:

$$\begin{aligned}(\Omega^+\sigma_k^+ + \Omega^-\sigma_k^-)\psi(\mathbf{x}) &= -\sum_{j=1}^{5}\sigma_j\int_{V_j}\frac{1}{R}\nabla^2\psi dV + \\ &\quad + \sum_{j=1}^{4}(\sigma_j^- - \sigma_j^+)\int_{S_j}\psi_j(\mathbf{x_0})K_j(\mathbf{x},\mathbf{x_0})\cdot d\mathbf{S_j}\end{aligned} \tag{3.11}$$

met

$$K_j(\mathbf{x},\mathbf{x_0}) = -\frac{\partial}{\partial n}\frac{1}{|\mathbf{x}-\mathbf{x_0}|}, \qquad \mathbf{x}\in\mathbf{S_j} \tag{3.12}$$

voor $k = 1..4$.

3.3.3 Randelement vergelijking

Vervolgens gaan we de randintegraal vergelijking omschrijven in een randelement vergelijking. Dit doen we door de oppervlakken S_i op te delen in randelementen S_{ie} met

$$S_i = \bigcup_{e=1}^{n_{ie}} S_{ie}, \tag{3.13}$$

waarbij n_{ie} het aantal randelementen is waarin oppervlak S_i in verdeeld is en de oppervlakken S_ie disjunct zijn. We krijgen dan als randelement vergelijking:

$$\begin{aligned}(\Omega^+\sigma_k^+ + \Omega^-\sigma_k^-)\psi(\mathbf{x}) &= -\sum_{j=1}^{5}\sigma_j \int_{V_j} \frac{1}{R}\nabla^2\psi dV + \\ &+\sum_{j=1}^{4}\sum_{e=1}^{n_{j\,e}}(\sigma_j^- - \sigma_j^+)\int_{S_{je}} \psi|_{S_{j\,e}} K_j(\mathbf{x},\mathbf{x_0})\cdot d\mathbf{S_j},\end{aligned} \tag{3.14}$$

met $K_j(\mathbf{x},\mathbf{x_0})$ als in (3.12).

Nu we de randelement vergelijking hebben kunnen we op de standaard manier verder.
We kiezen driehoekjes voor S_{ie} driehoekjes en vervolgens benaderen we de potentiaal op schil k door:

$$\psi_k(x,y,z) \approx \sum_{j=1}^{m_k} \hat{\psi}_{k_j} N_{k_j}(x,y,z). \tag{3.15}$$

Vullen we deze in (3.14) in dan krijgen we een stelsel lineaire vergelijkingen die we in de volgende vorm kunnen schrijven:

$$A\mathbf{x} = \mathbf{b} \tag{3.16}$$

met $\mathbf{x}$ de op te lossen onbekende potentiaal. De bronterm

$$\mathbf{b_j} = \mathbf{M}\cdot\frac{(\mathbf{x_j} - \mathbf{x_{dip}})}{||\mathbf{x_j} - \mathbf{x_{dip}}||^3} \tag{3.17}$$

met $\mathbf{x_{dip}}$ de plaats van de bron in volume V_5. En verder is

$$A_{ij} = (\Omega^+\sigma^+ \quad + \quad \Omega^-\sigma^-)\delta_{ij} \qquad (3.18)$$
$$\qquad - \quad (\sigma^- - \sigma^+)\int_{S_e} N_j(\mathbf{x})K(\mathbf{x_i},\mathbf{x})\cdot d\mathbf{S},$$

met

$$K(\mathbf{x_i},\mathbf{x}) = -\frac{\partial}{\partial n}\frac{1}{||\mathbf{x_i}-\mathbf{x}||},$$
$$= -\frac{\mathbf{x_i}-\mathbf{x}}{||\mathbf{x_i}-\mathbf{x}||^3}\cdot\mathbf{n}. \qquad (3.19)$$

3.4 Oplosmethodes

We hebben nu een integraalvergelijking omgeschreven naar een lineair stelsel vergelijkingen. Het enige wat ons nog rest is om te bepalen wat voor elementen we het beste kunnen gebruiken om een zo klein mogelijke fout te krijgen.
Kiezen we driehoekjes dan kunnen we onderscheid maken tussen de volgende soorten:

- platte driehoekjes met lineaire basisfuncties,
- gekromde driehoekjes met lineaire basisfuncties, die weer te onderscheiden zijn in gekromde driehoekjes door transformatie en gekromde driehoekjes door gebruik te maken van subdriehoekjes,
- kwadratische driehoekjes met kwadratische basisfuncties of
- kwadratische driehoekjes met lineaire basisfuncties, dit levert hetzelfde resultaat als gekromde driehoekjes door vier subdriehoekjes.

In de komende hoofdstukken zullen we de eerste drie mogelijkheden uitvoerig beschouwen om een uitspraak te kunnen doen over welke mogelijkheid het beste is.

Hoofdstuk 4

Verfijning van het grid

Het eerste alternatief om een nauwkeuriger oplossing te krijgen is het toepassen van verfijning van het grid. In dit hoofdstuk wordt gekeken of dit wel een goede methode is.

Om een uitspraak te kunnen doen over de geschiktheid van dit alternatief, is een C-programma geschreven dat de oplossing van het probleem berekend. Hierin is zowel de analytische oplossing op een bol als de oplossing verkregen met de Boundary Element Methode geprogrammeerd en wordt gekeken naar het verschil tussen de twee antwoorden. Verder is er gekeken welk onderdeel van het oplossen met behulp van de BEM de meeste tijd in beslag neemt gegeven een bepaald aantal driehoekjes.

4.1 Verschillend aantal driehoekjes

4.1.1 Fout

In deze paragraaf zullen we naar een aantal situaties kijken waarbij we het aantal driehoekjes veranderen. Als eerste zullen we de situatie beschouwen waarbij we te maken hebben met één enkele schil. Later zullen we ook het meer-schillen geval onderzoeken.
In tabel 4.1 wordt de fout bij verschillende meshverfijningen bekeken. Onder "fout" verstaan we de afwijking van de numerieke oplossing, verkregen met behulp van de Boundary Element Methode, ten opzichte van de analytische oplossing. Verder is de "Meshgrootte" als volgt gedefinieerd:

$$\text{Meshgrootte} = \frac{\text{straal}}{\text{n}}$$

met n als de varieërende grootheid en de straal R, als de straal van de bol (in dit specifieke geval R=10).

Als we naar de "fout" kijken is deze als volgt gedefinieerd:

$$\text{Fout}_{rel} = \frac{|\text{analytisch} - \text{BEM}|}{|\text{analytisch}|} * 100\%$$

en de ratio is dan:

$$\text{Ratio} \quad (n = 2^i) = \frac{\text{Fout}_{rel} \quad (n = 2^{i-1})}{\text{Fout}_{rel} \quad (n = 2^i)} \qquad i = 1, .., n.$$

We nemen hier de ℓ_1-norm en er moet opgemerkt worden dat we te maken hebben met potentialen die op een constante na bepaald zijn. Daarom is bij zowel de analytische als bij de BEM oplossing de gemiddelde potentiaal afgetrokken om te zorgen dat we met potentiaal verschillen kunnen werken. Verder moeten we opmerken dat de integralen in de systeemmatrix met analytisch gesloten uitdrukkingen berekend zijn.

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
1	13	22	53.20 %	
2	50	96	16.58 %	0.31
4	203	402	2.80 %	0.17
8	802	1600	0.77 %	0.28

Tabel 4.1: "Meshgrootte" vs. "Fout" voor platte driehoekjes

Wat in deze tabel opvalt is dat bij n=1 en bij n=2 de fout heel groot is en de berekende oplossing dus een onrealistische weergave van de werkelijkheid is. In het vervolg zullen we daarom deze twee waarden van n niet meer beschouwen.
Verder hangt het aantal punten en het aantal driehoekjes als volgt samen:

$$\text{aantal driehoekjes} = 2 \times (\text{aantal punten} - 2). \tag{4.1}$$

De eis voor het inverse probleem is dat dit met een nauwkeurigheid van 1% wordt opgelost. Dat betekent dat we voor het voorwaartse probleem in ieder geval een fout moeten krijgen die kleiner is dan 1%. Dit kan uiteraard door het aantal driehoekjes op te voeren, maar dan neemt de rekentijd onevenredig veel toe. Omdat het voorwaartse probleem vele malen moet worden opgelost om het inverse probleem op te lossen is rekentijd een heel belangrijke factor.

Uit deze eerste analyse kunnen we al concluderen dat het zin heeft om naar nauwkeurigere methodes te kijken voor het oplossen van de integraalvergelijking. Immers de fout bij n=8 is zelfs voor dit zeer eenvoudige geval erg dicht bij 1%, terwijl verdere verfijning een grotere computer en aanzienlijk meer rekentijd vergt. We zoeken nu naar een methode die met vergelijkbare inspanning als voor n=8 toch een betere nauwkeurigheid biedt.

4.1.2 Tijd per onderdeel

Onderstaande tabel geeft een overzicht van de tijd, bepaald met de time-functie, die nodig is voor elk onderdeel gegeven in seconden.

Onderdeel:	n=4	n=8
Bepaal discretisatiepunten	0.00	0.00
Maak driehoeksverdeling	0.05	0.55
Maak systeemmatrix A	1.49	23.35
Maak LU decompositie	0.38	76.23
Bereken b	0.00	0.06
Los stelsel op	0.00	8.07
Totale tijd	1.92	108.43

Tabel 4.2: Tijd per onderdeel (sec) voor platte driehoekjes

Bij deze tabel moeten we opmerken dat, bij het verschillende malen uitvoeren van de testen, de tijd een afwijking van ongeveer 10% vertoonde.

In het algemeen kunnen we uit deze tabel concluderen dat de meeste tijd gaat zitten in het maken van de systeem matrix en het maken van een LU-ontbinding. De tijd die in de andere onderdelen gaat zitten is verwaarloosbaar klein ten op zichte van deze twee onderdelen. Wat hierbij opvalt is dat voor n=4 de meeste tijd nog in het maken van de systeem matrix gaat zitten , maar als we naar een grotere n kijken, de meeste tijd aan de LU-decompositie gespendeerd wordt.
Wel moeten we hierbij opmerken dat het maken van de driehoeksverdeling

in dit geval relatief simpel is, omdat het uitgangspunt, de bol, analytisch bekend is. In een praktijksituatie kan dit onderdeel meer tijd vergen.

Opmerking

Het probleem dat we nu hebben opgelost is het voorwaartse probleem, maar we willen uiteindelijk het inverse probleem oplossen.
Voor het oplossen van het inverse probleem moet één keer de systeemmatrix en zijn LU ontbinding bepaald worden. Voor het oplossen van het inverse probleem wordt nu telkens een ander rechterlid (vector b) ingevuld en wel op zodanige wijze dat we een zo klein mogelijke afwijking van de gemeten potentiaal krijgen.
Het zal dus kunnen voorkomen dat dit ritueel 100 - 1000 keer moet worden uitgevoerd voor een goed antwoord. Het is daarom ook niet erg dat er zoveel tijd in de opbouw van de matrix en LU ontbinding gaat zitten. De totale tijd voor het oplossen van het stelsel neemt, bij het 100 - 1000 keer oplossen, een zo'n groot deel in dat de totale tijd dat deze overheersend is.

4.1.3 Geheugen gebruik

In deze paragraaf zullen we bekijken hoeveel geheugen we nodig hebben bij een bepaalde meshgrootte.
De twee onderdelen die het meeste geheugen in gebruik nemen is een matrix ter grootte van

$$\text{aantal punten} \times \text{aantal punten}$$

en een vector ter grootte van het aantal punten. Deze twee onderdelen zijn respectievelijk de systeemmatrix en de vector met de brontermen. Tellen we deze samen op en vermenigvuldigen we dit met het aantal bytes (8) dan krijgen we tabel 4.3.

Wat hier opvalt is dat als we naar een n kijken die groter is dan 8, we op een PC met 16 Mb te weinig geheugencapaciteit hebben.

4.2 Dipool positie, richting en sterkte

Zoals al is opgemerkt gaan we bij het inverse probleem meerdere dipool posities invullen om te bepalen welke bijbehorende potentiaal het best de meting benadert. Bij zo'n meting kan de corresponderende dipool overal in de hersenen zitten. Het is daarom ook van belang dat we weten hoe de nauwkeurigheid afhangt van verschillende dipool posities. In Bijlage C

n:	Aantal punten:	Geheugen (bytes):
1	13	$1.5 * 10^3$
2	50	$2.0 * 10^4$
4	203	$3.3 * 10^5$
8	802	$5.2 * 10^6$
16	3215	$8.3 * 10^7$

Tabel 4.3: "Meshgrootte" vs. "Geheugen"

hebben we de fout voor een aantal dipool posities, richtingen en diepten onderzocht.

We kunnen uit dit onderzoek concluderen dat het haast niet uitmaakt waar we de dipool kiezen zolang hij maar op dezelfde afstand van het oppervlak zit. Kiezen we de dipool daarentegen dieper onder het oppervlak dan krijgen we een nauwkeuriger oplossing. De fout is dus afhankelijk van de dipool diepte.
Veranderen we de richting met de positie mee dan krijgen we ook haast geen verschillen, zolang we maar op dezelfde afstand van het oppervlak zitten.
Verder heeft de sterkte van de dipool geen invloed op de grootte van de relatieve fout, maar heeft wel invloed op de grootte van de potentiaal.

Kijken we naar de uiterste waarden die we verkregen hebben dan zien we dat we voor de verandering van de positie de resultaten krijgen uit tabel 4.4. Hier valt op dat bij n=4 er een aardig verschil is en bij n=8 het verschil veel

Positie	Fout (n=4)	Fout (n=8)
(5,5,5)	2.80 %	0.77 %
(0,0,-5$\sqrt{3}$)	5.92 %	0.87 %

Tabel 4.4: Verschillende dipool posities; vaste dipoolrichting

kleiner is, waardoor we kunnen stellen dat de positie geen invloed heeft op de fout zolang de dipool op dezelfde afstand onder het oppervlak blijft.

Veranderen we de richting van de dipool met de positie mee dan resulteert dat in tabel 4.5. Voor deze tabel geldt weer hetzelfde als voor de vorige, er

Positie	Richting	Fout (n=4)	Fout (n=8)
$(5\sqrt{3},0,0)$	$(\sqrt{3},0,0)$	3.92 %	0.97 %
$(0,0,-5\sqrt{3})$	$(0,0,-\sqrt{3})$	8.48 %	1.06 %

Tabel 4.5: Verschillende dipool posities; meeveranderende dipool richting

treden haast geen verschillen als we de richting van de dipool mee veranderen.

Verder is er ook gekeken naar wat er gebeurt als we de dipool op verschillende afstanden onder het oppervlak nemen en dat levert ongeveer de volgende twee uitersten van tabel 4.6. Hier zien we dat hoe dieper we onder

Positie	Fout (n=4)	Fout (n=8)
(0,0,0.1)	1.62 %	0.38 %
(0,0,9.9)	78.83 %	101.07 %

Tabel 4.6: Verschillende dipool diepte; vaste dipoolrichting

het oppervlak gaan zitten hoe beter de oplossing, verkregen met de BEM, wordt. Gaan we daarentegen dicht onder het oppervlak zitten dan krijgen we een zeer onnauwkeurige oplossing waarbij het een fijner rooster zelfs niet helpt. De oorzaak van dit probleem zou kunnen zijn dat we te maken hebben met een singulariteit. Verder is het onrealistisch dat de dipool dicht onder de rand zit en daarom zullen we deze situatie in het vervolg proberen te vermijden.

Vervolgens kijken we wat er met de fout gebeurt als we de richting van de dipool veranderen, maar de positie en de sterkte gelijk houden. Dit levert tabel 4.7. We zien hier dat voor n=4 geldt dat de richting wel iets uitmaakt, maar voor n=8 maakt het haast niet meer uit hoe de richting wordt gekozen. In deze tabel valt verder op dat in sommige gevallen geldt dat bij n=4 de fout nog groter is, maar bij n=8 de fout kleiner is geworden dan de andere gevallen.

Tot slot zullen we nog kijken naar het geval dat alleen de sterkte van de

Richting	Fout (n=4)	Fout (n=8)
(1,-1,-2)	2.78 %	0.75 %
(-1,1,2)	2.78 %	0.75 %
(1,1,2)	4.82 %	0.92 %
(0,0,$\sqrt{6}$)	2.92 %	0.72 %

Tabel 4.7: Verschillende dipool richting; vaste positie

dipool verandert en dat levert de resultaten in tabel 4.8. In deze tabel zien

Positie	Fout
(0.5,-0.5,1)	2.80 %
(4,-4,8)	2.80 %

Tabel 4.8: Verschillende dipool sterkte; vaste dipoolrichting en positie

we dat de sterkte van de dipool geen invloed heeft op de grootte van de fout, maar we moeten hierbij opmerken dat de sterkte wel invloed heeft op de grootte van de potentiaal.

Hoofdstuk 5

Gekromde driehoekjes op een bol

De eerste mogelijkheid die we zullen onderzoeken is of het vervangen van platte driehoekjes door gekromde driehoekjes op het oppervlak tot verbetering leidt. We zullen allereerst trachten een gekromd driehoekje, op een bol, af te beelden op een platte driehoek en de integralen met behulp van deze transformatie te berekenen. Voor de bol is dit analytisch nog uitvoerbaar.
Als het mogelijk is om het driehoekje op een bol te transformeren, die we kunnen gebruiken om de integraal uit te rekenen dan wil dat nog niet zeggen dat het ook mogelijk moet zijn om deze methode te gebruiken voor het algemene geval. Het is duidelijk dat als het voor een bol niet lukt het zeker niet zal lukken voor het algemene geval. Voor de resultaten geldt in principe hetzelfde, goede resultaten op een bol wil nog niet zeggen dat we automatisch ook goede resultaten op een willekeurig oppervlak krijgen.

De reden waarom we dit doen is, omdat we willen voorkomen dat we tussenpunten nodig hebben. Deze tussenpunten hebben we nodig als we gebruik willen maken van kwadratisch driehoeken en het is een probleem om deze tussenpunten te bepalen. Een oplossing om bij het grove rooster de punten uit het fijne rooster te gebruiken werkt niet, omdat de set punten van het grove rooster geen deelverzameling hoeft te zijn van de set punten uit het fijne rooster.
De punten die we krijgen worden, met behulp van Delaunay triangulatie, verbonden tot een driehoeksgrid van platte driehoekjes waarbij de drie hoekpunten gegeven zijn. Willen we gekromde driehoekjes maken dan kunnen we dat doen met behulp van kwadratische driehoekjes en hebben we dus

tussenpunten nodig. Maar het probleem waar we dan op stuiten is hoe we aan die tussenpunten komen. Het zal ons dus goed uitkomen als we helemaal geen tussenpunten nodig hebben en er een transformatie bestaat waarbij we alleen de drie knooppunten gebruiken.

5.1 Transformatie

Als we ons probleem hebben omgeschreven naar een randelement vergelijking (zie paragraaf 3.3.3) dan moeten we de integraal in de onderstaande vergelijking nog uitrekenen:

$$A_{dj} = (\sigma^- - \sigma^+) \int_{S_e} N_j(\mathbf{x}) K(\mathbf{x_d}, \mathbf{x}) \cdot d\mathbf{S}, \tag{5.1}$$

met

$$\begin{aligned} K(\mathbf{x_d}, \mathbf{x}) &= -\frac{\partial}{\partial n} \frac{1}{||\mathbf{x_d} - \mathbf{x}||}, \\ &= -\frac{\mathbf{x_d} - \mathbf{x}}{||\mathbf{x_d} - \mathbf{x}||^3} \cdot \mathbf{n}. \end{aligned} \tag{5.2}$$

Hier is er gedifferentieerd en het inprodukt met de normaal

$$\mathbf{n} = \frac{\mathbf{x}}{||\mathbf{x}||} \tag{5.3}$$

genomen, met $|| \cdot ||$ de Euclidische norm.

Het bepalen van de integraal over een elementje zullen we doen door een transformatie naar het platte vlak toe te passen. Hiervoor gebruiken we de volgende transformatie:

$$\mathbf{x} = R \frac{\sum_{i=1}^{3} \mathbf{x_i} \lambda_i}{|| \sum_{i=1}^{3} \mathbf{x_i} \lambda_i ||} \tag{5.4}$$

met R de straal van de bol en

$$\begin{aligned} \lambda_1 &= 1 - \xi - \eta \\ \lambda_2 &= \xi \\ \lambda_3 &= \eta \end{aligned} \tag{5.5}$$

als de basisfuncties.
Deze transformatie beeldt een kromme driehoek op een bol precies af op een platte driehoek. Daarnaast wordt de rand van aanliggende driehoeken op eenduidige wijze afgebeeld zodat precies de hele bol wordt meegenomen.

Voor de transformatie van dS krijgen we dan:

$$dS = Jd\xi d\eta \tag{5.6}$$

met

$$J = \sqrt{J_x^2 + J_y^2 + J_z^2}. \tag{5.7}$$

Hierbij is

$$\begin{aligned} J_x &= \left(\frac{\partial y}{\partial \xi}\frac{\partial z}{\partial \eta} - \frac{\partial z}{\partial \xi}\frac{\partial y}{\partial \eta}\right), \\ J_y &= \left(\frac{\partial z}{\partial \xi}\frac{\partial x}{\partial \eta} - \frac{\partial x}{\partial \xi}\frac{\partial z}{\partial \eta}\right), \\ J_z &= \left(\frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi}\frac{\partial x}{\partial \eta}\right) \end{aligned} \tag{5.8}$$

(zie [10]).
Uiteindelijk krijgen we dan dus een transformatie die er als volgt uitziet:

$$\int_{x,y,z} f(x_0, y_0, z_0) dS = \int_{\xi,\eta} f(\xi_0, \eta_0) J d\xi_0 d\eta_0. \tag{5.9}$$

We moeten nu de Jacobiaan gaan uitrekenen. We doen dat door de bovenstaande formules (5.7) en (5.8) voor het bepalen van een Jacobiaan te gebruiken. Allereerst moeten we $\frac{\partial \mathbf{x}}{\partial \xi}$ en $\frac{\partial \mathbf{x}}{\partial \eta}$ bepalen:

$$\begin{aligned} \frac{\partial \mathbf{x}}{\partial \xi} &= R\frac{(\mathbf{x_2} - \mathbf{x_1})}{||\sum \mathbf{x_i}\lambda_i||} \\ &- R\frac{\sum \mathbf{x_i}\lambda_i}{||\sum \mathbf{x_i}\lambda_i||^3}\Big(\mathbf{x_1} \cdot (\mathbf{x_2} - \mathbf{x_1}) + ||\mathbf{x_2} - \mathbf{x_1}||^2\xi + (\mathbf{x_2} - \mathbf{x_1}) \cdot (\mathbf{x_3} - \mathbf{x_1})\eta\Big) \end{aligned} \tag{5.10}$$

en

$$\begin{aligned} \frac{\partial \mathbf{x}}{\partial \eta} &= R\frac{(\mathbf{x_3} - \mathbf{x_1})}{||\sum \mathbf{x_i}\lambda_i||} \\ &- R\frac{\sum \mathbf{x_i}\lambda_i}{||\sum \mathbf{x_i}\lambda_i||^3}\Big(\mathbf{x_1} \cdot (\mathbf{x_3} - \mathbf{x_1}) + ||\mathbf{x_3} - \mathbf{x_1}||^2\eta + (\mathbf{x_2} - \mathbf{x_1}) \cdot (\mathbf{x_3} - \mathbf{x_1})\xi\Big). \end{aligned} \tag{5.11}$$

Om de Jacobiaan nu in zijn geheel te krijgen moeten we nog gebruik maken van vergelijking (5.7).
De vergelijking voor K_j (5.2) zouden we ook kunnen transformeren, maar omdat we voor de berekening van de integraal gebruik gaan maken van numerieke integratie hoeven we K_j alleen in de integratiepunten te berekenen. Het is dus wel noodzakelijk de coördinaten van de bijbehorende Gauss-punten $\mathbf{x}_{gauss}$ te bepalen op de bol.

Verder nemen we lineaire basisfuncties:

$$\begin{aligned} N_1(\lambda_2, \lambda_3) &= 1 - \lambda_2 - \lambda_3, \\ N_2(\lambda_2, \lambda_3) &= \lambda_2, \\ N_3(\lambda_2, \lambda_3) &= \lambda_3. \end{aligned} \tag{5.12}$$

Deze zijn niet lineair op de bol zelf (zie opmerking 5.2.2).

5.1.1 Gauss integratie formules

Omdat sommige integralen niet exact uit te rekenen zijn wordt de integraal, in het getransformeerde element, benaderd met een integratieregel.
De algemene vorm van een integratieregel luidt:

$$\int_e Int(\zeta) d\zeta = \sum_{k=1}^{r} w_k Int(\zeta_k) \tag{5.13}$$

met:

- r het aantal steunpunten in de integratieregel,
- w_k de gewichten,
- ζ_k de steunpunten,
- Int(ζ) de integrand.

Gauss-formules zijn formules waarbij de steunpunten en gewichten zo gekozen zijn dat een hoge graad van nauwkeurigheid bereikt wordt. Alhoewel de integraal beter benaderd wordt vergt de evaluatie van hogere graad Gauss-formules meer tijd. Voor enkele standaardelementen worden de gewichten en steunpunten van Gauss-regels in tabellen gegeven ([1],p.90).

w_i	ζ_1	ζ_2	ζ_3	m
1.000000000000000	0.333333333333333	0.333333333333333	0.333333333333333	1

Tabel 5.1: Eénpunts Gauss-formule met graad van nauwkeurigheid 2

w_i	ζ_1	ζ_2	ζ_3	m
0.333333333333333	0.666666666666667	0.166666666666667	0.166666666666667	3

Tabel 5.2: Driepunts Gauss-formule met graad van nauwkeurigheid 2

w_i	ζ_1	ζ_2	ζ_3	m
-0.562500000000000	0.333333333333333	0.333333333333333	0.333333333333333	1
0.520833333333333	0.600000000000000	0.200000000000000	0.200000000000000	3

Tabel 5.3: Vierpunts Gauss-formule met graad van nauwkeurigheid 3

w_i	ζ_1	ζ_2	ζ_3	m
0.109951743655322	0.816847572980459	0.091576213509771	0.091576213509771	3
0.223381589678011	0.108103018168070	0.445948490915965	0.445948490915965	3

Tabel 5.4: Zespunts Gauss-formule met graad van nauwkeurigheid 4

w_i	ζ_1	ζ_2	ζ_3	m
0.225030000300000	0.333333333333333	0.333333333333333	0.333333333333333	1
0.125939180544827	0.797426985353087	0.101286507323456	0.101286507323456	3
0.132394152788506	0.470142064105115	0.470142064105115	0.059715871789770	3

Tabel 5.5: Zevenpunts Gauss-formule met graad van nauwkeurigheid 5

Enkele voorbeelden van deze tabellen zijn weergegeven in de tabellen 5.1 tot en met 5.5. Hierbij is w_i de gewichtsfunctie behorende bij het punt, in de driehoek, met driehoeks coördinaten $(\zeta_1, \zeta_2, \zeta_3)$ en m is het aantal malen dat deze coördinaten voorkomen waarbij cyclische verwisseling wordt toegepast (zie ook [6]).

We moeten ons nu de vraag stellen welke formule we gaan gebruiken. Het is duidelijk dat de zevenpunts Gauss-formule het beste is, maar als we met bijvoorbeeld de vierpunts Gauss-formule dezelfde nauwkeurigheid kunnen bereiken dan is het verstandiger om deze te gebruiken. Een formule waarbij weinig punten worden meegenomen vergt immers minder rekentijd dan een formule met veel punten. In bijlage B is gekeken welke integratie formule we het beste kunnen gebruiken en het resultaat van dat onderzoek is dat de 3-punts Gauss-formule voor ons probleem goed genoeg is.

5.2 Test resultaten voor gekromde driehoekjes

Na het uitvoeren van de transformatie kunnen we het probleem gaan implementeren zodat we het kunnen vergelijken met de resultaten verkregen voor het geval we te maken hebben met platte driehoekjes op een bolschil.

5.2.1 Fout

De resultaten voor de fout, die verkregen zijn voor het geval we gekromde driehoekjes op een bol gebruiken, zijn weergegeven in onderstaande tabel:

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
4	203	402	1.96 %	0.16
8	802	1600	0.45 %	0.23

Tabel 5.6: "Meshgrootte" vs. "Fout" voor gekromde driehoekjes

Vergelijken we deze resultaten verkregen bij de platte driehoekjes (zie tabel 4.1) dan zien we dat de "Fout" kleiner is geworden zoals we wel konden verwachten. De gekromde driehoekjes zijn tenslotte een betere benadering van het oppervlak dan de platte driehoekjes.

5.2.2 Tijd per onderdeel

Tabel 5.7 geeft een overzicht van de tijd die nodig is voor elk onderdeel gegeven in seconden.

Over het algemeen kunnen we, net als in tabel 4.2, over deze tabel zeggen dat de meeste tijd gaat zitten in het maken van de systeem matrix en het maken van een LU-ontbinding ervan. De tijd die in de andere onderdelen gaat zitten is verwaarloosbaar klein ten op zichte van deze twee onderdelen.

Wel is duidelijk dat de matrixopbouw duurder is geworden bij het gebruik

Onderdeel:	n_{gauss3}=4	n_{gauss3}=8
Bepaal discretisatiepunten	0.00	0.00
Maak driehoeksverdeling	0.06	0.54
Maak systeemmatrix A	2.74	44.38
Maak LU decompositie	0.59	79.15
Bereken b	0.00	0.06
Los stelsel op	0.00	6.75
Totale tijd	3.30	131.05

Tabel 5.7: Tijd per onderdeel (sec) voor gekromde driehoekjes

van kromme driehoekjes. Ruwweg is er sprake van een verdubbeling van de rekentijd. Voor de totale rekentijd voor n=8 geldt dat de toename 15 à 20 % is.

Opmerking

In dit hoofdstuk hebben we gekeken naar gekromde driehoekjes. We hebben daarbij lineaire basisfuncties gebruikt. Deze basisfunctie zijn echter de basisfuncties die gebruikt worden op platte driehoekjes. Het zou beter zijn als de gebruikte basisfuncties lineair langs het gekromde oppervlak zouden lopen. Het probleem daarbij is echter dat deze niet op een eenvoudige wijze te bepalen zijn, daarom is ervoor gekozen als benadering de basisfuncties voor een plat driehoekje te gebruiken.

5.3 Meerdere Driehoeken

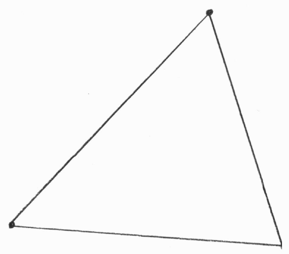
In deze paragraaf zullen we een tweede manier beschrijven voor het genereren van gekromde driehoekjes.

5.3.1 Hoe maken we gekromde driehoekjes?

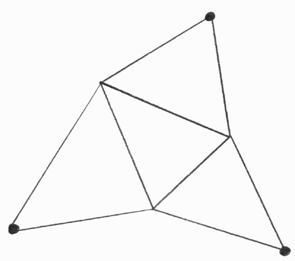
De tweede methode voor het maken van gekromde driehoekjes die we hebben bekeken komt op het volgende neer. We zijn uit gegaan van de drie punten van een driehoek zoals in de andere gevallen. Bij deze drie punten hebben we enkele tussenpunten bepaald door gebruik te maken van het feit dat we de geometrie van een bol helemaal kunnen beschrijven. Deze tussenpunten worden dan halverwege twee knooppunten op de bol gekozen en worden samen met de knooppunten gebruikt om een verfijnd driehoeksgrid te maken van vier driehoeken. De waarde van de potentiaal wordt echter alleen in de knooppunten als onbekend beschouwd waardoor de systeemmatrix even groot blijft maar het grid een nauwkeuriger beschrijving van het oppervlak is. De potentiaal in de tussenpunten wordt uitgedrukt in de potentiaal van de twee nabuur knooppunten, dat wil zeggen

$$\psi_{kj} = \frac{1}{2}(\psi_k + \psi_j).$$

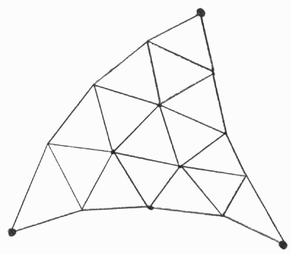
Verder wordt hier gebruik gemaakt van lineaire basisfuncties over de gekromde driehoek.
Om duidelijk te maken hoe zo'n gekromde driehoeken er dan uit gaat zien beschouwen we de onderstaande figuren.



Figuur 5.1: 1 driehoek



Figuur 5.2: 4 subdriehoeken



Figuur 5.3: 16 subdriehoeken

We zien hier dat we in alle drie de gevallen drie knooppunten hebben. Verder zien we dat in figuur 5.2 de platte driehoek in vier sub-driehoekjes is verdeeld en in figuur 5.3 we zelfs nog een stap verder gaan en we de driehoek in zestien sub-driehoeken hebben verdeeld. De punten van de subdriehoeken liggen op de bol, dus niet meer in de platte driehoek.

5.3.2 Fout

Voor de situatie dat we de driehoek in vieren gedeeld hebben krijgen we de resultaten voor wat betreft de fout uit tabel 5.8 en voor wat betreft het geval dat we het oppervlak in zestien driehoekjes verdelen krijgen we de resultaten uit tabel 5.9.

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
4	203	402	2.02 %	0.15
8	802	1600	0.47 %	0.23

Tabel 5.8: "Meshgrootte" vs. "Fout" bij gekromde driehoekjes als 4 subdriehoekjes

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
4	203	402	1.96 %	0.16
8	802	1600	0.45 %	0.23

Tabel 5.9: "Meshgrootte" vs. "Fout" bij gekromde driehoekjes als 16 subdriehoekjes

Als we deze twee tabellen (5.8 en 5.9) nu vergelijken met de waarden verkregen in paragraaf 5.2 dan kunnen we concluderen dat bij n=8, voor de fout, het niet uitmaakt welk soort gekromd driehoekje we kiezen. In het geval dat n=4 zien we dat het gekromde driehoekje bestaande uit 4 platte driehoekjes niet zo goed is als de twee andere soorten. In feite kunnen we dus concluderen dat de aanpak met de "kromme" driehoek relatief de beste is.

5.3.3 Tijd

In deze paragraaf zullen we de tijd die de verschillende onderdelen kosten beschouwen. We zullen dit voor zowel de variant met 4 driehoeken als de verfijning daarvan naar 16 driehoeken.
Als eerste kijken we naar het geval van één driehoek die onderverdeeld wordt in 4 driehoeken. De tijd die de verschillende onderdelen dan kosten is weergegeven in tabel 5.10.
Kijken we nu weer naar het geval dat we één driehoek in 16 driehoeken hebben onderverdeeld dan krijgen we tabel 5.11 voor de tijd.
Vergelijken we deze twee tabellen met tabel 5.7 dan valt het op dat het gebruik van de methode van het onderverdelen in subdriehoekjes veel meer tijd kost als het gebruik van de methode beschreven in paragraaf 5.1. De onderverdeling in 16 subdriehoekjes kost zelfs ongeveer 3 keer zoveel tijd.

Onderdeel:	n=4	n=8
Bepaal discretisatiepunten	0.00	0.00
Maak driehoeksverdeling	0.05	0.60
Maak systeemmatrix A	5.77	90.03
Maak LU decompositie	0.38	77.83
Bereken b	0.00	0.05
Los stelsel op	0.00	6.37
Totale tijd	6.20	175.05

Tabel 5.10: Tijd per onderdeel (sec) voor 4 subdriehokjes

Onderdeel:	n=4	n=8
Bepaal discretisatiepunten	0.00	0.00
Maak driehoeksverdeling	0.06	0.53
Maak systeemmatrix A	21.86	347.65
Maak LU decompositie	0.38	86.78
Bereken b	0.00	0.06
Los stelsel op	0.00	7.69
Totale tijd	22.36	442.92

Tabel 5.11: Tijd per onderdeel (sec) voor 16 subdriehoekjes

We kunnen uit deze gegevens wel concluderen dat, voor de bol, geldt dat de methode beschreven in paragraaf 5.1 de beste methode is. Daar is tenslotte de fout klein en het kost, vergeleken met de andere methode, weinig tijd.

Hoofdstuk 6

Kwadratische Elementen

Naast de lineaire elementen, die we in de voorgaande hoofdstukken hebben gebruikt kunnen we ook hogere orde elementen gebruiken. Dit zijn elementen, die naast de potentiaal in de knooppunten, ook de potentiaal in tussenpunten als onbekend beschouwen. Dat wil dus zeggen dat we meer onbekenden hebben en dus ook meer vergelijkingen nodig hebben, met andere woorden, we krijgen een groter systeem.
De meest voorkomende hogere orde elementen zijn de zogenaamde kwadratische elementen en deze zullen we in het vervolg van dit hoofdstuk gebruiken voor het oplossen van ons probleem.

6.1 Gebruik van kwadratische elementen

Even als bij lineaire elementen gaan we ook hier de potentiaal benaderen en dat gebeurt met hetzelfde principe. We moeten er alleen opletten dat we hier drie extra punten hebben en dus moet sommeren over zes punten.
We krijgen dan:

$$\mathbf{x}(\xi,\eta) = \sum_{i=1}^{6} \varphi_i(\xi,\eta)\mathbf{x_i} \tag{6.1}$$

waarbij φ_i de basisfuncties zijn, $\mathbf{x_1}$, $\mathbf{x_2}$ en $\mathbf{x_3}$ de knooppunten en $\mathbf{x_4}$, $\mathbf{x_5}$ en $\mathbf{x_6}$ de tussenpunten.
Deze gegevens kunnen we nu gebruiken voor het uitrekenen van de Jacobiaan en dat doen we op dezelfde manier als beschreven in 5.7 .
Verder benaderen we de potentiaal $\hat{\psi}$ door een kwadratisch polynoom ψ dat de waarde van $\hat{\psi}$ in de knooppunten heeft.

Daaruit volgt

$$\psi = \sum_{i=1}^{6} \varphi_i(\xi, \eta)\hat{\psi}_i. \tag{6.2}$$

Hierbij geldt dat

$$\begin{aligned} \varphi_1 &= \lambda_1(2\lambda_1 - 1) \qquad & \varphi_4 &= 4\lambda_1\lambda_2 \\ \varphi_2 &= \lambda_2(2\lambda_2 - 1) \qquad & \varphi_5 &= 4\lambda_2\lambda_3 \\ \varphi_3 &= \lambda_3(2\lambda_3 - 1) \qquad & \varphi_6 &= 4\lambda_3\lambda_1 \end{aligned} \tag{6.3}$$

waarbij λ_1, λ_2 en λ_3 gedefinieerd zijn als 5.5 .

6.2 Van lineair naar kwadratisch

Zoals al eerder vermeld hebben we bij het gebruik van kwadratische elementen een lijst met driehoeken nodig die bestaat uit drie knooppunten en drie tussenpunten.
Tot nu toe hebben we telkens gebruik gemaakt van een lijst van driehoeken die bestaan uit drie knooppunten. Het probleem waar we nu tegenaan lopen is hoe we aan de tussenpunten komen.
We zullen dus eerst een programma moeten schrijven dat van een lijst van drie punten een lijst van zes punten maakt en de lijst met coördinaten aanvult met de coördinaten van de tussenpunten. Het probleem hierbij is echter het nummeren van de drie zijden (de edges) van de driehoekjes. We mogen tenslotte een edge niet twee keer nummeren.
Dit nummeren van de edges is nodig, omdat we te maken hebben met kwadratische driehoekjes met kwadratische basisfuncties, dat wil zeggen dat de potentiaal ψ ook onbekende is op het midden van de zijde. Het bijbehorende nummer wordt bepaald door het edge nummer. De bijbehorende potentiaal moet uiteraard uniek zij, en één waarde hebben voor beide aanliggende driehoeken.

6.2.1 Programma

Het programma zal er dan, grofweg, als volgt uit komen te zien als we als **uitgangspunt** hebben

- een lijst met coördinaten van de knoopunten
- en een lijst met driehoeks indexen.

Het **doel** wat we voor ogen hebben is

- een lijst met coördinaten van de knoopunten en tussenpunten
- en een lijst met driehoeks indexen van zes punten.

Om dit te bereiken moeten we de onderstaande stappen volgen:

Stap 1:
als eerst maken we een lijst met daarin voor ieder punt in welke driehoeken hij ligt. Deze lijst zullen we gebruiken om te kijken of een bepaalde edge al eerder genummerd is.

Stap 2:
daarna gaan we voor elk driehoekje kijken of er een driehoekje met een lager nummer is dat twee knooppunten bevat die ook in het driehoekje waar we naar kijken liggen. Met ander woorden of een edge al eerder is genummerd in een eerder genummerd driehoekje.

Stap 3:
nu weten of een edge al eerder is genummerd of niet dan kunnen we de edges bij het beschouwde driehoekje gaan nummeren. Doe dat zo dat als we, in stap 2, ja krijgen het bijbehorende nummer wordt gegeven en als we, in stap 2, nee krijgen een nieuw nummer plaatsen. Bij dat nieuwe nummer moeten we ook de coördinaten bepalen.

Stap 4:
we hebben nu een lijst met knooppunten en een lijst met tussenpunten en deze twee moeten samengevoegd worden tot één lijst. Verder hebben we ook twee lijsten met coördinaten, van knoop- en tussenpunten, die we samen voegen.

We hebben nu een lijst met punten die samen een kwadratisch driehoekje vormen en een lijst van de bijbehorende coördinaten.

6.3 Fout

Ook voor dit geval zullen we weer naar de fout gaan kijken, maar eerst moeten we opmerken dat, zoals al gezegd, de systeemmatrix groter is geworden. Daardoor kunnen we geen vergelijking maken met de situaties n=4 en n=8 zoals in de voorgaande gevallen. Als eerste zullen we dan ook gaan kijken met welke gevallen we de voorgaande situaties wel kunnen vergelijken.

6.3.1 Aantal punten

Doordat we naar kwadratische driehoekjes gaan kijken, krijgen we te maken met een aantal extra punten. Deze punten zullen we in het vervolg, de tussenpunten noemen en we vinden ze halverwege de twee knooppunten, hier op de bol, terug.
Het aantal tussenpunten wordt bepaald door het aantal driehoekjes. Dit gebeurt met formule 6.4.

$$\text{aantal tussenpunten} = \frac{3}{2} \times \text{aantal driehoekjes.} \tag{6.4}$$

Voor het totaal aantal punten waar we dan mee te maken krijgen geldt dan dat

$$\text{aantal punten} = \text{aantal knooppunten} + \text{aantal tussenpunten.} \tag{6.5}$$

Samen met vergelijking 4.1 geldt dan dat we het totaal aantal punten als functie van het aantal knooppunten kunnen schrijven als

$$\text{totaal aantal punten} = 4 \times \text{aantal knooppunten} - 6. \tag{6.6}$$

Gebruiken we deze formule dan moeten we bij n=4 en n=8 naar een situatie zoeken die daarmee overeenkomt. Het ligt dan voor de hand om daar n=2 respectievelijk n=4 voor te kiezen.
Vergelijken we nu het aantal punten dan zien we dat dit aardig overeenkomt en dus zijn n=2 en n=4 een goede keuze om te gebruiken voor de vergelijking met de andere methodes.

6.3.2 Keuze Gauss-formule

Ook hier zullen we weer moeten beslissen welke Gauss-formule we zullen gaan gebruiken. Daarvoor hebben we weer naar de fout gekeken bij de verschillende formules en zijn er vanuit gegaan dat de Gauss-formule van de hoogste orde in ieder geval een goede benadering is. De vraag is alleen of er een Gauss-formule van lagere orde is die dezelfde resultaten levert. De resultaten van dit onderzoek zijn weergegeven in bijlage B.3 en de uitkomst is dat de Gauss3-formule al een redelijk goede benadering is.

6.3.3 Berekening fout

Nu we weten welke situatie we moeten gebruiken, om een vergelijking te kunnen maken met de andere methodes, kunnen we de fout bepalen en

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
2	206	96	6.02 %	0.19
4	818	402	0.54 %	0.09

Tabel 6.1: "Meshgrootte" vs. "Fout" bij kwadratische driehoekjes

dat levert de resultaten uit tabel 6.1. We zien dat het aantal punten niet helemaal overeenkomt met het aantal punten bij, de oude situatie, n=4 en n=8, maar het verschil is zo klein dat we ze wel kunnen gebruiken.
Vergelijken we de resultaten met de resultaten uit de tabellen 4.1 en 5.6 dan zien we dat we voor het geval van n=2 een grotere afwijking hebben dan bij de andere twee methodes.
Verder zien we dat voor het geval van n=4 de fout al heel dicht in de buurt komt van de waarde die we hadden verkregen door gekromde driehoekjes te gebruiken, maar dat er een veel grotere afname van de fout is bij de verfijning. Het is dus te verwachten dan bij een verdere verfijning de fout zelfs minder wordt dan bij gekromde driehoekjes.

6.3.4 Verdere verfijning

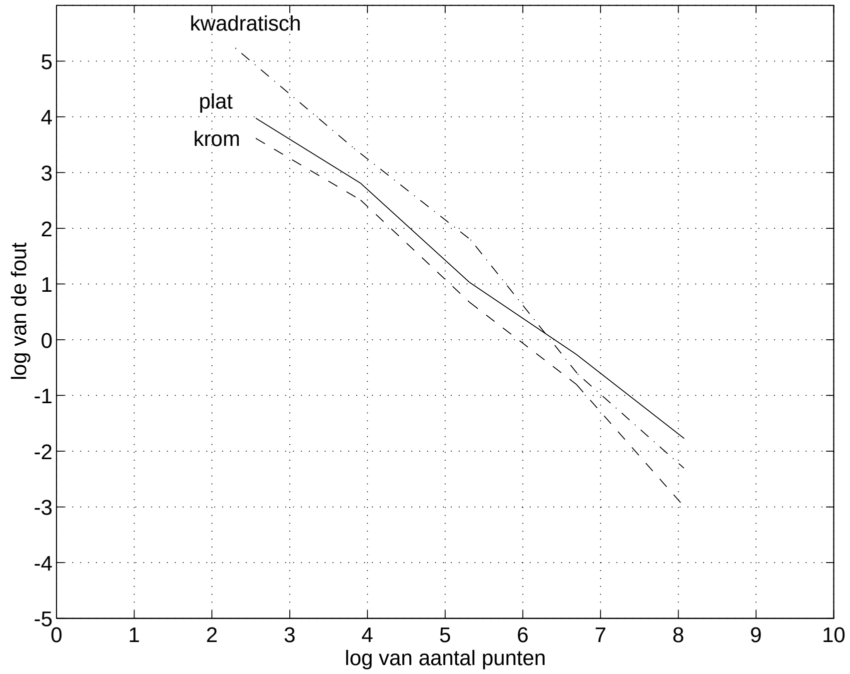
In deze paragraaf zullen we kijken of het gebruik van kwadratische driehoekjes inderdaad, bij een verdere verfijning, beter wordt dan de fout verkregen bij gekromde driehoekjes.
In paragraaf 4.1.3 hadden we al gezien dat voor verdere verfijning een PC met 16 Mb geheugen niet voldoende zou zijn om te gebruiken voor verdere verfijning. We zullen dus over moeten stappen naar een computer met meer geheugen.
Nadat we die overstap gemaakt hebben en we de testen, voor een fijner rooster opnieuw uitvoeren krijgen we de resultaten uit tabel 6.2. We zien hier

Driehoek:	n:	Aantal punten:	Fout:
plat	16	3215	0.17 %
krom	16	3215	0.05 %
kwadratisch	8	3202	0.10 %

Tabel 6.2: "Fout" bij verdere verfijning

dat we niet het verwachte resultaat hebben bereikt. De gekromde elementen zijn nog steeds de beste keuze en het ziet er, als we naar figuur 6.1 kijken, niet naar uit dat de kwadratische elementen een betere keuze zullen worden.

Zetten we de fout van de drie elementen in een grafiek dan krijgen we figuur 6.1.



Figuur 6.1: Fout vs. Aantal punten

Hier is duidelijk te zien dat gekromde elementen altijd beter zijn dan platte elementen en dat kwadratische elementen ook al snel beter zijn dan de platte elementen.
Wat verder opvalt is dat het er in het begin, naar uitziet dat kwadratische elementen bij meer punten beter wordt dan gekromde elementen, maar dat dit niet het geval is. Het is zelfs zo dat het verschil van de fout tussen de beide elementen groter is geworden.

6.4 Tijd

Gaan we nu ook hier kijken naar de tijd dan moeten we er rekening mee houden dat er een extra deel is bijgekomen, namelijk het omschrijven van een driehoekslijst van drie punten naar een driehoekslijst van zes punten. Dit deel neemt ook tijd in beslag, maar dan is het nog de vraag of dat van zo'n orde is dat we er wel degelijk rekening mee moeten houden.
De gemeten tijden zijn weergegeven in tabel 6.3. We zien in deze tabel

Onderdeel:	n=2	n=4
Bepaal discretisatiepunten	0.00	0.00
Maak driehoeksverdeling	0.00	0.05
Maak kwadratisch	0.00	0.00
Maak systeemmatrix A	0.61	13.84
Maak LU decompositie	0.33	86.94
Bereken b	0.00	0.06
Los stelsel op	0.00	6.59
Totale tijd	0.94	107.65

Tabel 6.3: Tijd per onderdeel (sec) voor kwadratische driehoekjes

dat het extra deel haast geen extra tijd kost. Verder valt in deze tabel op dat de totale tijd een stuk minder is dan de totale tijd voor het gebruik van gekromde driehoekjes uit tabel 5.7. Dit tijdsverschil komt voornamelijk tot uitdrukking in de opbouw van de matrix, omdat we hier met minder driehoekjes te maken hebben.

6.5 Conclusie

De conclusie die we uit het bovenstaande verhaal kunnen trekken is dat het gebruik van kwadratische driehoekjes, bij hetzelfde aantal driehoekjes, in eerste instantie onnauwkeuriger is dan gekromde driehoekjes, maar wel een kleinere fout levert dan platte driehoekjes.
Wat opvalt is dat de ratio bij kwadratische elementen veel kleiner is dan de andere elementen en dat zou kunnen betekenen dat bij n=16, kwadratische elementen de kleinste fout geven. Dit moest dus nog verder onderzocht

worden en het resultaat daarvan was dat dit niet het geval was. Gekromde elementen zijn dus, tegen de verwachting in, de beste keuze bij het model met 1 schil. Om één of andere reden krijgen we dus niet het te verwachten resultaat, dat kwadratische elementen de kleinste fout zouden leveren, en zou er dus nog onderzocht moeten worden wat de oorzaak daarvan is.

Hoofdstuk 7

Meer-schillen model

Tot nu toe hebben we telkens gekeken naar het geval dat we met één schil te maken hebben. Dit was de eerste stap naar een realistisch hoofdmodel. De tweede stap is het meer-schillen model. In het hoofd moeten we meerdere schillen onderscheiden, zoals de huid, de schedel en de hersenen. Daarom zullen we in dit hoofdstuk gaan kijken hoe goed de Boundary Element Methode, met de verschillende soorten driehoekjes, werkt voor het meer-schillen model.

7.1 Meer schillen

Door het gebruik van meerdere schillen zullen er enkele dingen veranderen ten opzichte van een enkele schil. In deze paragraaf zullen we deze onderdelen bespreken.

7.1.1 Model

We gaan nu dus naar een model kijken waarbij we te maken hebben met meerdere schillen. Het model daarvoor wat we daarvoor moeten oplossen was gegeven als

$$A\mathbf{x} = \mathbf{b}, \tag{7.1}$$

met $\mathbf{x}$ de op te lossen onbekende potentiaal. De bronterm

$$\mathbf{b_j} = \mathbf{M} \cdot \frac{(\mathbf{x_j} - \mathbf{x_{dip}})}{||\mathbf{x_j} - \mathbf{x_{dip}}||^3}, \tag{7.2}$$

met $\mathbf{x_{dip}}$ de plaats van de bron in volume V_5.
De matrix elementen worden gegeven door:

$$A_{ij} \quad = \quad (\Omega^+\sigma_k^+ + \Omega^-\sigma_k^-)\delta_{ij} - (\sigma_l^- - \sigma_l^+)\int_{S_e} N_j(\mathbf{x})K_l(\mathbf{x_i},\mathbf{x})\cdot d\mathbf{S}, \tag{7.3}$$

Dit alles met

$$K_l(\mathbf{x_i},\mathbf{x}) = -\frac{\mathbf{x_i}-\mathbf{x}}{||\mathbf{x_i}-\mathbf{x}||^3}\cdot\mathbf{n_l}, \tag{7.4}$$

en

$$i,j = 1....m_1, m_1+1....m_2, m_2+1..........m_{N-1}....m_N, \tag{7.5}$$

waarbij m_i het aantal punten op schil i is.
Verder geldt

$$\begin{array}{lllll} k & = & 1 & als & i = 1....m_1, \\ k & = & 2 & als & i = m_1+1....m_2, \\ & : & & & \\ & : & & & \\ k & = & N & als & i = m_{N-1}+1....m_N, \end{array} \tag{7.6}$$

en

$$\begin{array}{lllll} l & = & 1 & als & S_e = S_1, \\ l & = & 2 & als & S_e = S_2, \\ & : & & & \\ & : & & & \\ l & = & N & als & S_e = S_N. \end{array} \tag{7.7}$$

We zien hier dat we alle N schillen moeten beleggen waardoor we met een groter systeem te maken krijgen.

7.1.2 Aantal punten

Door het gebruik van meerdere schillen zal ook het aantal punten veranderen, we moeten immers meerdere schillen gaan beleggen.
Het aantal punten waar we nu mee te maken krijgen is de som van het aantal punten op elke schil. Dit aantal wordt bepaald door elke schil apart te

beleggen met een vaste afstand tussen de punten.
Dit wil dus zeggen dat op een schil met een grotere straal meer punten komen te liggen dan op een schil met een kleine straal.
Kijken we naar het aantal knooppunten wat dan op een schil komt te liggen dan krijgen we de volgende tabel:

n:	Schil 1 (R=10)	Schil 2 (R=9)	Schil 3 (R=8)
1	13	11	9
2	50	42	33
4	203	162	128
8	802	650	514
16	3215	2607	2058

Tabel 7.1: Aantal knooppunten per schil

7.1.3 Geheugen gebruik

Ook voor het geval van meerdere schillen hebben we gekeken hoeveel geheugen er minimaal nodig is voor het oplossen van het probleem en dat leverde de waarde uit tabel 7.2. We zien in deze tabel dat we aanzienlijk

n:	2 schillen	3 schillen
1	$4.8 * 10^3$	$9.0 * 10^3$
2	$6.8 * 10^5$	$1.3 * 10^5$
4	$1.1 * 10^6$	$1.3 * 10^6$
8	$1.7 * 10^7$	$3.1 * 10^7$
16	$2.7 * 10^8$	$5.0 * 10^8$

Tabel 7.2: Geheugen gebruik (in bytes) bij meerdere schillen

meer geheugen nodig hebben naar mate we meer schillen moeten leggen. We krijgen dan ook te maken met ongeveer twee keer respectievelijk drie keer zoveel punten en omdat de matrix het meeste geheugen in beslag neemt zal er dan ook ongeveer vier respectievelijk negen keer zoveel geheugenruimte nodig moeten zijn.

7.1.4 Conductiviteit en straal van de schil

Daar we nu naar meerdere schillen gaan kijken, krijgen we ook te maken met meerdere compartimenten die verschillen in conductiviteit. Dat wil dus zeggen dat dit ook van invloed is op de grootte van de potentiaal en misschien ook op de fout.
Ook de grootte van de verschillende schillen kan van invloed kan zijn op de potentiaal en op de fout. Dat wil zeggen dat als we naar een situatie kijken waarbij we te maken met kleine verschillen tussen de verschillende stralen, we andere resultaten zullen krijgen dan de situatie met grotere verschillen tussen de stralen.

We moeten er rekening mee houden dat we uit eindelijk naar een realistische situatie willen gaan kijken en in een realistische situatie is zowel de straal als de conductiviteit een vast gegeven. We zullen dus ook voor het bolschillen model moeten kijken naar een situatie waarbij we te maken hebben met compartimenten met een vaste straal en conductiviteit. Dit zou kunnen betekenen dat bij een andere keuze de fout zowel gunstiger als ongunstiger kan uitpakken.
Bij onze metingen we hebben we de keuzes uit tabel 7.3 voor het twee-schillen model gemaakt en voor het drie schillen model de keuzes uit tabel 7.4.

	Schil 1	Schil 2
Straal (R)	10	9
Conductiviteit (σ)	0.1	1.0

Tabel 7.3: Gegevens 2 schillen model

	Schil 1	Schil 2	Schil 3
Straal (R)	10	9	8
Conductiviteit (σ)	1.0	0.1	1.0

Tabel 7.4: Gegevens 3 schillen model

Hierbij stelt schil 1 de huid voor, schil 2 de slecht geleidende schedel en schil 3 de hersenen.
De reden waarvoor we tot de gekozen waarden van de conductiviteit zijn gekomen is, omdat we in de werkelijkheid ook met zulke, relatieve, verschillen te maken hebben. Dat wil dus zeggen dat we tussen twee goed geleidende schillen te maken hebben met een slecht geleidende schil.
Voor wat betreft de keuzes voor het twee schillen model hebben we een schil toegevoegd, maar de buitenstraal gelijk gehouden.

De gegevens zullen we in alle test situaties gebruiken, tenzij anders vermeld.

7.1.5 De berekening van de fout

De berekening van de fout zal weer op dezelfde manier gaan als eerder is gedaan. We zullen hier alleen de fout op de buitenste schil bepalen, omdat we uiteindelijk willen weten wat daar de potentiaal is.
We moeten er rekening mee houden dat de fout waarschijnlijk hoger zal uit vallen dan bij het model met een enkele schil. Dit komt door het feit dat we de potentiaal op meerdere schillen moeten uitrekenen en daarvoor hebben we de potentiaal van de andere schillen nodig. Dat wil dus zeggen dat de fout gemaakt bij een schil doorwerkt in de fout van de andere schillen.

7.1.6 Positie van de dipool

We moeten bij het kiezen van de positie van de dipool er rekening meehouden dat we die niet zomaar binnen elk compartiment kunnen kiezen. In de werkelijkheid zit de bron immers niet in de schedel, maar in de hersenen. Daarom is het van belang dat we de positie van de dipool op zodanige wijze kiezen dat hij in het binnenste compartiment ligt.
Kijken we naar de positie van de dipool bij het model met één schil, dan zien we dat deze waarschijnlijk niet voldoet, omdat hij dan of niet in de binnenste compartiment ligt of te dicht bij het oppervlak ligt.
We hebben daarom een andere positie gekozen en wel de volgende:

$$\text{Positie dipool:}(2, 2, 2)$$

Opmerking: we kunnen de resultaten die we bij de meer schillen modellen krijgen niet vergelijken met het model met één schil. We kunnen de resultaten die we daar verkregen hebben alleen gebruiken om te bepalen welke situaties we beter niet kunnen gebruiken en welke juist wel.

7.2 Twee-schillen model

Als eerste zullen we naar het twee-schillen model (N=2) gaan kijken. Dit model zal als tussenmodel fungeren voor het model waar we uit eindelijk naar willen gaan kijken.

7.2.1 Fout bij platte driehoekjes

Als eerste zullen we gaan kijken naar het geval we te maken hebben met platte driehoekjes.
De fout die we voor die situatie krijgen is gegeven als tabel 7.5. We hebben

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
1	24	40	552.36 %	
2	92	176	44.57 %	0.08
4	365	722	10.00 %	0.22
8	1452	2896	2.57 %	0.26

Tabel 7.5: "Meshgrootte" vs. "Fout" voor platte driehoekjes

bij het model met één schil al gezien dat n=1 en n=2 niet voldoen en een onrealistische weergave van de werkelijkheid is. Ook hier zien we die we dat n=1 en n=2 niet voldoen, maar nu zien we ook dat bij n=4 een grote fout krijgen.

We krijgen hier dus te maken met het probleem dat we naar situaties moeten gaan kijken waarbij we een groot aantal punten voor moeten gebruiken willen een realistische weergave van de werkelijkheid krijgen.

7.2.2 Fout bij gekromde driehoekjes

Ook voor het geval we gekromde driehoekjes gebruiken hebben we naar de fout gekeken. Het resultaat daarvan is weergeven in tabel 7.6.
We zien hier dat, zoals verwacht, de fout bij n=4 en n=8 kleiner is dan bij de platte driehoekjes, maar de fout is nog steeds niet kleiner dan 1 %.

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
4	365	722	8.53 %	0.24
8	1452	2896	1.83 %	0.21

Tabel 7.6: "Meshgrootte" vs. "Fout" voor gekromde driehoekjes

7.2.3 Fout bij kwadratische driehoekjes

Als laatste zullen we ook hier de kwadratische driehoekjes beschouwen.
Het blijkt dat hier de fout, bij een grof grid, nogal afhangt van de keuze van de Gauss-formule. We hebben daarom voor de zekerheid de Gauss-7 formule gebruikt en de fout die we daarbij krijgen is gegeven in tabel 7.7. We zien hier dat de fout bij n=4 kleiner is dan de fout bij platte en gekromde

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
2	356	176	16.12 %	0.41
4	1448	722	0.80 %	0.05

Tabel 7.7: "Meshgrootte" vs. "Fout" voor kwadratische driehoekjes

driehoekjes bij de vergelijkbare n=8. Verder valt op dat de ratio veel kleiner is en dat de fout dus sneller kleiner wordt.
Een oorzaak voor dit verschil zou kunnen zijn dat de Gauss-formules nog niet nauwkeurig genoeg zijn en we de verschillende driehoekjes dus pas kunnen vergelijken als we naar n=8 (bij kwadratische driehoekjes) kijken.
Het probleem hierbij is echter dat hiervoor een hele grote computer nodig is, omdat we te maken krijgen met een enorm aantal punten op twee schillen en dus is er heel veel geheugenruimte nodig om de systeem matrix op te slaan.

7.3 Drie-schillen model

Na het twee-schillen model, dat als tussenstap diende, gaan we naar het drie-schillen model (N=3) kijken. Dit is het model dat, wat betreft het bolschillen model, het dichtst bij de werkelijke beschrijving van het hoofd past.

7.3.1 Fout bij platte driehoekjes

Als eerste zullen we ook hier weer gaan kijken naar de platte driehoekjes en dat levert de onderstaande resultaten aangaande de fout.

De fout wordt op een identieke wijze uitgerekend als in voorgaande berekeningen van de fout en we krijgen dan de waarden uit tabel 7.8. We zien

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
1	33	54	58.45 %	
2	125	238	15.15 %	0.26
4	493	974	3.02 %	0.20
8	1966	3820	0.97 %	0.32

Tabel 7.8: "Meshgrootte" vs. "Fout" voor platte driehoekjes

dat ook hier n=1 en n=2 een onrealistische weergave van de werkelijkheid is en dat n=4 en n=8 al dicht in de buurt van de maximaal toelaatbare fout liggen.

7.3.2 Fout bij gekromde driehoekjes

Ook voor het drie-schillen model zullen we naar het geval met gekromde driehoekjes kijken. Dit levert de fout uit tabel 7.9. Hier zien we dat dit

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
4	493	974	3.35 %	0.05
8	1966	3820	0.50 %	0.15

Tabel 7.9: "Meshgrootte" vs. "Fout" voor gekromde driehoekjes

soort elementen een betere oplossing levert dan de platte driehoekjes, maar dat voor het geval n=4 dit nog niet het geval is.

7.3.3 Fout bij kwadratische driehoekjes

Tenslotte zullen we kwadratische elementen[1] beschouwen. Hier zien we dat

n:	Aantal punten:	Aantal driehoekjes:	Fout:	Ratio:
2	482	238	3.43 %	0.01
4	1954	974	0.20 %	0.06

Tabel 7.10: "Meshgrootte" vs. "Fout" voor kwadratische driehoekjes

deze methode voor n=8 beter is dan de andere methodes en dat de ratio kleiner is dan bij de andere gevallen. Dat wil dus zeggen dat de fout sneller afneemt bij gebruik van kwadratische elementen dan bij gebruik van één van de andere elementen.

7.4 Conclusie

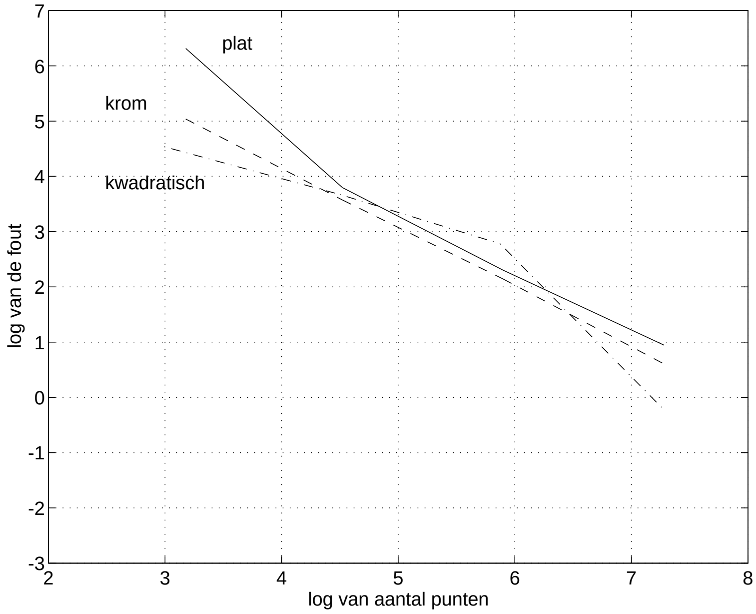
We hebben nu naar twee gevallen van het meer-schillen model gekeken. Hierbij was duidelijk te zien dat zowel bij het twee-schillen model als bij het drie-schillen model de kwadratische elementen het beste resultaat leverden, gevolgd door de gekromde elementen. Dit is ook duidelijk te zien in de figuren 7.1 en 7.2.
Wat hierbij opvalt is dat lijn behorende bij de fout van gekromde elementen in figuur 7.2 een rare knik aan het begin vertoont. Dit komt door het feit dat we gebruik hebben gemaakt van Gauss-formules, de invloed die de integratieformule op de fout heeft is hier nog te groot en leverde dus een fout die kleiner was dan de fout bij een fijner grid. Maar naarmate we verder gingen verfijnen is de invloed van de Gauss-formule op de fout heel klein geworden. We moeten hier nog wel opmerken dat ook bij de berekening van de fout bij de kwadratische elementen gebruik is gemaakt van Gauss-formules. Het zou dus kunnen zijn ook deze antwoorden een verkeerd beeld geven. Om die reden hebben we ook gebruik gemaakt van de nauwkeurigste Gauss-formule, namelijk de 7-punts formule.

Wat verder opvalt is dat de ratio bij de kwadratische elementen veel kleiner was dan bij de andere elementen, dat wil dus zeggen dat de fout sneller kleiner wordt als we een fijner grid kiezen.

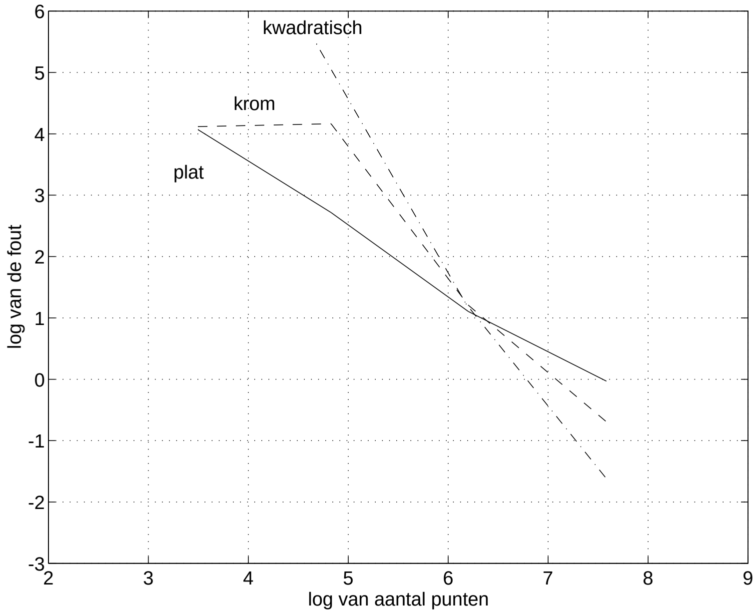
[1] Ook hier hebben we weer de Gauss-7 formule gebruikt voor de integratie

Wat verder opviel was dat we ook hier te maken kregen met het feit dat we met een beperkte geheugencapaciteit aan het werk waren. Bij het meer-schillen model moeten we immers meer schillen discretiseren waardoor de matrix een factor 4 bij het twee schillen model respectievelijk een factor 9 bij het drie schillen model groter wordt. En omdat we bij het model met een enkele schil al te maken kregen met een gebrek aan geheugenruimte gebeurt dat voor het meer-schillen model al in een veel vroeger stadium.
We zouden natuurlijk over kunnen stappen naar een computer met meer geheugenruimte, maar dan krijgen we toch te maken met het feit dat het niet meer praktisch is doordat het maken van de matrix en voornamelijk zijn LU-decompositie veel te veel tijd in beslag gaat nemen.

De conclusie die we uit het onderzoek naar het meer-schillen model kunnen trekken is dat we het best gebruik kunnen maken van kwadratische elementen, maar dat we al snel te maken krijgen met een gebrek aan geheugencapaciteit. Dat de kwadratische elementen beste keuze is is tegenstelling met de resultaten verkregen uit het model met één schil, maar daar hadden we ook te maken met een onverwacht resultaat wat we nog niet konden verklaren.



Figuur 7.1: fouten 2 schillen model



Figuur 7.2: fouten 3 schillen model

Hoofdstuk 8

Efficienter gebruik van geheugenruimte

In dit hoofdstuk zullen we verschillende mogelijkheden voor verbetering van het oplossen van het probleem bespreken. Daarbij zullen we als eerste kijken wat er verbeterd kan worden aan het oplossen van het bolschillen model en vervolgens zullen we alvast kijken naar het realistische geval, welke we uiteindelijk moeten oplossen.
We hebben nu het bolschillen model onderzocht en kunnen concluderen dat we voor een nauwkeurige oplossing te maken krijgen met een groot aantal knooppunten, waardoor de systeem matrix ook enorm groot wordt.
Doordat we met zo'n grote matrix gaan werken kost het heel wat tijd om deze te maken en er een LU decompositie van te maken.
Verder gaat het grootste gedeelte van het geheugen verloren aan het opslaan van de matrix. Hierdoor zitten we heel snel aan het maximum van onze geheugen capaciteit en zullen we genoegen moeten nemen met een kleiner aantal punten waardoor we een grotere fout krijgen.
Willen we de fout dus reduceren dan moeten we overstappen naar een grotere computer of iets verzinnen waardoor de matrix minder geheugen in beslag neemt. We zullen dus goed naar de matrix moeten kijken of daar nog enige winst valt te halen.

8.1 Iteratieve na-verfijning

Als we naar het programma kijken dan zien we dat we bij het oplossen van het stelsel nog iteratief gaan naverfijnen en daarvoor gebruiken we de oorspronkelijke matrix.

We hebben de matrix dus eerst ontbonden in L en U, en daarna gebruiken we de matrix nogmaals om na te verfijnen. Dit betekent dat we zowel de matrix als zijn LU decompositie moeten opslaan.
We kunnen ons nu afvragen of dit naverfijnen wel nodig is. Als we het achterwege kunnen laten zal het geheugen verbruik aanzienlijk verminderd kunnen worden. De geheugen ruimte die dan vrijkomt kunnen we goed gebruiken voor het verfijnen van het grid en dus het verkleinen van de fout.
Beschouwen we nu het geval dat we zowel met als zonder naverfijning te maken hebben dan krijgen we de resultaten uit tabel 8.1. We hebben hier

Naverfijning:	Fout:
met	2.80 %
zonder	2.80 %

Tabel 8.1: Invloed naverfijning

gekeken naar de invloed van het naverfijnen bij het geval we met platte driehoekjes te maken hebben bij het model met één schil en n=4.
We zien hier dat het voor dit geval helemaal niets uitmaakt of we naverfijnen of niet. Het zou dus verstandiger zijn om dit dan achterwege te laten en het geheugen te gebruiken voor het verbeteren van de fout.

We weten nu dat we het bij het model met één schil niet nodig is om naverfijning toe te passen, maar we zullen ook nog moeten onderzoeken of dit ook voor het meer schillen model geldt. Het is tenslotte het streven om het drie-schillen model zo goed mogelijk op te lossen.
De resultaten die we onder dezelfde omstandigheden krijgen voor het drieschillen model, dus met platte driehoekjes en n=4, zijn als volgt:

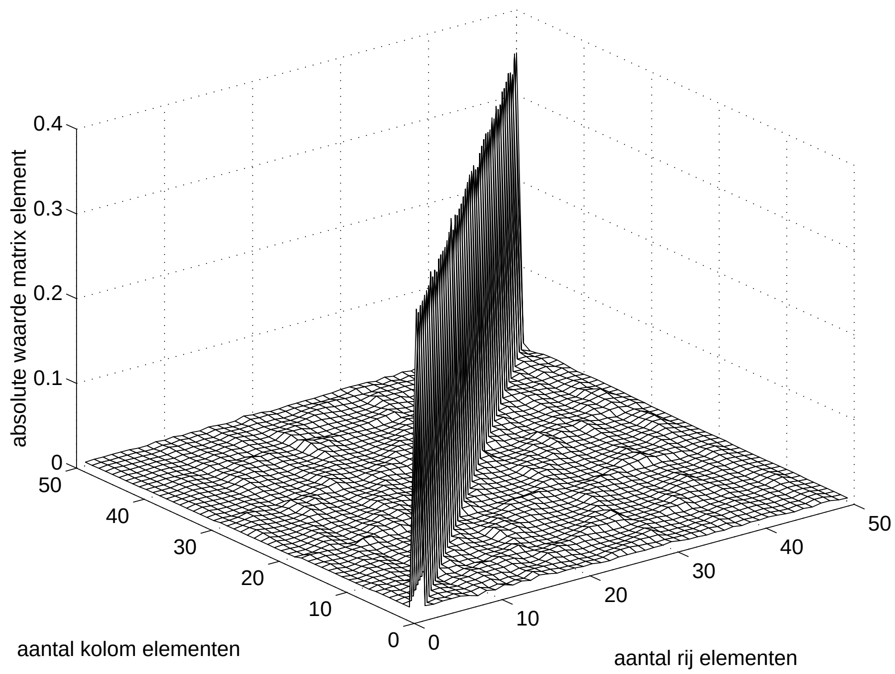
Naverfijning:	Fout:
met	3.02 %
zonder	3.02 %

Tabel 8.2: Invloed naverfijning 3 schillen model

Ook hier zien we dat we met en zonder iteratieve na-verfijning dezelfde fout krijgen. Het is dus niet nodig om na te verfijnen en daarom verstandiger om dit dan ook achterwege te laten.

8.2 Matrix elementen

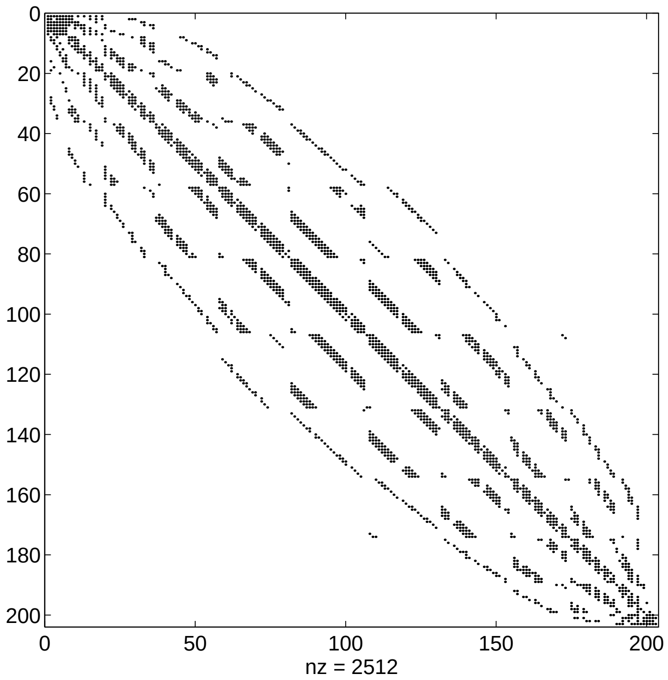
Wat we ons ook kunnen afvragen is hoe groot de matrix elementen zijn en of alle elementen wel opgeslagen dienen te worden. Als we te maken hebben met een ijle matrix dan hoeven we de nullen immers niet op te slaan.
We zullen dus moeten kijken of er elementen voorkomen die klein zijn en waarvan de invloed op de potentiaal klein is ten opzichte van de andere elementen. Komen deze elementen voor dan kunnen we ze gelijk aan nul nemen, omdat hun invloed op de potentiaal heel klein is en we toch al werken met een kleine fout.
Zetten we nu de absolute waarde van de matrix elementen tegen de positie in de matrix uit, dan krijgen we voor het model met één schil het 3D figuur 8.1.



Figuur 8.1: Matrix bij 1 schil model

We hebben hier gekeken naar geval we met platte driehoekjes en n=2 te maken hebben. Voor de matrices die we krijgen door een andere waarde

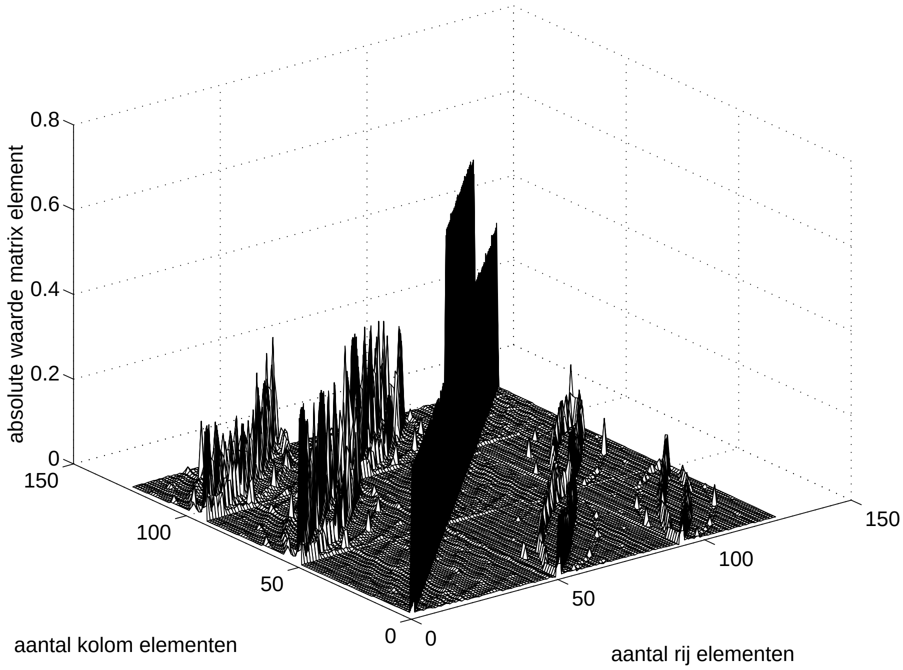
van n te kiezen of een ander soort driehoekje te nemen krijgen we het zelfde patroon.
Wat we hier zien is dat sommige elementen meer invloed uitoefenen dan andere. Het is duidelijk te zien dat de hoofddiagonaal de meeste invloed uitoefent, maar is de invloed van de andere elementen van zo'n aard dat we ze mogen verwaarlozen?
Als we nu aannemen dat we al de elementen die kleiner zijn dan 1 % van hoofddiagonaal mogen verwaarlozen, dan krijgen we voor n=4, figuur 8.2, waarbij we alleen de elementen die groter zijn dan 1 % van de hoofddiagonaal hebben afgedrukt.



Figuur 8.2: Elementen kleiner dan 1% van een diagonaal element

Wat we in dit figuur zien is dat een groot aantal van de elementen kleiner is dan 1 % van de waarde van een element op de diagonaal (waarde 0.45). De vraag is alleen nog of dit percentage klein genoeg is en of de fout dan niet te veel afwijkt? Dit zal dus nog onderzocht moeten worden. Vertoont de fout maar een kleine afwijking, dan kunnen we dus gebruik gaan maken van de bandmatrix en dan zal dit ook weer in geheugen ruimte schelen.

We zullen ook hier weer apart moeten gaan kijken naar het meer-schillen model om een uitspraak te kunnen doen of het eventueel zinnig is om de methode toe te passen. Zetten we daarom de matrix van het drie-schillen model tegen zijn positie, in de matrix, uit dan resulteert dat in figuur 8.3.

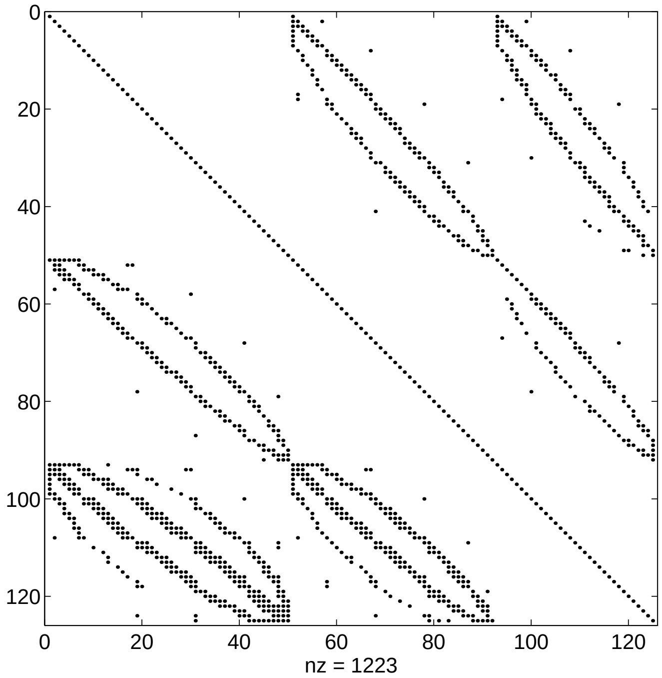


Figuur 8.3: Matrix bij 3 schillen model

Ook hier valt weer op dat de elementen op de hoofddiagonaal de meeste invloed uitoefenen, maar wat nog meer opvalt is dat er nog vier banden zijn die er boven uitsteken.
Laten we al de elementen die kleiner zijn dan 5 % van de hoofddiagonaal, dus

$$|a_{ij}| = 0.05 * |a_{ii}| \qquad\qquad i,j = 1,2,...,n \qquad (8.1)$$

weg dan resulteert dat in figuur 8.4. Hier zien we dat we met meer elementen rekening moeten houden, maar er ontstaat een duidelijke bandmatrix.



Figuur 8.4: Elementen kleiner dan 5% van de diagonaal

Ook voor het drie-schillen model geldt weer dat het bepalen van het aantal elementen dat eventueel verwaarloosd kan worden apart onderzocht dient te worden.

8.2.1 Fout bij het op nul zetten matrix elementen

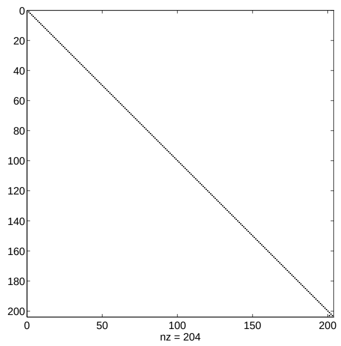
We bekijken nu hoe groot de fout is als we de kleinste elementen van de matrix op nul zetten. Als test hebben we gekeken wat er gebeurt als we, bij 1 schil, de elementen die kleiner zijn dan 1 % van 0.45 (ongeveer de waarde van de diagonaal elementen) gelijk aan nul nemen.
In onderstaande tabel zien we de fout die we dan krijgen: We zien hier dat een groot deel van de elementen op nul is gezet, maar dat de fout ook aardig is toegenomen. De fout is echter niet in die mate toegenomen dat het niet de moeite waard zou zijn om naar enkele andere percentages te kijken.

n = 4	Aantal elementen	Nullen	perc. nul	Fout:
0 %	41209	0	0 %	2.80 %
1 %	41209	38742	94.01 %	13.33 %

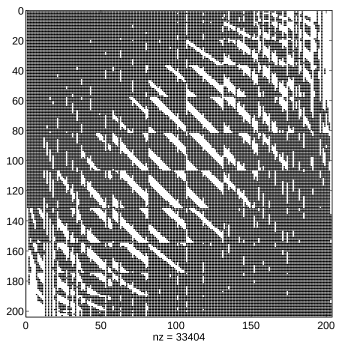
Tabel 8.3: Invloed van het nul nemen van elementen

We hadden al gezien dat bij 1 % van de hoofddiagonaal we een fout van 13.33 % kregen, dat was nog een redelijk groot verschil met de fout die we kregen door de hele matrix te gebruiken. We kunnen ons nu afvragen of de fout afneemt naarmate we een kleiner of groter percentage van de hoofddiagonaal (waarde is ongeveer 0.45) nemen.

Nemen we 2 % respectievelijk 0.3 % dan levert dat de figuren 8.5 en 8.6. We zien hier dat in figuur 8.5 we nagenoeg alleen de hoofddiagonaal overhouden en in figuur 8.6 we de structuur van figuur 8.2 bijna kwijt zijn.



Figuur 8.5: 2 % van 0.45



Figuur 8.6: 0.3 % van 0.45

Kijken we nu naar de fout dan krijgen we tabel 8.4.

n = 4	Aantal elementen	Nullen	perc. nul	Fout:
2.0 %	41209	41005	99.50 %	18.37 %
0.3 %	41209	7788	18.90 %	10.36 %

Tabel 8.4: Invloed van het nul nemen van elementen

We zien dat de als we meer elementen nul nemen, we een groter fout krijgen. Nemen we minder elementen nul dan wordt de fout, zoals verwacht, kleiner, maar nog niet in de mate die we graag willen.

Om de fout omlaag te krijgen moeten we dus minder elementen op nul zetten. Nemen we daarom de percentages 0.1 % en 0.2 % dan krijgen we de fout gegeven in tabel 8.5.

n = 4	Aantal elementen	Nullen	perc. nul	Fout:
0.1 %	41209	0	0 %	2.80 %
0.2 %	41209	79	0.19 %	2.81 %

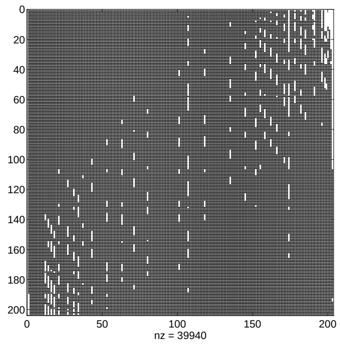
Tabel 8.5: Invloed van het nul nemen van elementen

We zien hier dat de matrices nagenoeg helemaal gevuld zijn. Voor het geval het percentage 0.1 % is het zelfs zo dat de matrix geheel gevuld is. In het andere geval zien we dat de matrix nog niet helemaal gevuld is, maar er zit nog weinig tot geen structuur in.
Verder zien we dat de fout zo goed als gelijk is aan de fout bij een geheel gevulde matrix, maar dat het percentage elementen dat we op nul zetten heel erg klein is.

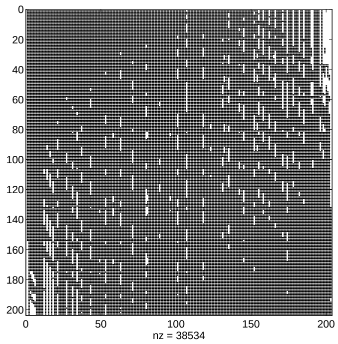
We zullen dus naar een percentage moeten kijken dat tussen 0.2 % en 0.3 % ligt om een matrix te krijgen waarbij een groot aantal elementen op nul gezet zijn en waarbij de fout niet veel groter is dan de fout die we krijgen bij een volle matrix. Nemen we daarom de percentage 0.25 % en 0.275 % dan krijgen we de figuren 8.7 en 8.8.
We zien hier dat er weinig tot geen structuur in zit, maar als we naar de fout kijken, in onderstaande tabel, dan zien we dat deze niet zo gek veel afwijkt van de fout met een volle matrix.

n = 4	Aantal elementen	Nullen	perc. nul	Fout:
0.25 %	41209	1270	3.08 %	3.55 %
0.275 %	41209	2675	6.49 %	4.55 %

Tabel 8.6: Invloed van het nul nemen van elementen



Figuur 8.7: 0.25 % van 0.45



Figuur 8.8: 0.275 % van 0.45

Wat hier wel weer opvalt is dat het percentage nullen erg laag is, vandaar ook dat de fout zo goed overeenkomt.

Het is wel gebleken dat het niet zo eenvoudig is om een groot aantal elementen op nul te zetten en een kleine afwijking van de fout te krijgen. Hebben we immers een kleine afwijking van de fout dan is er maar een klein aantal elementen op nul gezet en is de mooie structuur verdwenen. Zetten we een groot aantal elementen op nul dan vertoont de fout al een grote afwijking. Daar we streven naar een zo klein mogelijke afwijking van de fout kunnen we concluderen dat het geen zin heeft om de kleinste matrix elementen op nul te zetten, omdat we dan toch met een nagenoeg volle matrix moeten werken.

8.3 Symmetrie matrix

Hebben we te maken met een symmetrische matrix dan kunnen we de helft aan geheugenruimte besparen. De vraag is alleen of dat hier het geval is? Als we naar figuur 8.4 kijken dan hebben we ons antwoord al, het is duidelijk te zien dat de matrix niet symmetrisch is. Dit komt onder andere door het feit dat de view-points op verschillende schillen kunnen zitten die dan ook verschillende geleidingsfactoren boven en onder zich hebben.
Als het view-point en het punt dat daarop invloed uitoefent, ver van elkaar afliggen dan wordt de matrix daar bij benadering symmetrisch, maar dat geldt dus niet voor alle combinaties van view-points en punten die invloed uitoefenen.

Daar we graag willen werken met een symmetrische matrix moeten we kijken of het mogelijk is om de matrix symmetrisch te maken. Het blijkt dat dit mogelijk is door uitgaande van de randelement vergelijking 3.14 en daar vergelijking 3.15 in te vullen. We hebben nu eindig veel onbekenden en voordat we er een eindig stelsel van gaan maken, zoals we tot nu toe hebben gedaan, vermenigvuldigen we de vergelijking links en rechts met een gewichtsfunctie, gelijk aan de basisfunctie, en integreren we over het view-point. Zo krijgen we een eindig stelsel met een symmetrische matrix. We passen nu dus de methode van Galerkin in plaats van de collocatiemethode toe.
We moeten er nog wel rekening mee houden dat we nog het probleem van de geleidingsfactoren hebben, maar dit is ook op te lossen door links en rechts te vermenigvuldigen met $(\sigma^{-} - \sigma^{+})$, de geleidingsfactoren boven en onder, behorende bij de schil waar het view-point op ligt.

Hoofdstuk 9

Een realistisch hoofdmodel

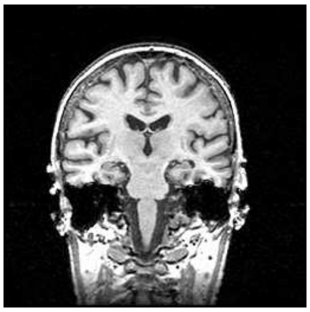
In deze paragraaf zullen we gaan kijken wat voor problemen we tegen kunnen komen willen we het realistisch hoofdmodelin de praktijk toepassen.

9.1 Gegevens

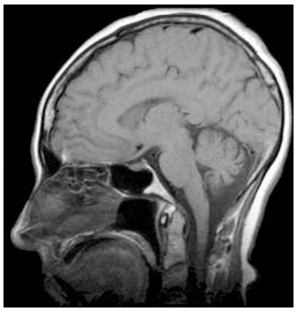
Voordat we kunnen beginnen zullen we eerst moeten weten met welke gegevens we moeten gaan werken. Zo zouden we graag willen weten hoe we aan de discretisatie punten komen en wat we er zoal mee kunnen doen. In deze paragraaf zullen we deze vragen proberen te beantwoorden.
De eerste vraag die we ons kunnen stellen is waar deze punten vandaan komen? Het antwoord op deze vraag is moeilijker te geven dan in de situatie waarin de geometrische beschrijving van een oppervlak bekend is. In dat geval zouden we de punten kunnen bepalen door ze op, bijvoorbeeld, een regelmatige afstand van elkaar te kiezen.
In ons geval ligt dat anders, omdat we niet precies weten hoe de geometrie van de verschillende volumes van het hoofd is beschreven. Voor het oplossen van dit probleem moet er eerst een plaatje van het hoofd gemaakt worden. Dit wordt gedaan met behulp van een zogenaamde MRI (Magnetic Resonance Imaging). Daarbij wordt het hoofd in verschillende plakken verdeelt, die zowel horizontaal als verticaal genomen kunnen worden.

Een voorbeeld daarvan zijn de onderstaande figuren[1] waarbij we een plak uit het hoofd genomen hebben en vanuit drie verschillende posities hebben gekeken.

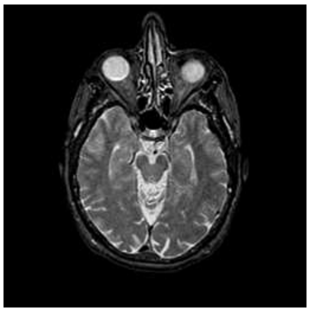
[1]Deze 3 aanzichten zijn niet van hetzelfde hoofd, maar geven wel een idee hoe een MRI eruit ziet.



Figuur 9.1: MRI vooraanzicht

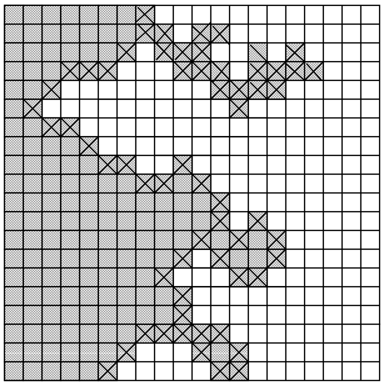


Figuur 9.2: MRI zijaanzicht

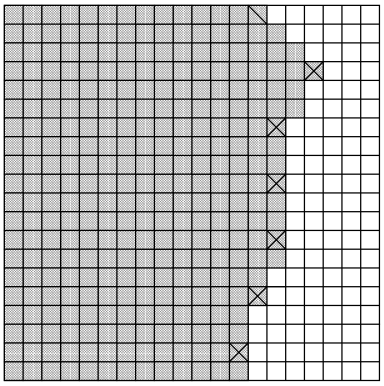


Figuur 9.3: MRI bovenaanzicht

We hebben nu telkens gezien hoe één plak eruit ziet, nemen we alle plakken samen dan krijgen we de MRI van het hoofd.
We hebben dan dus een X aantal plakken die onderverdeeld zijn in pixels van verschillende grijswaarden. Deze grijswaarden worden dan op een zodanige manier bijelkaar genomen dat er een onderscheid ontstaat tussen de verschillende compartimenten. Is dat gedaan dan worden deze overgangen omgezet in zwart-wit overgangen zodat er een nog beter onderscheid ontstaat tussen de verschillende compartimenten.
Nu de verschillende compartimenten in kaart gebracht zijn worden de knooppunten geplaatst. Dit gebeurt op de plaatsen waar een zwarte pixel naast een witte pixel zit en we krijgen dan het fijnst mogelijke grid (zie figuur 9.4).



Figuur 9.4: pixelmap MRI + knooppunten



Figuur 9.5: pixelmap grof rooster + knooppunten

Willen we liever werken met een grover grid dan wordt eerst het scheidings-

oppervlak afgevlakt en krijgen we te maken met een gladder oppervlak (zie figuur 9.5). Hierna wordt ook bij een zwart-wit overgang een knooppunt geplaatst, alleen nu niet meer bij iedere overgang, maar bijvoorbeeld om de andere overgang.
Uiteindelijk krijgen we een set punten waarvoor geldt dat de punten behorende bij het grove rooster géén deelverzameling hoeft te zijn van de punten verkregen uit de MRI, het fijnste rooster.

De software, die alreeds aanwezig is, zorgt er vervolgens voor dat bij een gegeven set punten een driehoeksbelegging wordt gemaakt. Verder zorgt de software ervoor dat de normaalrichting wordt gegeven.

Wat valt er nu eigenlijk met deze gegevens te doen? Als we willen werken met platte driehoekjes zijn er géén problemen te verwachten, omdat voor de generatie slechts drie punten nodig zijn. Willen we echter naar gekromde driehoekjes dan wordt dat moeilijker met deze gegevens. We gebruiken voor het construeren van gekromde driehoekjes de geometrie van het oppervlak en dat is nou net iets wat we hier niet weten.
Verder hebben we in hoofdstuk 5 en 6 de geometrie van het oppervlak gebruikt voor het bepalen van tussenpunten (bij de gekromde driehoekjes door middel van subdriehoekjes en bij kwadratische driehoekjes).
Het grote probleem waar we nu dus mee te maken krijgen is hoe we aan de tussenpunten komen als de geometrische beschrijving niet gegeven is? We kunnen ook geen gebruik maken van de gegevens uit het fijne rooster voor het oplossen van het grove rooster, omdat het grove rooster geen deelverzameling van het fijne rooster is.
We zullen dus een andere methode moeten vinden.

9.2 Mogelijke oplosmethodes

In deze paragraaf zullen we in het kort enkele oplosmethodes beschrijven die eventueel te gebruiken zijn voor het oplossen van het probleem

9.2.1 Methode 1

De eerste methode is door gebruik te maken van bolfuncties.
Hierbij gebruiken we het gemiddelde van de punten als oorsprong.
De bolcoördinaten van de discretisatie punten, vanuit de oorsprong gezien, kunnen we gebruiken om de onbekende constante in de bolfunctie te bepalen.
Hierdoor hebben we een bolfunctie gekregen waarbij we de tussenpunten

kunnen bepalen door θ en ϕ te nemen die halverwege twee discretisatie punten liggen. (Voor meer informatie zie bijlage E.)

Het grote probleem bij deze methode is dat hij hoogst waarschijnlijk alleen werkt als ons gediscretiseerde oppervlak sterk op een bol lijkt.
De vraag die we ons dan kunnen stellen is of dit voor het menselijke hoofd geldt? Dus is ons hoofd bolvormig genoeg om deze methode te kunnen gebruiken en een goede oplossing geeft?

9.2.2 Methode 2

De tweede methode is door gebruik te maken van interpolatie technieken en splines. Dat wil zeggen dat we een polynoom zoeken die door de discretisatie punten gaat.
Plakken we deze polynomen aan elkaar vast en voldoen we aan zekere gladheidseisen in de discretisatie punten dan krijgen we zogenaamde splines (zie bijlage E).

9.2.3 Methode 3

De derde methode is door gebruik te maken van zogenaamde driehoekige Coons-patches.
Hierbij wordt van een driehoek waarvan de edges gegeven zijn een fijnere driehoeks verdeling gemaakt (zie bijlage E).

Het is wel duidelijk dat het probleem hier is om de drie edges te construeren. Een mogelijke oplossing daarvoor zou kunnen zijn om een polynoom proberen te maken dat door meerdere discretisatie punten gaat. Maken we dan drie van die polynomen dan kunnen we ze gebruiken om een driehoekige Coons-patch te construeren.

9.2.4 Methode 4

De vierde en laatste methode is door gebruik te maken van de punten die verkregen zijn na het beleggen met punten van de gegevens uit de MRI.
We nemen dan als knooppunten de punten uit ons grove rooster en gaan een polynoom maken die door de knooppunten gaat en een zo goed mogelijke benadering is van de punten van het fijnste rooster die binnen een bepaald oppervlak liggen.

Een mogelijk probleem dat hier kan ontstaan is een tekort aan geheugencapaciteit. We moeten immers nu ook gebruik maken van de grote hoeveelheid punten verkregen uit de MRI.

9.2.5 Conclusie

Uit al deze problemen kunnen we concluderen dat het niet zo eenvoudig is om aan extra tussenpunten te komen om een betere benadering van het oppervlak te krijgen.
Wat wel mogelijkheid zou kunnen zijn om gebruik te maken van de technieken die in de auto- en luchtvaart industrie gebruikt worden, namelijk door het oppervlak interactief te beschrijven.
Het is dan misschien mogelijk om dit één keer te doen en vervolgens deze geometrische beschrijving te gebruiken als standaard hoofdmodel.

Hoofdstuk 10

Conclusies en aanbevelingen

10.1 Samenvatting

Meerkanaals EEG/MEG signalen kunnen gebruikt worden om onderliggende hersenactiviteit te localiseren. Daartoe is een mathematisch model opgesteld, dat een beschrijving geeft van de potentiaal en voor een bolschillen model analytisch oplosbaar is. Voor een realistisch model is dit echter niet meer het geval en moeten we overstappen naar een numerieke methode. De methode die we gebruikt hebben is de Boundary Element Methode. Hierbij worden de scheidingsoppervlakken tussen de verschillende compartimenten gediscretiseerd en de potentiaal aldaar uitgerekend.

Tot nu toe wordt er meestal gebruik gemaakt van het bolschillen model, om de bron in het hoofd te lokaliseren, omdat dit een snel algoritme levert maar met een beperkte nauwkeurigheid. Wegens deze beperkte nauwkeurigheid wil men graag overstappen naar een realistisch hoofdmodel. De vraag is welke variant van de Boundary Element Methode (aantal en soort elementen) een voldoende nauwkeurige oplossing geeft zonder dat de rekentijd en het geheugengebruik onacceptabel groot worden. Het probleem hier is dat we niets hebben om de resultaten mee te vergelijken. Om de nauwkeurigheid te bepalen hebben we daarom de numerieke methode vergeleken met de analytische oplossing van het bolschillen model.

10.2 Conclusie

Wat we uit het onderzoek kunnen concluderen is dat het nogal wat uitmaakt wat voor model we gebruiken. Nemen we het model met 1 schil dan kunnen we het best gebruik maken van de gekromde elementen. Dit is enigzins dubieus, want we hadden verwacht dat bij een heel fijn grid de kwadratische elementen een kleinere fout zouden leveren dan de gekromde elementen. Nemen we echter een model met meerdere schillen, wat een realistischer weergave is, dan genieten de kwadratische elementen de voorkeur.

Wat we ons wel moeten realiseren is dat het nog niet zo eenvoudig is om kwadratische elementen toe te passen op het realistische model. Het probleem is dat we niet zo eenvoudig aan de tussenpunten kunnen komen. Daar is nog verder onderzoek voor nodig.

Het grote probleem bij het onderzoek was dat we, voor een nauwkeurig antwoord, al snel in problemen komen met een gebrek aan geheugenruimte.
Daar we met een zo klein mogelijke fout willen werken is het aan te raden om daar iets aan te doen. Omdat de de opslag van de matrix het meeste geheugen in beslag neemt, hebben we onze aandacht daarop gericht.
Het blijkt nu dat er wel degelijk op geheugenruimte bespaard kan worden. Zo wordt er nu onnodig gebruik gemaakt van iteratieve na-verfijning. Dat wil zeggen dat we maar 1 keer de geheugenruimte die de matrix in beslag neemt nodig hebben en niet 2 keer. Dit is al een aanzienlijke besparing.
Verder hebben we onderzocht of het nut heeft om de kleinste elementen van de matrix nul op te zetten. Dit blijkt niet zonder meer het geval te zijn. Het is in ieder geval wel aan te raden om een andere oplossing voor het geheugenruimte probleem te zoeken, omdat we deze ruimte goed kunnen gebruiken voor het verder verfijnen van het grid, waardoor we een kleinere fout krijgen. Een goede mogelijkheid zou kunnen zijn om de matrix symmetrisch te maken waardoor we maar de helft van het aantal matrix elementen hoeven op te slaan. Deze besparing van geheugenruimte kunnen we goed gebruiken voor het verbeteren van de fout. Hebben we een kleine fout voor het voorwaartse probleem dan krijgen we ook een nauwkeuriger oplossing van het inverse probleem.

10.3 Punten die verder onderzocht dienen te worden

In deze paragraaf hebben we enkele punten opgesomd, die de moeite waard kunnen zijn om ze verder te onderzoeken. Daarbij hebben we onderscheid gemaakt tussen het bolschillen model en het realistische hoofdmodel.

10.3.1 Bolschillen model

Er zijn hier nog enkele vragen over die nog verder onderzocht dienen te worden:

- Het probleem met de geheugenruimte, met name de opslag van de matrix, omdat deze de meeste geheugenruimte inneemt.
- Het gebruik van een andere oplosmethode in plaats van LU-decompositie. Bijvoorbeeld de zogenaamde fast multipole methode (zie [8]).
- Het gebruik van de Eindige Elementen Methode in plaats van de Boundary Element Methode, omdat de BEM zijn voordeel verliest naarmate het aantal schillen groter wordt. Misschien is de EEM dan wel een goed alternatief.
- Uitwerken en toepassen van een oplossing met een symmetrische matrix, door de collocatiemethode te vervangen door de methode van Galerkin. De nauwkeurigheid moet dan wel weer opnieuw gecontroleerd worden.

De vraag die het meeste aandacht verdient is of er iets aan het tekort aan geheugenruimte valt te doen, omdat dit een beperking is op de nauwkeurigheid.

10.3.2 Realistische hoofdmodel

De antwoorden van onderzoeksvragen voor het bolschillen zijn ook van belang voor het realistische hoofdmodel. Daardoor blijven de volgende vragen over:

- Hoe komen we aan de tussenpunten die nodig zijn om kwadratische elementen te gebruiken?
- Is er veel verschil tussen de antwoorden verkregen met de kwadratische elementen met de antwoorden verkregen met platte elementen?
- Is het de moeite waard om kwadratische elementen te gebruiken?

Hebben we de antwoorden op deze vragen dan zouden we een keuze kunnen maken tussen de verschillende elementen.

Bibliografie

[1] J.van Kan A.Segal. *Numerieke methoden voor partiële differentiaalvergelijkingen.* Delftse Uitgevers Maatschappij, Delft, 1993.

[2] Prof.dr.H.G.Meijer/dr.M.G.de Bruin. *Bijzondere Functies.* Technische Universiteit Delft, collegediktaat a56.

[3] W. Buijze. *Inleiding Elektriciteit.* Delftse Uitgevers Maatschappij, Delft, 1990.

[4] Jan C. de Munck. *A mathematical and physical interpretation of the electromagnetic field of the brain.* PhD thesis, University of Amsterdam, 1989.

[5] G.Farin. *Curves and Surfaces for Computer Aided Design: A practical guide*, volume 4 of *Computer Science and Scientific Computing.* Academic Press, Inc., San Diego, 1997.

[6] Gilbert Strang George J.Fix. *An Analysis of the Finite Element Method.* Prentice Hall inc., 1973.

[7] J.W.Boerstoel. *Computational Fluid Dynamics.* Delft University of Technology, 1995, collegediktaat a110B.

[8] J. Rahola. Solution of dense systems of linear equations in the discrete-dipole approximation. *SIAM J. Sci. Comput.*, 17:78–89, 1996.

[9] Jukka Sarvas. Basic mathematical and electromagnetic concepts of the biomagnetic inverse problem. *Phys.Med.Biol.*, 32(1):11–22, 1987.

[10] R. van Gent. Een onderzoek aan de boundary element methode voor nauwkeurige localisatie van hersenactiviteit, Scriptie, 1997. TU Delft.

Bijlage A

Bijzondere functies

A.1 *Legendre* polynomen

Polynomen van *Legendre* zijn polynomen die oplossing zijn van de onderstaande differentiaalvergelijking:

$$(1-x^2)y'' - 2xy' + n(n+1)y = 0 \tag{A.1}$$

A.1.1 Polynomen

De polynomen worden als volgt gedefinieerd:

$$P_n(x) = \frac{(-1)^n}{2^n n!}\left(\frac{d}{dx}\right)^n (1-x)^n(1+x)^n \tag{A.2}$$

met

$$P_n = P_{-n-1} \quad , \quad P_n(1) = 1 \quad , \quad P_n(-1) = (-1)^n \tag{A.3}$$

A.1.2 Norm

$$\|P_n\|^2 = \int_{-1}^{1} P_n^2(x)dx = \frac{(-1)^n}{2^n n!}\int_{-1}^{1} P_n(x)D^n(1-x^2)^n dx = \frac{2}{2n+1} \tag{A.4}$$

A.1.3 Geassocieerde *Legendre* functie

De geassocieerde functie is:

$$P_n^m(x) = (1-x^2)^{\frac{m}{2}} D^m P_n(x) \tag{A.5}$$

met

$$D = \frac{d}{dx} \quad , \quad P_n^m(x) \equiv 0 \qquad m > n \tag{A.6}$$

(Voor meer over Legendre functies zie [2].)

Bijlage B

Welke Gauss formule ?

In deze bijlage beschouwen de verschillende Gauss integratie formules zoals gegeven in paragraaf 5.1.1 en kijken welke we het beste kunnen gebruiken.

B.1 Fout

Als we ons probleem oplossen en we gebruiken telkens een andere Gauss-formule dan krijgen we de volgende twee tabellen.
Voor het geval we platte driehoekjes gebruiken krijgen we de volgende tabel:

Formule:	n=4	n=8
Analytisch	2.80 %	0.77 %
Gauss 1	3.38 %	0.90 %
Gauss 3	2.81 %	0.77 %
Gauss 4	2.81 %	0.77 %
Gauss 6	2.80 %	0.77 %
Gauss 7	2.80 %	0.76 %

Tabel B.1: Gauss-formules voor platte driehoekjes

en voor we het geval we kromme driehoekjes gebruiken krijgen we tabel B.2. In beide gevallen zien we dat de 3-punts Gauss-formule al een goede benadering geeft. De fout wijkt in dat geval niet veel af van de veel nauwkeuriger 7-punts Gauss-formule. We nemen daarom aan dat de 3-punts Gauss-formule

Formule:	n=4	n=8
Analytisch		
Gauss 1	2.49 %	0.63 %
Gauss 3	1.96 %	0.45 %
Gauss 4	1.95 %	0.45 %
Gauss 6	1.96 %	0.45 %
Gauss 7	1.96 %	0.45 %

Tabel B.2: Gauss-formules voor gekromde driehoekjes

voor ons probleem een goede integratie formule is. Deze zullen we dan ook voor zowel platte als gekromde driehoekjes gebruiken.

B.2 Tijd

In deze paragraaf zullen we kijken hoe het zit met de tijd die nodig is voor de opbouw van de systeem matrix A bij de verschillende Gauss-formules. We hebben ook hier weer als eerste naar het geval van de platte driehoekjes gekeken en dat levert de volgende resultaten:

Formule:	n=4	Opbouw A	Totaal	n=8	Opbouw A	Totaal
Analytisch		1.49	1.92		23.35	108.43
Gauss 1		0.66	1.26		12.46	104.30
Gauss 3		1.37	1.87		23.73	128.47
Gauss 4		2.20	2.74		31.69	118.81
Gauss 6		2.59	3.08		42.78	129.95
Gauss 7		3.52	4.01		49.05	148.19

Tabel B.3: Tijden (sec) bij de Gauss-formules; platte driehoekjes

Wat hier opvalt is dat bij een hogere orde Gauss-formule de tijd voor de opbouw van de matrix toeneemt en wel zodanig dat de opbouw van de matrix bij de 6-punts Gauss-formule ongeveer twee keer zolang duurt als de

3-punts Gauss-formule.
Verder zien we dat de tijd voor de opbouw van de matrix met de 3-punts Gauss-formule overeenkomt met het analytisch uitrekenen van de integralen.

Beschouwen we nu de tijden voor het geval we met gekromde driehoekjes te maken hebben dan levert dat

Formule:	n=4	Opbouw A	Totaal	n=8	Opbouw A	Totaal
Analytisch						
Gauss 1		1.15	1.70		16.09	113.81
Gauss 3		2.74	3.30		44.38	131.05
Gauss 4		2.86	3.30		45.15	123.92
Gauss 6		4.01	4.51		64.10	143.03
Gauss 7		4.61	5.10		73.05	162.80

Tabel B.4: Tijden (sec) bij de Gauss-formules; gekromde driehoekjes

Ook over deze tabel kunnen we weer zeggen dat bij hogere orde Gauss-formules de tijd die nodig is voor de opbouw van de matrix het grootst is.

B.3 Gauss-formule voor kwadratische driehoekjes

Bij het gebruik van kwadratische driehoekje zullen we opnieuw moeten evalueren welke Gauss-formule we gaan gebruiken. Daarvoor hebben we weer naar de fout bij de verschillende formules gekeken en dat leverde de resultaten uit tabel B.5.
Ook hier zien we weer dat de Gauss1-formule niet voldoet, maar dat de Gauss3-formule al wel aardig voldoet en daarom zullen we deze formule ook gebruiken.

Formule:	n=2	n=4
Analytisch		
Gauss 1	10.10 %	1.04 %
Gauss 3	6.02 %	0.54 %
Gauss 4	6.08 %	0.54 %
Gauss 6	5.96 %	0.54 %
Gauss 7	5.96 %	0.54 %

Tabel B.5: Gauss-formules voor kwadratische driehoekjes

Bijlage C

Dipool positie, richting en sterkte

In het oorspronkelijke probleem wilden we graag weten wat, bij gegeven potentiaal (uit metingen), de locatie en sterkte van de bron, die voor de activiteit zorgt, wordt. Daarbij moeten we er rekening mee houden dat de bron overal in de hersenen kan zitten. Dit betekent ook dat we voor het voorwaartse probleem dat we oplossen, meerdere mogelijkheden moeten beschouwen, voor wat betreft de locatie, richting en sterkte van de dipool (de bron).
In deze bijlage zullen we enkele posities beschouwen en er onze conclusies uittrekken.

C.1 Platte driehoekjes

C.1.1 Verschillende posities (afstand gelijk) ,richting en sterkte vast

Als eerste beschouwen we enkele dipool posities die op een zelfde afstand van het middelpunt liggen. We gebruiken daarvoor de methode van de platte driehoekjes met n=4 en de vaste dipool richting (1,-1,2). De resultaten die dan verkregen worden zijn weergegeven in tabel C.1.
We zien hier dat de fout redelijk stabiel is, met de uiterste waarden, bij deze metingen, in het punt (5,5,5) en op de negatieve z-as.
We kunnen uit deze metingen concluderen dat het haast niet uitmaakt waar we zitten, zolang we maar op dezelfde afstand van de oorsprong zitten.

Positie	Fout		Positie	Fout
(5,5,5)	2.80 %		(-5,-5,-5)	4.30 %
(-5,5,-5)	5.45 %		(5,-5,5)	3.99 %
(5$\sqrt{3}$,0,0)	5.06 %		(-5$\sqrt{3}$,0,0)	4.58 %
(0,5$\sqrt{3}$,0)	3.71 %		(0,-5$\sqrt{3}$,0)	5.44 %
(0,0,5$\sqrt{3}$)	5.55 %		(0,0,-5$\sqrt{3}$)	5.92 %

Tabel C.1: Verschillende dipool posities; vaste dipoolrichting

C.1.2 Meeveranderende richting, sterkte vast

We kunnen ons nu nog afvragen of het uitmaakt als we de richting met de locatie mee laten veranderen.
Doen we dezelfde testen opnieuw dan krijgen we de volgende resultaten:

Positie	Richting	Fout
(5,5,5)	(1,1,1)	5.59 %
(-5,5,-5)	(-1,1,-1)	5.04 %
(-5,-5,-5)	(-1,-1,-1)	4.96 %
(5,-5,5)	(1,-1,1)	4.58 %
(5$\sqrt{3}$,0,0)	($\sqrt{3}$,0,0)	3.92 %
(0,5$\sqrt{3}$,0)	(0,$\sqrt{3}$,0)	4.94 %
(0,0,5$\sqrt{3}$)	(0,0,$\sqrt{3}$)	6.24 %
(-5$\sqrt{3}$,0,0)	(-$\sqrt{3}$,0,0)	5.77 %
(0,-5$\sqrt{3}$,0)	(0,-$\sqrt{3}$,0)	4.67 %
(0,0,-5$\sqrt{3}$)	(0,0,-$\sqrt{3}$)	8.48 %

Tabel C.2: Verschillende dipool posities; meeveranderende dipool richting

Ook hier zien we dat de fout aardig stabiel is en het dus ook niet veel uitmaakt dat de richting van de dipool mee veranderd met de locatie.

C.1.3 Verschillende posities (diepte), richting en sterkte vast

Een andere situatie die ook voor kan komen, is dat de diepte van de dipool verandert en daardoor de fout minder dan wel groter wordt. Daarom zullen we ook naar dit geval kijken en dat levert de volgende resultaten:

Positie	Fout		Positie	Fout
(0,0,-0.1)	1.62 %		(0,0,-6)	2.04 %
(0,0,-0.5)	1.62 %		(0,0,-7)	2.57 %
(0,0,-1)	1.62 %		(0,0,-8)	3.83 %
(0,0,-2)	1.65 %		$(0,0,-5\sqrt{3})$	5.92 %
(0,0,-3)	1.69 %		(0,0,-9)	8.32 %
(0,0,-4)	1.75 %		(0,0,-9.5)	16.44 %
(0,0,-5)	1.85 %		(0,0,-9.9)	79.83 %

Tabel C.3: Verschillende dipool diepte; vaste dipoolrichting

Wat bij deze tabel opvalt is dat hoe verder we van de schil gaan zitten hoe nauwkeuriger de oplossing wordt. Gaan we daarentegen heel dicht bij de rand zitten dan is de fout enorm groot. Dit is dus een situatie die we willen vermijden, omdat dit niet realistisch is.

C.1.4 Verschillende richtingen, positie en sterkte vast

In onderstaande tabel hebben we gekeken wat er met de fout gebeurt als we de richting van de dipool veranderen terwijl we de positie en sterkte gelijk houden. Dat levert dan de resultaten uit tabel C.4.
We zien in deze tabel dat de fout bij n=4 aardig verschillend is, maar dat bij n=8 de fout dicht bij elkaar zit. Verder valt op dat in sommige gevallen de fout bij n=4 groter is dan bij een ander geval, maar de fout bij n=8 weer kleiner is. De laatste opmerking over deze tabel is dat het voor de fout niet uitmaakt of de richting de ene kant opnemen of de andere kant op.

C.1.5 Verschillende sterktes, positie en richting vast

Als laatste test situatie zullen we kijken naar het geval dat de positie en richting gelijk blijven. De factor die nu verandert is de sterkte en dat levert de volgende resultaten:

Richting	Fout (n=4)	Fout (n=8)
(1,-1,2)	2.80 %	0.77 %
(-1,1,-2)	2.80 %	0.77 %
(1,-1,-2)	2.78 %	0.75 %
(-1,1,2)	2.78 %	0.75 %
(1,1,-2)	4.87 %	0.76 %
(1,1,2)	4.82 %	0.92 %
(0,0,$\sqrt{6}$)	2.92 %	0.72 %
(0,$\sqrt{6}$,0)	5.28 %	0.76 %
($\sqrt{6}$,0,0)	4.97 %	0.78 %

Tabel C.4: Verschillende dipool richting; vaste positie

Positie	Fout
(0.5,-0.5,1)	2.80 %
(1,-1,2)	2.80 %
(2,-2,4)	2.80 %
(3,-3,6)	2.80 %
(4,-4,8)	2.80 %

Tabel C.5: Verschillende dipool sterkte; vaste dipoolrichting en positie

Hier kunnen we concluderen dat het voor de fout niet uitmaakt hoe groot de dipool sterkte is. Voor de potentiaal maakt het daarintegen wel uit hoe groot de sterkte is, hebben we te maken met een zwakke dipool dan zal de potentiaal ook laag zijn en hebben we een sterke dipool dan zal de potentiaal ook groter zijn.

C.2 Gekromde driehoekjes

We zullen in deze paragraaf de zelfde stappen volgen als in de vorige paragraaf.

C.2.1 Verschillende posities (afstand gelijk) ,richting en sterkte vast

Ook in dit geval beginnen we met een vaste dipool richting en dit levert de volgende resultaten:

Positie	Fout		Positie	Fout
(5,5,5)	1.96 %		(-5,-5,-5)	2.96 %
(-5,5,-5)	4.18 %		(5,-5,5)	2.86 %
($5\sqrt{3}$,0,0)	3.67 %		($-5\sqrt{3}$,0,0)	3.73 %
(0,$5\sqrt{3}$,0)	2.87 %		(0,$-5\sqrt{3}$,0)	3.97 %
(0,0,$5\sqrt{3}$)	4.23 %		(0,0,$-5\sqrt{3}$)	4.19 %

Tabel C.6: Verschillende dipool posities; vaste dipoolrichting

We zien dat hier ook dat er haast geen verschillen optreden.

C.2.2 Meeveranderende richting, sterkte vast

Veranderen we ook hier de richting mee dan krijgen we de resultaten beschreven in tabel C.7.
Ook hier zien we weer dat fout aardig stabiel is.

C.2.3 Verschillende posities (diepte), richting en sterkte vast

Als laatste zullen we ook hier de diepte van de dipool beschouwen en dat levert tabel C.8.
Ook hier zien we weer dat een dipool die dicht onder het oppervlak ligt met de BEM geen goede benadering levert van de potentiaal. Dit was ook wel te verwachten, omdat in het geval van platte driehoekjes het ook al een probleem gaf.

C.2.4 Verschillende richtingen, positie en sterkte vast

Ook hier zullen we weer kijken wat er gebeurt als we de richting van de dipool veranderen. We krijgen dan de tabel C.9.
Uit deze tabel kunnen we hetzelfde concluderen als we eerder gedaan hadden voor de platte driehoekjes. Ook hier valt het weer op dat in sommige richtingen de sprong van n=4 naar n=8 veel groter is dan in andere richtingen.

Positie	Richting	Fout
(5,5,5)	(1,1,1)	4.77 %
(-5,5,-5)	(-1,1,-1)	3.62 %
(-5,-5,-5)	(-1,-1,-1)	3.38 %
(5,-5,5)	(1,-1,1)	3.64 %
($5\sqrt{3}$,0,0)	($\sqrt{3}$,0,0)	2.67 %
(0,$5\sqrt{3}$,0)	(0,$\sqrt{3}$,0)	3.93 %
(0,0,$5\sqrt{3}$)	(0,0,$\sqrt{3}$)	5.16 %
(-$5\sqrt{3}$,0,0)	(-$\sqrt{3}$,0,0)	4.86 %
(0,-$5\sqrt{3}$,0)	(0,-$\sqrt{3}$,0)	3.13 %
(0,0,-$5\sqrt{3}$)	(0,0,-$\sqrt{3}$)	6.35 %

Tabel C.7: Verschillende dipool posities; meeveranderende dipool richting

Positie	Fout
(0,0,-0.1)	0.17 %
(0,0,-0.5)	0.17 %
(0,0,-1)	0.18 %
(0,0,-2)	0.20 %
(0,0,-3)	0.24 %
(0,0,-4)	0.29 %
(0,0,-5)	0.37 %
(0,0,-6)	0.49 %
(0,0,-7)	0.81 %
(0,0,-8)	2.00 %
(0,0,-$5\sqrt{3}$)	4.19 %
(0,0,-9)	6.70 %
(0,0,-9.5)	14.38%
(0,0,-9.9)	80.74 %

Tabel C.8: Verschillende dipool diepte; vaste dipoolrichting

Het verschil is echter dat dit niet in dezelfde richtingen gebeurt als bij de platte driehoekjes.

Richting	Fout (n=4)	Fout (n=8)
(1,-1,2)	1.96 %	0.45 %
(-1,1,-2)	1.96 %	0.45 %
(1,-1,-2)	2.05 %	0.35 %
(-1,1,2)	2.05 %	0.35 %
(1,1,-2)	4.03 %	0.34 %
(1,1,2)	3.55 %	0.57 %
$(0,0,\sqrt{6})$	2.28 %	0.40 %
$(0,\sqrt{6},0)$	4.07 %	0.31 %
$(\sqrt{6},0,0)$	3.73 %	0.43 %

Tabel C.9: Verschillende dipool richting; vaste positie

C.2.5 Verschillende sterktes, positie en richting vast

Als laatste kijken we wat er gebeurt als we de sterkte varieëren. Dat levert dan de resultaten in tabel C.10. Ook hier maakt het, voor de fout, niet uit

Positie	Fout
(0.5,-0.5,1)	1.96 %
(1,-1,2)	1.96 %
(2,-2,4)	1.96 %
(3,-3,6)	1.96 %
(4,-4,8)	1.96 %

Tabel C.10: Verschillende dipool sterkte; vaste dipoolrichting en positie

dat we de sterkte varieëren, maar dat was ook wel te verwachten daar we dat voor platte driehoekjes al gezien hadden.

C.3 Conclusie

We kunnen uit al deze berekeningen concluderen dat wat betreft de dipool positie het haast niet uitmaakt waar we met onze dipool gaan zitten zolang we maar op dezelfde afstand van de oorsprong (of rand) gaan zitten. Verder maakt het ook niet uit als we de dipool richting mee laten veranderen als de positie wordt veranderd. De sterkte van de dipool heeft ook geen invloed op de fout en bij n=8 maakt het in de meeste gevallen ook niet meer zoveel uit hoe we die kiezen.
Het maakt daarentegen wel uit hoe diep we de dipool kiezen. Als de dipool te dicht bij de rand ligt dan is de fout veel te groot om het probleem met behulp van de BEM op te lossen. Nemen we de dipool in de buurt van de oorsprong dan hebben we juist een hele nauwkeurige benadering.

Opmerking: We hebben hier gekeken naar dipool posities binnen het bolschillen model.

Bijlage D

Grid opbouw

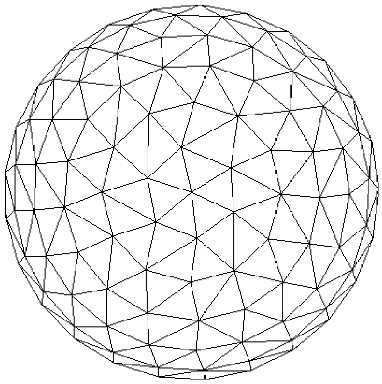
In deze bijlage laten we zien hoe ons grid is opgebouwd.
Als eerste is de bol belegd met punten en vervolgens is met behulp van Delaunay triangulatie een driehoeks grid van die punten gemaakt.

D.1 Punten op bol

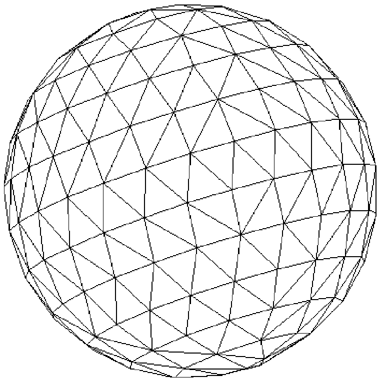
De punten op de bol zijn als volgt belegd.
We zijn begonnen in het bovenste punt, dus op de positieve z-as. Daarna gaan we telkens een niveau lager, waardoor we cirkels gaan beleggen totdat we in het onderste punt bereikt hebben (de negatieve z-as). De punten op de cirkels worden dan op een constante afstand van elkaar geplaatst tot we de cirkel rond zijn.

D.2 Triangulatie

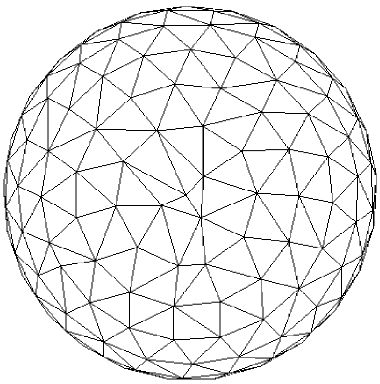
De verbinding van de punten tot driehoeken gebeurt met behulp van een methode die ook wel Delaunay triangulatie wordt genoemd. Daarbij komt het erop neer dat de drie punten die samen een driehoek vormen zo worden gekozen dat als er een cirkel doorheen wordt getrokken er geen andere punten binnen die cirkel liggen. We krijgen dan het rooster dat weergegeven is op de volgende pagina. Hier zien we respectievelijk een boven-aanzicht, zij-aanzicht en onder-aanzicht van het rooster.



Figuur D.1: boven-aanzicht



Figuur D.2: zij-aanzicht



Figuur D.3: onder-aanzicht

Bijlage E

Oplosmethode's

Willen we met gekromde driehoekjes gaan werken dan moeten we wel zeker weten dat de mogelijkheid bestaat om een willekeurig oppervlak ermee te beleggen. Dit willen we op zodanige manier doen dat het mathematisch allemaal in orde is en we ons probleem er mee kunnen oplossen.
In deze bijlage zijn enkele methodes uitgewerkt die mogelijk te gebruiken zijn.

E.1 Methode 1

E.1.1 Bolfuncties

De eerste methode is door gebruik te maken van bolfuncties.
We bepalen doorvoor eerst het gemiddelde van de verschillende knooppunten:

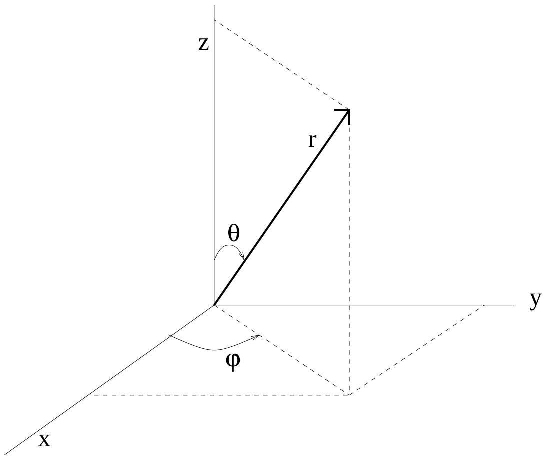
$$\overline{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x_i}. \tag{E.1}$$

Vervolgens schrijven we onze coördinaten om in bolcoördinaten

$$\begin{aligned} x - \overline{x} &= r\sin\theta\cos\varphi \qquad \text{(E.2)} \\ y - \overline{y} &= r\sin\theta\sin\varphi \\ z - \overline{z} &= r\cos\theta \end{aligned}$$

en dan geldt dat:

$$r_i = |\mathbf{x}_i - \overline{\mathbf{x}}| \tag{E.3}$$



Figuur E.1: Bolcoördinaten

$$\varphi_i = \arctan(\frac{y_i - \overline{y}_i}{x_i - \overline{x}_i}) \tag{E.4}$$

$$\theta_i = \arctan(\sqrt{\frac{(x_i - \overline{x}_i)^2 + (y_i - \overline{y}_i)^2}{(z_i - \overline{z}_i)^2}}) \tag{E.5}$$

voor $i = 1...N$.
In het algemeen stellen we dat:

$$r = \sum_{n=1}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} a_{nm\alpha} Y_{nm\alpha}(\theta, \varphi) \tag{E.6}$$

Met:

$$\begin{aligned} Y_{nm\alpha} &= P_{nm}(\cos\theta) X_m(\varphi), \\ X_m(\varphi) &= \cos m\varphi \qquad \alpha = 0, \\ &= \sin m\varphi \qquad \alpha = 1. \end{aligned} \tag{E.7}$$

Hier is $P_{nm}(\cos\theta)$ de geassocieerde *Legendre* functie (zie Bijlage A).
Verder zijn $Y_{nm\alpha}$ niet-genormaliseerde bolfuncties van de n^{de} orde met graad m, de functie is even als $\alpha = 0$ en oneven als $\alpha = 1$. De bolfuncties zijn de eigenfuncties van het θ en φ gedeelte van de *Laplace* operator:

$$\Big[\frac{1}{\sin\theta} \frac{\partial}{\partial\theta}[\sin\theta \frac{\partial}{\partial\theta}] + \frac{1}{\sin^2\theta} \frac{\partial^2}{\partial\varphi^2}\Big] Y_{nm\alpha} = -n(n+1) Y_{nm\alpha} \tag{E.8}$$

Voor de discretisatiepunten moet dan gelden dat

$$r_i = \sum_{n=1}^{\infty} \sum_{m=0}^{n} \sum_{\alpha=0}^{1} a_{nm\alpha} Y_{nm\alpha}(\theta_i, \varphi_i) \tag{E.9}$$

waardoor we een stelsel vergelijkingen krijgen met als onbekende de $a_{nm\alpha}$ die dan nog bepaald moeten worden. Het aantal vergelijkingen zal dus gelijk zijn aan het aantal discretisatiepunten en het aantal onbekende zal gelijk zijn aan het punt waar we de som afbreken. Het handigste is het punt te nemen waarbij we evenveel vergelijkingen als onbekenden hebben. We gaan de som dan afbreken bij het aantal knooppunten (hier gelijk aan N) en dus moeten er, bij elke i, N $a_{nm\alpha}$ zijn. Nu hebben we een $N \times N$ stelsel vergelijkingen waaruit we de onbekende $a_{nm\alpha}$ kunnen uitrekenen. Hebben we dit gedaan dan zijn de $a_{nm\alpha}$ uit vergelijking (E.6) bekend.
Nemen we nu de θ en φ die halverwege twee discretisatiepunten liggen dan kunnen we de bij behorende r uitrekenen met vergelijking (E.6).

E.2 Methode 2

De tweede methode is door gebruik te maken van interpolatie technieken en splines.

E.2.1 Cubische Hermite interpolatie

Interpolatie met behulp van polynomen is niet alleen beperkt tot interpolatie door de knooppunten. Het is ook mogelijk om door andere informatie te interpoleren, zoals door de afgeleide in de knooppunten. Dit resulteert in een interpolatie schema dat handiger is dan *Lagrange* interpolatie en wordt ook wel *Hermite* interpolatie genoemd.

In het cubische geval worden dan twee punten $\mathbf{p}_0$, $\mathbf{p}_1$ en twee tangentiële vectoren $\mathbf{m}_0$, $\mathbf{m}_1$ gegeven. Het doel is dan om een cubisch polynoom $\mathbf{p}$ te vinden die door de volgende data interpoleert:

$$\begin{aligned} \mathbf{p}(0) &= \mathbf{p}_0 \\ \mathbf{p}(1) &= \mathbf{p}_1 \\ \dot{\mathbf{p}}(0) &= \mathbf{m}_0 \\ \dot{\mathbf{p}}(1) &= \mathbf{m}_1 \end{aligned} \tag{E.10}$$

waarbij de punt voor differentiatie staat.
(Zie [5] pagina 87-91.)

Bij deze methode is het nodig om bij gegeven punten, de tangentiële vectoren van de curve te definiëren door het toepassen van differentie formules (lineair of niet-lineair) van de coördinaat vectoren van de gegeven punten. Deze differentie formules worden schattingsmodellen voor de tangentiële vectoren genoemd. Voorbeelden:

- Akima's model
- parabolische model
- 4de orde eindige differentie model

zie [7].

E.2.2 Splines

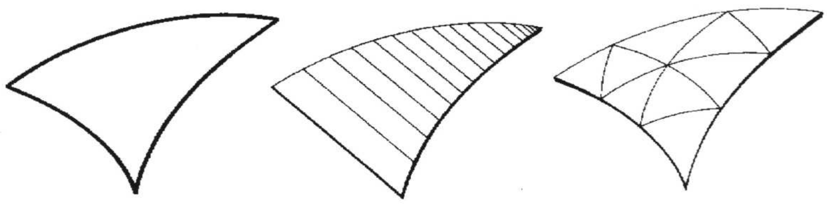
Wat is een spline? De meest simpele definitie is dat een spline een verzameling stuksgewijze polynomen is die aan de overgangen aan zekere gladheidseisen voldoen.
Wat we willen is een interpolant door al de knooppunten en dat doen we door al de interpolanten samen te nemen en we krijgen dan een spline.
Hermite splines worden gemaakt door het aan elkaar plakken van Hermitische interpolanten aan de gemeenschappelijke eindpunten.

E.3 Methode 3

E.3.1 Driehoekige Coons-patches

Stel er zijn drie randkrommen gegeven, zoals in figuur E.3.1.
We zoeken nu een oppervlak dat door alle drie de krommen interpoleert, i.e., een transfinite driehoeks interpolant. De constructie ervan volgt de standaard Coons-patch ontwikkeling op een manier dat het bestaat uit verscheidene "bouwstenen", die op een slimme manier gecombineerd zijn.
Laten we de drie randkrommen aangegeven door $\mathbf{x}(0, v, w)$, $\mathbf{x}(u, 0, w)$ en $\mathbf{x}(u, v, 0)$ nemen. We definiëren de drie "bouw stenen", elk een oppervlak met één rechte zijde, die door twee randkrommen interpoleert, als volgt:

$$\mathcal{P}_1\mathbf{x}(\mathbf{u}) = (1-r)\mathbf{x}(u, 0, w) + r\mathbf{x}(u, v, 0) \qquad r = \frac{v}{v+w}$$



Figuur E.2: De 3 curves links zijn gegeven. Er worden 3 oppervlakken (bestaande uit lijnen) gemaakt waarvan er één in het midden gegeven is. Als we deze 3 oppervlakken samen nemen dan krijgen het uiteindelijk oppervlak rechts.

$$\mathcal{P}_2\mathbf{x}(\mathbf{u}) = (1-s)\mathbf{x}(u,v,0) + s\mathbf{x}(0,v,w) \qquad s = \frac{w}{w+u} \tag{E.11}$$
$$\mathcal{P}_3\mathbf{x}(\mathbf{u}) = (1-t)\mathbf{x}(u,0,w) + t\mathbf{x}(0,v,w) \qquad t = \frac{v}{v+u}$$

Er zijn verschillende combinaties van deze oppervlakken die een $\mathcal{P}\mathbf{x}$ leveren die door alle drie de randen gaat: de Boolean som van twee, b.v. $\mathcal{P} = \mathcal{P}_1 \bigoplus \mathcal{P}_3$, voldoet aan die voorwaarde (zie [5] pagina 344-347).