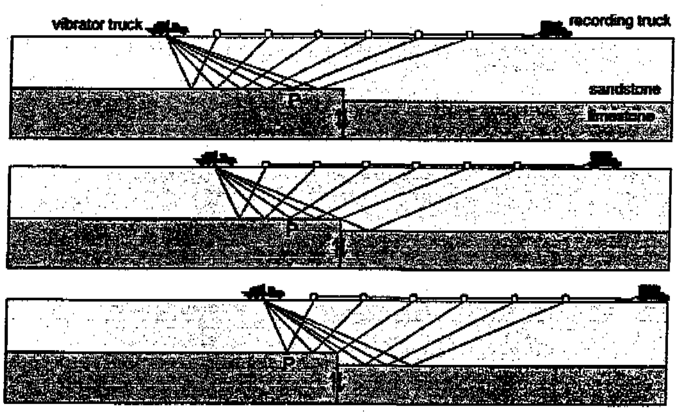
Olie zoeken met numerieke wiskunde

Yogi Erlangga, Kees Vuik en Kees Oosterlee

Voor oliemaatschappijen is het vinden van olie- en gasvoorraden van levensbelang. Een proefboring om te zien of er ergens olie of gas aanwezig is, kost veel geld. Daarom is het verstandig om alleen proefboringen te doen op die plaatsen waar de kans op het vinden van olie het grootst is. Om deze plaatsen te bepalen, is het nodig om de structuur van de aardlagen te kennen. Geologen kunnen op grond van de vorm van de aardlagen redelijk goed voorspellen of er op een bepaalde plaats olie verwacht kan worden.

Voor het bepalen van de positie van de aardlagen wordt gebruik gemaakt van seismisch onderzoek. Het principe werkt als volgt: aan de oppervlakte van de aarde of de zee wordt geluid opgewekt, via een trillende plaat onder een vrachtwagen of een ontploffing van een lading dynamiet. Het geluid verplaatst zich door de ondergrond. Een gedeelte van het geluid wordt teruggekaatst naar het oppervlak. Aan de oppervlakte staat gevoelige apperatuur, die het geluid opvangt en de tijd registreert waarop het geluid aankomt. Omdat de geluidssnelheid verschillend is in verschillende aardlagen, zoals zandsteen, lei, graniet etc., kan uit de tijdmetingen afgeleid worden hoe de ondergrond opgebouwd is. Shell maakt al 30 jaar gebruik van 3 dimensionale metingen. Het probleem is om ook nauwkeurige 3 dimensionale berekeningen te doen.



Figuur 1: Seismisch onderzoek op land

Om de lagen te bepalen merken we op dat geluidsdruk in de ondergrond voldoet aan de golfvergelijking

$$\frac{\partial^2 u}{\partial t^2} = c^2(x,y,z)\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) \qquad (1)$$

samen met geschikte randvoorwaarden. De geluidssnelheid $c(x,y,z)$ is verschillend in de verschillende aardlagen. Omdat à priori de lagen en dus $c(x,y,z)$ onbekend zijn, moeten we een zogenaamd invers probleem oplossen. Dit gaat als volgt: maak een schatting van $c(x,y,z) = c^{(1)}(x,y,z)$ en los de golfvergelijking (1) met randvoorwaarden op. We noemen dit het voorwaartse probleem. Vergelijk dan de berekende oplossing $u^{(1)}$ met de geluidsdruk, die gemeten is in de meetpunten. Als deze identiek zijn, dan is $c^{(1)}(x,y,z)$ een goede benadering van de geluidssnelheid in de ondergrond. Anders gebruiken we de verschillen in de metingen en de berekeningen om een betere schatting van de geluidssnelheid te bepalen $c^{(2)}(x,y,z)$. De procedure wordt zo vaak herhaald, totdat het verschil tussen de metingen en de berekeningen verwaarloosbaar is.

Vergelijking (1) moet dus heel vaak opgelost worden. Een efficiënte oplosmethode voor (1) is dan ook van groot belang. Heel vaak wordt hiervoor de ray-tracing methode gebruikt. Deze methode is voor een "eenvoudige" vorm van de aardlagen (een "eenvoudige" functie $c(x,y,z)$) betrouwbaar en de rekentijd is niet al te lang. Voor gecompliceerde problemen is de ray-tracing methode niet goed bruikbaar en moeten we proberen om (1) op te lossen met een nauwkeuriger methode. Hiervoor discretiseren we de plaatsafgeleiden met behulp van een eindige differentie methode. Voor de tijdafgeleide zijn twee klassen van methoden bekend:

- de tijddomein methode
- de frequentiedomein methode

In de eerste methode wordt de tijdafgeleide ook gedisretiseerd met behulp van een eindige differentie methode. Nadeel is dat er veel tijdstappen nodig zijn om het probleem stabiel op te lossen. Wij hebben er voor gekozen om (1) op te lossen met de frequentiedomein methode.

In de frequentiedomein methode schrijven we de oplossing als

$$u(x,y,z,t) = \sum_{f=1}^{n} \hat{u}_k(x,y,z)\mathrm{e}^{-i2\pi ft}. \qquad (2)$$

Als we (2) invullen in (1) dan volgt dat voor elke k, de functie $\hat{u}$ voldoet aan

$$-\left(\frac{\partial^2 \hat{u}}{\partial x^2} + \frac{\partial^2 \hat{u}}{\partial y^2} + \frac{\partial^2 \hat{u}}{\partial z^2}\right) - k^2\hat{u} = 0, \qquad (3)$$

waarbij $k = 2\pi\frac{f}{c}$. Voordeel is dat we één dimensie minder hebben, nadeel is dat we na de plaatsdiscretisatie een lineair stelsel moeten oplossen. De matrix A van het lineaire stelsel $Ax = b$ heeft de volgende eigenschappen:

1. A is een element van $\mathbb{C}^{n\times n}$, waarbij $n = 10^9$ in een 3D probleem.
2. $A = A^T$ en A is een ijle matrix (bevat veel nul elementen).
3. Er zijn eigenwaarden van A met een positief reëel deel en eigenwaarden van A met een negatief reëel deel.

Eigenschap 1 zorgt ervoor dat het stelsel te groot is om met Gauss eliminatie op te lossen. Eigenschap 2 geeft aan dat iteratieve methoden een goede oplosmethode zouden kunnen zijn. Echter eigenschap 3 leidt tot zeer trage convergentie van de bekende iteratieve methoden. Om dit probleem op te lossen, is in 2001 gestart met een onderzoek door de promovendus Yogi Erlangga. In het eerste jaar, dat voornamelijk besteed is aan het bestuderen van de literatuur, hebben we weinig vooruitgang geboekt. In het tweede jaar bleek dat een preconditionering gebaseerd op

$$-\left(\frac{\partial^2 \hat{u}}{\partial x^2}+\frac{\partial^2 \hat{u}}{\partial y^2}+\frac{\partial^2 \hat{u}}{\partial z^2}\right)+k^2\hat{u},$$

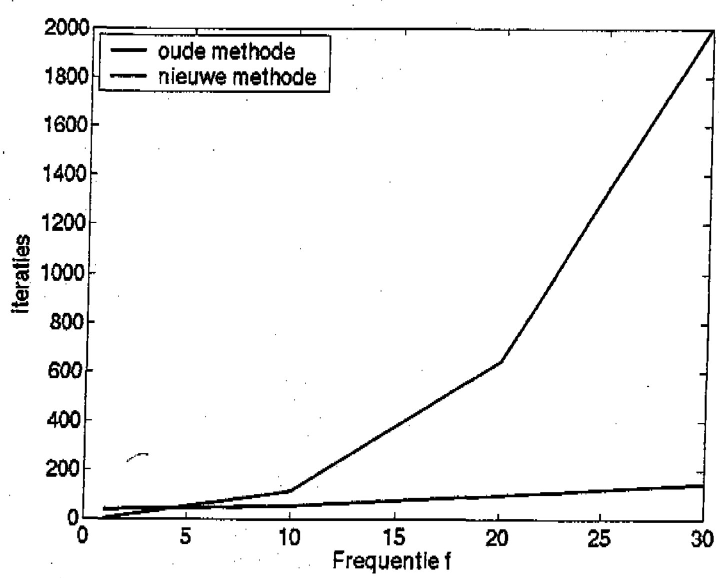
tot een behoorlijke versnelling van de iteratieve methode leidde. Daarna gingen de verbeteringen steeds sneller. Een volgende stap was om de preconditionering te baseren op

$$-\left(\frac{\partial^2 \hat{u}}{\partial x^2}+\frac{\partial^2 \hat{u}}{\partial y^2}+\frac{\partial^2 \hat{u}}{\partial z^2}\right)+ik^2\hat{u}.$$

Uiteindelijk gebruiken we nu als preconditioneringsoperator:

$$-\left(\frac{\partial^2 \hat{u}}{\partial x^2}+\frac{\partial^2 \hat{u}}{\partial y^2}+\frac{\partial^2 \hat{u}}{\partial z^2}\right)-(1-\frac{1}{2}i)k^2\hat{u},$$

die de beste resultaten geeft. In de gegeven preconditionering moet ook een lineair stelsel opgelost worden. De vraag is dan: wat zijn we nu opgeschoten? Het blijkt dat de preconditionerings operator gunstige eigenwaarden heeft. Het uitvoeren van 1 of 2 iteraties met een multigrid methode op dit preconditioneringsstelsel geeft al voldoende nauwkeurigheid voor een snelle convergentie van de totale methode. Om de verbetering van de convergentie te illustreren verwijzen we naar Figuur 2. De oude methode werkte goed voor kleine frequenties, maar het aantal iteraties explodeerde bij grotere frequenties. Bij de nieuwe methode neemt het aantal iteraties slechts lineair toe bij een toename van de frequentie. Omdat de nauwkeurigheid van de methode toeneemt als de frequentie toeneemt, is het heel belangrijk dat de methode geschikt is voor grote frequenties.



Figuur 2: Het aantal iteraties met de oude en de nieuwe methode

Om de vooruitgang te laten zien, merken we op dat aan het begin van ons onderzoek we 2D problemen konden oplossen met $500 \times 500 = 250.000$ roosterpunten. Bij Shell is er met de nieuwe methode onlangs een 3D probleem opgelost met $400 \times 400 \times 400 = 64 \times 10^6$ roosterpunten. De berekende resultaten van de methode hebben een goede nauwkeurigheid. Om nog grotere problemen op te kunnen lossen met $2000 \times 2000 \times 2000 = 8 \times 10^9$ roosterpunten moet de methode aangepast worden voor parallelle computers. Daar wordt nu hard aan gewerkt.