

Een nieuw soort multigrid voor problemen in 3D

P.W. Hemker

Voor TJD, bij zijn afscheid van de UvA.

Meerroostermethoden zijn zeer efficiënte iteratieve methoden voor het oplossen van grote stelsels vergelijkingen. Er is uitgebreid onderzoek gedaan naar het oplossen van stelsels die hun oorsprong vinden in partiële differentiaalvergelijkingen in het platte vlak. Voor stelsels die afkomstig zijn van problemen beschreven in de drie-dimensionale ruimte is veel minder onderzoek gedaan.

Het is welbekend hoe de gebruikelijke meerroostermethoden gewoonlijk voor twee-dimensionale elliptische problemen toegepast worden en dat dezelfde technieken ook direct ggeneraliseerd kunnen worden voor hoger dimensionale problemen. Hierbij wordt er vaak op gewezen dat de totale werklast van het werk dat op de groffe roosters gedaan moet worden, voor een 3D probleem relatief geringer is dan voor een 2D probleem. De keerzijde hiervan is dat ook slechts betrekkelijk weinig grof-rooster componenten van de fout op deze groffe roosters uitgeschakeld kunnen worden. Het gevolg hiervan is dat zeer krachtige relaxatiemethoden noodzakelijk zijn om de totale fout voldoende snel te reduceren. Voor de relaxatie worden dan bijvoorbeeld alternerende vlak-relaxaties aanbevolen, waarbij voor iedere vlak-relaxatie een 2D multiroostermethode gebruikt moet worden.

Dit is uit symmetrie-overwegingen niet aantrekkelijk, omdat altijd een bepaalde volgorde van coördinaat-richtingen gekozen moet worden, waarbij iedere keuze voor sommige problemen (on)gunstiger is dan voor andere. Hierdoor krijgt iedere keuze iets willekeurig.

Verder zijn deze 3D methoden i.h.a. bijzonder moeilijk te vectoriseren of te paralleliseren, zodat weinig profijt getrokken kan worden van de nieuwe computer-architecturen. Het oplossen van 3D problemen met behulp van multiroostermethoden is dan ook bij onderzoekers een minder populair onderwerp dan je zou verwachten.

Aan de andere kant was het het voor de multiroostermethoden al bij twee-dimensionale problemen duidelijk geworden dat het niet altijd gewenst is een fijner rooster te construeren door alle cellen –zoals gebruikelijk– in beide coördinaat-richtingen te halveren. Voor sommige problemen blijkt het gunstig om de maaswijdte in één richting ongewijzigd te laten en slechts in de andere richting te halveren. Zo onstrond het principe van “semi-coarsening” (halveren i.p.v. vierendelen). Ook hier bestaat natuurlijk een probleem afhankelijk argument om de ene coördinaatrichting een andere behandeling te geven dan de andere. Voor een algemene, probleem-

onafhankelijke methode kunnen beide halveringen tegelijk optreden. In dat geval heeft een fijn rooster twee verschillende “half-groffe” roosters. We moeten dan zien hoe deze twee half-groffe roosters de grof-rooster informatie omtrent de oplossing aan het fijne rooster doorspelen.

Ditzelfde principe van halveren kunnen we ook toepassen in 3 dimensies, zij het dat het aantal half-groffe roosters veel groter wordt. *In deze bijdrage wil ik laten zien hoe nu grof-rooster informatie gebruikt kan worden om de fout op een fijn rooster zo goed mogelijk te reduceren. Verder laat ik kort zien dat de foutcomponenten die wèl op een fijn rooster aanwezig zijn en niet op groffe roosters gevonden worden, van een heel speciale soort zijn, waardoor deze met een heel eenvoudige relaxatie-methode (zoals gedempt Jacobi) gemakkelijk bedwongen kunnen worden.*

Om te laten zien hoe dit gaat verwijs ik naar de figuur 0.3 We gaan uit van een rooster cel $(0,0,0)$ op niveau (‘level’) 0. Door de cel in één van de coördinaatrichtingen te halveren krijgen we 3 sooren rooster cellen (en dus 3 roosters) op niveau 1. We herhalen dit en krijgen zo 6 roosters op niveau 2 en 10 roosters op niveau 3. We kunnen dit process voortzetten, maar voor ons doel laten we zien hoe de informatie van van een benadering op rooster $(1,1,1)$ al voor het grootste deel gevonden kan worden uit de informatie op de roosters $(0,1,1)$, $(1,0,1)$, $(1,1,0)$, $(1,0,0)$, $(0,1,0)$, $(0,0,1)$ en $(0,0,0)$. Verder blijft er op het rooster $(1,1,1)$ nog één component over (van de acht) welke niet door de groffe roosters bepaald wordt. Dat is de schaakbord-component die eenvoudig door gedempte Jacobi-relaxatie geëlimineerd kan worden.

Voor de analyse nemen we aan dat de functie die in de 3D ruimte bepaald moet worden –in iedere cel van elk rooster– gerepresenteerd wordt door één getal: de gemiddelde waarde van de functie over die cel (d.i. een “cell-centered” benadering). In dat geval is het eenvoudig een grof-rooster representatie uit een fijn-rooster representatie van de functie af te leiden. Het gemiddelde van de waarden in twee fijn-rooster cellen is dan gelijk aan de waarde in de grof-rooster cel. Deze handeling beschrijft de restrictie-operator die een fijn-rooster representatie omzet in een grof-rooster representatie.

De omgekeerde bewerking (de prolongatie): het vinden van een fijn-rooster representatie bij een gegeven grof-rooster representatie lijdt altijd aan een gebrek aan kennis omtrent de te benaderen functie. We zullen aannemen dat we bij een gegeven grof-rooster functie niet méér weten dan de gemiddelde waarde over de cel, zodat bij splitsing over twee kleinere cellen beide cellen dezelfde gemiddelde waarde krijgen toegewezen (d.i. stuksgewijs constante interpolatie).

Tussen de representatie van de functie op het rooster $(1,1,1)$ en de representatie op de roosters $(0,1,1)$, $(1,0,1)$, $(1,1,0)$, $(1,0,0)$, $(0,1,0)$, $(0,0,1)$ en $(0,0,0)$ bestaat nu een duidelijke relatie. De representatie op de 7 groffe roosters kan direct uit die op het fijne rooster worden afgeleid. Eén soort functies die op het fijne rooster kan worden gerepresenteerd zal echter op de groffe roosters geen spoor achterlaten. Zulke functies hebben een beur-

telings positieve en negatieve waarde van dezelfde grootte in naast elkaar gelegen cellen op het rooster (1,1,1). We geven dit dan in figuur 0.1. Het is duidelijk dat de restrictie van een dergelijke fijn-rooster functie op alle groffe roosters identiek nul is. Deze component van de representatie op niveau (1,1,1) noemen we de "schaakbord-component", naar analogie met het twee-dimensionale geval.

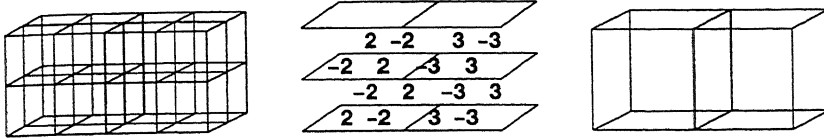


FIGURE 0.1. De 3D schaakbordcomponent op rooster (1,1,1).

Als we de representatie van de functie $f(x, y, z)$ op rooster (n, m, l) aangeven met $f_{n,m,l}(x, y, z)$ dan laat een eenvoudige berekening zien dat

$$\begin{aligned}
 f_{1,1,1}(x, y, z) &= +f_{0,1,1}(x, y, z) + f_{1,0,1}(x, y, z) + f_{1,1,0}(x, y, z) \\
 &\quad - (f_{1,0,0}(x, y, z) + f_{0,1,0}(x, y, z) + f_{0,0,1}(x, y, z)) \\
 &\quad + f_{0,0,0}(x, y, z) + r_{1,1,1}(x, y, z),
 \end{aligned}$$

waarbij $r_{1,1,1}(x, y, z)$ precies een schaakbordcomponent is. We zien dus dat –op de schaakbordcomponent na– de functies op niveau 3 bepaald wordt door de functies op niveau 2, 1, en 0. We zagen al dat de representaties op groffere roosters direct gevonden kunnen worden uit de representaties op fijnere roosters, zodat de representatie op niveau 3, –modulo de schaakbordcomponent– bepaald wordt door de representaties op niveau 2. Voor praktisch alle elliptische problemen is een schaakbord component van (de fout in) de benadering van de oplossing eenvoudig te bepalen d.m.v. een simpele relaxatiemethode zoals gedempte Jacobi-iteratie.

Op geheel analoge wijze kunnen we laten zien dat de benaderingen op niveau n gevonden kunnen worden door de benaderingen op niveau $n-1$ te gebruiken en door op ieder rooster Jacobi-relaxatie te gebruiken voor het vastleggen van de bijbehorende schaakbord component.

Op het eerste gezicht lijkt de nieuwe methode niet efficiënt. Wanneer een uniform fijnste rooster voor de benadering gebruikt wordt, met n^3 cellen, is het totaal aantal cellen op alle roosters bij de klassieke aanpak begrensd door $\frac{6}{7}n^3$, en bij de nieuwe aanpak door $8n^3$. Dit lijkt een verslechtering met een factor 7. Wanneer we echter in de beschouwing betrekken dat bij de nieuwe aanpak een veel eenvoudiger relaxatiemethode gebruikt kan worden, kan deze factor 7 al voor een groot gedeelte gecompenseerd worden.

Verder kan de eenvoudige relaxatie (Jacobi) veel beter gevectoriseerd worden dan de meer gecompliceerde relaxaties die in het klassieke geval nodig zijn. Bovendien zien we dat $7/8$ deel van het werk op roosters van een

groffer niveau moet gebeuren, en dat de berekeningen op de verschillende roosters van eenzelfde niveau geheel onafhankelijk van elkaar zijn. Hierdoor kunnen deze berekeningen parallel worden uitgevoerd. De nieuwe aanpak biedt zo uitstekende mogelijkheden voor de exploitatie van de moderne computerarchitecturen.

Tenslotte merk ik op dat het onnodig is uit te gaan van een uniform fijnste rooster. De geneste structuur van de roosters biedt een zeer goede mogelijkheid om met een grof (uniform) rooster te beginnen en –adaptief– uitsluitend op die plaatsen fijnere roostercellen te genereren waar dat voor het benaderen van de gewenste functie noodzakelijk is. Het is een uitdaging deze mogelijkheden van de nieuwe algoritme te realiseren en toe te passen op praktijkproblemen.



FIGURE 0.2. TJD geflankeerd door J.H. Wilkinson en L.D. Fosdick

Beste Dirk,

Met je college lineaire algebra heb je mij ingewijd in de kunst van het oplossen van stelsels vergelijkingen. En geregeld benadrukte je het belang van het gebruik van de nieuwe mogelijkheden die moderne computerarchitecturen ons bieden. Een combinatie van beide zaken en de toepassing daarvan op praktijkproblemen, levert nog steeds een boeiend en uitdagend onderwerp waaraan ik geregeld veel plezier beleef. Ik hoop –ook bij presentaties over dit onderwerp– je nog vaak te mogen ontmoeten.

Piet Hemker

Een nieuw soort multigrid in 3D

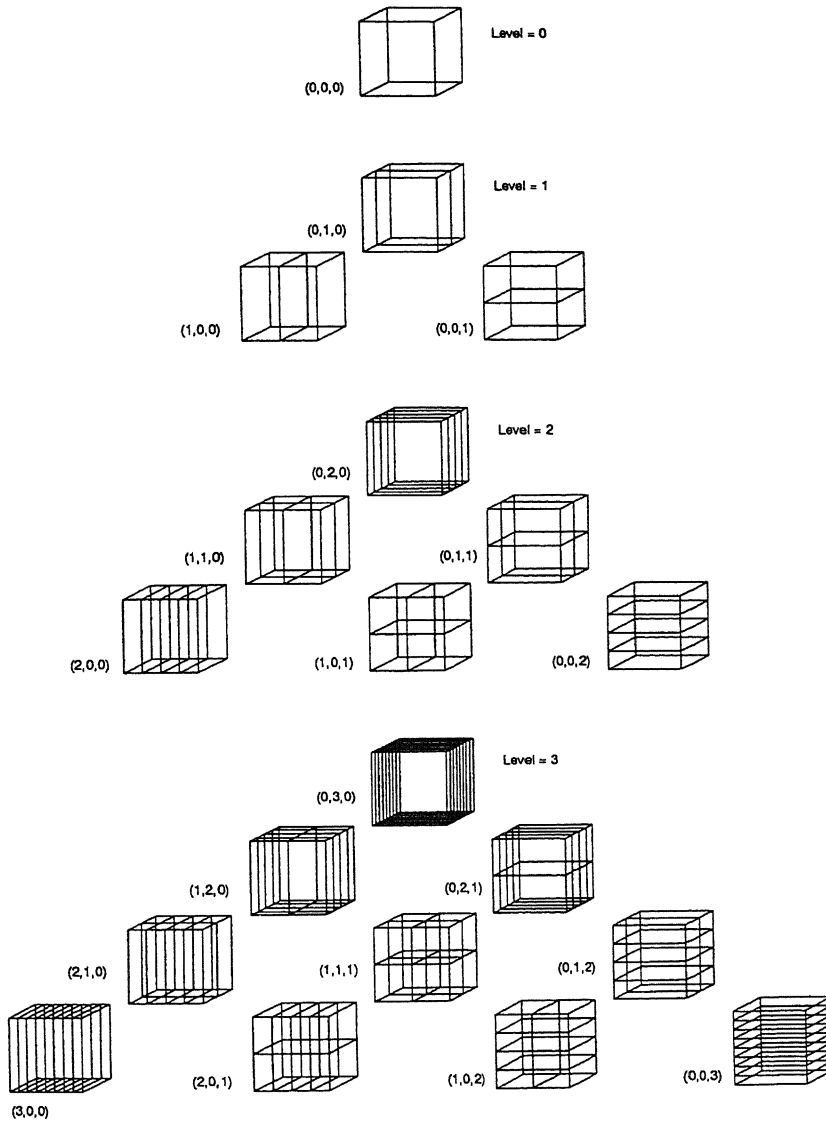


FIGURE 0.3. De roosters op de niveaus 0, 1, 2 en 3.