

Samenvatting

Wie kent niet het spelletje *Flight Simulator*? Gewapend met een joystick kunnen we tegenwoordig vanachter het bureau een (virtueel) vliegtuig de hele wereld rondsturen. Achter dit ogenschijnlijk onschuldige tijdsverdrijf gaat echter een serieuzere toepassing schuil. Door vliegtuigen na te bootsen op computers is het mogelijk om piloten te laten oefenen met vliegen zonder dat dure vliegtuigen de hangar hoeven te verlaten. Ook rampscenario's, zoals het uitvallen van een motor, kunnen worden getraind zonder noemenswaardige risico's of kosten.

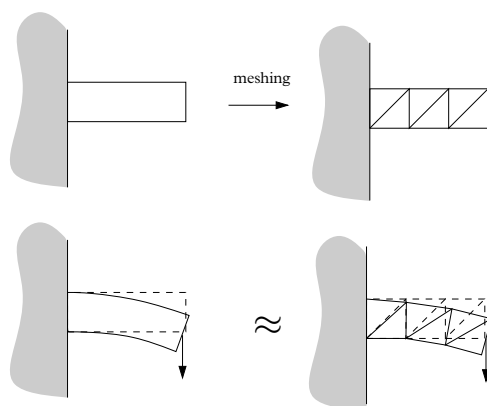
Net als piloten moeten chirurgen in hun werk complexe handelingen uitvoeren, waarbij het gevolg van een fout ernstig kan zijn. Het is lijkt daarom aantrekkelijk om ook chirurgen te trainen met behulp van computers. Een dergelijk simulatie systeem bootst de anatomie van een patient na, en stelt de te onderwijzen student in de gelegenheid om te experimenteren met snijden, schroeven, hechten, etc. in virtueel lichaamsweefsel. De student hanteert een speciale joystick, die de handelingen doorgeeft aan een computersimulatie. Deze simulatie berekent direct het resultaat van de manipulatie, bijvoorbeeld een vervorming of snede. Het resultaat wordt direct zichtbaar gemaakt, en via *force-feedback* ook voelbaar gemaakt in de joystick.

Een kernprobleem bij het construeren van zulke systemen is simuleren hoe menselijk weefsel vervormt. Enerzijds moet deze vervorming enigszins realistisch zijn, zodat de simulatie als geheel geloofwaardig is, anderzijds is er voor het berekenen van die vervormingen slechts zeer weinig rekentijd beschikbaar: om goede terugkoppeling voor de gebruiker te garanderen, moet het antwoord direct getoond moet worden. Tussen deze twee tegenstrijdige belangen moet dus een compromis gevonden worden.

Realistische simulatie van vervorming is op zich geen nieuw onderzoeksterrein. Botstests voor auto's worden bijvoorbeeld al jaren doorgerekend op de computer met een techniek die bekend staat als de Eindige Elementen Methode (*Finite Element Method*, FEM). De kern van deze techniek is dat het te vervormen object wordt opgedeeld in blokjes (in ons geval driehoeken of tetraeders). Uit de mechanische eigenschappen van dit collectief van blokjes worden vergelijkingen afgeleid, en daarmee wordt een oplossing berekend. De eindige elementen methode is geïllustreerd in figuur A.1.

Dit proefschrift onderzoekt aspecten van interactieve vervormings met de Eindige Elementen Methode, waarbij we de toepassing op chirurgiesimulatie in ons achterhoofd houden.

Bij het toepassen van eindige elementen op medische simulaties is er een extra complicatie. De opdeling in blokjes (het rooster), kan tijdens de simulatie veranderen. Bij-



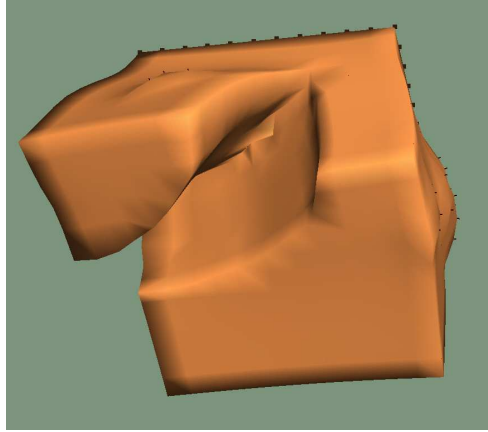
Figuur A.1: In de eindige elementen methode wordt het te simuleren object (boven links) in driehoekjes (“blokjes”) opgedeeld (boven rechts). De exacte oplossing van een probleem (onder links) wordt dan benaderd door het geroosterde object (onder rechts).

voorbeeld, als er gesneden wordt in een object, dan verandert de vorm van het object, en dus ook het rooster. Deze veranderingen hebben invloed op het ontwerp van simulaties, immers alle informatie die van het rooster afhangt verandert ook, en kunnen we beter helemaal niet uitrekenen of opslaan. Daarnaast is er een numeriek effect dat minder voor de hand ligt: de eigenschappen van het rooster zijn van invloed op de snelheid en de precisie van de berekeningen: mooiere roosters leveren preciezere en snellere berekeningen. Als het rooster veranderd wordt (bijvoorbeeld door een gesimuleerde snede), mag dit de kwaliteit van het rooster dus niet nadelig beïnvloeden.

Hoofdstuk 3 beschrijft onze eerste pogingen om een eindige elementen simulatie te construeren. De simulatie gebruikt een eenvoudig elasticiteitsmodel (lineaire elasticiteit), met een statische iteratieve oplossingsmethode: een numeriek proces dat stapje voor stapje naar de uiteindelijke oplossing toe gaat. Het systeem heeft ook de mogelijkheid om te snijden in het gesimuleerde object. Hierbij hebben we een techniek geïntroduceerd die de grootte van het rooster (het aantal blokjes) constant houdt. Dit is aantrekkelijk, aangezien dit de rekentijd per stap laag houdt. Figuur A.2 laat een plaatje zien van het prototype in actie.

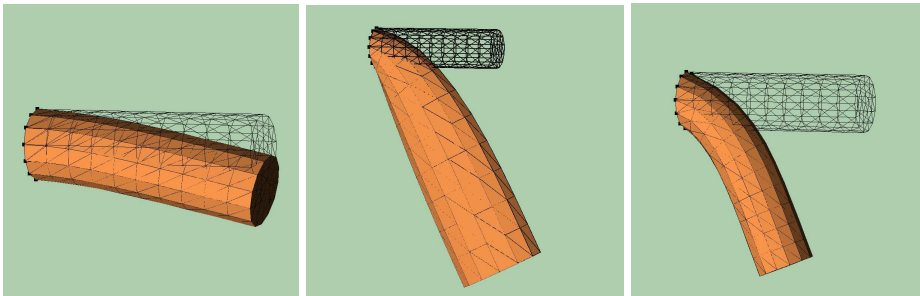
Het systeem voldeed aan onze eerste verwachtingen, maar tijdens de implementatie kwamen we onverwachte problemen tegen. Ten eerste, het lineaire elasticiteitsmodel is alleen realistisch voor beperkte gevallen, namelijk als de vervormingen relatief klein zijn. Ten tweede, onze snijtechniek verandert de vorm van de rooster-blokjes zodanig dat elke stap van het oplossingsproces maar traag naar de oplossing toe gaat: na het maken van snedes nemen de prestaties van het programma af.

Hoofdstuk 4 onderzoekt de statische iteratieve aanpak voor het doorrekenen van deformaties nauwkeuriger. De aanpak uit hoofdstuk 3 is hiervoor uitgebreid naar ingewikkelder materiaalmodellen. Het verschil tussen deze modellen is geïllustreerd in figuur A.3. Om na te gaan hoe snel onze algoritme werkt, is het vergeleken met de stan-



Figuur A.2: Een lineair elastisch object met een snede. Het object is aan de achterkant vastgezet, en wordt zijwaarts uitgerekt.

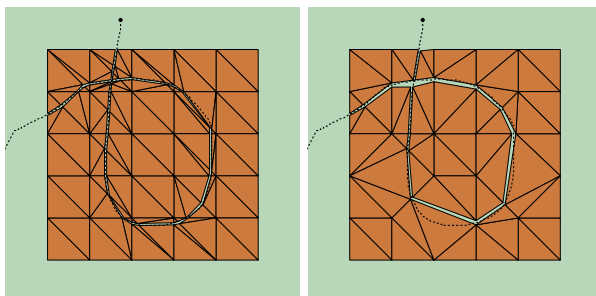
daard methode voor het doorrekenen van elasticiteitsvergelijkingen: dynamische relaxatie met demping. Uit experimenten kunnen we afleiden dat bij verstandige implementatie onze aanpak altijd minstens even snel is als de dynamische. Uit deze experimenten blijkt ook nogmaals dat de roosterkwaliteit van grote invloed is op de rekensnelheid, en dat drie-dimensionale simulaties te rekenintensief zijn voor serieuze interactieve toepassingen.



Figuur A.3: Lineaire elasticiteit is een benadering die geldig is voor kleine vervormingen (links). Bij grotere belastingen van zacht materiaal leidt het tot onrealistische resultaten (midden). Een niet-lineair materiaal, zoals neo-Hookeans materiaal (rechts), geeft betere resultaten, maar vereist ingewikkeldere algoritmes. Door te testen hoe snel het object de ruststand bereikt vanaf de beginstand (als *wire frame* weergegeven), kunnen we een nagaan hoe snel een bepaalde algoritme is.

Het is moeilijk het inwendige van ruimtelijke objecten te visualiseren en te begrijpen. Dit maakte het lastig de problemen van de snijtechniek in hoofdstuk 3 te begrijpen. In hoofdstuk 5 werpen we daarom nogmaals een blik op snijden in roosters, maar doen

dat voor platte, twee-dimensionale objecten. We presenteren een techniek om met een puntvormige scalpel snedes te maken in een rooster opgebouwd uit driehoeken. Om de kwaliteit van het rooster te waarborgen, gebruiken we daarbij een standaard methode om goede triangulaties te genereren, de Delaunay-triangulatie. Deze nieuwe snijtechniek produceert kleinere roosters met beter gevormde driehoeken, hetgeen ook te zien is in figuur A.4. We laten ook zien hoe deze aanpak te generaliseren is naar gebogen oppervlakken in de ruimte, waar één scalpel op meerdere punten tegelijk een incisie kan maken.

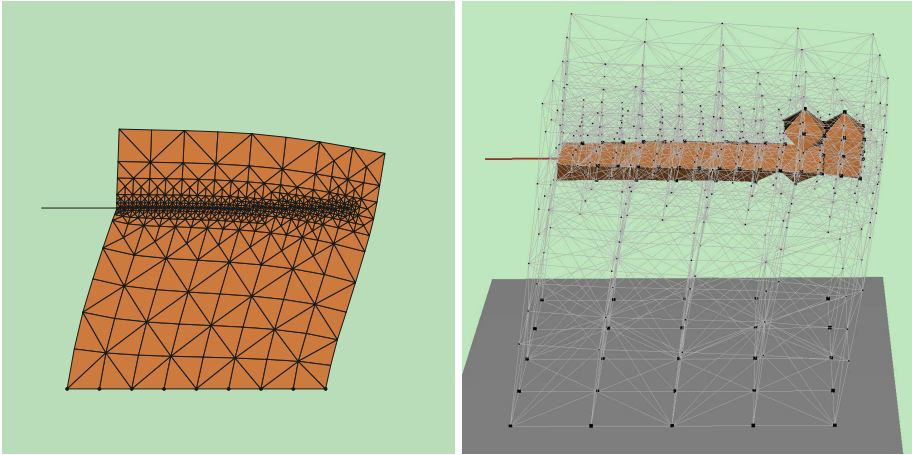


Figuur A.4: Het maken van snedes in roosters kan met behulp van *subdivisie*: driehoekjes worden onderverdeeld om een snede te representeren (links). Het resultaat bevat echter veel driehoekjes, die overwegend plat zijn, hetgeen ongewenst is uit oogpunt van de eindige elementen methode. Door Delaunay-triangulaties handig te gebruiken (rechts), kunnen we hetzelfde pad met minder driehoekjes weergeven, die ook nog minder plat zijn.

Zoals gezegd, voor serieuze toepassingen zijn drie-dimensionale modellen te duur om interactief door te rekenen. In hoofdstuk 6, bekijken we daarom een twee-dimensionaal probleem, namelijk simuleren hoe zacht weefsel vervormt als er naalden in worden gestoken. Dit probleem is al opgelost door anderen, maar daarbij werd een aanpak gebruikt die fundamenteel beperkt was tot platte objecten, en lineair materiaal. In dit hoofdstuk presenteren we een aanpak die ook andere modellen kan gebruiken, en bovendien generaliseert naar ruimtelijke problemen. In dit hoofdstuk wordt er niet gesneden, maar het rooster wordt wel gewijzigd door verfijningen. Door het rooster alleen daar te verfijnen waar nodig, kunnen we goedkoop een relatief precieze simulatie doen. Figuur A.2 geeft een voorbeeld van zo'n simulatie.

Al het werk in de voorgaande hoofdstukken is uitgeprogrammeerd. Bij al deze systemen, moet de simulatie bijhouden hoe de blokjes—driehoeken of tetraeders—met elkaar verbonden zijn. Bij het implementeren van deze systemen is een datastructuur ontwikkeld, die de laag-bij-de-grondse administratie van al die verbindingen onderbrengt in een aparte programma module. Roosterwijzigingen kunnen daardoor in de rest van het programma op een abstract nivo gespecificeerd en geprogrammeerd kunnen worden. Hoofdstuk 7 beschrijft deze datastructuur, en geeft algoritmes om de administratie op orde te houden.

De motivatie van dit onderzoek was hoe vervormingen ten behoeve van medische



Figuur A.5: Een simulatie van een naald insertie in 2D (links). Door rondom de naald het rooster te verfijnen, kan tegen geringe kosten een hoge precisie worden behaald. Door een eenvoudige verfijningstechniek te gebruiken, kan hetzelfde procedé ook in 3D worden toegepast (rechts).

trainingssimulaties realistisch gesimuleerd konden worden. Deze toepassing bracht ons op een eindige elementen aanpak met veranderende roosters. In hoeverre dit voldoende is om bruikbare simulaties te bouwen, blijft een open vraag: realisme is niet alleen een kwalitatief maar ook een kwantitatief begrip. Meer kwantitatief realisme, met andere woorden, een hogere precisie, wordt betaald met intensiever berekeningen en dus tragere programma's. De haalbaarheid van zulke simulaties staat of valt dus met de gewenste precisie, en die kan enkel in de praktijk worden bepaald. Dit geldt ook voor eventueel verder onderzoek. Er zijn vrijwel oneindig veel technieken om de huidige aanpak te verbeteren. De mechanica van levend weefsel is immers complex, terwijl een interactieve simulatie zeer weinig rekentijd toestaat. Om een zinvolle keuze te maken uit al deze mogelijkheden, is het nodig technieken te toetsen aan praktijktoepassingen.