

Cover Page



Universiteit Leiden



The handle <http://hdl.handle.net/1887/22875> holds various files of this Leiden University dissertation

**Author:** Reehuis, Edgar

**Title:** Guiding evolutionary search towards innovative solutions

**Issue Date:** 2013-12-17

# Samenvatting

Binnen de *optimalisatie* is het doel een *invoer* te vinden waarvoor een gegeven wiskundige functie de *optimale uitvoer* levert. Deze optimale uitvoer is de maximale dan wel de minimale waarde die de functie kan retourneren, een keuze die vooraf gespecificeerd is. Een *ontwerpprobleem* in de reële wereld, zoals het bepalen van de vorm van een vleugelprofiel, kan gezien worden als een geval van *kwaliteitsoptimalisatie*. De invoer is een voorgesteld ontwerp en de uitvoer diens vastgestelde kwaliteit. Aan een ontwerp refereert men dan als *oplossing* voor het optimalisatieprobleem.

Om een optimalisatieprobleem geautomatiseerd te kunnen behandelen moet de invoer op zo'n wijze aangeleverd, dat wil zeggen *gecodeerd*, worden dat het mogelijk is automatisch de oplossingen te variëren en aan te passen. Een oplossing is bijvoorbeeld gecodeerd als een rij getallen, waarna het daadwerkelijke ontwerp verkregen wordt door deze rij getallen te *decoderen*. Verder moet de kwaliteit van een ontwerp ook getalsmatig uitgedrukt worden. Vaak wordt een opzet gebruikt waarin de numeriek uitgedrukte oplossing ingevoerd wordt in een *simulator* die het reële gedrag van het bedoelde ontwerp benadert. Gebruikmakend van een gesimuleerde omgeving komt deze tot een benadering van de kwaliteit. De simulator neemt hier dus de rol aan van een functie die de kwaliteit uitdrukt.

Gegeven de numerieke voorstelling van oplossingen en de beschikbare geautomatiseerde kwaliteitsbepaling kan een *optimalisatiealgoritme* toegepast worden dat, zonder menselijke tussenkomst, de optimale oplossing zoekt door middel van het systematisch variëren van de numerieke invoer. *Evolutionaire Algoritmen* (EA's) zijn methoden die een functie optimaliseren gebruikmakend van een *populatie* van, doorgaans numeriek voorgestelde, oplossingen die *evolueren* richting de optimale configuratie. In tegenstelling tot klassieke, *deterministische* optimalisatiealgoritmen, berusten de *stochastische* EA's op *toeval*, zoals evolutie in de natuur berust op toeval. De functie die geoptimaliseerd dient te worden verzorgt feedback over "*fitness*", in de zin van aangepastheid, van oplossingen. Deze fitness wordt gebruikt om de betere oplossingen van de rest te scheiden. Gezien het feit dat ze werken door middel van een populatie van oplossingen zijn EA's relatief langzaam in het benaderen van een optimum, met betrekking tot het vereiste aantal functie-evaluaties. Ze zijn echter wel redelijk

opgewassen tegen bepaalde verraderlijke eigenschappen in een fitnesslandschap die andere methoden op het verkeerde spoor kunnen zetten. Dit, en de hoge flexibiliteit die EA's hebben in de exacte formulering van het optimalisatieprobleem maakt dat ze vaak worden gebruikt voor ontwerproblemen.

Optimalisatiemethoden in het algemeen, en EA's in het bijzonder, zijn in staat ondersteuning te bieden bij het vinden van *innovatieve oplossingen*, dit is ruwweg gedefinieerd als een vernieuwende manier om tot goede prestaties of kwaliteit te komen. Een optimalisatiemethode volgt namelijk mogelijk een ander pad door de zoekruimte dan een ingenieur ooit zou doen gebaseerd op diens genoten opleiding, maar zelfs in een EA convergeert de populatie met hoge waarschijnlijkheid naar één enkele geoptimaliseerde oplossing. Door hun stochastische opzet zijn EA's daarentegen wel in staat om alternatieve oplossingen te produceren als ze meerdere malen uitgevoerd worden.

Dit proefschrift heeft als doel een methode te ontwikkelen die maakt dat optimalisatiealgoritmen zich expliciet richten op het produceren van verschillende hoge-kwaliteitsoplossingen. Het idee is het optimalisatiealgoritme zoals gebruikelijk te laten convergeren naar een optimum, waarna een verkenningsfase start waarin gezocht wordt naar een nieuw startpunt voor de volgende kwaliteitsoptimalisatiefase. Deze sequentiële opzet is vereist gezien het uiteindelijke ontwerprobleem waarop de methode zal worden getest. Hierin wordt de zoektocht namelijk altijd gestart vanaf dezelfde initiële oplossing, omdat er grote, onduidelijk begrensde gebieden zijn in de zoekruimte met oplossingen waarvoor de simulator geen bruikbare uitkomst produceert.

Voor een mogelijke *exploratiemaat* om te verkenningsfase te sturen richten we ons op de ontwikkelingsrobotica: hierin worden zoekruimten van mogelijke acties doorzocht met als doel de vaardigheden van de robot te verbeteren door middel van het uitvoeren van nieuwe acties. Centraal hierin is het model van de werkelijkheid dat de robot stapsgewijs opbouwt en dat hem in staat stelt vaardigheden ten toon te spreiden. Dit *leren* door middel van acties wordt gedaan naar analogie van hoe mensen en dieren leren: de grens wordt opgezocht van de beschikbare kennis en het opgebouwde begrip want dat is uitdagend, maar indien iets te ver afwijkt van het bekende wordt het te ingewikkeld en verliest het zijn "*interessantheid*".

Deze interessantheid, uitgedrukt met betrekking tot een *benaderingsmodel* van de kwaliteitsfunctie dat opgebouwd wordt tijdens de zoektocht op basis van getoetste

---

oplossingen, gebruiken we om de verkenningsfase te sturen. In een denkbeeldig zoeklandschap met pieken en dalen beweegt de zoektocht zich in de optimalisatiefase omlaag, want we spreken af dat we op zoek zijn naar minima, terwijl hij in de verkenningsfase weer omhoog beweegt, mogelijk over meerdere pieken en door meerdere dalen.

Het uiteindelijke probleem betreft het ontwerp van een *statorblad* voor toepassing in een turbofan vliegtuigmotor voor kleine zakenjets. De niet-bewegende statorbladen worden achter bewegende rotorbladen geplaatst om de luchtstroom te ontwarren, dat wil zeggen weer recht te maken. Zoals vermeld is de zoekruimte van dit probleem onderhevig aan grote gebieden met ongeschikte oplossingen. Dit kunnen we echter uitbuiten als *stopconditie* voor de verkenningsfase. Zodra we een gebied met ontoelaatbare oplossingen betreden, stoppen we de verkenning en starten we de optimalisatiefase. De kwaliteitsoptimalisatie is in staat het gebied met ontoelaatbare oplossingen op eigen kracht te verlaten. Als stopconditie voor de optimalisatiefase wordt een bepaalde nauwkeurigheid in de bereikte waarden gebruikt, wat mogelijk is gezien het convergente karakter van de kwaliteitsoptimalisatie. Dit alles leidt tot een samengestelde aanpak waarin meer oplossingen worden gevonden, een hogere diversiteit aan oplossingen wordt bereikt en waarin de gevonden oplossingen van betere kwaliteit zijn dan in de standaard aanpak. Deze standaard aanpak is simpelweg het eerdergenoemde meerdere malen uitvoeren van het optimalisatiealgoritme, zonder enige hulp van een exploratiemaat.

