

Genetiese algoritmes vir rekenaargesteuende induktor-ontwerp

Jean Fivaz en Willem A. Cronjé

Elektriese, Elektroniese en Rekenaarwetenskap, Randse Afrikaanse Universiteit, Aucklandpark

E-pos: jeanfivaz@hotmail.com

UITTREKSEL

Die doelwit van hierdie ondersoek is om deur middel van 'n gevallestudie te bepaal watter voordele genetiese algoritmes bied in rekenaargesteuende ontwerp soos van toepassing op induktors. Daar word probeer om hierdie voordele te benut in ontwerpse probleme, waar die gewenste resultaat aan 'n aantal spesifikasies en beperkings moet voldoen, soos tipies aangetref in drywingselektronika tydens die ontwerp van praktiese induktors.

Van die programmatuur word verlang dat dit komponente soos geleiers en kerne vanuit 'n databasis van beskikbare komponente sal kies en die keuses sal evalueer en toets op grond van die komponente se karakteristieke data, wat gelees word vanuit 'n databasis wat die inligting bevat. Die voorgestelde ontwerp moet in elke geval prakties gekonstrueer kan word, aan al die beperkings wat gestel is voldoen en nogtans karakteristieke so naby moontlik aan die aanvanklike spesifikasies toon.

ABSTRACT

Genetic algorithms in computer aided inductor design

The goal of this investigation is to determine the advantages of using genetic algorithms in computer-aided design as applied to inductors. These advantages are exploited in design problems with a number of specifications and constraints, as encountered in power electronics during practical inductor design.

The design tool should be able to select components, such as cores and wires, from databases of available components, and evaluate these choices based on the components' characteristic data read from a database of manufacturers' data-sheets. The proposed design must always be practically realizable, as close to the desired specifications as possible and within any specified constraints.

1. INLEIDING

Genetiese algoritmes is 'n relatief jong programmeringstegniek, wat onder die kategorie van evolusiegebaseerde programmeringstegnieke val. Gedurende die afgelope dekade het hierdie tegniek reeds belowende resultate in verskeie toepassings getoon.^{1,2,3}

Binne die vakgebiede van elektronika en elektroniese ingenieurswese heers daar veral belangstelling in die aanwending van genetiese algoritmes vir die outomatisering van ontwerpse prosedures,¹ veral met die oog op toepassings in rekenaargesteuende ingenieurswese ("Computer-Aided Engineering", CAE).

Ontwerpse probleme soos effektiewe optimalisering onder meervoudige beperkings raak wiskundig ingewikkeld as nieliniêre en diskontinue komponentkarakteristieke in ag geneem word. Dikwels word die ervaring van die menslike ontwerper hier benodig en kan die ontwerpse proses nie direk geoutomatiseer word nie.

Met die bogenoemde probleme in ag geneem moet die rekenaarsteun-programmatuur met 'n ontwerp vorendag kom wat nie net werk nie, maar ook die beste voldoen aan die verlangde spesifikasies binne gegewe begrensings. Verder moet die ontwerp prakties realiseerbaar en soms goedkoop wees.

Genetiese algoritmes vind oplossings sonder enige kennis van die multi-dimensionele ruimte waarin die oplossing gesoek word. Die genetiese algoritme se doel is om ontwerpvoorstelle te genereer wat deur 'n kostefunksie getoets word teen die spesifikasies en beperkings waaraan die ontwerp moet voldoen. Die kostefunksie vorm die kern van die ontwerpse proses en verskaf die nodige terugvoer aan die genetiese algoritme. Die kostefunksie vervat al die relevante wiskundige en heuristiese

reëls benodig vir die evaluering van die ontwerpvoorstelle.

2. AGTERGROND

2.1 Genetiese algoritmes

Genetiese algoritmes is alreeds aangewend in die oplos van verskeie wiskundige en niewiskundige probleme, optimalisering van strategieë en stelsels, seinverwerking en beheerstelsels.^{2,3}

Die belangrikste voordele van genetiese algoritmes is:^{3,7}

- Die stelsel of probleem hoef nie voorspelbaar te wees nie. Baie praktiese voorbeelde bestaan waar die stelsel nie wiskundig beskryf kan word nie.
- Funksiekontinuiteit is nie 'n voorwaarde nie. Baie wiskundige tegnieke vir optimalisering vereis dat die probleem 'n kontinue funksie moet wees, maar dis nie die geval vir genetiese algoritmes nie.
- Die probleem-karakteristieke en -parameters hoef nie konstant te bly met tyd nie. Dit beteken genetiese algoritmes is selfaanpassende prosesse.

Die sentrale doel van 'n genetiese algoritme is die verbetering van 'n populasie of versameling voorgestelde oplossings se globale fiksheid teenoor die fiksheid van vorige generasies. Die fiksheid van 'n oplossing is 'n waarde wat aan die oplossing toegeken word deur 'n kostefunksie, sodat die fiksheid 'n indikasie kan wees van hoe effektief 'n bepaalde oplossing is.

Die kostefunksie se besonderhede word in afdeling 3.3 bespreek.

2.2 Induktor-ontwerp

Magnetiese komponente vir gebruik in drywingselektroniese opstellings, soos induktors en transformators, is selde beskikbaar as volledige komponente. Hierdie komponente word gevolglik na behoefte saamgestel uit 'n aantal windings van 'n spesifieke dikte geleier op 'n spoelvorm vir 'n spesifieke kerngrootte. 'n Illustrasie van 'n tipiese E-kern en spoelvorm word in figuur 1 getoon.

As 'n induktor in 'n drywingselektroniese baan benodig word, moet die induktor eerstens vir die spesifieke toepassing ontwerp word, byvoorbeeld as energiestoor in 'n spêrkapper of as 'n smoorpoel. Die gewenste induktansiewaarde, toelaatbare weerstand, asook die bedryfsfrekwensie en stroomvermoë volg direk uit die toepassing.^{8,9}

Verdere begrensings soos maksimum temperatuurstyging, toelaatbare volume of massa en selfs koste kan ook gespesifiseer word, onderhewig aan die beoogde einddoel. Die ontwerp van induktors behels dan dat bepaal moet word watter kerngrootte, geleierdeursnee, aantal windings en luggapinggrootte gebruik moet word om 'n induktor wat aan die volledige reeks spesifikasies voldoen, te realiseer. Hierdie proses is 'n sterk funksie van die beskikbare materiale. Veral in ontwikkelende lande is dit dikwels so dat daar nie gereedelik toegang gekry kan word tot gespesifiseerde materiale ensovoorts nie.

Konvensionele ontwerpmetodes vir induktors behels normaalweg die seleksie van 'n kern, waarna die aantal windings wat nodig is om die gespesifiseerde induktansie te verkry, bereken moet word.^{8,9} In figuur 2 word die tipiese ontwerp-proses voorgestel.

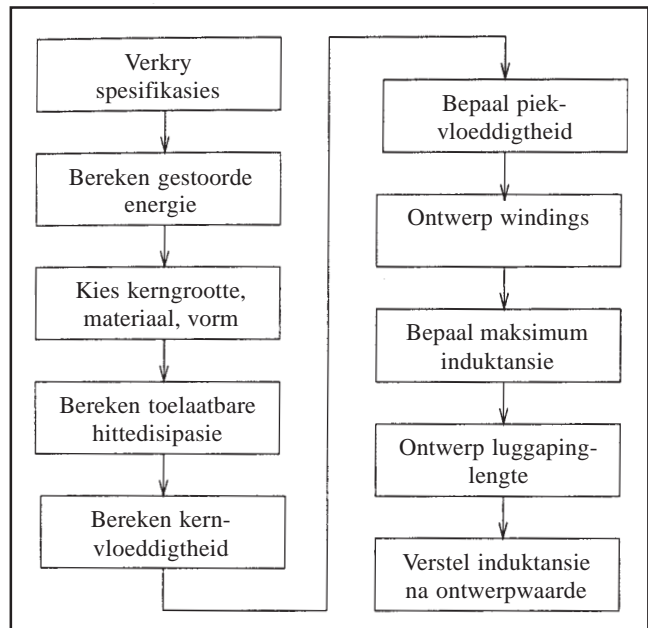
Die aanvanklike kernkeuse word gemaak op grond van die energie wat die induktor moet stoor. Die energie is afhanklik van die verlangde induktansie en die stroom wat die induktor moet kan gelei. Kernkeuses word ook beïnvloed deur die kern se drywingsdissipasieremoë.

Die minimum toelaatbare deursnee van die geleiers word bepaal op grond van die gespesifiseerde stroom wat die induktor

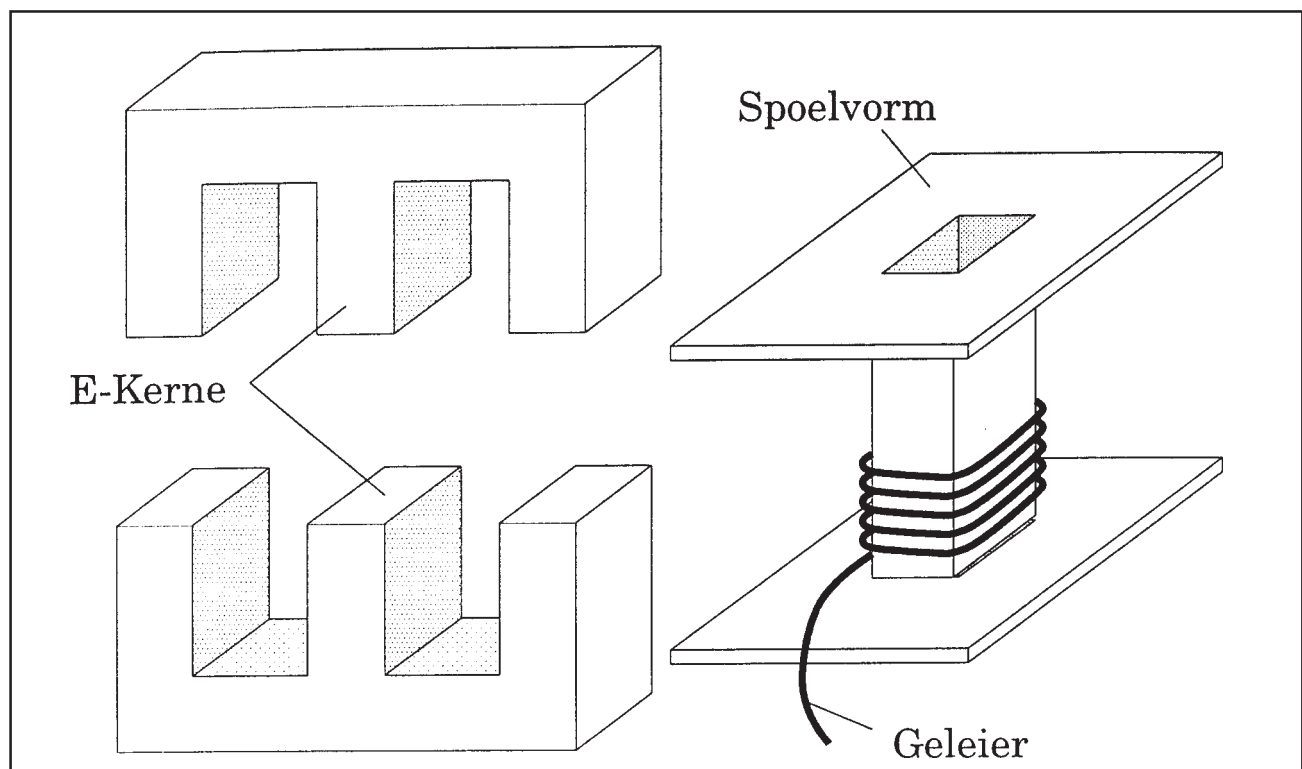
moet hanteer. 'n Stroomdigtheid van tipies 6 A/mm^2 word gebruik hiervoor. Vir elke geleier en kernkombinasie bestaan daar 'n maksimum aantal windings wat op die kern se spoelvorm sal pas.

Die maksimum induktansie wat die betrokke kernkeuse sou kon verskaf as geen luggaping in ag geneem word nie en as die spoelvorm vol windings is, word bepaal om te verseker dat dit hoër is as die verlangde induktansie. Dan kan die luggaping se lengte bepaal word, sodat die verlangde induktansie verseker word.

As enige fisiese beperkings tydens enige van die stappe van



Figuur 2: Konvensionele ontwerp-prosedure vir die ontwerp van induktore.



Figuur 1: Uitleg van die komponente waaruit 'n induktor bestaan.

die ontwerp oorskry word, moet 'n nuwe kern gekies word en die ontwerpproses herhaal word.

Terwyl die induktor ontwerp word, is dit moeilik om die koste van die kerne en geleiers so laag as moontlik en die induktor se fisiese grootte so klein moontlik te hou. Al die oorwegings soos die kernmateriaal se frekwensiekarakteristieke, verliese en permeabiliteite by verskillende veldsterktes speel deurgaans 'n belangrike rol in al die berekeninge.^{8,9,10,11} Dit is om hierdie rede dat rekenaarsteun en spesifiek genetiese algoritmes oorweeg is as tegniek vir optimale induktor-ontwerp.

Hier is nie werklik sprake van 'n optimale induktor in die klassieke sin van die woord nie. Eerder dui optimaal hier op die beste induktor wat steeds onder die verlangde spesifikasies met behulp van die komponente voor hande gekonstrueer sou kon word. Veral in die Suid-Afrikaanse konteks waar beskikbaarheid van nuwe materiale dikwels 'n probleem is, kan programmatuur die ontwerper in staat stel om meer suksesvolle ontwerpe te vind onder die begrensing van wat wel beskikbaar is.

3. ONTWERP MET GENETIESE ALGORITMES

Die ontwerpbenadering met genetiese algoritmes behels die evaluering van talle willekeurige ontwerpvoorstelle met behulp van 'n kostefunksie. Die fiksheid van elke ontwerp word deur die kostefunksie bepaal en aan die ontwerpvoorstel toegeken.

Die beste voorstelle word verbeter deur afsonderlike eienskappe van die oplossings te kombineer (kruising) of deur willekeurige veranderinge aan te bring (mutasie). Dan word die resultaat (nuwe voorstel) geëvalueer en die proses herhaal, totdat die beste oplossing in die populasie konvergeer na 'n oplossing.^{3,12}

3.1 Genetiese komponente en bewerkings

Wanneer genetiese algoritmes in praktiese toepassings gebruik word, word begin met 'n willekeurig saamgestelde populasie, bestaande uit 'n aantal kompeterende oplossings wat aan die

hand van die probleem gebring word. Nuwe oplossings word verwerk deur genetiese manipulasies op die bestaande oplossings uit te voer. 'n Objektiewe doeltreffendheidstoets word gebruik om die fout of fiksheid van die voorgestelde oplossings te bepaal, en 'n seleksiemeganisme bepaal watter oplossings as ouers van toekomstige oplossings mag optree.

Verdere komponente kan bygevoeg word wat die proses monitor om parameters soos mutasiewaarskynlikheid of populasiegrootte te verstel vir sekere konvergensiegevalle. As die proses byvoorbeeld konvergeer, bestaan die moontlikheid dat die oplossing slegs 'n lokale optimale punt en nie die globale optimum is nie. Deur die mutasiewaarskynlikheid te verhoog sal die proses ander verwyderde oplossings kan vind.

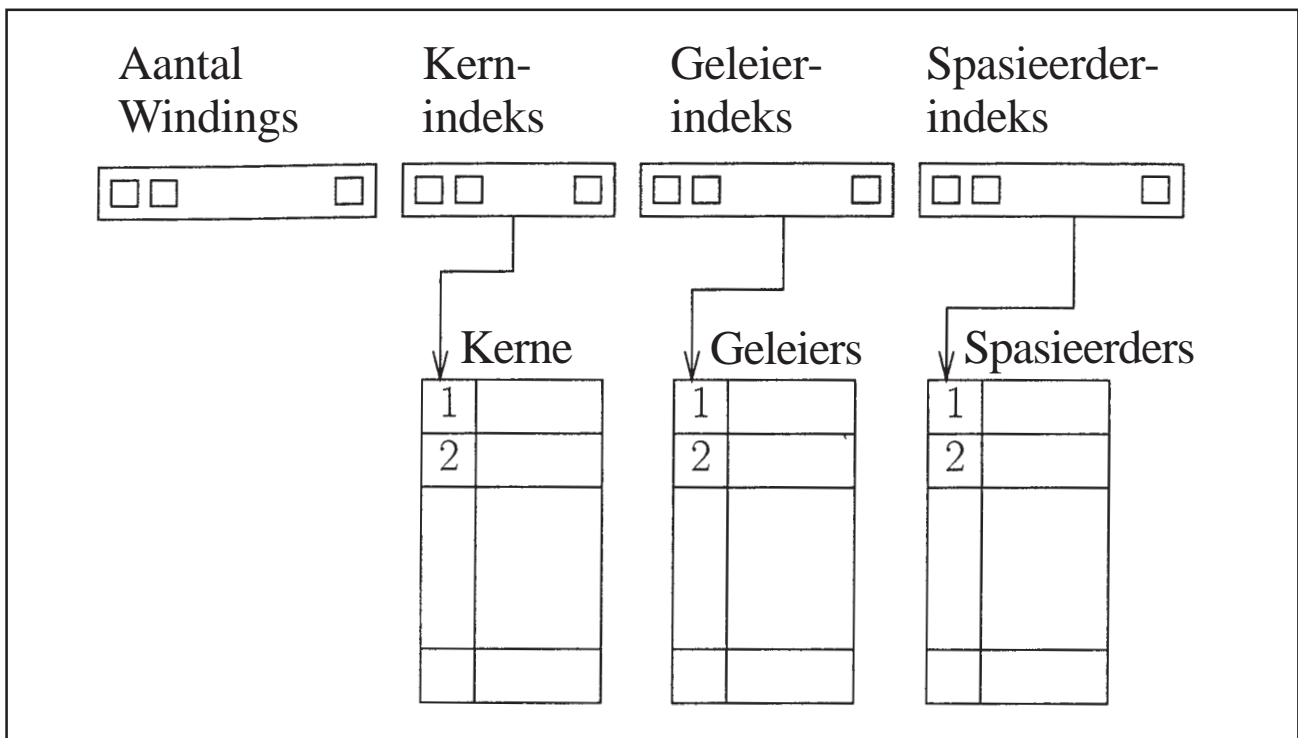
3.1.1 Chromosoom

Elke voorgestelde oplossing in die populasie word verteenwoordig deur 'n chromosoom. Die chromosoom is 'n eindige vektor veranderlikes wat elk uit 'n aantal binêre of ander simbole bestaan. Binêre getalvoorstellings is gerieflik om te manipuleer met behulp van rekenaars. Hierdie veranderlikes is normaalweg die parameters van die bepaalde ontwerp.

Vir ontwerpdoeleindes sal die chromosoom tipies, soos in figuur 3 voorgestel, uit ontwerpparameters en komponent-databasisindekse bestaan. Dit beteken 'n chromosoom verteenwoordig 'n volledige ontwerpvoorstel. In die geval van die induktor sal die ontwerpparameter van die chromosoom die aantal windings wees en die drie databasisindekse sal verwys na kern-, geleier- en luggapingkeuses.

Die lengte van die chromosoom word aangepas volgens die groottes van die databasisse, of die parameters volgens 'n grootte wat voldoende sal wees.

'n Chromosoom wat dus soos volg lyk (0111010011 1001 01 00), stel die volgende voor: 'n ontwerp van 0111010011₂ = 467 windings, op kernkeuse 1001₂ = 9, die E55/21-kern volgens tabel 3, met geleiers van indeks 01₂ = 1, die 0,25 mm deursnee-geleier volgens tabel 4 en geen luggaping, 00₂ nie.



Figuur 3: Die samestelling van 'n chromosoom vir induktor-ontwerpdoeleindes.

3.1.2 Populasie

'n Populasie is 'n versameling ontwerpopslossings, wat voorgestel word deur chromosome.

Elke ontwerp het geassosieerde fiksheidswaardes wat aangewend word in die opstel van waarskynlikheidstabelle. Die ontwerpe kan gerangskik word volgens die fiksheidswaardes, sodat die optimaliseringsproses se vordering gemonitor kan word. Die fiksheidswaardes word deur 'n kostefunksie bepaal, wat die ontwerp se parameters neem en daaruit die fiksheid van die ontwerp bereken (sien 3.3).

Die populasie het 'n beperkte grootte wat tot gevolg het dat beter oplossings die swakker oplossings moet vervang sodat swakker oplossings kan uitsterf.

Oplossings word vir verdere bewerkings gekies deur van 'n waarskynlikheidstabel gebruik te maak: 'n tegniek wat as roulette-wiel-seleksie bekend staan.

In 'n poging van N oplossings het die i -de oplossing, c_i met fiksheid, $f(c_i)$, die waarskynlikheid, p_i , om gekies te word vir verdere bewerkings.³

$$p_i = \frac{f(c_i)}{\sum_{k=1}^N f(c_k)} \tag{1}$$

In figuur 4 word die basiese elemente van 'n populasie uiteengesit. Die chromosome van die oorspronklike populasie met willekeurige waardes, nadat dit die eerste keer gerangskik is, lyk soos in tabel 5. Die eerste en tweede generasies van die sessie word in tabel 6 en 7 respektiewelik getoon. Die eerste generasie is bekom deur mutasie toe te pas op al die lede van die oorspronklike populasie. Die tweede generasie is bekom deur kruising op al die lede van die eerste generasie.

3.1.3 Mutasie

Mutasie is een van die bewerkings wat op 'n chromosoom uitgevoer kan word. Dit behels dat willekeurige veranderinge

aangebring word in die chromosoom deur 'n paar van die bisse se waardes te verander, soos in tabel 1 geïllustreer. 'n Vasgestelde aantal bisse sal verander word en die posisies van die betrokke bisse word willekeurig gekies.³

Mutasie verhoed konvergensie na lokale optimale punte en dwing inagneming van ander verwyderde moontlikhede op die populasie af.

Die waarskynlikheid dat mutasie plaasvind word laag gehou, omdat algoritmes met mutasie nie baie vinnig konvergeer nie. Aanvanklik word vinnige konvergensie verlang en dus moet kruising 'n heersende effek handhaaf. Tegnieke bestaan waar mutasiewaarskynlikhede verhoog word sodra die proses konvergeer om te verseker dat dit wel die globale optimale punt is.³

Mutasie is geneig om resultate te lewer wat ver verwyder is van die fiksheid van die chromosoom waarop mutasie uitgevoer is.

3.1.4 Kruising

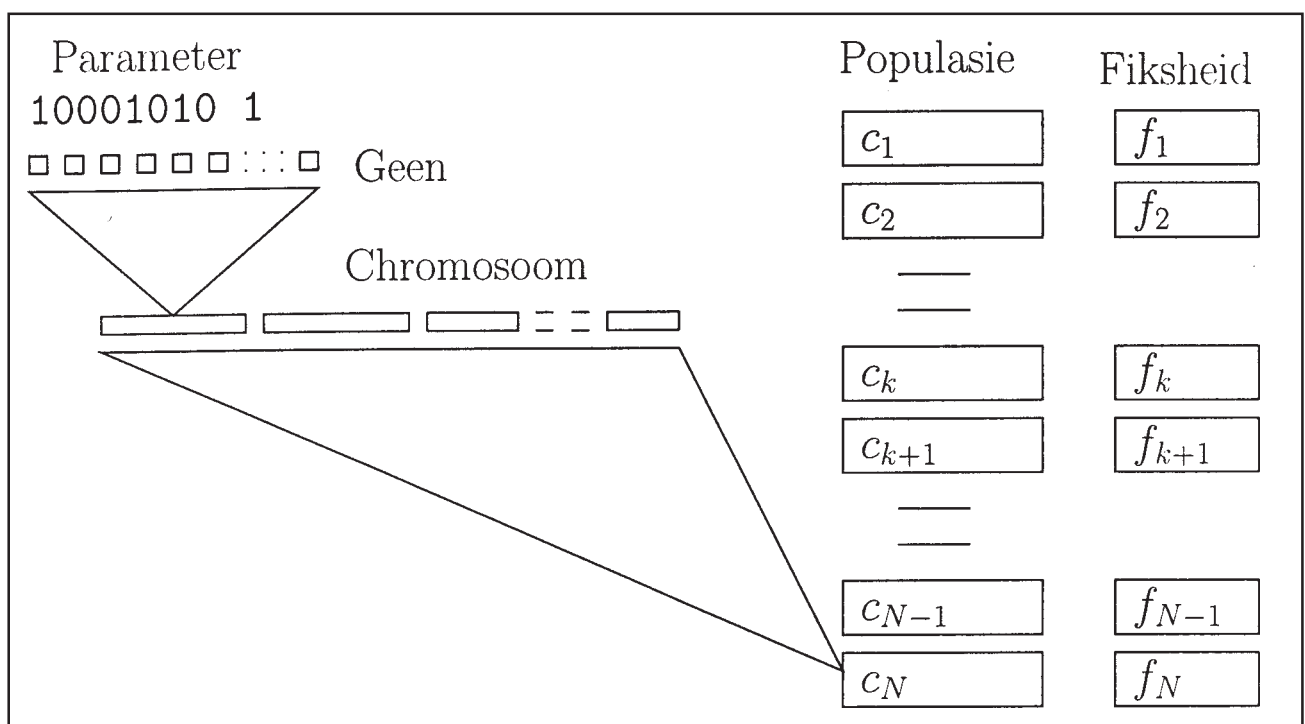
Kruising is 'n binêre bewerking wat afsonderlike eienskappe van twee chromosome kombineer om 'n kind te verkry, wat normaalweg uniek is, maar tog eienskappe van beide die ouers besit. Kruising word in tabel 2 geïllustreer.

Kruising is toegepas in die ontwerp van induktors deur vier kruispunte op die chromosoom willekeurig te kies.

Kruising veroorsaak sterk konvergensie in die populasie, maar dit is konvergensie na 'n lokale optimum. As slegs kruising herhaaldelik op die populasie uitgevoer word, word die globale optimale oplossing dus moeilik gevind. Om dit te voorkom word kruising gereeld afgewissel met mutasie om verwyderde moontlikhede in ag te neem, alhoewel kruising in hierdie toepassing die primêre evolusionêre bewerking bly.

3.2 Die algoritme

Die eerste stap in die algoritme is om die populasiegrootte en chromosoomvorm te definieer op grond van die perke en



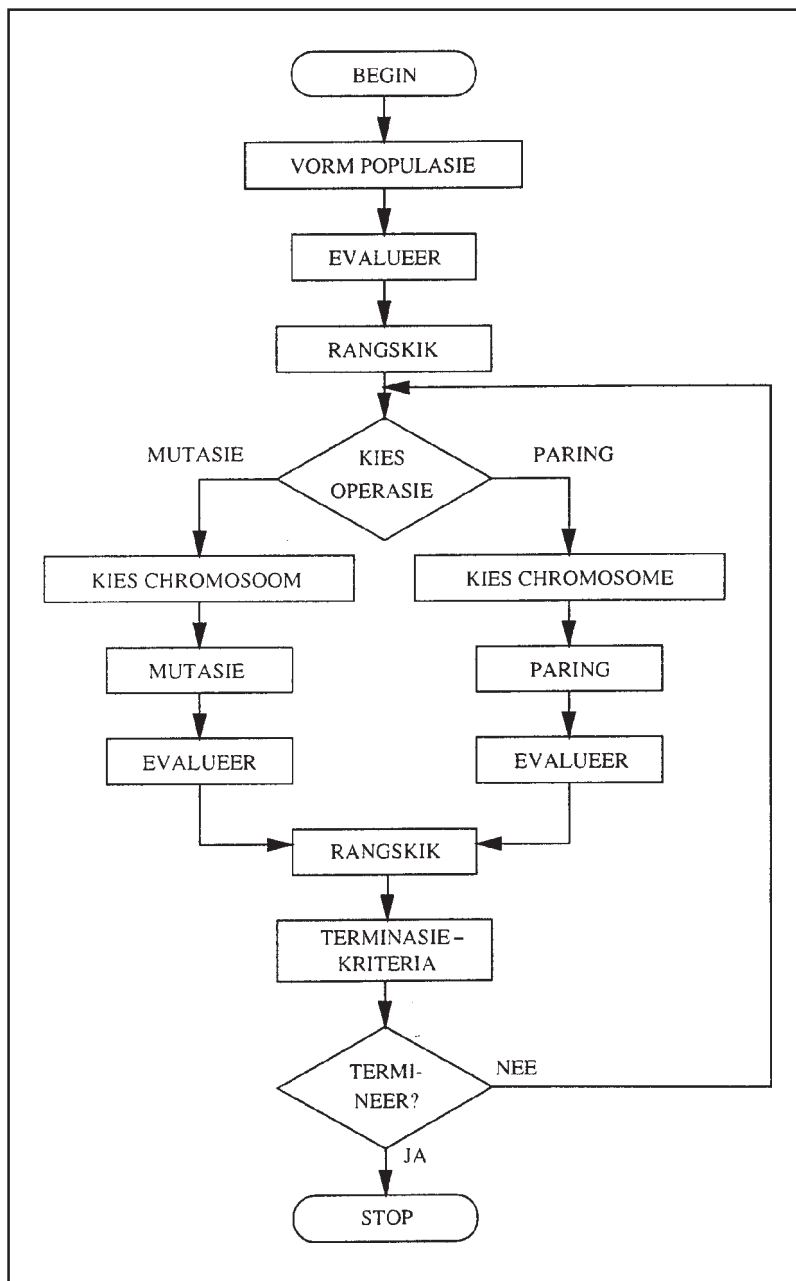
Figuur 4: Komponente van 'n populasie.

vereistes waaraan die ontwerp moet voldoen. Dit behels grootliks die definiëring van die chromosoomvorm soos in afdeling 3.1.1 behandel.

Die beginwaardes vir die chromosome kan vooraf gedefinieer wees, of uit 'n populasie van 'n vorige poging bestaan, of dit kan willekeurige waardes bevat. Hierdie populasie moet eers geëvalueer word en dit beteken die roep van die kostefunksie wat aan die probleem gekoppel is.

Die populasie word vervolgens gerangskik en 'n verwysing na die chromosome word in die waarskynlikheidsdistribusie-tabel opgeskryf met 'n waarskynlikheidsdistribusiewaarde, wat bepaal word deur vergelyking 2 te gebruik,

$$P_i = \sum_{k=1}^i P_k \quad (2)$$



Figuur 5: Die vloedidiagram vir die genetiese algoritme.

Die tabel word gebruik om die uniforme distribusie van die rekenaar se willekeurige getalle om te skakel in 'n distribusie wat sorg dat 'n oplossing se waarskynlikheid om bevorder te word in verhouding staan met die oplossing se fiksheid. Dit is beter bekend as Roulette-wiel-seleksie.¹²

Dit is belangrik om in ag te neem dat enige ontwerp in die populasie 'n kans het om gekies te word en dat die kandidaat slegs in die binêre bewerking gebruik word om 'n nuwe kandidaat te verwek en self geen verandering ondergaan nie. 'n Kandidaat sal slegs verplaas word as dit die swakste fiksheid het.

Mutasie en kruising is nie gelyktydig in dieselfde siklus uitgevoer nie, soos in figuur 5 aangetoon, alhoewel mutasie gewoonlik direk na kruising op die kinders toegepas word. Mutasie behoort 'n baie klein waarskynlikheid van voorkoms te handhaaf, tipies laer as 3%, maar daar mag uitsonderings wees. Vir induktor-ontwerp is hier van 'n klein populasie (80 lede) gebruik gemaak en onder die omstandighede het 5% mutasie-waarskynlikheid die vinnigste, baie goeie oplossings gelever.

Vervolgens word 'n chromosoom gekies waarop die betrokke operasie uitgevoer word, waarna die resultaat geëvalueer word en in die populasie geplaas word, as die resultaat beter is as die swakste in die populasie.

Tydens die verloop van die genetiese algoritme, word rekord gehou van die veranderinge in die populasie, sodat besluit kan word of die genetiese algoritme moet termineer of herhaal. Die terminasiekriteria moet aangepas kan word deur die gebruiker, want dit mag dalk van probleem tot probleem verskil, maar die werking behels dieselfde metode.

As die gespesifiseerde aantal beste oplossings in 'n populasie, vir 'n spesifieke aantal iterasies dieselfde bly, het die oplossing moontlik na 'n optimale antwoord gekonvergeer.

3.3 Die kostefunksie

Die sleutel tot die proses se sukses is seleksie en eliminasië. Seleksie vind plaas as die sterkste lede van 'n populasie gekies word vir verbetering. Eliminasië van swak lede sorg dat hierdie lede nie 'n verswakking in toekomstige populasies teweeg bring nie.

'n Ontwerp se geassosieerde fiksheidswaarde is 'n aanduiding van hoe goed die ontwerp aan die spesifikasies voldoen. 'n Ontwerp word baie laag geëvalueer as dit enige van die beperkings, waaraan die ontwerp moet voldoen, oorskry.

Die kostefunksie toets deurgaans die ontwerpvoorstel teen die beperkings wat gestel is en enige praktiese beperkings wat uit die voorstel self spruit. Die toets wat die ontwerp moet slaag is as volg:

1. Die geleiers wat gebruik word moet die gespesifiseerde stroom kan hanteer.
2. Vir elke keuse van kern en geleier is daar 'n maksimum aantal windings wat in die kern se venster sal pas, en dit is dus 'n beperking op die windings.
3. Verder mag enige ander vooraf gespesifiseerde beperkings bestaan waaraan die ontwerpvoorstel moet voldoen. Dit sluit in die bogrens vir die toelaatbare weerstand van die induktor en die toelaatbare toleransie in die induktansie.

As die ontwerpvoorstel enige van hierdie toetse nie slaag nie, word die oplossing met 'n fiksheid van nul geëvalueer. As 'n ontwerpvoorstel al die toetse, wat die deskundige stelsel uitvoer, slaag, kan die fiksheid van die ontwerp vervolgens geëvalueer word.

Vir die evaluasiedoeleindes moet die induktansie (L) en weerstand (R) vanuit die kern- en geleierkeuses en aantal windings bereken word. Dit word bereken deur middel van analitiese vergelykings soos algemeen gebruik word vir induktor-ontwerp.

Die finale evaluasie behels die bepaling van die minimum kwadraatfout wat vir die gebruik in genetiese algoritmes gedefinieer word as volg,

$$d = \sqrt{\sum_{i=1}^M \alpha_i (r_i - s_i)^2} \quad (3)$$

As r_i 'n verlangde ontwerpsspesifikasie is vir $i = 1, 2, \dots, M$, waar M die aantal spesifikasies vir die ontwerp is en s_i die voorgestelde ontwerp se spesifikasie is, dan is d die afstand tussen die voorgestelde oplossing en die verlangde spesifikasies. α_i is die gewig wat die betrokke spesifikasie in die ontwerp dra. d moet ideaal baie klein wees as die ontwerpvoorstel goed is.

Drie spesifikasies wat as mikpunte dien in induktor-ontwerp, wat tydens hierdie ondersoek suksesvol geïmplementeer is, is die induktansie en weerstand by die bedryfsfrekwensie, en die vulling van die kernvenster met geleierwindings.

Die waardes in die chromosoom impliseer relevante data vir die berekening van die induktansie (L), weerstand (R) en die aantal windings (N) en is direk van die chromosoom afkomstig. $R = 0$ word vir meeste toepassings aanvaar.

Die kern se venster area (A_w) is die area in die kern wat beskikbaar is vir geleiers. A_{cu} is die gekose geleier se deursnit-area en k is die pakkingsfaktor wat 'n aanpassing bewerkstellig vir die spasies tussen geleierwindings.

Vir die geval van induktor-ontwerp word vergelyking 3 soos volg omskryf as $\alpha_i = 1$ aanvaar word:

$$d = \sqrt{\Delta L^2 + \Delta R^2 + \left(A_w - \frac{N A_{cu}}{k} \right)^2} \quad (4)$$

ΔL en ΔR is in dié geval die verskil tussen die gespesifiseerde en ontwerpte induktansie en weerstand respektiewelik.

Om die fiksheid (f) van 'n ontwerp te bereken word die fout (d) genormaliseer deur die fout te deel deur die spesifikasielengte, gedefinieer as

$$D = \sqrt{L^2 + R^2 + A_w^2} \quad (5)$$

en van 1 af te trek.

$$f = 1 - \frac{d}{D} \quad (6)$$

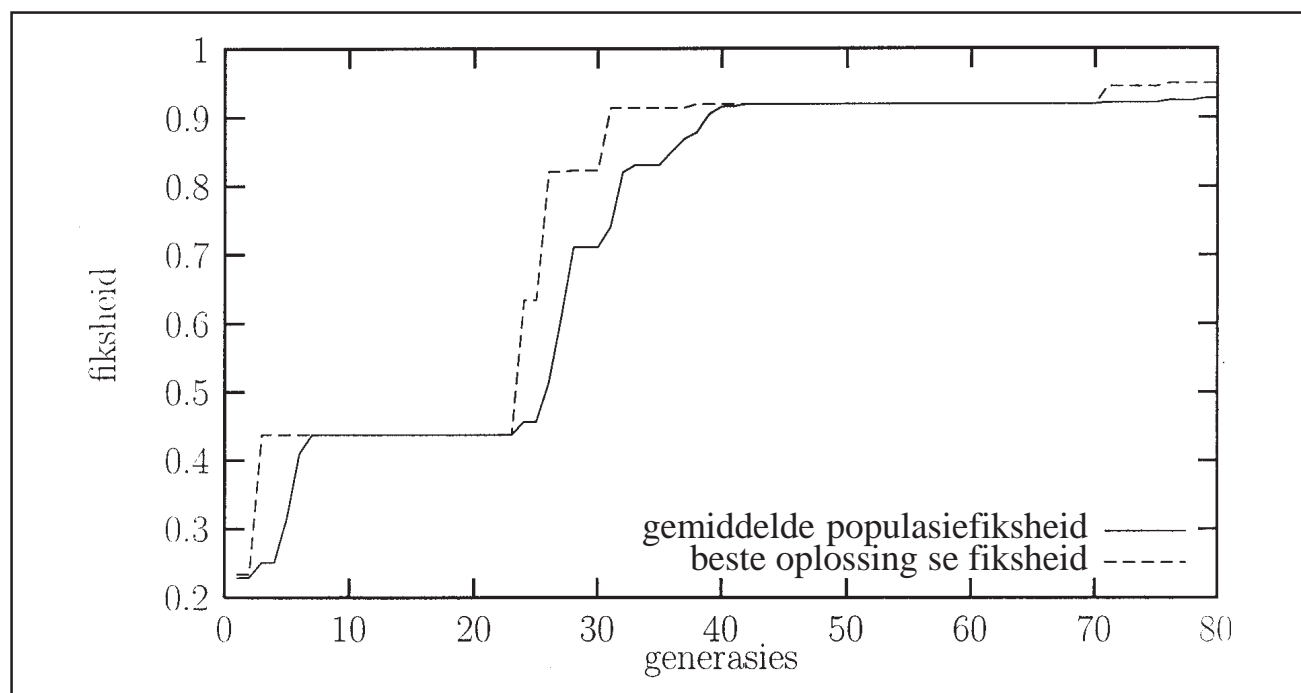
Dit is gedoen om 'n fiksheid tussen 0 en 1 te verkry wat Roulette-wiel-seleksie vergemaklik.

Verdere optimaliseringsdoelwitte soos koste, volume, massa, ensovoorts kan maklik by vergelykings 4 en 5 betrek word, indien dit verlang word.

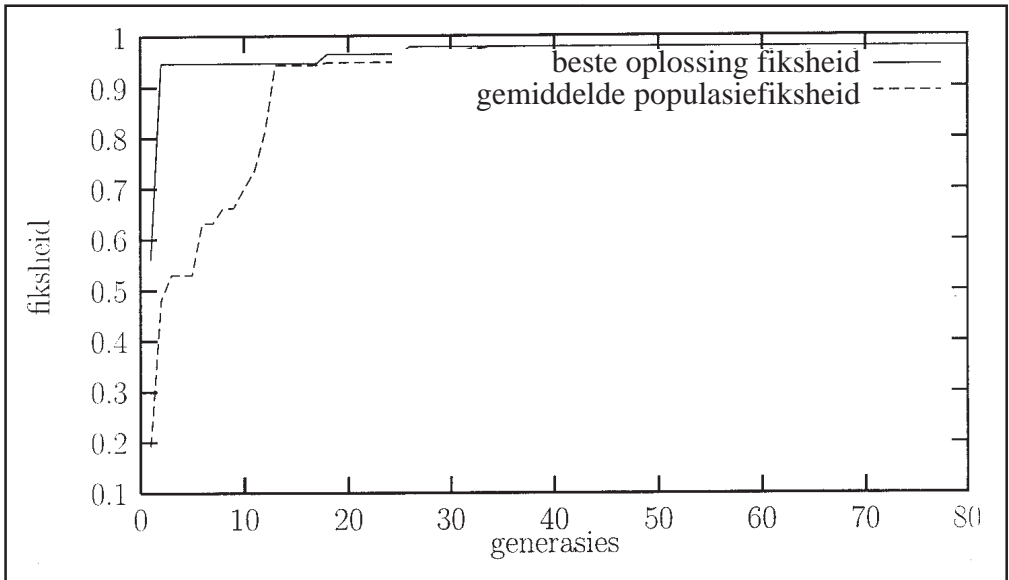
4. RESULTATE

Die program is in C++ geskryf en maak van 'n SQL-databasisverbinding gebruik vir kern-, materiaal- en geleierdata, omdat 'n lokale inventaris van onmiddellik beskikbare komponente tipies op so wyse gehandhaaf mag word.

Beskou die ontwerp van 'n $470 \mu\text{H}$ -induktor as 'n voorbeeld. Die spesifikasies vir die ontwerp en die resultate vir twee ontwerpvoorstelle word in tabel 8 gelys. Figuur 6 en 7 toon aan hoe die fiksheid van die beste ontwerp in die populasie verander tydens die program se ontwerpssessie. Figuur 8 en 9 wys hoe

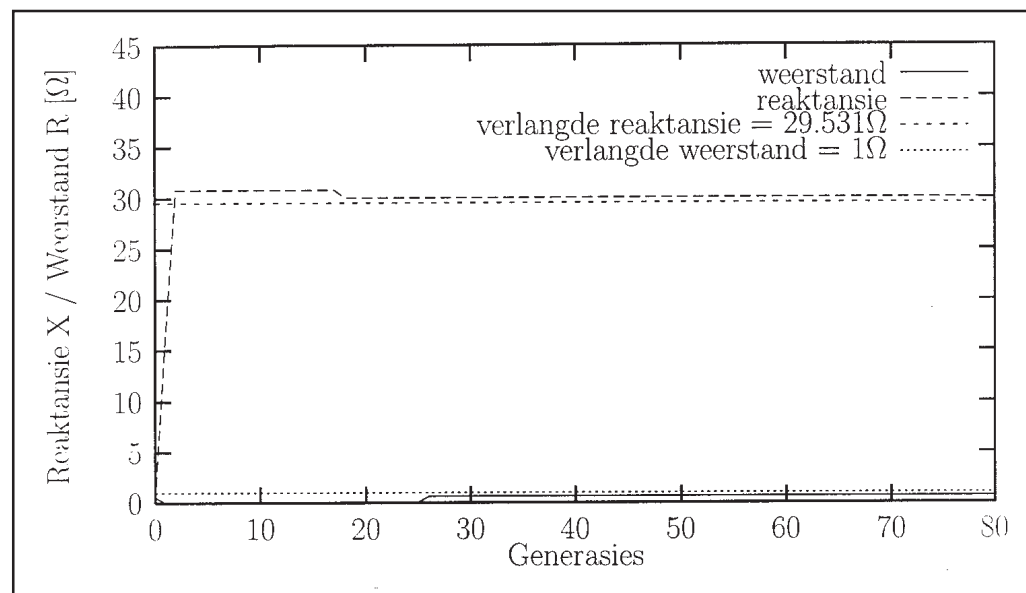
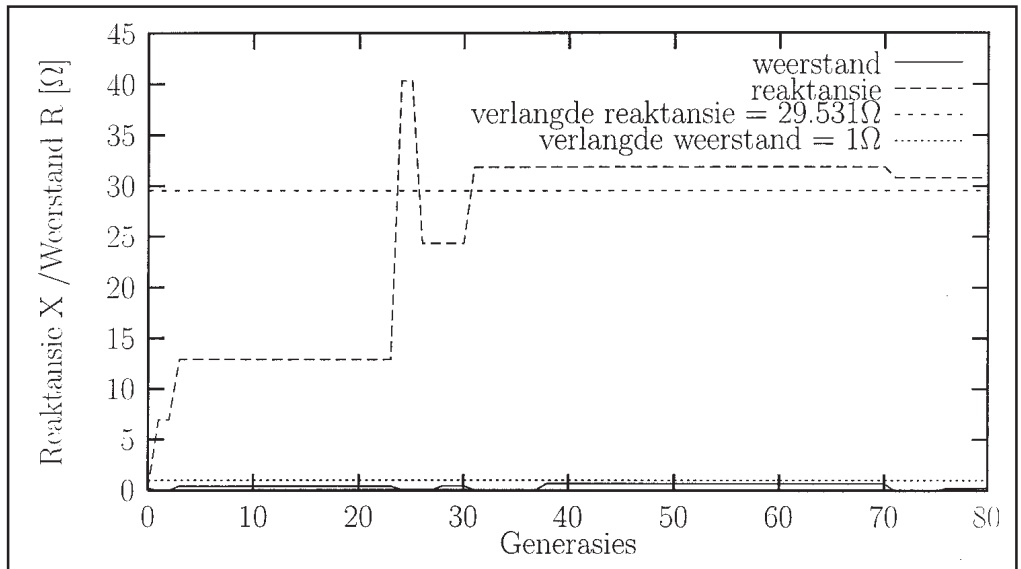


Figuur 6: Die beste en gemiddelde fikshede oor 80 iterasies vir die ontwerp van die $470 \mu\text{H}$ -induktor.



Figuur 7: Die beste en gemiddelde fikshede oor 80 iterasies vir die tweede ontwerpessie van die 470 μ H-induktor.

Figuur 8: Die reaktansie en weerstand teenoor iterasies.



Figuur 9: Die reaktansie en weerstand teenoor iterasies vir die tweede ontwerpessie.

Tabel 1 Mutasie op 'n chromosoom

voor mutasie:	0000000000011101	0101	010	00
na mutasie:	0001000000001101	0111	011	00

Tabel 2 Kruising van twee chromosome

voor kruising:	0000000011110011	1100	011	01
	0000110001011001	0101	011	10
na kruising:	0000000011111001	0100	011	11
	0000110001010011	1101	011	00

Tabel 4 Die geleierdatabasis bevat hier slegs 'n aantal gerieflike draaddiktes, maar kan uitgebrei word om 'n meer volledige inventaris te weerspieël

Indeks	D
0	0.15
1	0.25
2	0.75
3	1.5

Tabel 3 Die kerndatabasis bevat slegs kerne wat deel van die huidige inventaris uitmaak en kan maklik aangepas word

Indeks	Kode	Materiaal	θ_{eff}	$A_{venster}$	A_{kern}	l_e	l
0	EF12.6	F44	1380	12	13	29.6	30.6
1	EF16	F44	1428	21.6	20.1	37.6	33.0
2	E20/10/5	F44	1515	25	31	43	30
3	EF20	F44	1385	34.8	33.5	44.9	39
4	EF25	F44	1485	56.4	52.5	57.5	48
5	E25/9.5/6	F44	1510	56.2	38.1	48.7	41
6	E30/30/7	F44	1605	83.4	60	67	44.8
7	E42/15	F44	1490	180	181	97	88
8	E42/20	F44	1470	180	240	97	98
9	E55/21	F44	1550	280	355	123	110

Tabel 6 Na mutasie op die beter helfte van die populasie uitgevoer is, die swakker helfte met die resultate vervang is en die eerste generasie gerangskik is

Chromosoom	Fiksheid
0000000000001011 0101 011	0.560186
0000000000001010 0001 010	0.501242
0000000000000010 0101 011	0.463073
0000010010100101 0001 010	0.373138
0000100010111011 0011 011	0.0146328
000000000010001 0001 011	0.0089145
0000000010010101 0111 011	0
0000000000100110 0111 001	0
0000001101110000 0111 010	0
0000000001011110 1001 001	0
0000000000100011 0101 000	0
0000000000100000 0111 010	0
0000010101000110 0001 010	0
0000000000011111 0111 000	0
0000000011000000 0100 001	0
0000000000110000 0010 010	0
0000001000100101 1001 001	0
0000000100001000 0011 001	0
0000000110100111 0011 010	0
0000001100111110 0011 010	0

Tabel 5 Die aanvanklike willekeurige populasie nadat dit volgens fiksheid rangskik is

Chromosoom	Fiksheid
0000000000001011 0101 011	0.560186
0000000000001010 0101 011	0.463073
0000000000000010 0001 011	0.0089145
0000010010100101 1001 001	0
0000100010111011 0101 000	0
000000000010001 0111 011	0
0000000010010101 0100 001	0
0000000000100110 0010 010	0
0000001101110000 0111 000	0
0000000001011110 0011 010	0
0000000000100011 1001 011	0
0000000000100000 1000 011	0
0000010101000110 0111 011	0
0000000000011111 0100 010	0
0000000011000000 0000 010	0
0000000000110000 0110 010	0
0000001000100101 0011 000	0
0000000100001000 0011 001	0
0000000110100111 0000 001	0
0000001100111110 0110 001	0

Tabel 7 Nadat lede van die beter helfte gekruis is en in die swakker helfte in vervang is: die tweede generasie

Chromosoom			Fiksheid
0000000000001111	0101	010	0.946396
0000000000001011	0101	010	0.560324
0000000000001011	0101	011	0.560186
0000000000001011	0101	011	0.560186
0000000000001111	0001	010	0.501242
0000000000001010	0101	011	0.463073
0000000000001010	0101	011	0.463073
00000000000011011	0001	010	0.373138
00000000000011011	0001	010	0.373138
0000000000000010	0011	011	0.0146328
0000000000000010	0011	011	0.0146328
0000000000000010	0001	011	0.0089145
0000000000000001	0001	011	0.0022287
0000000000000001	0001	011	0.0022287
0000001010111101	0111	010	0
0000000111010011	0111	001	0
0000000000000000	0001	011	0
00000000000010001	0111	011	0
0000100000000010	1001	001	0
00000000000011011	0101	010	0

Tabel 8 Twee oplossings se spesifikasies vir dieselfde ontwerp

	Spesifikasie	Ontwerp 1	Ontwerp 2
Induktansie	470 μ H ²	490 μ H	478 μ H
Reaktansie	9.531 Ω	30.788 Ω	30.0455 Ω
Weerstand	1 Ω	0.253 Ω	0.64 Ω
Stroom	0.1A	0.278A	0.106A
Q-faktor	29.5	121.69	46.944
Ontwerpopslossings			
Aantal windings		15	16
Draaddeursnee		0.25	0.15
Kern		EF25	E25/9.5/6
Materiaal		F44	F44
Fiksheid		0.94961	0.978749

Tabel 9 Resultate van 'n paar induktor-ontwerpe

Induktansie (μ H)			N	D (mm)	kern	materiaal	gaping (mm)
Verlang	Ontwerp	Gemeet					
220	223	271.7	35	0.75	E25/9.5/6	F5A	0.1
870	851	849	56	0.75	E30/30/7	F44	0.1
450	451	610	43	0.75	EF25	F9	0.1
1800	1780	1800	81	0.75	E30/30/7	F44	0.1
1000	990	1016	64	0.75	EF25	F44	0.1
100	102	297	68	0.75	EF25	F44	1
470	474	490	51	0.75	E25/9.5/6	F5A	0.1

die reaktansie en weerstand na die verlangde konvergeer. Omdat genetiese algoritmes van verskeie willekeurige prosesse gebruik maak, sal 'n ontwerpessie nooit presies herhaal nie en gevolglik sal daar kere wees wat die proses vir langer moet loop om dieselfde optimale oplossing te verkry. Vir figuur 6 se geval is die globale optimum nie binne 80 generasies gevind nie, terwyl die geval van figuur 7 binne 30 generasies 'n beter oplossing verkry het.

Verdere ontwerpe word in tabel 9 uiteengesit om die uitwyking van die verlangde, ontwerpte en gemete induktansie duidelik te maak. Die ontwerpte induktorwaardes is die beste voorstel van die genetiese algoritme, en die gemete induktansie is verkry deur die induktors te bou en by die gespesifiseerde frekwensie te meet.

Hieruit blyk dat die gekonstrueerde induktors nie so naby aan ideaal is as wat verlang sou word nie, alhoewel die kostefunksie die mees teoretiese ontwerp gekies het. Die redes vir die uitwyking word toegeskryf aan die groot variasie in kernmateriaaleienskappe. Kernoppervlaktes en -materiaal-permeabiliteit word op datavelle aangedui met tipies 20% toleransie. Die teoretiese modelle wat deur die kostefunksie gebruik word vir evaluasie, is ongelukkig sensitief vir die materiaalparameters wat hier ter sprake is. Hierdie modelle is uit fisika afgelei en is tans algemeen in gebruik.

5. GEVOLGTREKKING

Rekenaargesteunde ontwerp kan gerieflik gerealiseer word deur van 'n eenvoudige genetiese algoritme en deeglike kostefunksie gebruik te maak. Dit skakel iterasies deur die ontwerper grootliks uit. Oplossings gebruik beskikbare komponente en die gebruik van databasisse vergemaklik hierdie proses.

Die proses vind wel prakties realiseerbare, klein, optimale oplossings wat teoretiese eienskappe naby aan die verlange spesifikasies toon. Die belangrikste oorsaak van afwykings in gemete waardes is die swak toleransie in die kernafmetings en veral die kernmateriaaleienskappe ten opsigte van gepubliseerde waardes. Die effek word verminder deur luggapings by te voeg en dit stem ooreen met algemene praktyk.

Die gebruik van hoër mutasiewaarskynlikheid vergroot die moontlikheid dat die globale optimum oplossing gevind sal word, maar verg meer generasies om deur te voer. Laer mutasiewaarskynlikheid gee voorlopige oplossings baie gouer. Die keuse vir 'n mutasiewaarskynlikheid sal baie van die probleem se kompleksiteit afhang.

Meer lede in die populasie verbeter die konvergensie en betroubaarheid van die oplossing ten koste van rekenaargeheue en tyd.

Die proses kan 'n bevredigende induktor-ontwerp lewer en het alreeds die ingenieur se werk vergemaklik. Die praktiese voordele verbonde aan die gebruik van genetiese algoritmes is alreeds duidelik. Verdere koste-optimalisering sal duidelike ekonomiese voordele in die industrie toon.

Toekomstige implikasies van genetiese algoritmes in ontwerp, sluit optimale stroombaan-, masjien- en komponent-ontwerp en ook komponentplasing in.¹³ Ervaring kan nie vervang word nie, maar met roetine-ontwerpsprosedures kan genetiese algoritmes help om goeie ontwerpe te verkry.

6. SUMMARY

6.1 Introduction

Genetic algorithms are a relatively new evolutionary programming technique which have shown promising results in many applications. In electrical and electronic engineering great interest exists in applying genetic algorithms to automation of the design process, especially in CAE.

Design problems like effective optimisation under multiple constraints become complex when working with discrete component sizes, and non-linear and discontinuous component characteristics. Considering these problems, the design aid tool must find a solution which falls within the specified constraints, display certain specific characteristics, and also be economically feasible and practical.

Using an expert system for evaluation, genetic algorithms find optimum solutions without specific knowledge of the problem or solution space. Incorporating an expert system with a genetic algorithm to evaluate the genetic algorithm's design suggestions is the approach that is discussed here.

6.2 Background

6.2.1 Genetic algorithm

The genetic algorithm's goal is to improve a population of solutions, usually for the purpose of optimisation.

Advantages of genetic algorithms above other iterative or mathematical techniques are:

- the problem may be unpredictable,
- it may be discontinuous,
- the problem may be time variant.

6.2.2 Inductor design

Magnetic components for power electronic applications, like inductors and transformers are rarely available as complete components, but are constructed as required, by winding a number of turns of a certain conductor size on a core.

When an inductor is required in a power electronic circuit it must be designed to achieve the inductance, resistance, operating frequency and current capability for the specific application. Other constraints, for example maximum temperature rise, allowed volume, mass and costs, may influence the design as well.

Conventional design methods for inductors require a selection of a core, after which the wire size and number of turns can be calculated. If at any stage the core or wire combination does not meet the requirements of the design specifications, a new core must be selected and the whole

process repeated. The iterative nature of this process is clear and the design solution provided in this way is still not optimised in any way. Such optimisation to minimize losses for example will require further iterations.

From this background the use of genetic algorithms is attractive because of their ability to optimise for multiple constraints.

6.3 Genetic algorithms in design

The design approach involves evaluating a population of design suggestions of which the best are used in genetic manipulations (crossover and mutation) to improve the design. The new candidate designs are evaluated and the process repeats.

6.3.1 Genetic components

When genetic algorithms are used in practical applications, the process starts by generating random solutions to the problem. New solutions are derived from these through genetic alterations, after which an objective effectiveness test is used to calculate the error or fitness of the suggested solution and a random selection mechanism is used to select the parents of future generations.

Monitoring components may be added to automatically adjust the mutation probability or population size, to assure that the solutions do not converge to a local optimum.

Chromosomes: The chromosome is a binary string of a certain length, which represents a design suggestion. For the purpose of design and particularly inductor design, the chromosome consists of component database indices and design parameters. For example for inductor design, the design parameter is the number of turns, and the three indices point to components in core, wire and spacer databases.

Population: A population is a collection of design solutions represented by chromosomes. Every design solution has an associated fitness value which is used in probability distribution tables and the solutions are ranked according to the fitness values. The fitness values are calculated by the fitness function. The population has a limited size which means that new solutions have to replace any weaker solutions in the population. The probability of a chromosome c_i being chosen for further operations in p_i according to equation 1, where $f(c_i)$ is the fitness of chromosome c_i .

Mutation: Mutation is an operation that may be performed on a chromosome and involves small random changes to the chromosome.

A certain number of bits anywhere on the chromosome are inverted during a mutation operation. Mutation prevents convergence to a local optimum point and forces consideration of alternative solutions to assure the final solution is the global optimum.

Crossover: Crossover is a binary operator which combines parts of two chromosomes to construct a new chromosome.

A number of crossover points on the chromosome structure are chosen which will be boundaries to the swapping sections. Crossover assures quick convergence to an optimum solution, which might not be a global optimum. For inductor design purposes four crossover points on the chromosome were randomly selected.

6.3.2 The algorithm

The first step in genetic algorithms is to define the population size and chromosome structure based on the requirements of the design. The initial chromosome values may be defined, acquired from previous runs or random values. This initial population is evaluated first, after which its members are ranked according to their fitness and a probability density table is created.

Candidates for further genetic alterations are randomly chosen from the population, and either mutation or crossover is performed on chosen chromosomes. Stronger candidates have higher probability of being chosen than weaker candidates. The results of the genetic operations are evaluated, and if their fitnesses are better than the weakest solutions in the populations, they replace these weaker solutions.

To adjust the probability of mutation or to terminate the process, convergence must take place. This happens when the best solutions stay the same for a certain number of iterations.

6.3.3 The fitness function

Genetic algorithms work because of selection of the fittest solutions and elimination of the weakest. To determine a solution's fitness a function with access to the component databases is used.

The fitness function first tests whether the solution is practical and within the given constraints. Designs which fail these tests are evaluated with zero fitness and quick extinction is thus ensured.

Further evaluation requires calculating the sum of the square of the errors between the required specifications (r_i) and the suggested solution's specifications (s_i), as shown by equation 3. For inductor design the three specifications used were the inductance and resistance of the inductor, and the optimal use of the core bobbin for the conductors as shown by equation 4.

To achieve a fitness value between 1 and 0, the error d is normalized and then subtracted from 1.

6.4 Results

Consider the design of a 470 μH inductor as an example. The specifications for the design are listed in table 8 with two design solutions. The two sessions did not reach the same solutions since any two sessions do not converge at the same rate, as shown by figures 6 and 7.

More design solutions and their designed and measured inductances are shown in table 9, showing that even though the theoretical values for the designs are satisfactory, the measured values could be better.

6.5 Conclusion

Design aid tools can be developed by using genetic algorithms and a suitable fitness function. The process finds practical solutions with simulated properties very close to the desired specifications. The deviations in measured values from the expected can be ascribed to the tolerance in manufactured cores' permeabilities.

The main advantage of this approach is its ability to design inductors from an ever-changing inventory of components. Further benefits can be gained from this approach by optimising for cost, which may have some impact on industry.

Future trends include the design of optimal circuits, machines and components, and the optimum placing of components on PC-boards. Though experience cannot be replaced, genetic algorithms have clear influence in ensuring optimal designs.

VERWYSINGS

1. S. Goonatilake, S. Khebbal (1995). *Intelligent Hybrid Systems*. (London: John Wiley & Sons).
2. J. Stender, E. Hillebrand, J. Kingdon (1994). *Genetic Algorithms in Optimisation, Simulation and Modelling*. (Amsterdam: IOS).
3. K.F. Man, K.S. Tang, S. Kwong, W.A. Halang (1997). *Genetic Algorithms for Control and Signal Processing*. (Reading, Ma.: Springer).
4. D.H. Wolpert, W.G. Macready (April 1997). No Free Lunch Theorems for Optimization, *IEEE Transactions on Evolutionary Computation*, (1).
5. Jong-Hwan Kim, Hyun Myung (1997). Evolutionary Programming Techniques for Constrained Optimization Problems, *IEEE Transactions on Evolutionary Computation*, (2). July.
6. B. Dengiz, F. Altiparmak, A.E. Smith (1997). Local Search Genetic Algorithms for Optimal Design of Reliable Networks, *IEEE Transactions in Evolutionary Computation*, (3). September.
7. D.E. Goldberg (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. (New York: Addison-Wesley).
8. N. Mohan, T.M. Undeland, W.P. Robbins (1995). *Power Electronics: Converters, Applications and Design*. (New York: John Wiley & Sons).
9. T.M. Undeland, J. Lode, R. Nilssen, W.P. Robbins, N. Mohan (1996). A Single-Pass Design Method for High-Frequency Inductors, *IEEE Industry Applications Magazine*, 44-51. September/October.
10. Colonel Wm.T. McLyman (1997). *Magnetic Core Selection for Transformers and Inductors. A User's Guide to Practice and Specification, second edition*. (New York: Marcel Dekker).
11. Colonel Wm. T. McLyman (1978). *Transformer and Inductor Design Handbook*. (New York: Marcel Dekker).
12. D.B. Fogel (2000) *Evolutionary Computation Toward a New Philosophy of Machine Intelligence, Second Edition*. (Piscataway, New Jersey: IEEE Press).
13. J.R. Koza, F.H. Bennett, D. Andre, M.A. Keane, F. Dunlap (1997). Automated Synthesis of Analog Electrical Circuits by Means of Genetic Programming, *IEEE Transactions on Evolutionary Computation*, 1(2). July.