

Ministerul Învățământului
Universitatea "POLITEHNICA" București
Facultatea de Electrotehnică

STUDIUL CÂMPULUI ELECTROMAGNETIC ÎN MEDII NELINIARE

- Contribuții privind optimizarea dispozitivelor electromagnetice
neliniare -

TEZĂ DE DOCTORAT

Autor:
Ing. Gabriela CIUPRINA

Conducător științific:
Prof.Dr.Doc.Ing. Constantin MOCANU

1998

Mulțumiri

După mai mulți ani de strădanie în conceperea acestei lucrări pot spune că a realiza o teză de doctorat este un maraton. Pentru a putea trece linia de sosire ai nevoie de încurajări în momentele dificile în care ai vrea să abandonezi, de sfaturi și îndrumări atunci când drumul nu mai e clar.

Este evident că, fără sprijinul mai multor persoane pe care le voi aminti aici, nu aș fi reușit să ajung la capăt. Voi încerca deci să le mulțumesc deși sunt sigură că vor fi prea sărace cuvintele pentru a exprima tot ceea ce aș vrea să transmit.

În primul rând doresc să mulțumesc domnului profesor dr.doc.ing. Constantin Mocanu care, prin numeroasele discuții avute, prin sfaturile și sugestiile competente pe care mi le-a dat m-a ajutat enorm să urc pe acest drum, uneori sinuos, al elaborării tezei până la capăt.

Cele mai calde mulțumiri i le adresez domnului profesor dr.ing. Daniel Ioan, cel care m-a sfătuit să aleg cariera universitară și mi-a călăuzit cel mai îndeaproape pașii pentru a învăța această meserie, având încredere în mine și scoțându-mă de foarte multe ori din impasurile și hățiturile științifice.

Doresc să mulțumesc domnului profesor dr.ing. Florea Hăntilă pentru valoroasele idei pe care mi le-a dat în timpul numeroaselor discuții avute, fără de care finalizarea acestei teze ar mai fi întârziat.

Mulțumiri călduroase adresez și domnului prof.dr.ing. F.M.G.Tomescu pentru sugestiile făcute și pentru atenția și răbdarea cu care a citit și analizat critic lucrările mele științifice.

Întregii echipe a Laboratorului de Metode Numerice în mijlocul căruia lucrez îi mulțumesc pentru că a fost alături de mine ca o adevărată familie. Trebuie să precizez că o parte din rezultatele prezentate în această teză înglobează și munca altor colegi cărora le mulțumesc din suflet: doamnei ș.l.dr.ing. Irina Munteanu, domnului ing. Tiberiu Chelcea, domnișoarei ing. Simona Irimie, domnului ing. Andras Szigeti și doamnei prep.ing. Suzana Stănescu.

Mulțumesc de asemenea tuturor celorlalți profesori care, chiar dacă nu au fost implicați în elaborarea acestei teze, au contribuit la formarea mea profesională. De asemenea

mulțumesc colegilor din Catedra de Electrotehnică precum și tuturor aceluia pe care n-am reușit să-i menționez în această listă dar care m-au ajutat fie și numai cu o vorbă bună adresată atunci când aveam nevoie de ea.

În cele din urmă, dar în nici un caz în ultimul rând, mulțumesc întregii mele familii. Mulțumesc părinților mei ce mi-au creat o copilărie extrem de fericită care mi-a dat tărie și curaj în a înfrunța viața, mulțumesc soțului meu pe care l-am simțit alături mai ales în ultima perioadă în care încheierea tezei devenise o obsesie și mulțumesc fetiței mele pentru toate bucuriile oferite care m-au făcut să descopăr frumusețea vieții.

Familiei mele

Cuprins

1	Introducere	1
1.1	Prezentarea lucrării	1
1.2	Analiza câmpului electromagnetic	3
1.2.1	Modele matematice	3
1.2.2	Modele numerice	4
1.2.3	Ipoteze folosite în aceasta lucrare	7
1.2.4	Tratarea neliniarităților	7
1.3	Optimizarea dispozitivelor electromagnetice	8
1.3.1	Evoluția cunoștințelor privind optimizarea	8
1.3.2	Algoritmi de optimizare	10
2	Stadiul actual	13
2.1	Metode de analiză a câmpului magnetic staționar	13
2.1.1	Abordări unificate pentru modele diferențiale	14
2.1.2	Modele diferențiale	17
2.1.3	Modele integrale	19
2.1.4	Modele mixte	20
2.2	Optimizarea dispozitivelor electromagnetice	22
2.2.1	Probleme de test propuse ("benchmark")	22
2.2.2	Abordări deterministe	23
2.2.3	Abordări stocastice	27
2.2.4	Abordări deterministe comparate cu abordări stocastice	33

2.2.5	Abordări hibride	35
2.2.6	Concluzii	39
3	Analiza numerică a câmpului	43
3.1	Algoritmi pentru rezolvarea problemelor neliniare	43
3.1.1	Formularea corectă a problemei	44
3.1.2	Metoda polarizației	45
3.1.3	Metoda polarizației cu relaxare	52
3.1.4	Metoda Newton	54
3.1.5	Comparații	64
3.1.6	Condiții de frontieră pentru domenii nemărginite	65
3.1.7	Modelul numeric hibrid diferențial-integral (FEM-BEM)	68
3.1.8	Indicatori de eroare a soluției numerice. Rețele adaptive de discretizare.	71
3.2	Teste numerice	80
3.2.1	Probleme de test	80
3.2.2	Rezultate numerice	82
3.2.3	Rata și raza de convergență	83
3.2.4	Concluzii	92
4	Analiza sensibilităților	95
4.1	Sensibilitățile mărimii principale	97
4.1.1	Folosirea unei probleme adjuncte	97
4.1.2	Exploatarea metodei numerice de rezolvare	99
4.2	Sensibilitățile mărimilor derivate	105
4.2.1	Calculul sensibilității inducției magnetice	106
4.2.2	Sensibilitatea fluxului magnetic	106
4.2.3	Sensibilitatea energiei	107
4.3	Utilizarea sensibilităților	107

5	Algoritmi evoluționiști de optimizare	109
5.1	Formularea problemei	110
5.2	Descrierea algoritmului secvențial	112
5.2.1	Reprezentarea genetică și dimensiunea populației	112
5.2.2	Populația inițială	112
5.2.3	Operatori folosiți	113
5.2.4	Mecanismul de selecție	115
5.2.5	Algoritmul folosit	116
5.2.6	Condiția de oprire	117
5.2.7	Parametrii algoritmului secvențial	117
5.3	Algoritmi evoluționiști distribuiți	118
5.3.1	Implementări paralele ale algoritmilor evoluționiști	118
5.3.2	Parametrii specifici unui algoritm paralel ce folosește migrația	119
5.3.3	Algoritmii folosiți	121
5.4	Teste numerice	122
5.4.1	Descrierea problemelor de test	122
5.4.2	Mediul de calcul distribuit	124
5.4.3	Studiul parametrilor algoritmului secvențial	125
5.4.4	Algoritmul distribuit în comparație cu algoritmul secvențial	131
5.4.5	Studiul parametrilor algoritmului distribuit	131
5.5	Algoritmi evoluționiști în optimizarea dispozitivelor	135
5.5.1	Funcții de test	135
5.5.2	Strategie de control optimal	137
5.6	Concluzii	139
6	Rezultate privind optimizarea dispozitivelor	145
6.1	Problema TEAM 22	145
6.1.1	Prezentarea problemei	146
6.1.2	Evaluarea funcției de cost	147

6.1.3	Utilizarea strategiei evoluționiste pentru optimizarea propriu-zisă	152
6.1.4	Calculul sensibilităților	160
6.2	Problema TEAM 25	168
6.2.1	Prezentarea problemei	168
6.2.2	Evaluarea funcției de cost	171
6.2.3	Rezultate numerice	180
7	Contribuții	187
	Anexe	188
A	Tipuri de probleme de optimizare	189
A.1	Enumerarea tipurilor de probleme de optimizare	189
A.2	Probleme de test	192
B	Optimizări vectoriale	193
B.1	Optimalitate Pareto	193
B.2	Stabilirea funcției obiectiv	194
C	Metode de optimizare	197
C.1	Metode de optimizare deterministe pentru probleme fără restricții	197
C.2	Tratarea restricțiilor	199
D	Programe bazate pe evoluție	203
D.1	Structura unui program de evoluție	203
D.2	Baza teoretică a algoritmilor genetici	205
D.3	Convergența algoritmilor genetici	206
D.3.1	Importanța mecanismului de selecție	206
D.3.2	Influența funcției obiectiv	208
D.3.3	Condiția de oprire a algoritmului	208
D.3.4	Algoritmi genetici contractivi	209

D.3.5	GAs cu dimensiunea populației variabilă	210
D.4	Tratarea restricțiilor	211
D.5	Reprezentarea genetică a soluției problemei	211
D.6	Reglajul fin	212
Listă de figuri		213
Listă de tabele		219
Bibliografie		221

Capitolul 1

Introducere

1.1 Prezentarea lucrării

”Studiul câmpului electromagnetic în medii neliniare” este o temă extrem de generoasă. Ea ar putea cuprinde toate regimurile de funcționare ale câmpului electromagnetic, de la static la general variabil, toate tipurile de probleme (rezolvare, optimizare sau probleme inverse, de sinteză), pentru fiecare din ele existând metode (și dificultăți) specifice de abordare. A trata toate problemele într-o lucrare de acest tip este cu siguranță peste puterile unui singur om. De aceea, din diversitatea aceasta am ales problema optimizării dispozitivelor electromagnetice. Optimizarea dispozitivelor electromagnetice este o problemă de mare actualitate, lucru care se observă din creșterea numărului de contribuții în domeniu în ultimii ani. Cea mai prestigioasă conferință din domeniul calculului câmpului electromagnetic (COMPUMAG ¹) are o secțiune dedicată special problemelor de optimizare.

Înlănțuirea rezolvare-optimizare-sinteză este foarte strânsă. O problemă de optimizare are nevoie de o metodă de rezolvare precisă și rapidă, iar o problemă de sinteză are nevoie de o metodă de optimizare potrivită. De aceea, pentru a încadra, clasifica și sistematiza conceptele legate de rezolvare și optimizare am simțit nevoia să scriu o introducere în care să prezint și să explic principalele concepte ale domeniului rezolvării problemelor de câmp și optimizării dispozitivelor electromagnetice.

Introducerea are astfel două părți principale. Prima parte se referă la analiza câmpului electromagnetic și conține definiția modelelor de câmp (fenomenologic, matematic, numeric) precum și evidențierea diversității lor. Partea a doua a introducerii se referă la optimizarea dispozitivelor electromagnetice prezentând evoluția conceptelor privind optimizarea precum și o clasificare a algoritmilor de optimizare.

¹”Conference on the Computation of Electromagnetic Fields”

Starea actuală a cercetărilor în domeniu este prezentată pe larg în capitolul al doilea. Acest capitol are de asemenea două părți importante. Prima dintre ele se referă la metodele de analiză a câmpului electromagnetic în regim magnetic staționar, iar al doilea paragraf reflectă stadiul actual al utilizării metodelor de optimizare în proiectarea dispozitivelor electromagnetice. Pe lângă clasificările de rigoare se fac și considerații critice asupra rezultatelor prezentate în literatură, în vederea identificării căilor și tendințelor de dezvoltare în viitor a acestui domeniu important al cercetării.

Capitolul al treilea este dedicat prezentării metodelor de calcul numeric pentru câmpul electromagnetic cu modelul diferențial și modelul mixt diferențial-integral. Sunt elaborați și prezentați algoritmi dedicați analizei câmpului electromagnetic. Algoritmii au fost implementați și depanați în mediul UNIX/C. Validarea lor a fost efectuată prin compararea rezultatelor numerice obținute cu cele provenite din pachetul de programe MEGA. În urma studiilor efectuate au fost propuși noi algoritmi care pe de o parte permit generarea rețelelor de discretizare adaptate optimal la soluție (prin rafinare succesivă), iar pe de altă parte comută automat între metode diferite de iterații neliniare (metoda polarizației cu relaxare optimală care este garantat convergentă și metoda Newton care este rapidă). Rafinarea rețelei de discretizare se poate aplica atât pe parcursul iterațiilor neliniare cât și pe parcursul procedurii de optimizare. Sunt stabilite criteriile pentru rafinare succesivă (un indicator local de eroare) și pentru comutarea polialgorithmului.

În capitolul al patrulea se studiază metodele de calcul al sensibilităților funcției obiectiv, mărimi necesare aplicării metodelor deterministe de optimizare de ordin superior. Sunt studiate posibilitățile de calcul al sensibilităților mărimilor principale și mărimilor derivate care pot să apară în expresia funcției obiectiv.

Capitolul al cincilea descrie algoritmul evoluționist folosit în optimizarea dispozitivelor electromagnetice prezentate în capitolul al șaselea. Algoritmul elaborat este destinat rulării în paralel pe o arhitectură de calcul distribuită. Sunt descriși și analizați algoritmul secvențial și cel distribuit precum și influența parametrilor specifici asupra convergenței procesului de optimizare.

Capitolul al șaselea conține optimizarea propriu-zisă a două dispozitive electromagnetice. Prima aplicație reprezintă optimizarea unui dispozitiv de stocare a energiei magnetice, iar a doua reprezintă optimizarea unei matrițe folosită pentru orientarea pulberilor în câmp magnetic. Ambele probleme sunt probleme de benchmark propuse de comunitatea internațională în cadrul "TEAM² Workshop". Ultimul capitol este rezervat prezentării contribuțiilor aduse în această teză. Lucrarea are patru anexe, primele trei conținând clasificări ale tipurilor și metodelor de rezolvare ale problemelor de optimizare, ultima anexă explicând principalele concepte legate de programele bazate pe evoluție.

²TEAM (Testing Electromagnetic Analysis Models) reprezintă un grup internațional de lucru constituit în scopul comparării diferitelor programe folosite la analiza câmpului electromagnetic.

1.2 Analiza câmpului electromagnetic

Una din categoriile de probleme fundamentale referitoare la câmpul electromagnetic în medii fixe sau mobile o reprezintă *problemele de analiză* (*calcul* sau *determinare*) a câmpului electromagnetic, în care se presupun cunoscute domeniul spațial de existență a câmpului, caracteristicile materialelor din acest domeniu, distribuția spațială și temporală a surselor de câmp și alte condiții suplimentare cerute pentru obținerea unei soluții unice precum condițiile inițiale și de frontieră și se cere determinarea mărimilor de stare macroscopică locală și instantanee a câmpului (\mathbf{E} , \mathbf{D} , \mathbf{B} , \mathbf{H}) în domeniul spațio-temporal supus analizei.

Formularea corectă a unei probleme de analiză a câmpului electromagnetic macroscopic presupune în primul rând definirea fenomenologiei de bază a problemei (relațiile de cauzalitate esențiale și ipotezele simplificatoare adoptate), deci stabilirea *modelului fenomenologic*, care se referă la regimul câmpului electromagnetic, identificarea surselor de câmp și a modelelor de material.

Urmează apoi reflectarea în plan formal a acestei fenomenologii esențiale într-un *model matematic* de câmp electromagnetic care trebuie prezentat sub forma unei probleme matematice bine formulate. Rezolvarea sa, în vederea obținerii soluției problemei de analiză a câmpului electromagnetic macroscopic, se poate efectua pe cale analitică sau numerică. Metodele analitice (separarea variabilelor, funcții de variabilă complexă, transformări conforme etc.) sunt cele mai riguroase în domeniile lor de aplicabilitate, dar devin inaplicabile în domenii de formă complicată sau în cazul materialelor neliniare. Metodele numerice, bazate pe utilizarea tehnicii moderne de calcul, prezintă o arie de aplicabilitate mai mare, fiind supuse la restricții mai puține. Deși aproximative, dacă satisfac anumite criterii de control a erorii de calcul, metodele numerice conduc la rezultate acceptabile din punct de vedere ingineresc, fiind în acest caz mult mai atractive decât cele analitice.

Perechea formată dintr-un model matematic de câmp electromagnetic și o metodă numerică de determinare a acestui câmp exprimată printr-un algoritm programabil definește un *model numeric* de câmp electromagnetic.

1.2.1 Modele matematice

Modelul matematic constă în esență dintr-o ecuație operatorială în care necunoscuta (pe care o vom numi în cele ce urmează "mărime principală") este o mărime de stare a câmpului sau o altă mărime scalară sau vectorială asociată. La aceasta se adaugă, în acord cu tipul ecuației operatoriale satisfăcută de mărimea principală, condiții suplimentare necesare asigurării unicității soluției ecuației. Din punct de vedere matematic, mărimea principală este de fapt o funcție de mai multe variabile (în care intervin cel

puțin coordonatele spațiale). Pentru ca modelul matematic să fie corect, pentru el trebuie demonstrate teoreme de existență, unicitate și stabilitate a soluției. Acestea din urmă trebuie să garanteze mărginirea abaterilor soluției în cazul unor date perturbate. Pentru problemele în care apar medii neliniare astfel de teoreme nu sunt ușor de formulat și demonstrat.

Este posibil ca mărimea principală să nu fie aceeași în tot domeniul de studiu. Domeniul spațial poate fi împărțit în mai multe subdomenii, fiecare din ele fiind caracterizat de altă mărime principală. La trecerea dintr-un subdomeniu în altul, din condițiile de trecere pentru mărimile de câmp se deduc condiții de trecere pentru mărimile principale.

În funcție de tipul ecuației operatoriale satisfăcute de mărimea principală se disting următoarele tipuri de modele matematice (formulări):

- *Formulări diferențiale* - În acest caz problema se reduce la o ecuație operatorială (sau diferite ecuații operatoriale pe porțiuni) de tip diferențial. Pentru rezolvare sunt necesare impunerea condițiilor de frontieră pentru soluție, și ca atare aceste formulări se utilizează în practica numerică numai în cazul domeniilor mărginite.
- *Formulări integrale* - În acest caz problema se reduce la o ecuație operatorială de tip integral. În această situație nu este necesară impunerea condițiilor pe frontieră și ca atare aceste formulări se pot aplica și domeniilor nemărginite. Modelul matematic integral conține de obicei *integrale pe domeniu* dacă mediul este neomogen și *integrale pe frontieră* dacă mediul este omogen.
- *Formulări mixte* - În acest caz pentru o parte a domeniului se folosește un operator diferențial și pentru altă parte a domeniului se folosește un operator integral. Domeniul spațial în care se folosește operatorul diferențial trebuie să fie mărginit. Cea mai importantă problemă este cea a condițiilor de cuplaj ce trebuie impuse la trecerea din domeniul unui operator în domeniul altui operator.

1.2.2 Modele numerice

Rezolvarea numerică a modelului matematic presupune aproximarea soluției prin reprezentarea sa în funcție de un număr finit de necunoscute scalare. Există mai multe posibilități de a realiza acest lucru.

• Cazul modelului matematic diferențial

Metoda numerică de rezolvare poate fi *metoda diferențelor finite*. Aceasta constă în aproximarea ecuațiilor cu derivate parțiale prin ecuații cu diferențe finite care conțin valorile funcției necunoscute în nodurile unei rețele de discretizare a domeniului de câmp. O altă abordare a metodei diferențelor finite constă în discretizarea formelor integrale ale legilor câmpului electromagnetic și nu a celor locale diferențiale [65, 105]. Precizia

de aproximare a soluției prin valorile sale discrete depinde de forma și pasul rețelei, de tipul relației de discretizare folosită pentru operatorii diferențiali sau integrali, de modul de aproximare a condițiilor la limită și de procedeele numerice de calcul utilizate.

O altă metodă numerică de rezolvare este *metoda elementelor finite*. Și în acest caz domeniul spațial se discretizează, dar rețeaua de discretizare poate avea topologie neregulată. Tratatul modelului matematic se face *variational*. Metoda variațională poate fi *de minimizare* caz în care se construiește mai întâi o problemă echivalentă cu ajutorul unei funcționale energetice ce depinde de mărimea principală, al cărei minim este atins în cazul în care necunoscuta principală este soluția modelului matematic inițial, sau *de proiecție* caz în care se impune anularea proiecției erorii reziduale într-un spațiu care de obicei are dimensiune numărabilă. Prima metodă se mai numește *metoda de minimizare Rayleigh-Ritz*, iar cea de a doua *metoda momentelor* sau *metoda reziduurilor ponderate* sau *metoda de proiecție Petrov-Galerkin*.

Mărimea principală se aproximează apoi cu ajutorul unui număr finit de mărimi necunoscute pe care le vom numi "necunoscute principale". Aceste necunoscute au de obicei semnificație fizică pentru problema dată. Ele pot reprezenta de exemplu valorile mărimii principale în diferite puncte ale domeniului, integrale de linie, suprafață, volum ale mărimii principale de-a lungul unor linii, suprafețe, volume bine definite în domeniul de studiu, derivate după anumite direcții ale mărimii principale în anumite puncte din domeniu, etc.

Fie de exemplu ecuația operatorială: $Kf = g$ unde K este un operator liniar, f este funcția necunoscută și g reprezintă sursa. f se aproximează cu $f = \sum_i \alpha_i f_i$ astfel încât f_i reprezintă un set de funcții liniar independente date iar α_i reprezintă necunoscutele problemei. Metoda momentelor proiectează eroarea reziduală $\varepsilon = \sum_i (Kf_i)\alpha_i - g$ într-un spațiu finit dimensional: $\langle w_j, \varepsilon \rangle = \sum_i \langle w_j, (Kf_i) \rangle \alpha_i - \langle w_j, g \rangle$ unde w_j reprezintă un set de ponderi liniar independente iar \langle, \rangle denotă un produs scalar. Eroarea este minimizată dacă relația anterioară se egalează cu zero. Astfel rezultă un sistem de ecuații liniare satisfăcut de α_i :

$$\sum_i \langle w_j, (Kf_i) \rangle \alpha_i = \langle w_j, g \rangle \quad (1.1)$$

În funcție de alegerea funcțiilor pondere, metoda momentelor ia diferite nume. Astfel: dacă $w_i = \delta$ metoda se numește a colocației, dacă $w_i = f_i$ se obține metoda Bubnov-Galerkin³, dacă $w_i = (Kf_i)$ metoda se numește a celor mai mici pătrate.

Metoda variațională Rayleigh-Ritz - bazată pe minimizarea funcționalei energiei - aplicată ecuației $Kf = g$ conduce la (1.1) dacă operatorul este autoadjunct și pozitiv definit. Pentru operatori care nu sunt autoadjuncți, o formulare variațională s-ar putea să nu

³De acum înainte, când vom specifica "metoda Galerkin" ne vom referi la această metodă.

existe, dar metoda Galerkin poate fi aplicabilă cu condiția ca problema adjuncată să existe [91].

• Cazul modelului matematic integral

În cazul formulărilor diferențiale, una din dificultățile principale și una din sursele principale de erori o constituie modelarea unui câmp care se extinde până la infinit, cu un câmp mărginit spațial. De obicei se alege, relativ arbitrar, o frontieră fictivă care să aproximeze linii de câmp sau suprafețe echipotențiale și care va constitui frontiera exterioară a problemei.

Formulările integrale sunt convenabile deoarece condițiile de frontieră cerute de formulările diferențiale nu trebuie specificate. Regiunile de aer pot fi excluse iar condiția de comportare asimptotică (de exemplu câmpul magnetic se anulează la infinit) este satisfăcută automat. Aceste formulări sunt potrivite pentru problemele de optimizare a formei, precum și în problemele de curenți turbionari în care există mișcare [20, 51].

Ideea folosită în modelele integrale este de a formula o problemă echivalentă în care mediul cu $\mu_r \neq 1$ este înlocuit cu o distribuție echivalentă de surse în vid [171]. Aceste surse trebuie alese astfel încât să determine un câmp egal cu câmpul problemei originale. Contribuția fiecărei surse se determină din ecuațiile Maxwell în vid, iar datorită liniarității, câmpul problemei echivalente se calculează prin superpoziție. Având în vedere acest lucru, tratarea neliniarității trebuie făcută înaintea formulării. Pentru a rezolva numeric ecuația operatorială integrală se poate folosi fie o tehnică de integrare numerică, fie o tehnică de proiecție de tip Galerkin.

În cazul în care mediul este omogen, ecuația integrală cuprinde doar integrale pe frontieră, metoda numerică numindu-se *metoda elementelor de frontieră*. În cazul în care mediul este neomogen și în cazul mediilor neliniare este necesară discretizarea în totalitate a mediului, nu numai a frontierei sale.

• Cazul modelului matematic mixt

Metodele numerice folosite în acest caz reprezintă combinații ale metodelor numerice diferențiale (pentru interiorul domeniului) cu cele integrale (pentru condițiile de frontieră). Din acest motiv se spune că metoda numerică folosită este o *metodă hibridă*.

În cele din urmă un model numeric conduce la un sistem liniar sau neliniar de ecuații algebrice cu un număr finit de necunoscute. Pentru rezolvarea acestuia se aplică algoritmi iterativi și de aceea trebuie studiate următoarele aspecte:

1. În ce condiții algoritmul converge către o soluție, deci care este *raza de convergență*?
2. Cât de repede converge către soluție, deci care este *rata de convergență*?

Dacă din aceste două puncte de vedere algoritmul este convenabil, ceea ce mai rămâne de făcut este evaluarea erorii rezultatelor. Studiul acestor aspecte este mai complicat în

cazul neliniar.

1.2.3 Ipoteze folosite în aceasta lucrare

În cele ce urmează va fi analizat doar cazul regimului magnetic staționar [106]. Mediile se consideră izotrope și neliniare magnetic. Nu se ține seama de fenomenul de histerezis.

1.2.4 Tratarea neliniarităților

Tratarea neliniarităților caracteristicilor de magnetizare se poate face în esență în două moduri.

O posibilitate este de a începe rezolvarea problemei cu tratarea neliniarității, respectiv de a extrage din relația $\mathbf{B} - \mathbf{H}$ neliniară dată o componentă liniară astfel încât de exemplu $\mathbf{B} = \mu\mathbf{H} + \mathbf{I}$. Se adoptă apoi o anumită formulare pentru această problemă liniară. Metoda numerică se aplică deci unei probleme liniare, rezultând un sistem liniar în necunoscutele principale. După obținerea soluției se corectează datele de material (se determină componenta neliniară din caracteristica de magnetizare în punctul corespunzător soluției numerice obținute pentru problema liniară) și se reia rezolvarea problemei liniare. Iterațiile continuă în acest mod. Unei astfel de metode îi vom spune în general *metoda polarizației*. Această metodă este o metodă de punct fix [29, 66]. Metoda numerică, bazată pe o formulare diferențială converge întotdeauna, dar convergența este relativ lentă atunci când materialul nu este prea saturat, în schimb ea converge rapid când saturația este puternică [30]. Acest stil de tratare a neliniarității se aplică aproape întotdeauna formulărilor integrale.

O a doua posibilitate este de a folosi o formulare direct pentru problema neliniară. Acest lucru se poate face de obicei pentru o formulare diferențială. Metoda numerică se aplică unei probleme neliniare, rezultând un sistem neliniar în necunoscutele principale. Pentru rezolvarea lui se poate folosi de exemplu *metoda Newton-Raphson* sau o metodă iterativă pe care o vom numi *metoda substituției* sau *metoda iterației X* , unde X este mărimea care se iterează. Această din urmă metodă constă în presupunerea anumitor valori inițiale pentru mărimea neliniară X , rezolvarea sistemului, corecția mărimii neliniare, rezolvarea sistemului, ș.a.m.d. Pentru modelele numerice diferențiale metoda Newton-Raphson este mai rapidă decât metoda polarizației, însă în cazul unor materiale foarte saturate metoda Newton-Raphson poate să nu convergă. Comportamentul complementar al celor două metode sugerează că, în cazul materialelor foarte saturate să se aplice mai întâi metoda polarizației, iar soluția obținută din aceasta să constituie punct de start pentru o procedură Newton-Raphson care converge mult mai repede (pătratic) decât metoda polarizației care are o convergență liniară. În [30] se propune o astfel de

metodă (formularea folosită este una diferențială). În cazul în care materialele nu sunt saturate, nu merită să se aplice o astfel de combinație. În astfel de cazuri, rezultatele metodei polarizației nu sunt precise după primele iterații, fiind preferabilă doar aplicarea metodei Newton-Raphson.

Formularea integrală pentru un mediu neliniar aduce probleme numerice suplimentare față de cazul formulării diferențiale pentru un mediu neliniar, motiv pentru care, în cazul unei formulări mixte se preferă de obicei formularea integrală pentru regiunile liniare și formularea diferențială pentru regiunile neliniare.

1.3 Optimizarea dispozitivelor electromagnetice

1.3.1 Evoluția cunoștințelor privind optimizarea

Problemele de optimizare apar în toate domeniile științei și ingineriei acolo unde este necesar să se minimizeze (sau maximizeze) anumite funcții obiectiv, sau să se aranjeze echipamente complexe, să se stabilească o succesiune de operații, etc. În ingineria electrică problemele de optimizare cele mai interesante presupun determinarea câmpului electromagnetic.

Optimizarea dispozitivelor electromagnetice este de multe ori o optimizare a dimensiunilor lor geometrice (sau forme ale pieselor componente) și a surselor de câmp (poziția și valoarea lor) astfel încât mai multe obiective să fie satisfăcute. Foarte pe scurt specificația de proiectare ("design") constă într-o mulțime de costuri și restricții. Un dispozitiv valid va satisface toate restricțiile, minimizând costurile. Costurile sunt de obicei mai simple decât restricțiile. Iată câteva exemple de funcții de cost: volumul total de cupru dintr-un dispozitiv, forța asupra unui conductor, costul materialelor, puterea disipată, inductivitatea bobinelor, mărimea sau neuniformitatea câmpului electromagnetic produs de dispozitiv. Aceste funcții depind direct de valorile variabilelor de proiectare. O altă clasă de funcții de cost implică o mulțime de soluții pentru un interval al parametrilor (de exemplu liniaritatea forței în funcție de deplasare). Restricțiile sunt de obicei mai complicate decât costurile. Condiția de "quench"⁴ (impusă câmpului magnetic dintr-un supraconductor) din problema TEAM 22 [98], care semnifică faptul că bobina este în stare supraconductoare, este un exemplu de restricție complicată.

Problemele care urmăresc satisfacerea simultană a mai multor obiective se numesc probleme de optimizări vectoriale⁵.

⁴"quench" - este un termen din metalurgie, el însemnând "revenire" - operația prin care un metal aflat la temperaturi înalte este răcit brusc, de exemplu cu apă.

⁵Pentru clarificarea terminologiei pe care o voi folosi: unui obiectiv de îndeplinit i se asociază o *funcție*

Din perspectivă istorică se constată că metodele matematice de optimizare și unele tehnici numerice asociate lor au început să se dezvolte prin anul 1950. Teoria programării neliniare cu restricții liniare bazată pe criteriul de optimalitate a lui Kuhn și Tucker a fost prezentată în 1951. Metodele de tratare a restricțiilor neliniare au început să se dezvolte la începutul anilor 1960. Numeroși algoritmi de optimizare folosind atât metode deterministe cât și stocastice au fost dezvoltați la sfârșitul anilor 1960. Numeroși algoritmi de optimizare utilizând atât metode deterministe cât și elemente stocastice au fost dezvoltați în cărțile lor de Wilde (1964), Rosenbrock (1966), Himmelblau (1972), Brent (1973) și Schwerel (1977) [137].

Una din primele aplicații în optimizarea dispozitivelor electromagnetice este cea a lui Halbach care în 1967 introducea o metodă pentru optimizarea aranjamentului unor bobine și forme polilor magnetilor, folosind metoda elementelor finite pentru calculul câmpului. Armstrong combina în 1982 algoritmi de optimizare cu metoda integralelor de volum pentru optimizarea profilului polului unui magnet. Girdinio optimizează în 1983 forma unui electrod. Toate aceste încercări erau totuși specifice aplicației, fără să aibă un caracter general [137].

Numai recent, programele generale de calculul câmpului au început să fie "legate" de algoritmi de optimizare. Motivele acestei întârzieri includ: restricții legate de puterea de calcul, probleme legate de discontinuitatea și nederivabilitatea funcțiilor de cost care rezultă din discretizarea problemei, lipsa de acuratețe în calculul numeric al câmpului. Începând cu 1989, un număr tot mai mare de contribuții ale cercetătorilor din domeniul calculului câmpului electromagnetic se referă la problemele de optimizare. Iată o extrem de mică selecție de lucrări și câteva nume dintre cele mai des întâlnite (ordinea este cea alfabetică): Barba [16, 17], Dughiero [15, 45], Dyck și Lowther [47], Gottvald [16, 54, 130], S.R.H. Hoole [60, 61, 62, 126, 127, 150, 166, 167], Mohammed [43, 88, 89, 108, 109, 158, 159, 160], Russenschuck [108, 136, 137], Saldanha [80, 138, 139, 163], Simkin și Trowbridge [147, 148], Park [94, 119, 120, 121, 122], Preis [9, 54, 100, 130, 142].

Interesul pentru acest tip de probleme reiese și din faptul că printre ultimele probleme apărute în lista TEAM Workshop, și anume problemele numărul 22 [98] și numărul 25 [151], sunt probleme de optimizare a dispozitivelor electromagnetice.

Recent, cercetătorii dezvoltă algoritmi sofisticăți, potriviți pentru procesare paralelă, care pot găsi optime globale și sunt potriviți pentru un număr mare de variabile de proiectare.

Din nefericire, varietatea de metode aplicate arată că nu există o metodă generală care să rezolve problema optimizării unui dispozitiv electromagnetic așa cum algoritmul sim-

de cost; funcția care este optimizată propriu-zis și care înglobează mai multe obiective de îndeplinit este numită *funcție obiectiv*.

plex există pentru rezolvarea problemei programării liniare. Există multe aplicații diferite și fiecare categorie își cere procedura proprie de optimizare. Dificultatea optimizării dispozitivelor electromagnetice este legată și de faptul că evaluarea funcției obiectiv necesită rezolvarea numerică a unei probleme de câmp care implică resurse de calcul (timp CPU, memorie) importante. Din acest motiv algoritmi de optimizare trebuie să fie adaptați problemei și să reducă pe cât posibil la minim evaluările inutile ale funcției obiectiv. În caz contrar execuția ar putea necesita un timp de calcul atât de mare încât programul dezvoltat pe baza lor ar fi inutil.

1.3.2 Algoritmi de optimizare

Algoritmii de optimizare pot fi clasificați din mai multe puncte de vedere. Din punct de vedere al ordinului derivatelor necesare metodele sunt *de ordin zero* - cele care au nevoie doar de evaluarea funcției obiectiv sau *de ordin superior* - cele care necesită și evaluarea derivatelor funcției obiectiv. În metodele de ordin superior traiectoria de optimizare dintr-un spațiu Euclidian n -dimensional este construită pas cu pas ca o secvență de căutări unidimensionale de-a lungul unei direcții (direcția unui gradient) care este recalculată la fiecare iterație. În aceste metode trebuie calculați vectorii gradient (sau chiar aproximații cu derivate de ordin superior) ale funcției obiectiv. Fiecare din metodele de ordin superior diferă prin modalitatea de calcul a direcțiilor.

Din alt punct de vedere metodele de optimizare pot fi clasificate în *metode deterministe* și *metode nedeterministe (stocastice, statistice)*. Algoritmii determiniști sunt reproductibili deoarece conduc la aceeași soluție pentru rulări diferite dacă pornesc de la aceleași condiții inițiale și au aceiași parametri. Acest lucru nu se întâmplă în cazul algoritmilor statistici. Dezavantajul algoritmilor nedeterminiști constă în faptul că ei nu garantează obținerea soluției chiar dacă timpul de rulare este infinit. Dezavantajul algoritmilor determiniști este acela că ei permit obținerea doar de extreme locale dependente de inițializare iar în cazul folosirii unor algoritmi de ordin superior ei necesită și proprietăți suplimentare ale funcției obiectiv (de exemplu derivabilitatea).

Algoritmii statistici sunt algoritmi de ordin zero ⁶. Există algoritmi determiniști atât de ordin zero (de exemplu metoda Brent, metoda simplexului coborâtor, metoda simplex, metoda Powell) cât și de ordin superior (metode de gradienti conjugați, metode quasi Newton, metoda pașilor descendenți).

Din punct de vedere al funcției obiectiv folosită pentru minimizare, metodele se împart în *metode primare* și *metode de transformare* (conform clasificării din [139]). Metodele

⁶Din punct de vedere al definiției de mai sus; unii autori se referă la algoritmi statistici de ordin superior [100], dar această definiție se referă la alt aspect.

primare sunt cele care folosesc funcția obiectiv așa cum este ea dată în specificația problemei. Metodele de transformare sunt cele care transformă problema originală cu restricții într-o succesiune de probleme fără restricții. Metodele de tratare a restricțiilor descrise în anexa C.2 sunt astfel de metode de transformare.

De multe ori într-o problemă de optimizare a dispozitivelor electromagnetice se caută extreme globale. Găsirea unui extrem global este, în general, o problemă foarte dificilă. Se folosesc pe scară largă două metode euristice. Prima constă în găsirea mai multor extreme locale pornind de la valori de start diferite ale variabilelor independente. Aceste valori de start trebuie împrăștiate în tot domeniul de căutare. În final se alege extremul global dintre valorile găsite. O a doua abordare perturbă un extrem local luând un pas de amplitudine finită și apoi văzând dacă algoritmul întoarce un punct mai bun sau revine la acel extrem. Alte strategii sunt metoda Tabu (standard sau aleatoare), căutarea după "pattern". Metoda Tabu standard a fost propusă de Glover în 1986 [46] pentru a rezolva probleme combinatoriale ca de exemplu problema comis-voiajorului. Căutarea este făcută într-un număr de pași și rezultatele sunt memorate într-un tabel (listă). Bazându-se pe acest tabel sunt stabilite niște restricții pe baza cărora se desfășoară căutarea viitoare. Parametrul fundamental este dimensiunea listei. Dacă lista Tabu este prea mică atunci vor apare repetiții excesive ale buclelor de căutare; dacă este prea mare atunci căutarea în vecinătatea unui optim global este inhibată. Fără a viola o restricție Tabu, o repetiție a căutărilor rafinează soluția și procesul conduce la găsirea optimelor globale, eliminându-le pe cele locale. Hu a modificat procedura lui Glover [46] și a adaptat-o rezolvării problemelor de optimizare cu restricții, procedura numindu-se "Tabu aleatoare". Se introduc doi noi parametri și anume numărul de regiuni din vecinătatea unui punct și numărul maxim de căutări care să se facă într-o vecinătate. Dacă x_0 este un punct, atunci se împarte vecinătatea lui într-un număr de subvecinătăți. O nouă mulțime de soluții posibile este generată la întâmplare în fiecare subvecinătate. Dacă vreun punct corespunde la o valoare mai bună decât $f(x_0)$ atunci acesta devine aproximația următoare. Procedura de bază este foarte similară cu metoda de căutare Hooke-Jeeves și cu metoda gridului. "Pattern search" caută să afle din panta locală a funcției obiectiv și din informațiile din încercările anterioare cea mai scurtă cale spre optim. Această metodă este potrivită pentru faza a doua a algoritmului de optimizare [44].

Ca o alternativă la aceste abordări, au început să se aplice metode de căutare globală (aleatoare, Simulated Annealing, strategii evoluționiste) care pot evita agățarea în extreme locale.

De asemenea, în dorința de a combina avantajele celor două tipuri de algoritmi și de a reduce dezavantajele lor, au început să apară algoritmi combinați care încep prin aplicarea unui algoritm statistic urmat de un algoritm determinist, sau amestecă un algoritm statistic cu unul determinist.

Pe lângă metodele amintite mai sus, există și abordări care vin dinspre rezolvarea problemelor inverse, și care intră în categoria *tehnichilor de inteligență artificială* cum ar fi folosirea rețelelor neurale [78] sau a logicii vagi (fuzzy) [149].

Anexele A-D sunt menite a completa și clarifica informațiile despre metodele de optimizare, astfel: anexa A descrie tipurile de probleme de optimizare, anexa B cuprinde informații despre optimizările vectoriale, criteriul de optimalitate a lui Pareto și stabilirea funcției obiectiv în cazul optimizărilor vectoriale, anexa C conține o clasificare a metodelor de optimizare deterministe pentru probleme fără restricții precum și modul de tratare a problemelor cu restricții, anexa D fiind rezervată programelor de evoluție și în particular algoritmilor genetici (bază teoretică, algoritmi, convergență).

Capitolul 2

Stadiul actual al metodelor folosite în optimizarea dispozitivelor electromagnetice

O metodă de proiectare (în sens de optimizare) a unui dispozitiv electromagnetic necesită analiza mai multor configurații posibile. Informațiile rezultate dintr-o astfel de analiză influențează mersul metodei de optimizare. De aceea primul aspect care trebuie rezolvat în vederea optimizării unui dispozitiv electromagnetic îl constituie *alegerea metodei de analiză* a dispozitivului. Având în vedere acest motiv am împărțit acest capitol în două părți. Prima din ele se referă la metodele de analiză a câmpului electromagnetic¹. Al doilea paragraf al acestui capitol reflectă stadiul actual al utilizării metodelor de optimizare în proiectarea dispozitivelor electromagnetice.

2.1 Metode de analiză a câmpului magnetic staționar

Analiza problemei de câmp presupune stabilirea unui model matematic și rezolvarea lui cu o metodă analitică sau numerică. Modelele matematice reflectă fenomenologia problemei de rezolvat. Ele constau în ecuații diferențiale sau integrale la care se adaugă alte condiții necesare asigurării existenței, unicității și stabilității soluției. Ecuațiile au drept necunoscute mărimile de stare locală \mathbf{B} , \mathbf{H} sau potențiale. Se va presupune că sunt satisfăcute ecuațiile fundamentale ale regimului magnetic staționar:

$$\operatorname{div} \mathbf{B} = 0 \quad , \quad \operatorname{rot} \mathbf{H} = \mathbf{J} \quad , \quad \mathbf{B} = f(\mathbf{H}) \quad (2.1)$$

¹Gândindu-mă la multitudinea de tipuri de probleme care există, cu siguranță nu aș fi putut scrie despre stadiul actual al acestui domeniu fără a face câteva limitări legate de clasa de probleme. Cunoscând problemele de optimizare pe care le aveam de rezolvat, toate fiind de regim magnetic staționar, am încercat să pun în evidență și să sintetizez metodele de analiză folosite în rezolvarea acestei categorii de probleme.

În cazul particular al mediilor liniare cu magnetizație permanentă relația $\mathbf{B} - \mathbf{H}$ are un caracter afin $\mathbf{B} = \mu\mathbf{H} + \mathbf{I}$. Este evident că la aceste ecuații trebuie adăugate condițiile de frontieră specifice problemei.

2.1.1 Abordări unificate pentru modele diferențiale

O abordare unificată a problemei este prezentată în [125]. Figura 2.1 ilustrează relațiile dintre diferitele mărimi care se pot defini (pentru simplificare am presupus că nu există magnetizație permanentă). Legătura între mărimi se face cu ajutorul operatorilor diferențiali. \mathbf{A} este *potentialului magnetic vector* continuu și ales astfel încât $\mathbf{B} = \text{rot } \mathbf{A}$. Ecuația satisfăcută de el este $\text{rot} \left(\frac{\text{rot } \mathbf{A}}{\mu} \right) = \mathbf{J}$ numită "ecuație primară" în schema din figura 2.1. Se observă că pentru a lega potențialul \mathbf{A} de sursa de câmp \mathbf{J} se poate merge prin \mathbf{B} și \mathbf{H} sau direct prin ecuația Poisson.

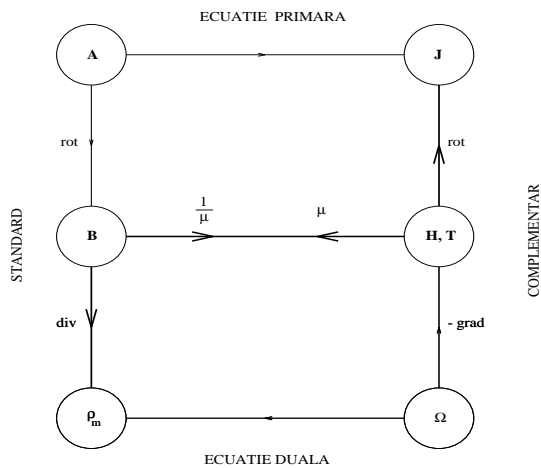


Figura 2.1: Relații de bază pentru regimul magnetic staționar

Pentru a face mai evidentă simetria schemei autorii notează $\text{div } \mathbf{B} = \rho_m$ unde ρ_m ar fi densitatea sarcinii magnetice libere, și este bineînțeles zero. Rezultă $\mathbf{B} = \mu\mathbf{H} = \mu(\mathbf{T} - \text{grad } \Omega)$ și ecuația duală $\text{div} [\mu(\mathbf{T} - \text{grad } \Omega)] = \rho_m$. Există o anumită simetrie în această figură. De exemplu \mathbf{J} și ρ_m sunt sursele de câmp, în timp ce \mathbf{A}, Ω^2 sunt potențialele. Totuși, operatorii care unesc \mathbf{A} de \mathbf{B} și \mathbf{H} de \mathbf{J} sunt rot și adjunctul său care este tot rot . \mathbf{B} este legat de ρ_m prin operatorul div . Adjunctul acestui operator este $-\text{grad}$ care apare într-adevăr în partea dreaptă a schemei. \mathbf{T} a fost plasat lângă \mathbf{H} . El are aceleași unități de măsură ca și \mathbf{H} și se comportă ca el. De aceea autorii îl numesc *vectorul intensității magnetice reduse*. Această reprezentare cuprinde și relațiile

O altă posibilitate este de a folosi o reprezentare pentru intensitatea câmpului magnetic \mathbf{H} . Deoarece $\text{div } \mathbf{J} = 0$, se poate găsi un vector \mathbf{T} astfel încât $\mathbf{J} = \text{rot } \mathbf{T}$ apoi, deoarece $\text{rot } \mathbf{H} = \text{rot } \mathbf{T}$ rezultă $\mathbf{H} = \mathbf{T} - \text{grad } \Omega$. \mathbf{T} este numit de unii autori [27, 105] *potențial electric vector* pentru că provine dintr-un curent, iar Ω *potențial magnetic scalar*. Penman și Fraser [125] propun un alt nume pentru \mathbf{T} și anume *intensitate magnetică redusă*. De asemenea Ω mai este numit *potențial scalar redus* spre a-l deosebi de cazul $\mathbf{J} = 0$ în care \mathbf{H} poate fi exprimat doar cu ajutorul unui potențial scalar care este numit *potențial scalar total*.

²Doar în acest subcapitol voi nota acest potențial cu Ω . În rest, cu Ω voi nota domeniul geometric al problemei.

de bază în mărimile de câmp și anume (împrumutând termenii din mecanică): relația de compatibilitate $\operatorname{div} \mathbf{B} = 0$, relația constitutivă $\mathbf{B} = \mu \mathbf{H}$, relația de echilibru (relația de stare) $\operatorname{rot} \mathbf{H} = \mathbf{J}$.

Există mai multe posibilități de a rezolva problema astfel formulată. Primul pas îl constituie alegerea ecuației cu derivate parțiale (primare sau duale). Dacă se presupune că aceasta se rezolvă variațional (fie prin minimizare, fie prin proiecție), mai trebuie făcută o alegere. Putem satisface ecuația de compatibilitate și apoi să se rezolve echilibrul sau putem satisface echilibrul și rezolva compatibilitatea. Tabelul 2.1 sintetizează cele patru posibilități:

Ec.dif. aleasă	satisfăcută direct	satisfăcută indirect	formulare (ecuație)
primară	compatibilitatea ($\operatorname{div} \mathbf{B} = 0$)	echilibrul ($\operatorname{rot} \mathbf{H} = \mathbf{J}$)	standard pentru ecuația primară (ec. în \mathbf{A})
	echilibrul ($\operatorname{rot} \mathbf{H} = \mathbf{J}$)	compatibilitatea ($\operatorname{div} \mathbf{B} = 0$)	complementară pentru ecuația primară (ec. în \mathbf{H})
duală	compatibilitatea ($\operatorname{div} \mathbf{B} = 0$)	echilibrul ($\operatorname{rot} \mathbf{H} = \mathbf{J}$)	standard pentru ecuația duală (ec. în \mathbf{B})
	echilibrul ($\operatorname{rot} \mathbf{H} = \mathbf{J}$)	compatibilitatea ($\operatorname{div} \mathbf{B} = 0$)	complementară pentru ecuația duală (ec. în Ω)

Tabelul 2.1: Posibilități de a rezolva un model matematic diferențial

Rezultatele obținute au proprietăți interesante. De exemplu dacă se folosesc ambele formulări (standard și complementară) pentru o anumită ecuație se obțin rezultate care corespund unor extreme pentru energie. Pentru formulări complete și demonstrații pentru încadrări trebuie consultate referințele articolului [125]. De exemplu, pentru detalii matematice și teste numerice legate de formularea primară standard și complementară se poate consulta [124, 57].

A. Bossavit în [21] prezintă structura matematică descrisă mai sus cu ajutorul complexului lui de Rham. Acest complex este alcătuit din patru spații (două copii ale lui \mathbf{L}^2 și două copii ale lui L^2) și trei operatori diferențiali (figura 2.2) care acționează între aceste spații. Domeniul de definiție al unui operator este doar un subspațiu al lui \mathbf{L}^2 sau L^2 .

$$\mathbf{L}^2 \xrightarrow{\operatorname{grad}} \mathbf{L}^2 \xrightarrow{\operatorname{rot}} \mathbf{L}^2 \xrightarrow{\operatorname{div}} \mathbf{L}^2$$

Figura 2.2: Complexul lui de Rham

Să considerăm că domeniul spațial este simplu conex, iar regiunile S_h și S_b în care este împărțită frontiera sunt de asemenea simplu conexe. Se presupun de asemenea că

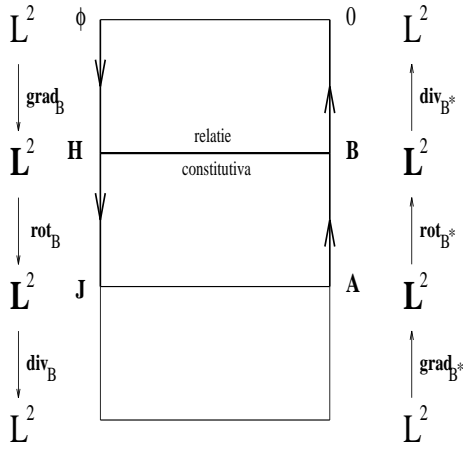


Figura 2.3: Diagrama Tonti primară

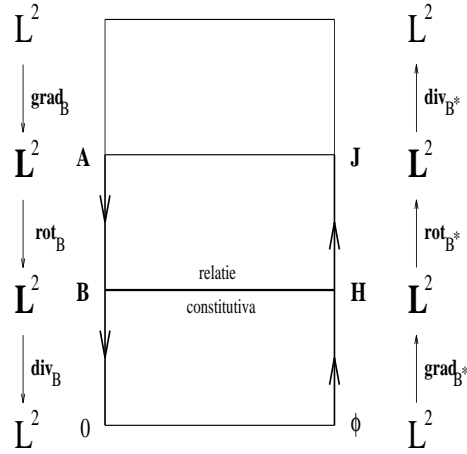


Figura 2.4: Diagrama Tonti duală

pe frontieră sunt satisfăcute condiții nule: $\mathbf{n} \times \mathbf{H} = \mathbf{0}$ pe B , $\mathbf{n} \cdot \mathbf{B} = 0$ pe B^* în care s-a notat $B = S_h$ și $B^* = S_b$, deci B^* este complementara lui B față de frontieră. Operatorii din acest complex depind de frontieră, așa cum se vede din tabelul 2.2.

Operator	Domeniul operatorului
grad_B	$\phi \in L^2 : \text{grad } \phi \in \mathbf{L}^2, \phi = 0$ pe B
rot_B	$\mathbf{H} \in \mathbf{L}^2 : \text{rot } \mathbf{H} \in \mathbf{L}^2, \mathbf{n} \times \mathbf{H} = \mathbf{0}$ pe B
div_B	$\mathbf{B} \in \mathbf{L}^2 : \text{div } \mathbf{B} \in L^2, \mathbf{n} \cdot \mathbf{B} = 0$ pe B^*

Tabelul 2.2: Domeniile operatorilor din complexul Rham

sunt simplu conexe, secvența nu va mai fi una exactă. Se poate construi acum un complex dual (tabelul 2.3), folosind adjuncții operatorilor de mai sus.

Aceste două structuri duale pot fi așezate vertical, prima la stânga coborând și a doua la dreapta urcând. Pe această structură mărimile de câmp și potențialele pot fi așezate în două moduri (figurile 2.3 și 2.4). Astfel de diagrame se numesc diagrame Tonti.

Dacă domeniul și frontiera sunt simplu conexe, structura de Rham este exactă (imaginea unui domeniu printr-un operator este exact nucleul operatorului de la dreapta sa). Dacă domeniul sau frontiera sa nu

Operator	Domeniul operatorului	Coincide cu:
$(\text{grad}_B)^*$	$\mathbf{B} \in \mathbf{L}^2 : \text{div } \mathbf{B} \in L^2, \mathbf{n} \cdot \mathbf{B} = 0$ pe B^*	$-\text{div}_{B^*}$
$(\text{rot}_B)^*$	$\mathbf{H} \in \mathbf{L}^2 : \text{rot } \mathbf{H} \in \mathbf{L}^2, \mathbf{n} \times \mathbf{H} = \mathbf{0}$ pe B^*	rot_{B^*}
$(\text{div}_B)^*$	$\phi \in L^2 : \text{grad } \phi \in \mathbf{L}^2, \phi = 0$ pe B^*	$-\text{grad}_{B^*}$

Tabelul 2.3: Domeniile operatorilor din complexul dual complexului de Rham

Existența și unicitatea perechii (\mathbf{B}, \mathbf{H}) rezultă din proprietățile de exactitate ale structurii în cazul mediilor liniare. Rezultatul se extinde și pentru medii neliniare, pentru demonstrație fiind necesare elemente de analiză convexă.

2.1.2 Modele diferențiale

În cele ce urmează voi comenta pe scurt diferitele formulări întâlnite. Explicații mai detaliate se găsesc în [33] și, evident, în sursele bibliografice citate.

Formulara în \mathbf{B} și \mathbf{H} - este prezentată deosebit de riguros în [63, 64, 65, 66]. Se demonstrează teoreme de unicitate, existență și stabilitate pentru domenii care pot fi multiplu conexe. Este necesar ca relația de material să fie lipschitziană și uniform monotonă.

Formulara în potențial scalar total Ψ poate fi aplicată în situația în care în domeniul de studiu nu există curent de conducție. În această situație pentru reprezentarea lui \mathbf{H} se poate folosi un singur potențial scalar, notat Ψ și numit potențial scalar total: $\mathbf{H} = -\text{grad } \Psi$. În schema din figura 2.1 el este reprezentat de potențialul Ω . Permeabilitatea magnetică μ depinde de câmp în Ω_{fier} și este egală cu permeabilitatea vidului în Ω_{aer} , iar \mathbf{B}_r este inducția remanentă din interiorul magneților permanenți. Condițiile de frontieră scrise pentru potențiale devin: Ψ dat pe S_h și $\mathbf{n} \cdot (\mu \text{grad } \Psi)$ dat pe S_b . Dezavantajul acestei formulări îl constituie condiția de a nu exista curenți de conducție, clasa de probleme la care se poate aplica fiind astfel extrem de redusă.

Formulara în potențial scalar redus Φ - introdusă de Zienkiewicz în [172] a apărut din dorința de a evita folosirea unui potențial vector care ar introduce de trei ori mai multe necunoscute. Această formulare are nevoie și de o soluție analitică obținută prin integrare din curenți. Spre deosebire de formularea în potențial scalar total Ψ , Ω poate fi multiplu conex, iar în el pot exista distribuții de curenți. Se calculează mai întâi un câmp \mathbf{H}_s astfel încât $\text{rot } \mathbf{H}_s = \mathbf{J}$ de exemplu cu formula Biot-Savart-Laplace. În schema din figura 2.1 acesta este reprezentat de potențialul \mathbf{T} . Rezultă deci $\oint_{\Gamma} (\mathbf{H} - \mathbf{H}_s) \cdot d\mathbf{r} = 0$ pentru orice curbă închisă Γ din domeniul Ω , deci câmpul $\mathbf{H} - \mathbf{H}_s$ poate fi scris ca fiind gradientul unui potențial scalar redus Φ : $\mathbf{H} - \mathbf{H}_s = -\text{grad } \Phi$. Ecuația diferențială pe care trebuie să o satisfacă potențialul scalar redus este: $\text{div} (\mu \text{grad } \Phi) = \text{div} (\mu \mathbf{H}_s) + \text{div } \mathbf{B}_r$ iar condițiile de frontieră scrise pentru potențiale devin: Φ dat pe S_h , $\mathbf{n} \cdot (\mathbf{H}_s - \mu \text{grad } \Phi)$ dat pe S_b .

Formulara în potențial scalar total - potențial scalar redus $\Psi - \Phi$ a fost prezentată de Simkin și Trowbridge în [145]. Ei au arătat că formularea în potențial scalar redus Φ cu \mathbf{H}_s calculat analitic duce la erori mari în interiorul materialelor feromagnetice. Acest lucru se explică în felul următor: deoarece $\mathbf{H} = \mathbf{H}_s - \text{grad } \Phi$ iar în interiorul materialelor feromagnetice $\mathbf{H} \ll \mathbf{H}_s$, înseamnă că $\text{grad } \Phi$ și \mathbf{H}_s au același ordin de mărime, ceea ce duce la erori numerice datorită fenomenului de anulare prin scădere. De aceea ei au propus

în [146] o metodă care folosește două potențiale scalare. În această metodă în regiunile fără curent se folosește potențial scalar total Ψ iar în regiunile în care $\mathbf{J} \neq 0$ se folosește potențial scalar redus Φ . Regiunea de potențial scalar total trebuie să fie simplu conexă. La frontiera dintre regiunea Ψ și regiunea Φ trebuie impuse condiții de interfață, deduse din condițiile de trecere pentru câmp, astfel încât să se asigure unicitatea soluției. Metoda are două dezavantaje majore [129]: pe de o parte dacă regiunea ocupată de materialul feromagnetic este multiplu conexă și înconjoară curenți, atunci potențialul scalar total trebuie făcut discontinuu pe anumite suprafețe numite "tăieturi". Pe de altă parte nu sunt permise surse de curent în regiunile de permeabilitate mare. Aici trebuie folosit potențial scalar total pentru a evita erorile de anulare prin scadere.

Formularea în potențial vector \mathbf{A} . Dezavantajele formulării $\Phi - \Psi$ pot fi depășite prin utilizarea potențialului vector fie în tot domeniul, fie doar în regiunile ocupate de materialele feromagnetice. Acum însă apar alte probleme, de natură numerică. Câmpul magnetic nedefinind în mod unic potențialul vector, apar probleme numerice dacă acest potențial nu este etalonat. În cazul problemelor plan-paralele sau axisimetrice potențialul vector este foarte popular deoarece el are numai o componentă, cea ortogonală pe planul de analiză. În cazul tridimensional toate cele trei componente există și trebuie calculate. Ecuația diferențială pe care trebuie să o satisfacă potențialul vector este $\text{rot}(\nu \text{rot } \mathbf{A}) = \mathbf{J}$. Condițiile de frontieră pentru o formulare corectă sunt ceva mai complicate și sunt prezentate în [65]. O altă variantă [129] constă în modificarea ecuației diferențiale în: $\text{rot}(\nu \text{rot } \mathbf{A}) - \text{div}(\nu \text{div } \mathbf{A}) = \mathbf{J}$ și înlocuirea condițiilor de frontieră în mod corespunzător. Acest mod de a privi problema corespunde unei funcționale cu penalizare atașate ecuației diferențiale inițiale și care conține sub formă variațională condiția de etalonare. Această din urmă prezentare este detaliată în [65].

Formularea în potențial vector redus \mathbf{A}_r - este prezentată în [129] și a apărut ca reacție la unul din dezavantajele formulării în \mathbf{A} și anume acela de a discretiza cu grijă bobinele, densitatea de curent apărând într-o integrală de volum de tipul $\int_{\Omega} \mathbf{J} \cdot \mathbf{w}_i d\Omega$. Potențialul vector este împartit în două părți: $\mathbf{A} = \mathbf{A}_s + \mathbf{A}_r$ unde \mathbf{A}_s este potențialul vector datorat distribuției de curenți în aer. Între acest potențial \mathbf{A}_s și câmpul magnetic \mathbf{H}_s dedus printr-o formulă Biot-Savart în formularea $\Phi - \Psi$ există relația: $\text{rot } \mathbf{A}_s = \mu_0 \mathbf{H}_s$. Condițiile de frontieră pentru \mathbf{A}_r sunt aceleași ca pentru potențialul vector total (formularea \mathbf{A}) iar calculul câmpului \mathbf{A}_s nu este necesar, deoarece el intervine numai în expresia $\text{rot } \mathbf{A}_s$ care este $\mu_0 \mathbf{H}_s$.

Formularea în potențial vector - potențial scalar redus $\mathbf{A} - \Phi$: Pentru a reduce numărul de grade de libertate care apar în formularea în potențial vector, este posibil să se utilizeze potențialul vector doar în regiunile feromagnetice, în rest utilizându-se doar potențial

scalar redus [129].³ Experiențele numerice au arătat că dacă frontiera aer-fier coincide cu frontiera $\Phi - \mathbf{A}$ apar comportări ciudate ale câmpului în zona acestei interfețe, comportare care nu este eliminată prin rafinarea rețelei ci prin extinderea regiunii Ω_A astfel încât frontiera $\Phi - \mathbf{A}$ se află în totalitate în aer. Comportarea ciudată în cazul în care frontiera aer-fier coincide cu frontiera $\Phi - \mathbf{A}$ se explică prin faptul că cuplarea între regiunile \mathbf{A} și Φ este asigurată de componenta normală a inducției în fier și de componenta tangențială a intensității câmpului magnetic în aer. Ambele cantități sunt foarte mici deoarece inducția este practic tangențială în fier iar intensitatea câmpului magnetic este practic normală în aer datorită schimbărilor rapide în permeabilitate.

Formulara $\mathbf{A}_r, \Phi - \Psi$ - este descrisă în [165] și calculează componenta intensității câmpului magnetic care are rotorul diferit de zero prin intermediul unui potențial vector redus \mathbf{A}_r . Intensitatea câmpului magnetic constituie apoi o excitație pentru o soluție în potențial scalar în tot domeniul. Avantajul acestei formulări este acela că neliniaritatea intervine doar în final, când se folosesc doar potențiale scalare, precum și în faptul că nu sunt necesare tăieturi pentru a asigura domeniul de potențial scalar total simplu conex (ca la formulara $\Psi - \Phi$).

2.1.3 Modele integrale

Dezvoltarea metodelor integrale a fost făcută în principal de cercetătorii din domeniul telecomunicațiilor, pentru că aici apar probleme de împrăștiere, radiații, care implică domenii nemărginite și pentru care formulările integrale constituie o alegere naturală. Formulările integrale sunt mai puțin populare decât formulările diferențiale și iată câteva motive: metodele numerice pentru rezolvarea ecuațiilor diferențiale sunt mai ușor de folosit și de implementat; în cazul ecuațiilor integrale apar singularități; funcțiile Green depind de problemă și în general nu sunt cunoscute sau ușor de obținut; expresiile integrale sunt mai dificil de evaluat. Ecuațiile integrale conțin integrale de volum care doar în cazul mediilor omogene pot fi simplificate la integrale de suprafață (pe frontiere), prin aplicarea teoremei divergenței.

Pentru a obține o formulare integrală se construiește o problemă echivalentă în care mediul feromagnetic este înlocuit cu o distribuție echivalentă de surse în vid. Aceste surse echivalente pot fi sarcini sau dipoli magnetici (surse de simplu strat, dublu strat), sau distribuții de curent echivalente. Se fac apoi raționamente bazate pe superpoziție sau pe folosirea funcțiilor Green. O altă posibilitate de a ajunge la formulări integrale constă în utilizarea identității Green, sau a metodei Treftz [134]. Rezultatele obținute în cazul

³Iată explicația notației prescurtate pentru o formulare: atunci când folosim diferite potențiale pe diferite regiuni, vom folosi liniuța ca separator între potențiale; când este vorba de aceeași regiune în care se folosesc mai multe potențiale, vom folosi virgula pentru a separa aceste potențiale.

folosirii surselor de simplu strat sau identității Green sunt similare. Metoda Treftz poate fi considerată un caz special al utilizării identității Green.

În [33] am detaliat cum se ajunge la formularea integrală în potențial scalar, formularea integrală în \mathbf{H} , formularea integrală în \mathbf{M} prin \mathbf{H} și apoi prin \mathbf{B} . Formularea integrală în potențial scalar este un model dipolar, bazat pe o problemă echivalentă în care apar surse de tipul unor dipoli magnetici (surse de dublu strat) [171]. Este necesar ca domeniul de studiu să fie simplu conex și în el să nu existe curenți de conducție, ipoteze destul de limitatoare. Ipotezele acestea nu sunt necesare în formularea integrală în \mathbf{H} în care prezența materialului feromagnetic este reprezentată printr-o distribuție echivalentă de curent electric [171]. În formularea integrală în \mathbf{M} prin \mathbf{H} problema echivalentă se construiește astfel [59]: materialul feromagnetic se înlocuiește cu aer în care există două tipuri de distribuții de sarcină, una de volum cu densitatea $\operatorname{div} \mathbf{M}$ și una de suprafață cu densitatea $\operatorname{div}_s \mathbf{M} = \mathbf{n}_{12} \cdot (\mathbf{M}_2 - \mathbf{M}_1) = -\mathbf{n} \cdot \mathbf{M}$, unde \mathbf{n} este normala exterioară la Ω_{fier} - domeniul ocupat de materialul feromagnetic, iar \mathbf{M} este vectorul magnetizație din materialul feromagnetic. În formularea integrală în \mathbf{M} prin \mathbf{B} , pornind de la ecuațiile satisfăcute de potențialul vector \mathbf{A} , în problema echivalentă materialul feromagnetic se înlocuiește cu aer în care există două tipuri de distribuții de curent: una de volum cu densitatea $\operatorname{rot} \mathbf{M}$ și una de suprafață cu densitatea $\operatorname{rot}_s \mathbf{M} = \mathbf{n}_{12} \times (\mathbf{M}_2 - \mathbf{M}_1) = \mathbf{M} \times \mathbf{n}$, unde \mathbf{n} este normala exterioară la Ω_{fier} - domeniul ocupat inițial de materialul feromagnetic. Bineînțeles se păstrează și restul curenților de conducție care existau în problema originală și care reprezintă cauza magnetizației corpului feromagnetic. Deci cele două probleme sunt echivalente din punct de vedere al potențialului vector \mathbf{A} (deci și a câmpului \mathbf{B}) peste tot, iar din punct de vedere al lui \mathbf{H} doar în domeniul $\Omega - \Omega_{fier}$ [171].

Aceste formulări nu epuizează modalitățile prin care se poate ajunge la ecuații integrale în cazul regimului magnetic staționar (medii neliniare). Alte referințe bibliografice pentru ecuații integrale sunt: [91, 92] - ecuații integrale în potențial scalar pentru medii liniare și omogene (se folosesc funcții Green); [90] - ecuație integrală în sarcini magnetice pentru medii neliniare; [10] - ecuație integrală în potențial scalar pentru medii liniare și omogene pe porțiuni (se folosesc funcții Green); [135] - ecuație integrală în potențiale scalare pentru medii liniare; [106] - expresii integrale pentru potențialul vector și ecuații integrale în \mathbf{B} pentru ecuația Laplace. [143] - ecuație integrală în potențial vector (2D), tehnica abordată constând în descompunerea operatorului diferențial neliniar într-o sumă formată dintr-un operator liniar și unul neliniar. Termenul neliniar este trecut în membrul drept și privit ca o sursă. Pornind de la această ecuație diferențială neliniară se obține ecuația integrală cu ajutorul formulei Green; [6] - ecuație integrală în potențial vector (3D) pentru medii liniare; accentul este pus pe calculul integralelor singulare în sensul valorii principale Cauchy.

2.1.4 Modele mixte

Avantajele și dezavantajele rezolvărilor numerice pentru formulările diferențiale și pentru cele integrale au condus în mod natural la apariția unei formulări combinate. Pe anumite porțiuni se folosește un model diferențial, pe altele se folosește un model integral. Vom spune că un astfel de model este un model mixt. Pentru rezolvarea numerică a lui se combină metodele numerice corespunzătoare diferitelor formulări. Vom înțelege astfel prin termenul de *metoda hibridă* o combinație între o metodă numerică corespunzătoare unei formulări diferențiale și o metodă numerică corespunzătoare unei formulări integrale. Cu o astfel de metoda se pot rezolva de exemplu problemele cu frontiera deschisă ("open boundary"). Dificultățile adaptării unei formulări integrale în cazul mediilor neliniare a făcut ca metoda hibridă să folosească formularea integrală pe porțiunea liniară și formularea diferențială pe porțiunea neliniară a domeniului.

Idea folosirii acestei metode a provenit din dorința de a exploata avantajele și a evita dezavantajele fiecărei metode [140]. Astfel, metoda elementelor finite generează ecuații multe pe când metoda elementelor de frontieră generează ecuații puține. Metoda elementelor finite se aplică domeniilor închise, restricție care nu apare la elemente de frontieră. Metoda elementelor finite este mult mai ușor adaptabilă la materiale neliniare decât metoda elementelor de frontieră. Matricile generate de elemente finite sunt rare spre deosebire de cele generate de elemente de frontieră care sunt pline. Matricea elementelor finite este diagonal dominantă și simetrică. La elemente de frontieră diagonal dominanța nu este asigurată, iar matricea este nesimetrică (devine simetrică dacă se folosește o formulare variațională). De asemenea utilizarea elementelor de frontieră are în anumite cazuri avantaje față de alte metode propuse pentru problemele cu frontieră deschisă ("ballooning" sau "elemente infinite" [28, 95, 97, 103]).

Iată alte câteva puncte de reper: [123, 141] - probleme 2D, formulare în potențial vector; [155] - probleme 2D formulate în potențial vector și probleme 3D formulate în potențiale scalare; [11] - probleme 3D formulate în potențiale scalare; [117] - probleme 3D - formulări în potențial vector și în \mathbf{H} ; [14] - formulare mixtă în \mathbf{H} , deosebit de riguros prezentată.

2.2 Optimizarea dispozitivelor electromagnetice

Tehnicile matematice de optimizare descrise pe scurt în capitolul 1 și anexe sunt un instrument esențial în optimizarea dispozitivelor electromagnetice. Principala problemă constă în a găsi proceduri robuste care să conducă la o soluție optimală în sens Pareto și care să satisfacă restricțiile.

În acest paragraf vom descrie (fără a epuiza subiectul) principalele dispozitive electromagnetice și metodele folosite pentru optimizarea lor întâlnite în literatura de specialitate. Gama de aplicații este vastă: de la determinarea geometriei magneților care trebuie să producă un câmp uniform în sistemele MRI ⁴[101, 154], determinarea formei polilor în acceleratoarele de particule și dispozitive de deflecție [115], optimizarea magneților din diverse dispozitive, tot felul de motoare [19, 69, 136], transformatoare [93] și alte dispozitive care să satisfacă anumite cerințe cu costuri minime, în optimizarea formei izolatoarelor sau a electrozilor acestora [162], proiectarea dispozitivelor de levitație magnetică [41, 45, 118] până la proiectarea dispozitivelor de încălzire prin inducție [40, 126].

2.2.1 Probleme de test propuse ("benchmark")

În domeniul proiectării automate a dispozitivelor electromagnetice, cercetătorii au devenit conștienți de necesitatea clarificării mai multor aspecte: definirea unui limbaj neambiguu, o clasificare revăzută a metodelor de optimizare adaptată aplicațiilor din ingineria electrică și o listă de probleme de test, standard, capabile să evalueze cu severitate eficiența metodelor de optimizare.

În [16] se propune prima problemă de test, luată din domeniul rezonanței magnetice nucleare care referă la așa numitul solenoid al lui Loney. Sistemul, cu doar două grade de libertate care identifică geometria, constă dintr-o bobină principală (solenoid) de dimensiuni date, situată în aer și două bobine corectoare (tot solenoidale) plasate în exterior și coaxial cu bobina principală. Problema constă în a găsi lungimea bobinelor corectoare (L) și distanța dintre ele (S) astfel încât în centrul bobinei principale câmpul magnetic să fie uniform pe o anumită distanță. Funcția obiectiv propusă este $D = (B_{max} - B_{min})/B_0$ unde B_0 este valoarea dorită în zona de interes. Problema este de tip minimax deoarece se urmărește minimizarea deviației maxime, fiind prost condiționată și cu multe minime locale. În [15] se reia problema solenoidului lui Loney de data aceasta cu patru variabile de proiectare. Cele două variabile noi introduse sunt dimensiunile bobinelor corectoare (raza și grosimea).

Alte două probleme de test pentru proceduri de optimizare au fost propuse la TEAM

⁴Magnetic Resonance Imaging

Workshop. În [98] se propune optimizarea unei configurații de stocare a energiei magnetice într-un sistem supraconductor (problema TEAM 22) în raport cu trei obiective: energia să aibă o anumită valoare, câmpul magnetic de dispersie să fie cât mai mic iar în interiorul înfășurărilor trebuie să fie respectată condiția de câmp magnetic critic care asigură starea supraconductoare a conductorului. O altă problemă este prezentată în [151] și se referă la optimizarea formei unei matrițe cu electromagnet folosită la crearea materialelor anizotrope care sunt răcite în câmp magnetic (problema TEAM 25). Obiectivul de îndeplinit este realizarea unui câmp magnetic de o anumită valoare și orientare în cavitatea în care se obțin materialele anizotrope. De rezolvarea acestor două probleme se ocupă capitolul al șaselea.

2.2.2 Abordări deterministe

Pentru primele optimizări ale dispozitivelor electromagnetice au fost folosite metode deterministe. În continuare se prezintă principalele contribuții întâlnite în literatura de specialitate, fiind scoase în evidență ideile principale.

- **Metode de ordin zero**

Dintre metodele deterministe de ordin zero, metoda simplexului coborâtor este folosită de exemplu în [17] pentru optimizarea solenoidului lui Loney.

- **Metode de ordin superior**

Mai des folosite însă sunt metodele de ordin superior (gradienți conjugați, pași descendenți, quasi-Newton) în care sunt folosite și informații despre derivatele (senzitivitățile) funcției obiectiv în raport cu variabilele de proiectare. Din acest motiv analiza senzitivităților este de importanță majoră în utilizarea algoritmilor determinați de ordin superior, fiind foarte important să se deducă cât mai precis posibil formule pentru senzitivități.

- **Senzitivități - abordare discretă**

Există două abordări pentru calculul analitic al senzitivităților [119]. O abordare posibilă este una bazată pe sistemul discretizat de ecuații ce rezultă de exemplu prin aplicarea metodei elementelor finite, abordare folosită în [120, 132]. Metoda elementelor finite duce la un sistem de ecuații care conține în el informații despre dimensiunile geometrice, proprietățile de material și sursele de câmp. Din expresiile termenilor ce apar în acest sistem sunt deduse expresii pentru calculul senzitivităților. În [132] analiza senzitivităților este făcută chiar pe modele tridimensionale și folosește potențialul magnetic vector ca variabilă de stare. Formule pentru senzitivități în această abordare discretă sunt deduse pentru o formulare pentru câmpuri magnetice și termice cuplate în cazul optimizării unui sistem de încălzire prin inducție în [126], formule utilizate apoi într-un algoritm de gradienți conjugați. Cazul unui regim armonic permanent se întâlnește în [47].

• **Senzitivități - abordare continuă (modelul adjunct)**

O altă abordare pentru calculul analitic al sensibilităților este una continuă, bazată pe diferențierea formulării variaționale înainte de a fi discretizată. Astfel de abordare folosesc [87, 89, 93, 119, 121]. Formulele sensibilităților sunt într-o formă în care intră variabilele de stare și variabile adjuncte și ele conțin integrale doar pe suprafețele care se modifică. Pentru evaluarea numerică codul de elemente finite este folosit atât pentru variabilele de stare cât și pentru variabilele adjuncte. O astfel de abordare este folosită în [94] pentru minimizarea pierderilor în fier dintr-un transformator monofazat, folosind metoda gradientilor conjugați. Același tip de analiză a sensibilităților este folosită și în [40], unde se optimizează un dispozitiv de încălzire prin inducție din punct de vedere al eficienței sale. Se deduc formule pentru sensibilități în concordanță cu formularea problemei (în mai multe potențiale, curenți turbionari) și metoda variabilei adjuncte. Aceeași strategie este folosită în [122] unde se folosește metoda pașilor descendenți pentru optimizarea formei polilor.

• **Senzitivități pentru metode integrale**

Formule pentru sensibilități au fost deduse și în cazul folosirii unor metode integrale pentru evaluarea funcției obiectiv. În [162] se folosește abordarea continuă pentru calculul sensibilităților, cuplată cu tehnica funcției Lagrange modificate pentru optimizarea unui izolator, pentru rezolvarea problemei fiind folosită metoda elementelor de frontieră. În [153] se optimizează dimensiunile unui miez feromagnetic dintr-un circuit magnetic astfel încât să se păstreze valorile dorite ale inducției magnetice. Pentru rezolvarea problemei se folosește o metodă integrală în \mathbf{M} . Ecuația în \mathbf{M} este neliniară și rezolvată cu metoda Newton-Raphson. Abordarea problemei cu restricții se face cu ajutorul funcției Lagrange modificate. Pentru minimizarea acestei funcții se folosește o metodă quasi-Newton. Tot o abordare cu o metodă integrală și calculul sensibilităților adaptat acestei metode este făcut în [62] pentru optimizarea unui condensator de reglaj și în [86] pentru optimizarea formei unui electromagnet. În [156] se optimizează profilul unui izolator folosind metodele deterministe de ordin superior: Gauss-Newton, quasi-Newton, metoda pașilor descendenți, metoda gradientilor conjugați. Pentru rezolvarea problemei de regim electrostatic se folosește o metodă integrală care necesită astfel doar discretizarea suprafeței izolatorului. Dintre aceste metode, Gauss-Newton a fost cea mai rapid convergentă. Autorii afirmă că metoda Gauss-Newton poate conduce uneori la matrice prost condiționate, situații în care este mai bine să fie folosite metode quasi-Newton sau gradienti conjugați. În [85] sunt deduse formule pentru sensibilități pentru o problemă tridimensională (un electromagnet) rezolvată prin metoda elementelor de frontieră. Formulele se deduc prin diferențierea ecuațiilor integrale de frontieră în raport cu variabilele de proiectare.

• **Metode analitice pentru evaluarea funcția de cost**

În cazul în care pentru funcția obiectiv pot fi deduse (printr-o metodă analitică sau semi-

analitică) formule în care parametrii de proiectare să intervină explicit, sensibilitățile se calculează mai simplu, prin derivarea acestor formule. Astfel, [112] evidențiază utilitatea unor formule analitice pentru sensibilități chiar dacă ele sunt determinate în ipoteze simplificatoare (material liniar, neglijarea efectelor de capăt) pentru dispozitivul de optimizat (o matriță de turnare cu magneți permanenți). Metoda poate fi aplicată pentru primii pași ai unei metode de optimizare, înaintea aplicării unei metode mai riguroase. În [83] se optimizează forma polului unui magnet cu ajutorul unei proceduri deterministe quasi-Newton. Modelul de analiză este unul liniar plan paralel pentru care autorii dezvoltă expresii analitice de calcul a câmpului folosind funcții de variabilă complexă. Câmpul magnetic poate fi datorat unei distribuții uniforme de curent, și magneților permanenți magnetizați uniform. În [17] metoda folosită pentru calculul câmpului este metoda spectrală a armonicilor zonale, care permite o evaluare cu acuratețe a câmpului magnetic în interiorul unui solenoid. În [15] se reia problema solenoidului lui Loney de data aceasta cu patru variabile de proiectare, abordarea problemei făcându-se determinist pentru problema de optimizare (metoda quasi-Newton) și analitic pentru rezolvare (prin calculul unor integrale eliptice de speța a doua).

• Aproximări pentru funcția de cost

Algoritmii determiniști sunt instabili în anumite condiții, de exemplu în problemele de curenți turbionari, unde soluția variază foarte repede lângă suprafețele conductoare datorită efectului pelicular. Aceste variații tind să facă instabilă metoda de optimizare. Pentru a evita aceste perturbații numerice în [127] se recomandă folosirea unor forme normalizate de tipul celor mai mici pătrate pentru funcția obiectiv, forme care se comportă mult mai bine la optimizare. Astfel, pentru optimizarea formei unei bobine dintr-un dispozitiv de încălzire prin inducție astfel încât să se minimizeze pierderile în înfășurare, se folosește o funcție obiectiv de forma $F = \sum_{k=1}^p (q_k/q_{d,k} - 1)^2$, relația indicând deviația densității de putere q_k obținută din rezolvarea problemei față de densitatea de putere dorită $q_{d,k}$ într-un punct de inspecție k . O aproximare polinomială a funcției obiectiv, aproximare implicând produse de puteri ale tuturor variabilelor se folosește în [41] care optimizează secțiunea unui magnet permanent component al unui tren MAGLEV⁵. A optimiza un dispozitiv cu levitație magnetică este mai dificil decât a optimiza un motor electric. Vehiculul trebuie să fie stabilizat din punct de vedere al levitației și al ghidajului, minimizarea greutateii este esențială și bineînțeles și costul lui. De obicei sursa câmpului magnetic este plasată pe vehicul iar curenții care interacționează cu acest câmp sunt plasați (sau induși) în linii. Problema constă în a determina secțiunea magnetului care minimizează curentul cerut pentru accelerarea vehiculului la o anumită viteză. Metoda folosită pentru optimizare este de tip quasi-Newton (metodă de variabilă metrică). Având în vedere cerințele acestei metode dificultatea constă în obținerea unei funcții netede și diferențiabile care să reprez-

⁵MAGLEV = magnetically levitated vehicle

inte cu acuratețe performanța vehiculului. Algoritmii de variabilă metrică merg cel mai bine dacă funcția de cost și derivatele ei în raport cu variabilele de proiectare (Hessianul, Jacobianul) sunt calculate cu ușurință și de aceea se preferă aproximarea polinomială a funcției obiectiv. O abordare asemănătoare se folosește în [118] care optimizează tor un dispozitiv de levitație manetică folosind metoda programării pătratice secvențiale.

• Algoritmi rapizi

Pentru evaluarea rapidității unui algoritm de optimizare se ia drept criteriu de apreciere numărul de apeluri (evaluări) ale funcției obiectiv, știut fiind faptul că în optimizarea dispozitivelor electromagnetice această evaluare este costisitoare din punct de vedere al timpului de calcul deoarece necesită rezolvarea unei probleme de câmp prin metode numerice, problemă care poate avea mii până la milioane de necunoscute scalare. Algoritmii determinați au avantajul convergenței rapide către un optim. Iată de exemplu în [40] este prezentată optimizarea unui dispozitiv de încălzire prin inducție cu un astfel de algoritm. Au fost necesare numai 8 evaluări de funcții obiectiv pentru un dispozitiv simplu de încălzire și 44 evaluări de funcții obiectiv pentru un cuptor de inducție cu o geometrie mai complicată.

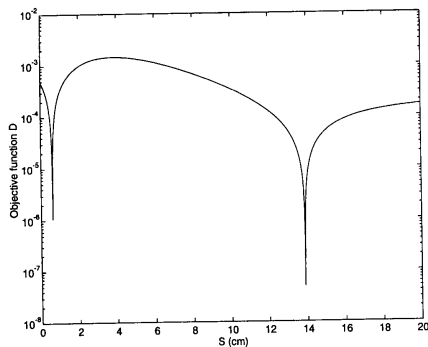


Figura 2.5: Funcția de cost a solenoidului Loney în funcție de S ($L = 5.5$ cm)

dau rezultate similare. Majoritatea punctelor finale (mai mult de 90 % dintre ele) se așează la S constant, dar dispersia lui L este foarte mare indicând faptul că sensibilitatea problemei este mult mai mare în raport cu S decât cu L . Dintre cele două metode, metoda simplexului coborător s-a dovedit a fi mai robustă (indicele de eroare al subrutinei simplex (NAG) este zero, pe când indicii de eroare ai subrutinei quasi-Newton (biblioteca NAG) sunt diferiți de zero). Procesul de optimizare este reluat apoi pentru $S = 14$ cm (valoarea spre care au converș majoritatea rulărilor cu diferite puncte de start) și găsește valoarea optimă a lui $L = 5.5$ cm. Pentru o validare a rezultatului s-a păstrat $L = 5.5$ cm și a fost considerat ca parametru de optimizare doar S . Variația funcției de cost cu S este

• Multistart

Deși sunt rapizi, algoritmii determinați nu garantează obținerea unui optim global. De aceea mulți autori recomandă tehnica multistart [17, 115]. Iată un exemplu de strategie de optimizare folosit în [17]. Autorii abordează problema solenoidului lui Loney cu două metode deterministe: metoda simplexului coborător și metoda quasi-Newton. Metodele deterministe depind foarte mult de punctul de start, și de aceea problemei i se aplică o tehnică multistart, folosind o sută de puncte de start împrăștiate aleator în domeniu. Distribuția punctelor finale indică faptul că cele două metode

prezentată în figura 2.5. Iată deci că pentru o configurație extrem de simplă se observă prezența unui minim local. Minimul a fost regăsit pentru $S = 14$ cm.

2.2.3 Abordări stocastice

Funcțiile obiectiv care apar în problemele de optimizare din electromagnetism sunt adesea puternic nelinare, cu multe extreme și sunt de multe ori nediferențiabile. Metodele deterministe au dezavantaje mari atunci când sunt aplicate la funcții cu multe extreme conducând la algoritmi de optimizare care sunt foarte interactivi și costisitori. De aceea au început să fie preferați algoritmi euristici care sunt insensibili la perturbații ale funcției obiectiv datorită evaluării numerice, sunt algoritmi de ordin zero, care nu necesită calculul derivatelor. Mai mult, metodele de optimizare stocastice au o probabilitate mai mică de a converge către un extrem local decât metodele deterministe.

• Simulated annealing (SA)

Călirea simulată ("Simulated Annealing") a fost folosită cu succes în rezolvarea problemelor complexe de optimizare a dispozitivelor electromagnetice [79, 88, 116]. Acest algoritm se bazează pe analogia dintre procesul optimizării globale și răcirea unui topituri. Topitura își atinge configurația cristalină care corespunde, în cazul răcirii încete, unei energii minime [8, 148]. Astfel, algoritmul atinge minimul global dacă optimizarea (căutarea) se face suficient de încet. Ideea de bază a metodei constă în faptul că, pentru a evita agățarea în minime locale, mișcări ascendente (corespunzătoare unei valori a funcției obiectiv mai proaste) ar putea fi totuși acceptate. Esențial în algoritmul SA este criteriul Metropolis care garantează o anumită rată de acceptabilitate a configurațiilor proaste. Acest criteriu poate fi enunțat astfel: dacă noua funcție obiectiv F_2 al unui nou punct \mathbf{x}_2 este mai mică decât valoarea corespunzătoare configurației anterioare F_1 , noua configurație este acceptată; dacă $F_2 > F_1$ atunci \mathbf{x}_2 este acceptat cu o probabilitate $P = \exp[-(F_2 - F_1)/T]$ unde T este un parametru de control care corespunde temperaturii dintr-un proces fizic de călire. Procesul este condus cu astfel cu ajutorul parametrului T , numit "temperatură". Cu cât temperatura este mai mare, cu atât este mai mare probabilitatea acceptării configurațiilor mai proaste. Parametrul T trebuie redus treptat astfel încât rata de acceptabilitate a încercărilor mai rele să se diminueze. În cele din urmă, când temperatura este suficient de scăzută vor fi acceptate numai configurații mai bune. Teoretic, în final minimul global va fi obținut dacă au fost încercate un număr suficient de mare de configurații la T constant și dacă temperatura a fost redusă la zero cu o viteză foarte mică. Algoritmul este descris în [44, 131, 152]. Algoritmul SA este considerat ca o metodă de căutare locală generalizată [43]. O metodă de căutare locală admite numai puncte mai bune pe toată durata algoritmului.

Acest algoritm a fost mai întâi propus în domeniul optimizării combinatoriale, adică în

cazul în care funcția de cost este definită într-un domeniu discret. În acest caz mișcările aleatoare corespund unor permutări în lista de mișcări posibile, așa cum se întâmplă în problema comis-voiajorului în care mișcările aleatoare corespund unor permutări în lista orașelor de vizitat. De aceea multe probleme importante, definite ca funcții de variabile continue cer în cazul aplicării SA o strategie eficientă în alegerea pasului aleator. Această strategie trebuie să fie independentă de problemă [161]. Dezavantajul algoritmului este acela că cere foarte multe evaluări ale funcției obiectiv, deci este un foarte mare consumator de timp. În [101] se afirmă că pentru obținerea unor rezultate de încredere sunt necesare un număr de evaluări ale funcției obiectiv de ordinul sutelor de mii. Foarte multe articole se preocupă astfel de îmbunătățirea convergenței algoritmului, în vederea reducerii timpului de calcul.

În vederea reducerii timpului de calcul în [8] se propune aproximarea funcției obiectiv cu funcții de interpolare pătratice. O idee asemănătoare se găsește în [102]. Aici se optimizează o configurație cu solenoizi astfel încât câmpul într-o anumită regiune să fie uniform, folosindu-se pentru calculul funcției obiectiv formule analitice. Funcția de cost într-un punct nou este calculată incremental, calculând doar contribuția elementelor care s-au schimbat. Algoritmul SA a avut nevoie de aproximativ 8000 evaluări de funcții obiectiv pentru a rezolva această problemă.

Bazat pe algoritmul SA, în [133] se propune o metodă numită de autori MFA ("Mean Field Annealing"). În această metodă unele mărimi sunt înlocuite cu valorile medii, mediile făcându-se pe lanțuri Markov. Pe măsură ce temperatura descrește algoritmul recalculează aceste medii în funcție de valorile la temperatura precedentă. Cu o astfel de procedură, echilibrul unui lanț Markov se atinge mult mai repede la fiecare temperatură decât la SA. Dezavantajul este că MFA nu garantează convergența nici măcar pentru probleme pentru care SA converge.

În [152] se încearcă investigarea sistematică a factorilor care influențează convergența SA. Deoarece SA converge destul de încet, în [170] se propune o îmbunătățire a convergenței folosind o schemă de răcire adaptivă.

Cea mai interesantă trăsătură a SA [8], mai ales în unele din cele mai recente implementări [44] constă în probabilitatea foarte mare cu care poate fi identificat minimul global. În [43, 44] se procedează astfel: fie $\mathbf{x} \in \mathbb{R}^n$ vectorul variabilelor și $\mathbf{s} \in \mathbb{R}^n$ un vector de deplasări. Atunci $x_j^{i+1} = x_j^i + r s_j$, unde x_j^i este parametrul j la încercarea i . Componentele vectorului deplasare \mathbf{s} depind de rata de acceptabilitate $R_a = N_a/N_T$ unde N_T este numărul de puncte de încercare (lungimea lanțului Markov) iar N_a numărul de puncte acceptate la T constant, r este $+1$ sau -1 în mod aleator. După unele recomandări valoarea inițială a temperaturii este obținută astfel încât la această temperatură $R_a = 1$. În [43] se determină temperatura inițială astfel. Mai întâi se ia un vector de deplasare \mathbf{s} suficient de mare și o valoare de test T_0 , se rulează algoritmul de $n \times N_s$ ori,

unde $N_s = 10 \div 20$, apoi se calculează R_a . Dacă $R_a < 1$ se crește T_0 . Dacă $R_a = 1$ se reduce s până când R_a devine puțin mai mic decât 1. Se ia T_0 ca valoare inițială pentru pentru temperatură. Se repornesc iterațiile cu acest T_0 și cu valoarea lui s dedusă. După fiecare $n \times N_s$ iterații se verifică R_a și se modifică s astfel încât R_a să fie în jurul valorii de 0.5. Când condiția pentru R_a este satisfăcută se pornește un nou ciclu cu o temperatură redusă, multiplicând temperatura curentă cu un factor între $0.8 \div 0.9$. Pentru fiecare temperatură vectorul deplasare are valoarea curentă (a pasului anterior). Apoi se reduce s pentru a se menține R_a în jurul valorii de 0.5. În acest fel, atunci când parametrul de control T este foarte mic, deplasarea va fi foarte mică.

Deși algoritmul SA este intrinsec secvențial, constând în generarea unei secvențe de puncte în domeniul funcției, au apărut și implementări paralele ale acestuia. Dezvoltarea unui algoritm paralel fără violarea acestei secvențe nu este o sarcină ușoară. Ideea principală este să se realizeze căutări după mai multe direcții, adaptând pașii de căutare și numărul de încercări la comportarea funcției pentru a economisi timpul de calcul (prin reducerea numărului total de evaluări) fără a micșora probabilitatea de a găsi minimul global. În [101] se folosește o abordare speculativă, în care procesoarele sunt văzute ca fiind nodurile unui arbore binar.

• Algoritmi evoluționiști

La jumătatea secolului al nouăsprezecelea știința biologiei a fost revoluționată prin descoperirea lui Charles Darwin a procesului prin care natura selectează și optimizează organismele vii. Creșterea în ultimul deceniu a puterii de calcul a făcut posibilă aplicarea principiilor procesului de optimizare din natură la dispozitive construite pe baza teoretică a ecuațiilor lui Maxwell.

În forme felurite, algoritmi evoluționiști și în particular algoritmi genetici au fost aplicați problemelor din domenii extrem de diferite, de la inginerie până la economie, în domeniul inteligenței artificiale. Algoritmi genetici pot fi foarte ușor aplicați problemelor care implică funcții nederivabile și spațiilor de căutare discrete. De asemenea ei au o natură paralelă inerentă, care îi face foarte atractivi pentru implementarea pe rețele de calculatoare. Există o teoremă a paralelismului implicit [168], acesta implicând faptul că dublarea populației ar putea să reducă timpul de calcul cu mai mult de 50 %. Anexa D explică pe scurt programele de evoluție și în particular algoritmi genetici, detalii suplimentare putând fi găsite în [104].

Rechenberg a transformat schema evoluției biologice într-un algoritm simplu [81]. Această procedură elementară este numită strategia (1+1). Un părinte (un vector soluție) generează un copil printr-o mutație a unei variabile. După evaluarea funcției obiectiv, vectorul de calitate mai bună este părinte în noua generație. Populația constă deci într-un singur părinte. Pasul de mutație se ajustează din când în când astfel încât rata mutațiilor cu succes să fie $1/5$ [100]. Schema (1+1) nu ia în considerare o populație. Bazându-se

pe strategia (1+1) strategii mai generale și mai puternice au fost introduse de Schoefel. Strategia de evoluție $(\mu + 1)$ implică μ părinți. O altă caracteristică vitală poate fi considerată durata de viața a unui individ. După ce dau naștere unui copil, părinții nu mai supraviețuiesc în generația următoare. Aceasta este strategia $(\mu, 1)$. Dacă din μ părinți posibili doar ρ participă cu informații genetice la un copil, aceasta este strategia $(\mu/\rho, 1)$. Dacă numărul de descendenți este crescut de la 1 la λ atunci strategia se numește $(\mu/\rho, \lambda)$. Iată algoritmic această strategie:

1. inițial μ părinți

2. repetă

2.1. alege λ mulțimi (nu neaparat disjuncte) de câte ρ părinți

2.2 din fiecare mulțime cu ρ părinți rezultă 1 copil

(acum sunt deci λ copii)

2.3 aplică mutația celor λ copii

2.4. din cei λ copii alege μ cei mai buni;

(aceștia devin părinți în noua generație)

până când este îndeplinită condiția de stop.

Se observă că $\rho \leq \mu \leq \lambda$. Strategiile în care generația următoare se alege din părinți și copii se mai numesc și *strategii plus*⁶. Acestea sunt deci strategiile $(\mu + \lambda)$ și $(\mu/\rho + \lambda)$ atunci când la crearea unui copil contribuie ρ părinți. Un copil are de la un părinte un factor de ereditate $1/\rho$. Strategiile în care generația următoare este selectată doar din cei mai buni copii se numesc *strategii virgulă*⁷ și acestea sunt (μ, λ) sau $(\mu/\rho, \lambda)$. Strategia virgulă este numită strategie de ordin superior în [100], denumire ce ar putea induce confuzii, de obicei metodele de ordin superior fiind cele care necesită calculul derivatei.

Pentru a transforma procedura într-una eficientă este necesar un pas (folosit pentru mutație) de lungime adaptivă. Dacă acest pas este prea mic atunci părinții și copiii sunt foarte similari și metoda are o rată de convergență slabă. Nu se va observa îmbunătățirea soluției. Rata de convergență a strategiilor evoluționiste este maximă în benzi înguste ale pasului ("fereastră de evoluție" mică). O descreștere a pasului în medie asigură convergența în timpul optimizării.

Prima optimizare a unui dispozitiv electromagnetic folosind algoritmi genetici este prezentată în [159]. Aici se optimizează forma polului unui motor astfel încât distribuția câmpului magnetic să fie sinusoidală în întrefier. Fiecare cromozom are o lungime de 64 de biți (4 variabile de proiectare), dimensiunea populației fiind 30. Pentru reproducere se folosește tehnica selecției tip roată de ruletă. Probabilitatea de încrucișare este de 0.95 iar cea de mutație 0.005. Aceiași autori aplică algoritmi genetici pentru optimizarea unui

⁶"plus strategy"

⁷"comma strategy"

magnetizor [160]. Se caută forma polilor dispozitivului astfel încât să se obțină forma dorită a câmpului magnetic. Dimensiunea populației este 40, lungimea cromozomului este 80 (15 gene și 6 variabile), algoritmul fiind oprit după 50 generații. Variabilele sunt codificate cu ajutorul codului Gray.

Un articol de sinteză despre aplicarea algoritmilor genetici în electromagnetism este [168], el trecând în revistă folosirea algoritmilor genetici la proiectarea antenelor, a filtrelor optice multistrat, a formei pieselor polare și izolatorilor, în rețele de comunicație, joncțiunilor de ghiduri de undă. Autorii spun că natura stocastică a algoritmilor genetici nu trebuie confundată cu o căutare pur aleatoare așa cum evoluția însăși nu este ceva aleator; principiul supraviețuirii celui mai bun lasă loc pentru foarte multe interpretări. Deoarece teoria algoritmilor genetici este incompletă, multe informații despre folosirea cu succes a algoritmilor genetici vine din experiență. Iată câteva sfaturi [168]:

1. Pentru algoritmi genetici (codificare binară a variabilelor) probabilitățile de mutație și încrucișare sunt adesea folosite între limitele $0.6 \leq p_{incr} \leq 0.9$ și $0.001 \leq p_{mut} \leq 0.01$. În general p_{mut} corespunde mutației a una, cel mult două gene din cromozom și a cel puțin câțiva cromozomi dintr-o generație. Algoritmii genetici cu codificare reală (programele de evoluție) utilizează adesea probabilități de mutație mai mari.
2. Problemele care au parametri care variază continuu sunt adesea (dar nu întotdeauna) mai potrivite pentru algoritmi genetici cu codificare reală. În general codificarea problemei fizice este mai bună dacă ea seamănă mai mult modelului.
3. Includerea unui algoritm de optimizare locală fie pentru a crea populația inițială, fie pentru a optimiza populațiile intermediare, fie pentru a îmbunătăți populația finală grăbesc procesul de optimizare.
4. Selecția "binary tournament" în general lucrează mai bine decât roata de ruletă și evită necazurile legate de convergență.
5. Când algoritmul genetic stagnează, creșterea probabilității de mutație dă informații utile: dacă funcția obiectiv a celei mai bune configurații se schimbă semnificativ atunci înseamnă că algoritmul a fost temporar agățat într-un optim local și continuarea optimizării este utilă. Dacă nu, un nou start al algoritmului ar avea mai multe șanse decât a continua cu vechea populație.

Deși nu se referă la optimizarea unui dispozitiv electromagnetic ci la identificarea unui model de histerezis, în [23] se găsesc câteva informații utile. Autorii folosesc pentru un algoritm genetic cu codificare binară probabilitățile de 53 % pentru încrucișare și 7 % pentru mutație. Valorile acestor probabilități reprezintă principala diferență între algoritmii genetici și programele de evoluție. În programele de evoluție mutația joacă

un rol central, pe când în algoritmi genetici încrucișarea este cea esențială. Valorile acestor parametri sunt determinate cu ajutorul unui meta-algoritm genetic care folosește o combinație de strategii stocastice și deterministe. Acest meta-algoritm este costisitor dar poate îmbunătăți convergența algoritmului principal, factorul de accelerare ajungând până la valoarea 100.

Problema alegerii parametrilor unui algoritm de evoluție μ, ρ, λ este rezolvată în [100] făcându-se teste pe funcții bine cunoscute (funcția lui Rosenbrock, funcția lui Rastrigin, definite de exemplu și în [104]). Parametrii astfel găsiți sunt folosiți apoi pentru optimizarea dispozitivului electromagnetic propriu-zis. De asemenea se constată că introducerea unui "dezastru" (după un număr de generații amplitudinea mutației este foarte mare) îmbunătățește performanța globală a algoritmului.

O idee asemănătoare se găsește în [169], unde analiza influenței parametrilor unui algoritm genetic asupra găsirii unui optim global se face pe o funcție analitică

$$f(x,y) = 1 + \frac{\sin(\sqrt{x^2 + y^2})}{\sqrt{x^2 + y^2}},$$

care are extreme locale și un extrem (maxim) global cunoscut. Autorii definesc un algoritm genetic ca fiind eficient dacă satisface două cerințe: prima este aceea că algoritmul trebuie să găsească soluția prin evaluarea doar a câteva combinații din spațiul de căutare; a doua cerință este aceea ca algoritmul genetic să aibă o probabilitate de succes cel puțin mai mare decât a unui algoritm de căutare complet aleatoare. Prima condiție este foarte importantă căci aplicațiile reale au un domeniu de căutare foarte mare. A doua cerință pare evidentă dar autorii arată că pot exista configurații ale parametrilor algoritmului genetic care îl pot face mai puțin eficient decât un algoritm de căutare aleatoare. A măsura performanța unui algoritm genetic numai după evoluția valorii medii a funcției de cost poate fi înșelător pentru că se poate întâmpla ca după câteva generații majoritatea punctelor (indivizilor) să fie concentrate în jurul unui optim local. Strategia de a măsura performanțele unui algoritm genetic este următoarea:

1. Se alege o funcție bine cunoscută (ca cea de mai sus) care are mai multe extreme cunoscute.
2. Pe această funcție se testează mai multe configurații de algoritmi genetici (diferite codificări, diferite probabilități de mutație și încrucișare, diferite dimensiuni ale populației, diferite mecanisme de selecție);
3. Numărul de apeluri ale funcției obiectiv se limitează la un anumit procent din numărul de puncte posibile care există în spațiul de căutare (să notăm cu A acest procent care reprezintă numărul maxim de combinații evaluate);

4. Se determină probabilitatea de succes a algoritmului genetic în această configurație $Pag(A)$. Algoritmul a avut succes dacă a găsit configurația optimă. Probabilitatea de succes se determină rulând algoritmul de 1000 ori și aplicând apoi formula: $Pag(A) = \text{nr.rulări cu succes} / 1000$.
5. Pentru a măsura performanța reală a algoritmului genetic $Pag(A)$ este comparată cu probabilitatea $Prand(A)$ de succes a unui algoritm de căutare pur aleatoare care face același număr de evaluări de funcții obiectiv. Se poate demonstra că probabilitatea teoretică de a găsi p optime într-un spațiu de căutare cu n combinații posibile, prin inspectarea a A % din numărul de combinații posibile este dat de formula

$$Prand(A) = \sum_{i=1}^k \frac{p}{n} \left(1 - \frac{p}{n}\right)^{i-1}, \quad \text{unde } A = \frac{k}{n}100. \quad (2.2)$$

Din testele făcute pe funcția $f(x, y)$ de mai sus autorii au ajuns la concluzia că o codificare reală este mult mai eficientă decât una binară iar un mecanism de selecție de tip "stochastic remainder" este mai bun decât roata de ruletă. De asemenea folosirea unei strategii elitiste duce la un algoritm eficient decât nefolosirea ei, caz în care cel mai bun individ s-ar putea pierde prin mutații și încrucișări.

6. Algoritmul genetic ai cărui parametri sunt pregătiți ca mai sus este aplicat apoi problemei reale de optimizat.

Autorii aplică această strategie optimizării numărului de fire conductoare dintr-un conductor multifilar. Problema care apare este cât de asemănător se comportă funcția reală de optimizat cu funcția test folosită în stabilirea parametrilor. Ceea ce rămâne de făcut este de a găsi un set de funcții test cunoscute care să fie potrivite în testarea configurației algoritmilor genetici.

- *Strategii evoluționiste combinate cu simulated annealing*

În ideea de a combina avantajul algoritmului simulated annealing de a evita agățarea în minime locale și abilitatea algoritmilor evoluționiști de a folosi informațiile despre stările deja inspectate, au apărut variante combinate de algoritmi [31, 32, 55, 81, 84]. Variațiile aleatoare ale variabilelor sunt făcute pe principiile strategiilor evoluționiste iar selecția noilor valori în concordanță cu regulile algoritmului simulated annealing. În [54] se descrie un astfel de algoritm, dându-i-se interpretări din statistică, biologie, termodinamică.

2.2.4 Abordări deterministe comparate cu abordări stocastice

Multe lucrări investighează mai multe metode de optimizare, aplicate aceleiași probleme [18, 58, 108, 130, 142, 147]. Din păcate o mare parte dintre ele se mulțumesc doar cu

compararea rezultatelor finale, nefurnizând informații despre timpul de calcul (numărul de evaluări ale funcției de cost).

În [130] se optimizează forma polilor (cazuri 2D, 3D, neliniare) astfel încât să se minimizeze neomogenitatea câmpului magnetic într-o anumită zonă. Pentru aceasta au fost testate diferite tehnici de optimizare, forma polilor fiind aproximată cu polinoame de ordin patru, șase sau opt. Autorii compară trei metode deterministe (pașilor descendenți, gradienti conjugați și quasi-Newton) cu strategii evoluționiste (1+1) (metoda Monte-Carlo) și (1,6). Pentru a obține rezultate într-un timp rezonabil, pentru rezolvarea problemei neliniare s-a folosit un algoritm care combină iterațiile directe cu metoda Newton, pentru rezolvarea sistemului în cazul iterațiilor directe folosind descompunere incompletă Cholesky. Concluzia autorilor este că metodele evoluționiste sunt comparabile cu metodele deterministe de ordin superior din punct de vedere al vitezei de convergență, dar mai robuste decât acestea, fiind deci superioare lor.

Concluzia de mai sus nu se desprinde din [147] unde sunt folosite pentru optimizarea configurației unui sistem de conductoare metoda gradientilor conjugați și metoda simulated annealing, pentru aceasta din urmă fiind nevoie de aproximativ 10^6 evaluări de funcții. Și în [18] se afirmă că metoda deterministă este mult mai rapidă decât cea stocastică (simulated annealing), dar este puternic dependentă de punctul inițial.

În [108] se compară eficiența mai multor proceduri de optimizare: deterministe, bazate pe strategii de evoluție și algoritmi genetici. Problema se referă la proiectarea formei polilor unui magnet, având 11 variabile de proiectare. Unul din algoritmi de optimizare determiniști folosiți a constatat în minimizări unidimensionale prin intermediul extrapolării Powell într-o direcție principală de căutare (definită de utilizator) și o direcție ortogonală evaluată prin ortogonalizare Gram-Schmidt. După ce aceste minimizări au fost făcute (sfârșitul pasului de căutare) direcția principală de căutare a fost recalculată ca fiind un vector care unește punctul anterior cu minimumul găsit în pasul de căutare. Un al doilea algoritm determinist folosit este algoritmul Davidon-Fletcher-Powell care folosește o aproximație pătratică a funcției obiectiv, direcția de căutare fiind dată de $\Delta \mathbf{x} = -\mathbf{H}^{-1} \text{grad } f(\mathbf{x})$. Matricea Hessian nu este calculată de fiecare dată. Algoritmii determiniști au necesitat 218 evaluări de funcții și, în restricțiile date, a dus la un profil colțuros. Pentru a obține un profil neted ar fi nevoie de restricții suplimentare. Algoritmii determiniști au convins foarte repede însă nu există siguranța că extremele găsite sunt globale. Strategia evoluționistă folosită a fost de tip $(\mu/\rho, \lambda)$ cu o rulare simulated annealing la fiecare 50 iterații. Problema a fost abordată și cu algoritmi genetici în care fiecare variabilă a fost reprezentată de un cromozom cu 8 biți (codul Gray), un individ fiind reprezentat pe 88 biți. Numărul de indivizi dintr-o generație este 30 iar numărul de generații este 100. Autorii evidențiază necesitatea analizei sensibilităților pentru a determina comportarea funcției obiectiv la variația parametrilor de optimizare. De asemenea curba care definește forma polilor ar

trebui să fie derivabilă în orice punct pentru a evita colțurile.

Algoritmii genetici sunt comparați cu algoritmi determiniști și în [58], problemele de test fiind legate de optimizarea antenelor. Concluzia este aceea că algoritmii bazați pe gradient sunt mai eficienți pentru un număr mic de parametri care pot varia continuu, iar algoritmii genetici sunt foarte potriviți în cazul în care sunt mulți parametri de optimizat și mărimile variază discret.

Dispozitivele SMES ⁸ apar foarte des în articolele care tratează optimizarea dispozitivelor electromagnetice. Un astfel de articol este și [142]. O primă abordare este una analitică în care autorii au inspectat distanța de la care câmpul magnetic scade sub o anumită valoare. O a doua abordare a fost o strategie de evoluție (1+1) cu maximum 300 generații, o a treia a fost o strategie deterministă (gradienti conjugați) în care seziitivitățile au fost calculate prin derivare numerică, iar apoi o strategie cuplată: evoluție + gradienti conjugați. Concluzia este aceea că o strategie cuplată este cea mai eficientă (din punct de vedere al numărului de evaluări ale funcției obiectiv). Nu este specificat criteriul de trecere de la o strategie la alta.

2.2.5 Abordări hibride

Tendința naturală de a cupla avantajele algoritmilor determiniști (viteza de convergență cu avantajele algoritmilor stocastici (căutarea globală)) a dus la apariția algoritmilor hibridi, în care cele două tipuri de algoritmi sunt aplicate succesiv sau sunt amestecate rezultând o strategie nouă.

- **Combinații cu simulated annealing**

În [50] se arată că dispozitivele de stocare a energiei (SMES) cu mai mult de două bobine pot să fie mai eficiente decât cele cu două bobine. Analiza este făcută atât pentru cazul fără ecran magnetic cât și cu ecran magnetic. În acest din urmă caz optimizarea formei ecranului feromagnetic este făcută combinând strategii stocastice cu deterministe. Lucrarea arată cum pot fi extrase informații utile prin evaluarea senzitivităților. În cazul în care nu există ecran feromagnetic analiza problemei se face prin proceduri analitice; în cazul în care problema este neliniară se folosește metoda elementelor finite. Pentru a reduce necesarul de timp se procedează astfel: în primul pas se optimizează dispozitivul în aer pentru a obține energia dorită și un câmp de dispersie cât mai mic; al doilea pas constă în alegerea formei ecranului feromagnetic astfel încât să se minimizeze zona interzisă din exteriorul dispozitivului. Optimizarea este astfel mai ușoară din cauză că se diminuează numărul de parametri, dar ea ar putea conduce la un optim local. Procedura folosită pentru cazul liniar se bazează pe algoritmul simulated annealing urmat de o procedură deterministă (Powell) care pornește din punctul optim găsit de algoritmul stocastic.

⁸”Superconducting Magnetic Energy Storage System”

Pentru etapa a două se folosește doar o procedură deterministă (gradienti conjugați) și analiza sensibilităților. Funcția obiectiv folosită este una în care diferitele obiective sunt ponderate. Forma finală obținută depinde puternic de alegerea acestor ponderi.

O strategie hibridă pentru optimizarea polilor unui motor sincron este prezentată în [43]. Suprafața polilor trebuie proiectată astfel încât să producă o distribuție dreptunghiulară a inducției magnetice în întrefierul mașinii. Suprafața polilor este modelată cu ajutorul unor funcții de interpolare spline. Estimarea funcției obiectiv se face cu ajutorul metodei elementelor finite (metoda Newton-Raphson pentru rezolvarea problemei neliniare) iar optimizarea se face cu algoritmul simulated annealing urmat de o metodă de căutare directă (de tip metoda simplexului). În urma executării algoritmului simulated annealing cele mai bune $n + 2$ puncte (unde n este numărul de parametri) sunt memorate, iar condiția de trecere de la un algoritm la altul este: $(F_{max} - F_{min})/F_{av} \leq \varepsilon$ unde F_{max} reprezintă valoarea maximă dintre cele $n + 2$ puncte alese, F_{min} este valoarea minimă iar F_{av} este valoarea medie. Implementarea metodei arată că timpul de calcul al metodei hibride este 27 % din timpul de calcul al algoritmului SA singur și că soluția optimă nu depinde de punctul inițial de start.

Tot o strategie hibridă este folosită și în [5], pentru optimizarea bobinelor dintr-un sistem MRI. Algoritmul începe cu simulated annealing și continuă cu un algoritm determinist (metoda Powell). Autorii nu dau informații despre criteriul de trecere de la un algoritm la altul.

Algoritmul simulated annealing este cuplat cu o căutare Hooke și Jeeves în [38] și cu metoda de căutare Tabu în [161].

• Combinații cu strategii evoluționiste

În [163] se aplică o tehnică hibridă pentru optimizare globală combinând algoritmi genetici cu metodele deterministe. Avantajul unei astfel de strategii este acela că soluția finală este mai precisă. O problemă intrinsecă a unei tehnici hibride este momentul în care se trece de la o metodă la alta, criteriul după care se face această comutare fiind dependent de problemă, un astfel de criteriu trebuind să lanseze algoritmul determinist după ce a fost identificată zona în care se află minimumul global. Puține lucrări care folosesc metode hibride dau și criterii care să indice când să se treacă de la o metodă la alta. Încrucișarea este cea mai importantă operație care generează noi puncte în domeniul de căutare al optimului global. Mutația este cel mai important operator care introduce caracteristici genetice noi în populație. La fiecare generație cel mai bun cromozom este preluat de algoritmul de optimizare determinist. Probabilitățile de aplicare a operatorilor mutație și încrucișare sunt schimbate în mod dinamic în funcție de raportul dintre valoarea medie a funcției de cost și valoarea ei maximă. Când acest raport este apropiat de 1 înseamnă că nu există o diversitate prea mare în populație așa încât probabilitatea de a aplica mutația crește și probabilitatea de a aplica încrucișarea scade. Când raportul este apropiat de zero

înseamnă că varietatea genetică în populație este prea mare și probabilitatea de mutație este micșorată iar cea de încrucișare este crescută. Este folosit de asemenea și operatorul de inversiune. Se încearcă definirea mai multor criterii de comutare de la o metodă la alta și se testează aceste criterii pe o problemă tipică de optimizare globală (funcția lui Rastringin - funcție analitică care are 10^n minime locale, n fiind un parametru întreg al funcției). Criteriile propuse au fost: numărul de generații, diferența între valoarea medie a funcției de cost între două generații succesive, diferența dintre cele mai bune valori între generații, diferența între cele mai bune două valori în aceeași generație. Pentru orice criteriu propus au existat și situații în care algoritmul nu a converș către optimul global.

În variantele lor tradiționale algoritmi genetici, deși sunt destul de robusți, se consideră că nu sunt competitivi cu alte abordări euristice precum căutarea Tabu sau simulated annealing [48]. De aceea majoritatea implementărilor algoritmilor genetici sunt variante modificate sau cuplate cu algoritmi de tip "hill-climbing". Algoritmi care să cupleze simulated annealing cu proceduri de optimizare deterministe au fost prezentați [44] și s-au dovedit a fi foarte eficienți. În ultimul timp au apărut și algoritmi hibridi care cuplează algoritmi genetici cu alți algoritmi [22]. [48] propune un algoritm genetic modificat cuplat cu metoda deterministă a simplexului coborător a lui Nelder și Mead. Algoritmul este testat pe o problemă de NMR ⁹. Ideile principale ale algoritmului genetic modificat sunt următoarele:

1. Calitatea populației inițiale afectează puternic performanțele algoritmului genetic, aceasta trebuind să aibă indivizi destul de buni și destul de diverși.
2. Încrucișarea este un operator critic pentru succesul unui algoritm genetici, în particular individul cel mai bun nu trebuie pierdut.
3. Pentru a evita convergența prematură a algoritmului este convenabil să se introducă diversitate în populație prin intermediul mutației sau al migrării.

După mai multe încercări optimizarea principală este făcută de o serie de rulări ale algoritmului simplex. Algoritmul genetic face o explorare largă a domeniului de căutare și este crucial în pregătirea ultimelor rulări ale metodei simplex. Codificarea variabilelor este reală, mai potrivită pentru un domeniu de variație continuă a parametrilor. Asigurarea diversității populației inițiale este făcută alegând aleator puncte în interiorul fiecărei celule a unei rețele uniforme ce discretizează domeniul de căutare. Populația inițială este determinată în doi pași. Mai întâi un număr de cele mai bune configurații este preluat de algoritmul simplex. Minimele locale rezultate constituie prima parte a populației inițiale. Restul populației inițiale se ia din mulțimea generată aleator (tot în ordinea valorică). Cu această abordare populația inițială este un eșantion bun atât din punctul de vedere

⁹Nuclear Magnetic Resonance

al distribuției punctelor în domeniu cât și din punctul de vedere al valorii lor. Autorii propun și o variantă nouă de operator de încrucișare. Rezultate bune se obțin dacă se cuplează la întâmplare doi părinți, cuplul generând un singur copil care îl înlocuiește pe părintele mai puțin valoros. Aici copilul nu ia de la părinte valoarea cromozomului ci intervalul în care se află, luând aleator o valoare în acel interval. O îmbunătățire suplimentară se observă dacă în plus se selectează aleator una din variabile și acesteia i se atribuie valoarea medie a părinților. Pentru a evita convergența prematură a populației și pentru a mări capacitatea de explorare a algoritmului este esențial să se aplice mutația după încrucișare. O alternativă a mutației este migrarea care constă în înlocuirea unor indivizi cu alții selectați din mulțimea inițială de indivizi rezultată din împărțirea domeniului în celule. Pentru a reduce numărul total de iterații se execută o optimizare locală finală care se aplică unui număr de indivizi din populația finală. Pentru a asigura și aici o căutare într-un spațiu mai larg punctele de start ale optimizării finale nu sunt indivizii cei mai buni ci sunt alese sortând indivizii după valoare și luând uniform indivizi de la cei mai buni ajunând până la cei mai răi. Această optimizare finală este făcută în paralel. Acest algoritm genetic hibrid a fost testat pe o problemă de MRI, cu un obiectiv care își propune un câmp magnetic uniform în interiorul unei suprafețe sferice în interiorul structurii. Structura magnetică conține șase bobine axiale, conectate în serie, având raza interioară și grosimea radială fixate. Variabilele de optimizare sunt poziția axială a centrului bobinelor și grosimea lor axială. Valorile câmpului sunt calculate prin integrare analitică a formulei Biot-Savart-Laplace. Au fost folosite 10 subintervale pentru fiecare domeniu al unei variabile, numărul de celule inițiale 1000, dimensiunea populației 50, 20 rulări simplex pentru calculul populației inițiale și 10 rulări simplex pentru optimizarea finală. Pentru comparație au fost făcute și rulări cu alți algoritmi: algoritmi genetici standard, simulated annealing, simplex cu multistart. Fiecare algoritm a fost rulat de 300 ori pornind din puncte de start diferite. Algoritmul propus s-a dovedit a fi cel mai bun pentru problema de test. El este urmat de simulated annealing. Algoritmul simplex nu a dus niciodată la soluții mai bune chiar dacă se folosesc pentru start punctele inițiale ale noului algoritm.

O combinație a strategiilor evoluționiste cu metoda simplexului se întâlnește și în [22]. Optimumul se localizează cu ES și o discretizare rară a domeniului, căutarea finală fiind făcută cu simplex pentru o discretizare fină.

Algoritmii genetici sunt combinați cu metode deterministe bazate pe gradient în [82, 109] pentru analiza sensibilităților fiind folosite abordări discrete.

2.2.6 Concluzii

Există multe dificultăți în optimizarea dispozitivelor electromagnetice. Ele sunt legate de:

1. acuratețea și timpul de calcul necesar evaluării funcției de cost;
2. acuratețea evaluării derivatelor în metodele de ordin superior;
3. convergența algoritmului de optimizare către un optim global;
4. viteza de convergența a algoritmului de optimizare, aceasta implicând timpul de calcul necesar.

Primele două aspecte sunt strâns legate de metoda de analiză a câmpului electromagnetic. Dacă problema are soluție analitică atunci aceasta este de preferat unei metode numerice. Calculele de gradienti sunt întotdeauna mai precise pentru o astfel de metodă. Urmează apoi metodele integrale (care nu necesită discretizarea aerului), iar cele mai puțin precise calcule de gradient se fac atunci când metoda de analiză a câmpului se bazează pe un model diferențial. Dacă rețeaua de discretizare folosită pentru metoda elementelor finite nu este construită în anumite condiții atunci calculul gradientilor este eronat [60]. A face discretizări diferite pentru fiecare configurație înseamnă a folosi modele numerice diferite în timpul procesului de optimizare. A fost recunoscut faptul că topologii diferite de rețea cauzează discontinuități în mărimile de câmp ca funcțiile de variabilele de proiectare, conducând la funcții obiectiv discontinue [167]. Majoritatea lucrărilor care folosesc metoda elementelor finite nu își pun însă problema măsurii în care rețele de discretizare diferite pot induce extreme locale false în funcția de cost. Pentru a obține valori bune pentru gradienti proprietățile topologice ale rețelei trebuie menținute constante [150]. Se propune chiar aplicarea tehnicii din ingineria mecanică unde un nod are drep necunoscute și valorile coordonatelor sale. Într-o problemă 2D vor fi deci trei necunoscute pe nod. În [150] se propune ca rețeaua să fie schimbată numai în zona interfețelor care se modifică, iar această schimbare să se facă continuu. O altă alternativă este prezentată în [80] care prezintă conceptul de "rețea elastică"¹⁰ care se adaptează schimbărilor parametrilor. Aceste aspecte fac ca metodele deterministe să fie de fapt pseudo-deterministe. În [54] se afirmă că dacă o metodă deterministă se comportă acceptabil, aceasta nu se întâmplă din cauză că sunt satisfăcute condițiile de eficiență (convexitate, diferențiabilitate, acuratețe) ci dimpotrivă, aceste principii sunt atât de violate încât procedura de optimizare ascunde în ea ceva aleator.

Multe lucrări scot în evidență rapiditatea cu care o metodă deterministă ajunge la un optim (numărul de evaluări de funcții obiectiv este de ordinul zecilor). În plus, aceste

¹⁰"rubber mesh"

metode furnizează valori pentru multiplicatorii Lagrange asociați restricțiilor, informații deosebit de utile pentru proiectanți. De aceea metodele deterministe sunt mai degrabă recomandate de alți autori [167]. Performanța unei metode deterministe depinde de punctul de start (de cele mai multe ori se practică tehnica multistart), de acuratețea evaluării direcției descendente, de metoda folosită pentru minimizarea unidimensională și de criteriul de oprire.

Dezavantajul metodelor stocastice (aplicate în variante clasice) este acela că necesită foarte multe evaluări de funcții obiectiv. Cum această evaluare este costisitoare în cazul optimizării unui dispozitiv electromagnetic, o astfel de metodă necesită un timp de calcul mare. Componenta aleatoare a metodelor stocastice face posibil faptul că, la rulări diferite dar care au aceiași parametri și aceleași condiții inițiale rezultatele sunt diferite. De aceea în cazul folosirii unei astfel de metode se practică rulările multiple în aceleași condiții pentru a vedea câte (dintr-un anumit număr) reușesc să găsească același optim (global).

O alegere potrivită a strategiei stocastice (de exemplu cea care combină strategiile evoluționiste cu algoritmul simulated annealing recomandată în [54]) face ca o astfel de metodă să fie mai rapid convergentă decât orice metodă deterministă. Figura 2.6 prezintă variația funcției obiectiv în funcție de numărul de apeluri ale funcției de cost în cazul strategiei globale aleatoare și în cazul folosirii metodelor deterministe. Graficul reprezintă o medie făcută pe patru probleme de test. În medie, algoritmul stocastic global este comparabil cu metodele deterministe de ordin superior pentru problemele simple și mult superior acestora pentru problemele complexe (unde metodele deterministe nu există în sensul strict al cuvântului).

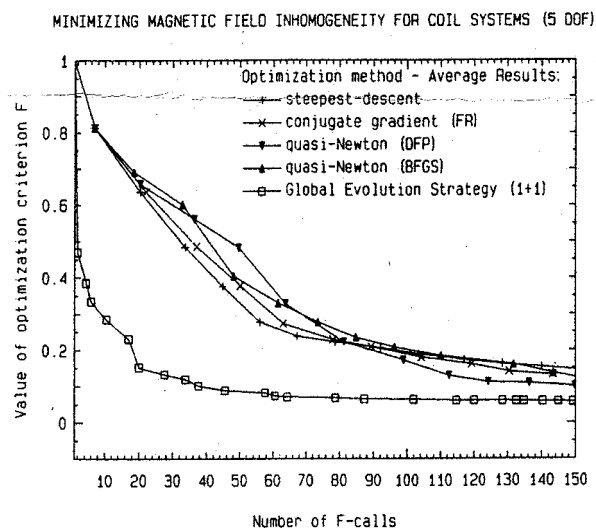


Figura 2.6: Funcția de cost în funcție de numărul de apeluri ale ei

La terminarea unei execuții a unui algoritm de optimizare trebuie făcute investigații suplimentare cum ar fi:

- Este rezultatul obținut într-adevăr cel optim?
- Cum sunt obiectivele și restricțiile influențate de toleranțele variabilelor de proiectare?

La aceste întrebări răspunsul poate fi găsit prin aplicarea analizei sensibilităților și estimarea multiplicatorilor Lagrange [136]

Un alt aspect care trebuie luat în considerare este acela că rezultatul final trebuie să garanteze un dispozitiv manufacturabil. De aceea parametrizarea geometriei trebuie să impună restricții legate de netezimea conturului geometric, care trebuie să fie cel puțin continuu. Multe probleme de optimizare permit numai o variație discretă a parametrilor (de exemplu diametrele și secțiunile sunt standardizate) și pentru aceste probleme nu pot fi aplicate metode deterministe.

În concluzie nu există metodă perfectă de optimizare. Pentru orice problemă trebuie încercate și analizate mai multe metode. A utiliza algoritmul de optimizare ca o "cutie neagră" este foarte periculos, fiecare algoritm comportându-se diferit în funcție de aplicația pe care dorește să o rezolve. Metoda de rezolvare trebuie și ea aleasă cu grijă. Acolo unde problema o permite sunt de preferat metodele analitice sau formule matematice aproximative, cel puțin într-o primă etapă. În [60] se sugerează combinarea avantajelor tuturor metodelor (stocastice, deterministe, de inteligență artificială) într-o strategie de tipul următor. Proiectarea inițială trebuie făcută clasic, adică utilizând modele de circuite magnetice pentru dispozitive. Aceasta conduce la o configurație aproximativă. Se folosește apoi un program de elemente finite (de exemplu) pentru a genera perechi parametru-funcție de cost, cu acestea fiind "antrenată" o rețea neurală. După ce aceasta învață funcția, ea va furniza parametrul pentru o valoare dorită a funcției de cost. Folosind acest rezultat ca punct de start se poate aplica o metodă deterministă de ordinul unu pentru a rafina soluția. Totul seamănă cu o orchestră în care fiecare devine solist la momentul potrivit și fiecare cântă, sub bagheta dirijorului, pentru a desăvârși simfonia.

Capitolul 3

Analiza numerică a câmpului magnetic staționar în medii neliniare

3.1 Algoritmi pentru rezolvarea problemelor neliniare de regim magnetic staționar

Modelul numeric folosit în acest capitol se bazează pe formularea diferențială în potențial vector și metoda elementelor finite. Regimul de studiu al câmpului este cel magnetic staționar în medii neliniare magnetice. Nu sunt studiate mediile cu histerezis, iar detalierea metodelor se face pentru problemele plan-paralele. Scopul capitolului este de a genera algoritmi pentru analiza numerică a câmpului magnetic, algoritmi care să fie robusți și optimali din punctul de vedere al resurselor de calcul necesare, în special timpul CPU.

Notății folosite:

- Domeniul de studiu se notează cu $\Omega \subset \mathbb{R}^3$ și se consideră mărginit de suprafața închisă $\partial\Omega$. $\partial\Omega$ se consideră partiționată în S_h și S_b , S_h fiind reuniunea unui număr finit n de suprafețe disjuncte S_{hi} :

$$\partial\Omega = S_h \cup S_b, \quad S_h \cap S_b = \emptyset, \quad S_h = \cup S_{hi}.$$

Domeniul Ω poate fi multiplu conex, ordinul de conexitate fiind $p + 1$.

- În cazul problemei plan-paralele, $\Omega \subset \mathbb{R}^2$, frontiera se notează tot cu $\partial\Omega$, partițiile fiind în acest caz C_h și C_b .

Restul notațiilor folosite sunt cele uzuale.

3.1.1 Formularea corectă a problemei

Pentru detalierea și studiul metodelor prezentate vom folosi formularea în \mathbf{B} și \mathbf{H} și formularea în potențial magnetic vector.

Formularea în \mathbf{B} și \mathbf{H} :

Pentru analiza câmpului magnetic staționar trebuie rezolvat sistemul:

$$\operatorname{div} \mathbf{B} = 0 \quad , \quad \operatorname{rot} \mathbf{H} = \mathbf{J} \quad , \quad \mathbf{B}(P) = f(\mathbf{H}(P)), \quad (3.1)$$

unde $P \in \Omega$ iar \mathbf{J} și relația constitutivă $f \in \mathcal{U}$ (mulțimea funcțiilor lipschitziene și uniform monotone) sunt cunoscute (presupunem deci că nu există pânze de curent în Ω) și de asemenea se cunosc condițiile pe frontieră:

- (a) Pe S_h se dă $\mathbf{H}_t = \mathbf{f}$;
- (b) Pe S_b se dă $B_n = g$;
- (c) Pe $n - 1$ din cele n suprafețe S_{hi} se dau fluxurile lui \mathbf{B} , deci $\int_{S_{hi}} B_n dA = \Phi_i$, cu $i = 1, \dots, n - 1$;
- (d) Pe curbele $\Gamma_1, \dots, \Gamma_p$ ce înconjoară găurile lui Ω se dau circulațiile lui \mathbf{H} , deci $\oint_{\Gamma_m} \mathbf{H} d\mathbf{r} = i_m$, cu $m = 1, \dots, p$.

În aceste condiții problema este corect formulată [65] existând soluție unică și stabilă.

Formularea în \mathbf{A} :

Pentru rezolvarea problemei de câmp magnetic staționar trebuie rezolvată ecuația

$$\operatorname{rot} (\nu \operatorname{rot} \mathbf{A}) = \mathbf{J} \quad (3.2)$$

cu condiția de etalonare

$$\operatorname{div} (\sigma \mathbf{A}) = 0 \quad (3.3)$$

și condițiile de frontieră:

- (a) Pe S_h se dă $(\nu \operatorname{rot} \mathbf{A})_t = \mathbf{f}$;
- (b) $\sigma \mathbf{A} \cdot \mathbf{n} = 0$ pe S_h ;
- (c) Pe S_b se dă $\mathbf{A}_t = \mathbf{g}$;
- (d) Dacă S_b este formată din n suprafețe disjuncte S_k , atunci pe $n - 1$ din ele se dau fluxurile lui $\sigma \mathbf{A}$, deci $\int_{S_k} \sigma \mathbf{A} \cdot \mathbf{n} dA = 0$, pentru $k = 1, \dots, n - 1$.

Mărimile ν și σ sunt funcții scalare pozitive definite pe Ω , prima fiind mărimea de material (reluctivitatea) iar a doua fiind aleasă arbitrar. Conform teoremei de unicitate [65] și această problemă este corect formulată.

3.1.2 Metoda polarizației

Pentru a prezenta această metodă vom folosi mai întâi formularea în \mathbf{B} și \mathbf{H} .

Deoarece $f \in \mathcal{U}$ atunci există f^{-1} și $f^{-1} \in \mathcal{U}$. În cele ce urmează, notăm $h = f^{-1}$, deci

$$\mathbf{H} = h(\mathbf{B}). \quad (3.4)$$

Ideea metodei polarizației constă în înlocuirea relației (3.4) cu următoarele două relații:

$$\mathbf{B} = \mu \mathbf{H} + \mathbf{I}, \quad (3.5)$$

$$\mathbf{I} = \mathbf{B} - \mu h(\mathbf{B}) = g(\mathbf{B}). \quad (3.6)$$

În aceste relații \mathbf{I} este de natura unei polarizații magnetice iar μ de natura unei permeabilități. Dacă alegem cu grijă μ (și anume $\mu \in (0, \frac{2\lambda}{\Lambda^2})$, unde λ, Λ sunt constantele funcției h - de monotonie, respectiv Lipschitz), atunci g este o contracție [66].

Observații legate de alegerea permeabilității de calcul μ :

1. În funcție de dependența h se pot alege permeabilități de calcul μ astfel încât să avem anumite avantaje, de exemplu:

- factorul de contracție să fie minim;
- mediul de calcul să fie omogen (μ constant în Ω);
- mediul de calcul să fie omogen pe porțiuni.

2. În cazul în care mediul neliniar de calcul este omogen pe porțiuni, se demonstrează în [66] că g este contracție în fiecare punct al lui Ω dacă μ se alege astfel încât

$$\mu \in \left(0, \frac{2}{\nu_{max}}\right) = (0, 2\mu_{min}), \quad (3.7)$$

valoarea optimă a lui μ (pentru care factorul de contracție este minim) fiind

$$\mu_{optim} = \frac{2}{\nu_{max} + \nu_{min}}. \quad (3.8)$$

În relațiile de mai sus

$$\begin{aligned} \mu_{max} &= \sup_{H', H''} \frac{f(H') - f(H'')}{H' - H''} = \frac{1}{\nu_{min}}, \\ \nu_{min} &= \inf_{B', B''} \frac{h(B') - h(B'')}{B' - B''}, \\ \mu_{min} &= \inf_{H', H''} \frac{f(H') - f(H'')}{H' - H''} = \frac{1}{\nu_{max}}, \\ \nu_{max} &= \sup_{B', B''} \frac{h(B') - h(B'')}{B' - B''}. \end{aligned}$$

μ_{max} reprezintă cea mai mare creștere de pe curba $\mathbf{B} - \mathbf{H}$, fiind un fel de permeabilitate dinamică dar care nu necesită derivabilitatea funcției $\mathbf{B} - \mathbf{H}$ (condiție cerută de metoda Newton).

Deoarece procedura este convergentă pentru $\mu \in (0, 2\mu_{min})$ și cum orice material are practic $\mu \geq \mu_0$, putem alege de exemplu valoarea de calcul a lui μ ca fiind: $\mu = \mu_0$.

Alegerea lui μ poate fi făcută la modul general dacă $\mu \in (0, \frac{2\lambda}{\Lambda})$. În termenii ecuației caracteristice $H = h(B)$, λ provine din monotonie, deci $\lambda = \nu_{min}$, iar Λ provine de la condiția Lipschitz, deci $\Lambda = \nu_{max}$. Rezultă că putem alege

$$\mu \in \left(0, \frac{2\nu_{min}}{\nu_{max}^2}\right) = \left(0, \frac{2\mu_{min}^2}{\mu_{max}}\right). \quad (3.9)$$

Algoritmul metodei polarizației - iterația prin \mathbf{B} :

1. inițializează \mathbf{I}

2. stabilește μ

3. repetă

3.1. calculează câmpul pentru problema liniară:

$$\operatorname{div} \mathbf{B} = 0, \operatorname{rot} \mathbf{H} = \mathbf{J}, \mathbf{B} = \mu \mathbf{H} + \mathbf{I}$$

cu \mathbf{I} fixat și cu condițiile de frontieră (a),(b),(c),(d)

3.2. corectează polarizația $\mathbf{I} = \mathbf{B} - \mu h(\mathbf{B})$

până când (diferența între două polarizații succesive devine suficient de mică).

În [66] se demonstrează că această procedura iterativă este convergentă.

Observații:

1. Procedura de calcul are avantajul că mediul are aceeași permeabilitate de calcul μ la toate iterațiile, ca urmare matricea sistemului liniar obținut prin discretizarea ecuațiilor este aceeași în toate iterațiile neliniare.

2. Metoda are o convergență liniară, deci este mai lentă decât metoda Newton-Raphson, însă are avantajul convergenței asigurate.

3. Algoritmul iterativ poate fi gândit și ca funcție de \mathbf{H} în loc de \mathbf{B} . În acest caz, caracteristica $\mathbf{B} = f(\mathbf{H})$ se înlocuiește prin

$$\mathbf{B} = \mu(\mathbf{H} + \mathbf{M}), \quad (3.10)$$

$$\mathbf{M} = \nu f(\mathbf{H}) - \mathbf{H} = p(\mathbf{H}), \quad (3.11)$$

unde \mathbf{M} este un termen de natura unei magnetizații, iar ν este de natura unei reluctivități.

Toate raționamentele se fac dual cu cele de la iterația prin \mathbf{B} . Dacă se alege $\nu \in (0, 2\nu_{min})$ atunci p este contracție. Valoarea optimă a lui ν este

$$\nu_{optim} = \frac{2}{\mu_{min} + \mu_{max}}. \quad (3.12)$$

Ca p să fie contracție este necesar ca

$$\nu \in \left(0, \frac{2}{\mu_{max}}\right) \implies \mu \in \left(\frac{\mu_{max}}{2}, \infty\right), \quad (3.13)$$

condiție esențial diferită de cea de la iterația prin \mathbf{B} . În acest caz, dacă domeniul de calcul conține zone de aer și fier, nu mai putem alege un mediu de calcul omogen.

Algoritmul metodei polarizației - iterația prin \mathbf{H} :

1. inițializează \mathbf{M}

2. stabilește μ

3. repetă

3.1. calculează câmpul pentru problema liniară:

$$\operatorname{div} \mathbf{B} = 0, \operatorname{rot} \mathbf{H} = \mathbf{J}, \mathbf{B} = \mu(\mathbf{H} + \mathbf{M})$$

cu \mathbf{M} fixat și cu condițiile de frontieră (a),(b),(c),(d)

3.2. corectează polarizația $\mathbf{M} = \mu^{-1} f(H) - \mathbf{H}$

până când (diferența între două magnetizații succesive devine suficient de mică).

Rezolvarea sistemului liniar (pasul 3.1.) se poate face prin orice metodă numerică cunoscută. Se demonstrează că dacă problema liniară de câmp se rezolvă cu metoda elementelor finite folosind potențialul scalar, atunci procedura iterativă descrisă anterior converge atunci când corecția se face prin \mathbf{H} ; dacă problema liniară de câmp se rezolvă cu metoda elementelor finite folosind potențialul vector, atunci procedura iterativă descrisă anterior converge atunci când corecția se face prin \mathbf{B} .

În cele ce urmează ne vom concentra atenția asupra pasului 3.1. (rezolvarea problemei liniare) în cazul în care corecția se face prin \mathbf{B} . La acest pas trebuie rezolvat sistemul

$$\operatorname{div} \mathbf{B} = 0, \quad \operatorname{rot} \mathbf{H} = \mathbf{J}, \quad \mathbf{B} = \mu \mathbf{H} + \mathbf{I}, \quad (3.14)$$

la care se adaugă condițiile de frontieră. Polarizația magnetică \mathbf{I} este cunoscută iar μ este permeabilitatea de calcul aleasă la pasul 2.

În cele ce urmează vom folosi formularea în \mathbf{A} pentru rezolvarea problemei liniare de câmp. Considerând metoda elementelor finite, în care se lucrează cu potențialul magnetic vector (din această cauză am ales iterația prin \mathbf{B}), ecuația de rezolvat devine

$$\operatorname{rot} \left(\frac{\operatorname{rot} \mathbf{A}}{\mu} \right) = \mathbf{J} + \operatorname{rot} \left(\frac{\mathbf{I}}{\mu} \right), \quad (3.15)$$

unde condițiile de frontieră sunt cele prezentate în paragraful 3.1.1 (formularea în \mathbf{A}).

Pentru *problemele plan-paralele* (unde $\mathbf{A} = Ak$):

$$\begin{aligned} \operatorname{rot} \left(\frac{\operatorname{rot} \mathbf{A}}{\mu} \right) &= \mathbf{k} \operatorname{div} \left(-\frac{1}{\mu} \operatorname{grad} A \right), \\ \operatorname{rot} \left(\frac{\mathbf{I}}{\mu} \right) &= \mathbf{k} \left[\frac{\partial}{\partial x} \left(\frac{I_y}{\mu} \right) - \frac{\partial}{\partial y} \left(\frac{I_x}{\mu} \right) \right], \end{aligned}$$

unde $\mathbf{I} = I_x \mathbf{i} + I_y \mathbf{j}$ (deoarece \mathbf{B} și \mathbf{H} sunt în planul XY, rezultă că și \mathbf{I} este plasat în acest plan).

Ecuatia (3.15) devine

$$\operatorname{div} \left(-\frac{1}{\mu} \operatorname{grad} A \right) = J + \frac{\partial}{\partial x} \left(\frac{I_y}{\mu} \right) - \frac{\partial}{\partial y} \left(\frac{I_x}{\mu} \right), \quad (3.16)$$

unde pe o parte a frontierei (C_b) se dă A (condiție esențială), iar pe cealaltă parte (C_h) se dă \mathbf{H}_t (condiție naturală).

Să notăm cu \mathbf{I}^T vectorul de componente I_y și $-I_x$, deci

$$\mathbf{I}^T = I_y \mathbf{i} - I_x \mathbf{j}. \quad (3.17)$$

Ecuatia (3.16) se scrie

$$\operatorname{div} \left(-\frac{1}{\mu} \operatorname{grad} A \right) = J + \operatorname{div} \left(\frac{\mathbf{I}^T}{\mu} \right). \quad (3.18)$$

Formulara slabă a ecuației (3.18) are forma

$$\int_{\Omega} \Psi_i \operatorname{div} \left(-\frac{1}{\mu} \operatorname{grad} A \right) d\Omega = \int_{\Omega} J \Psi_i d\Omega + \int_{\Omega} \Psi_i \operatorname{div} \left(\frac{\mathbf{I}^T}{\mu} \right) d\Omega. \quad (3.19)$$

Aplicând formulele Green, relația (3.19) devine

$$\begin{aligned} & \int_{\Omega} \frac{1}{\mu} \operatorname{grad} A \cdot \operatorname{grad} \Psi_i d\Omega - \int_{\partial\Omega} \Psi_i \frac{1}{\mu} \operatorname{grad} A \cdot \mathbf{n} ds = \\ & = \int_{\Omega} J \Psi_i d\Omega - \int_{\Omega} \frac{\mathbf{I}^T \cdot \operatorname{grad} \Psi_i}{\mu} d\Omega + \int_{\partial\Omega} \Psi_i \frac{\mathbf{I}^T \cdot \mathbf{n}}{\mu} ds. \end{aligned} \quad (3.20)$$

Pe C_b (frontiera pe care se impune A) $\Psi_i = 0$ iar pe C_h din \mathbf{H}_t se poate deduce o condiție de frontieră de tip $\nu \frac{\partial A}{\partial n} \stackrel{\text{not}}{=} g$. Relația (3.20) devine

$$\begin{aligned} & \int_{\Omega} \frac{1}{\mu} \operatorname{grad} A \cdot \operatorname{grad} \Psi_i d\Omega - \int_{C_h} \Psi_i g ds = \\ & = \int_{\Omega} J \Psi_i d\Omega - \int_{\Omega} \frac{\mathbf{I}^T \cdot \operatorname{grad} \Psi_i}{\mu} d\Omega + \int_{\partial\Omega} \Psi_i \frac{\mathbf{I}^T \cdot \mathbf{n}}{\mu} ds. \end{aligned} \quad (3.21)$$

Dacă \mathbf{A} se aproximează cu

$$A = \sum_{j=1}^N \Psi_j A_j, \quad (3.22)$$

și descompunând domeniul Ω în elemente finite $\Omega = \cup \Omega_e$ atunci $\int_{\Omega} = \sum_e \int_e, \int_{C_h} = \sum_{e1} \int_{C_h^{e1}}$ (unde $e1$ sunt elemente ce au laturi pe C_h), relația (3.21) se scrie sub forma:

$$\begin{aligned} & \sum_e \int_e \sum_{j=1}^N A_j \frac{1}{\mu} \operatorname{grad} \Psi_j \cdot \operatorname{grad} \Psi_i d\Omega - \sum_{e1} \int_{C_h^{e1}} \Psi_i g ds = \\ & = \sum_e \int_e J \Psi_i d\Omega - \sum_e \int_e \frac{\mathbf{I}^T \cdot \operatorname{grad} \Psi_i}{\mu} d\Omega + \sum_{e1} \int_{C_h^{e1}} \Psi_i \frac{\mathbf{I}^T \cdot \mathbf{n}}{\mu} ds. \end{aligned} \quad (3.23)$$

Din această relație se observă că sistemul de rezolvat este de tipul

$$SA = T \quad (3.24)$$

și putem identifica contribuțiile fiecărui element și fiecărui segment de pe frontieră la matricea S și vectorul T . Astfel:

- *Contribuția fiecărui element e (de noduri i, j, k)*

Fiecare nod (fie nodul i) al elementului e contribuie cu:

$$s_{ii}^{(e)} = \int_e \frac{1}{\mu} (\text{grad } \Psi_i)^2 d\Omega, \quad (3.25)$$

$$s_{ij}^{(e)} = \int_e \frac{1}{\mu} \text{grad } \Psi_i \cdot \text{grad } \Psi_j d\Omega, \quad (3.26)$$

$$s_{ik}^{(e)} = \int_e \frac{1}{\mu} \text{grad } \Psi_i \cdot \text{grad } \Psi_k d\Omega, \quad (3.27)$$

$$t_i^{(e)} = \int_e J \Psi_i d\Omega - \int_e \frac{\mathbf{I}^T \cdot \text{grad } \Psi_i}{\mu} d\Omega. \quad (3.28)$$

- *Contribuția fiecărui segment de pe frontieră cu condiție Neumann (de noduri-capete i, j)*

Fiecare capăt (fie nodul i) contribuie cu

$$t_i^{(e1)} = \int_{C_h^{e1}} \Psi_i g ds + \int_{C_h^{e1}} \Psi_i \frac{\mathbf{I}^T \cdot \mathbf{n}}{\mu} ds. \quad (3.29)$$

În cazul elementelor triunghiulare de ordinul întâi (la care Ψ_i are o variație liniară pe element), expresiile se simplifică. Astfel

$$s_{ii}^{(e)} = \frac{1}{\mu} \frac{1}{4\Delta} (b_i b_j + c_i c_j) = \frac{1}{\mu} \Delta \alpha_{ij}, \quad (3.30)$$

unde $\alpha_{ij} = (b_i b_j + c_i c_j) / (4\Delta^2)$, Δ este aria elementului e , $b_i = y_j - y_k$, $b_j = y_k - y_i$, $c_i = x_k - x_j$, $c_j = x_i - x_k$, i, j, k fiind nodurile elementului ordonate trigonometric.

În ce privește $t_i^{(e)}$, presupunând J constant pe element și știind că

$$\text{grad } \Psi_i = \frac{b_i}{2\Delta} \mathbf{i} + \frac{c_i}{2\Delta} \mathbf{j}, \quad (3.31)$$

rezultă

$$t_i^{(e)} = \frac{J\Delta}{3} - \frac{1}{\mu} \left(\frac{I_y b_i}{2} - \frac{I_x c_i}{2} \right). \quad (3.32)$$

A mai ramas doar

$$t_i^{(e1)} = g \frac{l_{lat}}{2} + \frac{\mathbf{I}^T \cdot \mathbf{n}}{\mu} \frac{l_{lat}}{2}, \quad (3.33)$$

unde l_{lat} reprezintă lungimea laturii elementului $e1$ care se află pe frontiera C_h , $(\mathbf{I}^T \cdot \mathbf{n})$ este constant pe elementul $e1$, iar μ este permeabilitatea de calcul.

Deoarece μ este o permeabilitate de calcul fixată la început, matricea S rămâne aceeași pe parcursul iterațiilor. Ceea ce se schimbă la fiecare iterație este vectorul T , datorită polarizației \mathbf{I}^T . Din termenii care participă la formarea vectorului T , o parte nu depinde de \mathbf{I}^T . De aceea, scriem vectorul T ca fiind

$$T = T_1 + T_2, \quad (3.34)$$

unde T_1 este o matrice constantă pe parcursul iterațiilor și ca atare, ea va fi calculată o singură dată, ca și matricea S .

Condițiile de frontieră Dirichlet sunt impuse înainte de rezolvarea sistemului astfel generat. Ecuațiile corespunzătoare nodurilor Dirichlet se elimină, iar termenii corespunzători acestor noduri în celelalte ecuații sunt trecuți în membrul drept.

Algoritmul metodei polarizației

- regim magnetic staționar, 2D, XY, MEF, A, iterația prin B -

1. citește datele problemei:

- geometria;
- datele de material;
- sursele (\mathbf{J});
- condițiile de frontieră:
 - (a) noduri în care se dau valorile lui A ;
 - (b) segmente pe care se dă \mathbf{H}_t (notăm $\frac{\partial A}{\partial n} = g$, funcție ce se poate deduce din \mathbf{H}_t)

2. inițializează \mathbf{I}

pentru $e \in \{ \text{toate elementele} \}$

$$\mathbf{I}_e = 0$$

3. stabilește permeabilitatea de calcul

pentru $e \in \{ \text{toate elementele} \}$

dacă elementul este liniar **atunci** $\mu_e = \mu_{\text{liniar}}$

dacă elementul este neliniar **atunci** $\mu_e = 2/(\nu_{\text{max}} + \nu_{\text{min}})$ (contractie optimă)

4. assemblează S și T_1

pentru $e \in \{ \text{toate elementele} \}$ (e de noduri i, j, k)

pentru fiecare nod $\in e$ (fie nodul i)

$$\text{temp} = 4\Delta\mu$$

$$s_{ii}^{(e)} = (b_i^2 + c_i^2)/\text{temp}$$

$$s_{ij}^{(e)} = (b_i b_j + c_i c_j)/\text{temp}$$

$$s_{ik}^{(e)} = (b_i b_k + c_i c_k)/\text{temp}$$

$$t_{1i}^{(e)} = J\Delta/3$$

pentru toate segmentele pe care se impune \mathbf{H}_t (fie i, j capetele):

$$t_{1i} = t_{1i} + \frac{1}{\mu} g \frac{l_{iat}}{2}$$

$$t_{1j} = t_{1j} + \frac{1}{\mu} g \frac{I_{iat}}{2}$$

itnelin = 1; contor iterații neliniare

5.repetă

5.1. anulează elementele lui T_2

pentru $i \in \{ \text{toate nodurile} \}$

$$t_{2i} = 0$$

5.2. asamblează T_2

dacă itnelin $\neq 1$ **atunci**

pentru $e \in \{ \text{toate elementele} \}$ (e de noduri i, j, k)

pentru fiecare nod $\in e$ (fie nodul i)

$$t_{2i}^{(e)} = -\frac{1}{\mu} \left(\frac{I_y b_i}{2} - \frac{I_x c_i}{2} \right)$$

5.3. calculează câmpul pentru problema liniară

rezolvă $SA = T_1 + T_2$

Obs: aici se impun valorile lui A în nodurile "esențiale"

conform procedelor cunoscute (vezi de exemplu [77])

5.4. corectează polarizația

pentru $e \in \{ \text{toate elementele} \}$ (e de noduri i, j, k)

calculează \mathbf{B} :

$$\begin{aligned} \mathbf{B} &= \frac{\partial A}{\partial y} \mathbf{i} - \frac{\partial A}{\partial x} \mathbf{j} = \mathbf{i} \left(\sum_{j \in e} A_j \frac{\partial \Psi_j}{\partial y} \right) - \mathbf{j} \left(\sum_{j \in e} A_j \frac{\partial \Psi_j}{\partial x} \right) = \\ &= \mathbf{i} \left(\sum_{j \in e} A_j \frac{c_j}{2\Delta} \right) - \mathbf{j} \left(\sum_{j \in e} A_j \frac{b_j}{2\Delta} \right) \end{aligned}$$

calculează \mathbf{H} : $\mathbf{H} = \mathbf{H}(\mathbf{B})$

corectează \mathbf{I} : $\mathbf{I}_e = \mathbf{B} - \mu \mathbf{H}$

(Obs: pentru elementele liniare, $\mathbf{I} = 0$ întotdeauna)

5.5. itnelin = itnelin + 1

până când ($\|\delta I\| < \text{valoare impusă}$)

În algoritmul de mai sus $\|\delta I\|$ este norma L_2 a corecțiilor polarizațiilor. S-a demonstrat că procedeul este convergent în această normă [66]. În cazul metodei elementelor finite

$$\|\delta I\| = \sqrt{\sum_e \frac{1}{\mu} \Delta \|dI_e\|^2} = \sqrt{\sum_e \frac{1}{\mu} \Delta (dI_x^2 + dI_y^2)}. \quad (3.35)$$

Cât de mică trebuie să fie valoarea impusă? Testele numerice au arătat că ea depinde de problemă. Pentru a opri algoritmul este totuși mai folositoare o mărime relativă, de exemplu $\|\delta I\|/\|I\|$, unde $\|I\|$ este norma L_2 a polarizației, în cazul metodei elementelor finite

$$\|I\| = \sqrt{\sum_e \frac{1}{\mu} \Delta \|I_e\|^2} = \sqrt{\sum_e \frac{1}{\mu} \Delta (I_x^2 + I_y^2)}. \quad (3.36)$$

3.1.3 Metoda polarizației cu relaxare

Ideea relaxării aplicată metodei polarizației [67] poate fi explicată simplu folosind o funcție f reală de o singură variabilă reală. Presupunem că f este contracție (fie λ factorul de contracție) și notăm cu x^* punctul ei fix. Un algoritm iterativ fără relaxare calculează termenii succesiv astfel:

$$x_0, \quad x_1 = f(x_0), \quad x_2 = f(x_1). \quad (3.37)$$

Se poate arăta foarte ușor că

$$\|x_2 - x^*\| \leq \frac{\lambda}{1 - \lambda} \|x_2 - x_1\|. \quad (3.38)$$

Într-o procedură cu relaxare, succesiunea este următoarea:

$$x_0, \quad x_1 = f(x_0), \quad (3.39)$$

x_1 se relaxează

$$y_1 = x_0 + \omega(x_1 - x_0), \quad (3.40)$$

iar următoarea valoare este

$$y_2 = f(y_1). \quad (3.41)$$

Se poate arăta ușor că

$$\|y_2 - x^*\| \leq \frac{\lambda}{1 - \lambda} \|y_2 - y_1\|. \quad (3.42)$$

În aceste condiții y_2 este mai bun decât x_2 dacă

$$\begin{aligned} \|y_2 - y_1\| &\leq \|x_2 - x_1\|, \\ \Leftrightarrow \|f(y_1) - y_1\| &\leq \|f(x_1) - x_1\|, \end{aligned} \quad (3.43)$$

unde $y_1 = x_0 + \omega(x_1 - x_0)$.

Iată **algoritmul procedurii de relaxare pentru f** (o funcție reală de o variabilă reală):

1. inițializează x_0

2. $k = 1$

3. **repetă**

3.1. calculează $x_k = f(x_{k-1})$

3.2. **dacă** $\|f(x_k) - x_k\| < \varepsilon$ **stop**

3.3. $\Delta x = x_k - x_{k-1}$

3.4. minimizează $g(\omega) = \|f(x_{k-1} + \omega\Delta x) - (x_{k-1} + \omega\Delta x)\| \implies \omega$

3.5. $x_k = x_{k-1} + \omega\Delta x$

3.6. $x_{k+1} = f(x_k)$

3.7. $k = k + 1$

până când $(1 < 0)$; ciclu infinit

În cazul rezolvării problemei de câmp electromagnetic, acest algoritm se traduce ușor, înlocuind x cu \mathbf{I} și f cu $\hat{\mathbf{I}}\hat{\mathbf{B}}_a$ unde $\hat{\mathbf{B}}_a$ este operatorul care rezolvă problema liniară de câmp, iar $\hat{\mathbf{I}}$ este operatorul care realizează corecția potențialelor.

Iată **algoritmul metodei polarizației cu relaxare** pentru rezolvarea unei probleme de câmp:

1. inițializează \mathbf{I}_0

2. $k = 1$; k reprezintă contorul iterațiilor neliniare

3. repetă

3.1. $\mathbf{B}_k = \hat{\mathbf{B}}_a(\mathbf{I}_{k-1})$; calculează câmpul

$\mathbf{I}_k = \hat{\mathbf{I}}(\mathbf{B}_k)$; corectează polarizația

3.2. $\mathbf{B}_{k+1} = \hat{\mathbf{B}}_a(\mathbf{I}_k)$;

dacă $\|\hat{\mathbf{I}}(\mathbf{B}_{k+1}) - \mathbf{I}_k\|^2 < \varepsilon$ **stop**

3.3. $\Delta\mathbf{I} = \mathbf{I}_k - \mathbf{I}_{k-1}$

3.4. $\Delta\mathbf{B} = \mathbf{B}_{k+1} - \mathbf{B}_k$

rezolvă $g'(\omega) = 0 \implies \omega$

dacă $g(\omega) > g(1.) \implies \omega = 1.0$; punctul găsit este un maxim

3.5. $\mathbf{I}_k = \mathbf{I}_{k-1} + \omega\Delta\mathbf{I}$

3.6. $\mathbf{I}_{k+1} = \hat{\mathbf{I}}(\mathbf{B}_k + \omega\Delta\mathbf{B})$

3.7. $k = k + 2$

până când $(1 < 0)$; ciclu infinit

Expresia funcției de minimizat este

$$\begin{aligned} g(\omega) &= \|f(x_{k-1} + \omega\Delta x) - (x_{k-1} + \omega\Delta x)\|^2 = \\ &= \|\hat{\mathbf{I}}\hat{\mathbf{B}}_a(\mathbf{I}_{k-1} + \omega\Delta\mathbf{I}) - (\mathbf{I}_{k-1} + \omega\Delta\mathbf{I})\|^2 = \\ &= \|\hat{\mathbf{I}}(\mathbf{B}_k + \omega\Delta\mathbf{B}) - (\mathbf{I}_{k-1} + \omega\Delta\mathbf{I})\|^2, \end{aligned} \quad (3.44)$$

iar derivata ei este

$$\frac{1}{2}g'(\omega) = \left\langle \frac{\partial \hat{\mathbf{I}}}{\partial \mathbf{B}} \Big|_{\mathbf{B}_k + \omega\Delta\mathbf{B}} \cdot \Delta\mathbf{B} - \Delta\mathbf{I}, \hat{\mathbf{I}}(\mathbf{B}_k + \omega\Delta\mathbf{B}) - (\mathbf{I}_{k-1} + \omega\Delta\mathbf{I}) \right\rangle. \quad (3.45)$$

Observație: $\hat{\mathbf{B}}_a$ nu este un operator liniar ci unul afin. El se poate descompune însă ca suma dintre un operator corespunzător curenților și condițiilor de frontieră nenule (să îl notăm $\hat{\mathbf{B}}_J$) și un operator liniar ($\hat{\mathbf{B}}_{al}$) corespunzător polarizațiilor. De aceea

$$\begin{aligned} \hat{\mathbf{B}}_a(\mathbf{I}_{k-1} + \omega\Delta\mathbf{I}) &= \hat{\mathbf{B}}_J(0) + \hat{\mathbf{B}}_{al}(\mathbf{I}_{k-1} + \omega\Delta\mathbf{I}) = \\ &= \hat{\mathbf{B}}_J(0) + \hat{\mathbf{B}}_{al}(\mathbf{I}_{k-1}) + \omega\hat{\mathbf{B}}_{al}(\Delta\mathbf{I}) = \\ &= \hat{\mathbf{B}}_a(\mathbf{I}_{k-1}) + \omega(\hat{\mathbf{B}}_{al}(\mathbf{I}_k) + \hat{\mathbf{B}}_{al}(\mathbf{I}_{k-1}) + \hat{\mathbf{B}}_J(0) - \hat{\mathbf{B}}_J(0)) = \\ &= \hat{\mathbf{B}}_a(\mathbf{I}_{k-1}) + \omega(\mathbf{B}_{k+1} - \mathbf{B}_k) \end{aligned} \quad (3.46)$$

$$= \mathbf{B}_k + \omega\Delta\mathbf{B}, \quad (3.47)$$

expresie folosită la pasul 3.6 al algoritmului.

3.1.4 Metoda Newton

Să presupunem că materialul magnetic al problemei este caracterizat de o relație constitutivă neliniară $\mathbf{H} = \nu(B)\mathbf{B}$, unde ν este reluctivitatea statică. Am preferat să schimbăm notația pentru că condițiile pe care trebuie să le satisfacă funcția neliniară în metoda Newton sunt altele decât cele cerute funcției de material în metoda polarizației. S-a notat cu

$$w_m = \int_0^{\mathbf{B}} \mathbf{H}(\mathbf{b})d\mathbf{b} \quad (3.48)$$

densitatea de energie magnetică în fiecare punct.

Pentru prezentarea metodei vom folosi formularea în potențial magnetic vector prezentată în paragraful 3.1.1. În [65] se demonstrează următoarea teoremă:

Rezolvarea problemei liniare (cu ν fixat):

$$\text{rot}(\nu \text{rot } \mathbf{A}) = \mathbf{J} \quad (3.49)$$

cu condiția de etalonare

$$\text{div}(\sigma \mathbf{A}) = 0 \quad (3.50)$$

și condițiile de frontieră:

- (a) Pe S_h se dă $(\nu \text{rot } \mathbf{A})_t = \mathbf{f}$;
- (b) $\sigma \mathbf{A} \cdot \mathbf{n} = 0$ pe S_h ;
- (c) Pe S_b se dă $\mathbf{A}_t = \mathbf{g}$;
- (d) Dacă S_b este formată din n suprafețe disjuncte S_k , atunci pe $n - 1$ din ele se dau fluxurile lui $\sigma \mathbf{A}$, deci $\int_{S_k} \sigma \mathbf{A} \cdot \mathbf{n} dA = 0$, pentru $k = 1, \dots, n - 1$, este echivalentă cu minimizarea funcționalei:

$$\mathcal{F}(\mathbf{A}) = \frac{1}{2} \int_{\Omega} \nu \text{rot}^2 \mathbf{A} d\Omega - \int_{\Omega} \mathbf{J} \cdot \mathbf{A} d\Omega - \int_{S_h} (\mathbf{A} \times \mathbf{f}) \cdot \mathbf{n} dA \quad (3.51)$$

unde $\mathcal{F} : H \rightarrow \mathbb{R}$, H fiind mulțimea funcțiilor \mathbf{A} care îndeplinesc condiția de etalonare și condițiile de frontieră (b), (c) și (d).

Observații:

1. În teorema de mai sus am presupus că în domeniul de studiu nu există pânze de curent, presupunere care o vom menține și în cele ce urmează.
2. Condiția de frontieră (a) trebuie să fie compatibilă cu ecuația (3.2), adică $\int_{\partial S} \mathbf{f} d\mathbf{r} = \int_S \mathbf{J} \mathbf{n} dA$ pentru $(\forall) S \subset \partial\Omega$ cu $\partial S \subset S_h$.
3. Condițiile (b) și (d) nu au semnificație fizică, dar sunt necesare pentru a asigura unicitatea potențialului vector \mathbf{A} .

Ar fi de dorit să găsim și în cazul în care materialul nu este liniar magnetic o funcțională de tipul (3.51). În (3.51) se observă imediat că prima integrală reprezintă energia magnetică înmagazinată în Ω . Deci (3.51) poate fi scrisă și sub forma

$$\mathcal{F}(\mathbf{A}) = \int_{\Omega} w_m(\mathbf{A}) \, d\Omega - \int_{\Omega} \mathbf{J} \cdot \mathbf{A} \, d\Omega - \int_{S_h} (\mathbf{A} \times \mathbf{f}) \cdot \mathbf{n} \, dA, \quad (3.52)$$

unde $\mathcal{F} : H \rightarrow \mathbb{R}$, H fiind mulțimea funcțiilor \mathbf{A} care îndeplinesc condiția de etalonare și condițiile de frontieră (b), (c) și (d) iar $\mathbf{f} = \mathbf{H}_t$ în punctele de pe S_h .

Relația (3.52) poate fi privită ca fiind valabilă pentru materiale neliniare magnetic dacă pentru densitatea de energie se consideră expresia (3.48). Se poate arăta (conform [144]) că, deoarece primul termen în (3.52) reprezintă o energie, condițiile pentru existența și unicitatea soluției sunt: funcțiile caracteristice de magnetizare $B(H)$ ale materialelor din problemă să fie monoton crescătoare, în timp ce prima derivată $\mu_d(H) = dB/dH$ să fie monoton descrescătoare.

Cazul problemelor plan-paralele.

În cele ce urmează vom urmări minimizarea funcționalei (3.52) în cazul problemelor plan-paralele. Planul problemei se consideră a fi planul XY , densitatea de curent \mathbf{J} este orientată în direcția axei Z (de versor \mathbf{k}), potențialul magnetic vector \mathbf{A} este presupus de asemenea orientat în direcția axei Z (asigurându-se în acest fel satisfacerea condiției de etalonare $\operatorname{div} \mathbf{A} = 0$).

Vom nota

$$\mathbf{A} = A(x, y)\mathbf{k}, \quad \mathbf{J} = J(x, y)\mathbf{k}. \quad (3.53)$$

Minimizarea funcționalei (3.52) pentru cazul neliniar urmărește aceeași tehnică ca în cazul problemelor liniare.

Presupunem că domeniul problemei este discretizat într-o multime de elemente finite și ne vom concentra atenția asupra unui element.

Potențialul A este aproximat prin

$$A(x, y) = \sum_{i \in e} A_i \Psi_i(x, y), \quad (3.54)$$

unde (x, y) este un punct în interiorul elementului e , iar indicele i parcurge nodurile elementului e . A_i reprezintă valoarea lui A în nodul i iar $\Psi_i(x, y)$ reprezintă funcția de formă corespunzătoare nodului i .

Cu această substituție, $\mathcal{F}(A)$ devine o funcție de un număr finit de variabile, iar procesul de minimizare este reprezentat de

$$\frac{\partial \mathcal{F}}{\partial A_i} = 0, \quad (3.55)$$

cu i parcurgând nodurile. Relația (3.55) scrisă dezvoltat devine

$$\frac{\partial}{\partial A_i} \left(\int_{\Omega} w_m \, d\Omega \right) - \frac{\partial}{\partial A_i} \left(\int_{\Omega} JA \, d\Omega \right) - \frac{\partial}{\partial A_i} \left(\int_{C_h} (\mathbf{A} \times \mathbf{f}) \cdot \mathbf{n} \, ds \right) = 0. \quad (3.56)$$

Iată contribuțiile fiecărui element e la termenii ecuației (3.56):

- *Contribuția elementului e la primul termen din (3.56)*

$$\frac{\partial}{\partial A_i} \left(\int_e w_m \, d\Omega \right) = \int_e \frac{\partial w_m}{\partial A_i} \, d\Omega. \quad (3.57)$$

Dar

$$\frac{\partial w_m}{\partial A_i} = \frac{\partial}{\partial A_i} \left(\int_0^B \nu(b)b \, db \right). \quad (3.58)$$

În cele ce urmează este mai convenabil să se considere reluctivitatea ca o funcție de pătratul inducției magnetice. De aceea facem schimbarea de variabilă

$$b = \sqrt{b_1} \quad \Rightarrow \quad db = \frac{1}{2\sqrt{b_1}} db_1; \quad (3.59)$$

(3.58) devine

$$\begin{aligned} \frac{\partial w_m}{\partial A_i} &= \frac{\partial}{\partial A_i} \left(\int_0^{B^2} \nu(\sqrt{b_1}) \sqrt{b_1} \frac{1}{2\sqrt{b_1}} db_1 \right) = \\ &= \frac{1}{2} \frac{\partial}{\partial A_i} \left(\int_0^{B^2} \nu_1(b^2) \, d(b^2) \right), \end{aligned} \quad (3.60)$$

unde am notat $\nu(\sqrt{b_1}) = \nu_1(b_1)$ (deci $\nu_1(b^2) = \nu(b)$). Cu această notație (3.60) se mai poate scrie

$$\begin{aligned} \frac{\partial w_m}{\partial A_i} &= \frac{1}{2} \frac{\partial}{\partial B^2} \left(\int_0^{B^2} \nu_1(b^2) \, d(b^2) \right) \frac{\partial B^2}{\partial A_i} = \\ &= \frac{1}{2} \nu_1(B^2) \frac{\partial B^2}{\partial A_i}, \end{aligned} \quad (3.61)$$

deoarece $\frac{\partial}{\partial x} \int_0^x f(y) \, dy = f(x)$. Reluând (3.57)

$$\frac{\partial}{\partial A_i} \left(\int_e w_m \, d\Omega \right) = \int_e \frac{1}{2} \nu_1(B^2) \frac{\partial B^2}{\partial A_i} \, d\Omega. \quad (3.62)$$

Dar

$$\mathbf{B} = \text{rot } \mathbf{A} = \frac{\partial A}{\partial y} \mathbf{i} - \frac{\partial A}{\partial x} \mathbf{j}, \quad (3.63)$$

de unde

$$\begin{aligned}
B^2 &= \left(\frac{\partial A}{\partial y} \right)^2 + \left(\frac{\partial A}{\partial x} \right)^2 = \\
&= \left(\frac{\partial}{\partial y} \sum_{j \in e} A_j \Psi_j(x, y) \right)^2 + \left(\frac{\partial}{\partial x} \sum_{j \in e} A_j \Psi_j(x, y) \right)^2 = \\
&= \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial y} \right)^2 + \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial x} \right)^2. \tag{3.64}
\end{aligned}$$

Rezultă că, dacă i este unul din vârfurile elementului e , atunci

$$\begin{aligned}
\frac{\partial B^2}{\partial A_i} &= 2 \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial y} \right) \frac{\partial}{\partial A_i} \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial y} \right) + \\
&+ 2 \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial x} \right) \frac{\partial}{\partial A_i} \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial x} \right) = \\
&= 2 \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial y} \right) \frac{\partial \Psi_i(x, y)}{\partial y} + 2 \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial x} \right) \frac{\partial \Psi_i(x, y)}{\partial x}.
\end{aligned}$$

Reluând (3.62)

$$\begin{aligned}
&\frac{\partial}{\partial A_i} \left(\int_e w_m \, d\Omega \right) = \\
&= \int_e \nu_1(B^2) \left[\frac{\partial \Psi_i(x, y)}{\partial y} \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial y} \right) + \frac{\partial \Psi_i(x, y)}{\partial x} \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial x} \right) \right] d\Omega = \\
&= \int_e \nu_1(B^2) \sum_{j \in e} \left[A_j \left(\frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} + \frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} \right) \right] d\Omega = \\
&= \sum_{j \in e} A_j \int_e \nu_1(B^2) \left(\frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} + \frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} \right) d\Omega = \\
&\stackrel{def}{=} \sum_{j \in e} A_j s_{ij}^{(e)}. \tag{3.65}
\end{aligned}$$

Observație: dacă j nu este unul din vârfurile elementului e atunci $s_{ij}^{(e)} = 0$.

- Contribuția elementului e la al doilea termen din (3.56)

$$\begin{aligned}
\frac{\partial}{\partial A_i} \left(\int_e J A \, d\Omega \right) &= \int_e \frac{\partial}{\partial A_i} \left(J \sum_{j \in e} A_j \Psi_j(x, y) \right) d\Omega = \\
&= \int_e J \Psi_i(x, y) \, d\Omega \stackrel{def}{=} t_i^{(e)}. \tag{3.66}
\end{aligned}$$

Observație: dacă i nu este unul din vârfurile elementului e atunci $t_i^{(e)} = 0$.

- *Contribuția elementului e la al treilea termen din (3.56)*

Fie e un element care are o latură pe C_h . Atunci, contribuția lui este:

$$\begin{aligned} \frac{\partial}{\partial A_i} \left(\int_{C_h^e} (\mathbf{A} \times \mathbf{f}) \cdot \mathbf{n} \, ds \right) &= \int_{C_h^e} \frac{\partial}{\partial A_i} \left(\sum_{j \in e} A_j \Psi_j(x, y) (\mathbf{k} \times \mathbf{f}) \cdot \mathbf{n} \right) \, ds = \\ &= \int_{C_h^e} \Psi_i(x, y) (\mathbf{k} \times \mathbf{f}) \cdot \mathbf{n} \, ds = \\ &\stackrel{def}{=} t_i^{(e)}. \end{aligned} \quad (3.67)$$

În final, sistemul de rezolvat este de tipul

$$SA = T, \quad (3.68)$$

unde la matricea S contribuie toate elementele cu

$$s_{ij}^{(e)} = \int_e \nu_1(B^2) \left(\frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} + \frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} \right) \, d\Omega, \quad (3.69)$$

iar la vectorul T contribuie elementele în care $J \neq 0$ cu

$$t_i^{(e)} = \int_e J \Psi_i(x, y) \, d\Omega \quad (3.70)$$

și elementele care au laturi pe frontiera pe care se dă \mathbf{H}_t cu

$$t_i^{(e)} = \int_{C_h^e} \Psi_i(x, y) (\mathbf{k} \times \mathbf{f}) \cdot \mathbf{n} \, ds. \quad (3.71)$$

Metoda Newton. Algoritmul pentru probleme plan-paralele.

Considerăm funcționala de energie

$$\mathcal{F}(\mathbf{A}) = \int_{\Omega} w_m(\mathbf{A}) \, d\Omega - \int_{\Omega} \mathbf{J} \cdot \mathbf{A} \, d\Omega - \int_{S_h} (\mathbf{A} \times \mathbf{f}) \cdot \mathbf{n} \, dA, \quad (3.72)$$

unde $\mathcal{F} : H \rightarrow \mathbb{R}$, H fiind mulțimea funcțiilor \mathbf{A} care îndeplinesc condiția de etalonare și condițiile de frontieră (b), (c) și (d) iar $\mathbf{f} = \mathbf{H}_t$ în punctele de pe S_h .

În cazul problemelor plan-paralele, funcționala (3.72) devine

$$\mathcal{F}(A) = \int_{\Omega} w_m(A) \, d\Omega - \int_{\Omega} JA \, d\Omega - \int_{C_h} A (\mathbf{k} \times \mathbf{f}) \cdot \mathbf{n} \, ds. \quad (3.73)$$

Fie A^* soluția exactă a problemei și

$$A = A^* - \delta A \quad (3.74)$$

o estimare aproximativă a potențialului, apropiată de A^* .

În metoda elementelor finite $A = \sum_i A_i \Psi_i(x, y)$, deci \mathcal{F} poate fi privită ca o funcție multidimensională $\mathcal{F}(A_1, \dots, A_N)$.

Dezvoltarea în serie Taylor a unei funcții $f(x_1, x_2, \dots, x_N)$ în jurul punctului $(x_{01}, x_{02}, \dots, x_{0N})$ este

$$\begin{aligned} f(x_1, x_2, \dots, x_N) &= f(x_{01}, x_{02}, \dots, x_{0N}) + \\ &+ \frac{\partial f}{\partial x_1}(x_{01}, x_{02}, \dots, x_{0N})(x_1 - x_{01}) + \\ &+ \dots + \\ &+ \frac{\partial f}{\partial x_N}(x_{01}, x_{02}, \dots, x_{0N})(x_N - x_{0N}) + \\ &+ \dots \end{aligned} \quad (3.75)$$

Pentru $f = \frac{\partial \mathcal{F}}{\partial A_i}$ obținem

$$\frac{\partial \mathcal{F}}{\partial A_i}(A^*) = \frac{\partial \mathcal{F}}{\partial A_i}(A) + \sum_j \frac{\partial^2(\mathcal{F})}{\partial A_i \partial A_j}(A) \underbrace{(A^* - A)_j}_{(\delta A)_j} + \dots \quad (3.76)$$

Dar

$$\frac{\partial \mathcal{F}}{\partial A_i}(A^*) = 0 \quad (3.77)$$

și, neglijând termenii de ordin superior din (3.76), rezultă

$$\delta A = -P^{-1}V, \quad (3.78)$$

unde

$$(\delta A)_j = (A^* - A)_j, \quad (3.79)$$

$$p_{ij} = \frac{\partial^2(\mathcal{F})}{\partial A_i \partial A_j}(A), \quad (3.80)$$

$$V_j = \frac{\partial \mathcal{F}}{\partial A_j}(A). \quad (3.81)$$

Metoda iterativă pornește de la niște valori inițiale pentru A , apoi calculul abaterii față de soluția exactă cu formula (3.78), abatere care este apoi adăugată estimării inițiale a potențialului. Deoarece derivatele de ordin mai mare sau egal cu 3 au fost neglijate în (3.76) rezultă că (3.78) nu dă abaterea exactă ci una aproximativă. Această abatere ar fi exactă dacă $\mathcal{F}(A)$ ar fi o funcțională pătratică; acesta este cazul problemelor liniare în care procedura Newton converge într-un singur pas. În cazul problemelor neliniare, deplasarea calculată cu (3.78) se adaugă vechilor valori pentru a obține o estimare mai bună și întregul proces se repetă.

Implementarea acestei metode necesită evaluarea derivatelor (3.80) și (3.81). Când am discutat despre minimizarea funcționalei în cazul problemelor plan-paralele (paragraful

3.1.4, pagina 56), am văzut tipurile de termeni care apar prin derivarea $\frac{\partial \mathcal{F}}{\partial A_j}$. Rămâne de analizat (3.80). Din (3.73) rezultă

$$\frac{\partial^2(\mathcal{F})}{\partial A_i \partial A_j} = \int_{\Omega} \frac{\partial^2(w_m)}{\partial A_i \partial A_j} d\Omega. \quad (3.82)$$

Reamintind (3.61)

$$\frac{\partial w_m}{\partial A_i} = \frac{1}{2} \nu_1(B^2) \frac{\partial B^2}{\partial A_i},$$

rezultă că

$$\frac{\partial^2(w_m)}{\partial A_i \partial A_j} = \frac{\nu_1(B^2)}{2} \frac{\partial^2(B^2)}{\partial A_i \partial A_j} + \frac{1}{2} \frac{\partial \nu_1(B^2)}{\partial B^2} \frac{\partial B^2}{\partial A_j} \frac{\partial B^2}{\partial A_i}.$$

- Contribuția elementului e la integrala (3.82)

$$p_{ij}^{(e)} = \int_e \frac{\nu_1(B^2)}{2} \frac{\partial^2(B^2)}{\partial A_i \partial A_j} d\Omega + \int_e \frac{1}{2} \frac{\partial \nu_1(B^2)}{\partial B^2} \frac{\partial B^2}{\partial A_j} \frac{\partial B^2}{\partial A_i} d\Omega. \quad (3.83)$$

Dar (formula (3.64))

$$B^2 = \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial y} \right)^2 + \left(\sum_{j \in e} A_j \frac{\partial \Psi_j(x, y)}{\partial x} \right)^2.$$

Deci elementul e contribuie la p_{ij} doar dacă nodurile i și j aparțin elementului e . În cazul utilizării *elementelor triunghiulare de ordinul întâi*, calculul se simplifică. Astfel, considerând i, j, k nodurile elementului e , rezultă

$$\begin{aligned} \frac{\partial^2(B^2)}{\partial A_i \partial A_j} &= 2 \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} \right), \\ \frac{\partial B^2}{\partial A_i} &= 2A_i \left[\left(\frac{\partial \Psi_i(x, y)}{\partial x} \right)^2 + \left(\frac{\partial \Psi_i(x, y)}{\partial y} \right)^2 \right] + \\ &+ 2A_j \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} \right) + \\ &+ 2A_k \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_k(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_k(x, y)}{\partial y} \right), \\ \frac{\partial B^2}{\partial A_j} &= 2A_j \left[\left(\frac{\partial \Psi_j(x, y)}{\partial x} \right)^2 + \left(\frac{\partial \Psi_j(x, y)}{\partial y} \right)^2 \right] + \\ &+ 2A_i \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} \right) + \\ &+ 2A_k \left(\frac{\partial \Psi_j(x, y)}{\partial x} \frac{\partial \Psi_k(x, y)}{\partial x} + \frac{\partial \Psi_j(x, y)}{\partial y} \frac{\partial \Psi_k(x, y)}{\partial y} \right). \end{aligned} \quad (3.84)$$

Reluând (3.83) rezultă

$$\begin{aligned}
p_{ij}^{(e)} &= \int_e \nu_1(B^2) \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} \right) d\Omega + \\
&+ \int_e 2 \frac{\partial \nu_1(B^2)}{\partial B^2} [A_i \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_i(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_i(x, y)}{\partial y} \right) + \\
&+ A_j \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} \right) + \\
&+ A_k \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_k(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_k(x, y)}{\partial y} \right)] \cdot \\
&\cdot [A_i \left(\frac{\partial \Psi_i(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} + \frac{\partial \Psi_i(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} \right) + \\
&+ A_j \left(\frac{\partial \Psi_j(x, y)}{\partial x} \frac{\partial \Psi_j(x, y)}{\partial x} + \frac{\partial \Psi_j(x, y)}{\partial y} \frac{\partial \Psi_j(x, y)}{\partial y} \right) + \\
&+ A_k \left(\frac{\partial \Psi_j(x, y)}{\partial x} \frac{\partial \Psi_k(x, y)}{\partial x} + \frac{\partial \Psi_j(x, y)}{\partial y} \frac{\partial \Psi_k(x, y)}{\partial y} \right)] d\Omega.
\end{aligned}$$

Pentru a simplifica scrierea, notăm

$$\alpha_{ab} = \frac{\partial \Psi_a}{\partial x} \frac{\partial \Psi_b}{\partial x} + \frac{\partial \Psi_a}{\partial y} \frac{\partial \Psi_b}{\partial y}, \quad (3.85)$$

unde $a, b \in \{i, j, k\}$. Deci

$$\begin{aligned}
p_{ij}^{(e)} &= \int_e \nu_1(B^2) \alpha_{ij} d\Omega + \\
&+ 2 \int_e \frac{\partial \nu_1(B^2)}{\partial B^2} [A_i \alpha_{ii} + A_j \alpha_{ij} + A_k \alpha_{ik}] [A_j \alpha_{jj} + A_i \alpha_{ij} + A_k \alpha_{jk}] d\Omega. \quad (3.86)
\end{aligned}$$

Dar, pentru elementele triunghiulare de ordinul întâi, B , ν , deci și ν_1 sunt constante în interiorul unui element, precum și derivatele $\frac{\partial \Psi_i(x, y)}{\partial x}$, $\frac{\partial \Psi_i(x, y)}{\partial y}$. Integralele din (3.86) sunt de aceea ușor de evaluat, integranzii fiind mărimi constante. Deci

$$\begin{aligned}
p_{ij}^{(e)} &= \nu_1(B^2) \alpha_{ij} \Delta + \\
&+ 2 \frac{\partial \nu_1(B^2)}{\partial B^2} \Delta [A_i \alpha_{ii} + A_j \alpha_{ij} + A_k \alpha_{ik}] [A_j \alpha_{jj} + A_i \alpha_{ij} + A_k \alpha_{jk}], \quad (3.87)
\end{aligned}$$

unde Δ este aria elementului e . Se demonstrează că pentru elementul triunghiular de ordinul întâi $\alpha_{ij} = \frac{b_i b_j + c_i c_j}{4\Delta^2}$ unde $b_i = y_j - y_k$, $b_j = y_k - y_i$, $c_i = x_k - x_j$, $c_j = x_i - x_k$, i, j, k fiind nodurile elementului, ordonate în sens trigonometric.

În cele ce urmează vom detalia **algoritmul Newton**:

1. inițializare potențiale

A = valori inițiale

2. repetă

- 2.1. calculează termenul liber V
- 2.2. calculează matricea Jacobian P cu elementele date de relația (3.87)
- 2.3. rezolvă sistemul linear $P\delta A = -V$ și determină corecția δA
- 2.4. corectează A : $A = A + \delta A$

până când (δA este neglijabil)

Calculul lui V (etapa 2.1.) presupune:

- 2.1.1. asamblarea matricei S conform relației (3.65), unde

$$s_{ij}^{(e)} = \int_e \nu_1(B^2)\alpha_{ij} d\Omega = \nu_1(B^2)\alpha_{ij}\Delta$$

pentru elemente triunghiulare de ordinul întâi;

- 2.1.2. asamblarea vectorului T conform relației (3.66), unde

$$t_i^{(e)} = \int_e J\Psi_i(x, y) d\Omega = \frac{J\Delta}{3}$$

pentru elemente triunghiulare de ordinul întâi (în care J se consideră constant în interiorul unui element);

- 2.1.3. asamblarea vectorului T' conform relației (3.67), unde

$$t_i'^{(e)} = \int_{C_h^e} \Psi_i(x, y)(\mathbf{k} \times \mathbf{f}) \cdot \mathbf{n} ds = (\mathbf{k} \times \mathbf{f}) \cdot \mathbf{n} \frac{l_{\text{segm}}}{2}$$

pentru elemente triunghiulare de ordinul întâi (l_{segm} fiind lungimea laturii elementului e care este situată pe frontiera C_h , frontiera pe care se dă $\mathbf{H}_t = \mathbf{f}$);

- 2.1.4 $V = SA - T - T'$.

Observație: În cazul problemei neliniare, condiția de frontieră de tip natural este cea care dă valoarea lui \mathbf{H}_t pe o parte a frontierei (S_h), condițiile esențiale care trebuie impuse sistemului fiind (vezi și pagina 44):

- Pe S_h : $A_n = 0$, condiție îndeplinită automat în cazul problemelor plan-paralele în care $\mathbf{A} = \mathbf{A}\mathbf{k}$;
- Pe S_b : se dă \mathbf{A}_t , condiție care în cazul problemelor plan-paralele semnifică impunerea valorii scalare A în noduri;
- Dacă S_b este formată din n suprafețe disjuncte, pe $n - 1$ se dau fluxurile $\int_{S_k} A_n dA = 0, k = 1, \dots, n - 1$, condiție îndeplinită automat în cazul problemelor plan-paralele.

Algoritm Newton - regim magnetic staționar, 2D, XY, MEF:

1. citește datele problemei:

- geometria;
- datele de material;

- sursele (\mathbf{J});
- condițiile de frontieră:
 - (a) noduri în care se dau valorile lui A ;
 - (b) segmente pe care se dă \mathbf{H}_t (notăm $(\mathbf{k} \times \mathbf{H}_t)\mathbf{n} = g$).

2. inițializări

pentru $i \in \{ \text{toate nodurile} \}$

$$A_i = 0$$

pentru $i \in \{ \text{toate nodurile pe care se impune } A \}$

$$A_i = \text{valoarea impusă}$$

3. procedura Newton

repetă

3.1. calculează V și Jacobianul P

pentru $e \in \{ \text{toate elementele} \}$ (e de noduri i, j, k)

pentru fiecare nod $\in e$ (fie nodul i)

$$h = b_i^2 + c_i^2 ; h = 4\alpha_{ii}\Delta^2$$

$$h_{ij} = b_i b_j + c_i c_j ; h_{ij} = 4\alpha_{ij}4\Delta^2$$

$$h_{ik} = b_i b_k + c_i c_k ; h_{ik} = 4\alpha_{ik}4\Delta^2$$

$$\text{temp3} = \frac{d\nu(B)}{dB^2} / (8\Delta^3)$$

asamblează S și începe asamblarea lui P

$$s_{ii}^{(e)} = p_{ii}^{(e)} = h\nu / (4\Delta)$$

$$s_{ij}^{(e)} = p_{ij}^{(e)} = h_{ij}\nu / (4\Delta)$$

$$s_{ik}^{(e)} = p_{ik}^{(e)} = h_{ik}\nu / (4\Delta)$$

asamblează T

$$t_i^{(e)} = J\Delta / 3$$

dacă e este neliniar ($\text{temp3} \neq 0$), continuă asamblarea lui P :

$$\text{temp}_i = A_i h + A_j h_{ij} + A_k h_{ik}$$

$$h_{jj} = b_j^2 + c_j^2$$

$$h_{jk} = b_j b_k + c_j c_k$$

$$h_{kk} = b_k^2 + c_k^2$$

$$\text{temp}_j = A_j h_{jj} + A_i h_{ij} + A_k h_{jk}$$

$$\text{temp}_k = A_k h_{kk} + A_i h_{ik} + A_j h_{jk}$$

$$p_{ii}^{(e)} = p_{ii}^{(e)} + \text{temp3} \cdot \text{temp}_i^2$$

$$p_{ij}^{(e)} = p_{ij}^{(e)} + \text{temp3} \cdot \text{temp}_i \cdot \text{temp}_j$$

$$p_{ik}^{(e)} = p_{ik}^{(e)} + \text{temp3} \cdot \text{temp}_i \cdot \text{temp}_k$$

pentru toate segmentele pe care se impune \mathbf{H}_t (fie i, j capetele):

asamblează T'

$$t_i^{(segm)} = gl_{segm} / 2$$

$$t_j^{(segm)} = gl_{segm} / 2$$

$$V = SA - T - T'$$

3.2. rezolvă $P\delta A = -V$

3.3. **pentru** $i \in \{\text{toate nodurile în afara de cele pe care se impune } A\}$

$$A_i = A_i + (\delta A)_i$$

până când ($\|\delta A\| < \text{valoare impusă}$)

Majoritatea timpului consumat pe iterație se datorează rezolvării sistemului liniar și evaluării derivatei $\frac{d\nu(B)}{dB^2}$.

3.1.5 Comparații

În acest paragraf am prezentat mai detaliat două metode de rezolvare a problemelor de câmp magnetic staționar în medii neliniare magnetice: metoda Newton și metoda polarizației (cu și fără relaxare). Metoda Newton are avantajul unei convergențe rapide (pătratică) în apropierea soluției dar are dezavantajul că nu este întotdeauna convergentă. Metoda polarizației are avantajul că este întotdeauna convergentă, dar are dezavantajul unei convergențe liniare.

Poate că un algoritm înțelept ar trebui să combine avantajele celor două metode și anume să înceapă prin iterații ale metodei polarizației după care la un moment dat ar trece pe iterații Newton-Raphson. În alegerea acestui moment de trecere ar trebui să se țină cont și de alți indicatori și/sau estimatori de eroare.

O problemă importantă o constituie și rezolvarea sistemului liniar. În cazul metodei polarizației, matricea coeficienților sistemului este întotdeauna aceeași, de aceea în acest caz se poate folosi o metodă directă de rezolvare a sistemului (factorizare LU - varianta Cholesky pentru că matricea sistemului este simetrică și pozitiv definită). În cazul metodei Newton-Raphson se pot aborda două strategii. Una din ele este să se rezolve sistemul liniar printr-o metodă directă, caz în care se recomandă ca matricea Jacobian să nu se recalculeze la fiecare iterație având în vedere că ea nu se schimbă prea mult de la o iterație la alta. Matricea Jacobian se calculează pentru primele L iterații Newton-Raphson apoi se menține neschimbată M iterații, se recalculează la iterația $L + M + 1$ și apoi se menține neschimbată ș.a.m.d. Având în vedere că pe parcursul a M iterații matricea coeficienților sistemului de rezolvat este neschimbată, se poate folosi și aici o factorizare LU. Un avantaj suplimentar îl putem obține în cazul unei numerotări convenabile a nodurilor și anume se începe cu nodurile de care sunt agățate doar elemente liniare și apoi se continuă cu restul nodurilor. În cazul în care astfel de noduri sunt majoritare algoritmul pentru factorizarea LU a matricei coeficienților corespunzătoare unui grup de M iterații poate ține cont de factorizarea grupului anterior în vederea micșorării timpului de calcul. O altă posibilitate este să se folosească o metodă iterativă pentru rezolvarea sistemului liniar, avantajul

principal al acestei soluții constând în posibilitatea aplicării tehnicilor eficiente de matricare rare.

În ce privește curbele de magnetizare, ele trebuie să fie monotone pentru a putea fi folosite de metoda polarizației și necesită o prelucrare suplimentară (interpolare) pentru a putea fi folosite de metoda Newton-Raphson.

3.1.6 Condiții de frontieră pentru domenii nemărginite

Metoda elementelor finite asociată unui model matematic diferențial necesită ca domeniul de calcul al problemei să fie mărginit. În problemele în care domeniul este nemărginit, modelul numeric are o frontieră artificială plasată suficient de departe astfel încât pe ea câmpul să se anuleze și să poată fi presupuse adevărate condiții nule de frontieră de tip Dirichlet sau Neumann. În consecință problema cuprinde o regiune mare de aer care trebuie discretizată. O alternativă la această abordare ar fi găsirea unei condiții asociate unei frontiere "deschise"¹ (care nu trebuie să fie dusă cât mai departe de sursele de câmp), dar care să simuleze infinitatea mediului înconjurător. Acest lucru se realizează folosind o formulare integrală pentru domeniul exterior frontierei. Discretizarea acestei formulări integrale se face cu ajutorul metodei elementelor de frontieră. Elementele de frontieră apar în mod natural prin discretizarea domeniului interior în elemente finite.

Există mai multe posibilități de a trata formulările integrale, deci metoda elementelor de frontieră atunci când sunt cuplate cu formulări diferențiale, deci cu metoda elementelor finite. Un articol de referință despre acest subiect este [134] în care se clasifică abordările în: metoda surselor de simplu strat, metoda identității Green și metoda Treftz. Pentru primele două metode, pentru obținerea sistemului discretizat, se poate folosi fie metoda colocației, fie o metodă variațională.

În acest paragraf vom arăta cum se tratează o frontieră deschisă în cazul unei probleme plan paralele de regim magnetic staționar folosind metoda identității Green. Înainte de aceasta vom prezenta două teoreme de unicitate pentru regimul magnetic staționar (cazul 3D și cazul 2D) în care frontiera domeniului este caracterizată de o condiție mai specială, care apare din considerarea mediului exterior.

- **Formularea problemei cu frontieră deschisă (cazul 3D)**

Domeniul problemei va fi împărțit în două părți disjuncte. Un domeniu interior \mathcal{D}_i , simplu conex, în care se găsesc materiale neliniare și surse de câmp, mărginit de o suprafață închisă Σ și un domeniu exterior suprafeței Σ , notat \mathcal{D}_e , care se extinde până la infinit. Normala exterioară la domeniul interior o notăm cu \mathbf{n} , iar cea interioară este $\mathbf{n}' = -\mathbf{n}$. Suprafața închisă Σ va fi aleasă astfel încât să treacă numai prin aer iar domeniul exterior (care are $\mu = \mu_0$) să nu conțină surse de câmp (figura 3.1).

¹"open boundary"

În aceste condiții în domeniul exterior (care este simplu conex) și cel puțin pe o mică "coașă" lipită de Σ din domeniul interior putem folosi potențialul magnetic scalar $V(P)$ definit astfel încât $\mathbf{H} = -\text{grad } V$. Acesta satisface în domeniul exterior ecuația lui Laplace $\Delta V = 0$ și tinde către zero când $P \rightarrow \infty$ mai repede decât $1/R_P$. Pe frontiera Σ presupusă suficient de netedă potențialul V va fi continuu la fel ca și derivata sa după normală.

Folosind identitatea Green se poate arăta ușor că

$$V(P) = \oint_{\Sigma} V(Q) \frac{\partial G(P, Q)}{\partial n_Q} dA_Q, \quad (3.88)$$

unde P este un punct în domeniul exterior iar G este funcția Green care satisface $-\Delta G(P, Q) = \delta(P, Q)$, $G|_{\Sigma} = 0$. Derivând relația (3.88) în punctele de pe frontiera Σ rezultă

$$\left(\frac{\partial V}{\partial n} \right) (P) = \oint_{\Sigma} V(Q) \frac{\partial^2 G(P, Q)}{\partial n_P \partial n_Q} dA_Q. \quad (3.89)$$

Relația (3.89) definește un operator integral 3D

care exprimă o relație între valorile derivatei după normală ale potențialului pe frontieră și valorile lui în toate punctele frontierei. Vom nota acest operator cu \mathcal{P} . El se numește operatorul Poincaré-Steklov, este un operator liniar și se demonstrează că este simetric și negativ definit [74].

TEOREMĂ DE UNICITATE (3D): Problema determinării câmpului magnetic staționar (\mathbf{B}, \mathbf{H}) în domeniul interior \mathcal{D}_i , care satisface în \mathcal{D}_i ecuațiile

$$\text{rot } \mathbf{H} = \mathbf{J} \quad , \quad \text{div } \mathbf{B} = 0 \quad , \quad \mathbf{B} = F(\mathbf{H}), \quad (3.90)$$

unde F este o funcție uniform monotonă, admite soluție unică dacă pe frontiera Σ se impune o relație de tipul

$$\left. \frac{\partial V}{\partial n} \right|_{\Sigma} = \mathcal{P}(V_{\Sigma}), \quad (3.91)$$

unde \mathcal{P} este un operator liniar negativ.

Negativitatea operatorului \mathcal{P} se scrie explicit $\oint_{\Sigma} V \frac{\partial V}{\partial n} dA \leq 0$. Pentru demonstrație se presupune prin reducere la absurd că există două perechi diferite de soluții $(\mathbf{B}', \mathbf{H}')$ și $(\mathbf{B}'', \mathbf{H}'')$ și notăm cu $(\mathbf{B}_d, \mathbf{H}_d)$ diferența lor. Deoarece $\text{rot } \mathbf{H}_d = 0$ se poate defini un potențial scalar V_d astfel încât $\mathbf{H}_d = -\text{grad } V_d$. În aceste condiții putem scrie că

$$\begin{aligned} \int_{\mathcal{D}_i} \mathbf{H}_d \cdot \mathbf{B}_d dv &= - \int_{\mathcal{D}_i} \text{grad } V_d \cdot \mathbf{B}_d dv = - \int_{\mathcal{D}_i} \text{div} (V_d \mathbf{B}_d) dv = - \oint_{\Sigma} V_d \mathbf{B}_d \cdot \mathbf{n} dA = \\ &= - \oint_{\Sigma} V_d \mu_0 \mathbf{H}_d \cdot \mathbf{n} dA = \mu_0 \oint_{\Sigma} V_d \text{grad } V_d \cdot \mathbf{n} dA = \mu_0 \oint_{\Sigma} V_d \frac{\partial V_d}{\partial n} dA \leq 0 \end{aligned} \quad (3.92)$$

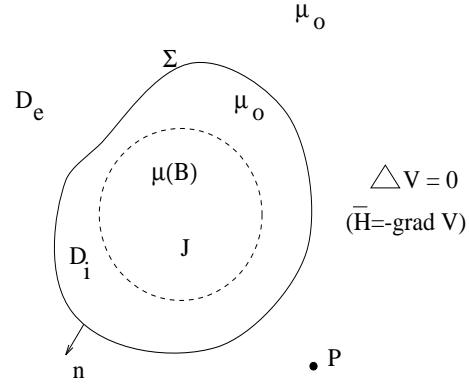


Figura 3.1: Împărțirea în domenii în cazul

3D

Pe de altă parte însă

$$\begin{aligned} \int_{\mathcal{D}_i} \mathbf{H}_d \cdot \mathbf{B}_d \, dv &= \langle \mathbf{B}_d, \mathbf{H}_d \rangle = \langle F(\mathbf{H}') - F(\mathbf{H}''), \mathbf{H}' - \mathbf{H}'' \rangle \\ &\geq \lambda \|\mathbf{H}' - \mathbf{H}''\|^2 \geq 0 \end{aligned} \quad (3.93)$$

din condiția de monotonie. Ecuațiile (3.92) și (3.93) nu pot fi satisfăcute simultan decât dacă cele două soluții sunt identice, concluzie ce contrazice ipoteza inițială.

Ca un corolar la această teoremă se poate demonstra că soluția problemei formulată anterior în domeniul interior este identică cu restricția la domeniul interior a soluției unei probleme definită în întreg spațiul infinit (cu condiția de regularitate la infinit corespunzătoare). Pentru demonstrație se consideră tot câmpul diferență. Acesta are divergență zero, rotorul zero, satisface aceeași relație pe frontieră (pentru că operatorul \mathcal{P} este liniar). Relația constitutivă dintre \mathbf{B}_d și \mathbf{H}_d este $\mathbf{B}_d = F(\mathbf{H} + \mathbf{H}_2) - \mathbf{B}_2$, tot uniform monotonă. Se observă că un câmp diferență zero satisface toate aceste condiții, și cum sunt îndeplinite toate condițiile teoremei de unicitate rezultă ca această valoare este unică.

• **Formularea problemei cu frontieră deschisă (cazul 2D)**

În cazul problemei plan paralele suprafața Σ devine curba închisă Γ , iar formularea problemei se face exclusiv în A , componenta după axa z a potențialului vector (figura 3.2).

În domeniul exterior potențialul satisface ecuația Laplace. Pe frontiera Γ suficient de regulată potențialul este continuu și de asemenea și derivata sa după normală este continuă.

TEOREMĂ DE UNICITATE (2D): *Problema determinării potențialului magnetic vector $\mathbf{A} = A\mathbf{k}$ al câmpului magnetic staționar în domeniul interior \mathcal{D}_i , care satisface în \mathcal{D}_i ecuația*

$$-\text{div}(F^{-1}\text{grad } A) = J, \quad (3.94)$$

unde F este o funcție uniform monotonă cu $\mathbf{H} \times F(\mathbf{H}) = 0$ pentru orice \mathbf{H} , admite soluție unică dacă pe frontiera Γ se impune o relație de tipul

$$\left(\frac{\partial A}{\partial n}\right)_{\Gamma} = \mathcal{P}(A_{\Gamma}), \quad (3.95)$$

unde \mathcal{P} este un operator liniar negativ.

Pentru demonstrarea acestei teoreme vom face din nou un raționament de reducere la absurd. Presupunem că ar exista două soluții A' și A'' ale problemei și vom nota

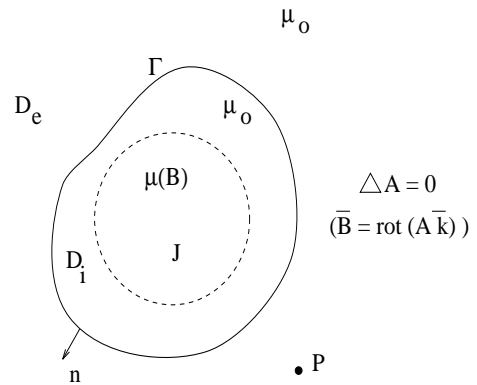


Figura 3.2: Împărțirea în domenii în cazul 2D

cu A_d diferența lor. Vom considera în același timp și câmpurile asociate lor. Rezultă că $\mathbf{B}_d = \text{rot } \mathbf{A}_d$ (unde $\mathbf{A}_d = A_d \mathbf{k}$). Deoarece $\text{rot } \mathbf{H}_d = \mathbf{0}$, există V astfel încât $\mathbf{H}_d = -\text{grad } V$. Similar ca în relația (3.92) se arată că

$$\int_{\mathcal{D}_i} \mathbf{H}_d \cdot \mathbf{B}_d \, dv = - \oint_{\Gamma} V \mathbf{B}_d \cdot \mathbf{n} \, dl. \quad (3.96)$$

Și integrala (3.96) este negativă deoarece

$$- \oint_{\Gamma} V \mathbf{B}_d \cdot \mathbf{n} \, dl = \nu_0 \oint_{\Gamma} A_d \frac{\partial A_d}{\partial n} \, dl, \quad (3.97)$$

cu $\nu_0 = 1/\mu_0$. Membrul drept din (3.97) este mai mic sau egal cu zero datorită negativității operatorului \mathcal{P} . Pentru demonstrarea egalității (3.97) se înlocuiesc în ea relațiile:

$$\mathbf{B}_d = \frac{\partial A_d}{\partial y} \mathbf{i} - \frac{\partial A_d}{\partial x} \mathbf{j}, \quad (3.98)$$

$$\nu_0 \frac{\partial A_d}{\partial y} = - \frac{\partial V}{\partial x}, \quad (3.99)$$

$$\nu_0 \frac{\partial A_d}{\partial x} = \frac{\partial V}{\partial y}, \quad (3.100)$$

rezultând o relație echivalentă

$$\oint_{\Gamma} \text{rot}(V \mathbf{A}_d) \cdot \mathbf{n} \, dl = 0, \quad (3.101)$$

întotdeauna adevărată. Relațiile (3.99) și (3.100) sunt deduse din relația $\mathbf{H}_d = -\text{grad } V$ unde pe frontieră $\mathbf{H}_d = \nu_0 \mathbf{B}_d$.

Rezultă astfel că mărimile de câmp diferență sunt nule, deci $\text{rot } \mathbf{A}_d = 0$ și cele două soluții diferă până la o constantă. Dacă se impune într-un punct din domeniul interior (sau de pe frontieră) o anumită valoare (de exemplu 0), atunci A' și A'' coincid.

3.1.7 Modelul numeric hibrid diferențial-integral (FEM-BEM)

În cazul problemelor plan-paralele, o soluție fundamentală dată de funcția Green este $G(P, Q) = \frac{1}{2\pi} \ln\left(\frac{1}{R_{PQ}}\right)$. Aplicând formula Green domeniului exterior (figura 3.2), rezultă

$$2\pi A(P) = \oint_{\Gamma} \frac{\partial A}{\partial n'}(Q) \ln\left(\frac{1}{R_{PQ}}\right) \, dl_q - \oint_{\Gamma} A(Q) \frac{\mathbf{R}_{PQ} \cdot \mathbf{n}'}{R_{PQ}^2} \, dl_q, \quad (3.102)$$

unde P este un punct în domeniul exterior. Când P se apropie de Γ integrala a doua din membrul drept al relației (3.102) este improprie. Fie P_k un punct pe Γ . Eliminăm o mică vecinătate (curba C_ε) a lui din Γ (figura 3.3). Integrala a doua devine

$$- \oint_{\Gamma} A(Q) \frac{\mathbf{R}_{PQ} \cdot \mathbf{n}'}{R_{PQ}^2} \, dl_q = - \int_{\Gamma - C_\varepsilon} A(Q) \frac{\mathbf{R}_{PQ} \cdot \mathbf{n}'}{R_{PQ}^2} \, dl_q + \int_{C_\varepsilon} A(Q) \frac{(-\mathbf{R}_{PQ}) \cdot \mathbf{n}'}{R_{PQ}^2} \, dl_q. \quad (3.103)$$

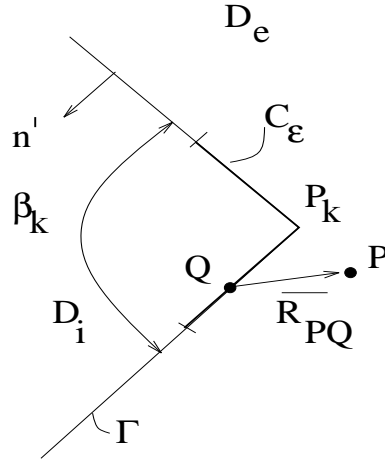


Figura 3.3: Tratarea integralei improprii

Când P tinde către P_k , integrala pe C_ε tinde către $\alpha A(P_k)$, unde pe C_ε (care este extrem de mică) se poate presupune că potențialul este constant, iar α este unghiul sub care se vede frontiera din P atunci când P tinde către P_k . Dacă am calcula integrala direct în punctul P_k atunci integrala pe C_ε este zero deoarece \mathbf{R}_{P_kQ} este perpendiculară pe \mathbf{n}' . În consecință această integrală are un salt descrescător de $\alpha A(P_k)$ la trecerea din domeniul exterior pe frontieră. De aceea, dacă dorim să scriem relația (3.102) într-un punct P_k de pe frontieră, din membrul stâng trebuie să scădem αA_k . Notăm $2\pi - \alpha = \beta_k$. Relația (3.102) scrisă într-un punct P_k al frontierei devine

$$\beta_k A_k = \oint_{\Gamma} A(Q) \frac{\mathbf{R}_{P_kQ} \cdot \mathbf{n}}{R_{P_kQ}^2} dl_Q - \oint_{\Gamma} \frac{\partial A}{\partial n}(Q) \ln \left(\frac{1}{R_{P_kQ}} \right) dl. \quad (3.104)$$

Relația (3.104) este cea care se discretizează. Pentru aceasta aproximăm frontiera Γ cu o linie poligonală formată din m segmente. Vom nota cu S_i un segment $[P_i, P_{i+1}]$ unde $i = 1, \dots, m$ iar P_{m+1} coincide cu P_1 și P_m coincide cu P_0 . Pentru a asigura continuitatea cu modelul numeric al elementelor finite ce discretizează domeniul interior vom aproxima potențialul A pe fiecare segment S_i ca având o variație liniară și derivata lui după normală ca fiind constantă pe fiecare segment S_i . Ecuația (3.104) discretizată devine

$$\beta_k A_k = \sum_{i=1}^m \int_{S_i} A(Q) \frac{\mathbf{R}_{P_kQ} \cdot \mathbf{n}}{R_{P_kQ}^2} dl_Q - \sum_{i=1}^m \int_{S_i} \frac{\partial A}{\partial n}(Q) \ln \left(\frac{1}{R_{P_kQ}} \right) dl_Q. \quad (3.105)$$

Notând $A_i = A(P_i)$, $A'_i = \left(\frac{\partial A}{\partial n} \right)_{S_i}$ și folosind interpolarea liniară pentru potențial pe segmentul S_i prima integrală din 3.105 devine

$$\int_{S_i} A(Q) \frac{\mathbf{R}_{P_kQ} \cdot \mathbf{n}}{R_{P_kQ}^2} dl_Q = A_{i+1} \int_{P_i}^{P_{i+1}} \frac{\mathbf{R}_{P_kQ} \cdot \mathbf{n}}{R_{P_kQ}^2} \frac{l_{P_iQ}}{L_i} dl_Q + A_i \int_{P_i}^{P_{i+1}} \frac{\mathbf{R}_{P_kQ} \cdot \mathbf{n}}{R_{P_kQ}^2} \left(1 - \frac{l_{P_iQ}}{L_i} \right) dl_Q. \quad (3.106)$$

Am notat cu L_i lungimea segmentului S_i iar $l_{P_i Q}$ este variabila de integrare, o coordonată care unește capătul P_i al segmentului S_i cu punctul curent Q aflat pe segmentul $[P_i, P_{i+1}]$ (figura 3.4). A doua integrală din (3.105) devine

$$\int_{S_i} \frac{\partial A}{\partial n} \ln \left(\frac{1}{R_{P_k Q}} \right) dl_Q = A'_i \int_{S_i} \ln \left(\frac{1}{R_{P_k Q}} \right) dl_Q. \quad (3.107)$$

Sistemul discretizat poate fi scris mai compact

$$\beta_k A_k = \sum_{i=1}^m G'_{ki} A_i - \sum_{i=1}^m G_{ki} A'_i \quad (3.108)$$

sau

$$\sum_{k=1}^m (G'_{ki} - \beta_k \delta_{ki}) A_i = \sum_{i=1}^m G_{ki} A'_i. \quad (3.109)$$

Scriind relația (3.109) pentru $k = 1, \dots, m$ rezultă un sistem de ecuații pe care îl scriem matriceal

$$Cp = Bd, \quad (3.110)$$

unde C și B sunt matrice pătrate de dimensiune m iar p și d sunt vectori coloană de dimensiune m reprezentând valorile potențialului A în nodurile de pe frontieră și respectiv derivatele după normală A' pe segmentele frontierei. Expresiile termenilor matricelor sunt:

$$\begin{aligned} c_{ki} &= G'_{ki} = \int_{P_{i-1}}^{P_i} \frac{\mathbf{R}_{P_k Q} \cdot \mathbf{n}}{R_{P_k Q}^2} \frac{l_{P_{i-1} Q}}{L_{i-1}} dl_Q + \int_{P_i}^{P_{i+1}} \frac{\mathbf{R}_{P_k Q} \cdot \mathbf{n}}{R_{P_k Q}^2} \left(1 - \frac{l_{P_i Q}}{L_i} \right) dl_Q, \quad i \neq k, \\ c_{kk} &= -\beta_k, \\ b_{ki} &= G_{ki} = \int_{P_i}^{P_{i+1}} \ln \left(\frac{1}{R_{P_k Q}} \right) dl_Q. \end{aligned} \quad (3.111)$$

Din ecuația (3.110) rezultă

$$d = B^{-1} Cp. \quad (3.112)$$

Matricea $P = B^{-1}C$ reprezintă discretizarea operatorului integral asociat frontierei.

Calculul integralelor din formulele (3.111) conduce la următoarele expresii adecvate implementării într-un limbaj de programare. Pentru ușurința citirii formulelor am notat cu $L = L_i$ lungimea segmentului $[P_i, P_{i+1}]$, cu $L1 = L_{i-1}$ lungimea segmentului $[P_{i-1}, P_i]$, r distanța dintre punctul P_k și punctul P_i , R distanța dintre punctul P_k și punctul P_{i+1} și $r1$ distanța dintre punctul P_k și punctul P_{i-1} . Cu A am notat aria triunghiului determinat de punctele P_k, P_i, P_{i+1} și cu $A1$ am notat aria triunghiului determinat de punctele P_k, P_i, P_{i-1} . Termenul c_{ki} l-am scris ca fiind suma dintre expresiile $i1$ și $i2$ unde $i1$ este rezultatul integralei de la P_{i-1} la P_i și $i2$ este rezultatul integralei de la P_i la P_{i+1} . Cu d am notat distanța dintre punctul P_{i+1} și punctul P_{i-1} .

- Elementul b_{ki} :

$$b_{ki} = \frac{L}{2} \left(-2 + \frac{(-L^4 + 2L^2 r^2 - r^4 + 2L^2 R^2 + 2r^2 R^2 - R^4) \operatorname{ArcTanh}\left(\frac{-L^2 + r^2 - R^2}{\sqrt{L^4 - 2L^2 r^2 + r^4 - 2L^2 R^2 - 2r^2 R^2 + R^4}}\right)}{L^2 \sqrt{L^4 - 2L^2 r^2 + r^4 - 2L^2 R^2 - 2r^2 R^2 + R^4}} - \frac{(-L^4 + 2L^2 r^2 - r^4 + 2L^2 R^2 + 2r^2 R^2 - R^4) \operatorname{ArcTanh}\left(\frac{L^2 + r^2 - R^2}{\sqrt{L^4 - 2L^2 r^2 + r^4 - 2L^2 R^2 - 2r^2 R^2 + R^4}}\right)}{L^2 \sqrt{L^4 - 2L^2 r^2 + r^4 - 2L^2 R^2 - 2r^2 R^2 + R^4}} - \frac{(-L^2 - r^2 + R^2) \log(r^2)}{2L^2} + \log(R^2) + \frac{(-L^2 - r^2 + R^2) \log(R^2)}{2L^2} \right)$$

- Elementul $c_{ki} = i1 + i2$ cu $k \neq i$:

$$i1 = 2A l \left(\frac{(Ll^2 - r^2 + r l^2) \operatorname{ArcTanh}\left(\frac{-Ll^2 - r^2 + r l^2}{\sqrt{Ll^4 - 2Ll^2 r^2 + r^4 - 2Ll^2 r l^2 - 2r^2 r l^2 + r l^4}}\right)}{Ll^2 \sqrt{Ll^4 - 2Ll^2 r^2 + r^4 - 2Ll^2 r l^2 - 2r^2 r l^2 + r l^4}} - \frac{(Ll^2 - r^2 + r l^2) \operatorname{ArcTanh}\left(\frac{Ll^2 - r^2 + r l^2}{\sqrt{Ll^4 - 2Ll^2 r^2 + r^4 - 2Ll^2 r l^2 - 2r^2 r l^2 + r l^4}}\right)}{Ll^2 \sqrt{Ll^4 - 2Ll^2 r^2 + r^4 - 2Ll^2 r l^2 - 2r^2 r l^2 + r l^4}} + \frac{\log(r^2)}{2Ll^2} - \frac{\log(r l^2)}{2Ll^2} \right)$$

$$i2 = 2A \left(\frac{(L^2 - r^2 + R^2) \operatorname{ArcTanh}\left(\frac{-L^2 + r^2 - R^2}{\sqrt{L^4 - 2L^2 r^2 + r^4 - 2L^2 R^2 - 2r^2 R^2 + R^4}}\right)}{L^2 \sqrt{L^4 - 2L^2 r^2 + r^4 - 2L^2 R^2 - 2r^2 R^2 + R^4}} - \frac{(L^2 - r^2 + R^2) \operatorname{ArcTanh}\left(\frac{L^2 + r^2 - R^2}{\sqrt{L^4 - 2L^2 r^2 + r^4 - 2L^2 R^2 - 2r^2 R^2 + R^4}}\right)}{L^2 \sqrt{L^4 - 2L^2 r^2 + r^4 - 2L^2 R^2 - 2r^2 R^2 + R^4}} + \frac{\log(r^2)}{2L^2} - \frac{\log(R^2)}{2L^2} \right)$$

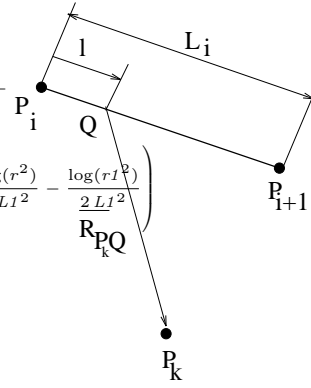


Figura 3.4. Notății relative la un segment de frontieră

- Elementul $c_{ii} = -\beta_i$:

$$c_{ii} = \arccos\left(\frac{-d^2 + L^2 + Ll^2}{2LlL}\right)$$

Cuplarea propriu-zisă cu metoda elementelor finite se face asamblând mai întâi matricea elementelor finite ca și cum ar exista doar condiții de frontieră Neumann. Termenii asociați derivatelor după normală se substituie apoi cu expresiile deduse mai sus.

3.1.8 Indicatori de eroare a soluției numerice. Rețele adaptive de discretizare.

Scopul acestui paragraf este de a caracteriza abaterea dintre soluția numerică și cea exactă a unei probleme de câmp magnetic. Evaluarea riguroasă a acestei abateri ar presupune cunoașterea soluției exacte. Majoranții ai acestei abateri sunt cunoscuți sub numele de *estimatori de eroare*. Din păcate, determinarea estimatorilor de eroare necesită un efort de calcul important iar relevanța lor nu este în toate cazurile cea așteptată (de multe ori ei dau caracterizări ale erorii prea pesimiste). Din aceste motive vom prefera utilizarea *indicatorilor de eroare* care au proprietatea că se anulează odată cu abaterea dintre soluția

exactă și cea numerică. Ei permit identificarea punctelor din domeniul de calcul în care abaterea dintre soluția exactă și cea numerică este semnificativă. Prin folosirea unor algoritmi de rafinare succesivă locală a rețelei de discretizare în aceste puncte se obține în final o eroare mică (așa cum este caracterizată ea de indicatorul ales) atât din punct de vedere global cât și local (prin adaptarea rețelei de discretizare la soluție).

Să presupunem că se lucrează cu potențialul vector \mathbf{A} , iar \mathbf{A}' este aproximarea numerică a soluției exacte \mathbf{A} . În aceste condiții ecuația $\text{div } \mathbf{B}' = 0$, unde $\mathbf{B}' = \text{rot } \mathbf{A}'$, este automat satisfăcută. În plus, dacă $\mathbf{H}' = F(\mathbf{B}')$ atunci și ecuația constitutivă de material este satisfăcută exact. Singura ecuație care este satisfăcută aproximativ este $\text{rot } \mathbf{H} = \mathbf{J}$. În consecință se propune calculul *densității de curent reziduale* definită prin

$$\mathbf{J}_r = \text{rot } \mathbf{H}' - \mathbf{J} \quad (3.113)$$

și care va caracteriza local eroarea soluției numerice (va fi deci un indicator de eroare).

În cazul metodei elementelor finite în probleme plan-paralele, pentru o discretizare în elemente triunghiulare de ordinul întâi, câmpul este constant pe triunghi și în consecință $\text{rot } \mathbf{H} = \mathbf{0}$ pe fiecare element. Curentul rezidual va avea doar distribuție superficială pe laturile elementului și, eventual, pe volumul elementului dacă $\mathbf{J} \neq \mathbf{0}$. Direcția curentului rezidual va fi evident tot direcția \mathbf{k} și notând $\mathbf{J}_r = J_r \mathbf{k}$ rezultă că

$$J_r = \text{div}(\nu \text{grad } A') + J. \quad (3.114)$$

Soluția numerică A' poate fi deci interpretată ca fiind soluția exactă a unei probleme în care sursele de câmp sunt curenții J'

$$-\text{div}(\nu \text{grad } A') = J'. \quad (3.115)$$

Ecuațiile (3.114) și (3.115) sunt privite în sensul teoriei distribuțiilor. Pornind de la sistemul de ecuații care se rezolvă în cazul folosirii metodei Galerkin

$$-\int_{\Omega} \nu \text{grad } \Psi_i \cdot \text{grad } A' \, d\Omega + \int_{\Omega} \Psi_i J \, d\Omega + \oint_{\Gamma} \nu \Psi_i g \, dl = 0 \quad i = 1, \dots, n, \quad (3.116)$$

unde Ω este domeniul bidimensional al problemei, Ψ_i este funcția de formă asociată nodului i , iar $g = \partial A / \partial n$ este condiția de frontieră Neumann, asociată unei părți C_N din frontiera Γ a domeniului, se poate demonstra ușor că

$$\int_{\Omega} \Psi_i J_r \, d\Omega - \int_{C_N} \Psi_i J_{rN} \, dl = 0, \quad (3.117)$$

unde $J_{rN} = \nu(\frac{\partial A'}{\partial n} - g)$ este densitatea unei pânze de curent reziduale corespunzătoare frontierei Neumann. Pe frontiera Dirichlet $\Psi = 0$ și de aceea integrala pe frontieră Γ a fost redusă la integrala pe porțiunea C_N . Ecuația (3.117) indică o autoechilibrare globală a reziduurilor.

În consecință, metoda numerică rezolvă exact o altă problemă în care curenții inițiali din elemente au fost redistribuiți pe laturile acestora astfel încât ei se autoechilibrează.

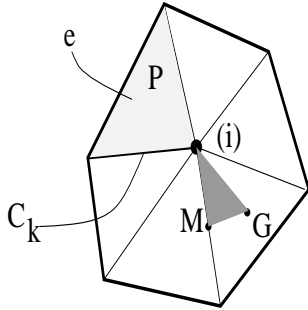


Figura 3.5: Domeniul Ω_i

Pentru un nod interior, ținând cont de proprietățile funcției Ψ_i , rezultă următoarea formă a condiției (3.117):

$$\begin{aligned} \int_{\Omega_i} \Psi_i J_r d\Omega &= 0 \iff \\ \int_{\Omega_i} \Psi_i J' d\Omega &= \int_{\Omega_i} \Psi_i J d\Omega \iff \\ \sum_{C_k \subset \Omega_i} \int_{C_k} \Psi_i J'_k dl_k &= \int_{\Omega_i} \Psi_i J d\Omega, \end{aligned} \quad (3.118)$$

unde s-a notat cu Ω_i domeniul determinat de mulțimea triunghiurilor care conțin nodul i (figura 3.5), cu C_k o muchie care concură la nodul i și cu J'_k densitatea pânzei de curent reziduale de pe muchia C_k . Relațiile (3.118) arată că autoechilibrarea erorilor are loc și local, în jurul fiecărui nod. (Curentul real din vecinătatea nodului i , care este o treime din curentul total al elementelor ce înconjoară nodul, este egal cu curentul redistribuit în vecinătatea nodului i , care este egal cu jumătate din curenții laturilor care concură la nod.) Din acest motiv indicatorul local de eroare pare mai natural să fie asociat entității element sau celei de laturi decât entității nod.

• **Indicator de eroare bazat pe curentul rezidual**

Celula elementară în care se analizează balanța curentului rezidual reprezintă un triunghi care are un vârf în nodul (i), un vârf în centrul de greutate al elementului analizat și al treilea vârf în mijlocul laturii elementului care trece prin nodul (i) (figura 3.5). Curentul rezidual al tuturor celulelor elementare care concură la nodul (i) este nul. În schimb, curenții reziduali din cele patru celule elementare ce conțin segmente dintr-o latură precum și curenții din cele șase celule elementare ale unui element triunghiular nu se anulează în mod necesar. Mărimile acestor curenți reprezintă criterii de performanță (indicatori de eroare) pentru laturi și respectiv pentru elemente. Din analiza cazului unidimensional rezultă că valorile curenților reziduali pe celulele elementare sunt corelate cu densitatea spațială a curentului real pe când curentul total rezidual pe element este corelat cu derivata a doua a curentului real.

Din păcate, acești indicatori de eroare nu pot fi transformați ușor în mărimi relative, deoarece este dificil de stabilit o mărime de raportare de tip curent pentru o problemă arbitrară (de exemplu în cazul în care $J = 0$, $g = 0$).

• **Indicator de eroare bazat pe o mărime de câmp magnetic**

Datorită redistribuirii curenților, eroarea este chiar diferența dintre intensitatea câmpului magnetic produsă de curentul inițial (soluția exactă) și cea produsă de curenții redistribuiți (soluția aproximativă). Determinarea acestei mărimi este complicată și imposibilă fără cunoașterea soluției exacte, de aceea algoritmul propus calculează diferența a două mărimi de tip intensitate a câmpului magnetic determinate cu formula Biot-Savart-Laplace. În centrul P al fiecărui element se evaluează câmpul produs de curentul inițial din element și câmpul produs de curenții redistribuiți. Curenții din elementele și laturile vecine sunt ponderate cu funcțiile de formă astfel încât au o influență mai mică. Fiecărui element i se asociază un număr indicator de eroare. Chiar dacă mediul inițial este neomogen, calculul câmpului se face pentru un mediu omogen, acesta fiind punctul slab al metodei. Neomogenitatea este ascunsă în valorile curenților redistribuiți pe laturile elementelor.

Algoritm de calcul al indicatorului de eroare de tip intensitate a câmpului magnetic:

1. **pentru** fiecare element e
 - 1.1. identifică centrul său (punctul P)
 - 1.2. $\mathbf{H} = \mathbf{0}$
 - 1.3. **pentru** fiecare nod i al elementului e
 - 1.3.1. **pentru** fiecare element f ce conține nodul i
 - 1.3.1.1. calculează $\mathbf{H}_1(P)$ datorat curentului real J din elementul f ponderat cu Ψ_i

$$1.3.1.2. \mathbf{H} = \mathbf{H} + \mathbf{H}_1(P)$$

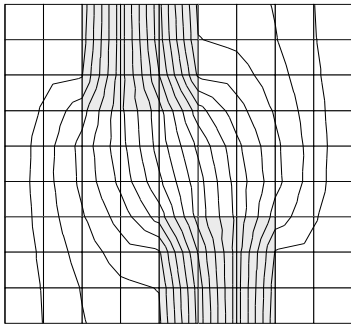
1.3.2. **pentru** fiecare latură ce concură la nodul i

1.3.2.1. calculează $\mathbf{H}_2(P)$ datorat curentului J'_s de pe latură ponderat cu Ψ_i

$$1.3.2.2. \mathbf{H} = \mathbf{H} - \mathbf{H}_2(P)$$

$$1.4 \text{ indicator}[e] = |\mathbf{H}|$$

După calculul indicatorilor, elementele sunt sortate crescător după valoarea indicatorului de eroare, și un anumit procent din coada listei este rafinat. Figurile 3.7 și 3.8 reprezintă rafinări consecutive ale problemei din figura 3.6.



[CC] Contents of A

Figura 3.6: Discretizarea inițială și spectrul inducției magnetice

Indicatorul de eroare propus pare a indica corect zonele unde este necesară rafinarea discretizării și anume la interfețele dintre materiale diferite și mai ales în zonele în care aceste interfețe nu sunt netede (în colțuri). Mărimea propusă poate fi făcută relativă prin raportarea ei la valoarea maximă a intensității câmpului magnetic din problemă.

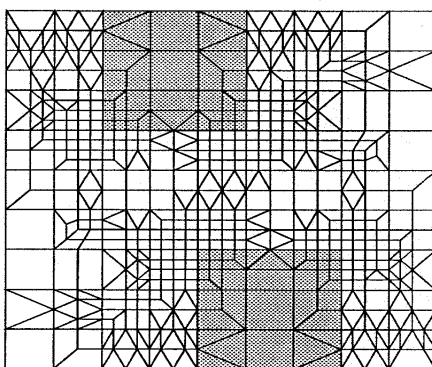


Figura 3.7: Discretizarea după două rafinări succesive

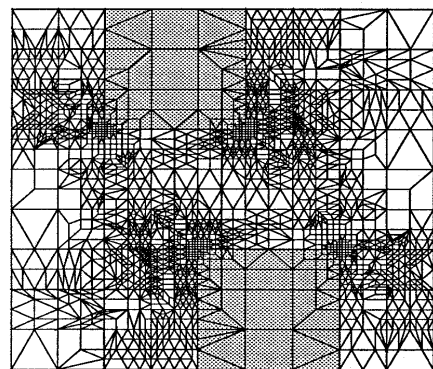


Figura 3.8: Discretizarea după patru rafinări succesive

• **Indicator de eroare bazat pe o mărime de tip energetic**

Pentru a caracteriza local eroarea soluției numerice se poate folosi circulația lui \mathbf{H}' pe o curbă ce înconjoară fiecare nod (n). Fie această curbă Γ (figura 3.9) alcătuită din segmente de mediane.

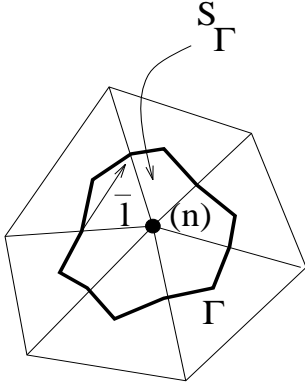


Figura 3.9: Domeniul Ω_n și curba Γ

Ea va trece prin centrele de greutate ale triunghiurilor care îl au pe (n) ca nod. *Curentul rezidual asociat unui nod* este curentul rezidual pe suprafața S_Γ

$$\begin{aligned} i_{r_n} &= \int_{S_\Gamma} (\text{rot } \mathbf{H}' - \mathbf{J}) \cdot \mathbf{n} \, dA = \oint_\Gamma \mathbf{H} \cdot d\mathbf{l} - \int_{S_\Gamma} \mathbf{J} \cdot \mathbf{n} \, dA = \\ &= \sum_{e \ni (n)} \left(\int_{C_e} \mathbf{H} \cdot d\mathbf{l} - \frac{J_e \Delta_e}{3} \right) = \sum_{e \ni (n)} \left(\mathbf{H}_e \mathbf{l}_e - \frac{J_e \Delta_e}{3} \right), \end{aligned} \quad (3.119)$$

în care C_e este o curbă ce unește mijloacele laturilor din e care concură la nodul (n) iar Δ_e este aria elementului (triunghiului) e . Curba C_e poate fi aleasă arbitrar în interiorul elementului e deoarece \mathbf{H} este constant și integrala nu depinde de forma ei ci doar de punctele ei extreme. În consecință ea va fi aleasă ca segmentul de dreaptă \mathbf{l} care unește cele două mijloace ale laturilor (figura 3.9).

Conform relației (3.118), pentru o soluție numerică determinată exact curentul i_{r_n} are valoare nulă pentru orice nod n .

Proiecția reziduuului peste soluție reprezintă un indicator global de eroare de natură energetică, care va fi numit *reziduu energetic global*

$$W_r = \int_\Omega \mathbf{J}_r \cdot \mathbf{A} \, d\Omega = \sum_e \int_e \mathbf{J}_r \cdot \mathbf{A} \, d\Omega = \sum_{(n)} W_{r_n} = \sum_{(n)} i_{r_n} A_n. \quad (3.120)$$

Deoarece \mathbf{A} este o combinație liniară de funcții de formă Ψ_i și proiecția lui \mathbf{J}_r pe aceste funcții este nulă conform (3.118) rezultă că și W_r trebuie să fie nulă pentru soluțiile numerice determinate exact. În fiecare nod contribuția W_{r_n} la reziduu energetic global se calculează înmulțind potențialul A_n cu valoarea curentului rezidual i_{r_n}

$$W_{r_n} = A_n \sum_{e \ni (n)} \left(\mathbf{H}_e \mathbf{l}_e - \frac{J_e \Delta_e}{3} \right). \quad (3.121)$$

Deoarece o mărime relativă este mai utilă decât una absolută, este necesară definirea unei mărimi globale de tip energie care să fie folosită pentru raportare. Una potrivită ar putea fi

$$W = \sum_e \nu_e B_e^2 \Delta_e, \quad (3.122)$$

măsurată în Joule și care este nenulă în cazul soluțiilor netriviiale.

În consecință, parametrul global ce caracterizează relativ eroarea cu care a fost determinată soluția numerică este

$$\varepsilon = \frac{W_r}{W} = \frac{\sum_{(n)} [A_n \sum_{e \ni (n)} (\mathbf{H}_e \mathbf{l}_{en} - J_e \Delta_e / 3)]}{\sum_e \nu_e B_e^2 \Delta_e}. \quad (3.123)$$

Din punct de vedere local, fiecărui nod i se poate asocia un parametru de performanță

$$\varepsilon_{(n)} = \frac{W_{r_n}}{W} = \frac{A_n}{W} \sum_{e \ni (n)} (\mathbf{H}_e \mathbf{l}_{en} - J_e \Delta_e / 3), \quad (3.124)$$

astfel încât $\varepsilon = \sum_{(n)} \varepsilon_{(n)}$. ε permite compararea calității soluției numerice obținute prin metode diferite și în consecință ar putea fi folosit drept criteriu unic de oprire a iterațiilor neliniare în tehnicile polialgoritmice (Newton și iterație simplă).

Parametrii $\varepsilon_{(n)}$ pot fi calculați cu relația (3.124) prin parcurgerea o singură dată a elementelor și adăugarea contribuției lor la componentele vectorului $[\varepsilon]$. (Fiecare element contribuie cu expresia $A_n(u_m - J_e \Delta_e / 3)$ la fiecare nod al său, în care u_m este pe rând $\mathbf{H}_e \mathbf{l}_{e1}$, $\mathbf{H}_e \mathbf{l}_{e2}$, $\mathbf{H}_e \mathbf{l}_{e3}$ iar A_n este A_1 , A_2 și respectiv A_3 . Cu această ocazie se poate evalua și W (fiecare triunghi contribuind cu $\nu_e B_e^2 \Delta_e$).

• Rafinarea rețelei

După cum s-a constatat, vectorul $[\varepsilon_n]$ nu este un indicator de eroare, el caracterizează mai mult abaterea soluției numerice aproximative (de exemplu obținută după câteva iterații neliniare) față de cea numerică exactă (limita șirului de iterații neliniare), decât abaterea soluției numerice față de cea exactă.

Pentru o obține un indicator de eroare se propune folosirea unui vector $[\varepsilon_e]$ la care fiecare componentă este asociată unui element e

$$\varepsilon_e = \frac{W_{r_e}}{W} = \frac{1}{W} \int_e \mathbf{J}_r \cdot \mathbf{A} \, d\Omega = \frac{1}{W} \left[\frac{1}{2} \sum_{f=1}^3 A_f (\mathbf{H}_e - \mathbf{H}_f) \cdot \mathbf{l}_{ef} - J_e A_e \Delta_e \right], \quad (3.125)$$

în care s-a notat cu A_e soluția în centrul elementului e , A_f soluția în mijlocul laturii f din elementul e (de lungime l_{ef}), \mathbf{H}_e și \mathbf{H}_f câmpul magnetic din elementul e și respectiv din elementul vecin f . Suma componentelor acestui vector este chiar ε (formula (3.123)), deci reprezintă un indicator pentru exactitatea determinării soluției numerice, în schimb componentele sale reprezintă indicatori locali de eroare și implicit norma sa reprezintă un indicator global de eroare.

Un alt indicator de eroare ar putea fi asociat unei laturi l . El are expresia

$$\varepsilon_l = \frac{W_{r_l}}{W} = \frac{1}{W} \int_{\Omega_e} \mathbf{J}_r \cdot \mathbf{A} \, d\Omega = \frac{1}{W} [A_l (\mathbf{H}_e - \mathbf{H}_f) \cdot \mathbf{l}_{ef} - J_e A_e \Delta_e / 3 - J_f A_f \Delta_f / 3], \quad (3.126)$$

unde Ω_e este reuniunea celor două triunghiuri (de arii $\Delta_e/3$ și $\Delta_f/3$ care au baza l_{ef} și vârfurile în centrele de greutate ale elementelor e și f care conțin muchia l_{ef} . A_e și A_f sunt valorile soluției în centrele de greutate ale acestor triunghiuri. Pornind de la indicatorul de eroare asociat fiecărei muchii se poate determina un indicator asociat triunghiurilor prin sumarea semiindicatorilor asociați celor trei laturi ale elementului. Aceasta se pare că asigură o robustețe maximă în identificarea triunghiurilor cu probleme.

Nodurile la care parametrul $|\varepsilon_e|$ are valori mari indică zone locale la care soluția este imprecis calculată. O metodă de identificare a elementelor care trebuie rafinate ar putea fi cea de a selecta din cele E elemente primele E/λ elemente ² la care erorile locale $|\varepsilon_e|$ au valorile cele mai mari.

Deoarece un triunghi se înlocuiește în medie cu alte 4 triunghiuri, rezultă că alegând $\lambda = 3$, la fiecare rafinare practic se dublează numărul de elemente (și implicit de noduri). Dacă se dorește ca după m rafinări succesive rețeaua să conțină de r ori mai multe elemente atunci trebuie aleasă valoarea $\lambda = 3/(\sqrt[m]{r} - 1)$. În mod uzual, probabil că mai puțin de o treime din numărul de noduri vor fi declarate cu probleme.

Calitatea rețelei de discretizare (adaptarea ei la variația câmpului) poate fi caracterizată prin dispersia componentelor vectorului $[\varepsilon_e]$ în jurul valorii medii, respectiv prin raportul

$$q = \frac{\|\varepsilon\|_2}{\|\varepsilon\|_{\max}} = \frac{\sqrt{\frac{1}{E} \sum_{e=1}^E \varepsilon_e^2}}{\max_{e=1,E} |\varepsilon_e|}, \quad (3.127)$$

care are valoare unitară în cazul unei rețele perfect adaptate (care asigură distribuția uniformă a erorilor locale). Cu cât valoarea factorului de calitate q este mai mică cu atât rețeaua este mai prost adaptată. Înmulțit cu zece el dă ”nota” rețelei.

Această abordare are avantajul că poate fi generalizată fără dificultăți în cazul 3D. Reziduul energetic asociat unui element e este

$$W_{r_e} = \int_{\Omega_e} \mathbf{J}_r \cdot \mathbf{A} \, d\Omega = \frac{1}{2} \sum_{f=1}^m \mathbf{A}_f \cdot [\mathbf{n}_{fe} \times (\mathbf{H}_e - \mathbf{H}_f)] S_f - \mathbf{J}_e \cdot \mathbf{A}_e \Omega_e, \quad (3.128)$$

în care m reprezintă numărul de fețe ale elementului ($m = 4$ în cazul elementelor tetraedrale), \mathbf{A}_f reprezintă potențialul vector în centrul feței f , S_f este aria feței f , \mathbf{H}_e și \mathbf{H}_f reprezintă intensitatea câmpului magnetic în elementul e și respectiv în cel vecin f , iar \mathbf{A}_e este potențialul vector în centru elementului e , de volum Ω_e .

²Gândindu-ne la evoluția rețelei de discretizare simultan cu procesul de optimizare, în SA λ poate fi corelat cu temperatura printr-o transformare afină astfel încât el să scadă spre final. În GA λ poate scade exponențial cu numărul generației curente (eventual același tip de scădere folosită de probabilitatea mutației). Folosind modelul metodei multigrad, rafinarea succesivă poate fi aplicată atât pe parcursul procesului iterativ de rezolvare a sistemului liniar de ecuații obținut prin discretizare cât mai ales pe parcursul iterațiilor neliniare.

Reziduul energetic asociat unei fețe f care separă elementul e de elementul g este

$$W_{r_f} = \int_{\Omega_f} \mathbf{J}_r \cdot \mathbf{A} \, d\Omega = \mathbf{A}_f \cdot [\mathbf{n}_{ge} \times (\mathbf{H}_e - \mathbf{H}_g)] S_f - \mathbf{J}_e \mathbf{A}_e V_e - \mathbf{J}_g \mathbf{A}_g V_g, \quad (3.129)$$

în care Ω_f este reuniunea celor două piramide de bază comună fața f și cu vârfurile plasate în centrele elementelor e și g , \mathbf{A}_f este potențialul vector în centrul feței f , \mathbf{n}_{ge} este versorul normal la fața f , \mathbf{H}_e , \mathbf{H}_g și \mathbf{J}_e , \mathbf{J}_g sunt intensitățile câmpurilor magnetice, respectiv densitățile de curent din cele două elemente e și g care au fața f comună, V_e și V_g sunt volumele celor două piramide iar \mathbf{A}_e și \mathbf{A}_g sunt potențialele în centrele de greutate ale acestor piramide.

Singura deosebire de principiu care apare în cazul 3D față de cazul 2D este aceea că $\lambda = 7/(\sqrt[3]{r} - 1)$ deoarece un tetraedru se înlocuiește prin rafinare cu 8 tetraedre (m fiind numărul de rafinări succesive, iar r fiind factorul de creștere a numărului de noduri pe întregul proces).

Algoritm de rafinare a rețelei pe parcursul iterațiilor neliniare:

1. $n_{nod} =$ număr de noduri
 2. $n_{nodmax} =$ număr maxim de noduri
 3. $k_{max} =$ număr maxim de iterații
 4. $eps0 =$ eroarea impusă
 5. $r = 100$; factor de creștere a numărului de noduri pe întregul proces
 6. $k = 0$; contor iterații
 7. $\lambda = 3/(\sqrt[3]{k_{max} r} - 1)$
 8. **repetă**
 - 8.1. $k = k + 1$
 - 8.2. **dacă** $k > k_{max}$ **atunci stop**
 - 8.3. efectuează iterație neliniară
 - 8.4. $e = \|\delta \mathbf{I}\|/\|\mathbf{I}\|$; corecția relativă a polarizației magnetice
 - 8.5. calculează vectorul indicator de eroare ε
 - 8.6. **scrie** $k, n_{nod}, e, \varepsilon, \|\varepsilon\|_2, q$
 - 8.7. **dacă** $\|\varepsilon\|_2 > eps0$ **și** $n_{nod} (\frac{3}{\lambda} + 1) < n_{nodmax}$
atunci rafinează rețea și interpolează soluție
- până când** $e < eps0$

3.2 Teste numerice

Acest paragraf cuprinde rezultatele numerice ale implementării celor trei algoritmi detaliați până acum: polarizație, polarizație cu relaxare, Newton. Implementarea³ a fost făcută în limbajul de programare C și sistemul de operare LINUX, programele fiind rulate pe un sistem PC Pentium (75 MHz). Pentru validarea acurateții au fost făcute comparații cu rezultatele unor probleme identice sau asemănătoare modelate cu ajutorul programului de calcul de câmp MEGA, produs de Universitatea din Bath, și care a fost rulat pe o stație grafică HP/720. În rulările cu MEGA s-au folosit parametrii implicați ai programului: metoda Newton, condiția de oprire fiind ca eroarea relativă maximă a inducției magnetice să fie sub 0.5 %, iar pentru rezolvarea sistemului liniar fiind folosită metoda gradientilor conjugați cu o eroare de 1.00E-12. Rezolvarea sistemului liniar în FAP a fost făcută cu metoda suprarelaxării succesive (cu factor de relaxare optim) cu o eroare de 1.00E-7.

3.2.1 Probleme de test

Prima problemă de test o constituie o bară feromagnetică situată într-un câmp uniform. Discretizarea este extrem de rară (figura 3.10), având 25 noduri și 36 elemente. Rețeaua de discretizare făcută cu MEGA este identică. Nodurile de pe latura din stânga au asociată o condiție Dirichlet nulă, iar nodurile de pe latura din dreapta au asociată o condiție Dirichlet $A = 0.02$, în rest s-au presupus condiții Neumann nule.

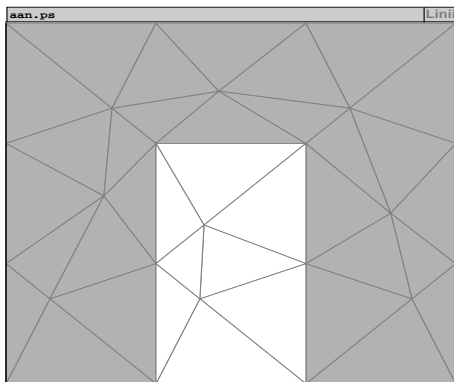


Figura 3.10: Test 1 - rețeaua de discretizare

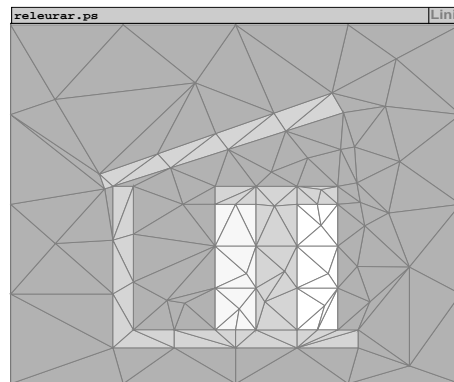


Figura 3.11: Test 2 - rețeaua de discretizare

Al doilea test îl constituie un circuit magnetic neliniar (de tip releu), având rețeaua de

³Discretizarea a fost făcută cu preprocesorul FAP[2] dezvoltat în LMN, iar autoarea a implementat și depanat modulul de rezolvare neliniară și postprocesare.

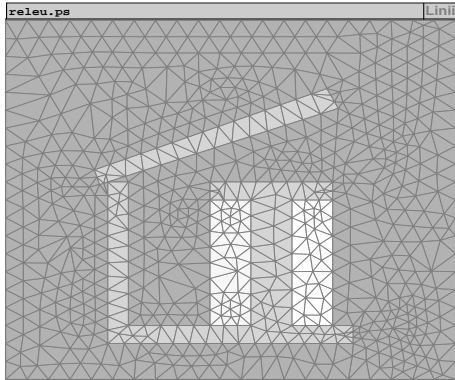


Figura 3.12: Test 3 - rețeaua de discretizare

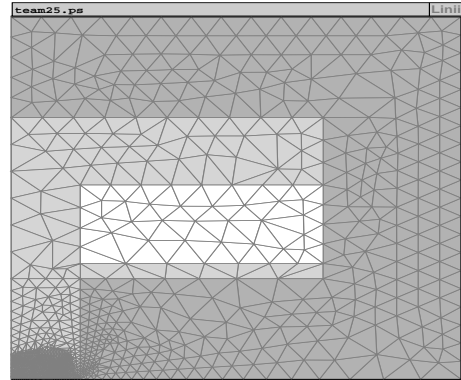


Figura 3.13: Test 4 - rețeaua de discretizare

discretizare din figura 3.11. Rețeaua are 94 noduri și 170 elemente. Prin bobină circulă un curent cu o densitate de $3.7143 \cdot 10^6$ A/m².

Testul al treilea se referă la aceeași problemă ca la testul 2, rețeaua de discretizare fiind însă mai fină (725 noduri, 1366 elemente, figura 3.12). Pentru acest caz rețelele de discretizare folosite de FAP și MEGA nu sunt identice însă au aproximativ aceleași dimensiuni (rețeaua generată cu MEGA are 748 noduri și 1431 elemente).

Testul al patrulea urmărește rezolvarea configurației problemei TEAM 25. Și în acest caz rețelele au aproximativ aceleași dimensiuni (rețeaua generată cu MEGA are 1446 noduri și 2749 elemente, iar rețeaua generată cu FAP are 1420 noduri și 2703 elemente).

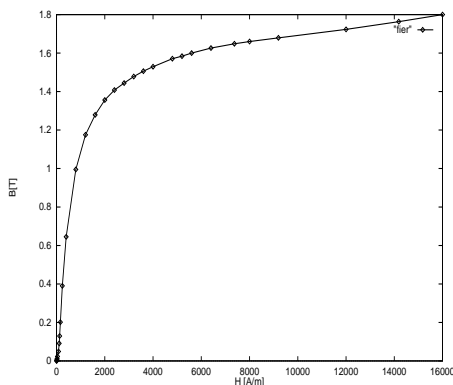


Figura 3.14: Curba de magnetizare pentru testele 1,2 și 3

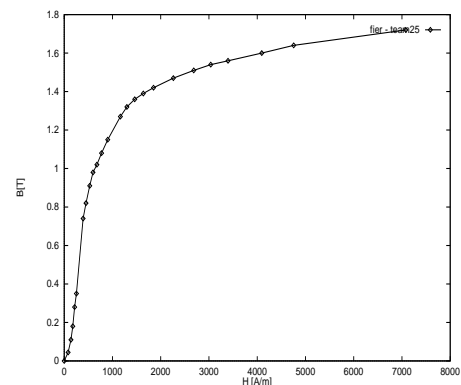


Figura 3.15: Curba de magnetizare pentru testul 4

Pentru primele trei teste am folosit curba de magnetizare din figura 3.14, iar pentru testul al patrulea am folosit curba de magnetizare din figura 3.15.

3.2.2 Rezultate numerice

Pentru validarea algoritmilor am comparat valorile potențialului în noduri (pentru testele 1 și 2, pentru care discretizările au fost identice cu cele folosite în programul MEGA). Tabelul 3.1 conține informații despre câteva valori nodale.

Test	Nr. nod	MEGA	FAP polarizație	FAP polarizație cu relaxare	FAP Newton
TEST 1	5	.30982805E-03	3.091263E-04	3.091232E-04	–
	10	.46465615E-02	4.646037E-03	4.646040E-03	–
	15	.19233620E-01	1.923497E-02	1.923498E-02	–
	20	.18848182E-01	1.884921E-02	1.884921E-02	–
TEST 2	6	.72526785E-04	7.250396E-05	7.250669E-05	7.252645E-05
	16	.31375288E-02	3.134394E-03	3.134594E-03	3.137492E-03
	47	.26050894E-02	2.602314E-03	2.602488E-03	2.605059E-03
	89	-2.197060E-04	-2.196935E-04	-2.196939E-04	-2.197061E-04

Tabelul 3.1: Câteva valori nodale

Se observă că rezultatele nu diferă până la 3 cifre semnificative, lucru adevărat pentru mai multe noduri inspectate nu numai pentru cele din tabel. Rezultatele au fost obținute pentru o condiție de oprire $\|\delta I\|/\|I\| < 10^{-5}$ în cazul metodei polarizației și pentru o condiție de oprire $\|\delta A\|/\|A\| < 10^{-5}$ în cazul metodei Newton pentru testul al doilea.

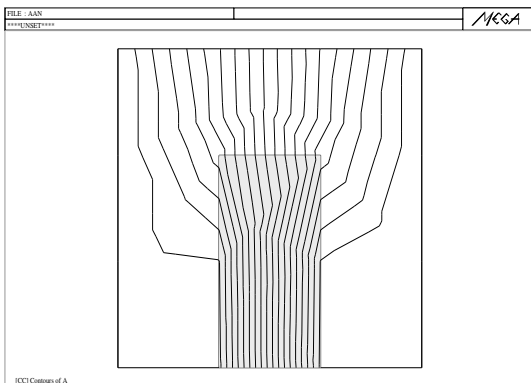


Figura 3.16: Test 1 - spectrul inducției (MEGA)

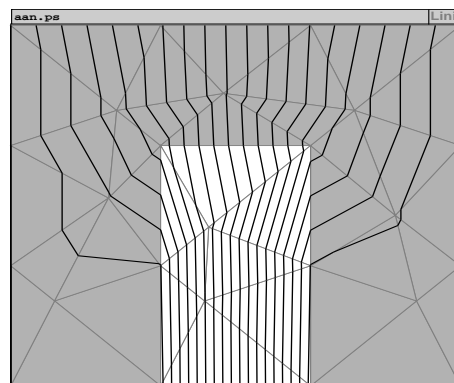


Figura 3.17: Test 1 - spectrul inducției (FAP)

Tabelul 3.2 conține valorile energiei obținute în situațiile descrise mai sus. Diferența FAP-MEGA este sub 0.06 % pentru primele două teste, de 3.26 % pentru testul 3 și de 0.3 % pentru testul 4 în cazul folosirii metodei polarizației și de 0.09 % pentru testul 2 și de 3.07 % pentru testul 3 în cazul metodei Newton.

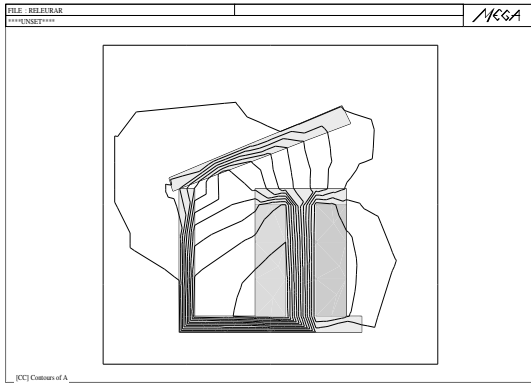


Figura 3.18: Test 2 - spectrul inducției (MEGA)

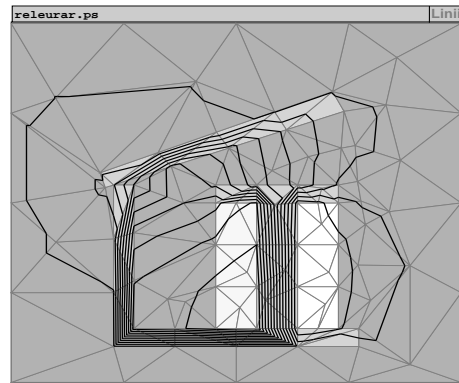


Figura 3.19: Test 2 - spectrul inducției (FAP)

Test	MEGA	FAP PB	FAP - PB cu relaxare	FAP Newton
TEST 1	71.1803	71.1744	71.1744	–
TEST 2	2.6327	2.6311	2.6313	2.6351
TEST 3	3.8626	3.7367	3.7367	3.7441
TEST 4	39.4790	39.5983	39.5982	–

Tabelul 3.2: Energii în model

3.2.3 Rata și raza de convergență

Pentru a evalua performanțele algoritmilor implementați a fost studiat modul în care aceștia converg către soluție.

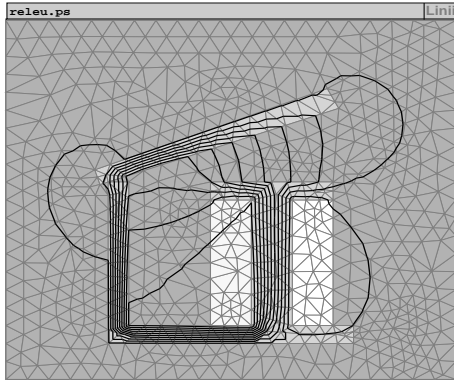


Figura 3.20: Test 3 - spectrul inducției (FAP)

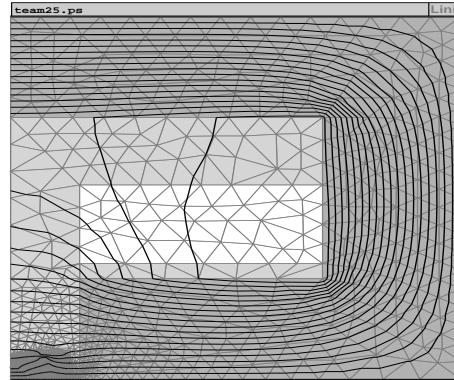


Figura 3.21: Test 4 - spectrul inducției (FAP)

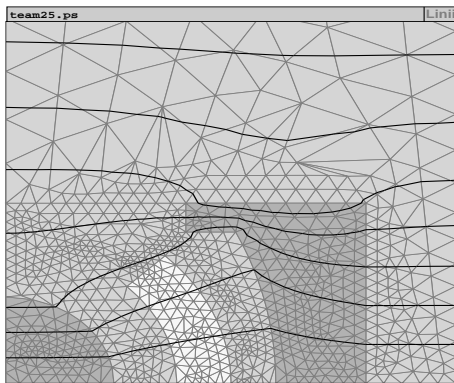


Figura 3.22: Test 4 - spectrul inducției în zona de interes ("zoom") obținut cu FAP

Figura 3.16 prezintă spectrul inducției magnetice așa cum este obținut de metoda polarizației, iar figura 3.17 același spectru obținut cu MEGA pentru prima problemă de test. Figurile 3.18 și 3.19 conțin spectrele pentru testul al doilea.

Figurile 3.20, 3.21 și 3.22 prezintă spectrele inducției magnetice pentru cazul releului discretizat fin și pentru cazul problemei TEAM 25. Rezultatele prezentate validează corectitudinea algoritmilor implementați de autoare în FAP.

Tabelele 3.3 și 3.4 conțin informații legate de rata de convergență. Timpul de calcul depinde de performanțele calculatorului utilizat și de eroarea impusă soluției numerice. De aceea o coloană specială din tabel conține valoarea erorii relative ($\|\delta I\|/\|I\|$ în cazul metodei polarizației și $\|\delta A\|/\|A\|$ în cazul metodei Newton). S-a notat cu PB^4 algoritmul polarizației cu iterații prin \mathbf{B} .

• Metoda polarizației

Se observă că, în majoritatea cazurilor studiate, metoda polarizației cu relaxare îmbunătățește timpul total de calcul (pentru același criteriu de oprire) al metodei polarizației chiar dacă timpul de calcul pe iterație este mai mare (cu 40 % ÷ 120 %) datorită calculării factorului de relaxare. Figura 3.23 prezintă variația cu factorul de relaxare a funcției de

⁴Picard-Banach

Metoda	eroare relativă	TEST 1			TEST 2		
		număr iterații	timp [sec]	energie [J]	număr iterații	timp [sec]	energie [J]
PB	10^{-2}	6	0.07	71.1814	40	2.23	1.9607
	10^{-3}	16	0.14	71.1750	90	4.60	2.5417
	10^{-4}	36	0.27	71.1744	145	6.64	2.6222
	10^{-5}	55	0.40	71.1744	253	9.59	2.6311
PB+relax	10^{-2}	3	0.05	71.1819	13	2.33	2.009
	10^{-3}	11	0.18	71.1744	29	3.03	2.5593
	10^{-4}	15	0.22	71.1744	45	3.76	2.6231
	10^{-5}	25	0.26	71.1744	77	4.99	2.6313
Newton	10^{-2}	Nu converge			3	0.65	2.6350
	10^{-3}	Nu converge			3	0.67	2.6351
	10^{-4}	Nu converge			5	1.04	2.6351
	10^{-5}	Nu converge			7	1.38	2.6351
MEGA		4	0.34	71.1803	4	0.38	2.6327

Tabelul 3.3: Timp de calcul - testele 1 și 2

minimizat $g(\omega)$ și a derivatei acesteia în cazul testului al doilea pentru obținerea primului factor de relaxare (care este 2.74).

Timpul de calcul se reduce cu cel puțin jumătate în cazurile în care criteriul de oprire este dur ($\|\delta I\|/\|I\| < 10^{-5}$). Pentru problemele extrem de simple (din punct de vedere al dimensiunii sistemului de rezolvat) de exemplul primul test micșorarea numărului de iterații nu duce la o reducere a timpului de calcul. Dimpotrivă, pentru primul test la o condiție de oprire de 10^{-3} timpul total crește. Acest test nu este însă relevant pentru aprecierea metodei. Pentru a fi rezolvate cu acuratețe problemele necesită discretizări mai fine, deci sisteme de rezolvat mai mari. De aceea, mai degrabă putem generaliza concluziile rezultate din testele 3 și 4 decât cele rezultate din primul test. În cazul testului 4 îmbunătățirea este spectaculoasă în cazul erorii de 10^{-4} , timpul necesar pentru atingerea acestui criteriu scăzând de la jumătate de oră în cazul procedurii fără relaxare la 8 minute în cazul

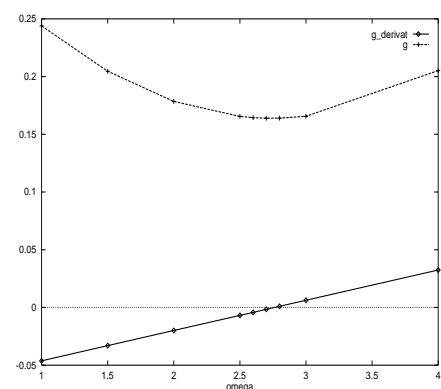


Figura 3.23: Funcția de minimizat și derivata ei în funcție de factorul de relaxare

Metoda	eroare relativă	TEST 3			TEST 4		
		număr iterații	timp [sec]	energie [J]	număr iterații	timp [sec]	energie [J]
PB	10^{-2}	49	58.74	2.3983	30	767	31.9413
	10^{-3}	122	106	3.5409	59	1350	38.6494
	10^{-4}	204	136	3.7170	344	1881	39.5924
	10^{-5}	633	264	3.7367	2352	3355	39.5983
PB+relax	10^{-2}	15	30.39	2.4101	11	226	33.5876
	10^{-3}	37	47.61	3.5457	21	379	39.025
	10^{-4}	59	56.70	3.7212	89	495	39.5925
	10^{-5}	135	81.15	3.7367	829	1204	39.5982
Newton	10^{-2}	6	73.91	3.7442	Nu converge		
	10^{-3}	13	126	3.7441			
	10^{-4}	20	186	3.7441			
	10^{-5}	27	271	3.7441			
MEGA		4	1.33	3.8626	5	16.07	39.4790

Tabelul 3.4: Timp de calcul - testele 3 și 4

folosirii relaxării (timpul scade de aproximativ 4 ori). Prin utilizarea relaxării la testul 3 viteza medie de calcul a crescut de 3.26 ori iar la testul 4 de 2.78 ori. În concluzie, utilizarea relaxării este recomandată în toate cazurile în care se aplică metoda iterațiilor simple.

O altă observație este aceea că timpul pe iterație nu este constant. Acest lucru se explică prin aceea că pentru rezolvarea sistemului liniar la fiecare iterație am folosit o procedură iterativă care folosește drept inițializare soluția găsită la iterația anterioară. Pe măsură ce iterațiile avansează, soluția se apropie de soluția exactă, inițializarea pentru algoritmul de rezolvare iterativ este din ce în ce mai bună și în consecință timpul necesar de rezolvare a sistemului liniar este din ce în ce mai mic.

Cât să fie de mare valoarea raportului $\|\delta I\|/\|I\|$? Pentru a avea o idee asupra acurateții rezolvării problemei tabelele 3.3 și 3.4 conțin și valorile energiei în model. Luând drept referință valoarea obținută cu MEGA observăm că o eroare de 1 % nu duce uneori (testele 3, 4) la rezultate prea bune ale energiei. De asemenea, a scădea eroarea de la 10^{-4} la 10^{-5} nu aduce o modificare substanțială a energiei. Mai mult, timpul necesar pentru a scădea cu încă un ordin de mărime este atât de mare încât nu merită efortul. De exemplu testul al patrulea, în cazul metodei fără relaxare programului îi trebuie jumătate de oră pentru a ajunge la o eroare de 10^{-4} și încă o jumătate de oră pentru a mai scădea cu încă un ordin de mărime. Scăderea ordinului de mărime a criteriului de oprire se face din ce în ce mai

greu. Tot în cazul testului 4, metoda fără relaxare, programul face 59 iterații pentru a ajunge la 10^{-3} în 22 minute și apoi mai face încă aproximativ 300 iterații pentru a ajunge la 10^{-4} . Pentru acestea îi mai trebuie încă 10 minute (am explicat mai sus de ce timpul pe iterație nu este constant).

Algoritmul polarizație converge în norma L_2 a polarizației. Criteriul de oprire este însă $\|\delta I\|/\|I\|$. Figurile 3.24 și 3.25 prezintă variația normei corecțiilor polarizațiilor în timpul procesului iterativ pentru testele 3 și 4 (în cazul unui criteriu de oprire de 10^{-4}) iar figurile 3.26 și 3.27 prezintă variația criteriului de oprire $\|\delta I\|/\|I\|$ pentru aceleași situații. Deși convergența algoritmului este demonstrată pentru $\|\delta I\|$ și nu pentru $\|\delta I\|/\|I\|$, se pare că și această din urmă mărime are o variație descrescătoare în timp, fiind un criteriu robust pentru oprirea algoritmului. El este un criteriu mai atrăgător decât $\|\delta I\|$ deoarece conține o mărime relativă și nu una absolută.

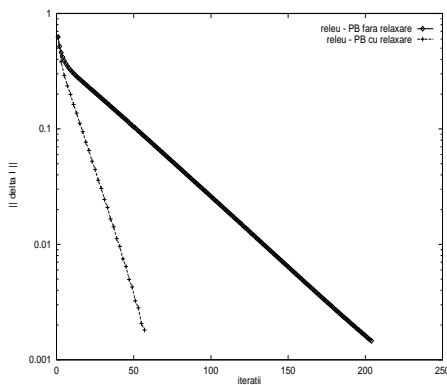


Figura 3.24: Test 3 - norma corecțiilor polarizațiilor $\|\delta I\|$ în metoda PB

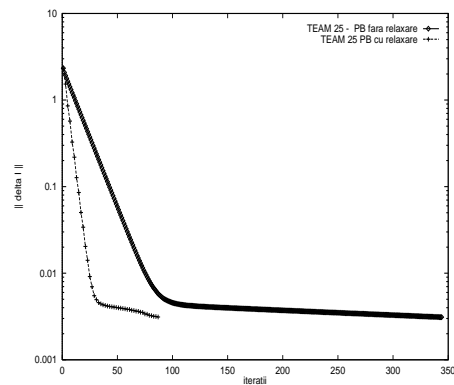


Figura 3.25: Test 4 - norma corecțiilor polarizațiilor $\|\delta I\|$ în metoda PB

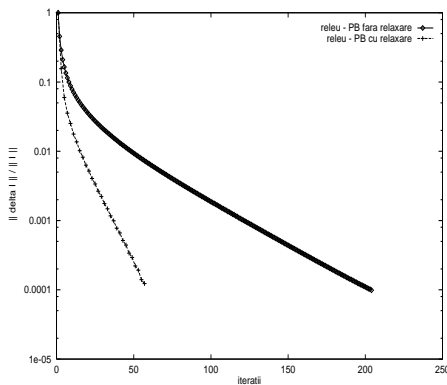


Figura 3.26: Test 3 - norma relativă a corecției $\|\delta I\|/\|I\|$ în metoda PB

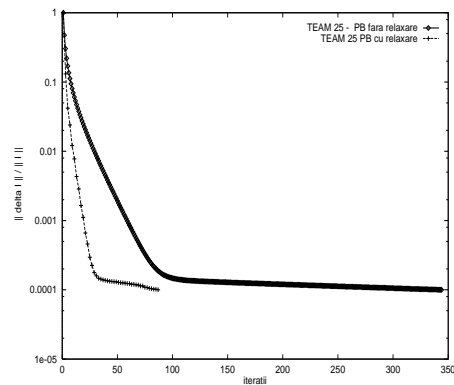


Figura 3.27: Test 4 - norma relativă a corecției $\|\delta I\|/\|I\|$ în metoda PB

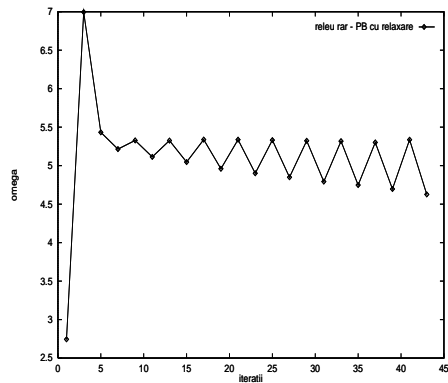


Figura 3.28: Test 2 - factorul de relaxare în metoda PB

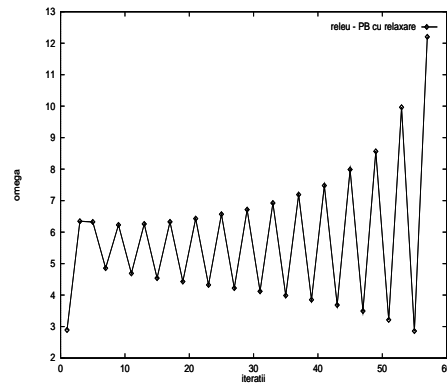


Figura 3.29: Test 3 - factorul de relaxare în metoda PB

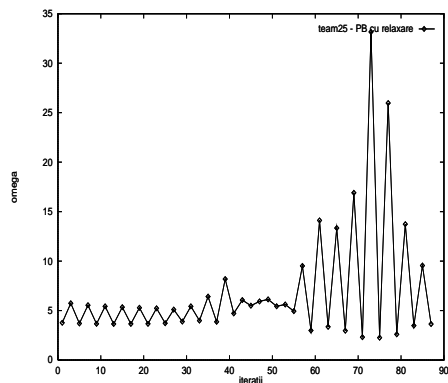


Figura 3.30: Test 4 - factorul de relaxare în metoda PB

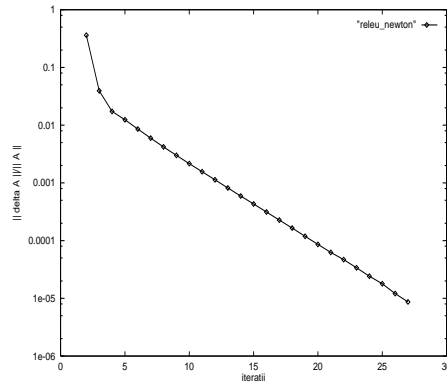


Figura 3.31: Test 3 - norma relativă a corecției $\|\delta A\|/\|A\|$ în metoda Newton

Figurile 3.28, 3.29 și 3.30 prezintă variația factorului de relaxare pentru testele 2, 3, și 4 (criteriu de oprire 10^{-4}). Ceea ce se observă în toate aceste figuri este variația oscilatorie a acestuia, precum și faptul că el poate lua valori spectaculoase (de exemplu 30).

În concluzie se constată că:

- Rata de convergență $-\delta \lg(\varepsilon)/\delta n$ este de $2 \div 3$ ori mai mare prin utilizarea relaxării (în anumite cazuri ajunge și de 6 ori mai mare);
- Pe parcursul iterațiilor rata de convergență scade la valori de $2 \div 6$ ori mai mici decât valorile inițiale (la testul 4 această modificare are loc practic brusc după 30 iterații).

• Metoda Newton

Metoda Newton a fost convergentă pentru două din cele patru teste.

Timpul de calcul pe iterație a fost mai mare la metoda Newton decât la metoda polarizației cu relaxare (de aproximativ 3 ori la testul 2 și de 17 ori la testul 3).

Dacă analizăm testul al treilea, la erori relative egale metoda Newton converge în timp CPU de circa 3 ori mai încet decât metoda polarizației cu relaxare chiar dacă numărul de iterații necesare este mult mai mic. Dacă analizăm însă și valorile energiei, observăm că aceasta este mai puțin sensibilă la criteriul de oprire al metodei Newton decât la criteriul de oprire al metodei polarizației. În consecință comparația algoritmilor polarizație și relaxare pentru valori egale ale criteriului de oprire devine lipsită de sens. Rezultă în acest moment necesitatea implementării unor indicatori sau estimatori de eroare potriviți mărimilor ce se cer calculate. Figura 3.31 prezintă variația normei relative a corecției în timpul iterațiilor Newton pentru testul al treilea. Pentru testul al doilea (mult mai mic din punct de vedere al numărului de necunoscute) metoda Newton este evident mult mai rapidă în orice situație, asigurând atingerea erorii de 10^{-3} într-un timp de 3.6 ori mai mic decât metoda polarizației cu relaxare.

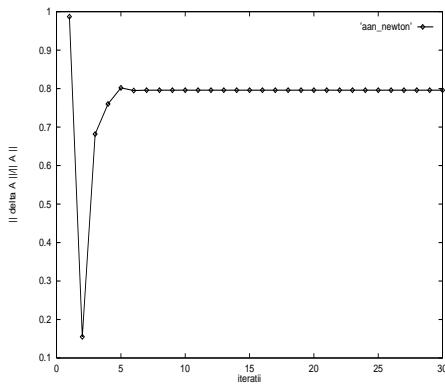


Figura 3.32: Test 1 - norma relativă a corecției $\|\delta A\|/\|A\|$ în metoda Newton

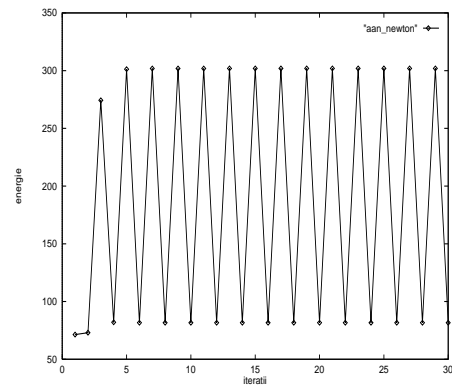


Figura 3.33: Test 1 - energia în timpul iterațiilor Newton

Metoda Newton nu converge pentru testele 1 și 4. În figura 3.32 se observă variația criteriului de oprire pentru primul test în cazul metodei Newton, iar figura 3.33 prezintă variația energiei în timpul aceluiași proces iterativ. Algoritmul oscilează și nu se poate opri natural. Rezultate similare se obțin și pentru testul 4 (figurile 3.34 și 3.35).

Dacă în cazul primului test variațiile energiei sunt foarte mari (150 %), în cazul testului 4 aceste variații sunt mici (4 ÷ 5 %). Unii autorii [107] folosesc drept criteriu de oprire relația

$$\left\| \frac{W_m^{i+1} - W_m^i}{W_m^i} \right\| \leq \varepsilon, \quad (3.130)$$

unde ε este o toleranță impusă, algoritmul oprindu-se atunci când relația 3.130 este valabilă trei iterații consecutive. Impunând o astfel de condiție de oprire și un $\varepsilon = 0.05$ algoritmul s-ar opri natural. Totuși, o astfel de condiție de oprire nu este foarte riguroasă, aceeași energie putând corespunde la mai multe configurații de câmp.

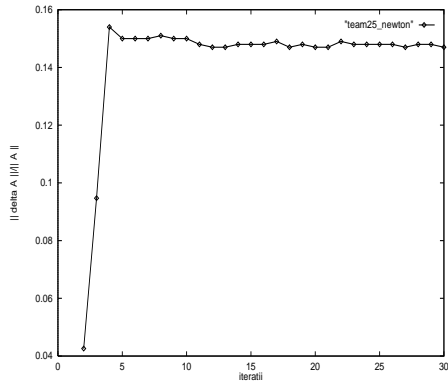


Figura 3.34: Test 4 - norma relativă a corecției $\|\delta A\|/\|A\|$ în metoda Newton

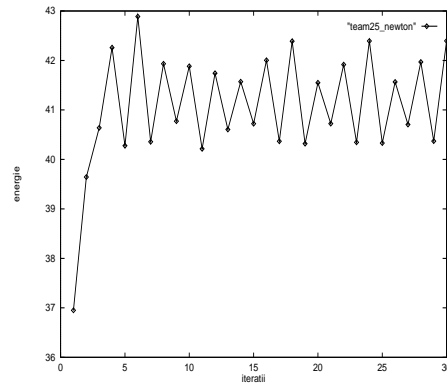


Figura 3.35: Test 4 - energia în timpul iterațiilor Newton

În consecință se poate afirma că rata de convergență pentru metoda Newton, atunci când ea converge, este de $10 \div 16$ ori mai mare decât a metodei polarizației cu relaxare. Și în acest caz se observă o micșorare cu $20\% \div 100\%$ a ratei de convergență pe parcursul iterațiilor.

De ce nu converg testele 1 și 4? De ce converg testele 2 și 3? Când am detaliat algoritmul Newton am văzut că avem nevoie de $\nu(B^2)$ și de $d\nu/dB^2$. În consecință avem nevoie de o interpolare a perechilor (B, H) date inițial și o extrapolare a acestei curbe. Extrapolarea am făcut-o astfel încât curba $B - H$ extrapolată să corespundă unei variații liniare de pantă μ_0 . Pentru porțiunea extrapolată materialul are deci o permeabilitate dinamică μ_0 . Dacă notăm cu B_{n-1}, H_{n-1} ultima pereche de puncte cunoscută, curba extrapolată va fi $B = \mu_0 H + B_{n-1} - \mu_0 H_{n-1}$. Rezultă că $\nu = H/B$ pentru porțiunea extrapolată va fi

$$\nu(B) = \frac{1}{\mu_0} - \frac{1}{B} \left(\frac{1}{\mu_0} B_{n-1} - H_{n-1} \right) \quad (3.131)$$

și se poate calcula ușor $d\nu/dB^2$. Pentru interpolare am folosit metoda Akima. În figurile 3.36 și 3.37 sunt prezentate curbele care rezultă din interpolarea și extrapolarea perechilor (B, H) .

Curba reluctivității pare rezonabilă, dar derivata ei are un salt important în zona de trecere de la interpolare la extrapolare. În momentul în care căutarea soluției se face în această zonă este de așteptat ca algoritmul să nu convergă ci să oscileze. Metoda Newton converge pentru cazurile 2 și 3 pentru că pentru ele nu au nevoie să facă extrapolarea curbei de material, în timp ce testele 1 și 4 au nevoie de așa ceva.

Interpolarea curbei reluctivității trebuie făcută cu grijă. Iată de exemplu în figura 3.38 cum interpolarea spline pentru punctele curbei $\nu - B^2$ are ca rezultat o curbă ce ia și valori negative fapt ce nu poate corespunde proprietății de material. Folosind această

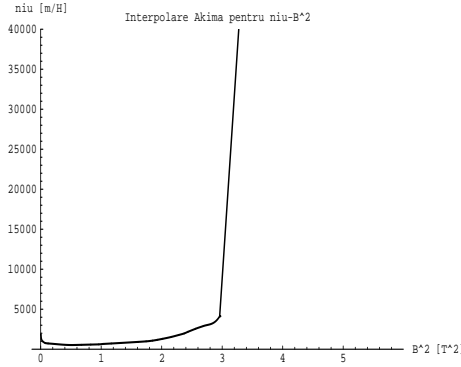


Figura 3.36: $\nu(B^2)$ pentru metoda Newton, interpolare Akima

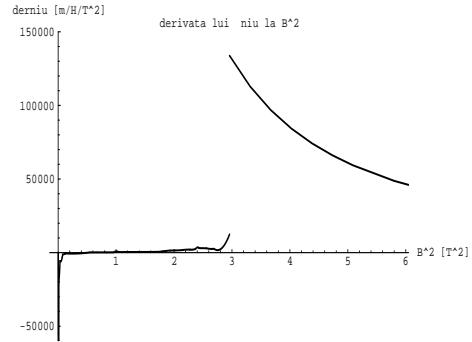


Figura 3.37: $d\nu/dB^2$ pentru Newton, interpolare Akima

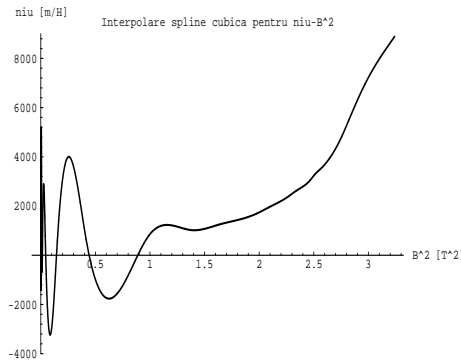


Figura 3.38: $\nu(B^2)$ pentru metoda Newton, interpolare spline

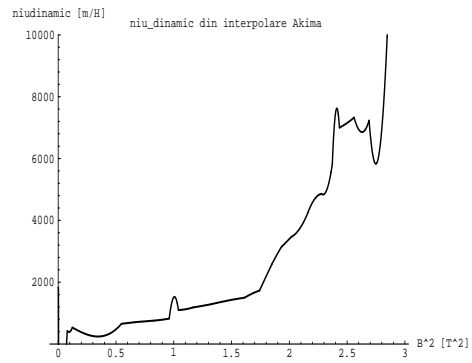


Figura 3.39: $\nu_{dinamic}$ așa cum rezultă din interpolarea Akima

curbă în locul celei din figura 3.36, testul al doilea nu a fost convergent.

În [144] se afirmă că pentru obținerea soluției sunt suficiente condițiile: funcțiile caracteristice de magnetizare $B(H)$ ale materialelor din problemă să fie monoton crescătoare, în timp ce prima derivată $\mu_d^{(H)} = dB/dH$ să fie monoton descrescătoare.

Mărimea $d\nu/dB^2$ conține în ea informații legate de permeabilitatea dinamică deoarece

$$\frac{d\nu}{dB^2} = \frac{1}{2B} \frac{d\nu}{dB} = \frac{1}{2B} \left(\frac{1}{B} \frac{dH}{dB} - \frac{H}{B^2} \right) = \frac{1}{2B} \frac{\nu_{dinamic} - \nu}{B}. \quad (3.132)$$

Graficul din figura 3.39 reprezintă reluctivitatea dinamică calculată cu formula 3.132 și cu rezultatele interpolării Akima (și curba de material a testului 4). Se observă că această mărime nu este monoton crescătoare. Punctele curbei de material al trebui rearanjate astfel încât interpolarea să fie satisfăcătoare.

3.2.4 Concluzii

În urma testelor numerice efectuate se poate ajunge la concluzia că metoda polarizației ridică mult mai puține probleme decât metoda Newton atât din punct de vedere al convergenței cât și din punctul de vedere al prelucrării curbei de material. Utilizarea relaxării optime conduce la îmbunătățirea acestui algoritm prin sporirea vitezei de calcul a soluției numerice de circa 3 ori, deci ea este puternic recomandată.

Algoritmul Newton standard nu este robust. Deoarece rezolvarea multor probleme eșuează cu algoritmul Newton, se întâlnesc în literatură algoritmi Newton-Raphson modificați, care folosesc subrelaxarea pentru a asigura convergența algoritmului [7, 52, 99, 113, 114], dar care evident conduc la micșorarea vitezei de convergență.

Pentru a obține algoritmi rapizi dar robusți (având convergența asigurată) se pare că cea mai bună metodă este de a folosi tehnica polialgoritmilor. Iterațiile inițiale se realizează cu o metodă garantat convergentă (de exemplu PB cu relaxare) iar în vecinătatea soluției se comută pe metoda Newton care este mai rapidă. Eventual dacă aceasta nu se dovedește convergentă se revine la metoda polarizației și ciclul se repetă. Singura problemă care trebuie rezolvată constă în alegerea criteriului de comutare între metode.

Pe baza rezultatelor numerice obținute în testele anterioare propunem drept criteriu de performanță a unui algoritm timpul prezumat pentru obținerea erorii impuse, obținut prin extrapolarea liniară a logaritmului corecției relative din ultimele două iterații. Pe baza acestui criteriu algoritmul următor comută "inteligent" între cele două metode.

Algoritm de comutare "inteligentă" între metodele polarizație cu relaxare și Newton:

1. $n_1 = 30$; număr maxim de iterații ale metodei PB la o trecere (conform rezultatelor de la testul 4)
2. $c_0 = 0.1$; impune scăderea erorii relative cu un ordin de mărime la fiecare trecere prin metoda PB, de asemenea stabilește numărul maxim de iterații ale metodei Newton = $n_1 c_0$
3. $n_2 = n_1 c_0$; număr maxim de iterații ale metodei Newton
4. $\text{eps}_0 = 10^{-3}$; eroare impusă
5. $\text{eps} = 1.0$; eroarea relativă inițială (la metoda PB, dacă polarizațiile se inițializează cu zero, norma relativă a corecției după prima iterație este 1)
6. $k = 1$; contor global iterații
7. $\text{nrmax} = \text{nr_noduri}$; număr maxim de iterații
8. **repetă**
 - 8.1. $\text{eps}_a = \text{eps}$
 - 8.2. $\text{eps}_1 = c_0 \text{eps}$
 - 8.3. $(t, n, \text{eps}) = \text{PBrelax}(n_1, \text{eps}_1)$; apelul metodei PB cu relaxare pentru

eroare eps și numărul maxim de iterații n_1 întoarce timpul de calcul, numărul de iterații efectuat și eroarea la care s-a oprit algoritmul

8.4. $k = k + n$

8.5. **dacă** ($\text{eps} < \text{eps}_0$) **atunci stop**

8.6. $\text{perf}_1 = t \cdot \lg(\text{eps}_0/\text{eps}_a)/\lg(\text{eps}/\text{eps}_a)$; estimează timpul perf_1 după care
PB ar atinge eroarea eps_0

8.7. $\text{eps}_a = \text{eps}$; noua eroare inițială

8.8. $(t, n, \text{eps}) = \mathbf{Newton}(n_2, \text{eps}_0)$; încearcă "finish" cu metoda Newton

8.9. $k = k + n$

8.10. **dacă** ($\text{eps} < \text{eps}_0$) **atunci stop**

8.11. $\text{perf}_2 = t \cdot \lg(\text{eps}_0/\text{eps}_a)/\lg(\text{eps}/\text{eps}_a)$; estimează timpul perf_2 după care
Newton ar atinge eroarea eps_0

8.12. **dacă** ($\text{perf}_2 < \text{perf}_1$) **atunci**

8.12.1. $n_3 = \text{minim}(n_2 + n \cdot \text{perf}_2/t, n_1)$; Newton cu mai multe iterații

8.12.2. $(t, n, \text{eps}) = \mathbf{Newton}(n_3, \text{eps}_0)$

8.12.3. $k = k + n$

până când (($\text{eps} < \text{eps}_0$) sau ($k > \text{nritmax}$))

Capitolul 4

Analiza sensibilităților și optimizarea dispozitivelor electromagnetice

Optimizarea dispozitivelor electromagnetice se reduce la găsirea extremelor unei funcții reale f numită *funcție obiectiv*, a cărei expresie este stabilită înaintea alegerii metodei de optimizare propriu-zise,

$$f(\text{"parametri"}) = \text{"expresie"}. \quad (4.1)$$

Parametrii problemei sunt mărimile care se cer găsite astfel încât să fie satisfăcute anumite obiective. Există două tipuri de abordări.

1. Regiuni omogene mari:

Una din abordări presupune cunoscută forma dispozitivului și poziția surselor de câmp (se poate desena o schiță a dispozitivului din care însă lipsesc valorile numerice ale anumitor cote și poate și valorile surselor de câmp). Într-o astfel de abordare parametrii problemei pot fi:

- dimensiuni geometrice principale ale obiectelor geometrice ale dispozitivului (de exemplu raze de cercuri, semiaxe de elipse, diferite distanțe) care fixează "interfețele" dintre materiale cu proprietăți diferite; se presupune deci că sunt cunoscute familiile de curbe care descriu "interfețele", optimizarea găsindu-le poziția optimă în spațiu.
- valorile surselor de câmp pentru care trebuie cunoscut tipul de variație temporală (pentru a putea stabili metoda de analiză potrivită).
- valorile parametrilor de material (presupus liniar) sau curbele neliniare ce descriu caracteristicile de material.

2. Regiuni omogene foarte mici, de tip celule:

O a doua abordare nu presupune cunoscută forma dispozitivului ci doar extinderea

lui maximală în spațiu. În această situație necunoscutele problemei sunt în număr infinit: în fiecare punct din spațiu proprietatea de material și sursa de câmp. Pentru a avea un număr finit de parametri domeniul maximal se împarte într-un număr finit de celule pe care proprietățile de material și sursele de câmp se presupun omogene. Și în această situație trebuie făcută o presupunere despre variația temporală a surselor de câmp. O astfel de abordare aparține mai degrabă de problema sintezei dispozitivelor.

Expresia din membrul drept al relației (4.1) poate conține o varietate de mărimi de tipul:

- dimensiuni geometrice ale problemei (pot apare de exemplu din considerații legate de cantitatea sau prețul anumitor materiale);
- mărimi locale ce caracterizează câmpul electromagnetic (de exemplu inducții magnetice în anumite puncte);
- mărimi globale ce caracterizează câmpul electromagnetic (de exemplu fluxuri, energii).
- mărimi locale sau globale ce caracterizează efecte ale câmpului electromagnetic (temperaturi, viteze, debite, cupluri, forțe etc.)

Ultimele trei categorii sunt de obicei mărimi derivate din mărimea principală a problemei (cea în care este formulată problema de câmp pentru rezolvare). Mărimea principală depinde la rândul ei de caracteristicile geometrice, de material și de sursele problemei, o parte dintre acestea fiind cunoscute iar altele fiind parametri a căror valori optime sunt căutate. Având în vedere aceste considerente, putem scrie relația (4.1) mai explicit astfel

$$f(p) = F(p', d'(A(p, d))). \quad (4.2)$$

→ p reprezintă vectorul *parametrilor de optimizat*, cei a căror valoare se cere determinată astfel încât să fie satisfăcute obiectivele problemei. Ei pot fi dimensiuni geometrice, constante de material sau valori ce identifică sursele de câmp.

→ p' sunt *parametri de optimizare care apar explicit în expresia F a funcției obiectiv*.

→ d reprezintă *datele problemei de câmp*, cele prin a căror specificare problema de câmp este bine formulată.

→ A este *mărimea principală a problemei*, cea în care este formulată problema de câmp pentru a fi rezolvată (de exemplu A este potențialul magnetic vector).

→ $d'(A)$ reprezintă *mărimile derivate din mărimea principală A* (de exemplu inducția magnetică, energia).

Dacă notăm cu \mathcal{P} mulțimea parametrilor de optimizare (elementele acestei mulțimi fiind componentele vectorului p), cu \mathcal{P}' mulțimea parametrilor de optimizare care apar explicit în expresia F și cu \mathcal{D} mulțimea datelor problemei de câmp atunci $\mathcal{P}' \subset \mathcal{P} \subset \mathcal{D}$.

Funcția f din (4.2) este în cele din urmă o funcție reală definită într-un spațiu multidimensional. Folosirea unei metode de optimizare deterministe de ordin superior presupune

cunoașterea gradientului funcției, deci a derivatelor ("senzitivităților") funcției în raport cu parametrii de optimizat.

O posibilitate de a calcula senzitivitățile funcției de cost în raport cu variabilele de proiectare este aceea de a folosi metode de tip diferențe finite. Dacă pentru o evaluare a funcției de cost este necesară o rezolvare a unei probleme de câmp, aceasta înseamnă că pentru calculul unei derivate parțiale sunt necesare mai multe rezolvări de probleme de câmp. Metoda este imprecisă datorită tuturor problemelor care apar la derivarea numerică (calculul pasului optim ar face ca metoda să fie costisitoare din punct de vedere al timpului de calcul). De aceea, în cele ce urmează vom prezenta alte tipuri de metode pentru calculul senzitivităților.

Aplicând derivatele în membrul drept din (4.2) rezultă (de exemplu aplicând reguli de derivare înlănțuită) că vom avea nevoie de calculul derivatelor ("senzitivităților") mărimilor de câmp (locale sau globale) în raport cu variabilele de proiectare.

În acest capitol vom considera primul tip de abordare în care parametrii pot fi dimensiuni sau valori de surse de câmp. Materialele (liniare sau neliniare) le vom presupune cunoscute.

4.1 Senzitivitățile mărimii principale față de parametrii problemei de optimizare

Vom nota cu A mărimea (necunoscută) principală a problemei (cea care rezultă din rezolvarea problemei de câmp), definită pe domeniul problemei (fiind astfel necunoscute o infinitate de valori scalare) și cu p_k ($k = 1, \dots, m$) parametrii problemei de optimizare (componentele vectorului p). Problema tratată în acest paragraf este calculul mărimii $\frac{\partial A}{\partial p_k}$. În cele ce urmează vom renota componenta p_k cu p și nu vom mai face referiri la vectorul parametrilor problemei de optimizare.

4.1.1 Folosirea unei probleme adjuncte

Vom considera cazul regimului magnetic staționar, în care mărimile de câmp satisfac

$$\begin{aligned} \operatorname{div} \mathbf{B} &= 0 \quad \Longrightarrow \quad \mathbf{B} = \operatorname{rot} \mathbf{A}, \\ \operatorname{rot} \mathbf{H} &= \mathbf{J}. \end{aligned}$$

În aceste condiții rezultă că

$$\begin{aligned} \operatorname{div} (\mathbf{A} \times \mathbf{H}) &= \mathbf{H} \cdot \mathbf{B} - \mathbf{A} \cdot \mathbf{J} \quad \Longrightarrow \\ \int_{\partial\Omega} (\mathbf{A} \times \mathbf{H}) \cdot \mathbf{n} \, dA &= \int_{\Omega} (\mathbf{H} \cdot \mathbf{B} - \mathbf{A} \cdot \mathbf{J}) \, d\Omega. \end{aligned}$$

Dacă frontiera $\partial\Omega$ este suficient de departe de sursele de câmp, putem presupune că pe ea intensitatea câmpului magnetic este nulă și în consecință rezultă că

$$\int_{\Omega} \mathbf{H} \cdot \mathbf{B} \, d\Omega = \int_{\Omega} \mathbf{A} \cdot \mathbf{J} \, d\Omega. \quad (4.3)$$

Relația (4.3) este valabilă indiferent de legătura dintre câmpurile de vectori \mathbf{B} și \mathbf{H} . Esențial este ca $\text{rot } \mathbf{H} = \mathbf{J}$ și $\mathbf{B} = \text{rot } \mathbf{A}$. În aceste condiții relația este valabilă și dacă \mathbf{H} și \mathbf{J} corespund unei probleme iar \mathbf{B} și \mathbf{A} corespund altei probleme (de regim magnetic staționar).

Vom considera trei probleme de regim magnetic staționar. Prima problemă este problema inițială, în care materialele sunt neliniare (dar izotrope):

$$\mathbf{B} = \text{rot } \mathbf{A} \quad , \quad \text{rot } \mathbf{H} = \mathbf{J} \quad , \quad \mathbf{B} = \mathbf{f}(\mathbf{H}).$$

A doua problemă reprezintă prima problemă cu mici perturbații:

$$\mathbf{B} + \delta\mathbf{B} = \text{rot } (\mathbf{A} + \delta\mathbf{A}) \quad , \quad \text{rot } (\mathbf{H} + \delta\mathbf{H}) = \mathbf{J} + \delta\mathbf{J}.$$

Rezultă că relațiile satisfăcute de perturbații sunt:

$$\delta\mathbf{B} = \text{rot } \delta\mathbf{A} \quad , \quad \text{rot } \delta\mathbf{H} = \delta\mathbf{J}.$$

A treia problemă este o problemă numită *adjunctă*:

$$\hat{\mathbf{B}} = \text{rot } \hat{\mathbf{A}} \quad , \quad \text{rot } \hat{\mathbf{H}} = \hat{\mathbf{J}}$$

Scriind relații de tipul (4.3) pentru perechile $(\hat{\mathbf{B}}, \hat{\mathbf{A}})$ și $(\delta\mathbf{H}, \delta\mathbf{J})$ și apoi pentru $(\delta\mathbf{B}, \delta\mathbf{A})$ și $(\hat{\mathbf{H}}, \hat{\mathbf{J}})$ rezultă relațiile:

$$\int_{\Omega} \hat{\mathbf{B}} \cdot \delta\mathbf{H} \, d\Omega = \int_{\Omega} \hat{\mathbf{A}} \cdot \delta\mathbf{J} \, d\Omega, \quad (4.4)$$

$$\int_{\Omega} \hat{\mathbf{H}} \cdot \delta\mathbf{B} \, d\Omega = \int_{\Omega} \hat{\mathbf{J}} \cdot \delta\mathbf{A} \, d\Omega. \quad (4.5)$$

Scăzând relațiile (4.4) și (4.5) rezultă

$$\int_{\Omega} (\hat{\mathbf{B}} \cdot \delta\mathbf{H} - \hat{\mathbf{H}} \cdot \delta\mathbf{B}) \, d\Omega = \int_{\Omega} (\hat{\mathbf{A}} \cdot \delta\mathbf{J} - \hat{\mathbf{J}} \cdot \delta\mathbf{A}) \, d\Omega \quad (4.6)$$

Fie p unul din parametrii de optimizare. Privind domeniul problemei ca fiind format din celule, fiecare celulă este caracterizată de două mărimi: $\mathbf{f}(\mathbf{H}, p)$ și $\mathbf{J}(p)$.

Folosind derivata Frechet a funcției \mathbf{f} , variația $\delta\mathbf{B}$ se poate exprima astfel:

$$\delta\mathbf{B} = \frac{d\mathbf{f}}{d\mathbf{H}} \delta\mathbf{H} + \frac{\partial\mathbf{f}}{\partial p} \delta p, \quad (4.7)$$

$$\delta\mathbf{J} = \frac{d\mathbf{J}}{dp} \delta p. \quad (4.8)$$

Înlocuind relațiile (4.7) și (4.8) în (4.6) rezultă că

$$\int_{\Omega} (\hat{\mathbf{B}} \cdot \delta \mathbf{H} - \hat{\mathbf{H}} \cdot \frac{d\mathbf{f}}{d\mathbf{H}} \delta \mathbf{H} - \hat{\mathbf{H}} \cdot \frac{\partial \mathbf{f}}{\partial p} \delta p) d\Omega = \int_{\Omega} (\hat{\mathbf{A}} \cdot \frac{d\mathbf{J}}{dp} \delta p - \hat{\mathbf{J}} \cdot \delta A) d\Omega. \quad (4.9)$$

Dacă alegem $\hat{\mathbf{J}} = \mathbf{J}_0 \delta(P, Q)$ (sursa problemei adjuncte concentrată în punctul în care se dorește calculul senzitivității mărimii principale \mathbf{A} în raport cu variabila de optimizat p) și $\hat{\mathbf{B}} = \frac{d\mathbf{f}}{d\mathbf{H}} \hat{\mathbf{H}}$ (materialele în problema adjunctă sunt liniare și au permeabilitatea magnetică egală cu permeabilitatea dinamică din problema inițială) atunci relația (4.9) devine

$$\mathbf{J}_0 \cdot \delta \mathbf{A} = \int_{\Omega} \hat{\mathbf{A}} \cdot \frac{d\mathbf{J}}{dp} \delta p d\Omega + \int_{\Omega} \hat{\mathbf{H}} \cdot \frac{\partial \mathbf{f}}{\partial p} \delta p d\Omega, \quad (4.10)$$

și pentru o problemă plan-paralelă rezultă că

$$\frac{\partial A}{\partial p} = J_0^{-1} \left(\int_{\Omega} \hat{A} \frac{dJ}{dp} d\Omega + \int_{\Omega} \hat{\mathbf{H}} \cdot \frac{\partial \mathbf{f}}{\partial p} d\Omega \right). \quad (4.11)$$

Dacă p este un parametru care reprezintă o dimensiune geometrică, atunci derivatele în raport cu p sunt diferite de zero pe interfețele parametrizate de p . Dacă p este o sursă de câmp atunci $\frac{\partial \mathbf{f}}{\partial p} = 0$ dar $\frac{d\mathbf{J}}{dp}$ este diferită de zero în regiunea parametrizată de p .

Dacă p este un parametru geometric, atunci relația (4.11) se reduce la integrale pe interfețele care se modifică, sub integrale apărând saltul lui J și saltul mărimii $\mathbf{B} \cdot \hat{\mathbf{H}}$ în sensul creșterii parametrului p .

Relația (4.10) este valabilă pentru o alegere corespunzătoare a problemei adjuncte. Aceasta este o problemă liniară, cu o anumită distribuție a surselor de câmp. Sursa de câmp a problemei adjuncte este de tip distribuție, plasată în punctul în care se dorește calculul senzitivității potențialului vector. O astfel de sursă are sens numai pentru problemele plan-paralele sau axisimetrice.

4.1.2 Exploatarea metodei numerice de rezolvare

În cazul folosirii unei metode numerice numărul de valori scalare necunoscute este finit. Aceste necunoscute le vom nota cu A_i ($i = 1, \dots, n$) și le vom numi necunoscute principale. Vom nota cu A vectorul n -dimensional având drept componente necunoscutele A_i . Prin utilizarea unei metode numerice, vectorul necunoscutelor principale se obține prin rezolvarea unui sistem de ecuații algebrice

$$SA = T, \quad (4.12)$$

unde S este o matrice pătrată de dimensiuni $n \times n$ iar T este vectorul de dimensiune n al termenilor liberi (vom nota componentele sale cu t_i). Este evident că pentru asamblarea matricei S și a vectorului T sunt folosite informații despre geometria problemei,

materialele și sursele de câmp. Elementele lor vor depinde în consecință de parametrii de optimizat p . Derivând fiecare ecuație a sistemului (4.12) și notând cu S' matricea de elemente s'_{ij} , cu A' vectorul de elemente A'_i și cu T' vectorul de elemente t'_i unde:

$$s'_{ij} = \frac{\partial s_{ij}}{\partial p_k}, \quad A'_i = \frac{\partial A_i}{\partial p_k}, \quad t'_i = \frac{\partial t_i}{\partial p_k}, \quad (4.13)$$

rezultă că

$$\begin{aligned} S'A + SA' &= T' \quad \implies \\ SA' &= T' - S'A. \end{aligned} \quad (4.14)$$

Pentru calculul celor n derivate în raport cu p_k (cu k fixat) este necesară rezolvarea sistemului (4.14). Pentru calculul tuturor derivatelor este necesară rezolvarea a m sisteme de tipul (4.14). Asamblarea celor m matrice S' și a celor m vectori T' se poate face simultan cu asamblarea matricei S și a vectorului T . Metoda de calcul al sensibilităților necunoscutelor principale față de parametrii de optimizare este incorporată în metoda numerică de rezolvare a problemei de câmp. Se observă de asemenea că sistemele (4.14) și (4.12) au aceeași matrice a coeficienților, rezultând că în cazul folosirii unei metode directe de rezolvare factorizarea matricei se face o singură dată.

Necunoscuta principală într-un punct oarecare al domeniului se va exprima ca fiind

$$A = g(A_1, \dots, A_n), \quad (4.15)$$

și rezultă că

$$\frac{\partial A}{\partial p_k} = \sum_{i=1}^n \frac{\partial g}{\partial A_i} \frac{\partial A_i}{\partial p_k}. \quad (4.16)$$

Funcția g este depinde liniar de A_1, \dots, A_n și, în consecință, derivatele $\frac{\partial g}{\partial A_i}$ sunt ușor de calculat.

În cazul unei probleme neliniare, calculul necunoscutei principale A nu se face prin rezolvarea unui singur sistem de tipul (4.12) ci a unei succesiuni de astfel de sisteme. Exploatarea metodei numerice se poate face și în acest caz, pentru detalierea calcului trebuie însă precizată metoda de tratare a neliniarității folosită, modelul matematic și numeric folosit.

• Integrarea calcului sensibilităților în metoda polarizației aplicată modelului diferențial tratat cu metoda elementelor finite

Conform algoritmului detaliat la paragraful 3.1.2, la fiecare iterație neliniară se rezolvă sistemul

$$SA = T_1 + T_2. \quad (4.17)$$

Matricea S și vectorul T_1 sunt întotdeauna aceleași. Vectorul T_2 se recalculează la fiecare iterație. Matricea S depinde exclusiv de rețeaua de discretizare și de permeabilitatea

magnetică. Vectorul T_1 depinde de rețeaua de discretizare și de sursele de câmp iar vectorul T_2 depinde de polarizațiile de la iterația respectivă, de rețeaua de discretizare și de material. De aceea, dacă S' și T'_1 pot fi asamblate simultan cu asamblarea matricei S și a vectorului T_1 , înainte de începerea iterațiilor neliniare, vectorul T'_2 este asamblat după ieșirea din ciclul iterațiilor neliniare.

◇ *Elementele matricei S'*

Dacă parametrul p este o sursă de câmp atunci elementele acestei matrice sunt zero. Să presupunem că p este un parametru geometric. Atunci

$$\frac{ds_{ij}}{dp} = \sum_{k=1}^N \left(\frac{\partial s_{ij}}{\partial x_k} \frac{\partial x_k}{\partial p} + \frac{\partial s_{ij}}{\partial y_k} \frac{\partial y_k}{\partial p} \right). \quad (4.18)$$

Cum s_{ij} este diferit de zero doar pentru nodurile i și j care aparțin unui același element, rezultă că asamblarea elementelor $\frac{ds_{ij}}{dp}$ se poate face în același ciclu ce parcurge elementele în care se face asamblarea lui s_{ij} . Fiecare element e poate contribui cu maximum 9 termeni de tipul

$$\sum_{k=1}^3 \left(\frac{\partial s_{ij}}{\partial x_k} \frac{\partial x_k}{\partial p} + \frac{\partial s_{ij}}{\partial y_k} \frac{\partial y_k}{\partial p} \right),$$

unde $i = 1, 2, 3$, $j = 1, 2, 3$ iar x_k, y_k sunt coordonatele nodurilor triunghiului e . Se observă că are sens să se calculeze derivatele $\frac{\partial s_{ij}}{\partial x_k}$ și $\frac{\partial s_{ij}}{\partial y_k}$ doar pentru elementele pentru care $\frac{\partial x_k}{\partial p} \neq 0$ sau $\frac{\partial y_k}{\partial p} \neq 0$. Acestea sunt elementele care au cel puțin un nod pe interfața generată de p .

În consecință elementele care contribuie cu termeni nenuli la matricea S' (în cazul în care parametrul p reprezintă o dimensiune geometrică) sunt elementele care ating interfața generată de parametru. Matricea S' este astfel și mai rară decât matricea S .

Vom detalia contribuțiile elementului e la matricea S' și anume termenii $s_{ii}^{(e)'}$, $s_{ij}^{(e)'}$, $s_{ik}^{(e)'}$ unde elementul e are nodurile (ordonate în sens trigonometric) de coordonate (x_i, y_i) , (x_j, y_j) și (x_k, y_k) . Vom nota aria acestui triunghi cu Δ , deci

$$2\Delta = x_j y_k - x_k y_j + x_k y_i - x_i y_k + x_i y_j - x_j y_i.$$

Contribuțiile corespunzătoare acestui element (pentru nodul i) în matricea S sunt

$$s_{ii}^{(e)} = \frac{b_i^2 + c_i^2}{4\mu\Delta}, \quad s_{ij}^{(e)} = \frac{b_i b_j + c_i c_j}{4\mu\Delta}, \quad s_{ik}^{(e)} = \frac{b_i b_k + c_i c_k}{4\mu\Delta},$$

unde am notat

$$\begin{aligned} b_i &= y_j - y_k, & b_j &= y_k - y_i, & b_k &= y_i - y_j, \\ c_i &= x_k - x_j, & c_j &= x_i - x_k, & c_k &= x_j - x_i. \end{aligned}$$

	$\frac{\partial s_{ii}}{\partial x_i}$	$\frac{\partial s_{ij}}{\partial x_j}$	$\frac{\partial s_{ik}}{\partial x_k}$
$\frac{\partial}{\partial x_i}$	$-\frac{b_i(b_i^2+c_i^2)}{8\mu\Delta^2}$	$\frac{2c_i\Delta-b_i(b_ib_j+c_ic_j)}{8\mu\Delta^2}$	$\frac{-2c_i\Delta-b_i(b_ib_k+c_ic_k)}{8\mu\Delta^2}$
$\frac{\partial}{\partial x_j}$	$\frac{-4c_i\Delta-b_j(b_i^2+c_i^2)}{8\mu\Delta^2}$	$\frac{-2c_j\Delta-b_j(b_ib_j+c_ic_j)}{8\mu\Delta^2}$	$\frac{2(c_i-c_k)\Delta-b_j(b_ib_k+c_ic_k)}{8\mu\Delta^2}$
$\frac{\partial}{\partial x_k}$	$\frac{4c_i\Delta-b_k(b_i^2+c_i^2)}{8\mu\Delta^2}$	$\frac{2(c_j-c_i)\Delta-b_k(b_ib_j+c_ic_j)}{8\mu\Delta^2}$	$\frac{2c_k\Delta-b_k(b_ib_k+c_ic_k)}{8\mu\Delta^2}$
$\frac{\partial}{\partial y_i}$	$-\frac{c_i(b_i^2+c_i^2)}{8\mu\Delta^2}$	$\frac{-2b_i\Delta-c_i(b_ib_j+c_ic_j)}{8\mu\Delta^2}$	$\frac{2b_i\Delta-c_i(b_ib_k+c_ic_k)}{8\mu\Delta^2}$
$\frac{\partial}{\partial y_j}$	$\frac{4b_i\Delta-c_j(b_i^2+c_i^2)}{8\mu\Delta^2}$	$\frac{2b_j\Delta-c_j(b_ib_j+c_ic_j)}{8\mu\Delta^2}$	$\frac{2(b_k-b_i)\Delta-c_j(b_ib_k+c_ic_k)}{8\mu\Delta^2}$
$\frac{\partial}{\partial y_k}$	$\frac{-4b_i\Delta-c_k(b_i^2+c_i^2)}{8\mu\Delta^2}$	$\frac{2(b_i-b_j)\Delta-c_k(b_ib_j+c_ic_j)}{8\mu\Delta^2}$	$\frac{-2b_k\Delta-c_k(b_ib_k+c_ic_k)}{8\mu\Delta^2}$

Tabelul 4.1: $\frac{\partial s_{ij}}{\partial x_k}$ și $\frac{\partial s_{ij}}{\partial y_k}$, unde i, j, k sunt nodurile elementului e .

Atunci

$$s_{**}^{(e)'} = \frac{\partial s_{**}}{\partial x_i} \frac{\partial x_i}{\partial p} + \frac{\partial s_{**}}{\partial x_j} \frac{\partial x_j}{\partial p} + \frac{\partial s_{**}}{\partial x_k} \frac{\partial x_k}{\partial p} + \frac{\partial s_{**}}{\partial y_i} \frac{\partial y_i}{\partial p} + \frac{\partial s_{**}}{\partial y_j} \frac{\partial y_j}{\partial p} + \frac{\partial s_{**}}{\partial y_k} \frac{\partial y_k}{\partial p}, \quad (4.19)$$

unde în loc de $*$ poate fi i, j sau k . Derivatele din s_{**} sunt trecute detaliate în tabelul 4.1.

◇ *Elementele vectorului T_1'*

Pentru fiecare element e și pentru fiecare nod al său i contribuția la vectorul T_1 este¹

$$t1_i^{(e)} = \frac{J\Delta}{3}.$$

Formulele de derivare sunt simple. Dacă parametrul p este J atunci $t1_i^{(e)'} = \Delta/3$ iar dacă parametrul p este o dimensiune geometrică atunci $t1_i^{(e)'} = J\Delta'/3$, unde

$$\begin{aligned} \Delta' &= \frac{\partial \Delta}{\partial x_i} \frac{\partial x_i}{\partial p} + \frac{\partial \Delta}{\partial x_j} \frac{\partial x_j}{\partial p} + \frac{\partial \Delta}{\partial x_k} \frac{\partial x_k}{\partial p} + \frac{\partial \Delta}{\partial y_i} \frac{\partial y_i}{\partial p} + \frac{\partial \Delta}{\partial y_j} \frac{\partial y_j}{\partial p} + \frac{\partial \Delta}{\partial y_k} \frac{\partial y_k}{\partial p} = \\ &= \frac{1}{2} \left(b_i \frac{\partial x_i}{\partial p} + b_j \frac{\partial x_j}{\partial p} + b_k \frac{\partial x_k}{\partial p} + c_i \frac{\partial y_i}{\partial p} + c_j \frac{\partial y_j}{\partial p} + c_k \frac{\partial y_k}{\partial p} \right). \end{aligned} \quad (4.20)$$

◇ *Elementele vectorului T_2'*

Pentru fiecare element e și pentru fiecare nod al său i contribuția este

$$t2_i^{(e)} = -\frac{1}{\mu} \left(\frac{I_y b_i}{2} - \frac{I_x c_i}{2} \right).$$

Și în acest caz formula de derivare este simplă în cazul în care p este un parametru geometric

$$t2_i^{(e)'} = -\frac{1}{2\mu} \left(I_x \frac{\partial x_j}{\partial p} - I_x \frac{\partial x_k}{\partial p} + I_y \frac{\partial y_j}{\partial p} - I_y \frac{\partial y_k}{\partial p} \right). \quad (4.21)$$

¹Am presupus că parametrii problemei afectează interiorul domeniului și nu frontiera lui și de aceea nu am mai discutat (derivat) contribuțiile frontierei.

Algoritmul metodei polarizației

- regim magnetic staționar, 2D, XY, MEF, A, iterația prin B -

în care este înglobat calculul sensibilităților potențialului magnetic vector în raport cu m parametri de optimizat $p_k (k = 1, \dots, m)$ care reprezintă dimensiuni geometrice

1. citește datele problemei:

- geometria;

- datele de material;

- sursele (\mathbf{J});

- condițiile de frontieră:

(a) noduri în care se dau valorile lui A ;

(b) segmente pe care se dă \mathbf{H}_t (notăm $\frac{\partial A}{\partial n} = g$, funcție ce se poate deduce din \mathbf{H}_t)

2. inițializează \mathbf{I}

pentru $e \in \{ \text{toate elementele} \}$

$$\mathbf{I}_e = 0$$

3. stabilește permeabilitatea de calcul

pentru $e \in \{ \text{toate elementele} \}$

dacă elementul este liniar **atunci** $\mu_e = \mu_{\text{liniar}}$

dacă elementul este neliniar **atunci** $\mu_e = 2/(\nu_{\text{max}} + \nu_{\text{min}})$ (contractie optimă)

4. assemblează S și T_1 simultan cu m matrice $S'_{(p)}$ și m vectori $T1'_{(p)}$, $p = 1, \dots, m$

pentru $e \in \{ \text{toate elementele} \}$ (e de noduri i, j, k)

pentru fiecare nod $e \in e$ (fie nodul i)

$$\text{temp} = 4\Delta\mu$$

$$s_{ii}^{(e)} = (b_i^2 + c_i^2)/\text{temp}$$

$$s_{ij}^{(e)} = (b_i b_j + c_i c_j)/\text{temp}$$

$$s_{ik}^{(e)} = (b_i b_k + c_i c_k)/\text{temp}$$

$$t_{1i}^{(e)} = J\Delta/3$$

pentru $p = 1, m$; contribuții la S' și $T1'$

$$s_{(p)ii}^{(e')} = \dots$$

$$s_{(p)ij}^{(e')} = \dots$$

$$s_{(p)ik}^{(e')} = \dots \text{ (conform (4.19))}$$

$$t_{(p)i}^{(e')} = \frac{J\Delta'}{3}$$

pentru toate segmentele pe care se impune \mathbf{H}_t (fie i, j capetele):

$$t_{1i} = t_{1i} + \frac{1}{\mu} g \frac{l_{iat}}{2}$$

$$t_{1j} = t_{1j} + \frac{1}{\mu} g \frac{l_{iat}}{2}$$

itnelin = 1; contor iterații neliniare

5. repetă

5.1. anulează elementele lui T_2

pentru $i \in \{ \text{toate nodurile} \}$

$$t_{2i} = 0$$

5.2. assemblează T_2

dacă $\text{itnelin} \neq 1$ **atunci**

pentru $e \in \{ \text{toate elementele} \}$ (e de noduri i, j, k)

pentru fiecare nod $\in e$ (fie nodul i)

$$t_{2i}^{(e)} = -\frac{1}{\mu} \left(\frac{I_y b_i}{2} - \frac{I_x c_i}{2} \right)$$

5.3. calculează câmpul pentru problema liniară

rezolvă $SA = T_1 + T_2$

Obs: aici se impun valorile lui A în nodurile esențiale conform procedelor cunoscute (vezi de exemplu [77])

5.4. corectează polarizația

pentru $e \in \{ \text{toate elementele} \}$ (e de noduri i, j, k)

calculează \mathbf{B} :

$$\begin{aligned} \mathbf{B} &= \frac{\partial A}{\partial y} \mathbf{i} - \frac{\partial A}{\partial x} \mathbf{j} = \mathbf{i} \left(\sum_{j \in e} A_j \frac{\partial \Psi_j}{\partial y} \right) - \mathbf{j} \left(\sum_{j \in e} A_j \frac{\partial \Psi_j}{\partial x} \right) = \\ &= \mathbf{i} \left(\sum_{j \in e} A_j \frac{c_j}{2\Delta} \right) - \mathbf{j} \left(\sum_{j \in e} A_j \frac{b_j}{2\Delta} \right) \end{aligned}$$

calculează \mathbf{H} : $\mathbf{H} = \mathbf{H}(\mathbf{B})$

corectează \mathbf{I} : $\mathbf{I}_e = \mathbf{B} - \mu \mathbf{H}$

(Obs: pentru elementele liniare, $\mathbf{I} = 0$ întotdeauna)

5.5. $\text{itnelin} = \text{itnelin} + 1$

pâna când ($\|\delta I\| < \text{valoare impusă}$)

6. anulează elementele celor m vectori $T2'_{(p)}$

pentru $p = 1, m$

pentru $i \in \{ \text{toate nodurile} \}$

$$t2'_{(p)i} = 0$$

7. assemblează vectorii $T2'_{(p)}$

pentru $e \in \{ \text{toate elementele} \}$ (e de noduri i, j, k)

pentru fiecare nod $\in e$ (fie nodul i)

pentru $p = 1, m$

$$t2'_{(p)i}^{(e)} = \dots \text{ conform (4.21)}$$

8. calculează senzitivități

pentru $p = 1, m$

$$\text{rezolvă } SA'_{(p)} = T1'_{(p)} + T2'_{(p)} - S'_{(p)} A$$

• Integrarea calcului senzitivităților în metoda Newton aplicată modelului diferențial tratat cu metoda elementelor finite

Cu notațiile algoritmului prezentat la paragraful 3.1.4 necunoscuta principală se cal-

culează recursiv în timpul procesului iterativ

$$A^k = A^{k-1} + \delta A^k, \quad (4.22)$$

unde k este iterația, A^0 este ales arbitrar iar δA^k este corecția la iterația k , calculată prin rezolvarea unui sistem

$$P^k \delta A^k = -V^k. \quad (4.23)$$

În acest caz senzitivitățile necunoscutei principale vor trebui calculate tot recursiv

$$(A')^k = (A')^{k-1} + (\delta A')^k, \quad (4.24)$$

unde alegem $(A')^0 = \mathbf{0}$, iar $(\delta A')^k$ este soluția sistemului

$$P^k (\delta A')^k = -(V')^k - (P')^k \delta A^k. \quad (4.25)$$

Efortul de calcul al integrării calcului senzitivităților în metoda Newton este astfel mult mai mare decât la metoda polarizației. Aici la fiecare iterație neliniară trebuie rezolvat un sistem în plus pentru fiecare parametru, pe când la metoda polarizației trebuia rezolvat un sistem în plus (pentru fiecare parametru) pe tot algoritmul.

La rândul lui, vectorul V^k se calcula cu formula

$$V^k = S^k A^{k-1} - T - T_1. \quad (4.26)$$

Presupunând că frontiera nu depinde de parametrii de optimizare rezultă că

$$(V')^k = (S')^k A^{k-1} + S^k (A')^{k-1} - T'. \quad (4.27)$$

Asamblarea lui T' poate fi scoasă în afara iterațiilor neliniare, dar la fiecare iterație neliniară trebuie asamblate m matrice S' și m matrice P' și rezolvate m sisteme de tipul (4.25).

4.2 Senzitivitățile mărimilor derivate față de parametrii problemei de optimizare

Pentru mărimi locale derivate din A , de exemplu $B(A)$ sau $w(B)$ se pot imagina derivate înlănțuite care să reducă problema găsirii senzitivității lor la problema calculării senzitivității necunoscutei principale față de parametrul de optimizare:

$$\frac{\partial B}{\partial p_k} = \frac{\partial B}{\partial A} \frac{\partial A}{\partial p_k}, \quad (4.28)$$

$$\frac{\partial w}{\partial p_k} = \frac{\partial w}{\partial B} \frac{\partial B}{\partial A} \frac{\partial A}{\partial p_k}. \quad (4.29)$$

4.2.1 Calculul sensibilității inducției magnetice

Presupunem că se cunoaște sensibilitatea potențialului magnetic vector într-o problemă plan-paralelă care a fost rezolvată cu metoda elementelor finite. Inducția magnetică într-un punct M din interiorul unui triunghi e este dată de relația

$$B^2(M) = \left(\frac{\partial A}{\partial y} \right)^2 + \left(\frac{\partial A}{\partial x} \right)^2, \quad (4.30)$$

unde $A = \sum_{k=1}^3 A_k \Psi_k(x, y)$, A_1, A_2, A_3 fiind potențialele din vârfurile triunghiului e . Rezultă atunci că sensibilitatea inducției magnetice se calculează în funcție de sensibilitatea potențialului magnetic vector astfel:

$$\begin{aligned} \frac{\partial B}{\partial p}(M) &= \frac{1}{2B} \frac{\partial B^2}{\partial p}(M) = \frac{1}{2B} \left[2 \left(\frac{\partial A}{\partial y} \right) \frac{\partial}{\partial p} \left(\frac{\partial A}{\partial y} \right) + 2 \left(\frac{\partial A}{\partial x} \right) \frac{\partial}{\partial p} \left(\frac{\partial A}{\partial x} \right) \right] \\ &= \frac{1}{B} \left(\sum_{k=1}^3 A_k \frac{\partial \Psi_k}{\partial y} \right) \left(\sum_{k=1}^3 \frac{\partial A_k}{\partial p} \frac{\partial \Psi_k}{\partial y} \right) + \frac{1}{B} \left(\sum_{k=1}^3 A_k \frac{\partial \Psi_k}{\partial x} \right) \left(\sum_{k=1}^3 \frac{\partial A_k}{\partial p} \frac{\partial \Psi_k}{\partial x} \right) \end{aligned} \quad (4.31)$$

4.2.2 Sensibilitatea fluxului magnetic

Fluxul magnetic printr-o suprafața deschisă se poate exprima cu ajutorul unei integrale pe o curbă închisă Γ (frontiera suprafeței deschise) din potențialul magnetic vector

$$\phi_{S\Gamma} = \oint_{\Gamma} \mathbf{A} \cdot \mathbf{n} \, dl. \quad (4.32)$$

Dacă curba Γ nu depinde de parametrul de optimizare atunci calculul sensibilității fluxului se reduce și în acest caz la calculul sensibilității potențialului deoarece

$$\frac{\partial \phi_{S\Gamma}}{\partial p} = \oint_{\Gamma} \frac{\partial \mathbf{A}}{\partial p} \cdot \mathbf{n} \, dl. \quad (4.33)$$

În cazul în care sursele de câmp ale problemei sunt constituite exclusiv din bobine filiforme iar mărimea de interes este fluxul printr-o bobină, se poate deduce o formulă pentru sensibilitatea acestui flux cu ajutorul unei probleme adjuncte. În această situație formula (4.3) devine

$$\int_{\Omega} \mathbf{H} \cdot \mathbf{B} \, d\Omega = \sum_{k=1}^b \phi_k i_k, \quad (4.34)$$

unde b este numărul total de bobine, ϕ_k este fluxul total al bobinei k străbătută de curentul i_k .

Considerând trei probleme: problema inițială, problema perturbată și problema adjunctă, se ajunge similar ca în cazul relației (4.6) la relația

$$\int_{\Omega} (\hat{\mathbf{B}} \cdot \delta \mathbf{H} - \hat{\mathbf{H}} \cdot \delta \mathbf{B}) \, d\Omega = \sum_{k=1}^b (\hat{\phi}_k \delta i_k - i_k \delta \phi_k). \quad (4.35)$$

Alegând în problema adjuncă materiale liniare având permeabilitatea egală cu permeabilitatea dinamică a materialelor problemei inițiale și presupunând că parametrul de optimizare nu afectează curenții (deci perturbațiile δi_k sunt zero), relația (4.35) devine

$$\int_{\Omega} \hat{\mathbf{H}} \frac{\partial \mathbf{f}}{\partial p} \delta p \, d\Omega = \sum_{k=1}^b \hat{i}_k \delta \phi_k. \quad (4.36)$$

Dacă în problema adjuncă presupunem doar bobina k străbătută de curentul $\hat{i}_k \neq 0$ (cu k fixat), restul curenților adjuncți fiind nuli, din relația (4.36) rezultă formula senzitivității fluxului

$$\frac{\partial \phi_k}{\partial p} = \frac{1}{\hat{i}_k} \int_{\Omega} \hat{\mathbf{H}} \cdot \frac{\partial \mathbf{f}}{\partial p} \, d\Omega \quad (4.37)$$

Deoarece p este un parametru geometric, formula (4.37) se reduce la o integrală pe interfața parametrizată de p . Sub integrală va apare saltul mărimii $\mathbf{B} \cdot \hat{\mathbf{H}}$ în sensul creșterii lui p .

4.2.3 Senzitivitatea energiei

Senzitivitatea energiei poate fi calculată ușor în cazul unei probleme liniare în care densitatea de energie este $\frac{B^2}{2\mu}$. Ea se reduce astfel la calculul senzitivității inducției magnetice. Efortul de calcul poate fi redus utilizând relația (4.3) prin integrare pe domeniile în care $\mathbf{J} \neq \mathbf{0}$.

4.3 Utilizarea senzitivităților în rezolvarea problemelor de optimizare

În acest moment ne interesează doar minimizarea unei funcții reale multidimensionale $f(\mathbf{x})$, $f: \mathbb{R}^m \rightarrow \mathbb{R}$ pentru care se poate calcula gradientul $-\nabla f$. Acest subiect este tratat pe larg în literatura de specialitate [49, 128, 131]. Iată de exemplu algoritmul metodei pașilor descendenți cu pas variabil. Algoritmul pleacă dintr-un punct P_0 , apoi, de câte ori este necesar, punctul curent se mișcă din P_i în P_{i+1} prin minimizare unidimensională de-a lungul unei linii care trece prin P_i și are direcția gradientului local luat cu semnul minus $-\nabla f(P_i)$.

1. Alege:

\mathbf{x}_0

; inițializarea soluției

maxit

; numărul maxim de iterații permis

ε

2. $k = 0$

; contor iterații

3. cât timp $k \leq \text{maxit}$ repetă

- 3.1. $\mathbf{g}^k = -\text{grad } f(\mathbf{x}^k)$; calculează direcția gradientului
- 3.2. $\rho^k = \arg \min_{\rho > 0} f(\mathbf{x}^k + \rho \mathbf{g}^k)$; calculează pasul pe direcția aleasă
- 3.3. $\mathbf{x}^{k+1} = \mathbf{x}^k + \rho^k \mathbf{g}^k$; noua soluție
- 3.4. **dacă** $\|\text{grad } f(\mathbf{x}^k)\| < \varepsilon$ **stop**

Capitolul 5

Algoritmi evoluționiști de optimizare pentru arhitecturi de calcul distribuite

"The subject of optimization is a fascinating blend of heuristics and rigour, of theory and experiment."

R. Fletcher

Metodele deterministe de optimizare au marele dezavantaj că sunt capabile să găsească doar extreme locale, dependente de inițializare. Metodele de optimizare care urmăresc găsirea extremelor globale folosesc de aceea și alte tehnici (euristice) de căutare. În ultimul deceniu au fost folosiți tot mai mult algoritmi bazați pe modele biologice evoluționiste în rezolvarea problemelor de optimizare din diferite domenii ¹. Anexa D descrie principalele concepte, structura unui program de evoluție și principalele probleme legate de conceperea unor astfel de algoritmi. Teoria algoritmilor genetici (în care codificarea indivizilor se face binar) furnizează explicații pentru convergența acestor algoritmi către extreme globale. Se constată totuși că și pentru alte tipuri de reprezentări genetice programele de evoluție tind către "mai bine". Charles Darwin spunea în *Origin of Species*: "As natural selection works solely by and for the good of each being, all corporeal and mental endowments will tend to progress toward perfection."

Folosirea programelor bazate pe modele evoluționiste se poate aplica și în optimizarea dispozitivelor electromagnetice. De aceea, în cele ce urmează, vom descrie programul de evoluție folosit în optimizarea dispozitivelor descrise în capitolul al șaselea. Deoarece, în

¹Aceasta a impus apariția de curând (aprilie 1997) a primului număr al revistei *IEEE Transactions on Evolutionary Computation*. O căutare a cuvintelor cheie "Evolutionary Computation" pe Internet generează o listă cu 80000 documente iar a cuvintelor cheie "Genetic Algorithms" o listă cu 420000 documente.

general, algoritmi evoluționiști sunt costisitori din punct de vedere al timpului de calcul, lucru și mai evident atunci când sunt folosiți în optimizarea dispozitivelor electromagnetice, am urmărit implementarea unor variante de algoritmi care să ruleze în paralel pe arhitecturi de calcul distribuite. Algoritmi implementați fac parte din categoria algoritmilor cu granularitate mare a paralelizării. Populația este divizată într-un număr (relativ mic) de subpopulații (insule) care evoluează în paralel și care schimbă din când în când informații (indivizi).

5.1 Formularea problemei

Într-o problemă de optimizare cu restricții forma geometrică a mulțimii de soluții acceptabile din \mathbb{R}^q (unde q este numărul parametrilor de optimizare) este o caracteristică deosebit de importantă. Există un tip special de mulțime, cea convexă, pentru care a fost dezvoltată o cantitate semnificativă de teorie, deoarece este un caz frecvent întâlnit în practică.

Din acest motiv, algoritmi prezentați în continuare tratează cazul în care funcția obiectiv f este definită pe un domeniu convex paralelipipedic

$$\mathcal{D} \subset \mathbb{R}^q, \quad \text{unde} \quad \mathcal{D} = \prod_{k=1}^q [l_k, u_k], \quad (5.1)$$

unde $[l_k, u_k]$ reprezintă intervalul pe care este definită variabila x_k .

Problema tratată poate fi formulată astfel:

Optimizează funcția $f(x_1, x_2, \dots, x_q)$ care are următoarele restricții liniare:

- q restricții de domeniu:

$$l_i \leq x_i \leq u_i \quad i = 1, \dots, q. \quad (5.2)$$

Aceste relații le putem scrie simbolic astfel: $\mathbf{l} \leq \mathbf{x} \leq \mathbf{u}$ unde $\mathbf{l} = (l_1, l_2, \dots, l_q)$, $\mathbf{u} = (u_1, u_2, \dots, u_q)$.

- p ecuații independente:

$$\mathbf{Ax} = \mathbf{b}, \quad (5.3)$$

unde $\mathbf{A} = (a_{ij})$, $i = 1, \dots, p$, $j = 1, \dots, q$, $\mathbf{b} = (b_1, b_2, \dots, b_p)$.

- m inecuații:

$$\mathbf{Cx} \leq \mathbf{d}, \quad (5.4)$$

unde $\mathbf{C} = (c_{ij})$, $i = 1, \dots, m$, $j = 1, \dots, q$, $\mathbf{d} = (d_1, d_2, \dots, d_m)$.

Această formulare este destul de generoasă pentru clasa de probleme de optimizare standard.

Algoritmul de optimizare propriu-zis nu se aplică problemei așa cum este definită mai sus. Deoarece cele p ecuații sunt independente, înseamnă că există p variabile $(x_{i_1}, x_{i_2}, \dots, x_{i_p})$, unde $\{x_{i_1}, x_{i_2}, \dots, x_{i_p}\} \subset \{1, 2, \dots, q\}$, care pot fi eliminate în funcție de celelalte.

Matricea \mathbf{A} se împarte în două matrice \mathbf{A}_1 și \mathbf{A}_2 astfel: coloana j a lui \mathbf{A} va fi coloană a matricei \mathbf{A}_1 dacă și numai dacă $j \in \{i_1, i_2, \dots, i_p\}$. Matricea \mathbf{A}_1 este o matrice pătrată și deoarece egalitățile sunt independente ea este și inversabilă. Există deci \mathbf{A}_1^{-1} .

Analog se împarte matricea \mathbf{C} în matricele \mathbf{C}_1 și \mathbf{C}_2 , precum și vectorii \mathbf{x} , \mathbf{l} și \mathbf{u} . Vom nota astfel de exemplu cu

$$\mathbf{x}^1 = [x_{i_1}, x_{i_2}, \dots, x_{i_p}] \quad (5.5)$$

vectorul variabilelor eliminate și cu \mathbf{x}^2 vectorul variabilelor în funcție de care se face optimizarea propriu-zisă, cu

$$\mathbf{l}_1 = [l_{i_1}, l_{i_2}, \dots, l_{i_p}], \quad (5.6)$$

$$\mathbf{u}_1 = [u_{i_1}, u_{i_2}, \dots, u_{i_p}] \quad (5.7)$$

restricțiile de domeniu ale variabilelor care vor fi eliminate.

Cele p egalități se scriu

$$\mathbf{A}_1 \mathbf{x}^1 + \mathbf{A}_2 \mathbf{x}^2 = \mathbf{b}, \quad (5.8)$$

de unde rezultă

$$\mathbf{x}^1 = \mathbf{A}_1^{-1} \mathbf{b} - \mathbf{A}_1^{-1} \mathbf{A}_2 \mathbf{x}^2. \quad (5.9)$$

Variabilele eliminate sunt deci combinații liniare de variabile care rămân. Pentru fiecare variabilă eliminată există o restricție de domeniu $l_{i_j} \leq x_{i_j} \leq u_{i_j}$. De aceea, fiecare restricție eliminată va face să apară un nou set de inegalități

$$\mathbf{l}_1 \leq \mathbf{A}_1^{-1} \mathbf{b} - \mathbf{A}_1^{-1} \mathbf{A}_2 \mathbf{x}^2 \leq \mathbf{u}_1, \quad (5.10)$$

care se adaugă la inegalitățile inițiale.

Setul inițial de inegalități se rescrie

$$\mathbf{C}_1 \mathbf{x}^1 + \mathbf{C}_2 \mathbf{x}^2 \leq \mathbf{d} \quad (5.11)$$

și înlocuind \mathbf{x}^1 rezultă

$$\mathbf{C}_1 (\mathbf{A}_1^{-1} \mathbf{b} - \mathbf{A}_1^{-1} \mathbf{A}_2 \mathbf{x}^2) + \mathbf{C}_2 \mathbf{x}^2 \leq \mathbf{d}. \quad (5.12)$$

În concluzie, după eliminarea celor p variabile $x_{i_1}, x_{i_2}, \dots, x_{i_p}$ setul final de restricții cuprinde doar următoarele inegalități:

• de domeniu:

$$\mathbf{l}_2 \leq \mathbf{x}^2 \leq \mathbf{u}_2, \quad (5.13)$$

- inegalități noi:

$$\mathbf{l}_1 \leq \mathbf{A}_1^{-1}\mathbf{b} - \mathbf{A}_1^{-1}\mathbf{A}_2\mathbf{x}^2 \leq \mathbf{u}_1, \quad (5.14)$$

- inegalități vechi rescrise:

$$(\mathbf{C}_2 - \mathbf{C}_1\mathbf{A}_1^{-1}\mathbf{A}_2)\mathbf{x}^2 \leq \mathbf{d} - \mathbf{C}_1\mathbf{A}_1^{-1}\mathbf{b}. \quad (5.15)$$

5.2 Descrierea algoritmului secvențial

Algoritmul evoluționist distribuit are la bază unul secvențial. În cele ce urmează vom descrie componentele principale ale algoritmului secvențial folosit.

5.2.1 Reprezentarea genetică și dimensiunea populației

Una din componentele principale ale unui program de evoluție este reprezentarea genetică, adică alegerea unei codificări potrivite pentru parametrii de optimizare. Există mai multe posibilități de alegere a reprezentării genetice. Se consideră că majoritatea celor ce utilizează astfel de programe preferă reprezentări legate de problemă [12]. Astfel, dacă într-o problemă de optimizare de parametri este permisă o variație continuă a acestora, atunci pentru ei se adoptă ca reprezentare genetică un număr real pentru fiecare parametru. Dacă variația este discretă, atunci se preferă o codificare binară pentru fiecare parametru. În această din urmă situație, pentru ca doi vecini în așa numitul “spațiu fenotip” (spațiul real) să fie vecini și în “spațiul genotip” (spațiul reprezentării genetice), se preferă codificarea binară GRAY în care reprezentarea binară a două numere consecutive diferă printr-un singur bit.

Din acest motiv, deoarece în problemele studiate parametrii pot varia continuu între anumite limite, reprezentarea genetică aleasă pentru un individ este un vector de numere reale de dimensiune q (numărul total de variabile). Fiecare componentă a vectorului reprezintă o valoare posibilă pentru un anumit parametru de optimizare.

Dimensiunea populației (sau subpopulațiilor în cazul algoritmilor distribuiți) am considerat-o fixă. Vom nota această dimensiune cu POP_SIZE.

5.2.2 Populația inițială

Populația inițială este formată din indivizi diferiți, uniform repartizați în întreg domeniul de căutare (generat de limitele minime și maxime admise ale parametrilor). Mai întâi se partiționează domeniul în celule disjuncte și apoi se generează aleator în fiecare celulă câte

un individ care să satisfacă restul restricțiilor (de tip inegalitate). Un astfel de individ va aparține populației inițiale.

5.2.3 Operatori folosiți

• Mutație uniformă

Acest operator se aplică unui singur părinte \mathbf{x} și produce un singur copil \mathbf{x}' . El este deci un operator unar. Operatorul selectează o componentă aleatoare $k \in \{1, 2, \dots, q\}$ a vectorului $\mathbf{x} = (x_1, \dots, x_k, \dots, x_q)$ și produce un vector $\mathbf{x}' = (x_1, \dots, x'_k, \dots, x_q)$, unde x'_k este o valoare aleatoare (probabilitatea de distribuție fiind uniformă) în limitele domeniului ei.

Acest operator joacă un rol important în fazele timpurii ale procesului de evoluție, permițând indivizilor să se miște în spațiul de căutare. Operatorul este esențial în algoritmul în care populația inițială constă în copii identice ale aceluiași individ ².

În fazele înaintate ale procesului de evoluție operatorul permite ieșirea din zona unui optim local.

• Mutație pe frontieră

Acest operator se aplică de asemenea unui singur părinte \mathbf{x} și produce un singur copil \mathbf{x}' . El este o variantă a mutației uniforme, unde x'_k este una din cele două limite ale domeniului pe care este definită variabila k .

Operatorul este folosit în problemele de optimizare în care soluția optimă se află pe sau lângă frontiera spațiului de căutare. În consecință un astfel de operator nu are nici un sens în problemele fără restricții și în care limitele unei variabile sunt mari. El se dovedește a fi foarte folosit în prezența restricțiilor.

• Mutație neuniformă

Mutația neuniformă este de asemenea un operator unar și ea este cea care asigură reglajul fin al căutării. Operatorul este definit astfel: pentru un părinte \mathbf{x} , dacă elementul x_k a fost selectat pentru această mutație, atunci rezultatul este $\mathbf{x}' = (x_1, \dots, x'_k, \dots, x_q)$ unde

$$x'_k = \begin{cases} x_k + \Delta(t, \text{right}(k) - x_k) & \text{daca un bit aleator este 0} \\ x_k - \Delta(t, x_k - \text{left}(k)) & \text{daca un bit aleator este 1.} \end{cases} \quad (5.16)$$

În formula de mai sus “right(k)” și “left(k)” reprezintă domeniul variabilei x_k . Funcția $\Delta(t, y)$ întoarce o valoare în domeniul $[0, y]$ astfel încât probabilitatea ca $\Delta(t, y)$ să fie cât

²De exemplu GENOCOP - program evoluționist ce poate fi găsit pe Internet la adresa <http://www.aic.nrl.navy.mil:80/galist/src>.

mai apropiată de 0 să crească pe măsură ce t crește, t fiind numărul generației curente. O astfel de definiție a lui Δ face ca operatorul să exploreze inițial (atunci când t este mic) uniform domeniul de căutare și să facă o exploare locală atunci când t este mare. Iată un exemplu de astfel de funcție ³:

$$\Delta(t, y) = yr \left[1 - \left(\frac{t}{T} \right)^b \right], \quad (5.17)$$

unde r este un număr aleator între 0 și 1, T este numărul maxim de generații și b este un parametru care determină gradul de neuniformitate.

• Încrucișare aritmetică

Încrucișarea aritmetică este un operator binar. Ea se aplică la doi părinți \mathbf{x}_1 și \mathbf{x}_2 . Din încrucișarea celor doi părinți rezultă doi copii \mathbf{x}'_1 și \mathbf{x}'_2 , fiecare din ei fiind o combinație liniară de cei doi părinți:

$$\mathbf{x}'_1 = a\mathbf{x}_1 + (1 - a)\mathbf{x}_2, \quad (5.18)$$

$$\mathbf{x}'_2 = a\mathbf{x}_2 + (1 - a)\mathbf{x}_1. \quad (5.19)$$

Parametrul a este o valoare aleatoare în intervalul $[0, 1]$. Într-un domeniu convex, cei doi copii vor fi întotdeauna în spațiul de căutare.

Iată alte denumiri care se cunosc pentru acest operator: *încrucișare medie garantată* (pentru $a = 0.5$), *încrucișare intermediară*, *încrucișare liniară*.

S-a constatat de asemenea că un algoritm genetic care folosește și acest fel de încrucișare este mai stabil, deviația standard a celor mai bune soluții (obținute din mai multe rulări) fiind mai mică.

Parametrul a s-ar putea determina prin optimizare unidimensională prin metode deterministe.

• Încrucișare simplă

Încrucișarea simplă este de asemenea un operator binar. Doi părinți $\mathbf{x}_1 = (x_1, \dots, x_q)$ și $\mathbf{x}_2 = (y_1, \dots, y_q)$ sunt încrucișati la poziția k , copiii rezultați fiind $\mathbf{x}'_1 = (x_1, \dots, x_k, y_{k+1}, \dots, y_q)$ și $\mathbf{x}'_2 = (y_1, \dots, y_k, x_{k+1}, \dots, x_q)$. Un astfel de operator ar putea produce copii în afara spațiului de căutare. Pentru a evita acest lucru, se poate folosi proprietatea mulțimilor convexe și anume faptul că există $a \in [0, 1]$ astfel încât

$$\mathbf{x}'_1 = (x_1, \dots, x_k, y_{k+1}a + x_{k+1}(1 - a), \dots, y_qa + x_q(1 - a)) \quad (5.20)$$

și

$$\mathbf{x}'_2 = (y_1, \dots, y_k, x_{k+1}a + y_{k+1}(1 - a), \dots, x_qa + y_q(1 - a)) \quad (5.21)$$

³folosită în GENOCOP

să fie în spațiul de căutare. Problema care mai rămâne de rezolvat este să se găsească cea mai mare valoare a lui a , aceasta corespunzând celei mai mari cantități de informație schimbată. Cea mai simplă metodă de a face acest lucru este să se pornească cu $a = 1$ și, dacă cel puțin unul din copii nu este în spațiul de căutare, se descrește a de ρ ori s.a.m.d până când cei doi copii sunt în spațiul de căutare.

Rezultatele numerice arată că un program fără încrucișare simplă este chiar mai puțin stabil decât un program fără încrucișare aritmetică.

• Mutație neuniformă a tuturor genelor

Toate genele (componentele) părintelui cărui i se aplică acest operator suferă o mutație neuniformă de tipul celei descrise la operatorul mutație neuniformă.

Operatorii se aplică succesiv astfel: de P_1 ori mutația uniformă, de P_2 ori mutația pe frontieră, de P_3 ori mutația neuniformă, de P_4 ori încrucișarea aritmetică, de P_5 ori încrucișarea simplă și de P_6 ori mutația neuniformă a tuturor genelor.

5.2.4 Mecanismul de selecție

În vederea selecției părinților pentru reproducere populația se ordonează astfel încât primul individ este cel mai bun și apoi urmează ceilalți în ordine descrescătoare. Pentru reproducere se selectează în mod independent $P = \sum_{i=1}^6 P_i$ cromozomi (nu neaparat distincți). Selecția se bazează pe poziția (rangul) unui cromozom în populație.

Cu ajutorul unui parametru $q \in (0, 1)$ se definește o funcție neliniară

$$\text{probab}(i) = q(1 - q)^{i-1} \quad (5.22)$$

unde i este un întreg între 1 și dimensiunea populației, $i = 1$ corespunde celui mai bun individ și $i = \text{POP_SIZE}$ celui mai prost. Această funcție reprezintă probabilitatea ca un individ din poziția i să fie selectat la o singură selecție. Această schemă permite utilizatorului să influențeze presiunea de selecție⁴.

Se calculează apoi probabilitățile cumulate

$$\text{cum_probab}(i) = \sum_{j=1}^i (\text{probab}(j)). \quad (5.23)$$

⁴Presiunea de selecție se referă la gradul în care indivizii buni sunt favorizați: cu cât presiunea de selecție este mai mare, cu atât mai mult sunt favorizați indivizii mai buni să devină părinți. Rata de convergență a unui algoritm evoluționist este determinată în mare măsură de presiunea de selecție: cu cât aceasta este mai mare, cu atât rata de convergență crește. Dacă presiunea de selecție este prea mare, algoritmul ar putea converge către puncte sub-optimale.

Procedeeul de selecție al părinților se bazează pe un proces asemănător ruletei. Roata ruletei se învâртеște de P ori și de fiecare dată se selectează un individ (cromozom) pentru a fi părinte. La o rotație a roții ruletei:

- Se generează un număr real aleator $random \in [0, 1]$;
- Dacă $random < cum_probab(1)$ atunci primul cromozom va fi selectat drept părinte, altfel va fi selectat drept părinte primul cromozom (să zicem i) pentru care $cum_probab(i - 1) < random \leq cum_probab(i)$.

Evident unii cromozomi vor fi selectați mai mult decât o dată pentru a deveni părinți. Cei mai buni cromozomi (cei având număr de ordine mic în populație) au șanse de a se reproduce de mai multe ori.

În figurile 5.1 și 5.2 se pot observa valorile vectorului `cum_probab` pentru diferite valori ale parametrului q în cazul unei populații cu 20 indivizi și în cazul unei populații cu 70 indivizi.

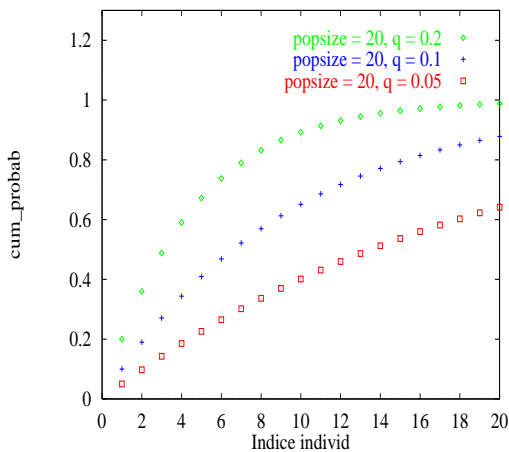


Figura 5.1: Vectorul `cum_probab` pentru `POP_SIZE = 20`

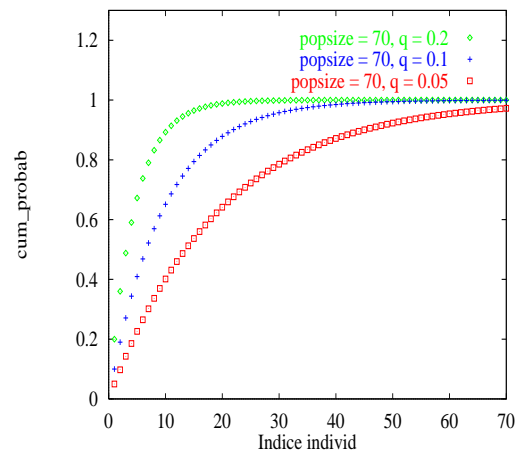


Figura 5.2: Vectorul `cum_probab` pentru `POP_SIZE = 70`

Se observă că pentru o populație cu 70 de indivizi, pentru $q = 0.2$ indivizii aleși drept părinți vor fi practic în primii 20. Cu cât q scade, pot fi aleși părinți și din indivizii mai slabi.

Selecția folosită este “on the fly” (din zbor): dacă un copil este mai bun decât părintele său el îl înlocuiește imediat, altfel nu este luat în considerare.

5.2.5 Algoritmul folosit

Algoritmul genetic (secvențial) folosit este următorul:

0. $t = 0$

1. inițializează $P(t)$

2. evaluează $P(t)$

3. ordonează $P(t)$

3. **repetă**

3.1. selectează P părinți

3.2. aplică de P_1 ori mutația uniformă (evaluează copii, înlocuiește sau nu părinți)

3.3. aplică de P_2 ori mutația pe frontieră (...)

3.4. aplică de P_3 ori mutația neuniformă

3.5. aplică de P_4 ori încrucișarea aritmetică

3.6. aplică de P_5 ori încrucișarea simplă

3.7. aplică de P_6 ori mutația neuniformă a tuturor genelor

3.8. ordonează $P(t + 1)$

3.6. $t = t + 1$

până când (este îndeplinită condiția de oprire)

5.2.6 Condiția de oprire

Algoritmul se oprește dacă a fost depășit un număr maxim de iterații (generații) impus. Alte criterii de oprire întâlnite sunt: impunerea unui timp maxim de rulare, impunerea unui număr maxim de evaluări de funcții, testarea convergenței algoritmului genetic. Această din urmă condiție de oprire este mai delicată. În general, se consideră că algoritmul genetic a converșit atunci când indivizii din populație sunt asemănători (se spune că populația a *degenerat*, și-a pierdut diversitatea), și în consecință încrucișarea lor nu mai are efect.

5.2.7 Parametrii algoritmului secvențial

Vom contabiliza acum parametrii algoritmului secvențial, cei care trebuie fixați de utilizator (în afara tipului de codificare și a mecanismului de selecție). Aceștia sunt:

- POP_SIZE = dimensiunea populației;
- q = parametru real între 0 și 1 care influențează presiunea de selecție;
- a = valoare reală între 0 și 1 folosită de operatorul încrucișare aritmetică;
- ρ = număr real mai mare decât 1 folosit de operatorul încrucișare simplă pentru a găsi copii în domeniul de căutare;
- b = valoare reală mai mare decât 1 folosită de operatorul mutație neuniformă;
- P_1 = numărul de mutații uniforme / generație;

- P_2 = numărul de mutații pe frontieră / generație;
- P_3 = numărul de mutații neuniforme / generație;
- P_4 = dublul numărului de încrucișări aritmetice / generație;
- P_5 = dublul numărului de încrucișări simple / generație;
- P_6 = numărul de mutații neuniforme a tuturor genelor / generație;
- T = numărul maxim de generații.

Sarcina utilizatorului este deosebit de complicată. Alegerea parametrilor influențează succesul sau eșecul algoritmului evoluționist. Un mic studiu al influenței parametrilor asupra evoluției procesului de optimizare este prezentat în paragraful 5.4.3. Soluția care se practică este aceea de a folosi un algoritm genetic (numit meta GA) care să optimizeze parametrii strategiei evoluționiste.

5.3 Algoritmi evoluționiști distribuiți

5.3.1 Implementări paralele ale algoritmilor evoluționiști

Implementările paralele ale algoritmilor evoluționiști pot fi clasificate astfel [26, 96, 104]:

- *Algoritmi cu paralelizare globală*⁵. Acești algoritmi efectuează evaluările indivizilor în paralel. Un individ se poate combina cu oricare alt individ al populației, operatorii fiind neschimbați față de modelul secvențial. Algoritmul se comportă exact ca unul secvențial, timpul de execuție scăzând proporțional cu numărul de procesoare.
- *Algoritmi cu granularitate mare a paralelizării*⁶. Populația este divizată în mai multe subpopulații (insule) care evoluează în paralel, operatorii genetici fiind aplicați în cadrul fiecărei subpopulații în parte. Schimbul de informații între subpopulații se efectuează cu ajutorul unui operator de migrare. În cadrul acestei categorii sunt cunoscute *modelul insulelor paralele*⁷ în care migrarea indivizilor se poate face către oricare altă populație, și *modelul trepte*⁸, în care apar restricții în ceea ce privește migrarea indivizilor, schimbul de indivizi neputând avea loc decât între subpopulații considerate vecine.

În general, algoritmi din această categorie sunt cunoscuți mai ales sub denumirea de *algoritmi genetici distribuiți* datorită implementărilor uzuale pe arhitecturi de

⁵ “micro-grain”

⁶ “coarse-grain”

⁷ “parallel island model”

⁸ “stepping-stone model”

tip MIMD⁹ cu memorie distribuită.

- *Algoritmi genetici cu granularitate fină a paralelizării*¹⁰. Din această categorie fac parte algoritmi care folosesc un număr larg de subpopulații de mici dimensiuni. Cazul limită este obținut când într-o subpopulație se află un singur individ, și poate fi implementat cu avantaje maxime pe calculatoare paralele cu un număr ridicat de procesoare (de obicei 2¹⁰ sau mai mult). Acești algoritmi se mai numesc și *algoritmi masiv paraleli*.
- *Algoritmi paraleli hibridi*. Modelul este similar cu modelul algoritmilor masiv paraleli deoarece există tot o corespondență unu la unu între procesoare și indivizi. Totuși, sunt folosite un număr mic de procesoare. De obicei astfel de algoritmi includ și tehnici deterministe pentru a îmbunătăți performanțele sistemului.

Sub numele de algoritmi hibridi se întâlnesc și combinații între tipurile de implementări paralele descrise mai sus.

Algoritmii evoluționiști cu paralelizare globală evoluează la fel ca algoritmii secvențiali. Celelalte implementări paralele introduc însă schimbări fundamentale.

Cei mai populari algoritmi evoluționiști paraleli sunt cei cu granularitate mare a paralelizării. Succesul algoritmilor în care apare migrația a fost explicat folosind argumente din biologie: într-o populație izolată în care apar schimbări în mediu, este probabil să apară repede noi specii. Numărul de emigranți determină nivelul acestor schimbări și noi soluții apar repede după migrație.

Experiența a arătat că, în general, un algoritm evoluționist paralel care folosește migrația este mai bun și mai rapid decât un algoritm paralel care nu folosește migrația, acesta la rândul lui fiind mai rapid decât un algoritm evoluționist secvențial.

5.3.2 Parametrii specifici unui algoritm paralel ce folosește migrația

Dimensiunea populației este probabil parametrul care afectează cel mai tare performanța unui algoritm genetic. De aceea studiul algoritmilor paraleli trebuie să înceapă cu **dimensiunea subpopulațiilor**. Există de asemenea un **număr optim de subpopulații** (de o anumită dimensiune) care maximizează speedup-ul.

În ceea ce privește migrația, aceasta este controlată de următorii parametri: topologia care definește conexiunea între subpopulații, rata de migrație care controlează numărul

⁹“Multiple Instructions Multiple Data”. Pentru clarificări și clasificări ale sistemelor paralele se pot consulta [75, 111].

¹⁰ “fine grain”

de indivizi care migrează, intervalul (frecvența) de migrație care arată cât de des are loc migrația. Mai trebuie stabilite cum are loc **selecția indivizilor pentru migrare** (aleator, sau în funcție de calitatea lor) și **inserarea indivizilor în noua populație**.

Iată câteva din concluziile cercetătorilor din domeniu [26]:

- **Topologia (schema de conexiune)** este un factor important pentru performanța unui algoritm genetic paralel, deoarece ea determină cât de repede (sau de încet) o soluție bună diseminează către alte subpopulații.

Alegerea topologiei nu este foarte importantă în performanțele unui algoritm paralel atât timp cât ea are *conectivitate* mare și *diametru* mic (Cohon, 1987). Dacă topologia are o conectivitate densă (sau diametru scurt, sau ambele), soluțiile bune vor ajunge repede la toate subpopulațiile. Dacă topologia are o conectivitate rară (sau diametru lung), soluțiile bune se vor împrăști încet, subpopulațiile sunt mai izolate unele de altele, permițând apariția de soluții diferite. Aceste soluții s-ar putea combina la un moment de timp ulterior, și ar putea forma indivizi mai buni.

Topologia de comunicație este importantă și pentru că ea determină timpul de migrație. De exemplu o topologie densă asigură o bună amestecare a indivizilor, dar costurile de comunicație pot fi mari.

Tendința generală este de a folosi topologii *statice*, care rămân neschimbate în timpul execuției algoritmului. Multe implementări cu topologii statice folosesc aceeași topologie ca și topologia rețelei de calculatoare disponibile. Implementări de hiper-cuburi și inele sunt astfel obișnuite. Studii empirice arată că algoritmi paraleli cu topologii dense găsesc soluția globală cu mai puține evaluări de funcții decât algoritmi paraleli cu topologii rare.

Există de asemenea și implementări care folosesc topologii *dinamice*. În acest caz o subpopulație nu comunică doar cu o mulțime fixă de alte subpopulații, ci ea trimite indivizi tuturor subpopulațiilor care satisfac anumite criterii. Motivația topologiilor dinamice este aceea de a identifica subpopulațiile în care indivizii care emigrează pot produce un anumit efect. De obicei, criteriile folosite pentru alegerea unei subpopulații drept destinație includ informații legate de diversitatea subpopulației, sau de distanța dintre două subpopulații, sau informații legate de anumiți indivizi (de exemplu cei mai buni). Avantajul unei topologii dinamice este și acela că, în cazul opririi unui procesor (nod), algoritmul ar putea reconfigura schema de conexiune dintre procesoarele rămase.

O abordare diferită a fost propusă de Marin, Trelles-Salazar și Sandoval (1994). Ei propun o schemă centralizată în care sclavii execută un algoritm genetic, periodic trimit cele mai bune rezultate unui proces *master*, acesta îi alege pe cei mai buni indivizi pe care îi trimite apoi tuturor proceselor *slave*.

- **Rata de migrație** are o valoare critică, sub care performanța algoritmului este micșorată datorită izolării subpopulațiilor, și peste care algoritmul se comportă ca unul secvențial (Grosso, 1985).
- **Frecvența migrației** influențează de asemenea performanțele algoritmului. Tanese (1989) face un studiu asupra frecvenței migrației și numărului de indivizi care migrează, ajungând la concluzia că migrarea a prea multor indivizi prea frecvent, sau a prea puțin indivizi prea rar, degradează performanțele algoritmului.

În majoritatea algoritmilor cu granularitate mare, migrația este *sincronă*, adică apare la intervale constante, predeterminate. Această abordare are dezavantajul că procesoarele rapide sunt încetinite de cele mai lente. Migrația poate fi *asincronă*, comunicația dintre subpopulații realizându-se după apariția unui anumit eveniment, nelegat de momentul evoluției subpopulației (model mai apropiat de modelul natural). Dezavantajul unei astfel de implementări este acela că un individ foarte bun (dar sub-optimal) poate fi inserat într-o populație proastă. El ar putea domina populația, ducând la o convergență prematură a ei.

Se pare că, dacă migrația apare prea devreme, emigranții sunt prea slabi pentru a influența căutarea în direcția corectă, aceasta însemnând irosirea resurelor. Braun (1990) prezintă un algoritm în care migrația apare după ce subpopulațiile degenerază (converg) ¹¹.

Un alt aspect este acela al **omogenității** procesoarelor (insulelor, nodurilor). Un algoritm paralel cu insule omogene folosește în fiecare nod aceeași parametri, operatori, tip de codificare. Un astfel de algoritm are avantajul că este simplu de implementat. În cazul insulelor neomogene în fiecare nod pot fi folosite diferite metode de codificare, operatori genetici diferiți, funcții obiectiv diferite. Implementarea este mai complicată (indivizii trebuie să-și schimbe tipul de codificare la migrație), dar în acest caz probabilitatea de convergență prematură este mai mică.

5.3.3 Algoritmii folosiți

Algoritmii genetici distribuiți adoptați sunt de tipul celor cu granularitate mare a paralelizării. Au fost implementate două tipuri de topologii: una statică, de tip inel unidirecțional și una dinamică, bazată pe ierarhizarea subpopulațiilor. Insulele sunt omogene și sincrone.

¹¹Populația se consideră degenerată atunci când deviația standard este mică și încrucișarea nu mai are efect.

5.4 Teste numerice

Algoritmul implementat a fost testat pe funcții analitice care au mai multe extreme. Exemple de astfel de funcții pot fi găsite în anexa A.

Deoarece un algoritm statistic se comportă diferit la rulări diferite care pornesc din același condiții inițiale, pentru comparație fiecare caz a fost rulat de 20 ori consecutiv.

Pentru estimarea efortului de calcul a fost contorizat numărul de evaluări de funcții obiectiv, această operație fiind cea mai costisitoare în optimizările de dispozitive electro-magnetice.

5.4.1 Descrierea problemelor de test

Funcția ”six-hump camel back”

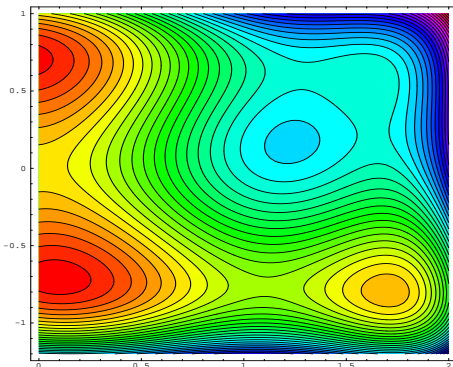


Figura 5.3: Harta funcției $C(x, y)$

pentru $0 \leq x \leq 3$. Figura 5.3 prezintă 40 curbe de echivalență în intervalul $(-1.1; 5.8)$, domeniile dintre curbe fiind colorate conform valorilor funcției (culoarea roșie corespunde zonelor de minim). Se observă prezența a trei minime (din care două sunt locale).

Prezența minimelor locale se poate observa și din graficele $C(x, -0.7126)$ (figura 5.4) și $C(0.0898, y)$ (figura 5.5).

Media funcției pe domeniul de căutare este 20.1609.

Funcția lui Bohachevsky

Această funcție are expresia:

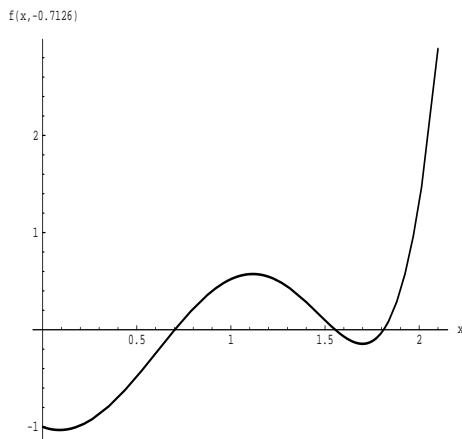
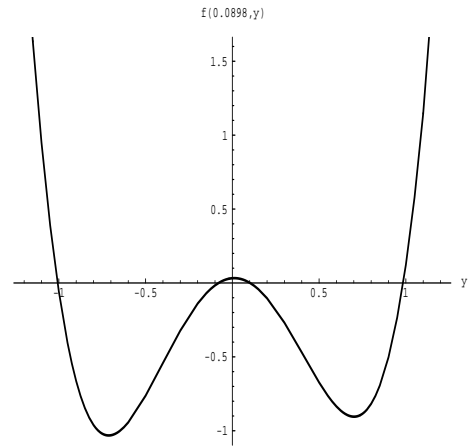
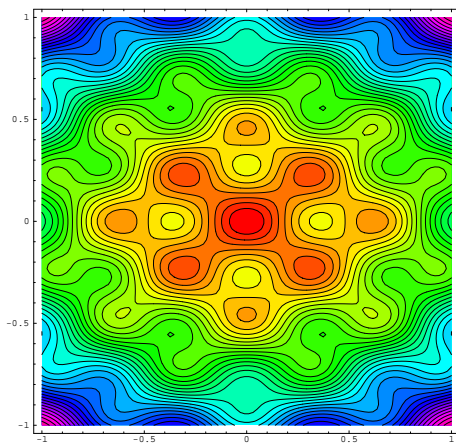
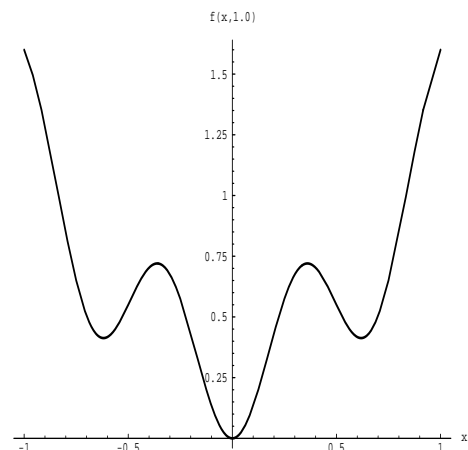
$$B(x, y) = x^2 + 2y^2 - 0.3 \cos(3\pi x) \cos(4\pi y) + 0.3, \quad (5.25)$$

Această funcție are expresia:

$$C(x, y) = \left(4 - 2.1x^2 + \frac{x^4}{3}\right)x^2 + xy + (-4 + 4y^2)y^2, \quad (5.24)$$

unde $-3 \leq x \leq 3$ și $-2 \leq y \leq 2$. Funcția are un minim global egal cu -1.03163 în două puncte diferite: $(x, y) = (-0.0898, -0.7126)$ și $(x, y) = (0.0898, -0.7126)$

Pentru ca extremul global să fie unic, funcția folosită în programul de optimizare este definită

Figura 5.4: $C(x, -0.7126)$ Figura 5.5: $C(0.0898, y)$ Figura 5.6: Harta funcției $B(x, y)$ Figura 5.7: $B(x, 1)$

unde $-50 \leq x \leq 50$ și $-50 \leq y \leq 50$. Funcția are un minim global egal cu 0 în $(x, y) = (0, 0)$.

Figura 5.6 prezintă 40 curbe de echivalenți în intervalul $(0; 4)$, domeniile dintre curbe fiind colorate conform valorilor funcției (culoarea roșie corespunde zonelor de minim). Se observă prezența a nouă minime (din care patru sunt locale). Prezența minimelor locale se poate observa și din graficul $B(x, -1)$ (figura 5.7). Media funcției pe domeniul de căutare este 2500.3.

Funcția lui Rosenbrock

Această funcție are expresia:

$$R(x, y) = 100(y - x^2)^2 + (1 - x)^2, \quad (5.26)$$

unde $-2.048 \leq x \leq 2.048$ și $-2.048 \leq y \leq 2.048$. Funcția are un minim global egal cu 0 în punctul $(x, y) = (1, 1)$. Minime locale nu există, dar funcția are un relief complicat pentru algoritmi de optimizare de ordinul unu. Pentru puncte de start depărtate de minim, direcția gradientului nu indică direcția corectă către minim, valea funcției fiind răsucită.

Figura 5.8 prezintă 7 curbe de echivalori în intervalul $(0.1; 10)$ iar figura 5.9 o secțiune pe la $y = 0.2$. Pe aceasta din urmă se observă două minime. Doar unul din ele este minim adevărat. Ele reprezintă de fapt aceeași vale care se răsucește. Pentru o astfel de funcție, minimizarea la un moment dat după o singură direcție ar putea de fapt să îndepărteze punctul curent de minimul adevărat.

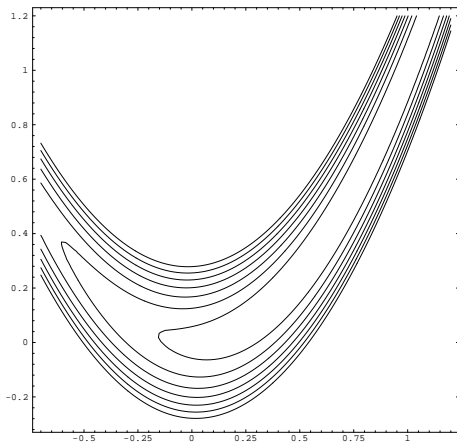


Figura 5.8: Echivalori de $R(x, y)$

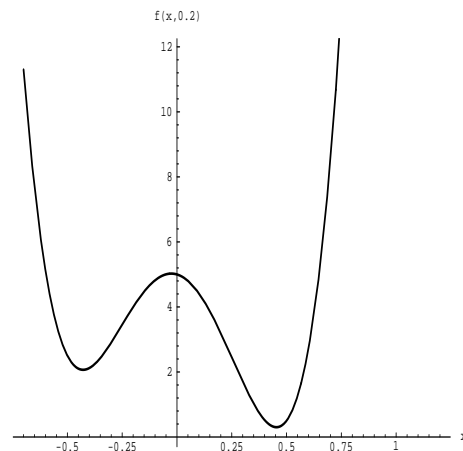


Figura 5.9: $R(x, 0.2)$

Media funcției pe domeniul de căutare este 494.052.

5.4.2 Mediul de calcul distribuit

Un sistem distribuit este alcătuit din mai multe calculatoare independente, conectate în rețea, simulând un calculator paralel. Un mediu distribuit poate fi privit ca un calculator paralel virtual, în general cu structură de tip MIMD cu memorie distribuită.

Mediul de calcul distribuit folosit în experimente este alcătuit din 9 calculatoare PC Pentium (166 MHz, 32 MB RAM), interconectate printr-o magistrală Fast Ethernet de

viteză 100 Mb/s, într-o structură asemănătoare cu structura unui calculator paralel de tip MIMD cu memorie distribuită. Programele au fost scrise în limbajul de programare C, sub sistemul de operare LINUX. Mediul software folosit pentru construirea calculatorului paralel virtual a fost PVM ¹² [53]. El se bazează pe un model de transmisie/recepție de mesaje între calculatoarele care compun mașina virtuală, performanțele globale ale acesteia depinzând atât de performanțele individuale ale fiecărei mașini cât și de viteza cu care este realizată comunicația între mașini.

5.4.3 Studiul parametrilor algoritmului secvențial

În cele ce urmează vom lua pe rând fiecare parametru al algoritmului secvențial și vom face câteva teste numerice pe funcția “six hump” pentru a observa influența parametrului respectiv asupra evoluției procesului de minimizare.

Trebuie să precizăm că algoritmul secvențial se oprește atunci când este atins un număr maxim de generații T , ales în testele prezentate egal cu 300.

- **Mecanismul de selecție și crearea noii generații**

Mecanismul de selecție este cel descris la paragraful 5.2.4. Același mecanism este folosit și în GENOCOP.

Crearea noii generații este făcută diferit față de GENOCOP. Acolo, copii creați se rețin într-o altă structură de date. După crearea tuturor copiilor, indivizi “condamnați la moarte” (care puteau fi alții decât părinții) erau înlocuiți cu copiii. Un copil prost putea astfel înlocui un individ bun. În metoda adoptată pentru crearea noii generații, copiii înlocuiesc imediat părinții (selecția este “on the fly”). Înlocuirea se face doar dacă copilul este mai bun decât părintele, algoritmul având astfel un caracter elitist.

Testele numerice au arătat că acest algoritm are, față de GENOCOP, mai puține eșecuri, iar valoarea celui mai bun individ apare mai repede.

- **Dimensiunea populației**

Pentru a observa influența dimensiunii populației asupra algoritmului, au fost studiate cazurile $POP_SIZE = 25, 49, 100$. În fiecare caz, valorile parametrilor P_i au fost modificate astfel încât procentul de indivizi cărora li se aplică operatorii să rămână aproximativ același.

Pentru ca selecția părinților să se facă aproximativ la fel, valoarea parametrului q , care influențează presiunea de selecție, a fost aleasă astfel încât $cum_probab[POP_SIZE/2] = 0.9$. Deoarece

$$cum_probab[i] = 1 - (1 - q)^i, \quad (5.27)$$

¹²“Parallel Virtual Machine”

rezultă $q = 1 - 0.1^{2/POP_SIZE}$. Ceilalți parametri (neschimbați la variația dimensiunii populației) sunt: $a = 0.25, b = 2, \rho = 10, T = 300$.

Tabelul 5.1 cuprinde valorile parametrilor ajustați după dimensiunea populației, astfel încât testele numerice să reflecte numai influența dimensiunii populației asupra evoluției algoritmului.

POP_SIZE	P_1	P_2	P_3	P_4	P_5	P_6	q
25	1	1	1	2	2	1	0.1682
49	2	2	2	4	4	2	0.089
100	4	4	4	8	8	4	0.045

Tabelul 5.1: Parametrii folosiți în testele cu diferite valori pentru POP_SIZE

costisitoare din punct de vedere al timpului de calcul, vom considera această operație drept operație de referință. În consecință, figura 5.11 (și orice figură care are în axa absciselor numărul de evaluări), o vom privi ca o evoluție în timp.

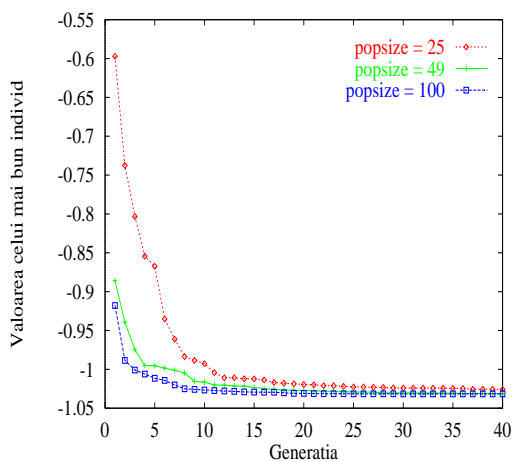


Figura 5.10: Evoluția celui mai bun individ în funcție de numărul generației, pentru diferite valori ale dimensiunii populației

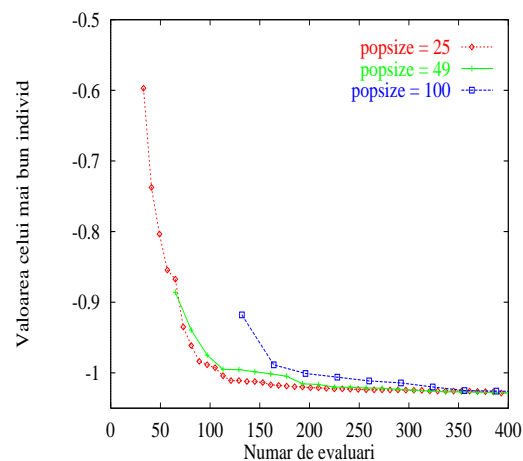


Figura 5.11: Evoluția celui mai bun individ în timp (număr de evaluări), pentru diferite valori ale dimensiunii populației

Este evident că, cu cât dimensiunea populației este mai mare, cu atât valoarea inițială a celui mai bun individ este mai bună (figura 5.10). De asemenea, cu cât dimensiunea populației este mai mare, cu atât indicele generației la care se atinge minimumul este mai mic. Aceasta nu înseamnă însă mai repede. Evoluția în timp (figura 5.11) arată că populații

Figura 5.10 prezintă evoluția celui mai bun individ în funcție de numărul generației curente, iar figura 5.11 evoluția celui mai bun individ în funcție de numărul de evaluări. Deoarece în optimizarea dispozitivelor electromagnetice operația de evaluare a funcției de minimizat este cea mai

cu 49 (sau chiar cu 25) indivizi se comportă mai bine decât populații mari care demarează greu datorită efortului mare de evaluare a populației inițiale. Totuși, algoritmul cu 25 indivizi în populație a înregistrat 2 eșecuri din 20 încercări, pe când cazurile cu 49 și, respectiv 100 indivizi nu au înregistrat nici un eșec în 20 de încercări. Eșecul (în cazul cu 25 indivizi) a însemnat convergența în minimum local egal cu -1 (corespunzător punctului $x = 0, y = 0.7$).

Figurile 5.12 și 5.13 prezintă evoluția în timp a punctului de minim (parametrilor de optimizat).

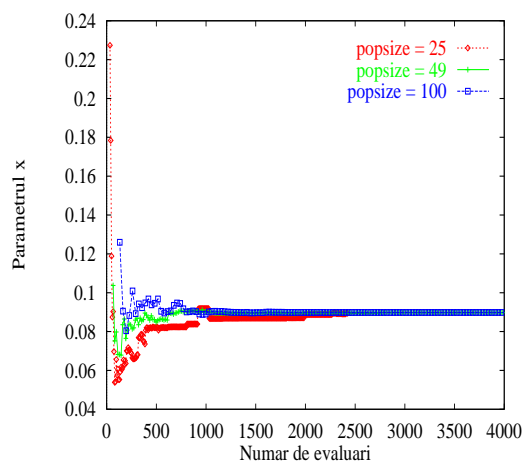


Figura 5.12: Evoluția parametrului x pentru diferite valori ale dimensiunii populației

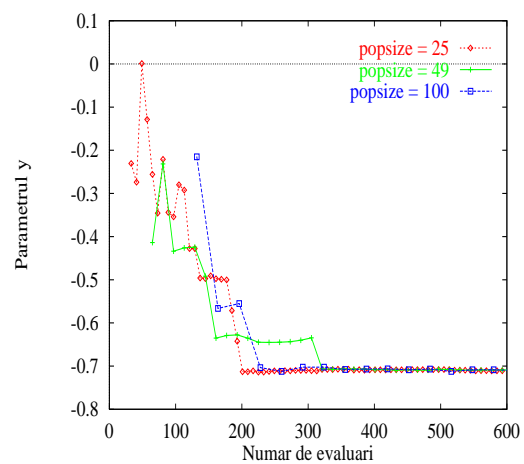


Figura 5.13: Evoluția parametrului y pentru diferite valori ale dimensiunii populației

Aspecte interesante se observă și din inspectarea evoluției în timp a dispersiilor^a funcției obiectiv (figura 5.14), parametrului x (figura 5.15) și parametrului y (figura 5.16). În prima parte a evoluției, populația este dispersată, examinând diverse regiuni ale domeniului de căutare, iar în final se observă îngheșuirea populației spre zona de minim global.

^aDispersia unei mulțimi de valori reale x_1, \dots, x_n a fost calculată cu relația: $\sqrt{[\sum_k (x_k - x_{\text{mediu}})^2] / n} / x_{\text{mediu}}$.

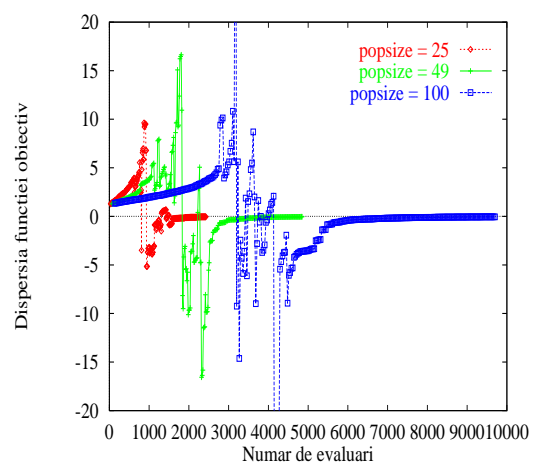


Figura 5.14: Evoluția dispersiei funcției obiectiv pentru diferite valori ale dimensiunii populației

În cazul unor parametri aleși corespunzător, dispersia are după un anumit timp o variație monotonă către zero. Ea nu are valori prea mici deoarece, la algoritmul secvențial, cu q ales ca mai sus, participă practic doar jumătate din populația algoritmului. Dispersia primei jumătăți din populație devine într-adevăr foarte mică (degenerează practic numai jumătate din populație).

• **Parametrii P_i**

Studiul parametrilor P_i a fost făcut pe populația cu 49 de indivizi.

În figurile 5.17, 5.18, 5.19, 5.20 se observă evoluția algoritmului cu 49 indivizi în populație și cu parametrii P_i : 2 2 2 4 4 2 (situație pe care o vom considera ca referință), cu un algoritm în care lipsesc toate mutațiile (P_i : 0 0 0 4 4 0), cu unul în care lipsesc toate încrucișările (P_i : 2 2 2 0 0 2), și unul aleator care face 16 încercări aleatoare pe generație.

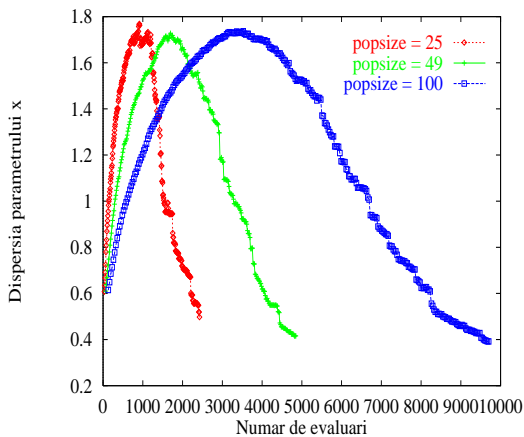


Figura 5.15: Dispersia parametrului x pentru diferite valori POP_SIZE

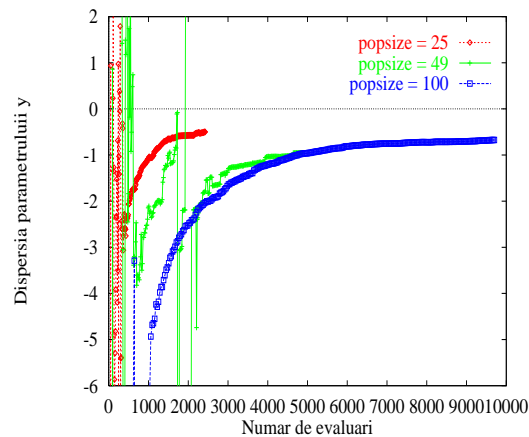


Figura 5.16: Dispersia parametrului y pentru diferite valori POP_SIZE

Din figura 5.17 se observă că algoritmul aleator este cel mai prost, cel mai bun individ tinde să convergă către minim, dar convergența este mult mai lentă decât în cazul algoritmului de referință. În cazul în care lipsesc mutațiile, datorită lipsei de diversitate, algoritmul eșuează de multe ori în minimul local -1 . De aceea, media celor 20 de încercări are alura din figură.

Evoluția dispersiilor funcției obiectiv și parametrilor sugerează detalii suplimentare despre convergența algoritmilor. Se observă clar caracterul aleator pe graficele dispersiilor funcției obiectiv și parametrului x . De asemenea, evoluția dispersiilor funcției obiectiv, parametrilor x și y pentru cazul în care lipsesc încrucișările, arată că procesul nu ar fi trebuit oprit. În cazul în care lipsesc mutațiile, algoritmul este convergent, dar el a convers în majoritatea cazurilor către un minim local.

În figurile 5.21, 5.22, 5.23, 5.24 se observă evoluția algoritmului de referință și a algoritmilor în care, pe rând, se renunță la aplicarea unuia din cele 4 tipuri de mutații.

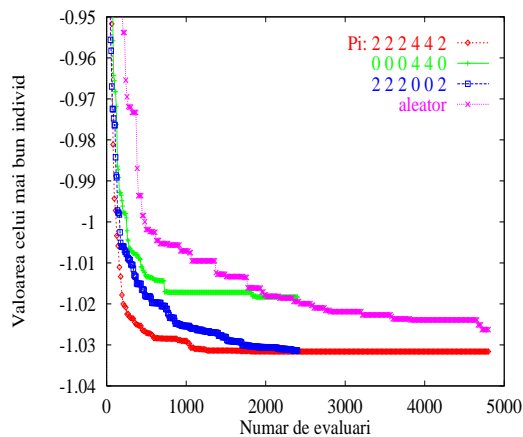


Figura 5.17: Cel mai bun individ: referință; fără mutații; fără încrucișări.

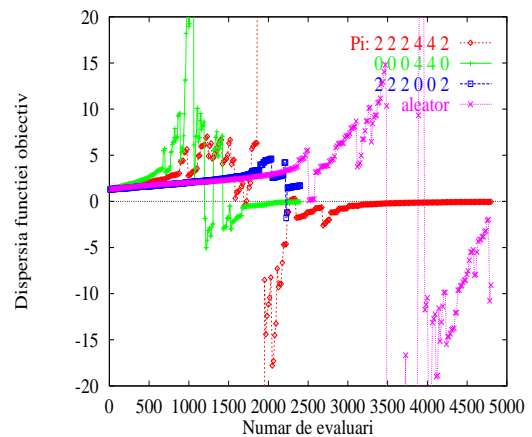


Figura 5.18: Dispersia funcției: referință; fără mutații; fără încrucișări.

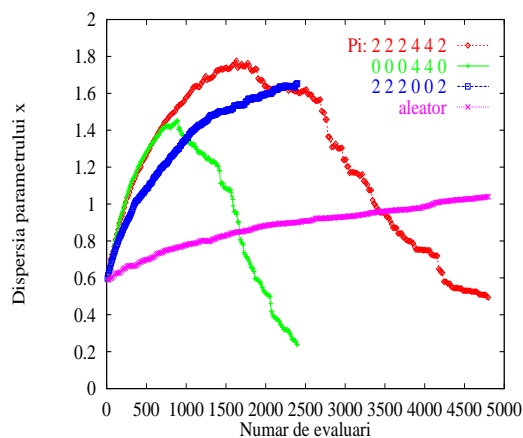


Figura 5.19: Dispersia parametrului x : referință; fără mutații; fără încrucișări.

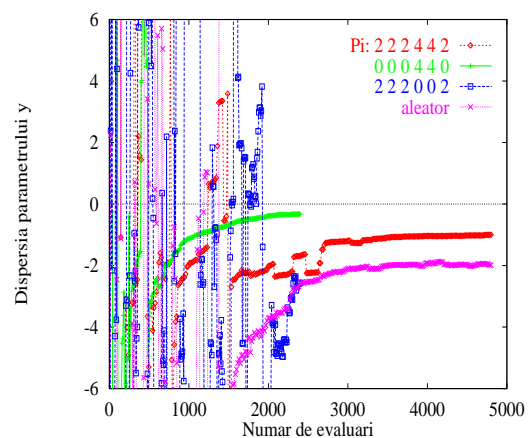


Figura 5.20: Dispersia parametrului y : referință; fără mutații; fără încrucișări.

Oricare din cazurile $P_i = 0$ unde $i = 1, 2, 3$ sau 6 converge mai repede decât situația de referință. Se pare că, situația de referință are prea multe mutații pe generație. Nu se poate spune clar care mutație este esențială. Lipsa tuturor este însă catastrofală. Se știe că rolul operatorilor este diferit pe parcursul evoluției.

În figurile 5.25, 5.26, 5.27, 5.28 se observă evoluția algoritmului de referință și a algoritmilor în care, pe rând, se renunță la aplicarea unuia din cele 2 tipuri de încrucișări.

Din punct de vedere al evoluției celui mai bun individ, algoritmi în care P_4 , respectiv P_5 sunt zero se comportă asemănător cu algoritmul de referință. Graficele dispersiilor funcției obiectiv și parametrului x arată că în cazul lipsei operatorului încrucișare aritmetică ($P_4 = 0$), care genera copii în interiorul unui segment ce unea părinții, algoritmul nu ajunge la convergență în intervalul de timp impus.

- Parametrul q

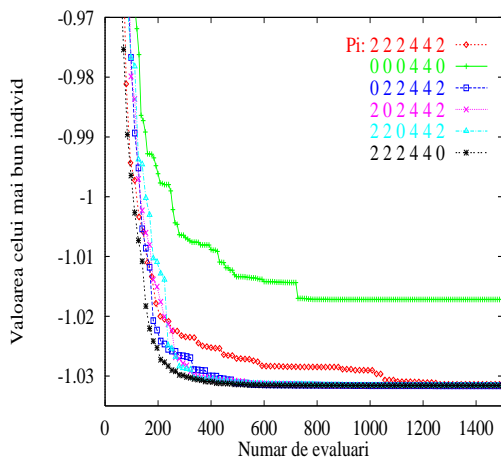


Figura 5.21: Cel mai bun individ: algoritm de referință; $P_1 = 0$; $P_2 = 0$; $P_3 = 0$; $P_6 = 0$.

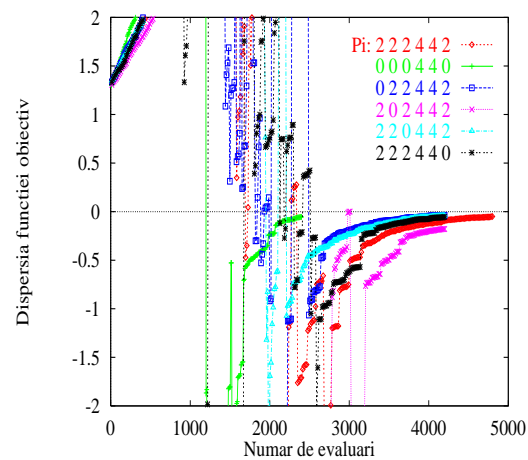


Figura 5.22: Dispersia funcției obiectiv: algoritm de referință; $P_1 = 0$; $P_2 = 0$; $P_3 = 0$; $P_6 = 0$.

Parametrul q (număr real cuprins între 0 și 1) este cel care influențează presiunea de selecție. Influența lui a fost studiată pentru populația cu 49 de indivizi (figurile 5.29, 5.30, 5.31, 5.32).

Din graficele dispersiei funcției obiectiv și parametrului x , se observă că valoarea $q = 0.2$ este cea mai nesatisfăcătoare, ea nu conduce la convergența algoritmului în timpul impus. Valoarea $q = 0.2$ corespunde unei presiuni de selecție mare, în care părinții dintr-o generație sunt selectați practic numai din indivizii foarte buni. De asemenea, valoarea $q = 0.05$ este mai bună decât valoarea $q = 0.089$. Aceasta înseamnă că presiunea de selecție este mai scăzută, putând fi aleși drept părinți și indivizi mai slabi.

• Parametrul b

Parametrul b (număr real mai mare decât 1) este cel care determină legea de variație ce intervine în mutația neuniformă. Cu cât el este mai mare, scăderea mărimii $[1 - (t/T)^b]$ este mai abruptă. Figurile 5.33, 5.34, 5.35, 5.36 arată rezultatele testelor efectuate pentru cazurile $b = 1, 2, 4$.

Rezultatele nu sunt foarte relevante. Valorile $b = 1$ și $b = 4$ par a fi mai bune decât valoarea $b = 2$.

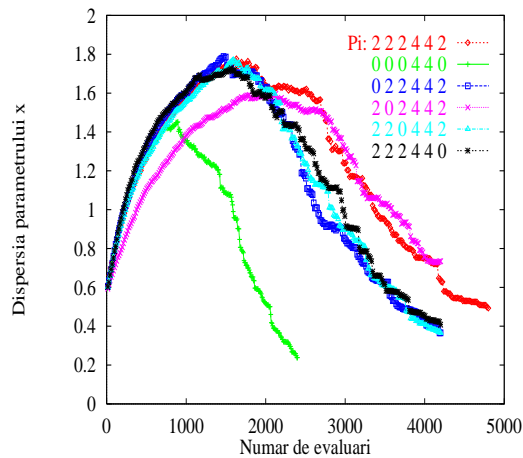


Figura 5.23: Dispersia parametrului x ; algoritm de referință; $P_1 = 0$; $P_2 = 0$; $P_3 = 0$; $P_6 = 0$.

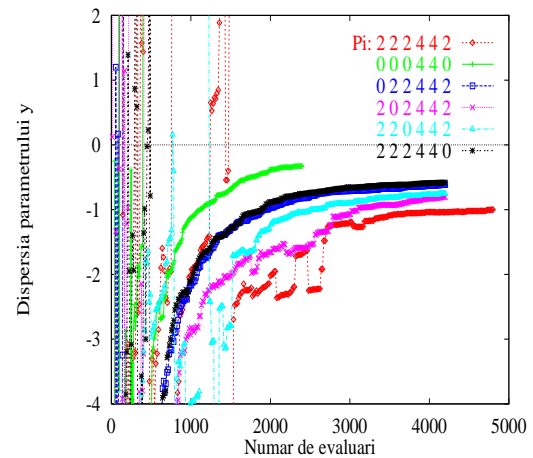


Figura 5.24: Dispersia parametrului y ; algoritm de referință; $P_1 = 0$; $P_2 = 0$; $P_3 = 0$; $P_6 = 0$.

5.4.4 Algoritmul distribuit în comparație cu algoritmul secvențial

Figura 5.37 prezintă evoluțiile unui algoritm secvențial cu 100 indivizi, un algoritm distribuit în care 4 subpopulații de câte 25 de indivizi sunt conectate într-o topologie în inel și un algoritm secvențial cu 25 de indivizi. Este evident că algoritmul distribuit converge mult mai repede decât algoritmul secvențial cu 100 indivizi. Se observă că o populație care cooperează cu altele converge mult mai repede decât în cazul în care ar fi izolată (algoritmul secvențial cu 25 indivizi).

Mai mult, în cazul algoritmului distribuit nu s-a înregistrat nici un eșec, așa cum s-a întâmplat în cazul algoritmului secvențial cu 25 indivizi în subpopulație.

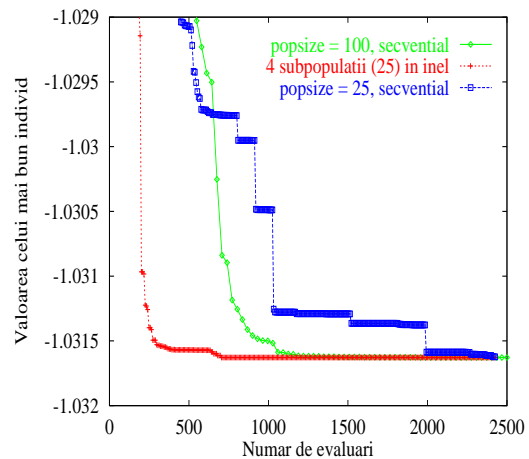


Figura 5.37: Evoluția celui mai bun individ: algoritmul secvențial față de cel distribuit

5.4.5 Studiul parametrilor algoritmului distribuit

În cele ce urmează vom prezenta rezultatele testelor pe un algoritm distribuit cu topologie în inel, în care dimensiunea unei subpopulații este de 25 de indivizi iar algoritmul se oprește

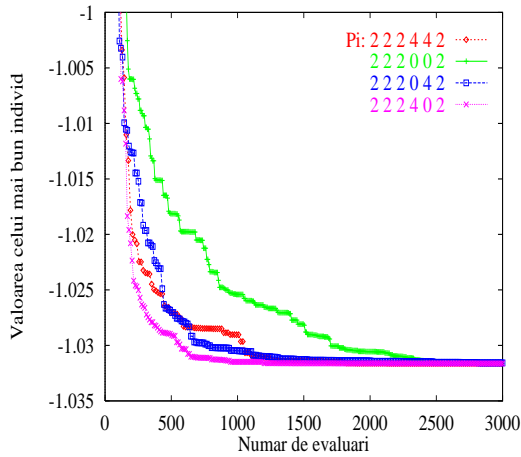


Figura 5.25: Cel mai bun individ: algoritm de referință; $P_4 = 0$; $P_5 = 0$.

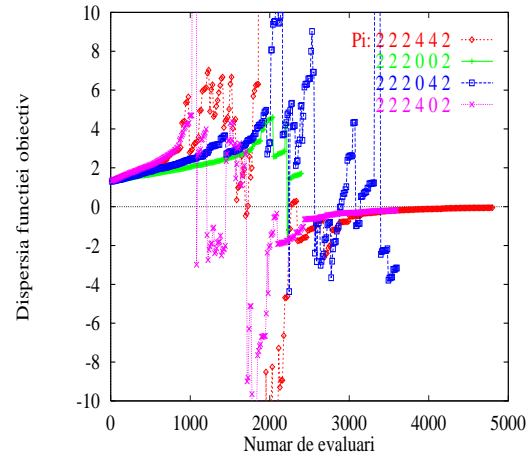


Figura 5.26: Dispersia funcției de cost: algoritm de referință; $P_4 = 0$; $P_5 = 0$.

atunci când fiecare subpopulație a efectuat 300 de generații¹³. La o astfel de dimensiune a subpopulației este rezonabil să migreze un singur individ (cel mai bun). El va înlocui un individ din populația destinație doar dacă este mai bun decât el.

Insulele sunt omogene și sincrone. A mai fost implementată și o topologie dinamică, stabilită la fiecare migrație în funcție de calitatea subpopulației. Rezultatele obținute (pentru numărul mic de insule folosit) nu au indicat diferențe radicale față de cazul topologiei în inel și de aceea nu le vom prezenta în continuare. O topologie dinamică ar putea avea efecte benefice în cazul folosirii mai multor insule.

• Cine este înlocuit la migrație?

Figurile 5.38, 5.39, 5.40, 5.41 prezintă rezultatele testelor în care un emigrant înlocuiește (dacă este mai bun) pe cel mai bun individ din populația destinație, pe cel mai prost, sau unul aleator.

Din punct de vedere al evoluției celui mai bun individ, rezultatele sunt asemănătoare. Se poate spune că înlocuirea celui mai prost individ face că o valoare foarte bună să apară mai repede. Din punct de vedere al evoluției dispersiei funcției obiectiv se observă că, înlocuirea celui mai prost individ face că populația să degenereze foarte repede, făcându-ne să suspectăm o convergență prematură a algoritmului. Populația degenerază ceva mai târziu în cazul înlocuirii unui individ aleator și mai greu în cazul înlocuirii celui mai bun. Este normal să se întâmple așa datorită mecanismului de selecție și crearea a unei noi generații. Prin selecție se aleg numai indivizi din prima jumătate a populației. Dacă

¹³Având în vedere că urmărim aplicarea unui astfel de algoritm la optimizarea dispozitivelor electromagnetice, nu am dorit să studiem cazuri cu mai mulți indivizi în subpopulație și nici cu număr maxim de generații mai mare. Astfel de cazuri ar duce la creșterea substanțială a efortului de calcul.

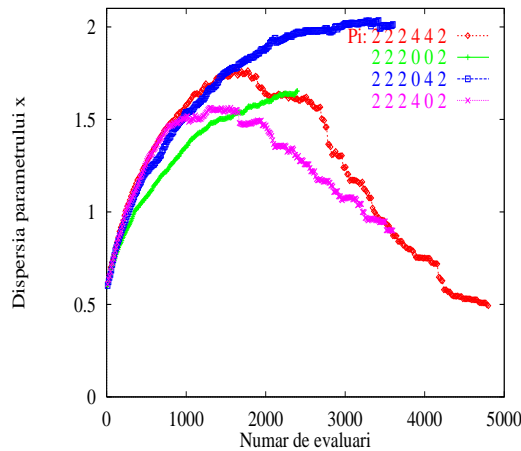


Figura 5.27: Dispersia parametrului x ; algoritm de referință; $P_4 = 0$; $P_5 = 0$.

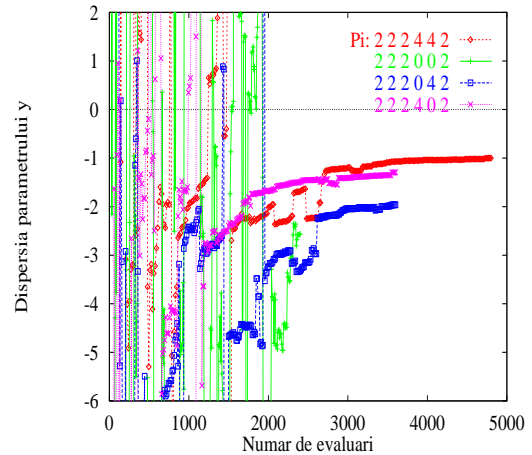


Figura 5.28: Dispersia parametrului y ; algoritm de referință; $P_4 = 0$; $P_5 = 0$.

se înlocuiește tot timpul cel mai bun, practic numai jumătate din populație participă la algoritmi. Înlocuirea celui mai prost sau unuia aleator dă șanse și jumătății de jos a populației să se îmbunătățească.

Supraconvergența în cazul înlocuirii celui mai prost individ o suspectăm și urmărind graficele dispersiei parametrilor (figurile 5.40 și 5.41).

De asemenea, din graficele dispersiilor, observăm că în cazul înlocuirii celui mai bun individ, dispersiile nu ajung prea mici (așa cum se întâmplă și în algoritmul secvențial). Motivul este din nou acela că practic numai jumătate din populație participă la algoritm. În cazul înlocuirii celui mai prost individ sau unuia aleator dispersiile sunt foarte mici, practic nule, populația este într-adevăr degenerată și continuarea procesului evoluționist nu mai are sens.

În concluzie, cel mai bine este ca un emigrant să înlocuiască (dacă este mai bun) un individ ales aleator din populația destinație. Așa vom proceda în toate testele următoare.

• Numărul de insule

Figurile 5.42, 5.43, 5.44, 5.45 prezintă rezultatele testelor în care sunt folosite 2, 4, 6 și respectiv 10 insule. Algoritmul folosește 25 indivizi într-o subpopulație, iar perioada migrației este 3.

Se observă că oricare algoritm distribuit este mai bun decât cel secvențial din punct de vedere al evoluției celui mai bun individ. Algoritmul secvențial a avut 2 eșecuri din 20 de încercări, ca și algoritmul distribuit cu 2 insule. Nu s-au înregistrat eșecuri în celelalte cazuri.

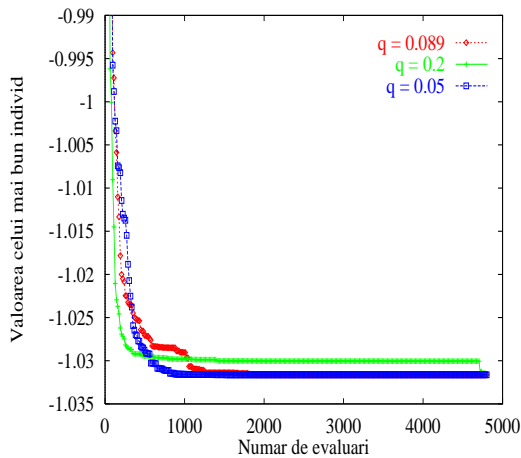


Figura 5.29: Cel mai bun individ: $q = 0.089$; $q = 0.2$; $q = 0.05$.

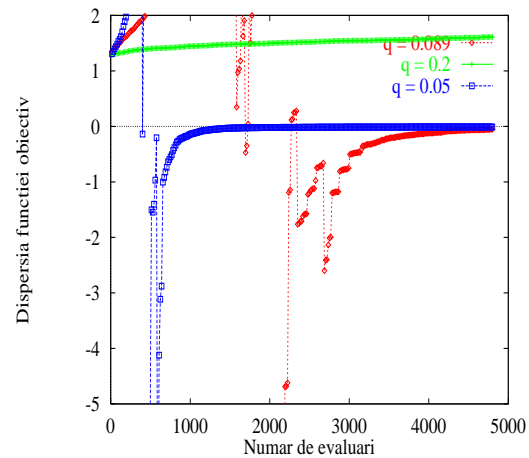


Figura 5.30: Dispersia funcției de cost: $q = 0.089$; $q = 0.2$; $q = 0.05$.

Algoritmii cu 4, 6 și respectiv 10 insule se comportă aproximativ la fel. De aceea, efortul necesar pentru creșterea numărului de insule (procesoare în plus, timp de comunicație mai mare), nu se justifică peste un anumit număr. Considerăm de aceea că un număr de 4 insule este rezonabil, cel puțin pentru problema de test considerată.

• Perioada migrației

Figurile 5.46, 5.47, 5.48, 5.49 prezintă rezultatele testelor unui algoritm distribuit cu 4 insule sincrone, omogene, în care perioada migrației este 1, 3, respectiv 6.

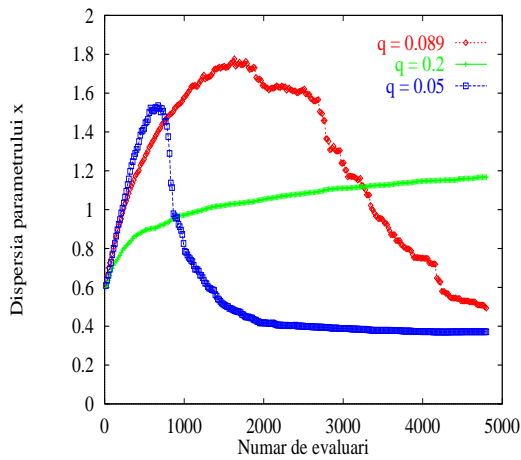


Figura 5.31: Dispersia parametrului x :
 $q = 0.089$; $q = 0.2$; $q = 0.05$.

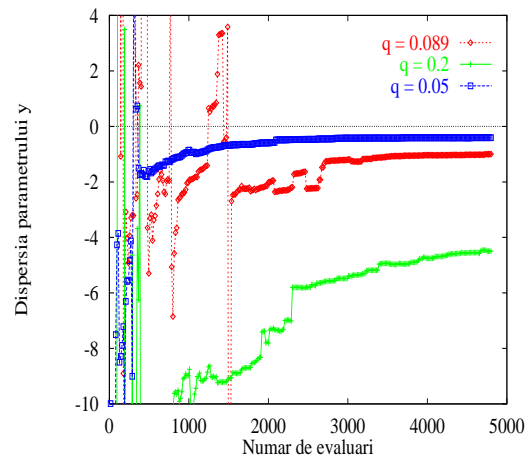


Figura 5.32: Dispersia parametrului y :
 $q = 0.089$; $q = 0.2$; $q = 0.05$.

Se observă că algoritmul cu perioada 1 este cel mai rapid convergent, poate prea rapid, și l-am putea suspecta de supraconvergență. În plus, migrarea indivizilor la fiecare generație necesită un timp de comunicație mai mare, care ar putea deveni semnificativ.

Cazul în care perioada migrației este 6 se comportă mai prost decât cazul cu perioada 3, atât din punct de vedere al convergenței cât și al eșecurilor. Considerăm de aceea că perioada 3 este rezonabilă, cel puțin pentru această problemă de test.

5.5 Algoritmi evoluționiști în optimizarea dispozitivelor electromagnetice

5.5.1 Funcții de test

Am văzut că alegerea parametrilor algoritmului evoluționist este deosebit de importantă. Testele numerice efectuate relevă faptul că valorile acestor parametri depind de funcția de test folosită. În cazul optimizării dispozitivelor electromagnetice, este posibil ca, pentru dispozitive complexe, evaluarea funcției obiectiv să fie costisitoare din punct de vedere al timpului de calcul. Dacă algoritmul de optimizare nu este adaptiv, atunci trebuie făcute mai multe teste, pentru valori diferite ale parametrilor. Dacă evaluarea funcției obiectiv durează mult (mult înseamnă chiar câteva minute), atunci toate aceste teste sunt foarte mari consumatoare de timp.

De aceea, pentru a acorda parametrilor algoritmului folosit pentru optimizare, propunem să se folosească în loc de funcția obiectiv F , o funcție de test F' , cu o expresie foarte simplă.

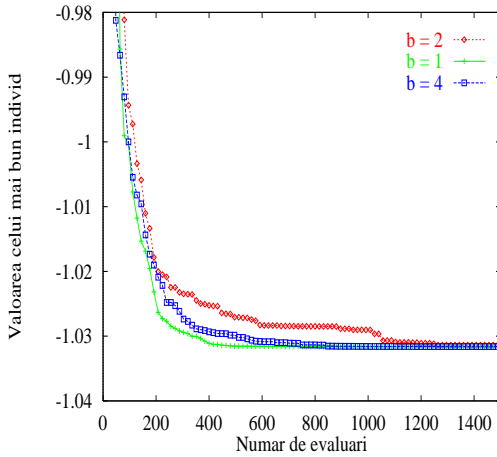


Figura 5.33: Cel mai bun individ: $b = 1; b = 2; b = 4$.

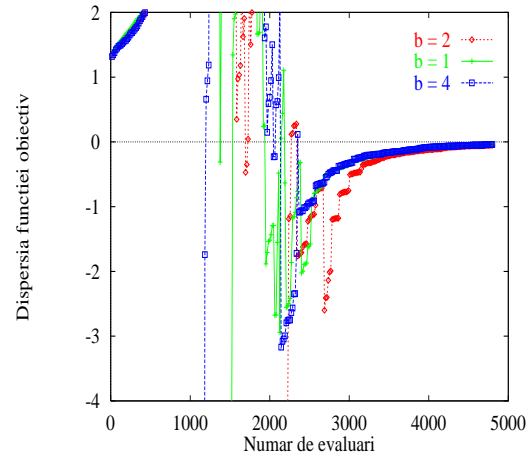


Figura 5.34: Dispersia funcției de cost: $b = 1; b = 2; b = 4$.

O astfel de funcție trebuie să aibă același număr de parametri ca și F și să aibă un relief asemănător cu cel al funcției F .

Pentru definirea unei astfel de funcții de test, definim o funcție auxiliară $f : [0, 1] \rightarrow \mathbb{R}$ ce caracterizează relieful funcției F , astfel:

- Funcția obiectiv F este evaluată în $n+1$ puncte $\mathbf{x}_k (k = 0, \dots, n)$ din domeniul de căutare;
- Cele $n + 1$ valori $F_k = F(\mathbf{x}_k)$ sunt sortate descrescător, iar șirul \mathbf{x}_k se renumerează. Fie $F_M = F(\mathbf{x}_0)$ cea mai mare valoare găsită și $F_m = F(\mathbf{x}_n)$ cea mai mică;
- Expresia funcției $f(\alpha)$ este

$$f(\alpha) = F_{\text{Int}(\alpha n)}. \quad (5.28)$$

Funcția f este o funcție în trepte, monoton descrescătoare de la F_M la F_m . De asemenea, atunci când n tinde la infinit, $(\forall) \alpha_1, \alpha_2 \in [0, 1]$ cu $\alpha_1 < \alpha_2$, diferența $\alpha_2 - \alpha_1$ reprezintă probabilitatea de a găsi în domeniul de căutare \mathcal{D} o valoare pentru F în intervalul $[f(\alpha_2), f(\alpha_1)]$. De aceea vom numi funcția f **funcția caracteristică a reliefului** funcției F .

Figura 5.50 prezintă funcțiile caracteristice reliefului a trei funcții de test binecunoscute: funcția “six-hump”, funcția Bohachevsky și funcția lui Schaffer precum și funcția caracteristică reliefului problemei TEAM Workshop 22 (cazul cu 3 parametri).

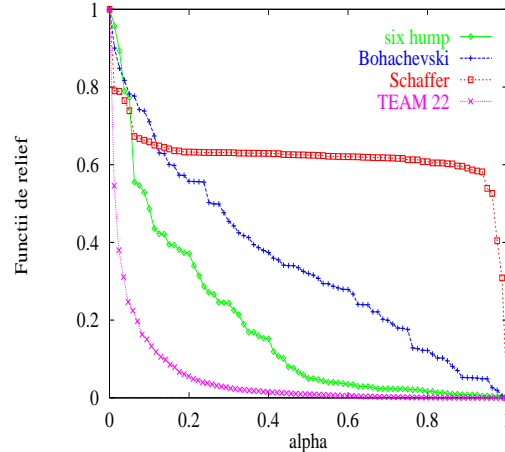


Figura 5.50: Funcții caracteristice de relief, mapate pe intervalul $[0,1]$

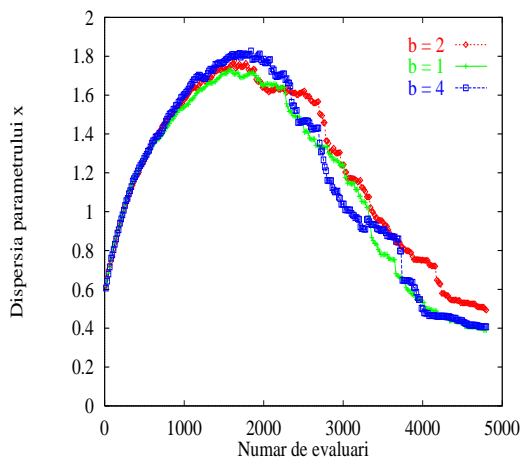


Figura 5.35: Dispersia parametrului x :
 $b = 1; b = 2; b = 4$.

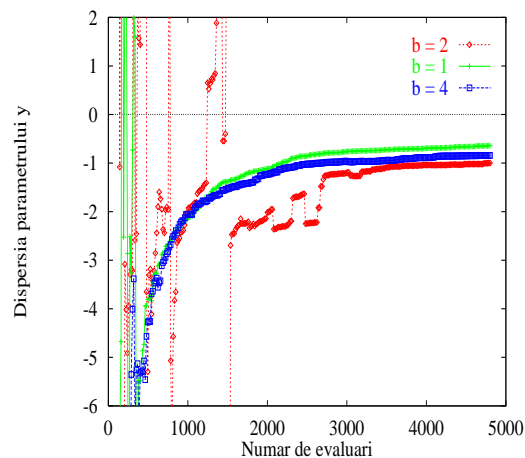


Figura 5.36: Dispersia parametrului y :
 $b = 1; b = 2; b = 4$.

Să considerăm l “puncte esențiale” ale graficului lui f . Prin aceasta înțelegem o mulțime de l puncte din care *funcția caracteristică a reliefului* poate fi reconstruită cu o eroare impusă (de exemplu prin interpolare liniară pe porțiuni). De exemplu, dacă $l = 3$ putem lua α egal cu 0, 0.5 și respectiv 1. Valorile F_k și punctele corespunzătoare $\mathbf{x}_k, k = 1, l$ reprezintă “puncte esențiale” în relieful funcției F . Utilizând aceste puncte, se construiește următoarea **funcție de test**:

$$F'(\mathbf{x}) = \sum_{k=1}^l f_k \prod_{\substack{j=1 \\ j \neq k}}^l \frac{\|\mathbf{x} - \mathbf{x}_j\|^2}{\|\mathbf{x}_k - \mathbf{x}_j\|^2}. \quad (5.29)$$

Funcția de test are următoarele proprietăți:

- F și F' au același număr de variabile q ;
- F' este o interpolare în l puncte a funcției obiectiv F ;
- Efortul de calcul necesar evaluării funcției de test F' este mult mai mic decât pentru F ;
- F' și F au același minim global când $n \rightarrow \infty$.

Pe scurt, F' este o aproximare pentru F , ambele funcții având aceleași valori în $\mathbf{x}_k, k = 1, l$. De aceea, adăugarea în lista punctelor esențiale a ultimelor puncte din lista sortată ameliorează aproximarea din vecinătatea minimului.

5.5.2 Strategie de control optimal

Presupunând că algoritmul evoluționist are parametri potriviți pentru problema de rezolvat, dacă evaluarea funcției obiectiv este costisitoare, algoritmul ar putea dura foarte mult (numărul de evaluări de funcții fiind cel puțin de ordinul miilor). În vederea reducerii

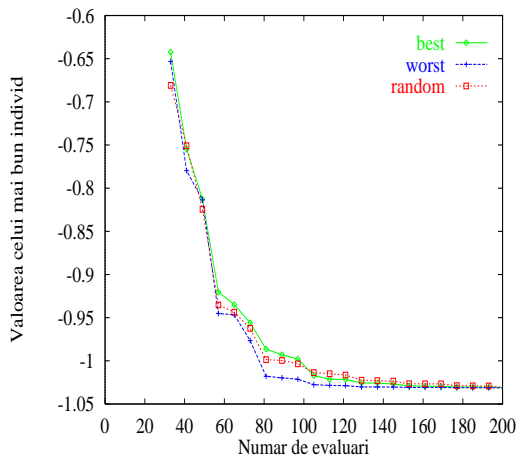


Figura 5.38: Cel mai bun individ: se înlocuiește cel mai bun; cel mai rău; unul aleator

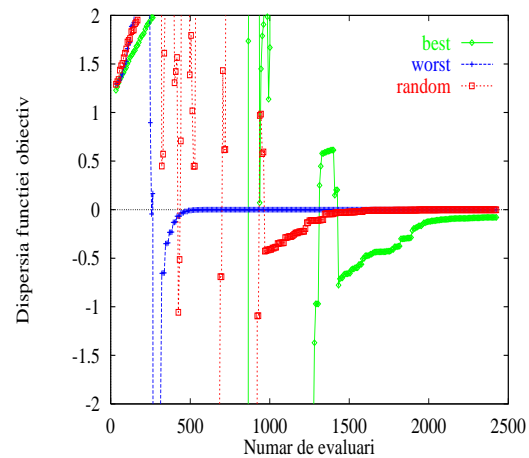


Figura 5.39: Dispersia funcției obiectiv: se înlocuiește cel mai bun; cel mai rău; unul aleator

timpului de calcul, propunem ca acuratețea evaluării funcției obiectiv să crească treptat în timpul algoritmului de optimizare. Pentru stabilirea strategiei de control a acurateții vom folosi de asemenea funcția de test F' .

Vom aborda problemele de optimizare a dispozitivelor electromagnetice în felul următor:

- **Pasul 1 - Construirea funcției de test.**

1.1. Funcția caracteristică a reliefului. Funcția obiectiv F este evaluată în $n = 2^q$ puncte (nu mai puțin de 20), împrăștiate aleator în întreg domeniul de căutare. Cu aceste informații se construiește funcția caracteristică a reliefului f ca în relația (5.28).

1.2. Funcția de test. Alegând l puncte esențiale din funcția caracteristică a reliefului (recomandăm $l \leq q + 1$), funcția de test F' se construiește ca în relația (5.29).

- **Pasul 2 - Acordarea algoritmului de optimizare.**

Parametrii algoritmului de optimizare se determină făcând teste pe funcția F' .

- **Pasul 3 - Strategia de control optimal.**

3.1. Stabilirea unei relații între timpul necesar evaluării funcției obiectiv și acuratețe. Într-un punct arbitrar \mathbf{x} din domeniul de căutare funcția obiectiv F este evaluată pentru diferite grade de finețe a rețelei de discretizare $j = 1, j_m$ (recomandăm $j_m \leq 6$) și se contorizează timpii de calcul T_j . Se evaluează erorile de discretizare ε_j ale funcției F . Se face o regresie a datelor obținute din ierarhia de discretizări și, aproximând relația dintre timpul de calcul și acuratețe ca fiind $T = C\varepsilon^{-m}$, se determină constantele C și m .

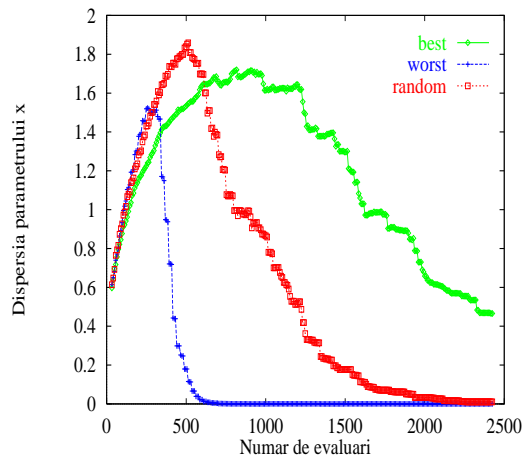


Figura 5.40: Dispersia parametrului x : se înlocuiește cel mai bun; cel mai rău; unul aleator

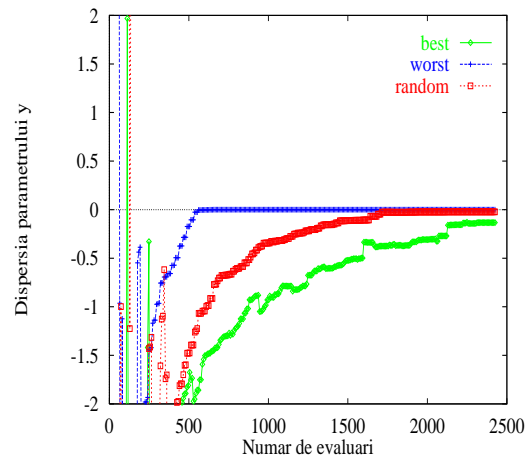


Figura 5.41: Dispersia parametrului y : se înlocuiește cel mai bun; cel mai rău; unul aleator

3.2. Strategia de control a preciziei. Considerând k generația curentă, se deduce variația $\varepsilon(k)$ pentru a obține timpul minim echivalent pentru minimizarea funcției F' . Se simulează pierderea de acuratețe în evaluarea funcției F' , iar timpul echivalent de calcul se calculează cu relația $T = C\varepsilon^{-m}$. Funcția $\varepsilon(k)$ trebuie să aibă valori suficient de mici la terminarea algoritmului.

- **Pasul 4 - Optimizarea finală.**

Se rezolvă problema reală. Funcția F se optimizează cu algoritmul stocastic având parametrii obținuți la pasul 2, acuratețea evaluării funcției obiectiv fiind crescută treptat în timpul algoritmului de optimizare, conform strategiei de control optimal stabilită la pasul 3. Populația inițială este cea folosită la pasul 1.

5.6 Concluzii

Algoritmii evoluționiști sunt capabili, în general, să găsească soluții bune într-un timp rezonabil. Deoarece ei încep să fie aplicați la probleme din ce în ce mai dificile (printre care și optimizarea dispozitivelor electromagnetice), există o creștere a timpului necesar găsirii soluției adecvate. De aceea, s-au făcut multiple eforturi pentru a face algoritmii genetici mai rapizi, și una dintre cele mai promițătoare alegeri constă în utilizarea implementărilor paralele.

Algoritmul de bază al unei implementări paralele este un algoritm evoluționist secvențial. Pentru acesta trebuie stabiliți: *mecanismul de selecție* - care identifică cei mai potriviți indivizi pentru a fi părinți, *operatori de încrucișare* - operatori primari ce explorează spațiul

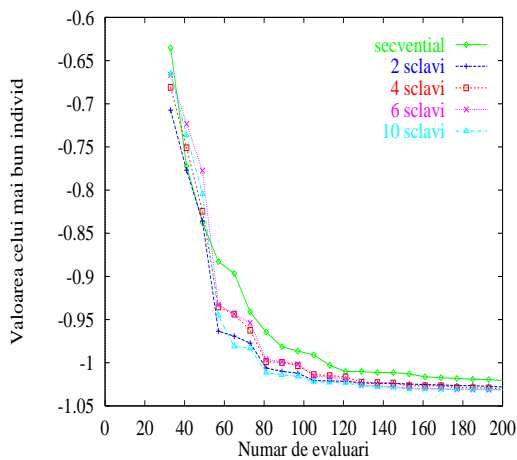


Figura 5.42: Cel mai bun individ: algoritm secvențial; 2, 4, 6, 10 insule

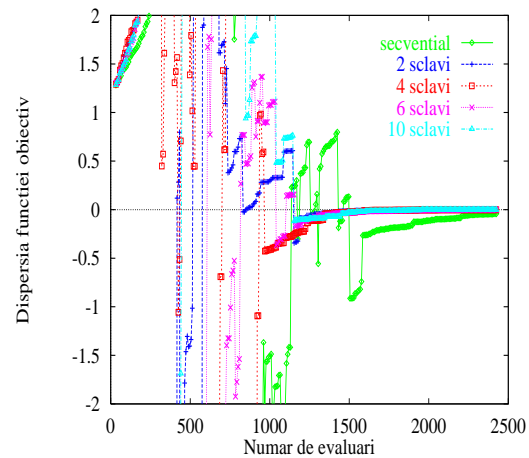


Figura 5.43: Dispersia funcției obiectiv: algoritm secvențial; 2, 4, 6, 10 insule

de căutare, *operatori de mutație* - care asigură diversitatea populației. O decizie deosebit de importantă o constituie alegerea *dimensiunii populației*. Goldberg afirmă că timpul cerut de un algoritm genetic să convergă este $O(n \log(n))$ evaluări de funcții, unde n este dimensiunea populației. Se spune că o populație a convergat atunci când toți indivizii sunt asemănători și o viitoare îmbunătățire este posibilă doar printr-o mutație. Algoritmii evoluționiști nu garantează găsirea soluției optime, dar cu cât n crește, cu atât șansa de a găsi soluția globală este mai mare (într-un timp mai mare). Cu un algoritm secvențial se pot obține rezultate bune și de încredere într-un timp mare. Reducerea timpului se face pe seama pierderii încrederii în rezultat.

În general sunt dificil de găsit parametri potriviți pentru un algoritm evoluționist, toate teoriile dezvoltate până acum referindu-se exclusiv la cazul algoritmilor genetici, în care codificarea indivizilor se face binar. Stabilirea lor se face mai ales după fler și intuiție. A început însă să se practice folosirea unor parametri adaptivi. Se folosește un meta-algoritm genetic care optimizează parametri ce se aplică algoritmului evoluționist principal.

Spre deosebire de implementările secvențiale, cele paralele pot găsi repede rezultate în care se poate avea încredere. Cea mai populară implementare este aceea care constă în mai multe subpopulații care evoluează separat și schimbă, ocazional, indivizi. Acest tip de implementare se numește cu granularitate mare, sau implementare distribuită (deoarece, de cele mai multe ori, este implementată într-o arhitectură distribuită de calculatoare, de tip MIMD). Implementările paralele ale modelului migrației au arătat că găsirea soluției globale are nevoie de mai puține evaluări de funcții decât un algoritm cu o singură populație. Astfel, cu algoritmii distribuiți, pentru orice funcție testată au fost obținute rezultate mai

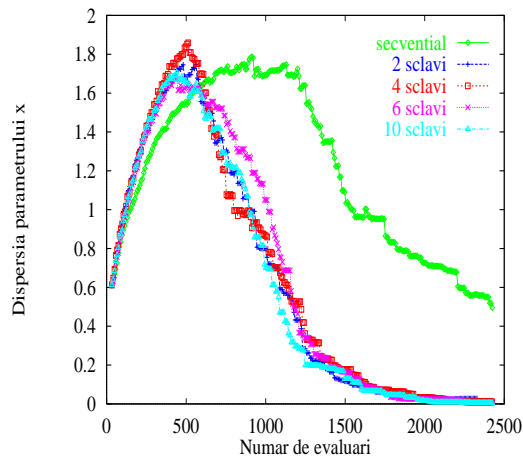


Figura 5.44: Dispersia parametrului x :
algoritm secvențial; 2, 4, 6, 10 insule

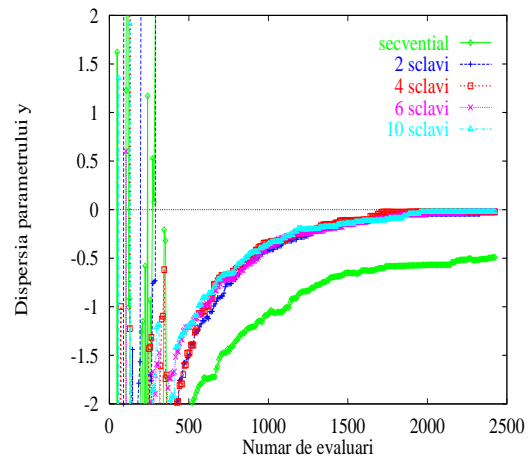


Figura 5.45: Dispersia parametrului y :
algoritm secvențial; 2, 4, 6, 10 insule

bune decât pentru o singură populație cu proporțional mai mulți indivizi.

Algoritmii evoluționiști paraleli sunt foarte complecși și există multe probleme încă nerezolvate, de exemplu:

1. Care este rata de de migrație optimă?
2. Care este topologia de comunicație adecvată care să permită amestecarea soluțiilor bune, dar care să nu ducă la costuri de comunicație excesive?
3. Care este numărul de subpopulații (și dimensiunea lor) care maximizează încrederea în rezultat?

Cele mai recente studii au arătat că îmbunătățirea drastică a unui algoritm evoluționist secvențial se face folosind:

- implementari paralele;
- selecție de tip turneu;
- niching;
- elitism.

Scopul acestui capitol a fost să privească și să înțeleagă algoritmi evoluționisti de optimizare. Testele numerice efectuate relevă importanța alegerii unor parametri corecți pentru un astfel de algoritm.

Cea mai importantă concluzie este aceea că un algoritm distribuit, în care subpopulațiile schimbă informații, se comportă mai bine decât un algoritm secvențial cu o populație mai mare. Există o valoare critică pentru numărul de subpopulații cât și pentru perioada migrației. Topologia se pare că nu contează prea mult la un număr mic de insule.

Dacă algoritmul nu este adaptiv, un sfat unanim acceptat, este acela că, în vederea creșterii încrederii în rezultate, un algoritm stocastic trebuie executat de mai multe ori

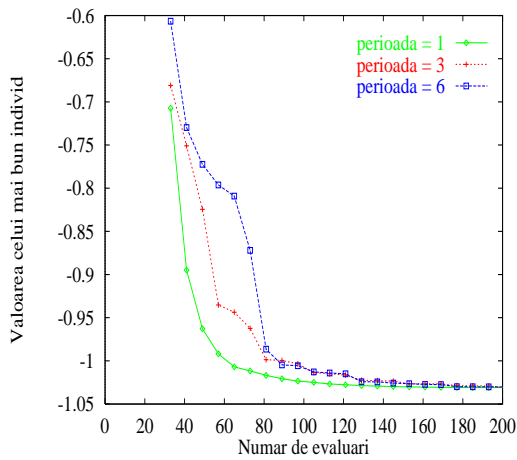


Figura 5.46: Cel mai bun individ: algoritm secvențial; perioada = 1, 3, 6.

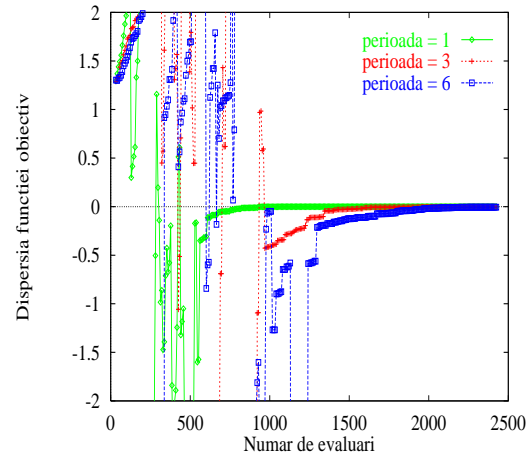


Figura 5.47: Dispersia funcției obiectiv: algoritm secvențial; perioada = 1, 3, 6.

(pentru aceeași parametri), și de asemenea cu mai multe seturi de parametri. În cazul aplicării unor astfel de algoritmi la optimizarea dispozitivelor electromagnetice, problema cea mai mare este legată de durata evaluării funcției obiectiv. Dacă aceasta este mare, toate aceste teste sunt foarte mari consumatoare de timp. De aceea, se obișnuiește să se folosească funcții de test cunoscute, cu o expresie algebrică simplă. Acest capitol propune construirea unor funcții de test adaptate problemei de rezolvat pentru determinarea parametrilor algoritmului.

Chiar și așa, cu parametri corecți, un algoritm evoluționist aplicat optimizării unui dispozitiv electromagnetic poate dura foarte mult. Deoarece la începutul algoritmului se explorează spațiul de căutare, calculul foarte precis al funcției obiectiv nu este folosit. De aceea, am propus o strategie în care precizia evaluării funcției obiectiv crește pe parcursul algoritmului, astfel încât, la terminarea acestuia, funcția obiectiv este evaluată foarte precis. O astfel de abordare este folosită în optimizarea dispozitivelor electromagnetice prezentate în capitolul 6.

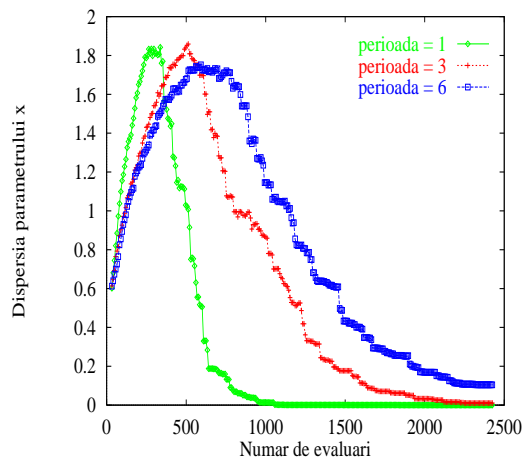


Figura 5.48: Dispersia parametrului x :
 algoritm secvențial; perioada = 1, 3, 6.

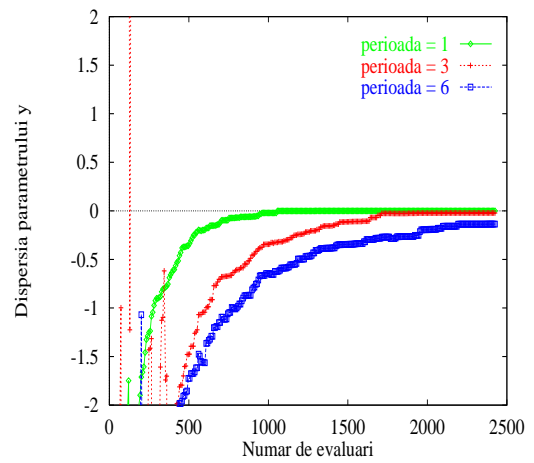


Figura 5.49: Dispersia parametrului y :
 algoritm secvențial; perioada = 1, 3, 6.

Capitolul 6

Rezultate privind optimizarea dispozitivelor electromagnetice

6.1 Problema TEAM 22

Problema TEAM Workshop 22 constă în optimizarea unui dispozitiv SMES¹. Dispozitivele SMES sunt dispozitive care stochează energia în câmpuri magnetice. În principiu ele sunt construite din bobine realizate din materiale supraconductoare. Bobinele sunt alimentate printr-un comutator de la un convertizor de putere, după care comutatorul se deschide simultan cu scurtcircuitarea bornelor bobinelor. Curentul circulă în bobine fără a scădea în timp datorită rezistenței nule a supraconductoarelor. Astfel de dispozitive pot fi folosite pentru stabilizarea fluctuațiilor de putere în sistemele energetice.

Există două tipuri diferite de bobine folosite în dispozitivele SMES: solenoizii și toroizii. Tehnica de realizare a unei bobine solenoidale este foarte simplă, în timp ce realizarea unei bobine toroidale este mult mai sofisticată și necesită o cantitate de material supraconductor aproape de două ori mai mare. Avantajul unei bobine toroidale constă în faptul că, datorită geometriei sale, câmpul magnetic în spațiul înconjurător este practic nul. Acest rezultat este valabil în cazul în care bobina toroidală este înfășurată perfect, lucru care nu se realizează în practică deoarece toroizii sunt construiți din mai mulți solenoizi plasați pe o formă de tor. Cu toate acestea, un astfel de toroid produce un câmp magnetic la mare depărtare de el mult mai mic decât câmpul magnetic produs de un singur solenoid.

Problema TEAM 22 constă într-o configurație SMES care are doi solenoizi prin care trec curenți de sensuri opuse. În acest fel câmpul de dispersie în cazul folosirii a doi solenoizi este mai mic decât câmpul de dispersie al unui singur solenoid. Această construcție simulează câmpul magnetic al unui quadripol care scade (la depărtare) cu puterea a 5-a

¹Superconducting Magnetic Energy Storage

a razei, spre deosebire de câmpul magnetic al unui solenoid (un dipol magnetic) care scade la depărtare cu puterea a 3-a a razei. Desigur, această construcție consumă mai mult material decât un singur solenoid, avantajul economiei de material (fața de cazul toroidului) nemaifiind semnificativ. Totuși, construcția cu solenoizi este mult mai simplă din punct de vedere tehnologic[1].

6.1.1 Prezentarea problemei

Un dispozitiv SMES (figura 6.1) trebuie optimizat astfel încât să fie atinse următoarele obiective:

- Energia magnetică stocată în dispozitiv să fie 180 MJ;
- Câmpul magnetic trebuie să satisfacă condiția fizică ce garantează supraconductibilitatea;
- Câmpul de dispersie (măsurat la o distanță de 10 metri de dispozitiv) să fie cât mai mic posibil.

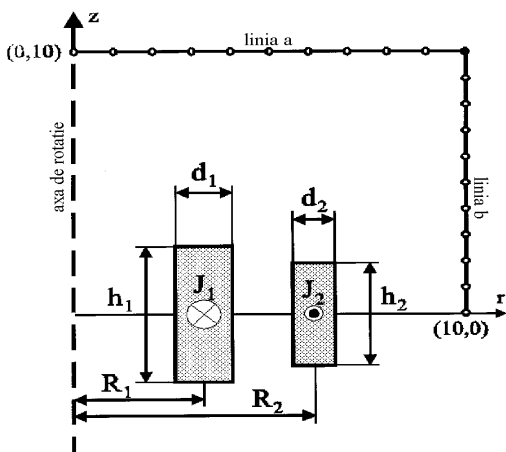


Figura 6.1: Dispozitiv SMES cu doi solenoizi

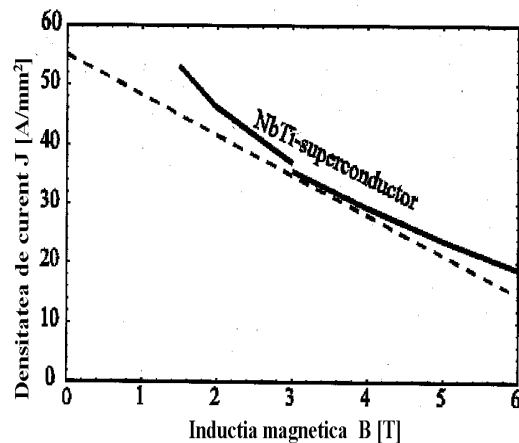


Figura 6.2: Restricția impusă pentru supraconductor

Problema are 8 parametri ($R_1, R_2, h_1/2, h_2/2, d_1, d_2, J_1, J_2$) ale căror restricții sunt prezentate în tabelul 6.1.

Condiția care asigură faptul că bobinele nu își pierd starea supraconductoare (“quench condition”) constă într-o relație între modulul densității de curent și valoarea maximă a

	R_1	R_2	$h_1/2$	$h_2/2$	d_1	d_2	J_1	J_2
	[m]	[m]	[m]	[m]	[m]	[m]	[MA/m ²]	[MA/m ²]
min	1.0	1.8	0.1	0.1	0.1	0.1	10.0	-30.0
max	4.0	5.0	1.8	1.8	0.8	0.8	30.0	-10.0

Tabelul 6.1: Restricții de domeniu ale variabilelor de optimizare

modulului inducției magnetice $|B|$ în bobine, așa cum arată figura 6.2. Ecuația (6.1) este o aproximare a curbei din figura 6.2,

$$|\mathbf{J}| = (-6.4|\mathbf{B}| + 54.0) \text{ A/mm}^2. \quad (6.1)$$

Funcția obiectiv propusă este

$$F = \frac{B_{\text{stray}}^2}{B_{\text{norm}}^2} + \frac{|E - E_{\text{ref}}|}{E_{\text{ref}}}, \quad (6.2)$$

unde $E_{\text{ref}} = 180 \text{ MJ}$, $B_{\text{norm}} = 2.0 \cdot 10^{-4} \text{ T}$ și

$$B_{\text{stray}}^2 = \frac{\sum_{i=1}^{22} |B_{\text{stray}_i}|^2}{22}. \quad (6.3)$$

Valoarea B_{stray}^2 este obținută după evaluarea câmpului în 22 de puncte echidistante de pe liniile a și b (figura 6.1).

6.1.2 Evaluarea funcției de cost

Evaluarea funcției de cost a fost făcută cu metoda elementelor finite.

• Rețeaua de discretizare

Având în vedere strategia de optimizare propusă la capitolul 5, în care precizia evaluării funcției de cost era crescută pe parcursul algoritmului de optimizare, finețea rețelei de discretizare (caracterizată prin numărul de noduri N) este corelată cu parametrul p asociat preciziei de calcul ($p = 0$ corespunde unei precizii proaste, deci unei rețele rare de discretizare, cu aproximativ 800 noduri, iar $p = 1$ corespunde unei precizii bune, deci unei rețele fine de discretizare, cu 10000 noduri).

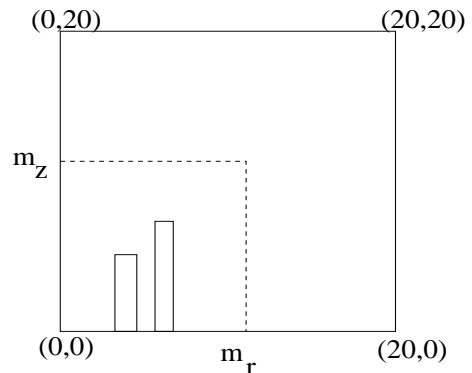


Figura 6.3: Distribuția nodurilor: în zona $[0, m_r] \times [0, m_z]$ sunt n^2 noduri.

Rețeaua de discretizare este alcătuită din triunghiuri dreptunghice, obținute din înjumătățirea unor dreptunghiuri ce discretizează un domeniu ce se întinde atât pe axa r cât și pe axa z între 0 și 20 metri. Numărul de noduri pe axa r este egal cu numărul de noduri pe axa z și este $2n$, astfel încât numărul total de noduri este $N = (2n)^2$. Împărțirea se face astfel: zona cuprinsă între 0 și m_r pe axa r și 0 și m_z pe axa z (zonă ce cuprinde bobinele) cuprinde n^2 noduri, restul de noduri rămase fiind împrăștiate în restul domeniului, astfel încât nodurile sunt dispuse mai rar spre frontiera de la 20 metri (figura 6.3).

• Rezolvarea și calculul mărimilor de interes

Rezolvarea (asamblarea ecuațiilor metodei elementelor finite) a fost făcută în două moduri. Primul din ele utilizează schimbarea de variabilă $A^* = Ar$ iar al doilea utilizează $A^* = A/\sqrt{r}$. Detalii despre rezolvarea și postprocesarea problemelor axisimetrice se pot găsi în [76].

• Teste numerice

Solver FAP	p	N	m_r [m]	m_z [m]	Nr. iter. SOR	Energie [MJ]	B_{stray}^2 [10^{-10} T^2]	F
$A^* = Ar$	0	784	3	3	383	171.22	4.43	$5.9 \cdot 10^{-2}$
			4.5	4.5	262	161.13	7.93	$1.2 \cdot 10^{-1}$
			6.5	6.5	340	157.76	59.66	$2.7 \cdot 10^{-1}$
			8.5	8.5	184	137.18	248.74	$8.6 \cdot 10^{-1}$
	0.5	2116	3	3	834	175.78	1.97	$2.83 \cdot 10^{-2}$
			4.5	4.5	620	174.11	2.38	$3.8 \cdot 10^{-2}$
			6.5	6.5	444	167.13	18.58	$1.18 \cdot 10^{-1}$
			8.5	8.5	397	161.08	8.16	$1.25 \cdot 10^{-1}$
	1	10000	3	3	822	178.67	3.11	$1.52 \cdot 10^{-2}$
			4.5	4.5	1712	177.72	2.68	$1.9 \cdot 10^{-2}$
			6.5	6.5	1393	175.83	1.70	$2.7 \cdot 10^{-2}$
			8.5	8.5	1189	174.43	1.79	$3.5 \cdot 10^{-2}$

Tabelul 6.2: Rezultate numerice pentru evaluarea funcției de cost a problemei TEAM 22 - metoda $A^* = Ar$

Tabelele 6.2 și 6.3 cuprind rezultatele diferitelor teste numerice efectuate pentru cele două tipuri de abordări, pentru diferite valori pentru m_r și m_z și pentru diferite precizii p (deci diferite grade de finețe a rețelei de discretizare). Configurația testată este cea raportată ca optimă de echipa de la Graz [1] și anume: $R_1 = 1.5703$ m, $R_2 = 2.0999$ m, $h_1/2 = 0.7846$ m, $h_2/2 = 1.4184$ m, $d_1 = 0.5943$ m, $d_2 = 0.2562$ m, $J_1 = 17.3367$ MA/m², $J_2 = -12.5738$ MA/m² pentru care autorii obțin o energie de 179.9924 MJ, valoarea $B_{\text{stray}}^2 = 2.1913 \cdot 10^{-10} \text{ T}^2$ și funcția obiectiv $F = 5.5203 \cdot 10^{-3}$. Pentru rezolvarea

sistemului linear a fost folosită metoda suprarelaxării succesive (SOR). Eroarea relativă la care se oprește algoritmul SOR a fost aleasă $2 \cdot 10^{-7}$.

Solver FAP	p	N	m_r [m]	m_z [m]	Nr. iter. SOR	Energie [MJ]	B_{stray}^2 [10^{-10} T^2]	F
$A^* = A/\sqrt{r}$	0	784	3	3	403	172.71	7.54	$5.9 \cdot 10^{-2}$
			4.5	4.5	444	164.66	5.77	$9.9 \cdot 10^{-2}$
			6.5	6.5	341	162.85	2.96	$1.02 \cdot 10^{-1}$
			8.5	8.5	280	147.22	27.23	$2.5 \cdot 10^{-1}$
	0.5	2116	3	3	492	176.29	9.33	$4.39 \cdot 10^{-2}$
			4.5	4.5	391	175.21	6.29	$4.23 \cdot 10^{-2}$
			6.5	6.5	509	168.78	9.87	$8.7 \cdot 10^{-2}$
			8.5	8.5	751	164.63	24.75	$1.5 \cdot 10^{-1}$
	1	10000	3	3	560	178.81	3.20	$1.46 \cdot 10^{-2}$
			4.5	4.5	784	177.99	31.46	$8.98 \cdot 10^{-2}$
			6.5	6.5	711	176.35	49.36	$1.43 \cdot 10^{-1}$
			8.5	8.5	563	175.41	53.32	$1.58 \cdot 10^{-1}$

Tabelul 6.3: Rezultate numerice pentru evaluarea funcției de cost a problemei TEAM 22 - metoda $A^* = A/r^{0.5}$

Sursa	Energie [MJ]	B_{stray}^2 [10^{-10} T^2]	F	Timp [s]	Metoda rezolvare	Nr. iter
Graz	179.99	2.1913	$5.52 \cdot 10^{-3}$	necunoscut		
MEGA (N = 2461)	177.0	3.068	$3.2 \cdot 10^{-2}$	3.4	ICCG	14
MEGA (N = 9671)	178.8	2.515	$1.3 \cdot 10^{-2}$	24.2	ICCG	29
MEGA (N = 34746)	179.3	2.295	$9.6 \cdot 10^{-3}$	156.4	ICCG	59

Tabelul 6.4: Rezultate de referință folosite în evaluarea algoritmului de calcul al funcției de cost pentru problema TEAM 22.

Acuratețea evaluării funcției de cost²

Pentru aceeași precizie (același număr de noduri) și metodă rețeaua de discretizare influențează foarte mult valoarea funcției de cost (până la un ordin de mărime). Comparând rezultatele numerice obținute cu cele de referință (tabelul 6.4³), se poate observa

²Este esențial ca pentru N foarte mare toate numerele reale să fie declarate de tip “double”. Aceasta îmbunătățește considerabil acuratețea mărimii B_{stray}^2 . Toate rezultatele prezentate în tabelele 6.2, 6.4 și 6.5 corespund cazului în care toate variabilele reale din program au fost declarate “double”.

³Programul MEGA a fost rulat pe o stație grafică HP/720.

că o rețea în care densitatea de noduri este mai mare în zona bobinelor conduce la rezultate mai bune. Cu o singură excepție (cazul $p = 0.5$ pentru solverul $A^* = A/\sqrt{r}$), cele mai bune rezultate se obțin pentru $m_r = m_z = 3$.

Distribuirea nodurilor rețelei de discretizare influențează foarte mult mărimea B_{stray}^2 mai ales în cazul $A^* = A/\sqrt{r}$. În cazul cu 10000 noduri ($p = 1$) rezultatele obținute pentru B_{stray}^2 în metoda $A^* = A/\sqrt{r}$ sunt mai sensibile la rețeaua de discretizare decât valorile obținute cu metoda $A^* = Ar$.

În general, metoda $A^* = A/\sqrt{r}$ conduce la valori mai bune pentru energie și mai proaste pentru B_{stray}^2 decât metoda $A^* = Ar$. La precizii mari (număr de noduri mai mare de 40000) metoda $A^* = Ar$ furnizează rezultate mai bune pentru funcția de cost decât metoda $A^* = A/\sqrt{r}$. În oricare din metode, se observă că, cu creșterea numărului de noduri, energia converge către energia de referință, lucru care nu se întâmplă cu valoarea lui B_{stray}^2 . Acesta scade către valoarea de referință și apoi crește. Lucrul acesta se întâmplă pentru N cuprins între 40000 și 102400 pentru metoda $A^* = Ar$ și între 10000 și 40000 pentru metoda $A^* = A/\sqrt{r}$. Acest fenomen se întâmplă și în cazul în care toate variabilele din program erau declarate “float” și numai rezolvarea cu SOR se făcea în dublă precizie, în acel caz acest fenomen apărând la un număr mai mic de noduri. S-ar putea ca pentru $N > 40000$ în cazul metodei $A^* = Ar$ și pentru $N > 10000$ în cazul metodei $A^* = A/\sqrt{r}$ nici reprezentarea ca “double” pentru numerele reale să nu fie de ajuns. Cea mai bună valoare este obținută pentru cazul $A^* = Ar$, $N = 40000$.

Figurile 6.4 și 6.5 prezintă spectrul câmpului magnetic în cele două cazuri extreme ($p = 0$ și $p = 1$) pentru situația $m_r = m_z = R_2 + 1$ și metoda $A^* = Ar$. Domeniul din aceste figuri este un “zoom” pe zona $(0,0,4,4)$.

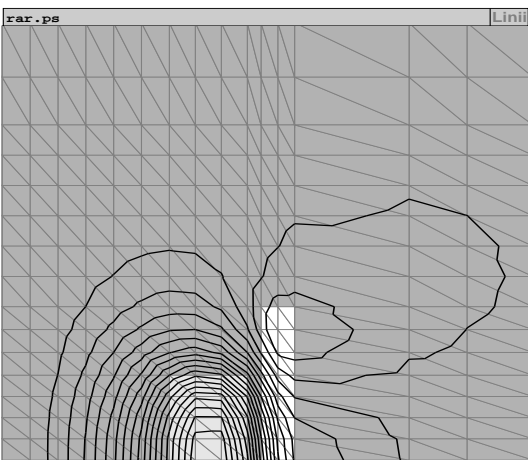


Figura 6.4: Linii de câmp în cazul $p = 0$

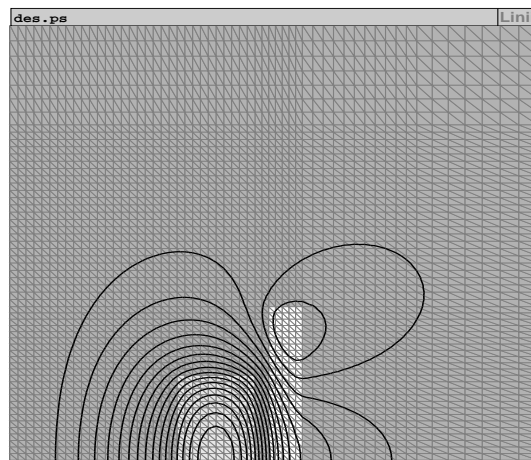


Figura 6.5: Linii de câmp în cazul $p = 1$

Timp de calcul

Efortul de calcul este contorizat pentru cazul $m_r = m_z = R_2 + 1$. Rezultatele sunt prezentate în tabelul 6.5 pentru metoda $A^* = Ar$ și în tabelul 6.6 pentru metoda $A^* = A/\sqrt{r}$. Programul a fost scris în limbajul de programare C și sistemul de operare LINUX, fiind rulat pe un PC Pentium (75 MHz).

p	N	Nr. iter. SOR	Energie [MJ]	B_{stray}^2 [10^{-10} T^2]	F	Timp de calcul [s]			
						prep.	solver	postp.	total
0	784	376	171.11	4.09	$5.9 \cdot 10^{-2}$	0	2	0	2
0.5	2116	799	175.68	1.82	$2.8 \cdot 10^{-2}$	0	18	1	19
1	10000	1263	178.65	3.41	$1.6 \cdot 10^{-2}$	1	144	0	145
-	40000	2442	179.24	2.10	$9.5 \cdot 10^{-3}$	1	1137	0	1138
-	102400	5065	179.36	5.9	$1.83 \cdot 10^{-2}$	3	6041	0	6044
-	160000	6910	179.39	13.84	$3.79 \cdot 10^{-2}$	3	13001	9	13013

Tabelul 6.5: Efortul de calcul pentru evaluarea funcției de cost a problemei TEAM 22 - metoda $A^* = Ar$

p	N	Nr. iter. SOR	Energie [MJ]	B_{stray}^2 [10^{-10} T^2]	F	Timp de calcul [s]			
						prep.	solver	postp.	total
0	784	397	172.52	8.88	$6.32 \cdot 10^{-2}$	0	4	0	4
0.5	2116	476	176.21	9.93	$4.59 \cdot 10^{-2}$	0	11	0	11
1	10000	696	178.79	7.89	$2.64 \cdot 10^{-2}$	0	82	3	85
-	40000	1243	179.27	22.05	$5.9 \cdot 10^{-2}$	1	584	8	593
-	102400	2692	179.37	82.99	$2.1 \cdot 10^{-1}$	3	3225	20	3248
-	160000	3805	179.39	151.07	$3.8 \cdot 10^{-1}$	4	7217	57	7278

Tabelul 6.6: Efortul de calcul pentru evaluarea funcției de cost a problemei TEAM 22 - metoda $A^* = A/r^{0.5}$

Se observă că numărul de iterații SOR este relativ mare și aceasta face ca timpul de calcul să fie mult mai mare decât cel necesar programului MEGA. Acesta din urmă folosește pentru rezolvarea sistemului liniar metoda gradientilor conjugați cu preconditionare incompletă Cholesky (ICCG⁴) care se dovedește a fi foarte eficientă (59 iterații în cazul cu 34746 noduri pentru a atinge o eroare de 10^{-10} față de 2442 iterații SOR ale programului FAP în cazul cu 40000 noduri, pentru o eroare de $2 \cdot 10^{-7}$).

⁴Incomplete Cholesky Conjugate Gradient

În concluzie, rezolvarea acestei probleme prin metoda elementelor finite este nesatisfăcătoare pentru un algoritm de optimizare în care este inclusă, deoarece timpul de calcul necesar unei evaluări este foarte mare (15 minute). Mai mult, ea nu este potrivită din punct de vedere al evaluării inducției magnetice în puncte din spațiu în care câmpul magnetic are valori foarte mici. În consecință ea nu a fost folosită în optimizarea propriu-zisă a problemei.

Metoda de rezolvare adoptată se bazează pe utilizarea formulelor Biot-Savart-Laplace. Metoda este prezentată în detaliu în [157].

6.1.3 Utilizarea strategiei evoluționiste pentru optimizarea propriu-zisă

Programul evoluționist descris în capitolul 5 a fost folosit pentru optimizarea propriu-zisă a acestei probleme.

Drept primă referință vom considera soluția obținută de echipa din Graz și anume:

$$\begin{aligned} R_1 &= 1.5703 \text{ m} & R_2 &= 2.0999 \text{ m} \\ h_1/2 &= 0.7846 \text{ m} & h_2/2 &= 1.4184 \text{ m} \\ d_1 &= 0.5942 \text{ m} & d_2 &= 0.2562 \text{ m} \\ J_1 &= 17.3367 \text{ MA/m}^2 & J_2 &= -12.5738 \text{ MA/m}^2 \end{aligned}$$

Autorii raportează pentru această configurație valorile:

$$E = 179.9924 \text{ MJ} \quad B_{\text{stray}}^2 = 2.1913 \cdot 10^{-10} \text{ T}^2 \quad F = 5.5203 \cdot 10^{-3}$$

Pentru o anumită configurație geometrică, se pot determina prin minimizare pătratică curenții pentru care energia are valoarea de 180 MJ [71]. Cu această abordare, pentru configurația geometrică Graz se pot găsi curenți care să corespundă unei valori mai bune a funcției obiectiv:

$$\begin{aligned} J_1 &= 17.3867 \text{ MA/m}^2 & J_2 &= -12.6166 \text{ MA/m}^2 \\ B_{\text{stray}}^2 &= 1.6239 \cdot 10^{-10} \text{ T}^2 & F &= 4.0599 \cdot 10^{-3} \end{aligned}$$

Problema de optimizat va fi tratată în consecință ca o problemă cu 6 parametri, și anume doar parametrii geometrici.

Cea mai bună configurație pe care am găsit-o folosind algoritmul evoluționist descris în capitolul anterior⁵este:

⁵Această configurație pare mai bună decât configurația Graz. Pentru siguranță ar trebui ca echipa de la Graz să valideze acest lucru.

$R_1 = 1.382670 \text{ m}$	$R_2 = 1.900561 \text{ m}$
$h_1/2 = 1.0894420 \text{ m}$	$h_2/2 = 1.561662 \text{ m}$
$d_1 = 0.451645 \text{ m}$	$d_2 = 0.172104 \text{ m}$
$J_1 = 19.3324 \text{ MA/m}^2$	$J_2 = -18.8948 \text{ MA/m}^2$
$E = 180 \text{ MJ}$	$B_{\text{stray}}^2 = 1.0828 \cdot 10^{-10} \text{ T}^2$
$F = 2.707 \cdot 10^{-3}$	

Pentru a putea aprecia comportarea algoritmului genetic, această configurație o vom considera de referință.

Tratarea condiției de quench se face foarte dur: crearea populației inițiale se face astfel încât toți indivizii să satisfacă această condiție, iar copiii care nu satisfac această condiție nu sunt luați în considerare⁶.

Pentru alegerea parametrilor s-au efectuat teste similare celor prezentate în capitolul 5. Am constatat că, dacă algoritmul nu durează suficient de mult, valoarea finală găsită s-ar putea să nu aibă curenții în domeniul impus. De aceea, o altă variantă a algoritmului penalizează și indivizii pentru care nu rezultă curenți în domeniul impus de valori, penalizarea făcându-se similar ca la condiția de quench.

În cele ce urmează vom prezenta rezultatele obținute pentru un algoritm genetic distribuit cu 5 procese (pe 5 procesoare diferite), intervalul migrației este 6, fiecare proces are 16 indivizi într-o populație, parametrii P_i sunt 1 1 1 4 4 1, q ales așa cum s-a specificat în capitolul 5, $a = 0.25$, $b = 2$. Algoritmul se oprește după 500 generații. Pentru fiecare caz s-au făcut câte 10 rulări independente. Informațiile despre cele 10 rulări se găsesc în tabelele 6.7 și 6.8 pentru situația în care funcția obiectiv este penalizată dacă curenții ies din domeniul impus și în tabelele 6.9 și 6.10 pentru cazul în care nu se aplică nici o penalizare în aceeași situație. Configurația de referință este configurația B.9. Tabelele 6.7 și 6.9 conțin configurațiile obținute. Numerele din paranteză reprezintă diferențele relative (în procente) față de configurația de referință. Sunt calculate de asemenea mediile și dispersiile celor zece rulări.

Trebuie specificat faptul că aceste rulări au fost făcute pentru $p = 0$, adică cu o precizie mai mică a evaluării funcției obiectiv. În tabelele 6.7 și 6.9 valoarea funcției obiectiv este calculată pentru $p = 1$. În tabelele 6.8 și 6.10 este trecută valoarea calculată cu $p = 0$ (valoarea obținută efectiv prin procesul de optimizare). Se observă că diferența dintre aceste valori nu este foarte mare, dar timpul necesar evaluării cu $p = 1$ crește foarte mult (de zeci de ori).

Este interesant de subliniat următorul aspect. În cazul în care funcția obiectiv este

⁶Acest stil de tratare a unei restricții se întâlnește sub numele de “death penalty”.

penalizată dacă curenții optimi ies din domeniul impus ne-am fi așteptat ca timpul de optimizare să fie mai mare datorită efortului mai mare de construire a populației inițiale. În acest caz, pentru găsirea unei populații inițiale cu 16 indivizi au fost necesare în medie 600 evaluări de funcții, pe când în situația în care se penaliza doar condiția de quench au fost necesare în medie 70 de evaluări de funcții. Algoritmul face în medie 6627 evaluări de funcții (pe proces) în varianta cu penalizare suplimentară și în medie 6065 evaluări de funcții (pe proces) în varianta fără penalizare suplimentară. Algoritmul de optimizare a durat însă în medie 3 ore în varianta cu penalizare și 4 ore în varianta fără penalizare. În concluzie **timpul de optimizare nu este proporțional cu numărul de evaluări**. Acest lucru era de așteptat pentru o problemă la care evaluarea funcției obiectiv este un proces iterativ, la același criteriu de oprire timpul de calcul depinzând de punctul de calcul. De aceea, graficele prezentate în figurile următoare au în axa absciselor numărul generației și nu numărul de evaluări.

Din punct de vedere al calității soluției găsite, funcția obiectiv fără penalizare suplimentară generează soluții mai bune (în medie funcția obiectiv găsită este $3.52 \cdot 10^{-3}$ față de $6.49 \cdot 10^{-3}$ cât este în cazul cu penalizare suplimentară).

Calculul diferențelor relative față de situația de referință scoate în evidență alte aspecte. **Problema este prost condiționată.** Perturbații mici în date duc la perturbații mari ale funcției obiectiv. De exemplu diferențele relative ale parametrilor testului A.6 sunt sub 15% față de testul de referință B.9 iar funcția obiectiv are o diferență relativă de 80 %. În testul B.10 parametrii sunt la o diferență relativă de maxim 21 % iar funcția obiectiv este la o diferență relativă de 103 %. De asemenea valori bune ale funcției obiectiv (testul B.8) pot fi relativ departe (cel puțin pentru anumiți parametri) de punctul de referință. Această proastă condiționare se reflectă și în dispersiile rezultatelor celor 10 rulări, dispersiile parametrilor fiind întotdeauna mai mici decât dispersia funcției obiectiv.

Vom încerca să apreciem succesul sau eșecul unei rulări în funcție de configurația găsită și nu de valoarea funcției obiectiv. Dacă considerăm de exemplu că zona unui anumit parametru a fost găsită dacă diferența relativă dintre parametru și valoarea lui de referință este sub 10 %, atunci rezultă că:

- zona lui R_1 a fost găsită în 10/10 cazuri în situația fără penalizare (B) și în 4/10 cazuri în situația cu penalizare (A).
- zona lui R_2 a fost găsită în 9/10 cazuri în situația B și în 3/10 cazuri în A.
- zona lui $h_1/2$ a fost găsită în 6/10 cazuri în situația B și în 3/10 cazuri în A.
- zona lui $h_2/2$ a fost găsită în 10/10 cazuri în situația B și în 7/10 cazuri în A.
- zona lui d_1 a fost găsită în 3/10 cazuri în situația B și în 2/10 cazuri în A.
- zona lui d_2 a fost găsită în 1/10 cazuri în situația B și în 0/10 cazuri în A.
- zona lui J_1 a fost găsită în 6/10 cazuri în situația B și în 3/10 cazuri în A.
- zona lui J_2 a fost găsită în 3/10 cazuri în situația B și în 4/10 cazuri în A.

În concluzie, rezultatele în cazul fără penalizare suplimentară (B) sunt mult mai bune decât cele în care configurațiile cu valori nepermise ale curenților sunt rejectate. Este de altfel recunoscut că un algoritm genetic în care valoarea funcția obiectiv este penalizată dacă nu sunt îndeplinite anumite restricții are șanse să găsească un optim mai bun decât un algoritm care practică “pedeapsa cu moartea”. Indivizi mai slabi din punct de vedere genetic pot produce “supercopii” mai degrabă decât indivizi buni, dar nu excepționali.

ID	R_1 [m]	R_2 [m]	$h_1/2$ [m]	$h_2/2$ [m]	d_1 [m]	d_2 [m]	J_1 [MA/m ²]	J_2 [MA/m ²]	$F \cdot 10^3$
A.1	1.5190 (9.8)	2.0836 (9.6)	1.2128 (11.3)	1.7386 (11.3)	0.2766 (38.7)	0.1029 (40.2)	25.2657 (30.6)	-25.2599 (33.6)	6.854 (153.1)
A.2	1.6125 (16.6)	2.1585 (13.5)	1.0821 (0.6)	1.6521 (5.7)	0.2939 (34.9)	0.1201 (30.2)	24.8971 (28.7)	-22.3517 (18.2)	7.402 (173.4)
A.3	1.6524 (19.5)	2.1931 (15.3)	1.0902 (0.07)	1.6657 (6.6)	0.2918 (35.3)	0.1343 (21.9)	24.6501 (27.5)	-19.9509 (5.5)	8.625 (218.6)
A.4	1.7029 (23.1)	2.2648 (19.1)	0.9589 (11.9)	1.6150 (3.4)	0.3103 (31.2)	0.1254 (27.1)	24.4072 (26.2)	-20.3451 (7.6)	9.268 (242.3)
A.5	1.4787 (6.9)	2.0464 (7.6)	0.9455 (13.2)	1.5400 (1.3)	0.4615 (2.1)	0.1579 (8.2)	19.0869 (1.2)	-18.0317 (4.5)	3.960 (46.2)
A.6	1.4622 (5.7)	2.0958 (10.2)	0.9345 (14.2)	1.5936 (2.04)	0.4758 (5.3)	0.1458 (15.2)	18.2147 (5.78)	-17.1176 (9.4)	4.870 (79.9)
A.7	1.6322 (18.05)	2.1335 (12.2)	0.7847 (27.9)	1.4010 (10.2)	0.6008 (33.03)	0.2003 (16.3)	17.0719 (11.6)	-16.9736 (10.1)	4.563 (67.5)
A.8	1.2922 (6.53)	1.8385 (3.26)	1.2981 (19.1)	1.7230 (10.3)	0.4014 (11.1)	0.2000 (16.2)	19.8748 (2.8)	-14.9531 (20.8)	3.072 (13.4)
A.9	1.5651 (13.1)	2.0961 (10.2)	1.1474 (5.3)	1.6717 (7.05)	0.2950 (34.6)	0.1245 (27.6)	24.6651 (27.5)	-22.4271 (18.6)	6.421 (137.1)
A.10	1.7425 (26.03)	2.3056 (21.3)	0.9013 (17.2)	1.5925 (1.9)	0.3208 (28.9)	0.1040 (39.5)	24.2110 (25.2)	-24.2105 (28.1)	9.887 (265.23)
medii	1.56	2.12	1.03	1.62	0.37	0.14	22.23	-20.16	6.49
dispersii	8.06 %	5.76 %	14.42%	5.73%	27.61%	23.55%	13.85 %	15.95 %	34.16 %

Tabelul 6.7: Rezultate numerice ale optimizării problemei TEAM 22 în cazul în care funcția obiectiv este penalizată dacă curenții ies din domeniul impus.

ID	$B_{\text{stray}}^2 \cdot 10^{10}$ [T ²]	Timp optimizare	Număr evaluări	Timp evaluare (la p = 0) [s]	$F \cdot 10^3$ (la p = 0)	Timp evaluare (la p = 1) [s]
A.1	2.7417	2 h 39 min	6677	0.61	7.065	17.84
A.2	2.9609	3 h 11 min	6468	0.59	7.660	15.70
A.3	3.4501	2 h 41 min	6665	0.59	8.939	16.25
A.4	3.7075	2 h 25 min	6623	0.60	9.614	16.43
A.5	1.5842	2 h 59 min	6519	0.61	4.096	26.42
A.6	1.9483	2 h 50 min	6694	0.62	5.013	22.92
A.7	1.8254	3 h 59 min	6650	0.79	4.685	33.86
A.8	1.2291	4 h 25 min	6649	0.91	3.139	41.77
A.9	2.5684	2 h 37 min	6677	0.59	6.660	16.43
A.10	3.9549	2 h 28 min	6651	0.58	10.243	14.93

Tabelul 6.8: Alte informații despre testele cu penalizare suplimentară

ID	R_1 [m]	R_2 [m]	$h_1/2$ [m]	$h_2/2$ [m]	d_1 [m]	d_2 [m]	J_1 [MA/m ²]	J_2 [MA/m ²]	$F \cdot 10^3$
B.1	1.5426 (11.5)	2.0333 (6.9)	0.9270 (14.9)	1.4609 (6.4)	0.5169 (14.4)	0.2028 (17.83)	18.1327 (6.2)	-17.0276 (9.8)	3.682 (36.01)
B.2	1.2537 (9.3)	1.9673 (3.5)	1.0995 (0.9)	1.6862 (7.9)	0.6873 (52.1)	0.2002 (16.3)	13.1652 (31.9)	-12.2676 (35.07)	4.007 (48.02)
B.3	1.4240 (2.9)	1.9542 (2.8)	1.0002 (8.1)	1.5218 (2.5)	0.4803 (6.3)	0.2005 (16.5)	18.8297 (2.6)	-15.8866 (15.9)	2.965 (9.5)
B.4	1.4652 (5.9)	1.9813 (4.2)	1.2301 (12.9)	1.6924 (8.3)	0.3041 (32.6)	0.1285 (25.33)	24.3136 (25.7)	-22.9611 (21.5)	4.659 (72.1)
B.5	1.4167 (2.4)	1.9494 (2.5)	0.9840 (9.6)	1.5083 (3.4)	0.5177 (14.6)	0.2022 (17.4)	17.9215 (7.2)	-15.9813 (15.4)	2.868 (5.9)
B.6	1.3541 (2.06)	1.9550 (2.8)	0.9844 (9.6)	1.5449 (1.06)	0.6234 (38.04)	0.2020 (17.3)	15.2558 (21.08)	-14.6411 (22.5)	3.073 (13.5)
B.7	1.3214 (4.4)	1.8273 (3.8)	1.3179 (20.9)	1.7111 (9.5)	0.4058 (10.1)	0.2007 (16.6)	19.7671 (2.2)	-16.2153 (14.1)	3.046 (12.5)
B.8	1.3764 (0.4)	1.9245 (1.2)	1.0010 (8.1)	1.5133 (3.09)	0.6079 (34.6)	0.2117 (23.01)	15.7861 (18.3)	-15.5763 (17.5)	2.765 (2.1)
B.9	1.3826 (0)	1.9005 (0)	1.0894 (0)	1.5616 (0)	0.4516 (0)	0.1721 (0)	19.3324 (0)	-18.8948 (0)	2.707 (0)
B.10	1.5149 (9.5)	2.1534 (13.3)	0.8582 (21.2)	1.5707 (0.5)	0.4934 (9.2)	0.1438 (16.4)	18.1607 (6.06)	-17.0036 (10.01)	5.513 (103.6)
medii	1.40	1.96	1.05	1.57	0.51	0.18	18.06	-16.64	3.52
dispersii	5.90 %	4.12 %	12.56 %	5.29 %	20.77 %	14.55 %	15.77 %	15.97 %	25.22 %

Tabelul 6.9: Rezultate numerice ale optimizării problemei TEAM 22 în cazul în care funcția obiectiv nu este penalizată dacă curenții ies din domeniul impus.

ID	$B_{\text{stray}}^2 \cdot 10^{10}$ [T ²]	Tim optimizare	Număr evaluări	Tim evaluare (la p = 0) [s]	$F \cdot 10^3$ (p = 0)	Tim evaluare (la p = 1) [s]
B.1	1.4729	3 h 55 min	6064	0.85	3.797	34.51
B.2	1.6028	4 h 23 min	6067	1.00	4.083	57.28
B.3	1.1862	3 h 38 min	6060	1.06	3.041	34.67
B.4	1.8639	3 h 37 min	6068	0.69	4.844	21.06
B.5	1.1473	3 h 54 min	6062	0.79	2.954	36.50
B.6	1.2295	4 h 11 min	6062	0.85	3.142	44.27
B.7	1.2186	5 h 08 min	6062	0.91	3.118	43.66
B.8	1.1062	4 h 11 min	6070	0.92	2.826	46.03
B.9	1.0828	4 h 15 min	6070	0.65	2.812	31.83
B.10	2.2053	2 h 55 min	6074	0.61	5.672	21.16

Tabelul 6.10: Alte informații despre testele fără penalizare suplimentară

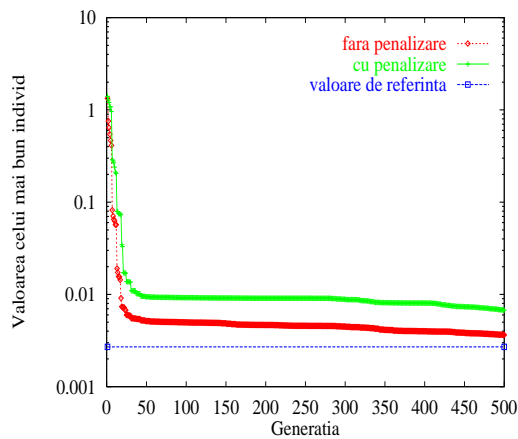


Figura 6.6: Evoluția celui mai bun individ în funcție de numărul generației

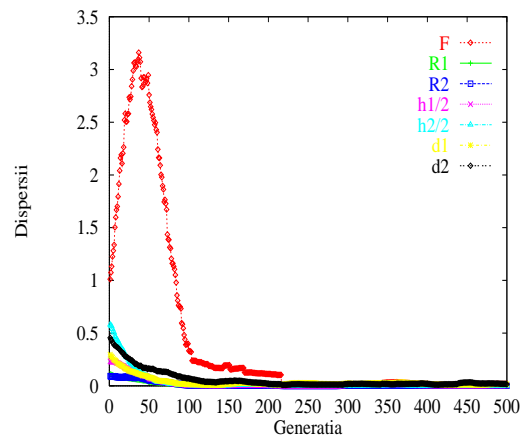


Figura 6.7: Evoluția dispersiilor (cazul funcției obiectiv nepenalizate)

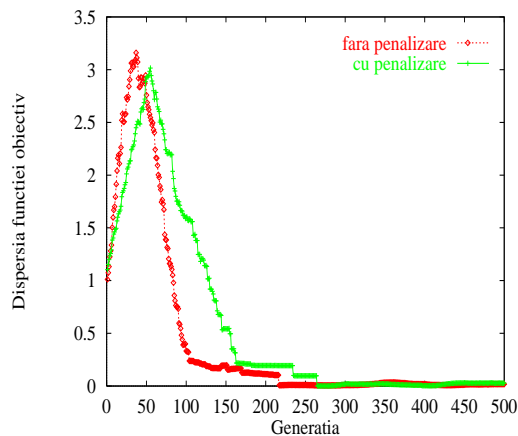


Figura 6.8: Evoluția dispersiei funcției obiectiv (scara liniară pe y)

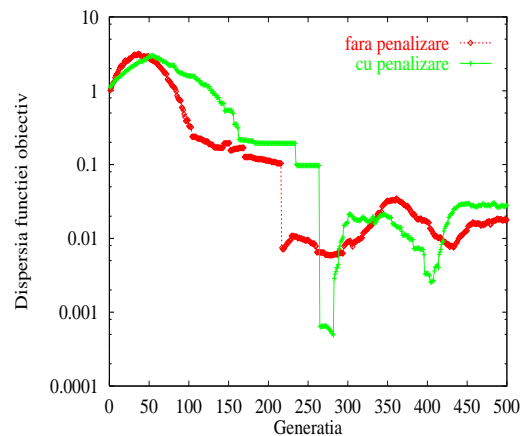
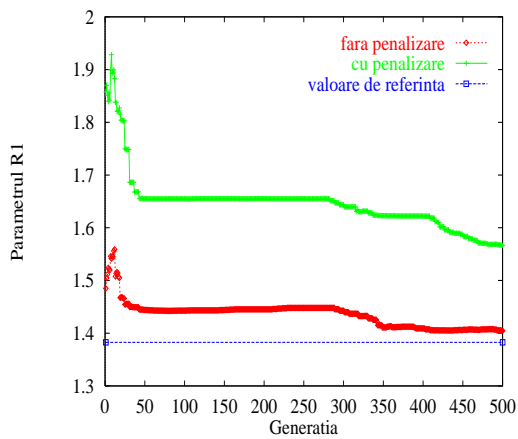
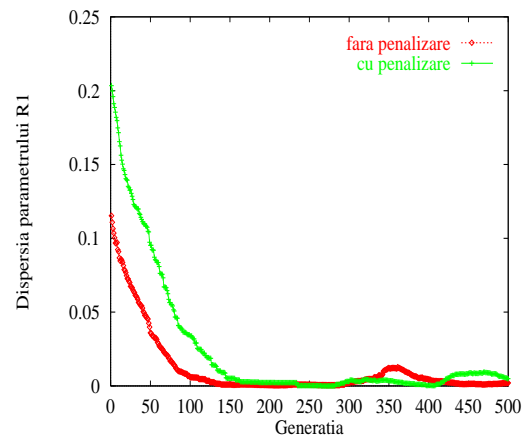
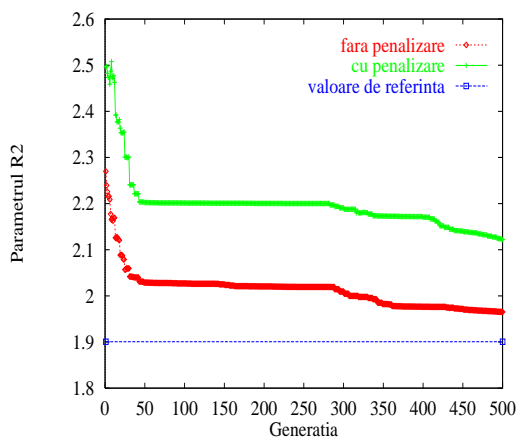
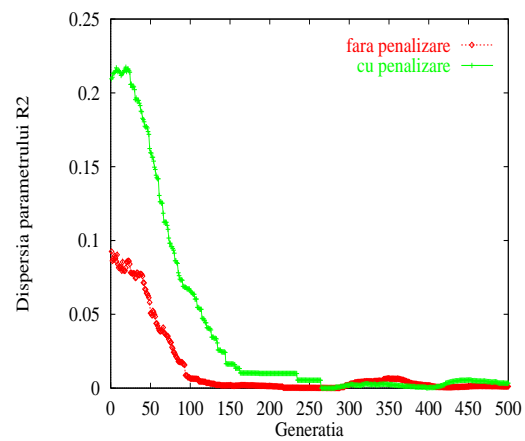
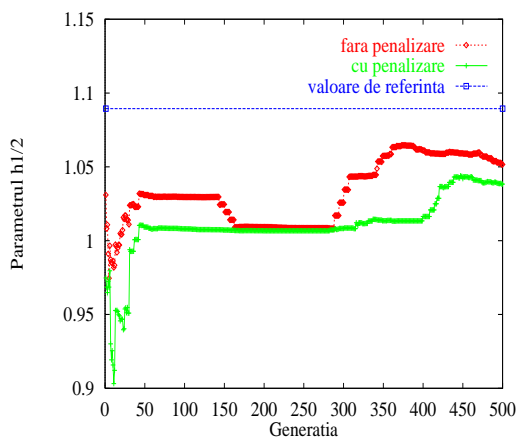
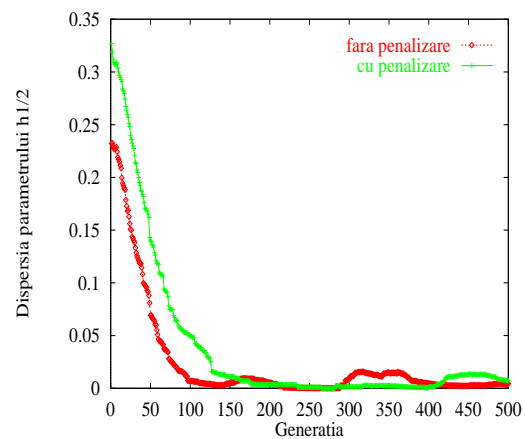


Figura 6.9: Evoluția dispersiei funcției obiectiv (scara logaritmică pe y)

Figura 6.10: Evoluția parametrului R_1 Figura 6.11: Evoluția dispersiei lui R_1 Figura 6.12: Evoluția parametrului R_2 Figura 6.13: Evoluția dispersiei lui R_2 Figura 6.14: Evoluția parametrului $h_1/2$ Figura 6.15: Evoluția dispersiei lui $h_1/2$

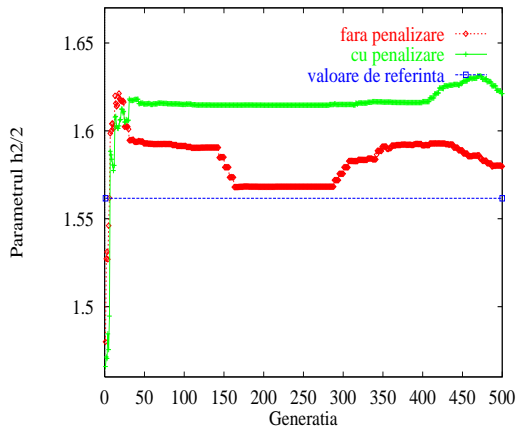


Figura 6.16: Evoluția parametrului $h_2/2$

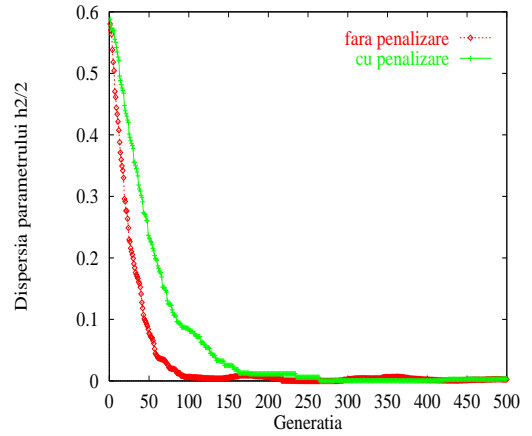


Figura 6.17: Evoluția dispersiei lui $h_2/2$

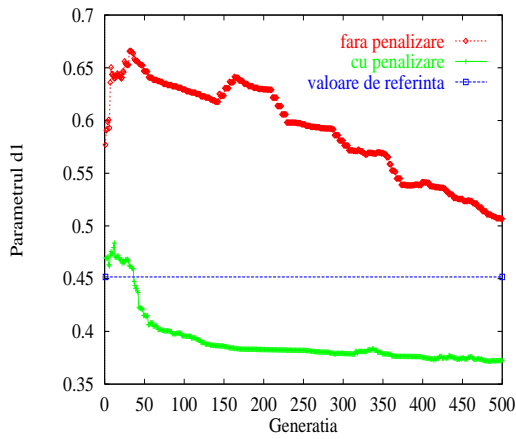


Figura 6.18: Evoluția parametrului d_1

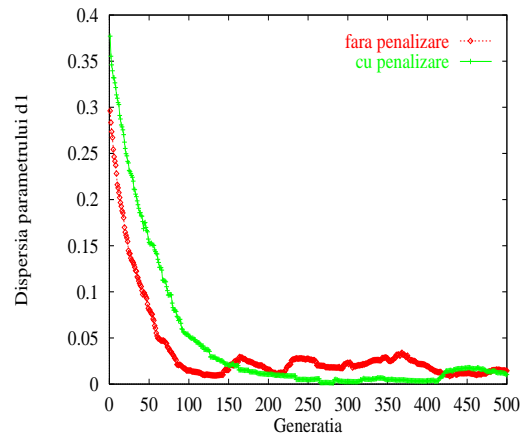


Figura 6.19: Evoluția dispersiei lui d_1

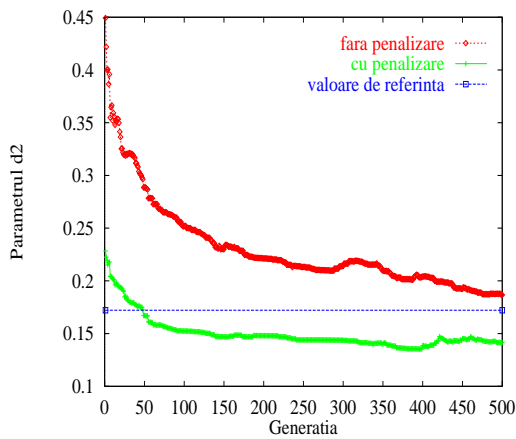


Figura 6.20: Evoluția parametrului d_2

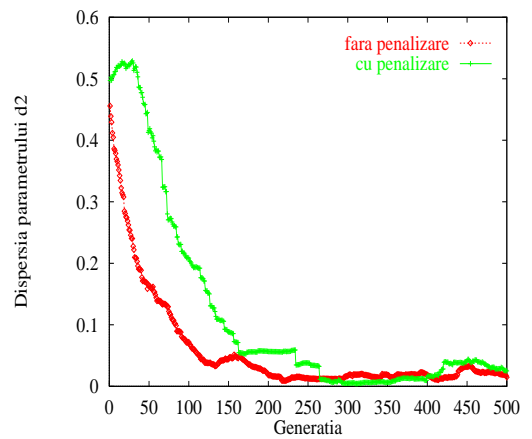


Figura 6.21: Evoluția dispersiei lui d_2

Evoluțiile dispersiilor din figurile de mai sus arată că algoritmul genetic converge către o populație de un anumit tip. Se observă că, de la un moment dat dispersiile oscilează sub o anumită valoare (aproximativ 5 %). Se știe că într-o astfel de situație o șansă mai mare de a obține o soluție mai bună este de a face o nouă rulare mai degrabă decât de a continua perpetuarea generației obținute.

Concluzia finală este aceea că pentru o astfel de problemă prost condiționată este necesar cel puțin un algoritm genetic adaptiv. Un algoritm genetic care să folosească tehnici de niching ar fi poate mult mai potrivit iar calculul sensibilităților devine obligatoriu.

6.1.4 Calculul sensibilităților

• Sensibilitățile funcției de cost

Funcția de cost se calculează cu relația

$$F = \frac{1}{22B_{\text{norm}}^2} \sum_{i=1}^{22} (B_{r_i}^2 + B_{z_i}^2) + \frac{1}{E_{\text{ref}}} |E - E_{\text{ref}}|. \quad (6.4)$$

Derivata funcției F în raport cu unul din parametrii de optimizat p va fi

$$\begin{aligned} \frac{\partial F}{\partial p} &= \frac{1}{22B_{\text{norm}}^2} \sum_{i=1}^{22} \left(2B_{r_i} \frac{\partial B_{r_i}}{\partial p} + 2B_{z_i} \frac{\partial B_{z_i}}{\partial p} \right) + \frac{1}{E_{\text{ref}}} \text{sgn}(E - E_{\text{ref}}) \frac{\partial E}{\partial p} = \\ &= \frac{1}{11B_{\text{norm}}^2} \sum_{i=1}^{22} \left(\mathbf{B}_i \cdot \frac{\partial \mathbf{B}_i}{\partial p} \right) + \frac{1}{E_{\text{ref}}} \text{sgn}(E - E_{\text{ref}}) \frac{\partial E}{\partial p}. \end{aligned} \quad (6.5)$$

Pentru calculul sensibilităților, termenii care trebuie evaluați sunt: $\frac{\partial \mathbf{B}_i}{\partial p}$, $\frac{\partial E}{\partial p}$. În cele ce urmează vom detalia formulele sensibilităților inducției magnetice și energiei la fiecare din cei 8 parametri de optimizare. Vom ține cont de faptul că evaluarea sensibilităților se face după evaluarea funcției de cost, și de aceea vom exprima, pe cât posibil, formulele sensibilităților în funcție de mărimi deja calculate.

• Utilizarea formulelor Biot-Savart-Laplace

Pentru această problemă, sensibilitățile se pot calcula mai simplu, fără a folosi metodele prezentate în capitolul 4.

Problema constă în distribuții de curenți aflate în aer și în consecință potențialul magnetic vector \mathbf{A} și inducția magnetică \mathbf{B} se pot exprima cu ajutorul formulelor Biot-Savart-Laplace (BSL):

$$\mathbf{A} = \frac{\mu_0}{4\pi} \int_{\mathcal{V}} \frac{\mathbf{J}}{R} dv, \quad (6.6)$$

$$\mathbf{B} = \frac{\mu_0}{4\pi} \int_{\mathcal{V}} \frac{\mathbf{J} \times \mathbf{R}}{R^3} dv. \quad (6.7)$$

În cazul axisimetric studiat $\mathbf{A} = \mathbf{u}_\theta A$ unde $A = A(r, z)$, $\mathbf{J} = \mathbf{u}_\theta J$ unde J este constant în fiecare din cei doi solenoizi, iar elementul de volum este $dv = r d\theta dr dz$. Formulele BSL se pot scrie mai detaliat astfel:

$$\mathbf{A} = \frac{\mu_0 J_1}{4\pi} \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \int_0^{2\pi} \frac{\mathbf{u}_\theta r}{R} d\theta dz dr + \frac{\mu_0 J_2}{4\pi} \int_{R_2 - \frac{d_2}{2}}^{R_2 + \frac{d_2}{2}} \int_{-\frac{h_2}{2}}^{\frac{h_2}{2}} \int_0^{2\pi} \frac{\mathbf{u}_\theta r}{R} d\theta dz dr, \quad (6.8)$$

$$\mathbf{B} = \frac{\mu_0 J_1}{4\pi} \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \int_0^{2\pi} \frac{\mathbf{u}_\theta \times \mathbf{R}}{R^3} r d\theta dz dr + \frac{\mu_0 J_2}{4\pi} \int_{R_2 - \frac{d_2}{2}}^{R_2 + \frac{d_2}{2}} \int_{-\frac{h_2}{2}}^{\frac{h_2}{2}} \int_0^{2\pi} \frac{\mathbf{u}_\theta \times \mathbf{R}}{R^3} r d\theta dz dr. \quad (6.9)$$

Energia din domeniu se poate calcula prin integrarea pe domeniile străbătute de curenți a mărimii $\mathbf{A} \cdot \mathbf{J}/2 = AJ/2$:

$$E = \pi J_1 \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} A r dr dz + \pi J_2 \int_{R_2 - \frac{d_2}{2}}^{R_2 + \frac{d_2}{2}} \int_{-\frac{h_2}{2}}^{\frac{h_2}{2}} A r dr dz. \quad (6.10)$$

Pentru simplificarea scrierii vom folosi notațiile din tabelul 6.11.

Notație	Formula
$\mathbf{I}_{A1}(R_1, d_1, h_1, J_1)$	$\frac{\mu_0 J_1}{4\pi} \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \int_0^{2\pi} \frac{\mathbf{u}_\theta r}{R} d\theta dz dr$
$\mathbf{I}_{A2}(R_2, d_2, h_2, J_2)$	$\frac{\mu_0 J_2}{4\pi} \int_{R_2 - \frac{d_2}{2}}^{R_2 + \frac{d_2}{2}} \int_{-\frac{h_2}{2}}^{\frac{h_2}{2}} \int_0^{2\pi} \frac{\mathbf{u}_\theta r}{R} d\theta dz dr$
$\mathbf{A}(R_1, d_1, h_1, J_1, R_2, h_2, d_2, J_2)$	$\mathbf{I}_{A1} + \mathbf{I}_{A2}$
$I_{A1}(R_1, d_1, h_1, J_1)$	$\mathbf{I}_{A1} \cdot \mathbf{u}_\theta$
$I_{A2}(R_2, d_2, h_2, J_2)$	$\mathbf{I}_{A2} \cdot \mathbf{u}_\theta$
$A(R_1, d_1, h_1, J_1, R_2, h_2, d_2, J_2)$	$I_{A1} + I_{A2}$
$\mathbf{I}_{B1}(R_1, d_1, h_1, J_1)$	$\frac{\mu_0 J_1}{4\pi} \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \int_0^{2\pi} \frac{\mathbf{u}_\theta \times \mathbf{R}}{R^3} r d\theta dz dr$
$\mathbf{I}_{B2}(R_2, d_2, h_2, J_2)$	$\frac{\mu_0 J_2}{4\pi} \int_{R_2 - \frac{d_2}{2}}^{R_2 + \frac{d_2}{2}} \int_{-\frac{h_2}{2}}^{\frac{h_2}{2}} \int_0^{2\pi} \frac{\mathbf{u}_\theta \times \mathbf{R}}{R^3} r d\theta dz dr$
$\mathbf{B}(R_1, d_1, h_1, J_1, R_2, h_2, d_2, J_2)$	$\mathbf{I}_{B1} + \mathbf{I}_{B2}$
$I_{E1}(R_1, d_1, h_1, J_1, R_2, h_2, d_2, J_2)$	$\pi J_1 \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} A r dr dz$
$I_{E2}(R_1, d_1, h_1, J_1, R_2, h_2, d_2, J_2)$	$\pi J_2 \int_{R_2 - \frac{d_2}{2}}^{R_2 + \frac{d_2}{2}} \int_{-\frac{h_2}{2}}^{\frac{h_2}{2}} A r dr dz$
$E(R_1, d_1, h_1, J_1, R_2, h_2, d_2, J_2)$	$I_{E1} + I_{E2}$

Tabelul 6.11: Notații folosite în formulele sensibilităților problemei TEAM 22

Mai sunt utile următoarele formule pentru calculul componentelor vectorului inducției

magnetice:

$$B_z(r, z) = \frac{1}{r} \frac{\partial}{\partial r} (rA), \tag{6.11}$$

$$B_r(r, z) = -\frac{\partial A}{\partial z}. \tag{6.12}$$

În calculul sensibilităților acestei probleme sunt utile formulele prezentate în tabelul 6.12 (a căror demonstrație poate fi găsită de exemplu în [39]).

$F(x)$	$F'(x)$
$F(x) = \int_{\Psi_1(x)}^{\Psi_2(x)} f(x, y) dy \Rightarrow$	$F'(x) = \int_{\Psi_1(x)}^{\Psi_2(x)} \frac{\partial f}{\partial x}(x, y) dy -$ $-\Psi_1'(x)f(x, \Psi_1(x)) + \Psi_2'(x)f(x, \Psi_2(x))$
$F(x) = \int_{\Psi_1(x)}^{\Psi_2(x)} f(y) dy \Rightarrow$	$F'(x) = -\Psi_1'(x)f(\Psi_1(x)) + \Psi_2'(x)f(\Psi_2(x))$

Tabelul 6.12: Formule utile pentru derivarea integralelor

• **Senzitivitățile față de curenți (J)**

Deducerea sensibilității inducției magnetice față de densitățile de curent este foarte simplă deoarece formula (6.9) indică liniaritatea inducției față de acești parametri. Se deduce imediat că:

$\frac{\partial \mathbf{B}}{\partial J_1} = \frac{\mathbf{I}_{B1}}{J_1}$	(6.13)
$\frac{\partial \mathbf{B}}{\partial J_2} = \frac{\mathbf{I}_{B2}}{J_2}$	(6.14)

Pentru energie, rezultatul nu mai este la fel de simplu pentru că în integrala (6.10) A depinde la rândul lui de curenți. În această situație cel mai simplu este să se utilizeze inductivitățile proprii și mutuale ale celor doi solenoizi. Astfel, dacă notăm $E(I_1, I_2)$ energia corespunzătoare curenților I_1, I_2 :

$$L_1 = \frac{2E(I_1, 0)}{I_1^2}, \tag{6.15}$$

$$L_2 = \frac{2E(0, I_2)}{I_2^2}, \tag{6.16}$$

$$M = \frac{2E(I_1, I_2) - L_1 I_1^2 - L_2 I_2^2}{I_1 I_2}. \tag{6.17}$$

Rezultă sensibilitățile:

$$\frac{\partial E}{\partial J_1} = (L_1 I_1 + M I_2) \frac{1}{d_1 h_1} = L_1 J_1 + M J_2 \frac{d_2 h_2}{d_1 h_1} \quad (6.18)$$

$$\frac{\partial E}{\partial J_2} = (L_2 I_2 + M I_1) \frac{1}{d_2 h_2} = L_2 J_2 + M J_1 \frac{d_1 h_1}{d_2 h_2} \quad (6.19)$$

• **Senzitivitățile față de parametrii geometrici (R, d, h)**

Fie p un astfel de parametru geometric. Atunci sensibilitatea componentelor inducției magnetice \mathbf{B} este:

$$\frac{\partial B_z}{\partial p} = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial A}{\partial p} \right) = \frac{1}{r} \frac{\partial A}{\partial p} + \frac{\partial}{\partial r} \left(\frac{\partial A}{\partial p} \right), \quad (6.20)$$

$$\frac{\partial B_r}{\partial p} = -\frac{\partial}{\partial z} \left(\frac{\partial A}{\partial p} \right). \quad (6.21)$$

Cu notațiile din tabelul 6.6 rezultă sensibilitatea potențialului vector față de parametrul p :

$$\frac{\partial A}{\partial R_1} = \frac{\partial I_{A1}}{\partial R_1}, \quad \frac{\partial A}{\partial d_1} = \frac{\partial I_{A1}}{\partial d_1}, \quad \frac{\partial A}{\partial h_1} = \frac{\partial I_{A1}}{\partial h_1} \quad (6.22)$$

$$\frac{\partial A}{\partial R_2} = \frac{\partial I_{A2}}{\partial R_2}, \quad \frac{\partial A}{\partial d_2} = \frac{\partial I_{A2}}{\partial d_2}, \quad \frac{\partial A}{\partial h_2} = \frac{\partial I_{A2}}{\partial h_2} \quad (6.23)$$

Trebuie astfel să detaliem doar calculul mărimilor: $\frac{\partial I_{A1}}{\partial R_1}$, $\frac{\partial I_{A1}}{\partial d_1}$ și $\frac{\partial I_{A1}}{\partial h_1}$.

Pentru calculul derivatelor integralei I_{A1} în raport cu R_1 și d_1 , aceasta se scrie

$$I_{A1} = \frac{\mu_0 J_1}{4\pi} \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} f(r) dr, \quad (6.24)$$

unde

$$f(r) = \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \int_0^{2\pi} \frac{\mathbf{u}_{\theta_0} \cdot \mathbf{u}_{\theta}}{R} r d\theta dz = \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} I(z, r, z_0, r_0) r dz. \quad (6.25)$$

Versorul \mathbf{u}_{θ_0} este versorul \mathbf{u}_{θ} în punctul de calcul de coordonate $r_0, z_0, \theta_0 = 0$. Deoarece $\mathbf{u}_{\theta_0} \cdot \mathbf{u}_{\theta} = \cos\theta$ și $R^2 = r_0^2 + r^2 + (z_0 - z)^2 - 2rr_0\cos\theta$, rezultă că

$$I(z, r, z_0, r_0) = \int_0^{2\pi} \frac{\cos\theta}{\sqrt{r_0^2 + r^2 + (z_0 - z)^2 - 2rr_0\cos\theta}} d\theta. \quad (6.26)$$

Această integrală poate fi adusă la forma

$$I(z, r, z_0, r_0) = \frac{2s}{rr_0} \left[F(m) \left(1 - \frac{m}{2} \right) - E(m) \right], \quad (6.27)$$

unde $s = \sqrt{(r_0 + r)^2 + (z_0 - z)^2}$, $m = \frac{4rr_0}{s^2}$ iar $F(m)$ și $E(m)$ sunt integralele eliptice complete Legendre definite prin:

$$E(m) = \int_0^{\frac{\pi}{2}} \sqrt{1 - m \sin^2 \phi} \, d\phi = \int_0^{\frac{\pi}{2}} \sqrt{1 - m \cos^2 \phi} \, d\phi, \quad (6.28)$$

$$F(m) = \int_0^{\frac{\pi}{2}} \frac{1}{\sqrt{1 - m \sin^2 \phi}} \, d\phi = \int_0^{\frac{\pi}{2}} \frac{1}{\sqrt{1 - m \cos^2 \phi}} \, d\phi. \quad (6.29)$$

Conform primei formule din tabelul 6.12 rezultă că:

$$\frac{\partial I_{A1}}{\partial R_1} = \frac{\mu_0 J_1}{4\pi} \left[-f\left(R_1 - \frac{d_1}{2}\right) + f\left(R_1 + \frac{d_1}{2}\right) \right] \quad (6.30)$$

$$\frac{\partial I_{A1}}{\partial d_1} = \frac{\mu_0 J_1}{4\pi} \left[\frac{1}{2} f\left(R_1 - \frac{d_1}{2}\right) + \frac{1}{2} f\left(R_1 + \frac{d_1}{2}\right) \right] \quad (6.31)$$

Pentru calculul derivatei integralei I_{A1} în raport cu h_1 , aceasta se scrie

$$I_{A1} = \frac{\mu_0 J_1}{4\pi} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} g(z) \, dz, \quad (6.32)$$

unde

$$g(z) = \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_0^{2\pi} \frac{\mathbf{u}_{\theta_0} \cdot \mathbf{u}_{\theta}}{R} r \, d\theta \, dz = \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} I(z, r, z_0, r_0) r \, dr. \quad (6.33)$$

Integrala I este dată de formula (6.27). Rezultă derivata integralei I_{A1} în raport cu h_1 :

$$\frac{\partial I_{A1}}{\partial h_1} = \frac{\mu_0 J_1}{4\pi} \left[\frac{1}{2} g\left(-\frac{h_1}{2}\right) + \frac{1}{2} g\left(\frac{h_1}{2}\right) \right] \quad (6.34)$$

Să calculăm acum derivatele inducției magnetice la parametrii geometrici.

Pentru a evita confuziile, atragem atenția că variabilele r și z au fost folosite ca variabile de integrare pentru calculul potențialului și derivatei lui la p . Rezultatul obținut este $A(r_0, z_0)$ și $\frac{\partial A}{\partial R_1}(r_0, z_0)$. În cele ce urmează r_0 și z_0 devin variabilele de derivare.

Observăm că (formulele (6.21) și (6.20)) mărimile care mai trebuie evaluate sunt derivatele în raport cu r_0 și cu z_0 ale sensibilității potențialului magnetic:

$$\frac{\partial B_z}{\partial R_1} = \frac{\partial I_{A1}}{\partial R_1} + r_0 \frac{\partial}{\partial r_0} \left(\frac{\partial I_{A1}}{\partial R_1} \right) \quad , \quad \frac{\partial B_r}{\partial R_1} = -\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A1}}{\partial R_1} \right) \quad (6.35)$$

$$\frac{\partial B_z}{\partial d_1} = \frac{\partial I_{A1}}{\partial d_1} + r_0 \frac{\partial}{\partial r_0} \left(\frac{\partial I_{A1}}{\partial d_1} \right) \quad , \quad \frac{\partial B_r}{\partial d_1} = -\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A1}}{\partial d_1} \right) \quad (6.36)$$

$$\frac{\partial B_z}{\partial h_1} = \frac{\partial I_{A1}}{\partial h_1} + r_0 \frac{\partial}{\partial r_0} \left(\frac{\partial I_{A1}}{\partial h_1} \right) \quad , \quad \frac{\partial B_r}{\partial h_1} = -\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A1}}{\partial h_1} \right) \quad (6.37)$$

$$\frac{\partial B_z}{\partial R_2} = \frac{\partial I_{A2}}{\partial R_2} + r_0 \frac{\partial}{\partial r_0} \left(\frac{\partial I_{A2}}{\partial R_2} \right) \quad , \quad \frac{\partial B_r}{\partial R_2} = -\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A2}}{\partial R_2} \right) \quad (6.38)$$

$$\frac{\partial B_z}{\partial d_2} = \frac{\partial I_{A2}}{\partial d_2} + r_0 \frac{\partial}{\partial r_0} \left(\frac{\partial I_{A2}}{\partial d_2} \right) \quad , \quad \frac{\partial B_r}{\partial d_2} = -\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A2}}{\partial d_2} \right) \quad (6.39)$$

$$\frac{\partial B_z}{\partial h_2} = \frac{\partial I_{A2}}{\partial h_2} + r_0 \frac{\partial}{\partial r_0} \left(\frac{\partial I_{A2}}{\partial h_2} \right) \quad , \quad \frac{\partial B_r}{\partial h_2} = -\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A2}}{\partial h_2} \right) \quad (6.40)$$

Relațiile (6.30) și (6.31) pot fi scrise astfel încât să punem în evidența dependența de r_0 și z_0 . Notând $R_{1i} = R_1 - \frac{d_1}{2}$ și $R_{1e} = R_1 + \frac{d_1}{2}$ rezultă că:

$$\frac{\partial}{\partial r_0} \left(\frac{\partial I_{A1}}{\partial R_1} \right) = \frac{\mu_0 J_1}{4\pi} \left[-R_{1i} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial}{\partial r_0} (I_{(z, R_{1i}, z_0, r_0)}) dz + R_{1e} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial}{\partial r_0} (I_{(z, R_{1e}, z_0, r_0)}) dz \right]$$

$$\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A1}}{\partial R_1} \right) = \frac{\mu_0 J_1}{4\pi} \left[-R_{1i} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial}{\partial z_0} (I_{(z, R_{1i}, z_0, r_0)}) dz + R_{1e} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial}{\partial z_0} (I_{(z, R_{1e}, z_0, r_0)}) dz \right]$$

$$\frac{\partial}{\partial r_0} \left(\frac{\partial I_{A1}}{\partial d_1} \right) = \frac{\mu_0 J_1}{8\pi} \left[R_{1i} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial}{\partial r_0} (I_{(z, R_{1i}, z_0, r_0)}) dz + R_{1e} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial}{\partial r_0} (I_{(z, R_{1e}, z_0, r_0)}) dz \right]$$

$$\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A1}}{\partial d_1} \right) = \frac{\mu_0 J_1}{8\pi} \left[R_{1i} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial}{\partial z_0} (I_{(z, R_{1i}, z_0, r_0)}) dz + R_{1e} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial}{\partial z_0} (I_{(z, R_{1e}, z_0, r_0)}) dz \right]$$

$$\frac{\partial}{\partial r_0} \left(\frac{\partial I_{A1}}{\partial h_1} \right) = \frac{\mu_0 J_1}{8\pi} \left[\int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \frac{\partial}{\partial r_0} (I_{(-\frac{h_1}{2}, r, z_0, r_0)} r) dr + \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \frac{\partial}{\partial r_0} (I_{(\frac{h_1}{2}, r, z_0, r_0)} r) dr \right]$$

$$\frac{\partial}{\partial z_0} \left(\frac{\partial I_{A1}}{\partial h_1} \right) = \frac{\mu_0 J_1}{8\pi} \left[\int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \frac{\partial}{\partial z_0} (I_{(-\frac{h_1}{2}, r, z_0, r_0)} r) dr + \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \frac{\partial}{\partial z_0} (I_{(\frac{h_1}{2}, r, z_0, r_0)} r) dr \right]$$

Mărimile cheie care trebuie calculate sunt derivatele în raport cu r_0 și z_0 ale integralei eliptice I (formula (6.27)). Rezultatele obținute sunt:

$$\frac{\partial I}{\partial r_0} = \frac{2}{rr_0} (\text{coef}_{F_r} F(m) + \text{coef}_{E_r} E(m)) \quad , \quad (6.41)$$

$$\frac{\partial I}{\partial z_0} = \frac{2}{rr_0} (\text{coef}_{F_z} F(m) + \text{coef}_{E_z} E(m)) \quad , \quad (6.42)$$

unde

$$\text{coef}_{Fr} = -\frac{r^2 + (z - z_0)^2}{sr_0}, \quad (6.43)$$

$$\text{coef}_{Er} = \frac{rr_0[(z_0 - z)^2 + r^2 - r_0^2] + [((z - z_0)^2 + r^2 + rr_0][(z - z_0)^2 + (r_0 - r)^2]}{(1 - m)r_0s^3}, \quad (6.44)$$

$$\text{coef}_{Fz} = \frac{z_0 - z}{s}, \quad (6.45)$$

$$\text{coef}_{Ez} = -\frac{(z_0 - z)[r_0^2 + r^2 + (z_0 - z)^2]}{(1 - m)s^3}. \quad (6.46)$$

Să calculăm acum derivatele energiei la parametrii geometrici:

$$\frac{\partial E}{\partial p} = \frac{\partial I_{E1}}{\partial p} + \frac{\partial I_{E2}}{\partial p} \quad (6.47)$$

Din nefericire, scrierea aditivă nu înseamnă în acest caz o separare a variabilelor ca în cazul inducției magnetice și de aceea trebuie să deducem formule de derivare a integralei I_{E1} la toți cei șase parametri geometrici. (Schimbând indicii 1 și 2 între ei rezultă formulele de derivare a integralei I_{E2} .)

Pentru calculul derivatelor integralei I_{E1} în raport cu R_1 și d_1 , aceasta se scrie

$$I_{E1} = \pi J_1 \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} h(R_1, r) dr, \quad (6.48)$$

unde

$$h(R_1, r) = \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} Ar dz. \quad (6.49)$$

Conform primei formule din tabelul 6.12 rezultă că:

$$\frac{\partial I_{E1}}{\partial R_1} = \pi J_1 \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial A}{\partial R_1} r dz dr - h_{(R_1, R_1 - \frac{d_1}{2})} + h_{(R_1, R_1 + \frac{d_1}{2})} \quad (6.50)$$

$$\frac{\partial I_{E1}}{\partial d_1} = \pi J_1 \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial A}{\partial d_1} r dz dr + \frac{1}{2} h_{(R_1, R_1 - \frac{d_1}{2})} + \frac{1}{2} h_{(R_1, R_1 + \frac{d_1}{2})} \quad (6.51)$$

Pentru calculul derivatei integralei I_{E1} în raport cu h_1 aceasta se scrie

$$I_{E1} = \pi J_1 \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} t(h_1, z) dz, \quad (6.52)$$

unde

$$t(h_1, z) = \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} Ar \, dr. \quad (6.53)$$

Conform primei formule din tabelul 6.12 rezultă că:

$$\frac{\partial I_{E1}}{\partial h_1} = \pi J_1 \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial A}{\partial h_1} r \, dz dr + \frac{1}{2} t_{(h_1, -\frac{h_1}{2})} + t_{(h_1, \frac{h_1}{2})} \quad (6.54)$$

Derivarea integralei I_{E1} la $p = R_2, d_2$ sau h_2 este mai simplă pentru că acești parametri nu apar în limitele de integrare:

$$\frac{\partial I_{E1}}{\partial p} = \pi J_1 \int_{R_1 - \frac{d_1}{2}}^{R_1 + \frac{d_1}{2}} \int_{-\frac{h_1}{2}}^{\frac{h_1}{2}} \frac{\partial A}{\partial p} r \, dz dr \quad (6.55)$$

Concluzii:

Calculul sensibilităților inducției magnetice față de parametrii geometrice presupune efectuarea de integrale unidimensionale din integranzi care presupun evaluări de funcții eliptice.

Calculul sensibilităților energiei față de parametrii geometrice presupune efectuarea de integrale tridimensionale din integranzi care presupun evaluări de funcții eliptice.

6.2 Problema TEAM 25

Problema TEAM 25 constă în optimizarea formei unei matrițe cu electromagnet, folosită pentru orientarea pulberilor magnetice în procesul de sinterizare a pieselor polare pentru micromașini. Matrița și electromagnetul sunt confecționate din oțel, forma matriței fiind astfel realizată încât să se genereze un câmp magnetic radial într-o cavitate ce va fi umplută cu pulberea magnetică [151].

6.2.1 Prezentarea problemei

Figura 6.23 reprezintă o secțiune transversală a matriței cu electromagnet, iar figura 6.24 prezintă un detaliu în zona de interes. Matrița trebuie optimizată pentru două valori ale solenațiilor bobinelor: 4253 amperi-spiră și respectiv 17500 amperi-spiră astfel încât câmpul magnetic în cavitate (de-a lungul curbei e-f din figura 6.24) să fie orientat radial și să aibă următoarele valori:

- în cazul solenației mici (4253 amperi-spiră)

$$B_x = 0.35 \cos \theta \text{ [T]} \quad B_y = 0.35 \sin \theta \text{ [T]} \quad (6.56)$$

- în cazul solenației mari (17500 amperi-spiră)

$$B_x = 1.5 \cos \theta \text{ [T]} \quad B_y = 1.5 \sin \theta \text{ [T]} \quad (6.57)$$

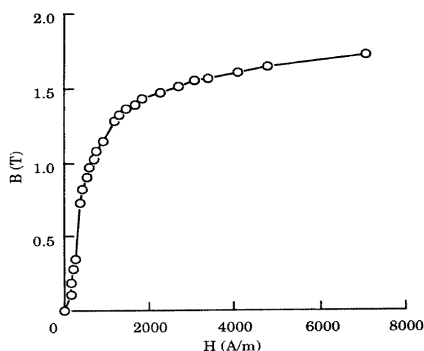


Figura 6.22: Curba de magnetizare a oțelului

Analizele preliminare făcute la curenți mici arată că distribuția de câmp cerută poate fi obținută (cu eroare acceptabilă) adoptând pentru matriță o formă obținută printr-o combinație de linie dreaptă, cerc și elipsă. Matrița este alcătuită din două părți, numite forme. Forma interioară a matriței este presupusă a fi un cerc de rază R_1 . Partea dinspre interior a formei exterioare este reprezentată de o elipsă de semiaxe L_2 și L_3 și o linie paralelă cu axa x, ca în figura 6.24.

Variabilele de proiectare sunt: R_1 - raza formei interioare, L_2 - axa lungă a elipsei, L_3 - axa scurtă a elipsei, L_4 - lungimea pintenului formei exterioare. Proiectatul are

libertatea să aleagă și alte parametrizări ale curbei g-h a formei interioare și ale curbei i-j-k-m corespunzătoare formei exterioare.

Se constată că la curenți mari, dacă forma interioară și cea exterioară sunt reprezentate prin cerc și elipsă rezultatele nu sunt satisfăcătoare.

Matrița și electromagnetul sunt din oțel, având curba de magnetizare prezentată în figura 6.22. Punctele acestei curbe sunt cele din tabelul 6.13.

B [T]	0.0	0.11	0.18	0.28	0.35	0.74	0.82	0.91
H [A/m]	0.0	140	178	215	253	391	452	529
B [T]	0.98	1.02	1.08	1.15	1.27	1.32	1.36	1.39
H [A/m]	596	677	774	902	1164	1299	1462	1640
B [T]	1.42	1.47	1.51	1.54	1.56	1.60	1.64	1.72
H [A/m]	1851	2262	2685	3038	3395	4094	4756	7079

Tabelul 6.13: Punctele curbei de magnetizare

Problema are 4 parametri (R_1, L_2, L_3, L_4) care pot varia continuu între limitele prezentate în tabelul 6.14.

	R_1 [mm]	L_2 [mm]	L_3 [mm]	L_4 [mm]
min	5	12.6	14	4
max	9.4	18	45	19

Tabelul 6.14: Restricții de domeniu ale variabilelor de optimizare

Funcția obiectiv propusă este

$$F = \sum_{i=1}^n [(B_{xp_i} - B_{xoi})^2 + (B_{yp_i} - B_{yoi})^2]. \quad (6.58)$$

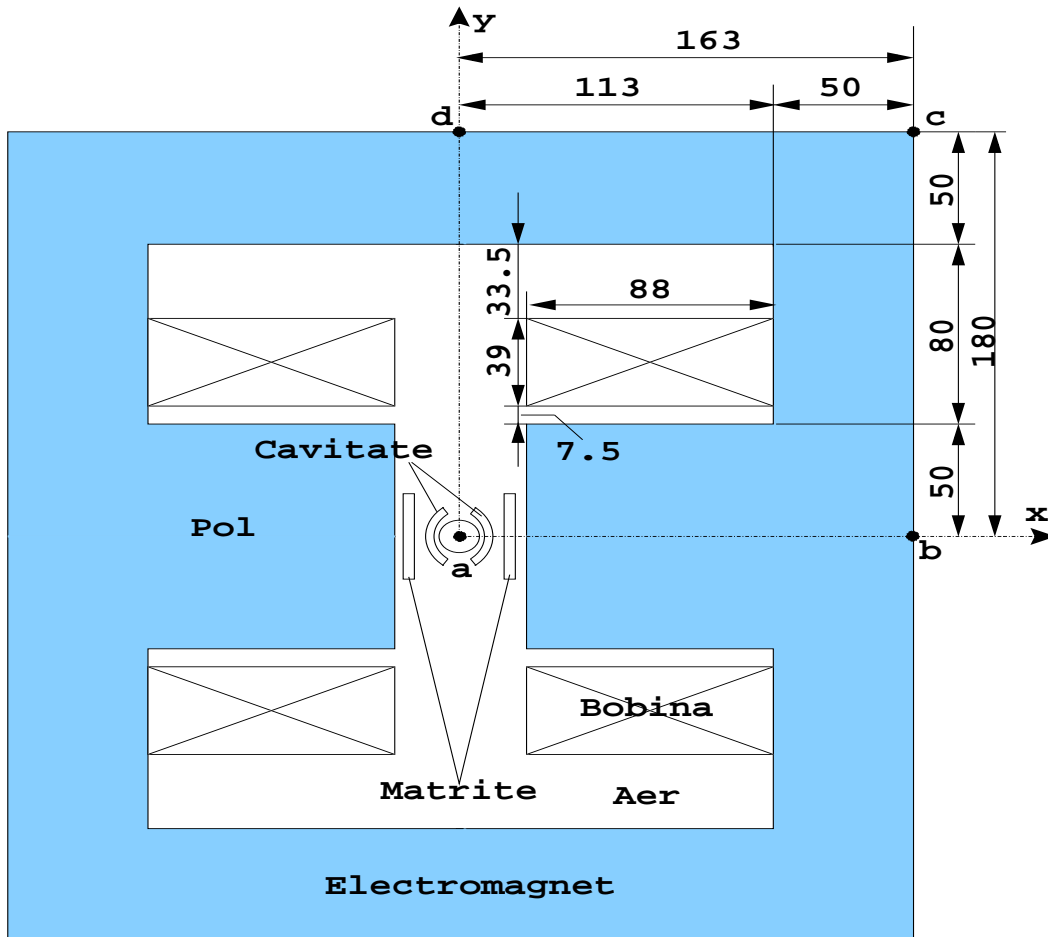
Indicii p și o se referă la valorile calculate, respectiv la valorile specificate. Se consideră $n = 10$ puncte situate pe curba e-f (un arc de cerc de rază 11.75 mm, conform figurii 6.24), în punctele corespunzătoare unghiurilor de: 0, 5, 10, 15, 20, 25, 30, 35, 40 și respectiv 45 de grade.

Se cere de asemenea calculul erorii maxime ε_{Bmax} a modulului și eroarea maximă $\varepsilon_{\theta max}$ a unghiului inducției magnetice, definite astfel:

$$\varepsilon_{Bmax} = \max_{1 \leq i \leq n} \left| \frac{B_{p_i} - B_{o_i}}{B_{o_i}} \right|, \quad (6.59)$$

$$\varepsilon_{\theta max} = \max_{1 \leq i \leq n} |\theta_{B_{p_i}} - \theta_{B_{o_i}}|. \quad (6.60)$$

Autorii acestei probleme au realizat experimental două astfel de matrițe, pentru care au măsurat valorile inducției în punctele specificate. Aceste rezultate experimentale sunt prezentate în [152].



a-b-c-d: Frontiera Dirichlet
d-a: Frontiera Neumann

(a) Vedere de ansamblu

Figura 6.23: Matriță cu electromagnet

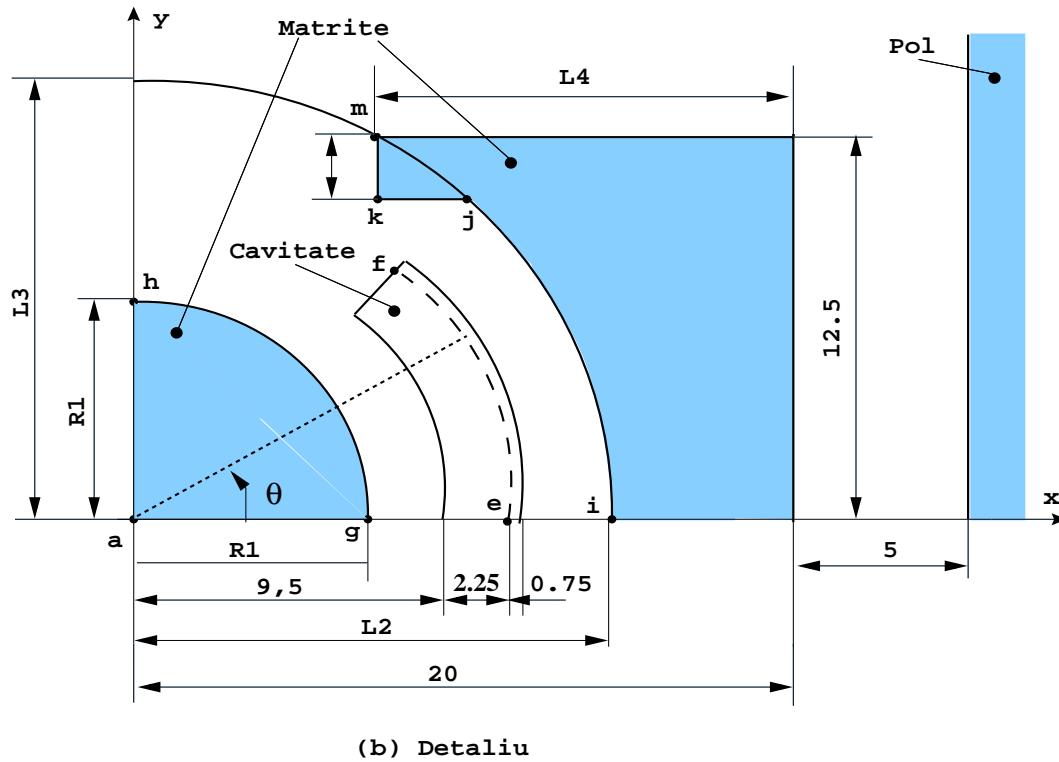


Figura 6.24: Detaliu în zona de interes

6.2.2 Evaluarea funcției de cost

Rezolvarea problemei de câmp cu metoda elementelor finite prezentată în capitolul 3 nu este satisfăcătoare din cauza timpului de calcul inacceptabil de mare pentru includerea într-o problemă de optimizare (8 minute pentru evaluarea funcției obiectiv, așa cum rezultă din testul 4 din capitolul 3). De aceea, vom încerca o altă metodă de rezolvare a problemei neliniare de câmp.

Se propune utilizarea metodei polarizației cu permeabilitatea de calcul egală cu permeabilitatea vidului, metodă bazată pe algoritmul:

1. inițializează \mathbf{I}

2. repetă

2.1. calculează câmpul pentru problema liniară:

$$\operatorname{div} \mathbf{B} = 0, \operatorname{rot} \mathbf{H} = \mathbf{J}, \mathbf{B} = \mu_0 \mathbf{H} + \mathbf{I}, \text{ cu } \mathbf{I} \text{ fixat}$$

2.2. corectează polarizația $\mathbf{I} = \mathbf{B} - \mu_0 \mathbf{h}(\mathbf{B})$

până când (diferența între două polarizații succesive devine suficient de mică)

În algoritmul de mai sus h reprezintă dependența neliniară $H = h(B)$. Pe parcursul algoritmului polarizația magnetică I va fi diferită de zero doar în domeniile feromagnetice.

Problema liniară care trebuie rezolvată la pasul 2.1 este echivalentă din punct de vedere al inducției magnetice \mathbf{B} cu rezolvarea problemei:

$$\operatorname{div} \mathbf{B} = 0 \quad , \quad \operatorname{rot} \mathbf{H}_1 = \mathbf{J} + \frac{1}{\mu_0} \operatorname{rot} \mathbf{I} \quad , \quad \mathbf{B} = \mu_0 \mathbf{H}_1, \quad (6.61)$$

adică cu o problemă omogenă (în care permeabilitatea este μ_0 peste tot), dar în care în loc de materialul neliniar există o distribuție de curenți cu densitatea de volum $\mathbf{J}_e = \operatorname{rot} \mathbf{I} / \mu_0$. Problema liniară fiind omogenă, pentru rezolvarea ei se poate aplica formularea integrală.

Soluția problemei (6.61) poate fi obținută prin superpoziția

$$\mathbf{B} = \mathbf{B}_S + \mathbf{B}_I, \quad (6.62)$$

unde \mathbf{B}_S reprezintă câmpul magnetic dat de curenții impuși (din bobine) \mathbf{J} , iar \mathbf{B}_I reprezintă câmpul magnetic dat de polarizațiile magnetice \mathbf{I} .

Singurele domenii care trebuie discretizate sunt doar domeniile feromagnetice. Aceste domenii vor fi împărțite în elemente în care polarizația \mathbf{I} este presupusă constantă. Pentru a asigura convergența procedurii iterative corecția polarizației în fiecare element trebuie făcută în funcție de valoarea medie a componentei \mathbf{B}_I din element [68]. Factorul de contracție se poate calcula cu formula

$$\theta = 1 - \frac{\mu_0}{\mu_{\max}} = 1 - \frac{4\pi 10^{-7}}{0.11/140} = 0.9984, \quad (6.63)$$

iar eroarea față de soluția exactă se poate evalua cu formula:

$$\|\mathbf{B}^* - \mathbf{B}^n\|_\nu \leq \frac{1}{1 - \theta} \|\mathbf{I}^n - \mathbf{I}^{n-1}\|_\nu \approx 625 \|\mathbf{I}^n - \mathbf{I}^{n-1}\|_\nu. \quad (6.64)$$

În aer, eroarea se poate evalua cu formula [42]

$$\|\mathbf{B}^* - \mathbf{B}^n\|_\nu \leq \frac{\theta}{2\sqrt{1 - \theta}} \|\mathbf{I}^n - \mathbf{I}^{n-1}\|_\nu \approx 12.5 \|\mathbf{I}^n - \mathbf{I}^{n-1}\|_\nu. \quad (6.65)$$

• Câmpul magnetic datorat curenților impuși

Fie un domeniu plan-paralel poligonal Ω , parcurs de un curent distribuit uniform cu densitatea $\mathbf{J} = J\mathbf{k}$. Câmpul magnetic produs de acest curent într-un punct este

$$\begin{aligned} \mathbf{B} &= \frac{\mu_0}{2\pi} \int_{\Omega} \frac{J\mathbf{k} \times \mathbf{R}}{R^2} d\Omega = -\frac{\mu_0 J}{2\pi} \mathbf{k} \times \int_{\Omega} \operatorname{grad}(\ln R) d\Omega = -\frac{\mu_0 J}{2\pi} \mathbf{k} \times \oint_{\partial\Omega} (\ln R) \mathbf{n}_{\Omega} dl_{\Omega} = \\ &= -\frac{\mu_0 J}{2\pi} \mathbf{k} \times \sum_{m \in \partial\Omega} \int_m (\ln R) \mathbf{n}_{\Omega} dl_{\Omega} = -\frac{\mu_0 J}{2\pi} \sum_{m \in \partial\Omega} \int_m (\ln R) (\mathbf{k} \times \mathbf{n}_{\Omega}) dl_{\Omega} = \\ &= -\frac{\mu_0 J}{2\pi} \sum_{m \in \partial\Omega} \mathbf{u}_t \int_m (\ln R) dl_{\Omega}. \end{aligned} \quad (6.66)$$

• **Media câmpului magnetic datorat polarizațiilor**

Fie un element Ω_k , în care polarizația este \mathbf{I}_k . Media câmpului magnetic într-un element Ω_i datorat polarizației \mathbf{I}_k este

$$\tilde{\mathbf{B}}_i(k) = \frac{1}{\sigma(\Omega_i)} \int_{\Omega_i} \text{rot } A \, d\Omega_i = \frac{1}{\sigma(\Omega_i)} \oint_{\partial\Omega_i} (\mathbf{n}_i \times \mathbf{A}) \, dl_i. \quad (6.67)$$

Potențialul vector este $\mathbf{A} = A\mathbf{k}$, unde

$$\begin{aligned} A &= -\frac{\mu_0}{2\pi} \int_{\Omega_k} J_e \ln R \, d\Omega_k = -\frac{\mu_0}{2\pi} \int_{\Omega_k} (\mathbf{J}_e \cdot \mathbf{k}) \ln R \, d\Omega_k = \\ &= -\frac{1}{2\pi} \int_{\Omega_k} (\text{rot } \mathbf{I} \cdot \mathbf{k}) \ln R \, d\Omega_k = -\frac{1}{2\pi} \oint_{\partial\Omega_k} (-\mathbf{n}_k \times \mathbf{I}) \cdot \mathbf{k} \ln R \, dl_k = \\ &= \frac{1}{2\pi} \oint_{\partial\Omega_k} \ln R \mathbf{I} \cdot d\mathbf{l}_k. \end{aligned} \quad (6.68)$$

Înlocuind (6.68) în (6.67), rezultă

$$\begin{aligned} \tilde{\mathbf{B}}_i(k) &= \frac{1}{2\pi\sigma(\Omega_i)} \oint_{\partial\Omega_i} \left(\mathbf{n}_i \times \oint_{\partial\Omega_k} \ln R \mathbf{I}_k \cdot d\mathbf{l}_k \right) dl_i = \\ &= -\frac{1}{2\pi\sigma(\Omega_i)} \sum_{m_i \in \Omega_i} \int_{m_i} \oint_{\partial\Omega_k} \ln R (\mathbf{I}_k \cdot d\mathbf{l}_k) d\mathbf{l}_i = \\ &= -\frac{1}{2\pi\sigma(\Omega_i)} \mathbf{I}_k \sum_{m_i \in \Omega_i} \sum_{m_k \in \Omega_k} \int_{m_i} \int_{m_k} \ln R (d\mathbf{l}_k; d\mathbf{l}_i). \end{aligned} \quad (6.69)$$

Vom nota

$$\tilde{\mathbf{B}}_i(k) = -\frac{1}{\sigma(\Omega_i)} \bar{\alpha}_{ik} \mathbf{I}_k, \quad (6.70)$$

unde

$$\begin{aligned} \bar{\alpha}_{ik} &= \frac{1}{2\pi} \sum_{m_i \in \Omega_i} \sum_{m_k \in \Omega_k} \int_{m_i} \int_{m_k} \ln R (d\mathbf{l}_k; d\mathbf{l}_i) = \\ &= \frac{1}{2\pi} \sum_{m_i \in \Omega_i} \sum_{m_k \in \Omega_k} \int_{m_i} \int_{m_k} \ln R \, dl_i \, dl_k (\mathbf{u}_i; \mathbf{u}_k) = \\ &= \frac{1}{2\pi} \sum_{m_i \in \Omega_i} \sum_{m_k \in \Omega_k} \left[\frac{1}{l_i l_k} \int_{m_i} \int_{m_k} \ln R \, dl_i \, dl_k \right] (\mathbf{l}_i; \mathbf{l}_k). \end{aligned} \quad (6.71)$$

Media câmpului magnetic într-un element Ω_i , datorat tuturor elementelor neliniare este

$$\tilde{\mathbf{B}}_i = -\frac{1}{\sigma(\Omega_i)} \sum_{k=1}^{ne} \bar{\alpha}_{ik} \mathbf{I}_k. \quad (6.72)$$

Pentru calculul funcției obiectiv mai avem nevoie de câmpul magnetic datorat polarizațiilor (6.73). De asemenea, pentru diferite reprezentări grafice sunt utile și proceduri care să calculeze potențialul vector (6.74) și (6.68).

• Câmpul magnetic datorat polarizațiilor

Fie un element Ω_k , în care polarizația este \mathbf{I}_k . Câmpul magnetic într-un punct, datorat polarizației \mathbf{I}_k este

$$\begin{aligned}
 \mathbf{B} &= \frac{\mu_0}{2\pi} \int_{\Omega_k} \frac{\mathbf{J}_e \times \mathbf{R}}{R^2} d\Omega_k = \frac{\mu_0}{2\pi} \int_{\Omega_k} \frac{J_e \mathbf{k} \times \mathbf{R}}{R^2} d\Omega_k = \frac{\mu_0}{2\pi} \int_{\Omega_k} \frac{(\mathbf{J}_e \cdot \mathbf{k})(\mathbf{k} \times \mathbf{R})}{R^2} d\Omega_k = \\
 &= \frac{1}{2\pi} \int_{\Omega_k} \frac{[(\text{rot } \mathbf{I}) \cdot \mathbf{k}](\mathbf{k} \times \mathbf{R})}{R^2} d\Omega_k = \frac{1}{2\pi} \oint_{\partial\Omega_k} \frac{[(-\mathbf{n}_k \times \mathbf{I}_k) \cdot \mathbf{k}](\mathbf{k} \times \mathbf{R})}{R^2} dl_k = \\
 &= -\frac{1}{2\pi} \oint_{\partial\Omega_k} \frac{\mathbf{k} \times \mathbf{R}}{R^2} (\mathbf{u}_t \cdot \mathbf{I}_k) dl_k = -\frac{1}{2\pi} \oint_{\partial\Omega_k} \frac{\mathbf{k} \times \mathbf{R}}{R^2} (\mathbf{I}_k \cdot d\mathbf{l}) = \\
 &= -\frac{I_k}{2\pi} \sum_{m \in \partial\Omega_k} \cos(\mathbf{I}_k, d\mathbf{l}) \int_m \frac{\mathbf{k} \times \mathbf{R}}{R^2} dl. \tag{6.73}
 \end{aligned}$$

• Potențialul vector datorat curenților impuși

Fie un domeniu plan-paralel poligonal Ω , parcurs de un curent distribuit uniform cu densitatea $\mathbf{J} = J\mathbf{k}$. Potențialul magnetic vector produs de acest curent într-un punct este $\mathbf{A} = A\mathbf{k}$, unde

$$\begin{aligned}
 A &= -\frac{\mu_0 J}{2\pi} \int_{\Omega} \ln R d\Omega = -\frac{\mu_0 J}{4\pi} \int_{\Omega} [\text{div}(\mathbf{R} \ln R) - 1] d\Omega = \\
 &= \frac{\mu_0 J}{4\pi} \oint_{\partial\Omega} (\mathbf{R} \cdot \mathbf{n}) (\ln R) dl + \text{const} = \\
 &= \frac{\mu_0 J}{4\pi} \sum_{m \in \partial\Omega} (\mathbf{R} \cdot \mathbf{n}) \int_m (\ln R) dl + \text{const}. \tag{6.74}
 \end{aligned}$$

Observații:

1. Datorită simetriei problemei structurile de date care conțin informații despre noduri, elemente și muchii reprezintă doar un sfert din domeniul problemei. Integralele trebuie însă realizate pe întregul domeniu de calcul. În această situație, coeficienții $\bar{\alpha}_{ik}$ nu reprezintă numai efectul polarizației I_k (corespunzătoare unui element din cadrantul 1) ci și a elementelor simetrice cu el.
2. Timpul necesar iterațiilor neliniare ar putea fi redus dacă inițializarea se află cât mai aproape de soluția exactă. În acest scop, un tabel cu soluții (de exemplu pentru cazul cel mai bun întâlnit) ar putea fi utilizat ca vector de inițializare.
3. Timpul de calcul pentru rezolvarea problemei de câmp magnetic depinde esențial de numărul elementelor de discretizare. Un algoritm de discretizare adaptivă poate conduce la un număr minim de elemente de discretizare. Drept criteriu de calitate a unui nod se poate adopta diferența dintre valoarea medie și cea minimă sau maximă (extremă aflată pe frontiera elementului).

4. Datorită faptului că în decursul optimizării se schimbă doar o mică parte din geometrie, câmpul datorat curenților impuși precum și matricea α se modifică puțin de la o evaluare la alta. Timpul de optimizare poate fi folosit în mod optim dacă coeficienții constanți de-a lungul procesului de optimizare sunt evaluați o singură dată.

• **Algoritmul principal**

1.; citește structuri de date

întreg NUMAR_NODURI, NUMAR_ELEMENTE, NUMAR_MUCHII

tablou real noduri[NUMAR_NODURI,2] ; coordonate x, y

tablou întreg elemente[NUMAR_ELEMENTE,7] ; noduri ale elementelor

tablou întreg muchii[NUMAR_MUCHII,4] ; nod inițial și final, elemente la care
; aparține (cu semn)

întreg SIM_X, SIM_Y ; tipuri de simetrie (0 = fără, 1 = par,
; -1 = impar)

; curba de material

.....

întreg N_SURSE ; descrierea bobinelor

tablou real surse[N_SURSE,2] ; J și numărul de vârfuri

tablou real vf_surse ; coordonatele vârfurilor surselor

; alte declarații

.....

2.calculeaza_Bimpus(BSx,BSy) ; câmpul impus în centrele elementelor

3.calculeaza_distante_geometrice(dg) ; integralele duble din $\ln R$
; dg este o matrice cu NUMAR_MUCHII
; linii și un număr de coloane egal cu
; nr_cadrane*NUMAR_MUCHII, unde
; nr_cadrane poate fi 1, 2 sau 4

4.calculeaza_alpha(α) ; α este o matrice pătrată de dimensiune
; 2*NUMAR_ELEMENTE

5.initializari(pol_x,pol_y) ; inițializări polarizații

6.repetă

norma_dI = 0

norma_I = 0

pentru e = 1, NUMAR_ELEMENTE

calcul_BImediu(e,bIx,bIy)

bx = bIx + BSx(e)

by = bIy + BSy(e)

corecteaza_pol(bx,by,px,py)

$$\begin{aligned} \text{pol_x_nou}(e) &= \text{px} \\ \text{pol_y_nou}(e) &= \text{py} \\ dI_x &= \text{pol_x}(e) - \text{px} \\ dI_y &= \text{pol_y}(e) - \text{py} \\ \text{norma_dI} &= \text{norma_dI} + \text{aria}(e) * (dI_x * dI_x + dI_y * dI_y) \\ \text{norma_I} &= \text{norma_I} + \text{aria}(e) * (\text{px} * \text{px} + \text{py} * \text{py}) \end{aligned}$$

$$\text{norma_dI} = \text{sqrt}(\text{norma_dI})$$

$$\text{norma_I} = \text{sqrt}(\text{norma_I})$$

pentru $e = 1$, NUMAR_ELEMENTE

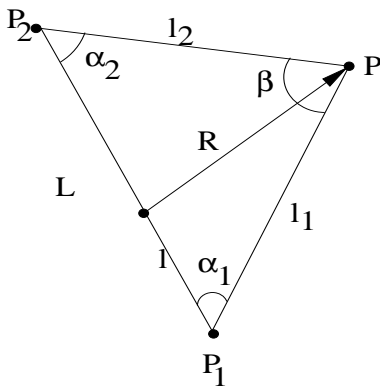
$$\text{pol_x}(e) = \text{pol_x_nou}(e)$$

$$\text{pol_y}(e) = \text{pol_y_nou}(e)$$

până când $(\text{norma_dI} \leq \text{norma_I})$

• **Calculul integralelor care intervin**

Pentru calculul câmpului magnetic datorat curenților impuși (formula (6.66)) și pentru calculul potențialului vector datorat polarizațiilor (formula (6.68)) sau curenților impuși (formula (6.74)) este necesar calculul integralei simple din $\ln(R)$.



Această integrala se poate calcula analitic, rezultatul fiind (vezi notațiile din figura 6.25, unde $L = P_1P_2$):

$$\begin{aligned} \int_{P_1}^{P_2} \ln R \, dl &= l_2 \cos \alpha_2 \ln l_2 + l_1 \cos \alpha_1 \ln l_1 - \\ &- L + \beta l_1 \sin \alpha_1 \end{aligned} \tag{6.75}$$

Figura 6.25: Notații folosite la calculul integralei simple

Pentru calculul coeficienților de proporționalitate (formula (6.71)) este necesar calculul integralei duble din $\ln(R)$. Pentru muchiile care au un nod comun se vor utiliza formule analitice, astfel:

Dacă P_1P_2 și P_3P_4 coincid, atunci:

$$\int_{P_1}^{P_2} \int_{P_3}^{P_4} \ln R \, dl_1 \, dl_2 = L^2 \left(\ln L - \frac{3}{2} \right) \tag{6.76}$$

unde am notat cu L lungimea muchiei.

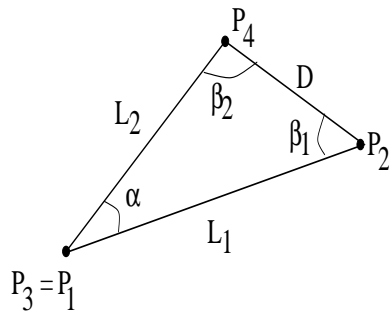


Figura 6.26: Notății folosite la calculul integralei duble - cazul în care muchiile au un nod comun

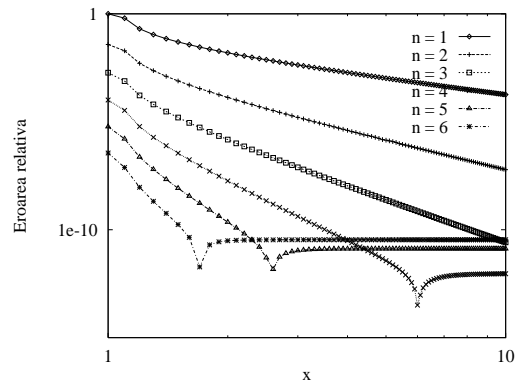


Figura 6.27: Eroarea în funcție de x

Dacă P_1P_2 și P_3P_4 au un nod comun, atunci (vezi notațiile din figura 6.26):

$$\int_{P_1}^{P_2} \int_{P_3}^{P_4} \ln R \, dl_1 \, dl_2 = \left(L_1 L_2 \sin^2 \alpha - \frac{D^2}{2} \cos \alpha \right) \ln D + \\ + \frac{\cos \alpha}{2} (L_1^2 \ln L_1 + L_2^2 \ln L_2) + \frac{\sin \alpha}{2} (L_2^2 \beta_2 + L_1^2 \beta_1) - \\ - \frac{3}{2} L_1 L_2 \quad (6.77)$$

În cazul muchiilor care nu au un punct comun, se va utiliza integrarea numerică. Ordinul de integrare se stabilește în funcție de poziția relativă a celor două muchii, și anume de distanța dintre centre raportată la lungimea unei muchii.

Analiza erorii a fost făcută pentru integrala simplă, în cazul în care segmentul este centrat în origine, orizontal, de lungime unitară, iar punctul de calcul are coordonatele $(x, 0)$. Variația erorii în funcție de x pentru ordine de integrare cuprinse între 1 și 6 este prezentată în figura 6.27.

Graficul în scară dublu logaritmică din figura 6.27 arată că: pentru $n = 1, 2$ și 3 variația este liniară pentru $x \in [1, 10]$, pentru $n = 4$ liniaritatea se păstrează pentru $x \in [1, 5.5]$, pentru $n = 5 \rightarrow x \in [1, 2.4]$ iar pentru $n = 6 \rightarrow x \in [1, 1.6]$. Pe aceste intervale $\ln \varepsilon = a \ln x + b$, respectiv $\varepsilon = Cx^a$, unde $C = e^b$. Făcând regresia liniară a datelor, rezultă următoarele valori:

n	C	a
1	.173862e+00	-3.096321
2	6.173626e-03	-5.128433
3	2.819749e-04	-7.160817
4	2.645225e-05	-9.933035
5	5.749159e-05	-11.629652
6	9.605960e-05	-13.928753

Din prelucrarea acestor date, rezultă ca majorant al erorii expresia:

$$\varepsilon_m = \frac{8000}{(20x)^{n+2}} \quad (6.78)$$

Figurile 6.28 ÷ 6.33 prezintă graficele de eroare pentru cele șase ordine de integrare, precum și graficul funcției care majorează aceste erori, funcție definită de relația (6.78).

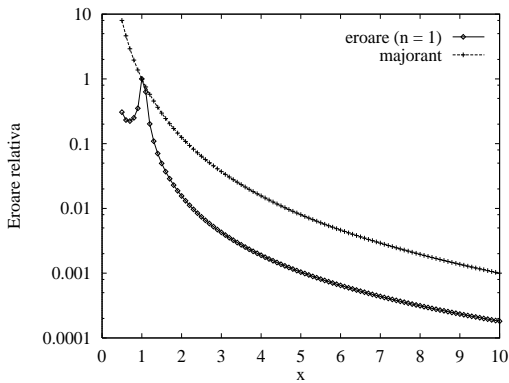


Figura 6.28: Eroare și majorant, $n = 1$

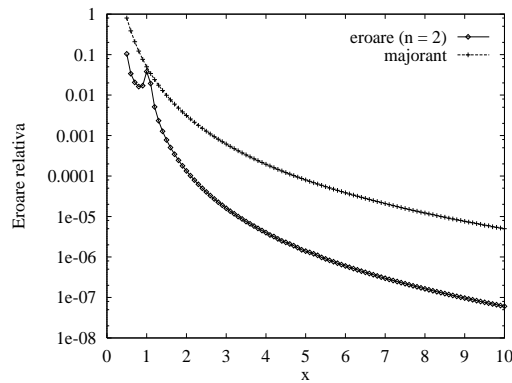


Figura 6.29: Eroare și majorant, $n = 2$

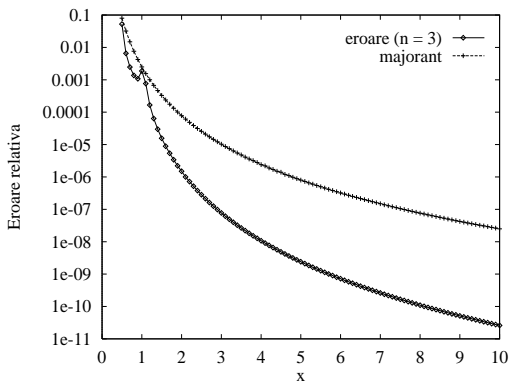


Figura 6.30: Eroare și majorant, $n = 3$

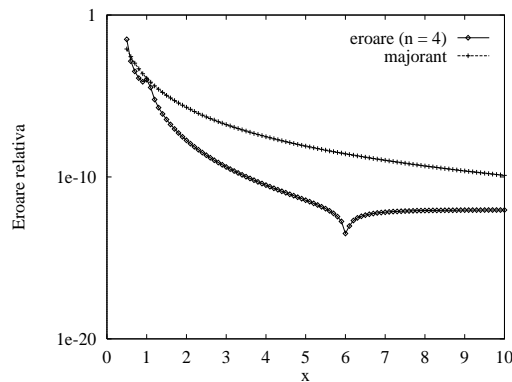
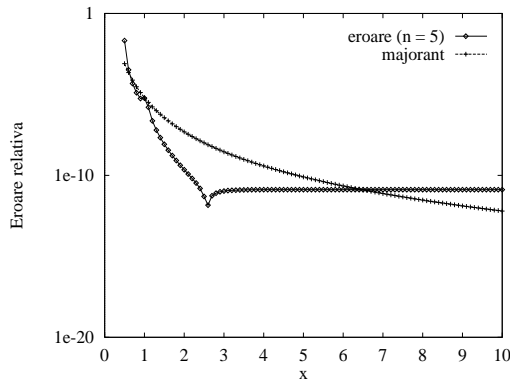
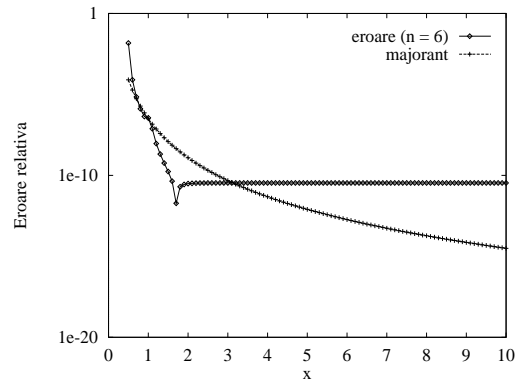


Figura 6.31: Eroare și majorant, $n = 4$

În concluzie, formula propusă este acceptabilă. În consecință, pentru o valoare impusă a erorii, ordinul de integrare se stabilește cu formula

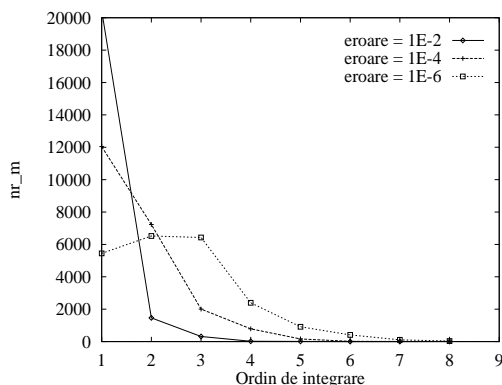
$$n = \left\lceil \frac{\ln \left(\frac{8000}{\varepsilon} \right)}{\ln(20x)} - 2 \right\rceil \quad (6.79)$$

Formula (6.79) a fost aplicată în rezolvarea problemei TEAM 25. Dacă n care rezultă din formulă este mai mic decât 1, atunci lui i se atribuie valoarea 1. Dacă n este mai mare decât 8, atunci lui i se atribuie valoarea 8.

Figura 6.32: Eroare și majorant, $n = 5$ Figura 6.33: Eroare și majorant, $n = 6$

Configurația testată are 57 de muchii și 4 simetrii. Rezultă că sunt de calculat aproximativ 11372 astfel de integrale duble. Fie nr_m un vector de întregi de dimensiune 8, inițializat cu zero. Pentru fiecare integrală dublă, rezultă două ordine de integrare n_1 și n_2 . Se incrementează atunci $nr_m(n_1)$ și $nr_m(n_2)$. În final, suma componentelor vectorului nr_m trebuie să fie de aproximativ $2 \cdot 11372 = 22744$. Tabelul următor arată componentele acestui vector pentru diferite valori ale erorii impuse. Timpul din tabel reprezintă timpul necesar (în Scilab) până la începerea iterațiilor neliniare (timpul necesar calcului tuturor integralelor duble dar și calcului altor mărimi necesare: arii, lungimi, asamblat matrice α).

eroare	$nr_m(1)$	(2)	(3)	(4)	(5)	(6)	(7)	(8)	timp [s]
1E-2	20450.	1463.	314.	19.	8.	0.	0.	4.	248
1E-4	12041.	7228.	2006.	788.	164.	16.	7.	8.	279
1E-6	5449.	6523.	6428.	2395.	913.	411.	108.	31.	341

Figura 6.34: Vectorul nr_m

Aceste rezultate sunt prezentate grafic în figura 6.34. Graficul validează formula (6.79), iar distribuția lor arată că ordinul maxim ales egal cu 8 este rezonabil (dacă ar fi fost prea mic, s-ar fi constatat o înghesuire la capăt)). De asemenea, din punct de vedere al timpului de calcul, trecerea la o eroare de 10^{-4} nu necesită un timp suplimentar mult mai mare. Aceasta va fi valoarea folosită în programul de optimizare.

Ultima integrală necesară este integrala simplă din $\mathbf{k} \times \mathbf{R}/R^2$, care intervine la calculul funcției obiectiv (câmpul magnetic datorat polarizațiilor, formula (6.73)). Să notăm

$$\mathbf{I}_0 = \int_m \frac{\mathbf{k} \times \mathbf{R}}{R^2} dl = I_{0x}\mathbf{i} + I_{0y}\mathbf{j}. \quad (6.80)$$

Utilizând notațiile din figura 6.25, componentele integralei \mathbf{I}_0 au expresiile:

Dacă $P(x_0, y_0)$ nu este pe dreapta P_1P_2 ($\sin \alpha_1 \neq 0$), atunci

$$I_{0x} = \frac{y_2 - y_1}{L} \left(\ln l_2 - \ln l_1 + \beta \frac{\cos \alpha_1}{\sin \alpha_1} \right) + (y_1 - y_0) \frac{\beta}{l_1 \sin \alpha_1}, \quad (6.81)$$

$$I_{0y} = -\frac{x_2 - x_1}{L} \left(\ln l_2 - \ln l_1 + \beta \frac{\cos \alpha_1}{\sin \alpha_1} \right) - (x_1 - x_0) \frac{\beta}{l_1 \sin \alpha_1}. \quad (6.82)$$

Dacă $P(x_0, y_0)$ este pe dreapta P_1P_2 ($\sin \alpha_1 = 0$), atunci

$$I_{0x} = \frac{y_2 - y_1}{L} \left(\ln l_2 - \ln l_1 - 1 - \frac{l_1 \cos \alpha_1}{L - l_1 \cos \alpha_1} \right) - (y_1 - y_0) \left(\frac{1}{l_1 \cos \alpha_1} + \frac{1}{L - l_1 \cos \alpha_1} \right), \quad (6.83)$$

$$I_{0y} = -\frac{x_2 - x_1}{L} \left(\ln l_2 - \ln l_1 - 1 - \frac{l_1 \cos \alpha_1}{L - l_1 \cos \alpha_1} \right) + (x_1 - x_0) \left(\frac{1}{l_1 \cos \alpha_1} + \frac{1}{L - l_1 \cos \alpha_1} \right). \quad (6.84)$$

6.2.3 Rezultate numerice

Tabelul 6.15 conține rezultatele obținute cu metoda integrală atât în cazul iterațiilor Picard-Banach cât și în cazul iterațiilor Newton iar figura 6.35 prezintă spectrul câmpului obținut cu cea mai rară discretizare folosită (cu 20 de elemente). Programele corespunzătoare au fost scrise în Scilab [4] și rulate pe un PC Pentium (166 MHz), sub Linux [3] iar configurația de test folosită a fost cea mai bună configurație raportată până acum pentru această problemă [24].

Se observă că timpul necesar calcului matricei $\bar{\alpha}$ este mult mai mare decât timpul necesar iterațiilor neliniare. În timpul procesului de optimizare, rediscrétizarea problemei ar implica recalcularea acestei matrice, deci un timp CPU foarte mare. Se observă de asemenea că metoda Newton este mult mai rapid convergentă decât metoda Picard-Banach. Aplicarea suprarelaxării la metoda Picard-Banach a condus la o înjumătățire a numărului de iterații, dar la același timp de calcul (datorită calcului factorului de suprarelaxare).

În această situație, metoda integrală ar merita să fie integrată într-un program de optimizare doar dacă se adoptă o abordare de tip celule, în care, geometria maximală este discretizată în celule dreptunghiulare, iar apoi, în funcție de dimensiunile actuale ale parametrilor, celulele sunt active sau nu (materialul este feromagnetic sau aer).

Nr.elem/ nr.muchii	Metoda	Nr. iterații	Timp (s)		Rezultate
			$\bar{\alpha}$	iterații	
20/ 59	PB	192	330	120	$f = 2.69E-2$ $\varepsilon_B = 0.213$ $\varepsilon_\theta = 0.158$ rad
	Newton	3		3	$f = 2.66E-2$ $\varepsilon_B = 0.211$ $\varepsilon_\theta = 0.157$ rad
32/ 89	PB	194	870	207	$f = 1.55E-2$ $\varepsilon_B = 0.186$ $\varepsilon_\theta = 0.081$ rad
	Newton	4		7.5	$f = 1.50E-2$ $\varepsilon_B = 0.184$ $\varepsilon_\theta = 0.081$ rad
44/ 117	PB	194	1898	346	$f = 1.34E-2$ $\varepsilon_B = 0.167$ $\varepsilon_\theta = 0.059$ rad
	Newton	4		13	$f = 1.28E-2$ $\varepsilon_B = 0.165$ $\varepsilon_\theta = 0.059$ rad
63/ 160	PB	417	5280	1355	$f = 9.72E-3$ $\varepsilon_B = 0.152$ $\varepsilon_\theta = 0.066$ rad
	Newton	4		28	$f = 9.07E-3$ $\varepsilon_B = 0.149$ $\varepsilon_\theta = 0.066$ rad

Tabelul 6.15: Rezultate numerice ale metodei integrale

Cu o abordare de tip celule, matricea $\bar{\alpha}$ trebuie calculată o singură dată înainte procesului de optimizare propriu-zis. Dezavantajul acestei abordări îl constituie faptul că interfețele dintre materialul feromagnetic și aer nu rezultă netede ci în trepte. Pentru ca acest lucru să nu perturbe foarte mult acuratețea funcției obiectiv celulele trebuie să fie suficient de mici. Pentru un grid în zona de interes cu un pas de 1 mm (figura 6.36) rezultă o rețea cu 317 elemente și 681 muchii. Calculul matricei $\bar{\alpha}$ durează în Scilab aproximativ 5 zile, deci nepermis de mult.

Având în vedere aceste rezultate s-a revenit la metoda elementelor finite dar algoritmul inițial a fost îmbunătățit cu proceduri de calcul numeric oferite de PETSc [13].

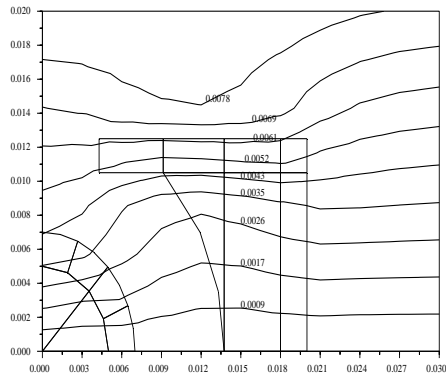


Figura 6.35: Linii de câmp obținute cu metoda integrală pentru rețeaua cu 20 elemente

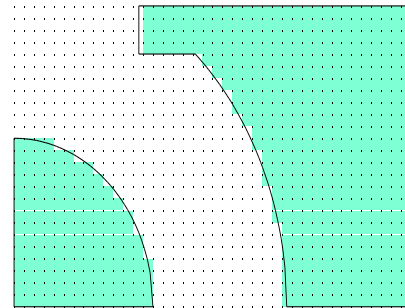


Figura 6.36: Discretizare de tip celule

Tabelul 6.16 prezintă rezultatele obținute pentru diferite metode de accelerare de tip Krilov folosite pentru rezolvarea problemei liniare: gradienti conjugați (CG) și reziduu minimal generalizat (GMRES). Iată câteva din metodele de preconditionare folosite: Schwartz aditiv (ASM), bloc Gauss-Seidel (BGS), bloc Jacobi (BJACOBI), suprarelaxare succesivă (SOR), SOR combinată cu metoda Eisentat (EISENSTAT), factorizare LU (LU), factorizare LU incompletă (ILU), solver liniar (SLES). Metoda CG s-a dovedit a fi mai rapidă decât metoda GMRES, și cel mai bun preconditionator a fost ILU. CG cu SOR și GMRES cu EISENSTAT nu au fost convergente

Precondiționator (PC)	CG		GMRES	
	Nr. iterații	Timp(s)	Nr. iteraii	Timp(s)
ASM	40	0.36	62	0.79
BGS	40	0.24	62	0.66
BJACOBI	40	0.22	62	0.79
EISENSTAT	713	3.44	-10000	95.32
ILU	40	0.21	62	0.63
JACOBI	145	0.49	630	4.96
LU	1	0.26	1	0.26
SLES	2	0.98	2	1.31
SOR	-10000	44.71	398	4.02

Tabelul 6.16: Teste numerice - problema liniară - diferiți preconditionatori

Metoda CG a fost aleasă pentru testele din tabelul 6.17, care se referă de asemenea

la problema liniară dar pentru diferite rețele de discretizare. Precondiționatorul ILU se comportă similar cu precondiționatorul BJACOBI, mult mai bine decât BGS. Numărul de iterații este independent de precondiționator.

Nr. elem.	PC	Timp(s)		Nr. it.	Funcție de cost
		total	solver		
302	BGS	1.64	0.05	16	1.27e-02
302	BJACOBI	1.54	0.01	16	1.27e-02
302	ILU	1.53	0.01	16	1.27e-02
1352	BGS	2.23	0.09	31	1.69e-03
1352	BJACOBI	2.22	0.09	31	1.69e-03
1352	ILU	2.20	0.08	31	1.69e-03
3284	BGS	3.48	0.33	44	1.73e-03
3284	BJACOBI	3.47	0.31	44	1.73e-03
3284	ILU	3.47	0.31	44	1.73e-03

Tabelul 6.17: Teste numerice - problema liniară - diferite rețele

Tabelul 6.18 conține rezultatele pentru cazul neliniar (iterații Picard-Banach), metoda liniară aleasă fiind CG și GMRES, cu precondiționare ILU. Numărul de iterații neliniare a fost același pentru CG și pentru GMRES, dar metoda CG a fost cu 10 % mai rapidă decât GMRES.

Metoda	Toleranța relativă	Nr. iterații neliniare	Timp de rezolvare (s)	Funcție de cost
GMRES	10^{-2}	32	94.86	1.77e-02
GMRES	10^{-3}	64	196.98	1.12e-03
GMRES	10^{-4}	315	1029.56	1.32e-03
CG	10^{-2}	32	87.43	1.77e-02
CG	10^{-3}	64	199.1	1.12e-03
CG	10^{-4}	315	899.03	1.32e-03

Tabelul 6.18: Problema neliniară - iterații Picard-Banach

Tabelul 6.19 conține rezultatele obținute pentru problema neliniară, rezolvată cu iterații Newton. Sistemul liniar a fost rezolvat cu CG și precondiționator BJACOBI. Convergența metodei Newton a fost îmbunătățită cu metodele “line search (ls)” sau “trust region (tr)”. În cazul “ls”, o creștere a toleranței relative peste 10^{-3} nu conduce la nici o îmbunătățire.

Pentru a obține cu acuratețe funcția de cost ($1.3e-03$) timpul de rezolvare în cazul

Metoda	Toleranța relativă	Nr.iterații neliniare	Timp rezolvare (s)	Funcție de cost
ls	10^{-1}	6	2.60	1.13e-03
ls	10^{-2}	30	11.02	1.25e-03
ls	10^{-3}	47	17.31	1.37e-03
ls	10^{-4}	64	23.74	1.37e-03
ls	10^{-5}	687	470.66	1.37e-03
tr	10^{-2}	1	231.09	1.22e+00

Tabelul 6.19: Problema neliniară - iterații Newton.

metodei Newton a fost de 50 de ori mai mic decât în cazul iterațiilor Picard-Banach. Reamintim că metoda Newton clasică nu a converș pentru această problemă și nici îmbunătățirea ei cu algoritmul “ts”.

Figura 6.37 reprezintă variația funcției de cost și a energiei magnetice cu inversul numărului de elemente. Creșterea numărului de elemente conduce la o instabilitate a funcției de cost, fenomen care nu se observă în cazul energiei. Se pare că este inutil a se folosi mai mult de 2000 elemente finite în acest caz.

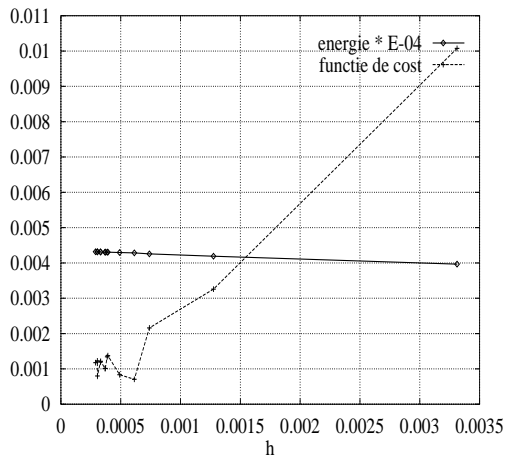


Figura 6.37: Variația energiei și a funcției de cost cu inversul numărului de elemente

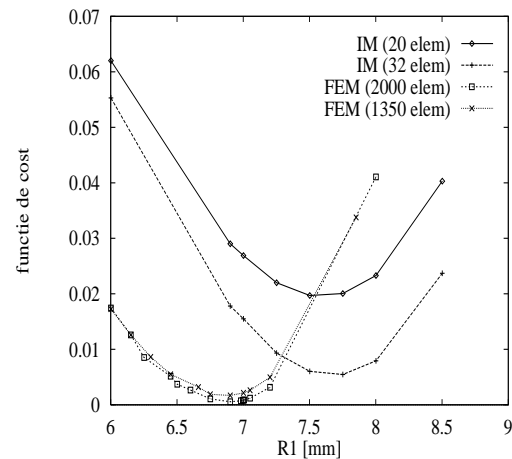
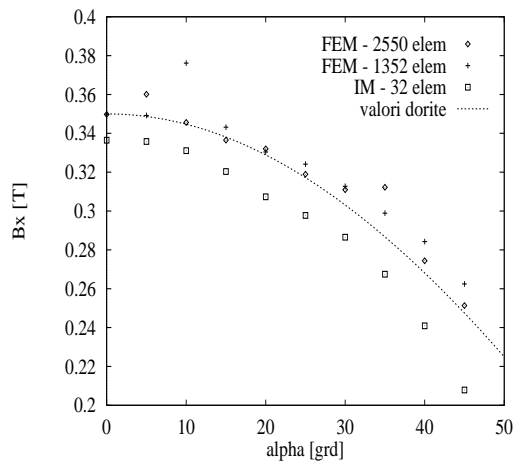
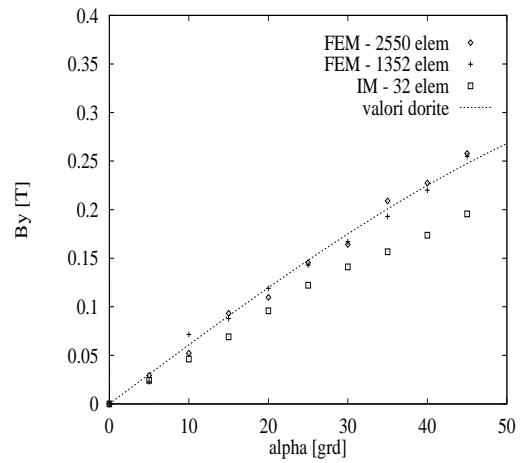
Figura 6.38: Variația funcției de cost pentru diferite valori ale parametrului R_1

Figura 6.38 prezintă variația funcției de cost pentru diferite valori ale parametrului R_1 . Este evident faptul că metoda integrală a folosit prea puține elemente, în timp ce pentru metoda elementelor finite 2000 elemente par a fi de ajuns.

Figurile 6.39 și 6.40 prezintă variația în spațiu a componentelor B_x și B_y ale intensității

Figura 6.39: B_x în funcție de unghiul α Figura 6.40: B_y în funcție de unghiul α

câmpului magnetic pentru aceeași configurație de referință, calculate cu cele două metode. Se poate observa că în metoda integrală variația acestor mărimi este mai netedă decât în metoda elementelor finite datorită faptului că zona de interes în care se calculează câmpul nu este discretizată.

Concluzii:

- Metoda aleasă pentru rezolvarea problemei de câmp în vederea calcului funcției obiectiv este deosebit de importantă într-un algoritm de optimizare. Trebuie făcut un compromis între acuratețe și timp de calcul, dar robustețea este obligatorie. Nu ne putem permite de exemplu ca algoritmul să nu convergă (ca de exemplu în metoda Newton clasică).
- Efort de calcul mare este acceptabil dar o singură dată, de aceea în cazul metodei integrale, abordarea de tip celule este mai potrivită, matricea $\bar{\alpha}$ fiind calculată o singură dată, înaintea procesului de optimizare propriu-zis.
- Pentru obținerea unei aceleiași acurateți pentru funcția obiectiv metoda integrală are nevoie de un număr de elemente de aproximativ 10 ori mai mic decât în cazul metodei elementelor finite, dar are nevoie de un timp de calcul mai mare.
- Metoda integrală mai are avantajul că soluția obținută în zona de interes este mult mai netedă decât în cazul metodei elementelor finite. Pentru a evita aceste discontinuități care rezultă în cazul metodei elementelor finite, pentru evaluarea funcției de cost este necesară o procedură de netezire.
- În cazul metodei elementelor finite, cel mai bun rezultat a fost obținut pentru metoda Newton cu subrelaxare optimală ("line search"), cu un solver iterativ pentru

rezolvarea sistemului liniar (GMRES - Generalised Minimal Residual + BJACOBI - preconditionator Block Jacobi), o eroare impusă de 10^{-3} , toți acești parametri dând un rezultat satisfăcător (timp CPU mai mic de 20 secunde), eroarea relativă a funcției obiectiv fiind $4E-4$.

- Pentru obținerea unei soluții aproximative propunem modificarea criteriului de oprire (de exemplu eroarea impusă poate fi crescută până la 10^{-1} sau numărul maxim de iterații să fie mai mic), mai degrabă decât să se creeze rețele cu un număr mai mic de elemente.

Aplicând o astfel de strategie, algoritmul evoluționist descris în capitolul 5 a fost aplicat optimizării acestei probleme, cu următorii parametri: 3 procese sclav, fiecare având 16 indivizi într-o populație, parametrii P_i : 1, 1, 1, 4, 4, 1, perioada migrației 3, $B = 2$, $A = 0.25$, numărul maxim de generații 60. Procesul a durat 33 ore (cei trei sclavi au fost pe aceeași mașină), iar cea mai bună configurație găsită a fost:

$R_1 = 6.879$ mm	$L_2 = 13.789$ mm
$L_3 = 14.065$ mm	$L_4 = 13.579$ mm
$F = 1.244 \cdot 10^{-3}$	

Diferența relativă dintre configurația găsită și cea de referință este următoarea: R_1 se află la 1.75 % față de valoarea de referință ($R_1 = 7.0018$ mm), L_2 se află la 0.32 % față de valoarea de referință ($L_2 = 13.7448$ mm), L_3 se află la 0.04 % față de valoarea de referință ($L_3 = 14.058$ mm) iar L_4 se află la 0.9 % față de valoarea de referință ($L_4 = 13.7108$ mm).

Se observă că parametrii L_2 , L_3 , L_4 sunt apropiați de cei ai configurației de referință. Valoarea parametrului R_1 ar putea fi îmbunătățită printr-o minimizare unidimensională deterministă. Având în vedere cum variază funcția de cost cu R_1 se poate aplica de exemplu metoda aproximării parabolice pentru găsirea unui valori optime a parametrului R_1 pentru valori fixate ale parametrilor L_2 , L_3 , L_4 . În acest fel, în algoritmul stocastic ar putea fi inclus un algoritm de optimizare deterministă unidimensională așa cum s-a procedat și pentru problema TEAM 22, indivizii care participă la evoluție fiind astfel dintre cei mai buni.

Capitolul 7

Contribuții

Optimizarea dispozitivelor electromagnetice este una dintre cele mai importante dar și cele mai dificile probleme. Încercând să clarifice, clasifice și să rezolve o parte din problemele legate de optimizare, această lucrare aduce următoarele contribuții:

1. Prezentarea pe larg a stadiului actual al metodelor folosite în optimizarea dispozitivelor electromagnetice. Pe lângă clasificările de rigoare se fac și considerații critice asupra rezultatelor prezentate în literatură, în vederea identificării căilor și tendințelor de dezvoltare în viitor a acestui domeniu important al cercetării.
2. Elaborarea și prezentarea unor algoritmi originali pentru rezolvarea problemei de analiză a câmpului electromagnetic. Algoritmii propuși permit, pe de o parte, generarea rețelelor de discretizare adaptate optimal la soluție (prin rafinare succesivă), iar pe de altă parte comută automat între metode diferite de iterații neliniare (metoda polarizației cu relaxare optimală care este garantat convergentă și metoda Newton care este rapidă). Sunt stabilite criteriile pentru rafinare succesivă (un indicator local de eroare) și pentru comutarea polialgoritmului.
3. Formularea și demonstrarea a două teoreme de unicitate în cazul problemelor plan-paralele cu frontieră deschisă, bazate pe folosirea unei formulări integrale pentru domeniul exterior frontierei.
4. Studiul metodelor de calcul a sensibilităților funcției obiectiv, necesare aplicării metodelor deterministe de optimizare de ordin superior fie prin folosirea unei probleme adjuncte, fie prin exploatarea metodei numerice de calcul.
5. Elaborarea unui algoritm evoluționist distribuit și analiza influenței parametrilor săi asupra convergenței procesului de optimizare.
6. Rezolvarea problemelor de optimizare propuse de comunitatea internațională.

Rezultatele din această lucrare au fost comunicate la diverse conferințe internaționale. Astfel, [34, 35, 37, 71, 164] se referă la optimizarea problemei TEAM 22 folosind algoritmi evoluționiști distribuți sau algoritmi evoluționiști distribuți combinați cu strategii deterministe de optimizare, [72] se referă la calculul sensibilităților folosind tehnica variabilelor adjuncte, [36] se referă la optimizarea problemei TEAM 25 și anume la analiza metodei optime pentru calculul funcției obiectiv.

Două din aceste lucrări au fost acceptate pentru publicare în revista *IEEE Transactions on Magnetics* [70, 73].

În concluzie nu există metodă perfectă de optimizare. Pentru orice problemă trebuie încercate și analizate mai multe metode. A utiliza algoritmul de optimizare ca o ”cutie neagră” este foarte periculos, fiecare algoritm comportându-se diferit în funcție de aplicația pe care dorește să o rezolve. Metoda de rezolvare trebuie și ea aleasă cu grijă. Acolo unde problema o permite sunt de preferat metodele analitice sau formule matematice aproximative, cel puțin într-o primă etapă. Până acum se practică combinarea a două metode de optimizare, dar este posibil ca în viitor să se elaboreze strategii mixte cu mai mult de două metode de optimizare. De asemenea acuratețea evaluării funcției obiectiv ar trebui să fie dinamică pe parcursul procesului de optimizare, ca și parametrii acestui proces. Paralelizarea algoritmilor se va impune tot mai mult dat fiind efortul foarte mare de calcul necesar optimizării dispozitivelor electromagnetice. Este posibil ca strategiile de optimizare să fie combinate tot mai mult cu metode de inteligență artificială.

Anexa A

Tipuri de probleme de optimizare

A.1 Enumerarea tipurilor de probleme de optimizare

În cele ce urmează vom nota cu $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$ funcția obiectiv (reală), cu $\mathbf{x} \in \mathbb{R}^n$ vectorul variabilelor, restricțiile sunt reprezentate de funcția $c : \mathbb{R}^n \rightarrow \mathbb{R}^p$ unde unele componente ale lui c sunt restricții de tip inegalitate, de forma $c_i(\mathbf{x}) \leq 0$ pentru i într-o mulțime de indici notată \mathcal{I} , alte componente ale lui c sunt restricții de tip egalitate de tip $c_i(\mathbf{x}) = 0$ pentru indici i dintr-o mulțime de indici notată \mathcal{E} . Pot exista restricții bilaterale de tipul $l_i \leq c_i(\mathbf{x}) \leq u_i$ unde l_i și u_i sunt marginile inferioară și superioară ale restricției, cazul $c_i(\mathbf{x}) = x_i$ corespunzând restricțiilor de domeniu.

Iată tipurile de probleme de optimizare [110]:

1. Probleme de programare neliniară cu restricții¹

În aceste probleme f_0 și c sunt funcții neliniare de \mathbf{x} . Două formulări tipice echivalente sunt

$$\min \{f_0(\mathbf{x}) \mid c_i(\mathbf{x}) \leq 0, i \in \mathcal{I}, c_i(\mathbf{x}) = 0, i \in \mathcal{E}\},$$

și

$$\min \{f_0(\mathbf{x}) \mid c(\mathbf{x}) = \mathbf{0}, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}\}.$$

2. Probleme de programare liniară²

În aceste probleme funcția obiectiv și restricțiile sunt liniare. Formularea standard a acestor probleme este:

$$\min \{\mathbf{c}^T \mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\},$$

¹ "Nonlinear programming"

² "Linear programming"

unde $\mathbf{c} \in \mathbb{R}^n$ este un vector de cost și $\mathbf{A} \in \mathbb{R}^{m \times n}$ este o matrice de restricții. De multe ori se folosește formularea mai convenabilă

$$\min \{ \mathbf{c}^T \mathbf{x} \mid l_i \leq \mathbf{a}_i^T \mathbf{x} \leq u_i, i \in \mathcal{I}, l_i \leq x_i \leq u_i, i \in \mathcal{B} \}.$$

În această formulare restricțiile de tip egalitate sunt descrise prin relații de încadrare în care $l_i = u_i$ pentru $i \in \mathcal{I}$.

3. Programare pătratică³

Aceste probleme au restricții liniare și funcții obiectiv pătratice, formularea problemei făcându-se astfel

$$\min \left\{ \mathbf{c}^T \mathbf{x} + \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} \mid \mathbf{a}_i^T \mathbf{x} \leq b_i, i \in \mathcal{I}, \mathbf{a}_i^T \mathbf{x} = b_i, i \in \mathcal{E} \right\},$$

unde $\mathbf{Q} \in \mathbb{R}^{n \times n}$ este o matrice simetrică. Aceste probleme sunt convexe dacă \mathbf{Q} este pozitiv semidefinită și neconvexe în caz contrar.

4. Probleme de cele mai mici pătrate cu restricții liniare⁴

Unele probleme convexe de programare pătratică pot fi formulate mai natural ca o problemă de cele mai mici pătrate

$$\min \left\{ \frac{1}{2} \|\mathbf{C}\mathbf{x} - \mathbf{d}\|_2^2 \mid \mathbf{a}_i^T \mathbf{x} \leq b_i, i \in \mathcal{I}, \mathbf{a}_i^T \mathbf{x} = b_i, i \in \mathcal{E} \right\},$$

unde matricea coeficienților \mathbf{C} nu este neapărat pătrată.

5. Probleme care au doar restricții de domeniu⁵

În aceste probleme singurele restricții sunt cele care mărginesc componentele vectorului \mathbf{x}

$$\min \{ f_0(\mathbf{x}) \mid \mathbf{l} \leq \mathbf{x} \leq \mathbf{u} \}.$$

6. Probleme fără restricții⁶

sunt cele care nu au nici un fel de restricții (nici măcar de domeniu).

7. Probleme neliniare de cele mai mici pătrate⁷

În aceste probleme funcția obiectiv are forma specială

$$f_0(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^m f_i(\mathbf{x})^2,$$

unde fiecare componentă f_i este numită reziduu. Grupând reziduurile într-o funcție vectorială $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$ putem scrie f_0 ca fiind

$$f_0(\mathbf{x}) = \frac{1}{2} \|f(\mathbf{x})\|_2^2.$$

³ "Quadratic Programming"

⁴ "Constrained linear least square problems"

⁵ "Bound Constrained Problems"

⁶ "Unconstrained problems"

⁷ "Nonlinear least square problems"

8. Sisteme de ecuații neliniare

Rezolvarea unui sistem de ecuații neliniare definit cu ajutorul funcției $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ este un vector \mathbf{x} astfel încât $f(\mathbf{x}) = \mathbf{0}$. Unii algoritmi sunt stânși legați de algoritmi de optimizare fără restricții și de cele mai mici pătrate neliniare, ei rezolvând

$$\min \{ \|f(\mathbf{x})\| \mid \mathbf{x} \in \mathbb{R}^n \},$$

unde $\|\cdot\|$ este de obicei norma L_2 din \mathbb{R}^n .

9. Probleme de optimizare a rețelelor ⁸

În aceste probleme f_0 și c_i au o structură specială care provine de la un graf constând în arce și noduri. Astfel de probleme apar în aplicații care implică distribuția produselor, transportul, comunicațiile. Restricțiile sunt de obicei liniare și fiecare restricție c_i implică de obicei numai una sau două componente ale lui \mathbf{x} . Funcția f_0 poate fi fie liniară fie neliniară, dar este de obicei "separabilă", adică se poate scrie sub forma

$$f_0(\mathbf{x}) = \sum_{i=1}^n f_i(x_i),$$

unde fiecare funcție scalară f_i depinde numai de argumentul x_i .

10. Programare întregă ⁹

Acest termen definește problemele de optimizare în care componentele lui \mathbf{x} sunt întregi. Problemele de tip 1÷9 au presupus valori reale ale componentelor. Termenul *programare mixtă* ¹⁰ definește problemele în care o parte a componentelor lui \mathbf{x} sunt reale iar cealaltă parte sunt întregi. Astfel de probleme sunt mai dificil de rezolvat, de aceea până acum au apărut metode care rezolvă doar problemele de programare liniară și unele cazuri de programare pătratică. Aceste metode se bazează pe tehnica "branch and bound".

Majoritatea problemelor de optimizare nu se pot încadra strict într-una din categoriile de mai sus. De exemplu o problemă de programare liniară este în același timp o problemă de programare pătratică care este în același timp o problemă de programare neliniară. O problemă trebuie plasată în categoria cea mai restrictivă pentru că astfel se pot folosi metode și implementa algoritmi care să țină cont de toate particularitățile problemei.

⁸"Network optimization problems"

⁹"Integer programming"

¹⁰"Mixed-integer programming"

A.2 Probleme de test

Testarea algoritmilor de optimizare care doresc să găsească extreme globale se face cu ajutorul unor probleme ale căror soluții sunt cunoscute. Iată câteva astfel de funcții [104].

1. De Jong (I):

$$\sum_{i=1}^3 x_i^2, \quad \text{unde } -5.12 \leq x_i \leq 5.12.$$

Funcția are un minim global egal cu 0 în punctul $(x_1, x_2, x_3) = (0, 0, 0)$.

2. De Jong (II):

$$100(x_1^2 - x_2)^2, \quad \text{unde } -2.048 \leq x_i \leq 2.048.$$

Funcția are un minim global egal cu 0 în punctul $(x_1, x_2) = (1, 1)$.

3. De Jong (III):

$$\sum_{i=1}^5 \text{integer}(x_i), \quad \text{unde } -5.12 \leq x_i \leq 5.12.$$

Funcția are un minim global egal cu -30 pentru $-5.12 \leq x_i < -5.0$.

4. Schaffer:

$$0.5 + \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1.0 + 0.001(x_1^2 + x_2^2)]^2}, \quad \text{unde } -100 \leq x_i \leq 100.$$

Funcția are un minim global egal cu 0 în punctul $(x_1, x_2) = (0, 0)$.

5. Goldstein-Price:

$$[1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \\ \cdot [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)],$$

unde $-2 \leq x_i \leq 2$.

Funcția are un minim global egal cu 3 în punctul $(x_1, x_2) = (0, -1)$.

6. Branin:

$$a(x_2 - bx_1^2 + cx_1 - d)^2 + e(1 - f)\cos(x_1) + e,$$

unde $-5 \leq x_1 \leq 10$, $0 \leq x_2 \leq 15$, și $a = 1$, $b = 5.1/(4\pi)$, $c = 5/\pi$, $d = 6$, $e = 10$, $f = 1/(8\pi)$.

Funcția are un minim global egal cu 0.397887 în trei puncte diferite: $(x_1, x_2) = (\pi, 12.275)$, $(\pi, 2.275)$, și $(9.42478, 2.475)$.

Anexa B

Optimizări vectoriale

Una din dificultățile întâmpinate în optimizarea dispozitivelor electromagnetice constă în cerința de a satisface mai multe obiective. Problemele care urmăresc satisfacerea simultană a mai multor obiective se numesc probleme de optimizări vectoriale.

B.1 Optimalitate Pareto

Caracteristic pentru problemele de optimizare vectorială este apariția unui conflict de obiective, în care soluțiile care ar minimiza fiecare obiectiv în parte sunt diferite și nu există soluție acolo unde toate obiectivele își ating minimumul. O problemă de minimizare vectorială se formulează astfel:

Să se minimizeze $\mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_K(\mathbf{x}))$ unde $\mathbf{F} : \mathbb{R}^n \rightarrow \mathbb{R}^K$, este supusă la restricțiile $g_i, h_j : \mathbb{R}^n \rightarrow \mathbb{R}$, $g_i(\mathbf{x}) \leq 0$, ($i = 1, 2, \dots, m$) (restricții de tip inegalități), $h_j(\mathbf{x}) = 0$, ($j = 1, 2, \dots, p$) (restricții de tip egalități) și $x_u \leq x_l \leq x_{lu}$, ($l = 1, 2, \dots, n$) (restricții de domeniu).

În definirea soluției optimale se aplică criteriul de optimalitate introdus inițial de Pareto încă din 1896 pentru problemele din economie [137]. *O soluție optimală \mathbf{x}^* în sens Pareto se caută acolo unde nu există o soluție \mathbf{x} în domeniul de căutare $M = \{\mathbf{x} \in \mathbb{R}^n | g_i(\mathbf{x}) \leq 0; h_j(\mathbf{x}) = 0; x_u \leq x_l \leq x_{lu} (\forall) i = 1, \dots, m; j = 1, \dots, p; l = 1, \dots, n\}$ pentru care $f_k(\mathbf{x}) \leq f_k(\mathbf{x}^*)$, ($\forall) k \in [1, K]$, $f_k(\mathbf{x}) < f_k(\mathbf{x}^*)$, pentru cel puțin un $k \in [1, K]$.* O problemă de optimizare în care îmbunătățirea unui obiectiv cauzează degradarea a cel puțin unui alt obiectiv nu are soluție decât în sens optimal Pareto. Figura B.1 arată interpretarea geometrică a soluției optimale în sens Pareto pentru cazul a două obiective care sunt în conflict.

Graficul din figură reprezintă dependența dintre f_1 și f_2 . Pe această curbă există soluțiile posibile ale problemei, fiind marcată regiunea soluțiilor optimale în sens Pareto. Se ob-

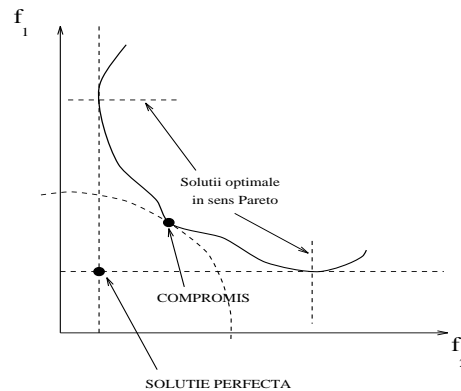


Figura B.1: Interpretarea geometrică a soluțiilor optimale în sens Pareto

servă că nu există un punct pentru care ambele obiective să își atingă minimul. Acel punct este numit "soluție perfectă". O soluție de compromis ar putea fi aceea pentru care distanța dintre soluția perfectă dar nefezabilă și mulțimea soluțiilor optimale în sens Pareto este minimă.

Abordarea unei probleme de optimizări vectoriale are trei aspecte importante: stabilirea funcției obiectiv¹, metodele de a trata restricțiile neliniare și alegerea algoritmului de optimizare care minimizează funcția obiectiv.

B.2 Stabilirea funcției obiectiv în cazul optimizărilor vectoriale

Aplicarea unui algoritm de optimizare are nevoie de o metodă de luare a deciziilor care garantează o soluție din mulțimea de soluții optimale în sens Pareto. Iată câteva metode care se aplică în optimizarea dispozitivelor electromagnetice [136, 137].

• Ponderarea obiectivelor

O primă posibilitate este aceea de a se folosi o funcție obiectiv care este suma ponderată a tuturor obiectivelor de minimizat. Noua funcție obiectiv este $u(\mathbf{F}(\mathbf{x})) = \sum_{k=1}^K t_k f_k(\mathbf{x})$ unde $x \in M$. Pentru probleme de optimizare în care toate funcțiile de cost f_k sunt convexe se poate arăta că problema minimizării funcției u are o soluție optimală în sens Pareto unică [137]. Problema care apare constă în găsirea unor ponderi potrivite, cunoscând faptul că obiectivele au valori numerice diferite și sensibilități diferite. Ponderarea obiectivelor este de aceea un proces iterativ în care trebuie făcute mai multe optimizări, cu ponderi recalculat.

¹Procesului de stabilire a funcției obiectiv i se mai spune și "alegerea criteriului de decizie" folosit de algoritmul de optimizare propriu-zis.

- **Ponderarea distanțelor**

O a doua posibilitate este de folosi metoda "funcției distanță". Presupunem că f_k^* sunt cerințele de atins (minimele funcțiilor de cost f_k). Funcția obiectiv care se folosește este o distanță (de cele mai multe ori în sensul celor mai mici pătrate). Și aici apare problema stabilirii ponderilor, funcția de minimizat fiind $\|\mathbf{z}(\mathbf{x})\|^2 = \sum_{k=1}^K t_k (f_k^*(\mathbf{x}) - f_k(\mathbf{x}))^2$. Pentru funcții convexe și dacă f_k^* sunt minimele fiecărei funcții obiectiv se poate arăta că $\|\mathbf{z}\|$ are un optim unic în sens Pareto. Dezavantajul folosirii unei astfel de norme euclidiene este sensitivitatea scăzută la reziduuri subunitare. De aceea trebuie folosiți factori de pondere suficient de mari.

- **Reformularea problemei cu restricții**

Problema factorilor de pondere poate fi depășită reformulând problema astfel: numai unul din obiective este minimizat, celelalte fiind considerate restricții suplimentare. Problema astfel reformulată minimizează doar $f_i(\mathbf{x})$ (cu i fixat) cu restricțiile suplimentare $f_k(\mathbf{x}) - r_k \leq 0, (\forall)k = 1, K, k \neq i$. Valoarea r_k reprezintă minimumul cerut pentru funcția de cost k . O astfel de formulare are avantajul că ei i se poate aplica o tehnică de tip Lagrange.

Anexa C

Metode de optimizare pentru probleme cu și fără restricții

C.1 Metode de optimizare deterministe pentru probleme fără restricții

Cunoașterea metodelor de optimizare fără restricții este foarte importantă pentru că de multe ori o problemă de optimizare cu restricții este redusă la o problemă de optimizare fără restricții. Iată o clasificare a metodelor de optimizare pentru probleme fără restricții [25, 56, 131].

- **Optimizare unidimensională**

O metodă *fără calculul derivatei* trebuie să încadreze mai întâi minimumul. O astfel de metodă de încadrare este metoda secțiunii de aur¹ care găsește un triplet de puncte $a < b < c$ astfel încât $f(b)$ este mai mic și decât $f(a)$ și decât $f(c)$. Dacă funcția de optimizat are caracteristici suplimentare (de exemplu derivata de ordin doi este continuă, atunci încadrarea obținută din metoda secțiunii de aur se poate rafina cu ajutorul unei interpolări parabolice. O astfel de metodă este **metoda Brent**.

Pentru minimizări unidimensionale *cu calculul derivatei* se poate folosi de exemplu o variantă a metodei Brent care utilizează și informații despre prima derivată.

- **Optimizarea multidimensională**

În acest caz trebuie ales între metode care au un necesar de memorie de ordinul N^2 (N reprezintă numărul de dimensiuni) sau de ordinul N . Pentru valori mici ale lui N această cerință nu este o restricție.

¹Golden Section

Dintre metodele *fără calculul derivatei* una din metode este **metoda simplexului descendent**², datorată lui Nelder și Mead. Ea nu trebuie confundată cu metoda simplex din programarea liniară. Această metodă este foarte încet convergentă, dar este foarte robustă și nu face presupuneri speciale despre funcție. Un simplex în N dimensiuni este un poligon convex având $N + 1$ vârfuri (triunghi pentru $N = 2$, tetraedru pentru $N = 3$). În cazul bidimensional, vârful unui triunghi caracterizat de valoarea cea mai mare a funcției obiectiv este reflectat în raport cu linia care trece prin celelalte două vârfuri. Valoarea funcției în acest punct nou este comparată cu valorile rămase în celelalte două puncte. În funcție de acest test noul punct poate fi acceptat sau rejectat. Când nu se mai realizează nici o îmbunătățire laturile triunghiului sunt contractate și procedura se repetă. În acest fel este posibil ca vârfurile triunghiului să convergeze către un punct în care funcția de minimizat este minimă. Necesarul de memorie este N^2 .

O altă metodă este metoda alegerii direcțiilor, **metoda Powell** fiind prototipul acestei categorii. Această metodă cere o minimizare unidimensională (de exemplu metoda Brent), iar necesarul de memorie este N^2 .

În ce privește metodele *cu calculul derivatei*, există două familii mari de algoritmi pentru minimizarea multidimensională, care necesită calculul de derivate. Ambele familii cer un subalgoritm de minimizare unidimensional care poate sau nu să folosească informații legate de derivată. Prima familie (de ordinul unu) cuprinde **metoda pașilor descendenți** și metode de **gradienți conjugați**. Metoda pașilor descendenți este una din cele mai populare și mai vechi metode folosite în optimizare. Soluția optimă este obținută printr-un număr de iterații consecutive, în fiecare din ele o soluție nouă fiind obținută printr-o deplasare plecând din punctul vechi pe o anumită direcție. Direcția este opusă direcției gradientului iar lungimea (pasul) este o mărime constantă sau nu. Există mai multe variante de algoritmi pentru metoda gradientilor conjugați, de exemplu algoritmul Fletcher-Reeves sau algoritmul Polak-Ribiere (mai bun). Necesarul de memorie este de ordinul N .

A doua familie (de ordinul doi) cuprinde **metoda Gauss-Newton** și **metode quasi-Newton**, de exemplu algoritmul Davidon-Fletcher-Powell, sau algoritmul Broyden-Fletcher-Goldfarb-Shanno. Într-o metodă quasi-Newton direcția de căutare $\Delta \mathbf{x}$ la iterația q este definită ca $\Delta \mathbf{x} = -\mathbf{H}^{-1} \text{grad } f(\mathbf{x})$ unde \mathbf{H} este o aproximare a matricei Hessian a funcției obiectiv F . La primul pas \mathbf{H} este luată matricea unitate (primul pas este deci identic cu cel al metodei pașilor descendenți); apoi \mathbf{H} este recalculat succesiv în concordanță cu o procedură de variabilă metrică. Necesarul de memorie este de N^2 . În metoda Gauss-Newton se folosește chiar matricea Hessian.

Pe scurt, o minimizare multidimensională se reduce la un șir de iterații de tipul: $\mathbf{x}^{k+1} =$

²Downhill simplex method

Metodă	\mathbf{d}^k	Observații
Pașilor descendenți	$-\nabla^k f$	
Gradientilor conjugați	$-\nabla^k f + \mathbf{d}^{k-1} \frac{(\nabla^k f)^T \nabla^k f}{(\nabla^{k-1} f)^T \nabla^{k-1} f}$	
Gauss-Newton	$-(\mathbf{G}^k)^{-1} \nabla^k f$	\mathbf{G} este matricea Hessian: $\mathbf{G} = 2\mathbf{J}^T \mathbf{J}$ unde \mathbf{J} este matricea Jacobi
quasi-Newton	$-(\mathbf{H}^k)^{-1} \nabla^k f$	\mathbf{H} este o aproximare a matricei Hessian

Tabelul C.1: Direcții de căutare într-o metodă de optimizare de ordin superior

$\mathbf{x}^k + \alpha^k \mathbf{d}^k$ unde \mathbf{d}^k este direcția de căutare la iterația k iar α^k este un coeficient determinat printr-o metodă de minimizare unidimensională. Metodele diferă prin alegerea lui \mathbf{d}^k . Tabelul C.1 cuprinde valorile acestor direcții pentru metodele de ordin superior:

C.2 Tratarea restricțiilor

În ceea ce privește problemele cu restricții, metoda simplex a fost prima dezvoltată special pentru o astfel de problemă [131]. Această metodă se bazează pe faptul că o funcție obiectiv liniară își atinge extremele pe frontieră și atunci căutarea lor se face numai pe frontieră. Mai există o clasă de metode numite ”metode de punct interior” [110], mai rar folosite decât metoda simplex.

Aceste două metode rezolvă însă doar problema programării liniare adică aceea în care atât funcția de minimizat cât și restricțiile sunt liniare, situație extrem de rar întâlnită în optimizarea dispozitivelor electromagnetice. De aceea o problemă foarte importantă o constituie tratarea restricțiilor. Există mai multe posibilități de a elimina restricțiile.

- **Folosirea funcției Lagrange** [25, 56, 110]

Această abordare nu folosește funcția propriu-zisă f ci o nouă funcție numită funcție Lagrange, fiind deci o metodă de transformare.

Pentru a putea folosi o scriere mai concisă vom nota cu $f : \mathbb{R}^n \rightarrow \mathbb{R}$ funcția obiectiv (reală), cu $\mathbf{x} \in \mathbb{R}^n$ vectorul variabilelor, restricțiile sunt reprezentate de funcția $c : \mathbb{R}^n \rightarrow \mathbb{R}^p$ unde unele componente ale lui c sunt restricții de tip inegalitate, de forma $c_i(\mathbf{x}) \leq 0$ pentru i într-o mulțime de indici notată \mathcal{I} , alte componente ale lui c sunt restricții de tip egalitate de tip $c_i(\mathbf{x}) = 0$ pentru indici i dintr-o mulțime de indici notată \mathcal{E} . Problema se

poate formula astfel

$$\min \{f(\mathbf{x}) | c_i(\mathbf{x}) \leq 0, i \in \mathcal{I}, c_i(\mathbf{x}) = 0, i \in \mathcal{E}\}, \quad (\text{C.1})$$

unde $c_i : \mathbb{R}^n \rightarrow \mathbb{R}$, \mathcal{I} și \mathcal{E} sunt mulțimile de indici corespunzătoare inegalităților și egalităților ³.

Pentru formularea C.1 funcția Lagrange se definește ca

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) + \sum_{i \in \mathcal{I} \cup \mathcal{E}} \lambda_i c_i(\mathbf{x}), \quad (\text{C.2})$$

unde λ_i se numesc multiplicatori Lagrange. Funcția Lagrange este folosită pentru exprimarea condițiilor de existență a unui minim local.

Condiția necesară de ordinul unu (condiție de existență) a unui minim local \mathbf{x}^* a problemei de optimizare cu restricții C.1 cere existența multiplicatorilor Lagrange λ_i^* astfel încât

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \lambda^*) = \nabla f(\mathbf{x}^*) + \sum_{i \in \mathcal{A}^*} \lambda_i^* \nabla c_i(\mathbf{x}^*) = 0, \quad (\text{C.3})$$

unde $\mathcal{A}^* = \{i \in \mathcal{I} | c_i(\mathbf{x}^*) = 0\} \cup \mathcal{E}$ este mulțimea restricțiilor "active" la \mathbf{x}^* și $\lambda_i^* \geq 0$ dacă $i \in \mathcal{A}^* \cap \mathcal{I}$.

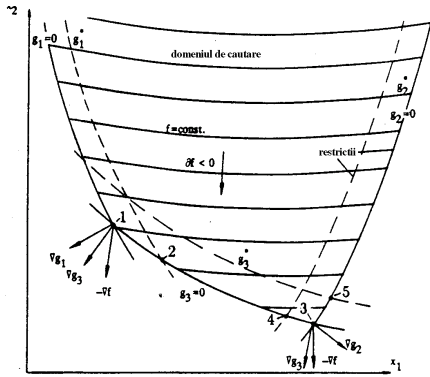


Figura C.1: Interpretarea geometrică a multiplicatorilor Lagrange

este punctul 5). În punctul optim (punctul 3) $-\nabla f = \lambda_2 \nabla g_2 + \lambda_3 \nabla g_3$ unde $\lambda_2 > \lambda_3$

Condiția de ordinul doi (existență și stabilitate) - cere ca $(\mathbf{x}^*, \lambda^*)$ să satisfacă condiția de ordinul unu $\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \lambda^*) = 0$ și, în plus, Hessianul funcției Lagrange

$$\nabla_{\mathbf{xx}}^2 \mathcal{L}(\mathbf{x}^*, \lambda^*) = \nabla^2 f(\mathbf{x}^*) + \sum_{i \in \mathcal{A}^*} \lambda_i^* \nabla^2 c_i(\mathbf{x}^*),$$

să satisfacă $\mathbf{w}^T \nabla_{\mathbf{xx}}^2 \mathcal{L}(\mathbf{x}^*, \lambda^*) \mathbf{w} > 0$ pentru vectorii nenuli \mathbf{w} din mulțimea

$$\{\mathbf{w} \in \mathbb{R}^n | \nabla c_i(\mathbf{x}^*)^T \mathbf{w} = 0, i \in \mathcal{I}_+^* \cup \mathcal{E}, \nabla c_i(\mathbf{x}^*)^T \mathbf{w} \leq 0, i \in \mathcal{I}_0^*\},$$

³Inegalitățile de domeniu sunt incluse în $c_i(\mathbf{x}) \leq 0, i \in \mathcal{I}_i$.

Multiplicatorii Lagrange sunt o măsură a modificării optimului în cazul în care restricțiile se modifică [136]. Figura C.1 reprezintă o interpretare geometrică a acestei afirmații. Funcția de minimizat f are trei restricții de tip inegalități g_1, g_2, g_3 , domeniul de căutare determinat de ele fiind "hașurat" cu linii de echivalență ale lui f . Se observă că minimul se atinge în punctul 3 (în care restricțiile active sunt g_2 și g_3). Mutarea restricției g_2 în g_2^* în domeniul admisibil nu modifică mai puțin valoarea minimului (noul minim este punctul 4), decât cazul în care se mută restricția g_3 în g_3^* (minimul în acest caz este punctul 5).

unde

$$\mathcal{I}_+^* = \{i \in \mathcal{A}^* \cap \mathcal{I} | \lambda_i^* > 0\}, \quad \mathcal{I}_0^* = \{i \in \mathcal{A}^* \cap \mathcal{I} | \lambda_i^* = 0\}.$$

Această condiție garantează că problema de optimizare are o comportare stabilă în jurul extremului.

În funcție de cum este folosită mai departe această funcție Lagrange (cum sunt calculați multiplicatorii Lagrange și cum este minimizată funcția Lagrange), metodele sunt: metode de gradient redus, metode de programare pătratică și liniară, metode bazate pe funcții Lagrange modificate⁴ și funcții de penalizare. Descrierea detaliată a metodelor se găsește în [110].

O altă metodă care folosește funcția Lagrange este metoda asimptotelor⁵ [139]. În această metodă ideea de bază este de a se înlocui problema inițială cu o secvență de subprobleme convexe care au o formă simplă. Pentru fiecare subproblemă se definește Lagrangeanul $L(\mathbf{x}, \lambda) = F(\mathbf{x}) + \sum \lambda_i F_i(\mathbf{x})$ unde F și F_i sunt aproximații convexe ale funcției obiectiv și restricțiilor, suma se face după indicii i aparținând restricțiilor active, și se rezolvă problema duală $\max_{\lambda \geq 0} \{ \min_{\mathbf{x}} L(\mathbf{x}, \lambda) \}$. Aceeași metodă este descrisă și folosită în [138].

• **Minimizări secvențiale pentru funcții obiectiv care includ restricțiile penalizate**⁶ [56, 139]

– *Eliminarea restricțiilor de tip egalitate după metoda lui Courant*

Există situații când tehnica Lagrange este dificil de aplicat. În aceste situații, în practică sunt preferate alte metode. Una din cele mai cunoscute tehnici alternative de tratare a restricțiilor a fost inițiată de R.Courant [56]. În această metodă se minimizează funcția

$$F(x_1, \dots, x_n) = f(x_1, \dots, x_n) + r \sum_{i=1}^p [h(x_1, \dots, x_n)]^2 \quad r > 0 \quad (\text{C.4})$$

ca o problemă fără restricții, pentru o secvență de valori crescătoare ale lui r . Funcția f este astfel ”penalizată” atunci când restricțiile nu sunt satisfăcute. Când r tinde către infinit, suma pătratelor restricțiilor este forțată să tindă spre zero. În acest fel, șirul minimelor succesive ale funcției F tind către soluția problemei inițiale.

– *Eliminarea restricțiilor de tip inegalitate*

Să presupunem că problema are m restricții neliniare de tip inegalitate

$$g_i(x_1, \dots, x_n) \geq 0 \quad i = 1, \dots, m. \quad (\text{C.5})$$

⁴Se întâlnește în literatură cu prescurtarea ALM - ”Augmented Lagrange Method”.

⁵”Method of Moving Asymptotes”

⁶”Penalty Methods”

Există două abordări posibile: una se numește "the boundary-following approaches" (abordări care urmăresc frontiera) și cealaltă este "penalty-function techniques" (tehnici bazate pe funcții de penalizare).

După cum le sugerează numele, abordările care urmăresc frontiera sugerează ca atunci când o restricție este (sau aproape este) violată, se urmărește frontiera domeniului variabilelor, corespunzătoare acelei restricții, până se găsește un punct satisfăcător. Dacă frontiera este puternic neliniară, convergența unor astfel de abordări este înceată. În aceste cazuri se folosește tehnica funcțiilor de penalizare. Să considerăm de exemplu funcțiile:

$$\begin{aligned} F_1(x_1, \dots, x_n) &= f(x_1, \dots, x_n) + r \sum_{i=1}^m \frac{1}{g_i(x_1, \dots, x_n)} \quad r > 0, r \rightarrow 0, \\ F_2(x_1, \dots, x_n) &= f(x_1, \dots, x_n) - r \sum_{i=1}^m \ln[g_i(x_1, \dots, x_n)] \quad r \rightarrow 0, \\ F_3(x_1, \dots, x_n) &= f(x_1, \dots, x_n) + r \sum_{i=1}^m \min^2(0, g_i(x_1, \dots, x_n)) \quad r \rightarrow \infty. \end{aligned}$$

Dacă aceste funcții sunt minimizate secvențial pentru un șir de valori pozitive ale lui r (monoton descrescător pentru F_1 și F_2 și monoton crescător pentru F_3), atunci șirul minimelor funcțiilor cu penalizare (fără restricții) tinde către minimul funcției originale.

O altă metodă de eliminare a restricțiilor de tip inegalitate $g_i(\mathbf{x}) \leq 0$ este următoarea [139]. Se folosește o funcție

$$W(\mathbf{x}, r) = f(\mathbf{x}) + r \sum_{i=1}^m F_i(g_i(\mathbf{x})),$$

unde r este un parametru de penalizare iar funcția F_i asociată restricției g_i este definită astfel

$$F_i(g_i(\mathbf{x})) = \begin{cases} -\frac{1}{g_i(\mathbf{x})} & \text{pentru } g_i(\mathbf{x}) \leq \varepsilon \\ \frac{1}{\varepsilon} \left[3\frac{g_i(\mathbf{x})}{\varepsilon} - \left(\frac{f_i(\mathbf{x})}{\varepsilon} \right)^2 - 3 \right] & \text{pentru } g_i(\mathbf{x}) > \varepsilon \end{cases},$$

unde ε este un parametru. Soluția problemei de optimizare este obținută prin minimizări succesive ale lui W pentru o secvență de valori descrescătoare a lui r . Aceeași metodă este folosită și în [69, 80].

Anexa D

Programe bazate pe evoluție

În ultimii treizeci de ani interesul pentru rezolvarea problemelor prin metode care se bazează pe principii de evoluție și ereditate a crescut foarte mult. Se vehiculează concepte ca: ¹

- Strategii de evoluție (*Evolution Strategies*) - algoritmi care imită principiile evoluției naturale în problemele de optimizare a parametrilor (Rechenberg 1973, Schwefel 1981);
- Programare evoluționistă (*Evolutionary Programming* - Fogel 1966);
- Căutare împrăștiată (*Scatter Search* - Glover 1977);
- Algoritmi genetici (*Genetic Algorithms - GAs*. Holland 1975);
- Programare genetică (*Genetic Programming*) - urmărește căutare a celui mai potrivit program pentru calculator care să rezolve o anumită problemă. (Koza 1990).

Pentru toate sistemele (inclusiv cele de mai sus) care se bazează pe evoluție se folosește un termen comun ”programe de evoluție” (*Evolution Programs - EP*).

Toate aceste abordări au comun faptul că sunt algoritmi cu caracter statistic, inspirați din paradigme biologice. Scopul lor este de a obține optime globale și de a le evita pe cele locale, folosind un timp de calcul cât mai redus. Cu toate că nu există teoreme matematice care să demonstreze indubitabil convergența acestor algoritmi, ei sunt tot mai răspândiți deoarece dau soluții satisfăcătoare din punct de vedere practic.

D.1 Structura unui program de evoluție

Un program de evoluție este un algoritm probabilistic care menține o **populație** $P(t)$.

$P(t)$ - populația de la iterația t este o mulțime de **indivizi** x_k^t :

$$P(t) = \{x_1^t, x_2^t, \dots, x_n^t\} \tag{D.1}$$

¹Această anexă cuprinde ideile principale ale cărții lui Michalewicz [104]

Un individ este o soluție potențială a problemei. El este implementat ca o structură de date \mathcal{S} . Un individ este evaluat și i se atribuie un **grad de potrivire** (*fitness measure*). Unui individ i se mai spune și genotip, cromozom. Ideea comună o oricărui EP este aceea că o populație de indivizi suferă transformări și în timpul acestui proces indivizii luptă pentru supraviețuire.

La fiecare iterație nouă ($t + 1$) se formează o populație nouă prin **selecția** indivizilor cu un grad de potrivire mai bun.

Unii membri suferă transformări, se **alterează**. Modificările se fac cu ajutorul **operatorilor genetici**. Aceștia pot fi de două tipuri:

- unari, de tip **mutație**: un astfel de operator m_i crează un individ nou printr-o schimbare a unui individ vechi:

$$m_i : \mathcal{S} \rightarrow \mathcal{S};$$

- binari, de tip **încrucșare** (*crossover*): un astfel de operator c_j crează un individ prin combinarea a doi indivizi vechi:

$$c_j : \mathcal{S} \times \mathcal{S} \rightarrow \mathcal{S}.$$

După un anumit număr de iterații (sau **generații**) programul converge (se speră): cel mai bun individ este o soluție apropiată de optim.

Iată **algoritmul unui program de evoluție (EP)**:

1. $t = 0$
2. inițializează $P(t)$
3. evaluează $P(t)$
4. **cât timp** (nu condiție de stop) **repetă**
 - 4.1. $t = t + 1$
 - 4.2. selectează $P(t)$ din $P(t - 1)$
 - 4.3. alterează $P(t)$
 - 4.4. evaluează $P(t)$

Pentru o anumită problemă pot fi formulate mai multe programe de evoluție. Ele pot diferi din mai multe puncte de vedere:

- structura de date pentru implementarea unui individ (\mathcal{S});
- operatorii genetici;
- metoda de creare a populației inițiale;
- metoda de tratare a restricțiilor;
- parametri: dimensiunea populației, probabilitatea aplicării diferitor operatori.

Structura unui algoritm genetic (GA) este aceeași ca cea prezentată pentru EP. Diferențele sunt la nivele mai joase. GAs clasici folosesc drept structură de date pentru reprezentarea unui individ un șir binar de lungime fixă și doi operatori: mutația binară și încrucișarea binară. În EP cromozomii pot fi și altceva decât structuri binare de lungime fixă, iar procesul de alterare poate include și alți operatori genetici.

Un algoritm genetic (ca orice program de evoluție), pentru o problemă particulară are următoarele cinci componente:

1. reprezentarea genetică pentru soluția potențială a problemei;
2. o modalitate de a crea o populație inițială a soluției potențiale;
3. o funcție de evaluare (cost) care joacă rolul mediului;
4. operatori genetici care alterează compoziția populației;
5. valori pentru parametri: dimensiunea populației, probabilitatea de a aplica operatorii genetici.

D.2 Baza teoretică a algoritmilor genetici

Fundamentul teoretic al GAs se bazează pe o reprezentare tip șir binar a soluției și pe noțiunea de **schemă**. O schemă S este construită prin introducerea unui nou simbol în alfabetul cromozomilor. Acest simbol este "*" (oricare). O schemă reprezintă toate șirurile care se potrivesc cu ea în toate pozițiile care nu sunt "*".

Să notăm cu m numărul de biți ai unei scheme și cu r numărul de simboluri "*". O schemă are două proprietăți importante:

- **ordinul schemei** este numărul de poziții fixe

$$o(S) = m - r. \quad (\text{D.2})$$

Ordinul folosește în calcularea probabilității de supraviețuire a schemei la mutații.

- **lungimea schemei** este distanța dintre prima și ultima poziție fixată. Se notează cu $\delta(S)$. Ea definește compactitatea informației conținută în schemă.

O altă proprietate este **gradul de potrivire al schemei S la momentul t** . Se notează cu $eval(S, t)$ și reprezintă media gradurilor de potrivire a indivizilor (șirurilor) din populația t care se potrivesc schemei S .

Se demonstrează următoarea teoremă:

Teorema schemei: Schemelor de lungime mică, de ordin mic și care au un grad de potrivire mai mare decât valoarea medie a gradurilor de potrivire a indivizilor, le vor corespunde, în generațiile care urmează, un număr tot mai mare (creșterea este exponențială) de indivizi din populație.

Un rezultat imediat al acestei teoreme este acela că un algoritm genetic trebuie să exploreze spațiul de căutare prin intermediul schemelor de lungime mică și ordin mic.

D.3 Convergența algoritmilor genetici

Teoria algoritmilor genetici dă explicații de ce pentru o problemă dată putem obține convergența către punctul optim căutat. Din nefericire, aplicațiile practice nu urmăresc întotdeauna teoria, principalele motive fiind:

- codul problemei face adesea ca algoritmul genetic să opereze într-un alt spațiu decât problema în sine;
- există o limită a numărului (ipotetic infinit) de iterații;
- există o limită a numărului (ipotetic infinit) a dimensiunii populației.

Din aceste motive, algoritmii genetici nu găsesc întotdeauna soluția optimă, ele convergând prematur către un optim local.

Cele mai multe cercetări se referă la ordinul de mărime și tipul erorii introduse de procesul de selecție și de asemenea la studiul funcției obiectiv. Au apărut de asemenea algoritmi genetici contractivi și algoritmi genetici având dimensiunea populației variabilă de la o generație la alta.

D.3.1 Importanța mecanismului de selecție

Există două probleme importante în procesul de evoluție al căutării genetice: **diversitatea populației și presiunea de selecție**.

Acești doi factori sunt strâns legați: o creștere a presiunii de selecție duce la scăderea diversității populației și viceversa. Cu alte cuvinte o presiune de selecție puternică este cauza convergenței premature a unei căutări GA. O presiune slabă duce la o căutare inefficientă. De aceea este necesară o balanță între acești doi factori, mecanismele de selecție încercând să atingă acest scop.

Dintre autorii care au propus modele ale mecanismelor de selecție, DeJong este primul (1975) și cel mai recunoscut. El a propus următoarele modele (în 1975):

- **modelul elitist** - forțează păstrarea celui mai bun cromozom;
- **modelul valorii așteptate** - pentru fiecare cromozom este introdus un contor care este descrescut cu 0.5 atunci când cromozomul este selectat pentru încrucișare și cu 1 atunci când el este selectat pentru mutație. Când valoarea acestui contor devine negativă, cromozomul nu mai este de loc disponibil pentru selecție.
- **modelul elitist și al valorii așteptate** - o combinație de cele două;

- **modelul factorului de ”înghesuială”** - un cromozom nou înlocuiește unul vechi, cel sortit pieirii fiind selectat dintre cei care seamănă cu cel nou.

In 1981 Brindel mai propune:

- **eșantionare stocastică în funcție de rest, cu sau fără înlocuire;**

Dintre acestea, modelul eșantionării stocastice în funcție de rest, cu înlocuire, în care un cromozom este selectat în concordanța cu partea întreagă a valorii ce reprezintă numărul de apariții al cromozomului într-o nouă populație și unde cromozomii ”concurează” în concordanța cu partea fracțională a restului de locuri din populație, a fost cea mai de succes, și adoptată de mulți cercetători drept standard.

Au mai apărut și alte variante:

- metode bazate pe introducerea unor ponderi artificiale: cromozomii sunt selectați mai degrabă proporțional cu poziția lor în populație decât în concordanță cu valorile rezultate din evaluare.

Aceste metode se bazează pe convingerea că motivul convergenței premature este prezența superindivizilor, care sunt mult mai buni decât media populației. Astfel de superindivizi au un număr mare de copii și (datorită dimensiunii constante a problemei) nu lasă alți indivizi să contribuie cu copii la noua generație. În câteva generații, un superindiviz poate elimina un material cromozomial bun, cauzând o convergență rapidă către optime posibil locale.

- **selecție tip turneu** - într-o iterație sunt selectați un număr de indivizi și cel mai bun este ales pentru generația următoare, procedeul repetându-se de un număr de ori egal cu dimensiunea populației

Procedurile de selecție s-ar putea clasifica astfel (după Back și Hoffmeister):

1. metode **dinamice** și metode **statice** - într-o selecție statică probabilitățile de selecție rămân constante între generații;
2. metode **extinctive** și metode **prezervative** - într-o metodă prezervativă probabilitatea de selecție este diferită de zero pentru orice individ, lucru care nu este adevărat pentru o metodă extintivă;
3. metode de **selecție pură** - părinții se reproduc într-o singură generație, timpul de viață al unui individ fiind de o generație indiferent de cât de bun ar fi el;
4. **selecții generaționale** - mulțimea de părinți este fixă până când sunt produși toți copiii din generația următoare;
5. **selecții ”din zbor”** - un copil își înlocuiește părintele imediat;
6. **selecții elitiste** - unii sau toți părinții pot suferi selecții împreună cu copiii lor.

D.3.2 Influența funcției obiectiv

Pentru îmbunătățirea convergenței algoritmului GA, ar putea fi de ajutor unele măsuri legate de caracteristicile funcției de optimizat. S-ar putea încerca ”repararea” funcției introducând un mecanism de scalare. După Goldberg aceste mecanisme se împart în trei categorii:

- scalare liniară

$$f' = af + b; \quad (D.3)$$

- trunchiere sigma

$$: f' = f + (\bar{f} - c\sigma), \quad (D.4)$$

unde σ este deviația standard a populației;

- scalare după o dependentă tip putere:

$$f' = f^k, \quad (D.5)$$

unde k este apropiat de 1 și depinde de problemă.

Problema care apare este următoarea: fie $f_1(x)$ o funcție de maximizat și $f_2(x) = f_1(x) + k$ unde k este o constantă (f_2 are același optim global ca și f_1). Dacă $k \gg \bar{f}_1$ atunci f_2 va avea o convergența mult mai înceată decât f_1 . În cazul extrem funcția a doua va fi optimizată folosind o căutare aleatoare totală; o astfel de comportare poate fi tolerată la începutul algoritmului dar ar fi devastatoare apoi. Dimpotrivă, f_1 ar putea converge prea repede, împingând algoritmul într-un minim local.

D.3.3 Condiția de oprire a algoritmului

Condiția de oprire este deosebit de importantă. Cea mai simplă ar verifica **numărul generației curente**, căutarea fiind terminată atunci când acest număr este mai mare decât ceva impus.

În multe versiuni de EP nu toți indivizii trebuie reevaluați; unii dintre ei trec de la o generație la alta fără alterare. În aceste cazuri ar putea avea semnificație (pentru comparare cu alți algoritmi) **numărul de evaluări ale funcției obiectiv**, algoritmul oprindu-se atunci când acest număr este mai mare decât ceva impus.

Cele două tipuri de condiții de mai sus presupun cunoscute caracteristicile funcției, acestea influențând durata de căutare. O condiție mai bună de oprire ar fi una care să exprime faptul că **șansa pentru o îmbunătățire semnificativă este redusă**. Există două categorii de bază drept condiții de oprire pentru algoritm. Una din acestea se bazează pe structura cromozomilor iar cealaltă pe semnificația unui anumit cromozom.

D.3.4 Algoritmi genetici contractivi

Demonstrarea convergenței GAs este una din problemele cele mai provocatoare din domeniul calculelor bazate pe evoluție. Mulți cercetători au abordat această problemă din perspective diferite. Goldberg și Segrest au făcut o analiză bazată pe lanțuri Markov finite. David și Principe au investigat posibilitatea extrapolării teoriei existente de la algoritmul *simulated annealing*(SA) într-un model de GA bazat pe lanț Markov. Eiben, Aarts și Van Hee au propus un GA abstract care unifica GA și SA. Recent, Rudolph a demonstrat că un GA clasic nu converge niciodată într-un optim global, dar versiuni modificate, care mențin cea mai bună soluție în populație (modele elitiste) converg.

O abordare posibilă se poate face cu ajutorul teoremei de punct fix a lui Banach. GAs reprezintă transformări între populații. Dacă se găsește un spațiu metric în care aceste transformări sunt contracții, atunci se poate găsi un șir de populații succesive, convergente către punctul fix, unic al contracției. Un astfel de spațiu metric complet se poate găsi, **algoritmul genetic contractiv** fiind următorul (pentru algoritmul de mai jos optimizare înseamnă maximizare):

1. $t = 0$
2. inițializează $P(t)$
3. evaluează $P(t)$
4. **cât timp** (nu condiție de stop) **repetă**
 - 4.1. $t = t + 1$
 - 4.2. selectează $P(t)$ din $P(t - 1)$
 - 4.3. recombina $P(t)$
 - 4.4. evaluează $P(t)$
 - 4.5. **dacă** $Eval(P(t - 1)) \geq Eval(P(t))$ **atunci** $t = t - 1$

În algoritmul de mai sus

$$Eval(P) = \frac{1}{n} \sum_{x_i \in P} eval(x_i). \quad (D.6)$$

La o primă privire s-ar părea că doar alegerea populației inițiale ar putea influența viteza de convergență. Acest algoritm ar converge către optimul global (in timp infinit). Totuși e posibil ca la un moment dat să nu fie acceptată o nouă populație după multe încercări. Cu alte cuvinte operatorii de mutație și încrucișare aplicați unei populații suboptimale particulare sunt incapabili să producă o populație mai bună.

D.3.5 GAs cu dimensiunea populației variabilă

Alegerea dimensiunii populației este una din cele mai importante decizii pe care trebuie să o ia un utilizator de GAs și poate fi critică în multe aplicații. Dacă acest număr este prea mic GA converge prea repede, iar un număr prea mare poate însemna irosirea resurselor de calcul, timpul de așteptare pentru o îmbunătățire putând fi prea mare.

O abordare mai naturală este aceea în care parametrul dimensiunea populației este variabil. Într-un astfel de algoritm nu există mecanism de selecție. Se introduce noțiunea de **durată de viață a cromozomului**, aceasta reprezentând numărul de generații în care cromozomul rămâne viu. Durata de viață a unui cromozom este atribuită la "nașterea" lui. Fiecare cromozom este ales să reproducă cu o probabilitate egală, indiferent de gradul lui de potrivire. Moartea unui cromozom survine atunci când vârsta lui devine mai mare decât durata lui de viață. Iată cum ar arăta un astfel de algoritm:

1. $t = 0$
2. inițializează $P(t)$
3. evaluează $P(t)$
4. **cât timp** (nu condiție de stop) **repetă**
 - 4.1. $t = t + 1$
 - 4.2. crește vârsta fiecărui individ cu 1
 - 4.3. recombina $P(t)$
 - 4.4. evaluează $P(t)$
 - 4.5. elimină din $P(t)$ toți indivizii cu vârsta expirată

Există mai multe strategii posibile de atribuire a duratei de viață. O valoare constantă, independentă de orice statistică a căutării ar genera o creștere exponențială a dimensiunii populației. Mai mult, deoarece nu există mecanism de selecție rezultă că nu există presiune de selecție, iar aceasta înseamnă performanțe slabe ale algoritmului. Pentru a introduce o presiune de selecție durata de viață trebuie calculată mai sofisticat. Strategiile de calcul trebuie să urmărească următoarele:

- să dea o durată de viață mai mare indivizilor cu grad de potrivire mai mare decât media, respectiv mai mică celor cu un grad de potrivire mai mic decât media.
- să regleze dimensiunea populației la o iterație astfel încât să prevină creșterea exponențială și să reducă costurile de simulare.

D.4 Tratarea restricțiilor

Tehnicile pentru tratarea restricțiilor în algoritmi genetici pot fi grupate în câteva categorii.

O modalitate de a trata candidații care violează restricțiile este de a genera soluții potențiale fără a considera restricțiile și de a le **penaliza** descrescând gradul lor de potrivire. Cu alte cuvinte o problemă cu restricții este transformată într-una fără restricții, asociind o penalizare violării restricțiilor. Aceste penalizări sunt incluse în evaluarea funcției. Penalizarea poate fi constantă, sau poate depinde de gradul de violare a restricției, sau poate fi chiar pedeapsa ”cu moartea” adică eliminarea soluției din populație. Această din urmă are unele neajunsuri: pentru anumite probleme, probabilitatea de a genera (prin intermediul operatorilor genetici standard) o soluție care să nu violeze restricțiile este relativ mică și algoritmul pierde mult timp pentru a evalua indivizi ”ilegali”.

O altă categorie de metode de tratare a restricțiilor se bazează pe **aplicarea unor algoritmi de reparare** care corectează soluțiile care violează restricțiile. Astfel de algoritmi sunt consumatori de timp și depind de aplicație. Mai mult, procesul de corecție poate fi la fel de dificil ca rezolvarea problemei generale.

O a treia abordare se concentrează pe folosirea unor funcții speciale care garantează (sau cel puțin cresc probabilitatea) generarea unor soluții care nu încalcă restricțiile, sau utilizarea unor operatori specifici problemei în același scop. Și astfel de funcții sunt costisitoare și depind de aplicație.

Tehnicile diferă în cazul restricțiilor liniare față de cazul restricțiilor neliniare.

D.5 Reprezentarea genetică a soluției problemei

Codificarea binară facilitează analiza teoretică și permite utilizarea de operatori genetici eleganți, dar ea are anumite neajunsuri atunci când este aplicată în cazul multidimensional. De exemplu, pentru 100 variabile, fiecare în domeniul $[-500, 500]$, unde se cere o precizie de 6 cifre semnificative, lungimea vectorului soluție binar este 3000. Aceasta generează un spațiu de căutare de aproximativ 10^{1000} . Pentru astfel de probleme, algoritmi genetici au performanțe slabe.

Pentru problemele de optimizări de parametri având variabile peste domenii continue se pot experimenta cromozomi codificați cu ajutorul numerelor reale și operatori genetici speciali. Scopul principal în spatele unei astfel de implementări este de a muta algoritmul genetic mai aproape de spațiul problemei. Într-o astfel de reprezentare două puncte apropiate în spațiul de reprezentare trebuie să fie apropiate în spațiul problemei și reciproc.

D.6 Reglajul fin

Algoritmii genetici au dificultăți inerente în a face o căutare locală. Holland sugerează că algoritmii genetici ar trebui folosiți pentru a face o căutare inițială, după care ar trebui folosiți algoritmi care să poată folosi informații despre domeniu și funcție pentru a face o căutare locală.

Căutarea locală implică folosirea unor scheme de ordin mai mare și de lungimi mai lungi decât cele sugerate de teorema schemei. Mai mult decât atât, există probleme unde numărul de parametri este foarte mare și este necesară o precizie crescută. În cazul unei reprezentări binare, această cerință implică o lungime foarte mare a vectorului soluție. Pentru astfel de probleme performanța algoritmilor genetici este slabă.

Pentru a îmbunătăți posibilitatea de "reglaj fin" al algoritmilor genetici se poate folosi un operator special de mutație. Operatorul mutație tradițional schimbă un bit al unui cromozom la un moment dat. Cu cât este mai semnificativ acest bit (cu cât este mai la stânga), cu atât valoarea absolută a schimbării vectorului datorită mutației este mai mare. Biții din partea dreaptă au o influență mai mică, atunci când ei sunt cei care suferă mutația. De aceea se poate proceda astfel: pe măsură ce populația îmbătrânește, biților localizați spre dreapta li se atribuie o probabilitate din ce în ce mai mare de a fi mutați, iar celor din partea stângă o probabilitate din ce în ce mai mică. O astfel de mutație cauzează o căutare globală la începutul procesului iterativ, și o explorare locală din ce în ce mai mare pe măsura succedării generațiilor.

Listă de figuri

2.1	Relații de bază pentru regimul magnetic staționar	14
2.2	Complexul lui de Rham	15
2.3	Diagrama Tonti primară	16
2.4	Diagrama Tonti duală	16
2.5	Funcția de cost a solenoidului Loney în funcție de S ($L = 5.5$ cm)	26
2.6	Funcția de cost în funcție de numărul de apeluri ale ei	40
3.1	Împărțirea în domenii în cazul 3D	66
3.2	Împărțirea în domenii în cazul 2D	67
3.3	Tratarea integralei improprii	69
3.4	Notății relative la un segment de frontieră	71
3.5	Domeniul Ω_i	73
3.6	Discretizarea inițială și spectrul inducției magnetice	75
3.7	Discretizarea după două rafinări succesive	75
3.8	Discretizarea după patru rafinări succesive	75
3.9	Domeniul Ω_n și curba Γ	76
3.10	Test 1 - rețeaua de discretizare	80
3.11	Test 2 - rețeaua de discretizare	80
3.12	Test 3 - rețeaua de discretizare	81
3.13	Test 4 - rețeaua de discretizare	81
3.14	Curba de magnetizare pentru testele 1,2 și 3	81
3.15	Curba de magnetizare pentru testul 4	81

3.16	Test 1 - spectrul inducției (MEGA)	82
3.17	Test 1 - spectrul inducției (FAP)	82
3.18	Test 2 - spectrul inducției (MEGA)	83
3.19	Test 2 - spectrul inducției (FAP)	83
3.20	Test 3 - spectrul inducției (FAP)	84
3.21	Test 4 - spectrul inducției (FAP)	84
3.22	Test 4 - spectrul inducției în zona de interes ("zoom") obținut cu FAP	84
3.23	Funcția de minimizat și derivata ei în funcție de factorul de relaxare	85
3.24	Test 3 - norma corecțiilor polarizațiilor $\ \delta I\ $ în metoda PB	87
3.25	Test 4 - norma corecțiilor polarizațiilor $\ \delta I\ $ în metoda PB	87
3.26	Test 3 - norma relativă a corecției $\ \delta I\ /\ I\ $ în metoda PB	87
3.27	Test 4 - norma relativă a corecției $\ \delta I\ /\ I\ $ în metoda PB	87
3.28	Test 2 - factorul de relaxare în metoda PB	88
3.29	Test 3 - factorul de relaxare în metoda PB	88
3.30	Test 4 - factorul de relaxare în metoda PB	88
3.31	Test 3 - norma relativă a corecției $\ \delta A\ /\ A\ $ în metoda Newton	88
3.32	Test 1 - norma relativă a corecției $\ \delta A\ /\ A\ $ în metoda Newton	89
3.33	Test 1 - energia în timpul iterațiilor Newton	89
3.34	Test 4 - norma relativă a corecției $\ \delta A\ /\ A\ $ în metoda Newton	90
3.35	Test 4 - energia în timpul iterațiilor Newton	90
3.36	$\nu(B^2)$ pentru metoda Newton, interpolare Akima	91
3.37	$d\nu/dB^2$ pentru Newton, interpolare Akima	91
3.38	$\nu(B^2)$ pentru metoda Newton, interpolare spline	91
3.39	$\nu_{dinamic}$ așa cum rezultă din interpolarea Akima	91
5.1	Vectorul cum_probab pentru POP_SIZE = 20	116
5.2	Vectorul cum_probab pentru POP_SIZE = 70	116
5.3	Harta funcției $C(x, y)$	122
5.4	$C(x, -0.7126)$	123

5.5	$C(0.0898, y)$	123
5.6	Harta funcției $B(x, y)$	123
5.7	$B(x, 1)$	123
5.8	Echivalori de $R(x, y)$	124
5.9	$R(x, 0.2)$	124
5.10	Evoluția celui mai bun individ în funcție de numărul generației, pentru diferite valori ale d	
5.11	Evoluția celui mai bun individ în timp (număr de evaluări), pentru diferite valori ale dime	
5.12	Evoluția parametrului x pentru diferite valori ale dimensiunii populației . .	127
5.13	Evoluția parametrului y pentru diferite valori ale dimensiunii populației . .	127
5.14	Evoluția dispersiei funcției obiectiv pentru diferite valori ale dimensiunii populației	127
5.15	Dispersia parametrului x pentru diferite valori POP_SIZE	128
5.16	Dispersia parametrului y pentru diferite valori POP_SIZE	128
5.17	Cel mai bun individ: referință; fără mutații; fără încrucișări.	129
5.18	Dispersia funcției: referință; fără mutații; fără încrucișări.	129
5.19	Dispersia parametrului x : referință; fără mutații; fără încrucișări.	129
5.20	Dispersia parametrului y : referință; fără mutații; fără încrucișări.	129
5.21	Cel mai bun individ: algoritm de referință; $P_1 = 0; P_2 = 0; P_3 = 0; P_6 = 0$. .	130
5.22	Dispersia funcției obiectiv: algoritm de referință; $P_1 = 0; P_2 = 0; P_3 = 0; P_6 = 0$.	130
5.23	Dispersia parametrului x ; algoritm de referință; $P_1 = 0; P_2 = 0; P_3 = 0; P_6 = 0$.	131
5.24	Dispersia parametrului y ; algoritm de referință; $P_1 = 0; P_2 = 0; P_3 = 0; P_6 = 0$.	131
5.37	Evoluția celui mai bun individ: algoritmul secvențial față de cel distribuit .	131
5.25	Cel mai bun individ: algoritm de referință; $P_4 = 0; P_5 = 0$	132
5.26	Dispersia funcției de cost: algoritm de referință; $P_4 = 0; P_5 = 0$	132
5.27	Dispersia parametrului x ; algoritm de referință; $P_4 = 0; P_5 = 0$	133
5.28	Dispersia parametrului y ; algoritm de referință; $P_4 = 0; P_5 = 0$	133
5.29	Cel mai bun individ: $q = 0.089; q = 0.2; q = 0.05$	134
5.30	Dispersia funcției de cost: $q = 0.089; q = 0.2; q = 0.05$	134
5.31	Dispersia parametrului x : $q = 0.089; q = 0.2; q = 0.05$	135

5.32	Dispersia parametrului y : $q = 0.089$; $q = 0.2$; $q = 0.05$	135
5.33	Cel mai bun individ: $b = 1$; $b = 2$; $b = 4$	136
5.34	Dispersia funcției de cost: $b = 1$; $b = 2$; $b = 4$	136
5.50	Funcții caracteristice de relief, mapate pe intervalul $[0,1]$	136
5.35	Dispersia parametrului x : $b = 1$; $b = 2$; $b = 4$	137
5.36	Dispersia parametrului y : $b = 1$; $b = 2$; $b = 4$	137
5.38	Cel mai bun individ: se înlocuiește cel mai bun; cel mai rău; unul aleator .	138
5.39	Dispersia funcției obiectiv: se înlocuiește cel mai bun; cel mai rău; unul aleator	138
5.40	Dispersia parametrului x : se înlocuiește cel mai bun; cel mai rău; unul aleator	139
5.41	Dispersia parametrului y : se înlocuiește cel mai bun; cel mai rău; unul aleator	139
5.42	Cel mai bun individ: algoritm secvențial; 2, 4, 6, 10 insule	140
5.43	Dispersia funcției obiectiv: algoritm secvențial; 2, 4, 6, 10 insule	140
5.44	Dispersia parametrului x : algoritm secvențial; 2, 4, 6, 10 insule	141
5.45	Dispersia parametrului y : algoritm secvențial; 2, 4, 6, 10 insule	141
5.46	Cel mai bun individ: algoritm secvențial; perioada = 1, 3, 6.	142
5.47	Dispersia funcției obiectiv: algoritm secvențial; perioada = 1, 3, 6.	142
5.48	Dispersia parametrului x : algoritm secvențial; perioada = 1, 3, 6.	143
5.49	Dispersia parametrului y : algoritm secvențial; perioada = 1, 3, 6.	143
6.1	Dispozitiv SMES cu doi solenoizi	146
6.2	Restricția impusă pentru supraconductor	146
6.3	Distribuirea nodurilor: în zona $[0, m_r] \times [0, m_z]$ sunt n^2 noduri.	147
6.4	Linii de câmp în cazul $p = 0$	150
6.5	Linii de câmp în cazul $p = 1$	150
6.6	Evoluția celui mai bun individ în funcție de numărul generației	157
6.7	Evoluția dispersiilor (cazul funcției obiectiv nepenalizate)	157
6.8	Evoluția dispersiei funcției obiectiv (scara liniară pe y)	157
6.9	Evoluția dispersiei funcției obiectiv (scara logaritmică pe y)	157
6.10	Evoluția parametrului R_1	158

6.11	Evoluția dispersiei lui R_1	158
6.12	Evoluția parametrului R_2	158
6.13	Evoluția dispersiei lui R_2	158
6.14	Evoluția parametrului $h_1/2$	158
6.15	Evoluția dispersiei lui $h_1/2$	158
6.16	Evoluția parametrului $h_2/2$	159
6.17	Evoluția dispersiei lui $h_2/2$	159
6.18	Evoluția parametrului d_1	159
6.19	Evoluția dispersiei lui d_1	159
6.20	Evoluția parametrului d_2	159
6.21	Evoluția dispersiei lui d_2	159
6.22	Curba de magnetizare a oțelului	168
6.23	Matriță cu electromagnet	170
6.24	Detaliu în zona de interes	171
6.25	Notății folosite la calculul integralei simple	176
6.26	Notății folosite la calculul integralei duble - cazul în care muchiile au un nod comun	177
6.27	Eroarea în funcție de x	177
6.28	Eroare și majorant, $n = 1$	178
6.29	Eroare și majorant, $n = 2$	178
6.30	Eroare și majorant, $n = 3$	178
6.31	Eroare și majorant, $n = 4$	178
6.32	Eroare și majorant, $n = 5$	179
6.33	Eroare și majorant, $n = 6$	179
6.34	Vectorul nr_m	179
6.35	Linii de câmp obținute cu metoda integrală pentru rețeaua cu 20 elemente	182
6.36	Discretizare de tip celule	182
6.37	Variația energiei și a funcției de cost cu inversul numărului de elemente . .	184
6.38	Variația funcției de cost pentru diferite valori ale parametrului R_1	184

6.39	B_x în funcție de unghiul α	185
6.40	B_y în funcție de unghiul α	185
B.1	Interpretarea geometrică a soluțiilor optimale în sens Pareto	194
C.1	Interpretarea geometrică a multiplicatorilor Lagrange	200

Listă de tabele

2.1	Posibilități de a rezolva un model matematic diferențial	15
2.2	Domeniile operatorilor din complexul Rham	16
2.3	Domeniile operatorilor din complexul dual complexului de Rham	16
3.1	Câteva valori nodale	82
3.2	Energii în model	83
3.3	Timp de calcul - testele 1 și 2	85
3.4	Timp de calcul - testele 3 și 4	86
4.1	$\frac{\partial s_{ij}}{\partial x_k}$ și $\frac{\partial s_{ij}}{\partial y_k}$, unde i, j, k sunt nodurile elementului e	102
5.1	Parametrii folosiți în testele cu diferite valori pentru POP_SIZE	126
6.1	Restricții de domeniu ale variabilelor de optimizare	147
6.2	Rezultate numerice pentru evaluarea funcției de cost a problemei TEAM 22 - metoda $A^* = A$	
6.3	Rezultate numerice pentru evaluarea funcției de cost a problemei TEAM 22 - metoda $A^* = A$	
6.4	Rezultate de referință folosite în evaluarea algoritmului de calcul al funcției de cost pentru	
6.5	Efortul de calcul pentru evaluarea funcției de cost a problemei TEAM 22 - metoda $A^* = A$	
6.6	Efortul de calcul pentru evaluarea funcției de cost a problemei TEAM 22 - metoda $A^* = A$	
6.7	Rezultate numerice ale optimizării problemei TEAM 22 în cazul în care funcția obiectiv este	
6.8	Alte informații despre testele cu penalizare suplimentară	156
6.9	Rezultate numerice ale optimizării problemei TEAM 22 în cazul în care funcția obiectiv nu	
6.10	Alte informații despre testele fără penalizare suplimentară	157
6.11	Notății folosite în formulele sensibilităților problemei TEAM 22	161

6.12	Formule utile pentru derivarea integralelor	162
6.13	Punctele curbei de magnetizare	169
6.14	Restricții de domeniu ale variabilelor de optimizare	169
6.15	Rezultate numerice ale metodei integrale	181
6.16	Teste numerice - problema liniară - diferiți preconditionatori	182
6.17	Teste numerice - problema liniară - diferite rețele	183
6.18	Problema neliniară - iterații Picard-Banach	183
6.19	Problema neliniară - iterații Newton.	184
C.1	Dir ecții de căutare într-o metodă de optimizare de ordin superior	199

Bibliografie

- [1] ***. http://www-igte.tu-graz.ac.at/_team.
- [2] ***. http://www.lmn.pub.ro/~tibi/mesh_gen/mesh_gen.html.
- [3] ***. Linux: <http://www.linux.org>.
- [4] ***. *Scilab Home Page*. Institut National de Recherche en Informatique et en automatique, <http://www-rocq.inria.fr/scilab/>, 2000.
- [5] K. Adamiak and A.J. Czaja. Optimizing Strategy for MR Imaging Gradient Coils. *IEEE Transactions on Magnetism*, 30(6):4311–4313, 1994.
- [6] J.P. Adriaens, F. Delince, P. Dular, A. Genon, W. Legros, and A. Nicolet. Vector Potential Boundary Element Method for Three Dimensional Magnetostatic. *IEEE Transactions on Magnetism*, 27(5):3808–3810, 1991.
- [7] R. Albanese and G. Rubinacci. Numerical Procedures for the Solution of Nonlinear Electromagnetic Problems. *IEEE Transactions on Magnetism*, 28(2):1228–1231, 1992.
- [8] P. Alotto, A. Caiti, G. Molinari, and M. Repetto. A Multiquadrics-based Algorithm for the Acceleration of Simulated Annealing Optimization Procedures. *IEEE Transactions on Magnetism*, 32(3):1198–1201, 1996.
- [9] P. Alotto, A. Kuntsevitch, C. Magele, G. Molinari, C. Paul, K. Preis, M. Repetto, and K. Richter. Multiobjective Optimization in Magnetostatics: A Proposal for Benchmark Problems. *IEEE Transactions on Magnetism*, 32(3):1238–1241, 1996.
- [10] B. Ancelle, P. Rafinejad, and J.C. Sabonnadiere. An Improved Implementation of Boundary Integral Equation Method to Compute Two-Dimensional Magnetic Field. *IEEE Transactions on Magnetism*, 14(5):470–472, 1978.
- [11] M. Ayoub, F. Roy, F. Bouillault, and A. Razek. Numerical Modelling of 3D Magnetostatic Saturated Structures with a Hybrid FEM-BEM Technique. *IEEE Transactions on Magnetism*, 28(2):1052–1055, 1992.

- [12] T. Back, U. Hammel, and H.P. Schwefel. Evolutionary Computation: Comments on the History and Current State. *IEEE Transactions on Evolutionary Computation*, 1(1):3–17, 1997.
- [13] S. Balay, W. Gropp, L.C. McInnes, and B. Smith. *PETSc 2.0 User Manual*. <http://www.mcs.anl.gov/petsc>, 1998.
- [14] B. Bandelier, C. Daveau, and F. Rioux-Damidau. An h-Formulation for the Computation of Magnetostatic Fields. Implementation by Combining a Finite Element Method and a Boundary Element Method. *Journal de Physique III France 3*, pages 995–1004, 1993.
- [15] P. Di Barba, F. Dughiero, and F. Trevisan. Optimization of the Loney’s Solenoid through Quasi-Analytical Strategies: A Benchmark Problem Reconsidered. *IEEE Transactions on Magnetics*, 33(2):1864–1867, 1997.
- [16] P. Di Barba, A. Gottvald, and A. Savini. Global Optimization of Loney’s Solenoid: a Benchmark Problem. *Int. J. of Applied Electromagnetics and Mechanics*, 6(4):273–276, 1995.
- [17] P. Di Barba and A. Savini. Global Optimization of Loney’s Solenoid by Means of a Deterministic Approach. *Int. J. of Applied Electromagnetics and Mechanics*, 6(4):247–254, 1995.
- [18] F. Bellina, P. Campostrini, G. Chitarin, A. Stella, and F. Trevisan. Automated Optimal Design Techniques for Inverse Electromagnetic Problems. *IEEE Transactions on Magnetics*, 28(2):1549–1552, 1992.
- [19] J.M. Biedinger, S. Poullain, and J.P. Yvon. Optimal Control of the Torque of Solid Iron Rotor Induction Motor. *IEEE Transactions on Magnetics*, 28(2):1585–1588, 1992.
- [20] D.S. Bloomberg and V. Castelli. Reformulation of Nonlinear Integral Magneto-static Equations for Rapid Iterative Convergence. *IEEE Transactions on Magnetics*, 21(2):1174–1180, 1985.
- [21] A. Bossavit. A Rationale for "Edge-Elements". *IEEE Transactions on Magnetics*, 24(1):74–79, 1988.
- [22] U. Brammer and P. Rasch. Optimization of Ferromagnetic Shields for Solenoidal SMES. *IEEE Transactions on Magnetics*, 32(3):1274–1277, 1996.
- [23] B. Brandstatter, C. Magele, C. Ragusa, and M. Repetto. Use of Stochastic Optimization Procedures in the Identification of the Generalized Moving Preisach Model.

- International Symposium on Non-Linear Electromagnetic Systems, Braunschweig, 1997.*
- [24] B.R. Brandstatter and W. Ring. Some Results on the TEAM Workshop Problem 25. *Proc. of the TEAM Workshop in the 6-th Round, Rio de Janeiro, Brazil*, pages 39–41, 1997.
- [25] J.I. Buchanan and P.P. Turner. *Numerical Methods and Analysis*. McGraw-Hill International Editions, 1992.
- [26] Erick Cantu-Paz. A Survey of Parallel Genetic Algorithms. *IlliGAL Report no.97003* (<http://gal4.ge.uiuc.edu>), 1997.
- [27] C.J. Carpenter. Comparison of alternative formulations of 3-dimensional magnetic-field and eddy-current problems at power frequencies. *IEE Proceedings*, 124:1026–1034, 1977.
- [28] M.V.K. Chari, J. D' Angelo, C. Crowley, J. Roeth, and H. Hurwitz. Solution of Open Boundary Problems by Differential and Integral Methods. *IEEE Transactions on Magnetics*, 22(5):1037–1039, 1986.
- [29] M. Chiampi, A. Negro, and M. Tartaglia. A Finite Element Method to Compute Three-Dimensional Magnetic Field Distribution in Transformer Cores. *IEEE Transactions on Magnetics*, 16(6):1413–1419, 1980.
- [30] M. Chiampi, M. Repetto, and D. Chiarabaglio. An Improved Technique for Nonlinear Magnetic Problems. *IEEE Transactions on Magnetics*, 30(6):4332–4334, 1994.
- [31] J.S. Chun, H.K. Jung, and J.S. Yoon. Shape Optimization of Closed Slot Type Permanent Magnet Motors for Cogging Torque Reduction using Evolution Strategies. *IEEE Transactions on Magnetics*, 33(2):1912–1915, 1997.
- [32] T.K. Chung, S.k. Kim, and S.Y. Hahn. Optimal Pole Shape Design for the Reduction of Cogging Torque of Brushless DC Motor Using Evolution Strategy. *IEEE Transactions on Magnetics*, 33(2):1908–1911, 1997.
- [33] G. Ciuprina. Formularea problemelor de câmp electromagnetic în medii neliniare imobile prin metode numerice; modele matematice și numerice pentru regimul magnetic staționar. Referat de doctorat, 1996.
- [34] G. Ciuprina and D. Ioan. Optimization of electromagnetic devices by distributed stochastic-deterministic algorithms - team 22 benchmark problem. *Seminarul Național de Electrotehnică teoretică, București.*, 1998.

- [35] G. Ciuprina and D. Ioan. Team problem 22 solved by a distributed stochastic-deterministic algorithm with accuracy control. *Proceedings of the TEAM Workshop, 7th Round, Tucson, Arizona*, pages 2–4, 1998.
- [36] G. Ciuprina, S. Stănescu, and D. Ioan. Efficiency and accuracy of field evaluation in team problem no. 25. *Proceedings of the TEAM Workshop, 8th Round, Graz, Austria.*, pages 581–584, 1998.
- [37] G. Ciuprina, B. Vasiliu, C. Dumitrescu, T. Chelcea, and D. Ioan. Use of Stochastic Algorithms for Distributed Architectures in the Optimization of Electromagnetic Devices. *Proceedings of the 11-th Conference on the Computation of Electromagnetic Fields (COMPUMAG), Rio de Janeiro, Brazil.*, pages 573–574, 1997.
- [38] E. Costamagna, A. Fanni, and M. Marchesi. Modified TEM Cell Design using Mixed Simulated Annealing-Deterministic Optimization. *IEEE Transactions on Magnetism*, 32(3):1202–1207, 1996.
- [39] R. Courant. *Differential and Integral Calculus*. Blackie & son limited, 1961.
- [40] S. Dappen and G. Hennenberg. A Sensitivity Approach for the Optimization of Loss Efficinecies. *IEEE Transactions on Magnetism*, 33(2):1836–1839, 1997.
- [41] K.R. Davey. Magnetic Design Optimization Using Variable Metric. *IEEE Transactions on Magnetism*, 31(6):3566–2568, 1995.
- [42] E. Demeter, F. Hăntilă, M. Vasiliu, and J.A. Tegopoulos. A New Method for Magnetic Field Calculation in Electrical Machines. *Electromotion*, 2(5):119–124, 1995.
- [43] X. Dexin, B. Baodong, Y. Yingying, W. Fengxiang, and O.A. Mohammed. Shaoe Design Optimization in Non-Linear Magnetic Problems Using Simulated Annealing with Complex Strategy. *IEEE Transactions on Magnetism*, 31(6):3569–3571, 1995.
- [44] G. Drago, A. Manella, M. Nervi, M. Repetto, and G. Secondo. A combined strategy for optimization in non linear magnetic problems using simulated annealing and search technique. *IEEE Transactions on Magnetism*, 28(6):1541–1544, 1992.
- [45] F. Dughiero, M. Guarnieri, and S. Lupi. An Optimization Procedure for Electromagnetic Confinement and Levitation Systems. *IEEE Transactions on Magnetism*, 29(2):1758–1761, 1993.
- [46] D. Dutta and C. Pal. Global Optimization of Electromagnets by Random Taby search Technique. *Int. J. of Applied Electromagnetics and Mechanics*, 6(4):255–271, 1995.

- [47] D.N. Dyck, D.A. Lowther, and E.M. Freeman. A Method of Computing the Sensitivity of Electromagnetic Quantities to Changes in Materials and Sources. *IEEE Transactions on Magnetism*, 30(5):3415–3418, 1994.
- [48] A. Fanni, M. Marchesi, A. Serri, and M. Usai. A Greedy Genetic Algorithm for Continuous Variables Electromagnetic Optimization Problems. *IEEE Transactions on Magnetism*, 33(2):1900–1903, 1997.
- [49] R. Fletcher. *Practical Methods of Optimization*. John Wiley & sons, 1987.
- [50] A. Formisano, R. Martone, and F. Villone. Optimization of the Magnetostatic Field in SMES Devices . *IEEE Transactions on Magnetism*, 33(2):1832–1835, 1996.
- [51] K. Forsman, L. Kettunen, and J.P. Nuutinen. Performance of a Nonlinear 2D and 3D Magnetostatic Volume Integral Formulation. *IEEE Transactions on Magnetism*, 31(3):1488–1491, 1995.
- [52] K. Fujiwara, T. Nakata, N. Okamoto, and K. Muramatsu. Method for Determining Relaxation Factor for Modified Newton-Raphson Method. *IEEE Transactions on Magnetism*, 29(2):1962–1965, 1993.
- [53] A. Geist, A. Bequelin, J. Dongarra, W. Jiang, R. Mancheck, and V. Sunderam. *PVM: Parallel Virtual Machine. A User's Guide and Tutorial for Networked Parallel Computing*. The MIT Press, Cambridge, Massachusetts, 1994.
- [54] A. Gottvald, K. Preis, C. Magele, O. Biro, and A. Savini. Global Optimization Methods for Computational Electromagnetics. *IEEE Transactions on Magnetism*, 28(2):1537–1540, 1992.
- [55] K. Hameyer and R. Hanitsch. Numerical Optimization of the Electromagnetic Field by Stochastic Search and MEC-Model. *IEEE Transactions on Magnetism*, 30(5):3431–3434, 1994.
- [56] R.W. Hamming. *Numerical Methods for Scientists and Engineers*. Dover Publications, Inc., New York, 1973.
- [57] P. Hammond and T.D. Tsiboukis. Dual Finite-Element Calculations for Static Electric and Magnetic Fields. *IEE Proceedings*, 130:105, 1983.
- [58] R. Haupt. Comparaison Between Genetic and Gradient-Based Optimization Algorithms for Solving Electromagnetics Problems. *IEEE Transactions on Magnetism*, 31(3):1932–1935, 1995.
- [59] C.S. Holzinger. Computation of Magnetic Fields Within Three-Dimensional Highly Nonlinear Media. *IEEE Transactions on Magnetism*, 6(1):60–65, 1970.

- [60] S.R.H. Hoole and M.K. Haldar. Optimization of Electromagnetic Devices: Circuit Models, Neural Networks and Gradient Methods in Concert. *IEEE Transactions on Magnetics*, 31(3):2016–2019, 1995.
- [61] S.R.H. Hoole and S. Subramaniam. Higher Finite Element Derivatives for the Quick Synthesis of Electromagnetic Devices. *IEEE Transactions on Magnetics*, 28(2):1565–1568, 1992.
- [62] S.R.H. Hoole and S. Subramaniam. Inverse Problems with Boundary Elements: Synthesizing a Capacitor. *IEEE Transactions on Magnetics*, 28(2):1529–1532, 1992.
- [63] I.F. Hăntilă. *Contribuții asupra teoriei mașinilor de curent continuu cu magneți permanenți*. PhD thesis, Institutul Politehnic București, 1974.
- [64] I.F. Hăntilă. Existence and Stability Theorems for Stationary Fields in Nonlinear Media. *Revue Roumaine des Sciences Techniques*, 26(4), 1981.
- [65] I.F. Hăntilă. *Calculul câmpului electromagnetic cu ajutorul calculatorului*. TEMPUS Postgraduate School of Computer Aided Electrical Engineering, Bucharest, 1993.
- [66] I.F. Hăntilă. Metode pentru calculul câmpului electromagnetic în medii neliniare. Notițe de curs, SPIEAC, 1993.
- [67] I.F. Hăntilă and G. Grama. An Overrelaxation Method for the Computation of the Fixed Point of a Contractive Mapping . *Revue Roumaine des Sciences Techniques*, 27(4):395–398, 1982.
- [68] I.F. Hăntilă et al. *Câmpul electromagnetic staționar în medii neliniare*. Editura ICPE, 1997.
- [69] D.H. Im, S.C. Oark, and J.W. Im. Design of Single-Sided Linear Induction Motor Using the Finite Element Method and SUMT. *IEEE Transactions on Magnetics*, 29(2):1762–1766, 1993.
- [70] D. Ioan, G. Ciuprina, and C. Dumitrescu. Use of stochastic algorithms for distributed architectures in the optimization of electromagnetic devices. *IEEE Transactions on Magnetics*, 34(5):3000–3003, 1998.
- [71] D. Ioan, G. Ciuprina, and A. Szigeti. Embedded stochastic-deterministic optimization method with accuracy control. *Proceedings of the Conference on Electromagnetic Field Computation (CEFC 98), Tucson, Arisona*, 1998.
- [72] D. Ioan, I.Munteanu, and G.Ciuprina. Adjoint Field Technique Applied in Optimal Design of a Nonlinear Inductor. *Proceedings of the 11-th Conference on the Computation of Electromagnetic Fields (COMPUMAG), Rio de Janeiro, Brazil*, pages 141–142, 1997.

- [73] D. Ioan, I. Munteanu, and G. Ciuprina. Adjoint field technique applied in optimal design of a nonlinear inductor. *IEEE Transactions on Magnetics*, 34(5):2849–2852, 1998.
- [74] D. Ioan et al. Tehnici avansate pentru analiza numerică a câmpului electromagnetic. Raport LMN-3/95-FLD-CC-ALL, 1995.
- [75] D. Ioan et al. Calcul de înaltă performanță: studii privind algoritmi, arhitecturi și metode specifice. Raport LMN-1/97-CMP-CC-ALL, 1997.
- [76] D. Ioan et al. Tehnici avansate pentru analiza numerică a câmpului electromagnetic. Raport LMN-3/97-FLD-CC-ALL, 1997.
- [77] B. Ionescu. *Numerical Methods in Electromagnetic Field Computations*. TEMPUS Postgraduate School of Computer Aided Electrical Engineering, Bucharest, 1993.
- [78] T. Ishikawa, Y. Tsukui, and M. Matsunami. Optimization of Electromagnetic Devices Using Artificial Neural Network with Quasi-Newton Algorithm. *IEEE Transactions on Magnetics*, 32(3):1226–1229, 1996.
- [79] A. Ishiyama, K. Shimizu, and A. Sakahara. An Optimal Design Method for Multi-section Superconducting Magnets using Modified Simulated Annealing. *IEEE Transactions on Magnetics*, 30(5):3435–3438, 1994.
- [80] K. Kadded, R.R. Saldanha, and J.L. Coulomb. Mathematical Minimization of the Time Harmonics of the E.M.F. of a DC-PM Machine using a Finite Element Method. *IEEE Transactions on Magnetics*, 29(2):1747–1752, 1993.
- [81] M. Kasper, K. Hameyer, and A. Kost. Automated Optimal Design of a Permanent Magnet DC Motor Using Global Evolution Strategies and FEM. *Int. J. of Applied Electromagnetics and Mechanics*, 6(4):367–376, 1995.
- [82] J. Kim, H. Lee, H. Jung, and S. Hahn. Optimal Design Technique for Waveguide Device. *IEEE Transactions on Magnetics*, 32(3):1290–1253, 1996.
- [83] M. Kitamura. A Shape Optimization Method Based on Complex Expressions of 2-Dimensional Magnetic Fields. *IEEE Transactions on Magnetics*, 33(2):1884–1887, 1997.
- [84] C.S. Koh. Magnetic Pole Shape Optimization of Permanent Magnet Motor for Reduction of Cogging Torque. *IEEE Transactions on Magnetics*, 33(2):1822–1827, 1997.
- [85] C.S. Koh and S.Y. Hahn. Design Sensitivity Analysis for Shape Optimization of 3D Electromagnetic Devices. *IEEE Transactions on Magnetics*, 29(2):1753–1757, 1993.

- [86] C.S. Koh, S.Y. Hahn, T.K. Chung, and H.K. Jung. A Sensitivity Analysis using Boundary Element Method for Shape Optimization of Electromagnetic Devices. *IEEE Transactions on Magnetics*, 28(2):1577–1580, 1992.
- [87] C.S. Koh, S.Y. Hahn, H.K. Jung, and K. Choi. A Continuum Approach in Shape Design Sensitivity Analysis of Magnetostatic Problems Using the Boundary Element Method. *IEEE Transactions on Magnetics*, 29(2):1771–1774, 1993.
- [88] C.S. Koh and O.A. Mohammed. Detection of magnetic body using artificial network with modified simulated annealing. *IEEE Transactions on Magnetics*, 30, 1994.
- [89] C.S. Koh, O.A. Mohammed, and S. Hahn. Nonlinear Shape Design Sensitivity Analysis of Magnetostatic Problems using Boundary Element Method. *IEEE Transactions on Magnetics*, 31(3):1944–1947, 1995.
- [90] B. Krstajic, Z. Andelic, and S. Salon. Nonlinear 3D Magnetostatic Field Calculation by the Integral Equation Method with Surface and Volume Magnetic Charges. *IEEE Transactions on Magnetics*, 28(2):1088–1091, 1992.
- [91] M.H. Lean. Application of Boundary Integral Equation Methods to Electromagnetics. *IEEE Transactions on Magnetics*, 21(5):1823–1829, 1985.
- [92] M.H. Lean and A. Wexler. Accurate Field Computation with the Boundary Element Method. *IEEE Transactions on Magnetics*, 18(2):331–335, 1982.
- [93] H. Lee, H.K. Jung, S. Hahn, C. Cheon, and K.S. Lee. Shape Optimization of H-Plane Waveguide Tee Junction Using Edge Finite Element Method. *IEEE Transactions on Magnetics*, 31(3):1928–1931, 1995.
- [94] H.B. Lee, S.B. Park, Il.H. Park, and S.Y. Hahn. Core Shape Design of Electrical Devices for Iron Loss Reduction. *Int. J. of Applied Electromagnetics and Mechanics*, 6(4):331–338, 1995.
- [95] K. Lee, M.J. Debortoli, M.J. Lee, and S.J. Salon. Coupling Finite Elements and Analytical Solution in the Airgap of Electric Machines. *IEEE Transactions on Magnetics*, 27(5):3955–3957, 1991.
- [96] S.C. Lin, W.F. Punch, and E.D. Goodman. Coarse-grain Genetic Algorithms, Categorization and new Approaches. *Proc. of the 6-th Parallel and Distributed Algorithms, IEEE Conference, Dalas TX*, pages 28–37, 1994.
- [97] D.A. Lowther, C.B. Rajanathan, and P.P. Silvester. A Finite Element Technique for Solving 2D Open Boundary Problems. *IEEE Transactions on Magnetics*, 14(5):467–469, 1978.

- [98] C. Magele, G. Fuertratt, I. Bardi, K.R. Richter, G. Schoenwetter, P. Alotto, G. Molinari, and M. Repetto. SMES Optimization Benchmark - Preliminary TEAM Workshop Problem 22. *TEAM Workshop, Okayama*, pages 38–48, 1996.
- [99] C. Magele, K. Preis, and W. Renhart. Some Improvements in Nonlinear 3D Magnetostatics. *IEEE Transactions on Magnetics*, 26(2):375–378, 1990.
- [100] C. Magele, K. Preis, W. Renhart, R. Dyczij-Edlinger, and K.R. Richter. Higher Order Evolution Strategies for the Global Optimization of Electromagnetic Devices. *IEEE Transactions on Magnetics*, 29(2):1775–1778, 1993.
- [101] M. Marchesi, G. Molinari, and M. Repetto. A Parallel Simulated Annealing Algorithm for the Design of Magnetic Structures. *IEEE Transactions on Magnetics*, 30(5):3439–3442, 1994.
- [102] M. Marchesi, G. Molinary, and M. Repetto. Global Optimization for Discrete Magnetostatic Problems. *IEEE Transactions on Magnetics*, 29(2):1779–1782, 1993.
- [103] I.D. Mayergoyz and T. Doong. Hybrid Boundary-Volume Galerkin's Method for Nonlinear Magnetostatic Problems. *Journal of Applied Physics*, 57(1):3838–3840, 1985.
- [104] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag Berlin, 1996.
- [105] Gh. Mîndru and M.M. Rădulescu. *Analiza numerică a câmpului electromagnetic*. Ed.Dacia, Cluj-Napoca, 1986.
- [106] C. Mocanu. *Bazele electrotehnicii. Teoria câmpului electromagnetic*. Ed. Didactică și Pedagogică, București, 1991.
- [107] O. Mohammed, N. Demerdash, and T. Nehl. Nonlinear Vector Potential Formulation and Experimental Verification of Newton-Raphson Solution of 3 Dimensional Magnetostatic Fields in Electrical Devices. *IEEE Transactions on Energy Conversion*, 1(1):177–185, 1986.
- [108] O.A. Mohammed, F.G. Uler, S. Russenschuck, and M. Kasper. Design Optimization of a Superferic Octupole Using Various Evolutionary and Deterministic Techniques. *IEEE Transactions on Magnetics*, 33(2):1816–1821, 1997.
- [109] O.A. Mohammed and G.F. Uler. A Hybrid Technique for the Optimal Design of Electromagnetic Devices Using Direct Search and Genetic Algorithms. *IEEE Transactions on Magnetics*, 33(2):1931–1934, 1997.

- [110] J.J. More and S. J. Wright. *Optimization Software Guide*, volume 14 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, 1993.
- [111] I. Munteanu. *Calculul numeric al câmpului electromagnetic - contribuții la analiza câmpului electromagnetic general variabil prin metoda volumelor finite*. PhD thesis, Universitatea Politehnica București, 1997.
- [112] I. Munteanu and F.M.G. Tomescu. Optimization of a Magnetic Device Based on Symbolic Analysis of the Field. *IEEE Transactions on Magnetics*, 33(2):1840–1843, 1997.
- [113] T. Nakata, N. Takahashi, K. Fujiwara, N. Okamoto, and K. Muramatsu. Improvements of Convergence Characteristics of Newton-Raphson Method for Nonlinear Magnetic Field Analysis. *IEEE Transactions on Magnetics*, 28(2):1048–1051, 1992.
- [114] C. Neagoe and F. Ossart. Analysis of Convergence in Nonlinear Magnetostatic Finite Elements Problems. *IEEE Transactions on Magnetics*, 30(5):2865–2868, 1994.
- [115] K. Nishimura, S. Nakata, and T. Nakagawa. Optimization of the Coil Distribution of the Deflection Yoke for CRT. *IEEE Transactions on Magnetics*, 33(2):1848–1851, 1997.
- [116] S. Noguchi and A. Ishiyama. Optimal Design Method for MRI Superconducting Magnets with Ferromagnetic Shield. *IEEE Transactions on Magnetics*, 33(2):1904–1907, 1997.
- [117] T. Onuki. Hybrid Finite Element and Boundary Element Method Applied to Electromagnetic Problems. *IEEE Transactions on Magnetics*, 26(2):582–587, 1990.
- [118] T. Onuki and Y. Toda. Optimal Design of Hybrid Magnet in MAGLEV System with Both Permanent and Electro Magnets. *IEEE Transactions on Magnetics*, 29(2):1783–1786, 1993.
- [119] Il. Park, J.L. Coulomb, and S. Hahn. Implementation of Continuum Sensitivity Analysis with Existing Finite Element Code. *IEEE Transactions on Magnetics*, 29(2):1787–1790, 1993.
- [120] Il. Park, B. Lee, and S. Hahn. Design Sensitivity Analysis for Nonlinear Magnetostatic Problems using Finite Element Method. *IEEE Transactions on Magnetics*, 28(2):1533–1536, 1992.
- [121] Il.H. Park, H. Lee, I. Kwak, and S. Hahn. Design Sensitivity Analysis for Steady State Eddy Current Problems by Continuum Approach. *IEEE Transactions on Magnetics*, 30(5):3411–3414, 1994.

- [122] Il.H. Park, S. Lee, and S. Hahn. Sensitivity Analysis Based on Analytic Approach for Shape Optimization of Electromagnetic Devices: Interface Problem on Iron and Air. *IEEE Transactions on Magnetics*, 27(5):4142–4145, 1991.
- [123] J.P. Peng and S.J. Salon. Hybrid Finite Element Boundary Element Solutions Using Half-Space Green's Functions. *Journal of Applied Physics*, 55(6):2198–2200, 1984.
- [124] J. Penman and J.R. Fraser. Complementary and Dual Energy Finite Element Principles in Magnetostatics. *IEEE Transactions on Magnetics*, 18(2):319, 1982.
- [125] J. Penman and J.R. Fraser. Unified approach to problems in electromagnetism. *IEE Proceedings*, 131(1):55–61, 1984.
- [126] T.H. Pham and S.R.S. Hoole. Unconstrained Optimization of Coupled Magneto-Thermal Problems. *IEEE Transactions on Magnetics*, 31(3):1988–1991, 1995.
- [127] T.H. Pham, S.J. Salon, and S.R.H. Hoole. Shape Optimization of Windings for Minimum Losses. *IEEE Transactions on Magnetics*, 32(3):4287–4289, 1996.
- [128] O. Pironneau. *Optimal Shape Design for Elliptic Systems*. Springer-Verlag, 1984.
- [129] K. Preis, I. Bardi, O. Biro, C. Magele, W. Renhart, K.R. Richter, and G. Vrisk. Numerical Analysis of 3D Magnetostatic Fields . *IEEE Transactions on Magnetics*, 27(5):3798–3802, 1991.
- [130] K. Preis, O. Biro, M. Friedrich, A. Gottvald, and C. Magele. Comparaison of different optimization strategies in the design of electromagnetic devices. *IEEE Transactions on Magnetics*, 27(5):4154–4157, 1991.
- [131] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C. The Art of Scientific Computing*. Cambridge University Press, 1992.
- [132] J.A. Ramirez, E.M. Freeman, C. Chat-uthai, and D.A. Lowther. Sensitivity Analysis for the Automatic Shape Design of Electromagnetic Devices in 3D using FEM. *IEEE Transactions on Magnetics*, 33(2):1856–1859, 1997.
- [133] L. Rao and W. Yan. Mean Field Annealing (MFA) and Optimal Design of Electromagnetic Devices. *IEEE Transactions on Magnetics*, 32(3):1218–1221, 1996.
- [134] Z. Ren, F. Bouillault, A. Razek, and J.C. Verite. Comparison of Different Boundary Integral Formulations when Coupled with Finite Elements in Three Dimensions. *IEE Proceedings*, 135(8):501–507, 1988.
- [135] W.M. Rucker and K.R. Richter. Three-Dimensional Magnetostatic Field Calculation Using Boundary Element Method. *IEEE Transactions on Magnetics*, 24(1):23–26, 1988.

- [136] S. Russenschuck. Application of Lagrange Multiplier Estimation to the Design Optimization of Permanent Magnet Synchronous Machines. *IEEE Transactions on Magnetics*, 28(2):1525–1528, 1992.
- [137] S. Russenschuck. A Survey of Mathematical Optimization and Inverse Problems in Electromagnetics - with Applications to the design of Superconducting Magnets. *Int. J. of Applied Electromagnetics and Mechanics*, 6(4):277–295, 1995.
- [138] R.R. Saldanha, J.L. Coulomb, A. Foggia, and J.C. Sabonnadiere. A Dual Method for Constrained Optimization Design in Magnetostatic Problems. *IEEE Transactions on Magnetics*, 27(5):4136–4141, 1991.
- [139] R.R. Saldanha, S. Pelissier, K. Kadded adn Y.P. Yonnet, and J.L. Coulomb. Non-linear Optimization Methods Applied to Magnetic Actuators Design. *IEEE Transactions on Magnetics*, 28(2):1581–1584, 1992.
- [140] S.J. Salon. The Hybrid Finite Element-Boundary Element Method in Electromagnetics. *IEEE Transactions on Magnetics*, 21(5):1829–1834, 1985.
- [141] S.J. Salon and J.P. Peng. A Hybrid Finite Elemene-Boundary Element Formulation of Poisson's Equation for Axsymmetric Vector Potential Problems. *Journal of Applied Physics*, 53(11):8420–8422, 1982.
- [142] G. Schonwetter, C. Magele, K. Preis, C. Paul, W. Renhart, and K.R. Richter. Optimization of SMES Solenoids with regard to their Stray Fiels. *IEEE Transactions on Magnetics*, 31(3):1940–1943, 1995.
- [143] K.R. Shao and K.D. Zhou. The Iterative Boundary Element Method for Nonlinear Electromagnetic Field Calculations. *IEEE Transactions on Magnetics*, 24(1):150–153, 1988.
- [144] P.P. Silvester and R.L. Ferrari. *Finite Elements for Electrical Engineers*. Cambridge University Press, second edition, 1990.
- [145] J. Simkin and C.W. Trowbridge. On the use of the total scalar potential in the solution of field problems in magnetostatics. *International Journal of Numerical Methods in Engineering*, 14:423–440, 1979.
- [146] J. Simkin and C.W. Trowbridge. Three-Dimensional Nonlinear Electromagnetic Field Computations Using Scalar Potentials. *IEE Proceedings*, 127(6):368–374, 1980.
- [147] J. Simkin and C.W. Trowbridge. Optimization problems in electromagnetics . *IEEE Transactions on Magnetics*, 27(5):4016–4019, 1991.

- [148] J. Simkin and C.W. Trowbridge. Optimizing electromagnetic devices combining direct search methods with simulated annealing. *IEEE Transactions on Magnetics*, 28(6):1545–1548, 1992.
- [149] D. Srinivasan and S. Ratnajeevan. Fuzzy Multiobject Optimization for the Starting Design of a Magnetic Circuit. *IEEE Transactions on Magnetics*, 32(3):1230–1233, 1996.
- [150] S. Subramaniam, A.A. Arkadan, and S.R.S. Hoole. Optimization of a Magnetic Pole Face Using Linear Constraints to Avoid Jagged Contours. *IEEE Transactions on Magnetics*, 30(5):3455–3458, 1994.
- [151] N. Takahashi. Optimization of Die Press Model- TEAM Workshop Problem 25. *TEAM Workshop, Okayama*, pages 61–69, 1996.
- [152] N. Takahashi, K. Ebihara, K. Yoshida, T. Nakata, K. Ohashi, and K. Miyata. Investigation of Simulated Annealing Method and Its Application to Optimal Design of Die Mold for Orientation of Magnetic Powder. *IEEE Transactions on Magnetics*, 32(3):1210–1213, 1996.
- [153] N. Takahashi, T. Nakata, and N. Uchiyama. Optimal Design Method of 3D Non-linear Magnetic Circuit by Using Magnetization Integral Equation Method. *IEEE Transactions on Magnetics*, 25(5):4144–4146, 1989.
- [154] T. Takahashi. Shape Optimization Method for Coils Consisting of Free Curves. *IEEE Transactions on Magnetics*, 29(2):1803–1806, 1993.
- [155] M. Trlep, L. Skerget, B. Kreco, and B. Hribernic. Hybrid Finite Element Boundary Element Method for Nonlinear Electromagnetic Problems. *IEEE Transactions on Magnetics*, 31(3):1380–1383, 1995.
- [156] H. Tsuboi and T. Misaki. The Optimum Design of Electrode and Insulator Contours by Nonlinear Programming Using the Surface Charge Simulation Method. *IEEE Transactions on Magnetics*, 24(1):35–38, 1988.
- [157] A. Țugulea et al. Calculul parametrilor electromagnetici ai unui solenoid supraconductor. Contract 3-1-1/81, 1981.
- [158] G.F. Uler and O.A. Mohammed. Ancillary Techniques for the Practical Implementation of GAs to the Optimal Design of Electromagnetic Devices. *IEEE Transactions on Magnetics*, 32(3):1194–1197, 1996.
- [159] G.F. Uler, O.A. Mohammed, and C.S. Koh. Utilizing genetic algorithms for the optimal design of electromagnetic devices. *IEEE Transactions on Magnetics*, 30(6):4296–4298, 1994.

- [160] G.F. Uler, O.A. Mohammed, and C.S. Koh. Design Optimization of Electrical Machines using Genetic Algorithms. *IEEE Transactions on Magnetics*, 31(3):2008–2011, 1995.
- [161] J.A. Vasconcelos. Simulated Annealing coupled with the tabu Search Method for Continuum Optimization in Electromagnetics. *IEEE Transactions on Magnetics*, 32(3):1206–1209, 1996.
- [162] J.A. Vasconcelos, L. Krahenbuhl, L. Nicolas, and A. Nicolas. Design Optimization in Electrostatic Field Analysis Using the BEM and the Augmented Lagrangean Method . *IEEE Transactions on Magnetics*, 30(5):3443–3446, 1994.
- [163] J.A. Vasconcelos, R.R. Saldanha, L. Krahenbuhl, and A. Nicolas. Genetic Algorithm Coupled with a Deterministic Method for Optimization in Electromagnetics. *IEEE Transactions on Magnetics*, 33(2):1860–1867, 1997.
- [164] B. Vasiliu, I. Munteanu, D. Ioan, and G.Ciuprina. Use of Message-Passing Distributed Architecture in Optimisation of a SMES. *Proceedings of the 4-th Romanian Conference on Open Systems (ROSE 96), Bucharest, Romania*, pages 72–79, 1996.
- [165] R. Wang and N.A. Demerdash. A Combined Vector Potential Method for FE Computation of 3D Magnetic Fields in Electrical Devices with Iron Cores. *IEEE Transactions on Magnetics*, 27(5):3971–3977, 1991.
- [166] K. Weeber and S.R.H. Hoole. The Subregion Method in Magnetic Field Analysis and Design Optimization. *IEEE Transactions on Magnetics*, 28(2):1561–1564, 1992.
- [167] K. Weeber and S.R.H. Hoole. Structural Design Optimization as a Technology Source for Developments in the Electromagnetic Domain. *IEEE Transactions on Magnetics*, 29(2):1807–1811, 1993.
- [168] D.S. Weile and E. Michielssen. Genetic Algorithm Optimization Applied to Electromagnetics: A Review. *IEEE Transactions on Antennas and Propagation*, 45(3):343–353, 1997.
- [169] F. Wurtz, M. Richomme, J. Bigeon, and J.C. Sabonnadiere. A Few Results for using Genetic Algorithms in the Design of Electrical Machines. *IEEE Transactions on Magnetics*, 33(2):1892–1895, 1997.
- [170] W. Youhua and Y. Weili. Adaptive Simulated Annealing for the Optimal Design of Electromagnetic Devices. *IEEE Transactions on Magnetics*, 32(3):1214–1217, 1996.
- [171] S.G. Zaky and S.D.T. Robertson. Integral Equation Formulation for the Solution of Magnetic Field Problems. Part I: Dipole and Current Models. *IEEE Transactions on Power Apparatus and Systems*, pages 808–815, 1973.

- [172] O.C. Zienkiewicz, J. Lyness, and D.J.R.Owen. Three dimensional magnetic field using a scalar potential. *IEEE Transactions on Magnetics*, 13(5):1649–1656, 1977.