

Prefață

În ultimii ani, a intrat în uz folosirea calculatorului pentru explicarea și însușirea corectă a unor concepte și aspecte semnificative specifice majorității specialităților științifice și tehnice. Prin îmbinarea puterii mari de calcul, cu o serie de facilități grafice și prin folosirea unor metode de simulare și modelare specifice, devine posibilă formarea unor imagini intuitive corecte asupra unor aspecte fenomenologice conținute în formule cunoscute, aparent banale, dar suficient de abstracte pentru a ascunde în primă instanță aspectul lor practic. O serie de exemple simple pot fi generalizate prin luarea în considerare a mai multor factori, așa încât în final să rezulte o imagine globală mult mai fidelă a unui anumit proces sau fenomen. De obicei, în monografiile consacrate studiului unor anumite discipline, sunt examinate cazuri idealizate care permit o tratare experimentală și teoretică relativ simplă, însă abordarea situațiilor reale complexe impune utilizarea metodelor numerice, care devin eficiente doar prin utilizarea tehnicilor moderne de calcul.

Prima parte a lucrării este consacrată descrierii unor metode numerice simple folosite pentru rezolvarea multor probleme cu caracter tehnico-științific. În prezentarea acestora s-a renunțat deliberat la o matematizare excesivă, specifică tratatelor de analiză numerică, deoarece un utilizator fără o pregătire superioară de specialitate este mult mai interesat de aspectul practic, legat de conceperea, redactarea, programarea și optimizarea unor algoritmi de calcul, decât de însușirea detaliilor teoretice pe care aceștia se fundamentează. Pentru a asigura o accesibilitate cât mai largă, nu sunt examinate unele metode sofisticate de dată mai recentă, acestea găsimându-și locul mai curând în tratatele de specialitate, decât într-o lucrare cu caracter instructiv.

Toate metodele descrise sunt însoțite de câte un program de calcul, de obicei de tip funcție externă sau subrutină, care implementează raționamentele din secțiunea aferentă. Aceste programe sursă nu conțin comentarii, tocmai pentru a obliga utilizatorul să nu se mulțumească numai cu o simplă copiere a acestora, ci și să-l determine să deslușească modul în care un algoritm, mai simplu sau mai complex, poate fi transpus într-o secvență de instrucțiuni care să respecte cu fidelitate obiectivele enunțate. Există și situații în care sunt prezentate mai multe variante ale aceleiași metode, precum și situații în care, păstrându-se aceeași structură logică, unele metode pot fi extrapolate la cazuri mai generale.

Metodele și programele de calcul descrise în partea a doua au un caracter expozitiv, ele având menirea de a indica modul în care pot fi dobândite și valorificate o serie de cunoștințe de fizică generală, folosind și alte procedee decât cele analitice sau experimentale. În marea majoritate a cazurilor, elementele de teorie sunt prezentate strict informativ, formulele folosite fiind preluate din tratatele de specialitate, eventual însoțite de unele comentarii, care să sublinieze importanța acestora în contextul general al subiectului analizat. De obicei, acesta nu este dezvoltat până la capăt, rămânând în sarcina utilizatorului examinarea altor cazuri mai complexe, sau extinderea raționamentelor la alte situații sugerate în text. Datele folosite în programe sunt uneori „forțate”, în sensul că ele sunt alese astfel încât rezultatele să fie cât mai intuitive și semnificative pentru fenomenul studiat. Programele de calcul prezentate pot fi împărțite în trei mari categorii.

Un prim grup îl reprezintă programele cu un conținut în principal descriptiv, care au ca obiect prezentarea și ilustrarea unor fenomene fizice mai simple sau mai complexe, fiind utilizate procedee și formule cunoscute, care țin mai mult de cultura generală a fizicianului. Calculatorul este folosit doar pentru a efectua calcule și reprezentări grafice, care la fel de bine pot fi efectuate și manual, dar ar lua un timp incomparabil mai lung. Nu sunt utilizate metode numerice, sau, în eventualitatea folosirii acestora, rolul lor este doar de a verifica un rezultat, care poate fi obținut și prin alte metode.

Un alt grup îl reprezintă programele care au ca scop rezolvarea unor probleme pentru care este cunoscută o soluție analitică exactă, obținută de obicei în urma unor calcule foarte lungi și complicate, interpretarea fizică a acesteia fiind de asemenea destul de dificilă. Aceeași soluție, deși aproximativă în limita unor erori acceptabile, poate fi obținută de multe ori mult mai ușor prin metode numerice. În acest caz soluția analitică cunoscută va fi utilizată doar pentru a selecta datele inițiale care conduc la rezultate semnificative ca interpretare și conținut.

În fine, obiectul unor programe este rezolvarea unor probleme pentru care, deși soluția există, ea nu poate fi obținută în formă exactă prin metode analitice. În astfel de situații, metodele numerice bazate pe algoritmi cunoscuți, pot conduce la o soluție a cărei corectitudine poate fi eventual verificată prin măsurători experimentale, sau prin compararea cu soluții analitice aproximative.

Din motive de spațiu, funcțiile externe și subrutinele care fac referiri la metodele numerice descrise în prima parte nu sunt incluse și în programele sursă, dar de câte ori este necesar, sunt făcute trimeri în text la paragraful unde acestea sunt descrise în detaliu. Procedurile grafice folosite sunt prezentate succint într-una din anexe, împreună cu descrierea în detaliu a subrutinei care a fost concepută special pentru a permite ilustrarea prin reprezentări grafice a unor rezultate semnificative.

Toate programele și subprogramele prezentate sunt redactate în Visual Basic și ele reprezintă transcrieri ale unor versiuni anterioare redactate în Fortran. Deși

limbajul Fortran rămâne totuși cel mai eficient din punctul de vedere al puterii de calcul, el fiind conceput și utilizat ani de-a rândul exclusiv în scopul efectuării de calcule științifice și tehnice, odată cu răspândirea pe scară largă a PC-urilor se manifestă tendința de a se utiliza și alte limbaje numerice de programare, cum ar fi Basic, Pascal sau C/C++.

Alegerea sistemului Visual Basic s-a bazat în primul rând pe faptul că limbajul Basic, în care își află originea, poate fi însușit fără nici un fel de dificultăți de orice utilizator, iar interfața grafică cu care este înzestrat permite programarea comodă, interactivă, a principalelor evenimente sub Windows, chiar de către cei cu cunoștințe relativ minime în domeniu. Nu trebuie uitat totuși faptul că sistemul este relativ lent, chiar și dacă este folosită versiunea pe 32 de biți, așa-zisul compilator fiind de fapt un interpretor de comenzi oarecum similar cu cel folosit în sistemul QuickBasic, iar prin convertirea proiectelor în programe executabile, sunt eliminate doar secvențele care realizează interoperativitatea dintre utilizator și sistem, optimizarea codului realizată de compilatoarele propriu-zise întrând exclusiv în sarcina programatorului. Această deficiență este parțial eliminată prin introducerea în secțiunea de declarații generale a fiecărui modul a perechii de instrucțiuni DefLng I-N: DefDb1 A-H, O-Z, ceea ce are ca efect faptul că tuturor variabilelor numerice li se atribuie de la început un tip predefinit, făcând astfel inutile o serie de declarații și conversii explicite (este de reținut că tipul implicit de dată în Visual Basic este variant-nedefinit). În același scop, deoarece datele numerice de tip complex nu sunt recunoscute de sistem, s-a preferat indicarea concretă a operațiilor specifice acolo unde era cazul, în locul elaborării unor proceduri proprii care prin apelare ar fi consumat doar timp de calcul. De asemenea, ori de câte ori a fost posibil, au fost efectuate optimizări ale secvențelor de cod, dacă prin aceasta nu se pierdea din claritate. În majoritatea cazurilor, controalele prin intermediul cărora este dirijată succesiunea acțiunilor sunt grupate pe o singură formă, programul ca atare fiind dezvoltat pe evenimentul Click al unui buton de comandă, iar reprezentarea grafică este realizată într-o casetă cu imagine situată pe aceeași formă. În fine, procedurile mai des folosite pot fi reunite într-o bibliotecă de module obiect, de unde pot fi extrase după caz, atât în faza de editare a legăturilor, cât și în cursul execuției.