

# Numeričke, simboličke i heurističke metode

---

**Dvornik, Josip**

*Source / Izvornik:* **Građevinar, 2003, 55, 575 - 582**

**Journal article, Published version**

**Rad u časopisu, Objavljena verzija rada (izdavačev PDF)**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:237:176067>

*Rights / Prava:* [In copyright](#) / [Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-10-05**

*Repository / Repozitorij:*

[Repository of the Faculty of Civil Engineering,  
University of Zagreb](#)



# Numeričke, simboličke i heurističke metode

Josip Dvornik

## Ključne riječi

računalo,  
numeričke metode,  
simboličke metode,  
heurističke metode,  
metode "umjetne  
inteligencije"

## Key words

computer,  
numerical methods,  
symbolic methods,  
heuristic methods,  
"artificial intelligence"  
methods

## Mots clés

ordinateur,  
méthode numériques,  
méthodes symboliques,  
méthodes heuristiques,  
méthodes d'intelligence  
artificielle

## Ключевые слова

компьютер (ЭВМ),  
числовые методы,  
символические методы,  
эвристические методы,  
методы "искусственной  
интеллигенции"

## Schlüsselworte:

Rechneranlage,  
numerische Methoden,  
symbolische Methoden,  
heuristische Methoden,  
Methoden der  
"künstlichen Intelligenz"

J. Dvornik

Pregledni rad

## Numeričke, simboličke i heurističke metode

Dan je pregled primjene računala pri čemu je težište na primjene u tehničkoj mehanici i projektiranju konstrukcija. Istaknuto je da su u primjeni računala najzastupljenije numeričke, simboličke i heurističke metode, a i metode "umjetne inteligencije". Spominju se i različiti grafički prikazi o kojima se u radu ne govori posebno. Istaknuto je da se još uvijek najviše upotrebljavaju numeričke metode ali i primjena ostalih metoda vrlo brzo raste i povećava se iz godine u godinu.

J. Dvornik

Subject review

## Numerical, symbolic and heuristic methods

An overview of computer use is given and, at that, an emphasis is placed on the use of computers in engineering mechanics and structural design. It is stressed that numerical, symbolic and heuristic methods, but also those based on "artificial intelligence", are mostly suited for computer use. Various graphical representations are also mentioned but are not separately analyzed. It is emphasized that, although numerical methods are still in leading position, an increase has been noted in the use of other methods which are gaining terrain year after year.

J. Dvornik

Ouvrage de synthèse

## Méthodes numériques, symboliques et heuristiques

L'article donne un aperçu de l'utilisation des ordinateurs, et un accent particulier est mis sur leur application dans la mécanique technique et dans l'étude des constructions. On souligne que l'utilisation des ordinateurs implique notamment les méthodes numériques, symboliques et heuristiques, ainsi que les méthodes d'intelligence artificielle. On mentionne aussi différentes représentations graphiques, sans approfondir cette question. On précise que les méthodes numériques restent toujours prépondérantes, mais que les autres méthodes s'imposent de plus en plus et que leur utilisation augmente d'année en année.

Й. Дворник

Обзорная работа

## Числовые, символические и эвристические методы

В работе дан обзор применения компьютеров (ЭВМ), при чём суть заключается в применении в технической механике и проектировании конструкций. Подчёркнуто, что при применении компьютеров самыми распространёнными являются числовые, символические и эвристические методы, а и методы "искусственной интеллигенции". Напоминается ещё и о разных графических изображениях, о которых в работе не упоминается особо. Подчёркивается, что чаще всего употребляются числовые методы, но применение и остальных методов быстро растёт и увеличивается с каждым годом.

J. Dvornik

Übersichtsarbeit

## Numerische, symbolische und heuristische Methoden

Vorgestellt ist ein Überblick über die Anwendung von Rechneranlagen, mit Schwerpunkt auf der Anwendung in der technischen Mechanik und dem Entwurf von Konstruktionen. Es wird hervorgehoben dass dass in der Anwendung von Rechneranlagen grösstenteils numerische, symbolische und heuristische Methoden vertreten sind, aber auch Methoden der "künstlichen Intelligenz". Angeführt sind verschiedene graphische Darstellungen die im Artikel nicht besonders erläutert werden. Es wird hervorgehoben dass noch immer meistens numerische Methoden angewendet werden, doch wächst die Anwendung der anderen Methoden sehr schnell und erweitert sich von Jahr zu Jahr.

Autor: Prof. dr. sc. **Josip Dvornik**, dipl. ing. grad., Građevinski fakultet Sveučilišta u Zagrebu, Kačićeva 26

## 1 Uvod

Početak šire upotrebe elektroničkih računala u svijetu je izazvao dvije suprotne reakcije, a obje su bile ekstremne. I mi smo u Zagrebu doživjeli to isto iako možda u nešto manjim razmjerima.

S jedne strane mnogi, posebno mladi ljudi su se oduševili. Bilo je na početku u tome i mnogo pretjerivanja. Novine su euforično pisale o "elektronskim mozgovima", koji su se prema njihovim opisima mogli uspoređivati s ljudskim mozgovima, ali su mnogo brži i ne mogu pogriješiti. Uskoro, kad je nešto veći broj ljudi pomalo naučio raditi s računalima, pa su ona prestala biti tako velika nepoznanica, postalo je svima očito da su navedene usporedbe s kompjutora s ljudskim mozgom pretjerane i besmislene. Tada su se pojavile prognoze da će "u skoroj budućnosti" računala postati doista inteligentna. Sadašnja inteligencija računala je po nekim današnjim procjenama još uvijek jedva između razine gliste i lignje. Neki stručnjaci ne prihvaćaju čak ni to. Procjena ovisi o izboru pokazatelja kojima se "mjeri" inteligencija. Ali i bez velike vlastite inteligencije pojava računala je bila veliki korak u razvoju čovječanstva, koji se po važnosti može usporediti primjerice s uvođenjem pisma. Računala enormno unapređuju ljudske mogućnosti u smislu brzine i kapaciteta, ali su za sada ipak samo alat. Kompjutor doista može raditi mnoge stvari mnogostruko brže i točnije od čovjeka, ali uvijek samo po zadanim pravilima a (još ?) ne može samoinicijativno nešto predložiti niti uvesti za sebe nova poboljšana pravila. Istina je da postoje programi koji mogu mijenjati čak i svoja pravila, ali samo prema zadanim "meta-pravilima" – pravilima za promjenu pravila koja su ipak konačno napisali ljudi. Ne bismo se usudili proricati budućnost i tvrditi da se jednog dana, možda i prije nego što očekujemo, neće pojaviti doista inteligentni strojevi.

S druge se strane pojavio među inženjerima, posebno starijima, otpor koji bi se iako mnogo blaži, mogao usporediti s pobunom radnika protiv uvođenja tkalačkih i drugih strojeva na početku industrijske revolucije u Engleskoj. Treba ipak priznati i činjenicu da je otpor protiv računala obuhvatio samo manji dio populacije, ali mnogo višeg stupnja naobrazbe. Mnogi matematičari i inženjeri – uglavnom stariji, a među njima i mnogi uvaženi stručnjaci i profesori, nisu razumjeli o kakvim se promjenama radi, pa su tvrdili da su kompjutori samo kratkotrajna moda koja će uskoro nestati. Kasnije su se i oni uvjerali da je ta moda traje mnogo dulje nego što su mogli očekivati. Računala su se u međuvremenu nevjerojatno usavršila, pa ni danas još nema znakova skorog izlaska iz mode. Baš naprotiv. Mnogi su od ranijih protivnika konačno prihvatili novu tehnologiju, a poslije su i zaboravili što su na početku govorili.

## 2 Primjena računala u tehničkoj mehanici i inženjerstvu

Računala su prodrla u sva područja ljudske djelatnosti i danas teško možemo zamisliti svijet bez njih. Imaju primjenu u medicini, bankarstvu, sportu, filmu i svim granama znanosti, tehnike i administracije. Nema ni smisla pokušavati nabrojiti sve.

Ovdje će nas najviše zanimati primjene u matematici i tehnici, a posebno u tehničkoj mehanici i projektiranju konstrukcija. Navest će se tri grupe metoda koje imaju primjenu u tim područjima. To su:

- Numeričke metode
- Simboličke metode
- Heurističke metode i "umjetna inteligencija"

## 3 Numeričke metode

Najveća primjena računala u inženjerstvu odnosi se na numeričke metode.

Numeričke su se metode počele sustavno razvijati još u antičkoj Grčkoj, i nastavile kroz cijelu kasniju povijest civilizacije. Od mnoštva velikih matematičara i fizičara koji su se uz ostala područja bavili i numeričkim metodama navodimo samo nekoliko odabranih imena onih koji nisu bili specijalisti za numeričke metode: Euklid, Arhimed, Fibonacci, Newton, Fermat, Descartes, Gauss, Euler, Pascal, Lagrange, Fourier, Rayleigh, Poincaré, Ljapunov, Courant... Navodimo također samo nekoliko nematematičara također dali veliki doprinos: Bairstow (aerodinamičar), Seidel (astronom), Richardson i Lorenz, (meteorolozi), Aitken (statističar), Pareto (ekonomist) ... Njihove rezultate su kasnije preuzeli matematičari i dalje ih usavršili, dokazali i generalizirali. Izvorni autori ne bi u novim interpretacijama više mogli ni prepoznati vlastite ideje.

Treba spomenuti i doprinose članova Zavoda za tehničku mehaniku Građevinskog fakulteta Sveučilišta u Zagrebu koji se odnose na metode primijenjene u tehničkoj mehanici: Timošenka (Timošenkov kvocijent), Čališeva (Iterativna metoda za rješavanje okvirnih konstrukcija), Wernera (metoda Werner – Csonka), Kanija (Kanijeva metoda)...

Time nisu ni približno nabrojena imena svih koji su svojim radovima zaslužili da ih se navede.

S pojavom kompjutora numeričke su metode dobile novi impuls. Mnoge metode poznate od ranije, na računalu su postale mnogo preciznije i brže, pa su se i znatno usavršile i danas se mogu upotrijebiti za mnogo složenije i opsežnije probleme nego prije.

Primjerice, iako je Gaussov algoritam eliminacije za rješavanje sustava linearnih jednadžbi bila dobro poznata

metoda među matematičarima i inženjerima, za čovjeka oboružanog samo papirom, olovkom i kalkulatorom bio je problem riješiti sustav od desetak linearnih jednadžbi, a rješenje sustava od dvadesetak jednadžbi je bio pravi podvig. Čudili smo se kad je Modor u svom doktoratu riješio nekoliko sustava od 24 linearne jednadžbe bez upotrebe kalkulatora. Nelinearne jednadžbe su bile još daleko veća poteškoća.

Danas uz pomoć računala ni sustavi od nekoliko milijuna nelinearnih jednadžbi ne predstavljaju nesavladivu prepreku. Ne znamo koliki je trenutni rekord, ali sjećamo se da je NASA pred tridesetak godina najavila do kraja dvadesetog stoljeća numeričko rješenje sustava od oko milijardu jako nelinearnih jednadžbi, nastalih modeliranjem strujanja zraka oko avionskog krila. Nepoznanice tog sustava su istovremeno diskretizirane veličine koje opisuju turbulentno strujanje zraka, ali i one koje opisuju prisilne oscilacije avionskog krila od dinamičkog opterećenja turbulentnim strujanjem. Oscilacije krila i strujanje zraka su međusobno zavisni, pa jednadžbe treba rješavati kao jedan sustav. Ne znamo je li NASA-ino predviđanje ostvareno, ali ako i nije to će se bez sumnje uskoro dogoditi. Ipak, nelinearne jednadžbe su vrlo raznolike, pa ni danas ne postoje univerzalne metode kojima bi se mogli rješavati svi tipovi jednadžbi.

Bilo je metoda koje su se u predkompjutorsko vrijeme razvijale u teoriji ali su bile zbog opsežnih proračuna neprimjenjive. Računala su omogućila da se mnoge od njih uspješno realiziraju. Među takvim metodama osobito je veliku primjenu u inženjerstvu doživjela metoda konačnih elemenata (MKE). Prva ideja je, koliko je poznato, potekla od Couranta, ali se u to vrijeme mogla primijeniti samo na nekoliko jednostavnih i po opsegu malih školskih primjera. Ta se metoda danas primjenjuje na sve linearne i nelinearne inženjerske probleme matematičke fizike. U tehničkoj mehanici i konstruktorskom inženjerstvu rješavaju se tom metodom vrlo složeni problemi statike, dinamike, stabilnosti i optimalizacije. Pomoću računala se generiraju modeli različitih konstrukcija i opterećenja, provodi proračun, a na kraju daje grafički prikaz i provjera rezultata, dimenzioniraju se armiranobetonske, čelične i drvene konstrukcije, te provjeravaju naprezanja i deformacije u temeljnom tlu i stijeni. Može se uvažiti bilo koji od poznatih nelinearnih statičkih i dinamičkih modela: geometrijska i materijalna nelinearnost raznih tipova: veliki pomaci i velike deformacije, nelinearna elastičnost, plastičnost, visko-elastičnost, viskoplastičnost, mehanika loma, kontaktni problemi itd.

Analitičkim istraživanjima, numeričkim eksperimentima i inženjerskim iskustvom je utvrđeno da ta metoda usprkos velikih prednosti ima teorijskih i praktičnih nedostataka, pa se osim daljih poboljšanja MKE, u novije vrije-

me razvijaju i mnoge alternativne metode. Primjer su "bezmrežne metode" koje se razvijaju i u Zavodu za tehničku mehaniku Građevinskog fakulteta Sveučilišta u Zagrebu (dr. K. Fresl) Nove metode već sada daju za neke pojave teorijski bolja rješenja, ali još ne mogu konkurirati MKE u svakodnevnoj inženjerskoj primjeni.

Numeričke metode, a posebno MKE još uvijek od nekih pojedinaca – znanstvenih puritanaca doživljavaju kritiku kao "neznanstvene" u usporedbi s analitičkim i eksperimentalnim. Sličan se stav provlači kroz gotovo cijelu povijest. U staroj Grčkoj se prezirala svaka primjena znanosti, posebno matematike, a filozofi i znanstvenici su tretirali inženjerstvo kao manje vrijedni praktični obrt. Oni čak ni numeričke metode nisu razvijali zbog primjene. Naš je stav da se usprkos protivljenju ne trebamo sramiti numeričkih metoda, kad znamo da su ih prihvaćali Newton i Gauss, a i praktički svi današnji znanstvenici i matematičari. Inženjerima, naprotiv, nije baš jako važno jesu li numeričke metode znanstvene ili obrtničke, jer uz njihovu pomoć možemo korektno riješiti daleko više inženjerskih problema nego s bilo kakvim drugim metodama. Često se može i dokazati ispravnost takvih rješenja. Numeričke metode omogućuju rješavanje čak i spoznajnih problema npr. kaotičnog odgovora u dinamici konstrukcija.

#### 4 Simboličke metode

Treba reći da su računala potaknula i renesansu analitičkih i simboličkih metoda u čiju znanstvenost ne sumnja ni najokorjeliji protivnik računala. Idejama automatske primjene simboličkih i analitičkih metoda su se bavili već Babbage i Turing prije pojave elektroničkih računala. (Babbage je doduše proizveo kompjutor, ali mehanički, s kojim se nije moglo baš mnogo računati zbog neprekidnog kvarenja). Već uskoro nakon uvođenja računala pojavio se simbolički programski jezik LISP uz čiju pomoć su se mogli rješavati simbolički problemi. Ubrzo zatim se pojavio i logički jezik PROLOG, a nakon toga još veliki broj drugih simboličkih i logičkih jezika. LISP je danas još uvijek u intenzivnoj upotrebi, dok je PROLOG u najnovije vrijeme zamijenjen novim još jačim, ali i još apstraktnijim i složenijim logičkim programskim jezikom *Gödel*.

Neće se ovdje nabrajati svi programski jezici kojima se mogu rješavati simbolički problemi, ali treba navesti da danas postoje i vrlo razvijeni matematički paketi, od kojih se ističu *Mathematica*, *Maple* i *MACSYMA*. Ti paketi sadrže veliki broj matematičkih funkcija, algoritama i transformacija. Uz pomoć tih paketa može se tražiti pojednostavljenje matematičkih izraza na više načina, analitički derivirati, rješavati određeni i neodređeni integrali te obične i parcijalne diferencijalne jednadžbe,

limesi, Taylorovi redovi i još mnogo toga. Ako se nešto ne može riješiti analitički, bilo zbog toga što je to načelno nemoguće bilo zbog toga što neki postupci još nisu implementirani u programski paket, može se odmah preći na numerički proračun. Prije pojave tih paketa često je važni i vremenski zahtjevni dio posla kod izrade magistarskih radova i doktorata iz matematike, prirodnih i tehničkih znanosti sadržavao međusobno množenje ili potenciranje polinoma, supstituciju matematičkog izraza umjesto varijable u nekom drugom izrazu, deriviranje i integriranje složenih funkcija i slično. Mala pogreška često je prouzročila pogrešan rezultat, pa se isti postupak trebao ponavljati i provjeravati nekoliko puta. Upotrebom tih paketa to su postale rutinske i automatske operacije koje se mogu napraviti brzo i bez pogrešaka – osim onih ljudskih u upisu podataka i mogućih (opet ljudskih) "bugova" u samom paketu. Paket *Mathematica* se već godinama upotrebljava na zagrebačkom Sveučilištu, pa i u našem Zavodu, pa ćemo se na njega detaljnije osvrnuti.

Kao primjer, ja sam u svom magistarskom radu pred više od trideset godina rješavao problem izbočenja pravokutne gipke ploče s početnom imperfekcijom. To sam radio numerički pomoću dvostrukih Fourierovih redova, a koeficijent svakog člana reda se također određivao pomoću dvostrukih Fourierovih redova. Za taj postupak mi je u ono vrijeme trebalo nekoliko mjeseci rada s olovkom i papirom, premda sam i tada numerički proračun izradio pomoću računala. Prije nekoliko godina pokušao sam zbog radoznalosti isti postupak ponoviti uz pomoć paketa *Mathematica*. Taj sam posao ponovio za samo nekoliko sati. Usporedba naravno nije sasvim korektna jer sam u ponovljenom postupku već znao što treba dobiti kao rezultat. Procijenjujem da bih i bez paketa *Mathematica* to sada napravio za desetak dana – ako bih imao sreće, pa ne bi mnogo griješio. Ipak i uz tu procjenu se vidi da je ušteda vrlo velika. Uz to iz istih podataka se vrlo brzo mogu dobiti grafički prikazi rezultata u raznovrsnim oblicima.

Transformacije i funkcije koje nisu ugrađene u paket *Mathematica*, teorijski upućeni korisnik može sam ugraditi u svoj program jer je paketu pridružen vrlo djelotvorni simbolički kompjutorski jezik, koji sadrži mnoge numeričke, grafičke i logičke mogućnosti. Taj jezik ima vrlo mnogo gotovih naredbi visoke razine tako da se duljina programa koji treba korisnik sam napisati svodi gotovo na teorijski minimum. Mogu se osim toga nabaviti i mnogi gotovi programi koje su izradili razni korisnici u svijetu.

Iako ne postoji formalni dokaz da nova verzija paketa *Mathematica* može riješiti sve analitički rješive integrale, nije do sada nađen ni jedan protuprimjer, premda su prema navodima autora paketa – firme Wolfram

Research, ispitani svi primjeri iz svih poznatih i dostupnih zbirki integrala i još mnogo integrala – generiranih na računalu "obratnom metodom" – deriviranjem složenih izraza i različitim transformacijama. Naravno, kako znamo, za mnoge integrale se može dokazati nemogućnost analitičkog rješenja elementarnim funkcijama. Oni se moraju rješavati razvojem u beskonačne redove ili numerički.

Navedenim se paketima mogu rješavati obične i parcijalne diferencijalne jednadžbe, za koje do sad nije bilo poznato rješenje. Primjerice, u zadnjih desetak godina je otkriveno najmanje dvadesetak do nedavno nepoznatih analitičkih rješenja Plateauovog problema minimalne plohe. Minimalna ploha je oblik koji bi poprimila opna od sapunice razapeta na zadanu prostornu krivulju. Taj se problem rješava pomoću Lagrangeove nelinearne parcijalne diferencijalne jednadžbe ili pomoću ekvivalentnog postupka minimizacije funkcionala. Nije se ni slutilo postojanje tolikog broja analitičkih rješenja. Bez računala taj bi posao bio mnogo teži.

Takodjer se u posljednje vrijeme razvijaju programi za automatsko dokazivanje matematičkih teorema i postignuti su već mnogi zanimljivi rezultati. U program se upisuju aksiomi koje smije upotrijebiti. Osim toga zadaje se hipoteza koju treba dokazati ili oboriti. Korake dokaza računalo ispisuje u jeziku razumljivom matematičarima, pa ih svaki specijalist za konkretno područje može provjeriti. Naravno događa se da neku tvrdnju program ne može ni dokazati ni oboriti. Iako će napretkom softvera i napretkom matematike kao znanosti takvih "neodlučenih" slučajeva biti sve manje, posve je sigurno da ih neće nikad posve nestati. Gödel je dokazao da postoje neodlučive tvrdnje koje se u načelu ne mogu (i nikada neće moći) matematičkim metodama ni dokazati ni oboriti. Najteže dokaze ipak će još dugo (možda i uvijek) morati provoditi ljudi sa svježim idejama, ali će im računala višestruko ubrzati i olakšati rad.

Do sad je ponovljeno mnogo poznatih matematičkih dokaza, primjerice klasičnih teorema iz euklidske planimetrije i stereometrije, a već su u "suradnji" čovjeka i računala dokazani neki do nedavno nedokazani teški matematičkih teorema.

Treba na kraju priznati da je za učinkovito programiranje u jeziku *Mathematica* – tako da se iskoriste njegove prednosti, potrebno mnogo više znanja iz teorije koja se upotrebljava i više vještine programiranja nego za primjenu standardnih proceduralnih kompjutorskih jezika. No uloženi trud se višestruko isplati. Kad se objavio taj paket bilo je komentara među našim kolegama: "Zašto učiti naredbe paketa *Mathematica* kad sve piše u knjigama?" Odgovor bi mogao biti protupitanje: "Zašto učiti sintaksu, gramatiku i riječi stranog jezika kad sve to piše u knjigama?"

## 5 Heurističke metode i "umjetna inteligencija"

Kao treća "grana" razvijaju se heurističke metode i metode tzv. "umjetne inteligencije", koje već sada imaju zanimljivu i vrlo uspješnu primjenu u mnogim područjima tehnike, ekonomije, biologije, ratnih operacija. U našem Zavodu su se upotrebljavale neke heurističke metode, ali ne i metode "umjetne inteligencije", ali ipak će se dati pregled tih metoda bez pretenzije na potpunost. Naziv "umjetna inteligencija" zbunjuje većinu ljudi jer je neadekvatan i preoptimističan. Vjerojatno je nastao u nadi da će se te metode uskoro razviti u nešto slično pravoj ljudskoj inteligenciji.

Primjene su npr. razumijevanje govora i pisma, prepoznavanje slika, automatsko prevođenje, orijentacija u prostoru, automatsko upravljanje vozilima i strojevima, medicinska dijagnostika, nalaženje rudnih ležišta iz morfoloških podataka o terenu, optimalizacija raznih sustava itd.

Općenito, heurističke metode i metode umjetne inteligencije su osobito prikladne za rješavanje "mutnih" problema, koji se ne mogu dobro matematički formulirati, kao što je primjerice medicinska dijagnostika i nalaženje potencijalnih rudnih ležišta iz podataka o konfiguraciji terena. Prikladne su i za dobro definirane probleme koji bi se u načelu mogli rješavati matematičkim metodama, ali se od toga mora odustati zbog nedostatka ili nepouzdanosti raspoloživih podataka.

Druga važna klasa problema za koje se intenzivno primjenjuju metode umjetne inteligencije su problemi opterećeni tzv. "kombinatoričkom eksplozijom". To su problemi za koje je poznat egzaktan matematički algoritam, ponekad je čak i jednostavan, ali bi njegova primjena kad je broj nepoznanica velik, zahtijevala neostvarivo mnogo vremena. Primjer kombinatoričke eksplozije je igranje šaha. Nije posebno teško napisati na nekom programskom jeziku algoritam koji bi pretraživanjem svih varijanata do kraja partije egzaktno odredio najbolji potez, ali realizacija takvog algoritma nije moguća u stvarnosti, zbog golemog broja varijanata koje bi trebalo istražiti, koje mnogostruko nadmašuju mogućnosti bilo kojeg računala. A treba priznati da šah po broju podataka i mogućih varijanata ne predstavlja osobito veliki problem - postoji samo 64 polja i 32 figure. Problemi koji se pojavljuju u drugim djelatnostima, npr. u automatskom projektiranju, često su veći za mnogo redova veličine. U literaturi se navode primjeri koji po formulaciji izgledaju prilično bezazleno, ali bi za njihovo egzaktno rješavanje na nekom budućem računalu mnogo djelotvornijem od današnjih trebalo mnogostruko više vremena od sadašnje starosti svemira. Naravno da se mora odustati rješavanja takvog problema pretraživanjem svih mogućnosti.

Metodama umjetne inteligencije nastoji se eliminirati pretraživanja za koja se približnim rezoniranjem može zaključiti da vjerojatno ne sadrže optimum. Tako se dobivaju rješenja u prihvatljivom vremenu, ali se ne može dokazati da su "apsolutno" najbolja. Samo se može tvrditi da su tako dobivena rješenja s vrlo velikom vjerojatnošću mnogo bolja od rješenja koja bismo mogli postići bez primjene tih metoda. Ako opet uzmemo primjer iz šaha, program će odrediti potez koji samo slučajno može biti egzaktno najbolji, ali je najbolji koji se može odrediti na raspoloživom kompjutoru odabranim programom u raspoloživom vremenu. Ni šahovski velemaistor ne može jamčiti da je njegov potez apsolutno najbolji – osim u slučajevima kad je rješenje jednostavno – primjerice kad je moguć forsirani matni napad ili pat te često u završnici kad je jako reducirana broj figura na ploči.

Naši problemi u tehničkoj mehanici i projektiranju konstrukcija su u pravilu jasno definirani pa metode umjetne inteligencije susrećemo u vezi s problemima kombinatoričke eksplozije. Takvi se problemi najčešće susreću u rješavanju velikih nekonveksnih i diskretnih problema optimalizacije, te slabo definiranih problema optimalizacije sa više ciljeva. U slučaju kontinuiranih i konveksnih problema mnogo su efikasnije klasične matematičke metode, primjerice gradijentne.

Navest će se nekoliko heurističkih metoda i metoda umjetne inteligencije upotrebljivih za široku klasu problema optimalizacije, bez pretenzije na potpunost, su:

### Metode Monte Carlo

Umjesto sustavnog pretraživanja čitavog područja definicije problema, pretražuju se samo "slučajno" odabrane točke u tom području, pa se traži optimum među tim točkama. ("Slučajno" na računalu najčešće znači kvazislučajno što se postiže pomoću determinističkih algoritama "generatora slučajnih brojeva". Generator slučajnih brojeva asociira na kockarnicu – odatle je i naziv *Monte Carlo*) Često se tako odabrane točke upotrebljavaju kao početni uvjeti za lokalnu matematičku optimalizaciju. Postoji mnogo varijanata metoda *Monte Carlo*. neke od njih se zasnivaju na slučajnim perturbacijama postignutih lokalnih optimuma u tijeku dotadašnjeg pretraživanja. Zapravo veliki broj metoda umjetne inteligencije upotrebljava metodu *Monte Carlo* u trenutku donošenja odluka.

### Ekspertni sustavi

Te se metode služe heurističkim pravilima kojima oponašaju odluke živog eksperta. Pravila se nalaze u suradnji s više stručnjaka, a mogu se i automatski određivati drugim metodama. Ekspertni sustavi su se pokazali djelotvornima u medicinskoj dijagnostici, geološkim prognozama i mnogim drugim primjenama.

*Neuralne mreže*

Sastoje se od sustava čvorova i veza između njih čime se nastoji simulirati funkcioniranje ljudskog mozga. U većini slučajeva nisu realizirani hardverski nego softverski. To su posebni programski paketi. (ali razvijaju se i posebna hardverska neuralna računala koja će kad dođu do praktične primjene moći raditi mnogo brže) Neuralne mreže se ne programiraju nego uče na primjerima. "Pokazuju" im se primjeri i rješenja tih primjera, pa računalo može automatski generirati empirijska pravila. Prije upotrebe takvi programi moraju "trenirati". Područje djelotvorne primjene neuralnih mreža je približno jednako području primjene ekspertnih sustava: medicinska dijagnostika, geološke prognoze prepoznavanje objekata itd.

*Fuzzy logika*

Postoje sustavi s tzv. "mutnom" logikom. Po klasičnoj logici neka smisljena tvrdnja može biti ili istinita ili lažna. Prema mutnoj logici tvrdnja nije posve istinita ni posve lažna, nego joj se može pripisati "stupanj istinitosti". Npr. neki predmet nije ni potpuno svijetao ni potpuno taman nego je 40% svijetao i 60% taman. Uvedena su pravila za baratanje s tom logikom, koja predstavljaju poopćenje normalne Booleove algebre. Neki se problemi mnogo brže i lakše rješavaju pomoću takve formulacije nego pomoću klasične logike. Ta se metoda pokazala kao vrlo uspješna, posebno za probleme automatskog upravljanja u realnom vremenu, kad je potrebno vrlo brzo približno rješenje. Primjeri su automatsko određivanje potrebne ekspozicije filma na foto aparatu kad je jedan dio slike u mraku a drugi dobro osvijetljen, ili prilagođavanje rada semafora trenutnoj situaciji u prometu. Kad se primjenjuje na optimalizaciju mutna se logika često primjenjuje u kombinaciji s drugim determinističkim i probabilističkim algoritmima.

*Genetički algoritmi*

Genetički algoritmi su inspirirani Darwinovom teorijom prirodne selekcije: U "nultom koraku" se pomoću generatora slučajnih brojeva generira "populacija" potencijalnih rješenja koja zadovoljavaju propisana ograničenja i koja se nazivaju "genomi". Između njih se odabere podskup odabranog broja najboljih. Najbolja rješenja su ona koja imaju najmanju vrijednost funkcije cilja. Svako od tih rješenja je definirano odabranim brojem "gena" U sljedećim koracima se generiraju "potomci" koji od svakog roditelja nasljeđuju dio genetskog koda. Kod toga ne treba doslovno kopirati prirodu pa svako "dijete" može imati i više od dva "roditelja". Također za razliku od prirode, rješenja koja su dovoljno dobra ne stare i ne umiru, nego mogu "živjeti" vječno. Unija skupa "roditelja" i skupa "potomaka" čini novi zajednički skup iz

kojega se opet odabire podskup najboljih. Osim "križanja" predviđene su i "mutacije" - slučajne promjene vrijednosti nekog gena. To je uvedeno zbog toga da se u konačnom rješenju omogući pojava i onih gena koje nema ni jedan od roditelja, a koji mogu dovesti do boljeg rješenja. Radi još veće raznolikosti mogu se uključiti i novi slučajni genomi koji nisu dobiveni modifikacijom starih. Oni se zovu "imigranti". Algoritam nema definiran kraj, nego se postupak može uvijek nastaviti. Ipak nakon izvjesnog broja koraka nova poboljšanja postaju vrlo rijetka i numerički gledano malena, pa se postupak obično prekida po nekom kriteriju. Genetički algoritmi se s velikim uspjehom primjenjuju za mnoge vrste problema, a osobito kombinatoričke optimalizacije, a čak i za probleme diskretizirane kontinuirane optimalizacije.

Kao i u drugim metodama umjetne inteligencije, još brže se dobivaju dobri rezultati "nečistim" postupkom u kojem se genetski algoritam kombinira s nekom od klasičnih tehnika optimalizacije: Svako od rješenja dobiveno u tijeku genetskog algoritma postaje početni uvjet za lokalni postupak klasične matematičke optimalizacije, primjerice gradijentnom metodom.

Lokalni optimumi dobiveni tim postupkom ponovo ulaze u novi korak genetskog algoritma. Dakle "djeca" se generiraju "križanjem" "roditelja" i optimalizacijom.

*Evolucijsko programiranje*

Zanimljivo je da se postupak sličan genetskom algoritmu može primijeniti i na izradu kompjutorskih programa. "Geni" u tom programu su različite programske naredbe. Na početku se generiraju slučajni nizovi naredbi uz provjeru sintakse – programi. Program koji za više zadanih setova ulaznih podataka dobiva više ispravnih rezultata se ocjenjuje kao bolji. Među programima se provode postupci "križanja" i "mutacije" te se odabere skup najboljih koji preživljavaju.

Istovremeno se provodi slična prirodna selekcija među setovima ulaznih podataka. Najbolji je set ulaznih podataka je onaj koji je najkritičniji, odnosno onaj na kojemu najviše programa daje pogrešne rezultate, pa ih se može eliminirati.

Ako neki program zadovoljava sve takve evoluirane setove podataka, znači da prolazi test.

Tako u toku "evolucije" programi zadovoljavaju sve više sve strožih testova. Na kraju procesa preostaju samo programi koji su ispravno riješili sve testove, pa se konačni pobjednik određuje između njih po nekom drugom kriteriju, npr. brzini. Naravno, nikada nije posve sigurno da ne postoji neki neotkriveni test na kojemu bi taj pobjednik "pao na ispitu". No i programima koje su na-

pisali ljudi se ponekad i nakon više godina ispravnog rada pojavi pogrešno rješenje za neki skup ulaznih podataka.

Vrijeme evolucijskog programiranja je za neke tipove problema mnogo kraće nego pisanje konvencionalnog programa koji rade programeri, ali su programi gotovo nečitljivi za ljude.

### *Simulirano kaljenje*

Taj je algoritam inspiriran poboljšanjem svojstava metala preslaganjem atoma na visokoj temperaturi za vrijeme kaljenja. Počevši od slučajnog ili drugim metodama nađenog inicijalnog rješenja pokušava se slučajnim varijacijama tog rješenja naći bolje. Pri tome se dopušta i pogoršanje u pojedinim koracima, da bi se izišlo iz lokalnog minimuma. U tijeku postupka se postepeno snizuje "temperatura". Matematičkim rječnikom simulirano kaljenje se može opisati kao nestacionani Markovljev lanac u diskretnom vremenu. Najnovije varijante algoritma automatski popravljaju parametre optimalizacije za vrijeme proračuna prema "iskustvu" iz dosadašnjeg tijeka procesa.

### *Tabu algoritam*

Jednostavni algoritmi probabilističkog i heurističkog pretraživanja dopustivog područja (*Monte Carlo*, simulirano kaljenje itd.) često ne konvergiraju zbog pojave višestrukog pretraživanja već pretraženih dijelova dopustivog područja te se pojavljuju beskonačni ciklusi. Tabu algoritam sprema podatke o povijesti već obavljenog pretraživanja, pa ne dopušta ponovno posjećivanje istih dijelova područja. Na taj se način istražuje uvijek novi dio područja, zbog čega raste vjerojatnost nalaženja globalnog optimuma. Taj se algoritam često i kombinira s drugim metodama.

### *Mravlja kolonija*

Mravi, termiti, pčele i ose imaju svojstvo tzv. "*kolektivne inteligencije*". Zadržimo se na primjeru mrava. Mravi svakog dana izlaze iz mravinjaka u potrazi za hranom. Prva skupina mrava su "izviđači". Oni lutaju nasumce i ostavljaju kemijski "*feromonski trag*" po kojemu ih ostali mogu slijediti. Onaj mrav koji prvi slučajno nađe hranu vraća se u mravinjak udvostručujući svoj trag. Ostali mravi kreću po tom tragu i dodatno ga pojačavaju svojim tragom. Oni mravi iz prve skupine koji nisu našli hranu ili su ju našli daleko od mravinjaka vraćaju se kasnije. Njihov je trag slabiji jer po njemu još nisu išli drugi mravi. S vremenom sve više mrava ide po najjačem tragu do najboljeg nalazišta hrane. Neki mravi u međuvremenu otkriju kraći put do cilja koji po istom algoritmu prihvaćaju i ostali. Tragovi s vremenom ispa-

ruju, pa se lošiji putovi kojima ide manje mrava postepeno "zaboravljaju". Tako kolonija mrava zajednički nalazi optimalni put.

Ovaj algoritam je najprije vrlo uspješno simuliran na računalu. Kasnije je i usavršen napuštajući analogiju s mravljom kolonijom, jer kopiranje mrava nije bilo cilj. U najnovije je vrijeme još i poboljšana u kombinaciji s genetskim algoritmom, a zatim još i matematičkim metodama.

### *Čovjek u petlji*

Mnoge od navedenih i drugih metoda optimalizacije se mogu još više unaprijediti i ubrzati, ako se omogući da čovjek donosi odluku na kritičnim mjestima u algoritmu. Stručnjak može iskoristiti svoje znanje, iskustvo i intuiciju da spriječi pretraživanje onih dijelova područja u kojima je manja mogućnost postizanja optimuma.

Postoji još i mnogo algoritama koji nisu spomenuti i mnogo mješovitih algoritama koji kombiniraju klasične metode i metode umjetne inteligencije.

Postoje i posebni heuristički postupci koji su posebno razvijeni za optimalizaciju konstrukcija. Spomenut će se ovdje samo jedan:

#### *Potpuno iskorištenje naprezanja (Fully stressed design).*

Možda je naivna, ali ipak često ispravna tvrdnja, da su u dobrom konstruktivnom rješenju naprezanja u svim točkama visoko i podjednako iskorištena. Najčešće to iskorištenje nije istovremeno nego doseže maksimum u raznim točkama konstrukcije za različite slučajeve opterećenja. Moglo bi se reći da ta ideja optimalizacije potiče od ideje dopuštenih naprezanja, ali to ne znači da se i kasnija analiza konstrukcije mora provoditi po kriteriju dopuštenih naprezanja. Obratna zamisao da je konstrukcija s potpuno iskorištenim napreznjima optimalna je također približno ispravna, pa može poslužiti kao baza za heurističku optimalizaciju.

Tok algoritma je slijedeći: Započinje se od odabranog rješenja konstrukcije koje zadovoljava sva ograničenja, ali nije optimalno. Proračunom se nalaze neiskorišteni dijelovi, pa se na tim mjestima dimenzije poprečnog presjeka smanjuju. Ponovni proračun daje drugačiju razdibu naprezanja, pa se prethodno rješenje iterativno korigira. Na kraju tog procesa, ako je proces konvergentan i regularan postiže se potpuno iskorištenje naprezanja. Obično postoji više mogućih rješenja s potpuno iskorištenim napreznjima, pa nije sigurno da će se ovim algoritmom naći najbolje. No svakako će redovito biti mnogo bolje od rješenja bez optimalizacije.

Taj je algoritam u primjeni uvijek složeniji jer neke dimenzije neće smanjiti ispod odabranog minimuma iako



napreznja nisu iskorištena. Pojavljuju se i primjerice problemi stabilnosti u kojima dolaze do izražaja integralna svojstva konstruktivnog elementa (u slučaju lokalnog problema stabilnosti) ili cijele konstrukcije (u slučaju globalnog problema). Dodatna je komplikacija kad se dimenzije presjeka moraju birati iz asortimana, kao što su tablice valjanih čeličnih profila.

Treba istaknuti da se metode umjetne inteligencije nažalost vrlo često (zlo)upotrebljavaju za probleme koje bi bilo mnogo lakše riješiti bez njih. To se često radi čak i (ne)namjerno. Neki zadatak koji se može lako i brzo riješiti poznatim deterministički matematičkim metodama postaje mnogo nerazumljiviji i "znanstveniji" ako se rješava pomoću "umjetne inteligencije". Nije čak uopće važno jesu li rezultati ispravni. Često se to radi pomoć nekog softverskog paketa za onu metodu umjetne inteligencije koja je trenutno u modi, a koji korisnik najvjerojatnije ni ne razumije i tako se proizvede "izvorni znan-

stveni rad". Recenzent obično ne želi priznati da nikad nije čuo za novu metodu koja se primijenila, pa mu je najjednostavnije prihvatiti takav rad. Neki su autori čak metodama umjetne inteligencije rješavali sustave algebarskih jednažbi, što je jedan od problema najprikladnijih za standardne determinističke matematičke postupke a ujedno i izrazito neprikladnih za primjenu umjetne inteligencije.

## 6 Zaključak

Upotrebom kompjutorskih metoda projektanti i istraživači sve se više oslobađaju rutinskog posla, što im ostavlja više mogućnosti i vremena da se posvete kreativnom radu. Može se očekivati da će se u budućnosti pojaviti programi, koji će povezati sve tri paradigme – analitičke, numeričke i heurističke metode – koje će dati novi impuls razvoju znanosti i tehnologije.

*Napomena: Ovaj rad je prijavljen i kao referat za simpozij RAČUNALSTVO U GRADITELJSTVU koji će se održati u prosincu 2003.*

## LITERATURA

- [1] Hofstadter, D. R.: *Gödel, Esher, Bach*, Basic Books, 1979.
- [2] Noor, A. K.; Andersen, C. M.: *Computerised Symbolic Manipulation in Structural Mechanics – Progress and Potential*, Computers & Structures Vol 10. pp 95.-118, Pergamon Press Ltd., 1979.
- [3] O'Shea, T.; Eisenstadt, M.: *Artificial Inteligence*, Harper & Row, 1984.
- [4] Shea, T. O'; Eisenstadt, M.: *Artificial Inteligence*, Harper & Row, 1984.
- [5] Dvornik, J.: *Razvoj matematičkih modela inženjerskih problema*, GRAĐEVINAR 42(1990)12, 507.-516.
- [6] Dvornik, J.: *Razvoj matematičkih modela inženjerskih problema*, GRAĐEVINAR 42 (1990) 12, 507.-516.
- [7] Hinton, E.; Sienz, J.: *Fully Stressed Topological Design of Structures using an Evolutionary Procedure*, Engineering Computations: Int J for Computer-Aided Engineering, Vol.12, No 3, pp.229-244, 1995
- [8] Dorigo, M.; Maniezzo, V.; Colomi, A.: *The Ant System: Optimization by a colony of cooperating agents*, IEEE, Transactions on Systems, Man and Cybernetics – Part B, Vol 26, No. 1, 1996. pp.1-13
- [9] Morey, C.; Scales, J.; Van Vleck, E. S.: *A Feedback Algorithm for Determining Search Parameters for Monte Carlo Optimization*, Journal of Computational Physics 146, 263-281 (1998)
- [10] Arantes e Oliveira, E. R.; Bento, J.: *The Sense of Progress in Structural Engineering*, Development of Knowledge-Based Systems for Engineering, 1998, New-York, pp. 37-58, Springer-Verlag, Wien
- [11] Andersson, J.; Krus, P.: *Metamodel Representations for Robustness Assesment in Multiobjective Optimization*, International Conference on Engineering Design, ICED 01, Glasgow, August 21-23, 2001
- [12] A.H.F. Dias; J. A. de Vasconcelos: *Multiobjective Genetic Algorithms Applied to Solve Optimization Problems*, IEEE Transactions on Magnetics, Vol. 38. No. 2, 2002.
- [13] E. R. de Arantes e Oliveira; Bento, J.: *The Sense of Progress in Structural Engineering*.
- [14] Lucas, C.: *Genetic Algorithms – Nature's Way*" <http://www.calresco.org/genetic.htm>
- [15] C. Lucas: *"Practical Multiobjective Optimization"* <http://www.calresco.org/lucas/pmo.htm>
- [16] SC Patodi: *"Applications of Artificial Intelligence in Structural Engineering"*, <http://www.ncst.ernet.in/kbcs/vivek/issues/12.2/patoairep/patoairep.html>