UDK 528.14:512.25
Pregledni rad

RJEŠAVANJE NORMALNIH JEDNADŽBI I METODA CHOLESKOG

*Damjan JOVIČIĆ, Miljenko LAPAINE, Svetozar PETROVIĆ — Zagreb**

Svrha ovog rada je da kroz nekoliko primjedbi da pregled problema koji su uočeni proučavanjem niza radova koji se bave rješavanjem normalnih jednadžbi. Posebno se razmatra metoda Choleskog, te predlaže takav algoritam i program u kojem osnovna ideja Choleskog dolazi do punog izražaja.

1. PRIMJEDBA

Ako je matrica A regularna, rješenje sistema linearnih jednadžbi

$$Ax = b$$

može se napisati u obliku

$$x = A^{-1} b.$$

U računu izjednačenja to se obično zove neodređeni način rješavanja normalnih jednadžbi. Iako je poznato da osim ovog načina postoji i niz drugih, npr. Gaussova metoda, Banachiewiczeva metoda, metoda Choleskog itd., ipak se može pročitati: »Radi rešavanja normalnih jednačina, *neophodno* je invertovati matricu ...« [7], str. 28 (podvukli autori ovog članka). U istom broju Geodetskog lista Z. Galić predlaže program za rješavanje sistema linearnih jednadžbi metodom Choleskog i kaže: »Za dobivanje vektora rješenja *potrebno je* izvršiti množenje inverzne matrice i vektora slobodnih članova« [3], str. 37 (kurzivom istakli autori ovog članka).

Međutim, očito da invertiranje matrice nije *ni potrebno ni neophodno*, jer kako smo naveli, postoje i druge metode rješavanja sistema linearnih jednadžbi. Što više, Stewart u [9] kaže da je takav pristup »naivan«. Razlog je jednostavan: samo računanje inverzne matrice zahtijeva oko tri puta više računskih operacija od neposrednog rješavanja jednog sistema. Povećani broj računskih operacija s jedne strane produžava vrijeme izvođenja programa, a s druge strane može smanjiti točnost rješenja. Ilustrirajmo ovo prema Forsytheu, Malcolmu i Moleru: »Jednostavan, ali poučan primjer je sistem koji se sastoji od jedne jednadžbe

* Adresa autora: mr Damjan Jovičić, Miljenko Lapaine, dipl. inž., Svetozar Petrović, dipl. inž., Geodetski fakultet Sveučilišta u Zagrebu, Zagreb, Kačićeva 26

$$7x = 21.$$

Najbolji način rješavanja ovog zadatka je dijeljenje

$$x = \frac{21}{7} = 3.$$

Korištenje inverzne matrice dalo bi

$$\begin{aligned} x &= 7^{-1} \cdot 21 \\ &= 0.142857 \cdot 21 \\ &= 2.999997 \end{aligned}$$

Drugi način zahtijeva više aritmetike — dijeljenje i množenje umjesto jednog dijeljenja — i daje netočniji rezultat.« ([2])

Kao odgovor na pitanje: kad je ipak pogodno izračunati inverznu matricu?, Scheid u [8] piše: »Ako treba riješiti nekoliko sistema jednadžbi koji imaju istu matricu A, ali različite vektore b, tada postaje *ekonomično* naći najprije inverznu matricu«. Slično kod Lončara u [6] stoji: »Međutim, postoje ipak slučajevi kada trebamo direktno računati inverznu matricu. To je *očito u slučaju* kod rješenja sistema Ax = b kada je A ista, a mijenja se samo b.« (Kurzivom istakli u oba citata autori ovog rada.)

Međutim, u slučaju kada treba riješiti veći broj sistema s istom matricom, nasuprot izjavama citiranih autora, nije *niti očito, niti ekonomično* određivati inverznu matricu. Dokaz ove tvrdnje izvodi se prebrojavanjem potrebnog broja računskih operacija, a može se naći npr. u [10] str. 93 ili [4] str. 36.

1. ZAKLJUČAK

Inverznu matricu nećemo računati da bismo pomoću nje riješili sistem linearnih jednadžbi, nego jedino onda kada je ona potrebna za neku drugu svrhu. Takav je slučaj u geodeziji gdje se ona koristi za ocjenu točnosti. No niti tada rješenje pripadnog sistema normalnih jednadžbi *nećemo računati pomoću inverzne matrice*, već iz reduciranih jednadžbi poznatom supstitucijom unazad. Na taj način je broj izvršenih računskih operacija nakon kojeg se dolazi do rješenja tri puta manji (što pozitivno utječe na njegovu točnost), a sveukupni broj potrebnih operacija da se dobije i rješenje i inverzna matrica je kod oba postupka jednak.

2. PRIMJEDBA

Cijena koju korisnik plaća prilikom izvođenja nekog programa ovisi o trajanju izvođenja, a ovo pak o broju računskih i drugih operacija. O ovome kao da nije vođeno računa u programu Z. Galića [3] gdje se ne koristi svojstvo simetrije normalnih jednadžbi i svojstvo simetrije metode Choleskog, te se nepotrebno ponavljaju iste računske opracije i svi koeficijenti, osim dijagonalnih, izračunavaju dva puta.

Drugi važan faktor o kojem ovisi cijena izvođenja programa je veličina korištene centralne memorije računala. Osim toga o memoriji naročito treba voditi računa ako imamo na raspolaganju računalo manjeg kapaciteta. Spretno sastavljen program omogućiti će invertiranje matrice i rješavanje sistema većeg broja normalnih jednadžbi. Različitim pristupima može se uštedjeti ili potrošiti vrlo mnogo memorije. Tako npr. za dobivanje inverzne matrice A, Z. Galić u [3] koristi polja A, R i X ukupne veličine $5n^2$, gdje je n broj normalnih jednadžbi. Već je u programu [1] K. Čolića za isti posao korišteno samo jedno polje veličine n^2. U ovom radu će se pokazati da metoda Choleskog omogućava i dalje smanjenje potrebne memorije na približno $n^2/2$, a da se time ne poveća ukupan broj računskih operacija.

2. ZAKLJUČAK

Da bi program bio što efikasniji potrebno je voditi računa o racionalnoj upotrebi memorije računala, a također ne programirati nepotrebna izračunavanja.

3. PRIMJEDBA

Budući da je matrica A normalnih jednadžbi oblika

$$A = B^t P B,$$

lako se može dokazati da je ona nenegativno definitna* (vidi npr. [9], str. 139). Teorija kaže da se zato prilikom primjene metode Choleskog ne može pri računanju dijagonalnih elemenata pojaviti vađenje drugog korijena iz negativnog broja (npr. [11], str. 126). Međutim, praktički se to ipak može dogoditi (vidi npr. [9], str. 225), ako su pogreške zaokruživanja dovoljno velike. No, to može nastupiti onda kada je matrica normalnih jednadžbi praktički, tj. gotovo singularna. Rad s ovakvim sistemima je vrlo osjetljive prirode (vidi npr. [5]).

3. ZAKLJUČAK

Pri dekompoziciji matrice normalnih jednadžbi metodom Choleskog nije potrebno uvoditi računanje s kompleksnim brojevima kao što to čini npr. Z. Galić u [3].*

U slučaju da se pojavi korjenovanje negativnog broja ili dijeljenje nulom, treba postupiti na neki od uobičajenih načina, kao što je npr. izbacivanje inkriminirane jednadžbe koje se temelji na činjenici da singularnost ukazuje na to da sistem nije nezavisan, dakle sadrži višak jednadžbi. Međutim, kako je singularnost često posljedica nekih ranijih grijeha (npr. lošeg izbora uvjeta kod uvjetnih mjerenja), izgleda da je tada ipak najbolja alternativa: prekidanje izvođenja programa uz poruku korisniku da je matrica numerički singularna. Korisnik će tada odlučiti o daljnjem postupku.

---

* U slučaju simetrične matrice koja nije nenegativno definitna metoda Choleskog bez specijalnog pivotiranja zakazuje, ne pomaže niti uvođenje kompleksnih brojeva (vidi [14], str. 643—645).

Pridržavajući se gornjih zaključaka sastavili smo algoritam i program koji se temelje na originalnoj ideji Choleskog, prema kojoj se sve može obaviti u »trokutu« koeficijenata normalnih jednadžbi.

ALGORITAM

Neka je

$$\mathbf{A}\,x = b \tag{1}$$

sistem normalnih jednadžbi. Treba odrediti rješenje x, inverznu matricu A^{-1} i član [pvv]. Sistem normalnih jednadžbi (1) možemo detaljnije prikazati ovako:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{12}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ \ldots \quad \ldots \quad \ldots \quad &\ldots \\ a_{1n}x_1 + a_{2n}x_2 + \ldots + a_{nn}x_n &= b_n \end{aligned} \tag{2}$$

I korak

Sistem (1), odnosno (2), najprije ćemo reducirati u ekvivalentni trokutasti sistem

$$R\,x = y, \tag{3}$$

odnosno detaljnije napisano

$$\begin{aligned} r_{11}x_1 + r_{12}x_2 + \ldots + r_{1n}x_n &= y_1 \\ r_{22}x_2 + \ldots + r_{2n}x_n &= y_2 \\ \ldots \quad \ldots \quad & \\ r_{nn}x_n &= y_n, \end{aligned} \tag{4}$$

gdje je R gornja trokutasta matrica dobivena faktorizacijom matrice A po metodi Choleskog tako da vrijedi

$$A = R^t R,$$

tj. elementi r_{ij} izračunavaju se po formulama

$$r_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} r_{ik}^2} \qquad i = 1, \ldots, n \tag{5}$$

$$r_{ij} = \left(a_{ij} - \sum_{k=1}^{i-1} r_{ik}\, r_{jk}\right) : r_{ii} \qquad j > i \tag{6}$$

Elementi y_i dobiju se istodobno istovrsnom redukcijom, tj. po formulama

$$y_i = \left(b_i - \sum_{k=1}^{i-1} r_{ik}\, b_k\right) : r_{ii} \qquad i = 1, \ldots, n. \tag{7}$$

Istovremeno se može izračunati i član [pvv] i to za posredna mjerenja prema

$$[pvv] = [pll] - \sum_{i=1}^{n} y_i^2, \tag{8}$$

a za uvjetna mjerenja prema

$$-[pvv] = 0 - \sum_{i=1}^{n} y_i^2. \tag{9}$$

Obrazložimo (8):

Neka su

$$Bx + l = v \tag{8a}$$

jednadžbe pogrešaka. Tada $v^t Pv$ = minimum daje

$$2v^t P B = 0$$

odnosno

$$B^t P (Bx + l) = 0$$

što su zapravo normalne jednadžbe

$$B^t P Bx + B^t Pl = 0. \tag{8b}$$

Sada možemo napisati

$$v^t Pv = (x^t B^t + l^t) P (Bx + l) = x^t (B^t P Bx + B^t Pl) + l^t P Bx + l^t Pl.$$

Zbog (8b)

$$v^t Pv = l^t Pl - x^t B^t P Bx.$$

Uz oznaku $A = B^t P B$, faktorizaciju matrice A po Choleskom i (3)

$$v^t Pv = l^t Pl - x^t Ax = l^t Pl - x^t R^t Rx = l^t Pl - y^t y,$$

što je relacija (8) u matričnom prikazu. Uočimo analogiju izraza (8) s formulama (5), osim korjenovanja.

Obrazložimo također i (9):

Neka su

$$Bv + w = 0 \tag{9a}$$

uvjetne jednadžbe. Tada $v^t Pv - 2k^t (Bv + w)$ = minimum daje

$$2v^t P - 2k^t B = 0$$

i odatle

$$v^t = k^t B P^{-1},$$

odnosno

$$v = P^{-1} B^t k.$$

Uvrstimo li ovaj v u (9a) dobijemo normalne jednadžbe

$$B P^{-1} B^t k + w = 0. \tag{9b}$$

Sada možemo napisati

$$v^t P v = k^t B v = - k^t w.$$

Uz oznaku $A = B P^{-1} B^t$, faktorizaciju matrice A po Choleskom i (3)

$$v^t P v = k^t A k = k^t R^t R k = y^t y = - (0 - y^t y)$$

što je relacija (9) prikazana matrično (uz promijenjeni predznak). Uočimo analogiju i izraza (9) s formulama (5), osim korjenovanja.

II korak

Sada iz reduciranog sistema (3), odnosno (4), možemo izračunati nepoznanice supstitucijom unazad, tj. po formulama

$$x_i = \left(y_i - \sum_{k=i+1}^{n} r_{ik} x_k \right) : r_{ii} \qquad i = n, n-1, \ldots, 1 \tag{9}$$

Istovremeno računamo elemente matrice $S = R^{-1}$ po formulama

$$s_{jj} = 1 : r_{jj} \qquad j = n, n-1, \ldots, 1 \tag{10}$$

$$s_{ij} = \left(- \sum_{k=i+1}^{j} r_{ik} s_{kj} \right) : r_{ii} \qquad i < j \tag{11}$$

Uočimo analogiju između (11) i (9).

III korak

Na kraju računamo elemente α_{ij} matrice $A^{-1} = R^{-1} (R^t)^{-1} = S S^t$ po formulama

$$\alpha_{ij} = \sum_{k=j}^{n} s_{ik} s_{jk} \qquad \begin{array}{l} i = 1, \ldots, n \\ j = i, i+1, \ldots, n \end{array} \tag{12}$$

NAPOMENA

Bjerhammar u [12] str. 330—331 opisuje jedan vrlo sličan algoritam koji naziva metodom Cholesky-Rubina. U odnosu na naš upravo opisani algoritam postoji jedna razlika, ali ta je bitna, a sastoji se u slijedećem: do rješenja normalnih jednadžbi ne dolazi se kao kod nas u II koraku prema relaciji (9), nego se računaju elementi matrice $S = R^{-1}$ po formulama (10) i (11) i tek nakon toga određuju nepoznanice prema izrazu

$$x = R^{-1} y.$$

Nedostaci ovog redoslijeda računanja navedeni su u našoj primjedbi broj 1.

OPIS PROGRAMA

Program je napisan u Basicu za stolno računalo HP 9845A. Podaci potrebni za izvođenje programa su: broj normalnih jednadžbi N, koeficijenti »trokuta« normalnih jednadžbi (uključujući desnu stranu, tj. slobodne koeficijente) i član [pll] kod posrednih mjerenja, dok ćemo kod uvjetnih staviti 0. Ove podatke ćemo spremiti u datoteku »ULAZR« na kazeti, koja služi kao vanjska memorija računala. Ukoliko se želi direktno učitavanje podataka pomoću naredbi DATA ili INPUT ili korištenje neke druge vanjske memorije u skladu s mogućnostima datog računala, treba na odgovarajući način modificirati naredbe 80—100, 190, 250 i 270.

Samo izvođenje programa počinje tako da se iz datoteke učitaju podaci u glavnu memoriju računala. Budući da koristimo svojstvo simetrije koeficijenata normalnih jednadžbi, tj. radimo samo s koeficijentima gornjeg trokuta matrice normalnih jednadžbi, a trokutasta polja u Basicu ne postoje, to ćemo sve koeficijente smjestiti u jednodimenzionalno polje A dimenzije (N + 1) (N + + 2)/2, jedan redak za drugim:

$$
\begin{array}{r}
a_{11}\ a_{12} \ldots a_{1n}\ b_1 \\
a_{22} \ldots a_{2n}\ b_2 \\
\ldots\ldots\ldots \\
\ldots a_{nn}\ b_n \\
[\mathrm{p}ll]
\end{array}
$$

U slučaju uvjetnih mjerenja, kao što je već spomenuto, umjesto [pll] stoji 0. Nakon što su podaci učitani, slijedi njihov ispis.

Bitna karakteristika ovog programa je ekonomiziranje memorijom. Opisani algoritam omogućuje da se sva zapisivanja međurezultata vrše u istom polju A. Tako nakon prvog koraka ((5) — (8)) polje A sadrži elemente

$$
\begin{array}{r}
r_{11}\ r_{12} \ldots r_{1n}\ y_1 \\
r_{22} \ldots r_{2n}\ y_2 \\
\ldots\ldots\ldots \\
r_{nn}\ y_n \\
[\mathrm{pvv}]
\end{array}
$$

U slučaju uvjetnih mjerenja na mjestu [pvv] stoji — [pvv]. Nakon drugog koraka ((9) — (11)) polje A sadrži elemente

$$
\begin{array}{r}
s_{11}\ s_{12} \ldots s_{1n}\ x_1 \\
s_{22} \ldots s_{2n}\ x_2 \\
\ldots\ldots\ldots \\
s_{nn}\ x_n \\
[\mathrm{pvv}]
\end{array}
$$

Nakon trećeg koraka (12) polje A konačno sadrži

$$\begin{array}{l} \alpha_{11}\ \alpha_{12} \dots \alpha_{1n}\ x_1 \\ \quad \alpha_{22} \dots \alpha_{2n}\ x_2 \\ \quad\quad \dots\dots\dots \\ \quad\quad\quad \alpha_{nn}\ x_n \\ \quad\quad\quad [pvv] \end{array}$$

Slijedi ispis svih rezultata na papir, odnosno u datoteku »IZLAZR« za eventualnu daljnju obradu. Ako se ne želi zapis rezultata i na kazetu treba iz programa izostaviti naredbe **980, 990, 1020, 1080, 1190** i **1260**.

U slučaju bilo kakvih poteškoća pri implementiranju ovog programa na drugo računalo, svaki potencijalni korisnik se slobodno može obratiti na adresu autora.

U nastavku prilažemo opisani program sa primjerom izvođenja. Zadatak je preuzet iz [13] (Example 3.5).

LITERATURA:

[1] Čolić K.: GIPOM — program u Fortranu za izjednačenje po posrednim mjerenjima, Geodetski fakultet, Zagreb, 1975

[2] Forsythe G. E., Malcolm M. A., Moler C. B.: Computer Methods for Mathematical Computations, Prentice-Hall, Englewood Cliffs, 1977 (ruski prijevod: Mir, Moskva, 1980)

[3] Galić Z.: Program za rješavanje sistema linearnih jednačina metodom Choleskog, Geodetski list br. 1—3, 1983, 31—37

[4] Isaacson E., Keller H. B.: Analysis of Numerical Methods, John Wiley, New York, 1966

[5] Jovičić D., Lapaine M., Petrović S.: Rješavanje normalnih jednadžbi pomoću računala, Zbornik radova 4. susreta geodeta SR Hrvatske, Osijek, 1981

[6] Lončar J.: Numerička analiza, Geodetski fakultet, Zagreb, 1972

[7] Molnar I.: Izravnanje mreža particijom matrica, Geodetski list br. 1—3, 1983, 24—30

[8] Scheid F.: Numerical Analysis, Schaum's Outline Series, McGraw-Hill, New York, 1968

[9] Stewart G. W.: Introduction to Matrix Computations, Academic Press, New York, 1973

[10] Stojaković Z., Herceg D.: Numeričke metode linearne algebre, Građevinska knjiga, Beograd, 1982

[11] Anđelić T. P.: Matrice, Građevinska knjiga, Beograd, 1979

[12] Bjerhammar A.: Theory of Errors and Generalized Inverses, Elsevier Scientific Publ. Comp., Amsterdam, 1973

[13] Gregory R. T., Karney L. K.: A Collection of Matrices for Testing Computational Algorithms, Wiley-Interscience, New York, 1969

[14] Bunch J. R., Parlett B. N.: Direct methods for solving symmetric indefinite systems of linear equations, SIAM J. Numer. Anal., Vol. 8, No. 4, 1971

=======
PRIMJER
=======

NORMALNE JEDNADZBE

1 . redak	5.000	7.000	6.000	5.000	23.000
2 . redak		10.000	8.000	7.000	32.000
3 . redak			10.000	9.000	33.000
4 . redak				10.000	31.000

[pll]= 120.000

[pvv]= 1.000

RJESENJE NORMALNIH JEDNADZBI

1.000
1.000
1.000
1.000

INVERZNA MATRICA

1 . redak	68.000	-41.000	-17.000	10.000
2 . redak		25.000	10.000	-6.000
3 . redak			5.000	-3.000
4 . redak				2.000

=======
PROGRAM
=======

```
1    ! ************************************************************
10   ! Ime programa: CHOL
20   ! Svrha programa: RJESAVANJE N NORMALNIH JEDNADZBI,
30   ! RACUNANJE INVERZNE MATRICE I [pvv]  prema Choleskom
40   ! Autori: M.Lapaine, S.Petrovic, D.Jovicic
50   ! Geodetski fakultet, Zagreb, Kaciceva 26
51   ! ************************************************************
60   OPTION BASE 1
70   REAL A(100)                       !Dimenzija polja A mora biti
80   ASSIGN #1 TO "ULAZR"              !najmanje (N+1)(N+2)/2
90   BUFFER #1
100  READ #1;N                         !N=broj normalnih jednadzbi
110  PRINT "NORMALNE JEDNADZBE"
120  PRINT "------------------"
130  Kd=1
140  FOR I=1 TO N
150  PRINT
160  PRINT I;". redak"
161  FIXED 3
170  PRINT TAB(10*I),
180  FOR J=I TO N+1
190  READ #1;A(Kd)
200  PRINT A(Kd);TAB(10*J+10),
210  Kd=Kd+1
220  NEXT J
230  PRINT
231  STANDARD
240  NEXT I
250  READ #1;A(Kd)
260  PRINT LIN(3),TAB(20),"[pll]=";A(Kd)
270  ASSIGN * TO #1
280  I=1                               !A=Rt*R
290  IF A(1)>0 THEN 330
300  PRINT "MATRICA NORMALNIH JEDNADZBI JE NUMERICKI SINGULARNA"
310  PRINT "DIJELJENJE NULOM U ";I;". RETKU"
320  STOP
330  A(1)=SQR(A(1))
340  FOR I=2 TO N+1
350  A(I)=A(I)/A(1)
360  NEXT I
370  X=N+2
380  FOR I=2 TO N+1
390  U=X
400  FOR K=0 TO N+1-I
410  FOR J=I-1 TO 1 STEP -1
420  U=U-N-1+J
430  A(X)=A(X)-A(U)*A(U+K)
440  NEXT J
450  IF (K<>0) OR (I=N+1) THEN 480
```

```
460  IF A(X)<=0 THEN 300
470  A(X)=SQR(A(X))
480  IF K>0 THEN A(X)=A(X)/A(X-K)
490  X=X+1
500  U=X-K-1
510  NEXT K                          !Po zelji umetnuti ispis
520  NEXT I                          !matrice R
530  X=1                             !Racunanje R^(-1) i rjesenja
540  FOR I=1 TO N
550  U=X
560  A(U)=1/A(U)
570  X=X+1
580  FOR K=1 TO N+1-I
590  A(X)=A(X)*A(U)
600  X=X+1
610  NEXT K
620  NEXT I
630  X=N*(N+3)/2-1
640  FOR I=N TO 2 STEP -1
650  U=X
660  FOR J=I-1 TO 1 STEP -1
670  U=U-N-1+J
680  FOR K=1 TO N+1-I
690  V=U+K
700  X=X+1
710  A(V)=A(V)-A(U)*A(X)
720  NEXT K
730  X=X-K+1
740  A(U)=-A(U)*A(X)
750  NEXT J
760  X=X-(N+3)+I                     !Po zelji umetnuti ispis
770  NEXT I                          !matrice R^(-1) i rjesenja
780  X=1                             !Racunanje A^(-1)
790  U=-1
800  V=0
810  FOR I=1 TO N
820  P=U+2
830  V=P-1
840  S=0
850  FOR J=0 TO N-I
860  FOR K=J TO N-I
870  U=P+K
880  V=V+1
890  S=S+A(U)*A(V)
900  NEXT K
910  A(X)=S
920  X=X+1
930  V=V+1
940  S=0
950  NEXT J
960  X=X+1
970  NEXT I
```

```
980 ASSIGN #2 TO "IZLAZR"
990 BUFFER #2
1000 Kd=(N+1)*(N+2)/2
1010 PRINT LIN(3);TAB(20),"[pvv]=";A(Kd)
1020 PRINT #2;A(Kd)
1030 Kd=N+1
1040 PRINT LIN(3);"RJESENJE NORMALNIH JEDNADZBI"
1050 PRINT "------------------------------"
1060 FOR J=1 TO N
1061 FIXED 3
1070 PRINT TAB(26),A(Kd)
1080 PRINT #2;A(Kd)
1090 Kd=Kd+N-J+1
1100 NEXT J
1101 STANDARD
1110 PRINT LIN(3),"INVERZNA MATRICA "
1120 PRINT "-----------------"
1130 Kd=1
1140 FOR I=1 TO N
1150 PRINT
1160 PRINT I;". redak"
1170 PRINT TAB(10*I),
1171 FIXED 3
1180 FOR J=I TO N
1190 PRINT #2;A(Kd)
1200 PRINT A(Kd);TAB(10*J+10),
1210 Kd=Kd+1
1220 NEXT J
1230 PRINT
1240 Kd=Kd+1
1241 STANDARD
1250 NEXT I
1260 ASSIGN * TO #2
1270 END
```


SAŽETAK

U ovom radu dano je nekoliko primjedbi na načine rješavanja normalnih jednadžbi uočenih proučavanjem niza radova koji se bave ovom problematikom. Posebno se razmatra metoda Choleskog, te predlaže takav algoritam i program u kojem osnovna ideja Choleskog dolazi do punog izražaja.

ABSTRACT

In the present paper some remarks are given concerning the ways of solving normal equations, which were perceived studying a series of writings that deal with this subject. Particularly, Cholesky's method is discussed, and such an algorithm and a computer program are proposed, which emphasize Cholesky's fundamental idea.

Primljeno: 1983-12-20