

JÓZSEF ATTILA TUDOMÁNYEGYETEM

Természettudományi Kar

TOEPLITZ-TIPUSU SZALAGMÁTRIX EGYÜTTHATÓJU LINEÁRIS
EGYENLETRENDSZEREK MEGOLDÁSA GYORS FOURIER
TRANSZFORMÁCIÓVAL

Egyetemi doktori értekezés

Beck György

SZEGED

1982



TARTALOMJEGYZÉK

	old.
BEVEZETÉS	4
2. A DOLGOZATBAN HASZNÁLT JELÖLÉSEK ÉS LEGFONTOSABB FOGALMAK DEFINÍCIÓJA	7
2.1. Jelölések	7
2.2. Definíciók	9
3. AZ ELJÁRÁS MATEMATIKAI MEGALAPOZÁSÁHOZ SZÜKSÉGES TÉTELEK	19
4. A MEGOLDÁSI MÓDSZER LEÍRÁSA	30
4.1. Előzmények	30
4.2. Az eljárás elméleti kidolgozása	32
4.3. Az algoritmus optimalizálása	42
4.4. További műveletigény csökkentés minimális memóriaigény növeléssel	63
4.5. Az eljárás folyamatábrája	65
4.6. Az algoritmus bemutatása egy konkrét példán	70
5. A JELENLEG HASZNÁLT ALGORITMUSOK ÉS A BEMUTATOTT ELJÁRÁS ÖSSZEHASONLÍTÁSA	77



	old.
6. TOVÁBBI HATÉKONYSÁGNÖVELESI ÉS ÁLTALÁNOSÍTÁSI LEHETŐSÉGEK	83
6.1. Hatékonyságnövelés hardware és elméleti uton	83
6.2. Általánosítási lehetőségek	84
7. AZ ELJÁRÁS SZÁMITÓGÉPES IMPLEMENTÁCIÓJA	96
MELLÉKLETEK	98
IRODALOMJEGYZÉK	114

A gyakorlatban felmerülő igen sok fizikai és matematikai probléma számítógépes megoldása esetén szükség van valamilyen véges lineáris egyenletrendszer megoldására. Ezen egyenletrendszereknek - a gyakorlati alkalmazás szempontjából - egy igen jelentős részét alkotják az ún. szalag-Toeplitz rendszerek, azaz az olyan speciális lineáris egyenletrendszerek, melyek együtthatója Toeplitz-típusu szalagmátrix. Néhány alapvető felhasználási terület:

- digitális szűrők tervezése [13, 16]
- képfeldolgozás [18, 19]
- spektrálanalízis [22, 26]
- jelfeldolgozás [29, 35]
- idősorok elemzése [11, 28]

Jelenleg a gyakorlatban, a felmerülő Toeplitz-típusu szalagmátrix együtthatóju lineáris egyenletrendszerek megoldására az egyenletszám harmadik, ill. második hatványával arányos műveletszükségletű mód-

szereket alkalmaznak. /Egy műveletnek egy szorzás vagy osztás és egy összeadás vagy kivonás együttesét tekintjük./

A dolgozatban megmutatjuk, hogy megadható olyan megoldási algoritmus, amelynek műveletigénye $O(n \log_2 n)$. /Az n a megoldandó rendszer egyenleteinek a számát jelöli./ A továbbiakban megadunk egy ilyen megoldási algoritmust és annak számítógépes implementációját. A szükséges műveletigény értékek összehasonlítása alapján látható, hogy a dolgozatban bemutatásra kerülő eljárás nagy elemszámú egyenletrendszerek megoldása esetén számottevő műveletidő megtakarítást eredményez.

Mivel az eljárás kialakításában és az elért hatékonyságnövekedésben alapvető szerepet játszik a diszkrét Fourier transzformáció, ill. a transzformáció hatékony végrehajtására - J. Cooley és J. Tukey által 1965-ben - kidolgozott ún. gyors Fourier transzformációs algoritmus [14], ezért a kidolgozott módszer a gyors Fourier transzformáció egy újabb gyakorlati alkalmazásának tekinthetjük.

A témakörben rejlő feladatokra és lehetőségekre dr. Móricz Ferenc és Sebestyén Pál hívta fel a figyelmemet. Tudományos iránymutatásukért és a dolgozat készítése során nyújtott sokoldalú és értékes segítségükért ezuton is szeretnék köszönetet mondani.

Ugyancsak köszönettel tartozom munkatárimnak, különösen Simonfai Lászlónak és Szalay Imrének hasznos észrevételeikért és tanácsaikért.

2. A DOLGOZATBAN HASZNÁLT JELÖLÉSEK ÉS A LEGFONTOSABB FOGALMAK DEFINÍCIÓJA

2.1. Jelölések

A dolgozatban az egyes fogalmak világosabb megkülönböztetése céljából egymástól eltérő betűtípusokat alkalmazunk.

A mátrixokat nagybetűkkel, a skaláris mennyiségeket kisbetűkkel, a vektorokat pedig aláhuzott kisbetűkkel jelöljük. Ezenkívül a következő jelöléseket és rövidítéseket használjuk:

a diszkrét cirkuláris konvolúció jele: $\otimes$

két vektor komponensenként képzett szorzatának a jele: $\ast$

két vektor komponensenként képzett hányadosának a jele: $\oslash$

az a vektor diszkrét Fourier transzformáltja: $DFT(\underline{a})$

az a vektor inverz diszkrét Fourier transzformáltja: $IDFT(\underline{a})$

az a vektor gyors Fourier transzformációval meghatározott diszkrét Fourier transzformáltja:

$$\text{FDFT}(\underline{a})$$

az a vektor inverz gyors Fourier transzformációval meghatározott inverz diszkrét Fourier transzformáltja:

$$\text{FIDFT}(\underline{a})$$

A dolgozatban gyakran előfordul, hogy egy különbséget valamilyen modulus szerint kell képeznünk. Erre a különbségképzésre a szokásos:

$$i-j \text{ /mod } m/$$

jelölés mellett az

$$\underline{i-j}_m$$

jelölést is alkalmazzuk abban az esetben, ha a különbség egy indexkifejezésben szerepel. Ha a különbséget $\text{/mod } n/$ kell tekintenünk, akkor az $\underline{i-j}_n$ helyett csak $\underline{i-j}$ -t írunk.

A továbbiakban a $\log n$ mindig a kettes alapu logaritmust jelöli, azaz a $\log_2 n$ -t helyettesíti.

2.2. A legfontosabb fogalmak definíciója

2.1. DEFINÍCIÓ: /D.2.1./

Egy kvadratikus n -edrendű T mátrixot Toeplitz-típusúnak nevezünk [1], ha elemeire teljesül a

$$/2.1./ \quad t_{i,j} = t_{i-1,j-1} \quad 1 < i, j \leq n$$

egyenlőség.

A definíció egy másik, szemléletesebb formája a következő [30]:

2.2. DEFINÍCIÓ: /D.2.2./

Ha egy mátrixban a fődiagonális és minden azzal párhuzamos "ferde sor" csupa megegyező elemből áll, azaz, ha a mátrix elemei csupán oszlop- és sorindexük különbségétől függnének, akkor a mátrixot Toeplitz-típusúnak nevezzük.

Tehát ezen definíció alapján az A mátrix Toeplitz-típusu, ha

$$/2.2./ \quad a_{i,j} = a_{i-j} \quad 1 \leq i, j \leq n .$$

Az A mátrixot a következőképpen ábrázolhatjuk:

$$A = \begin{bmatrix} a_0 & a_{-1} & a_{-2} & \cdots & a_{-(n-2)} & a_{-(n-1)} \\ & a_1 & & & & a_{-(n-2)} \\ & a_2 & a_1 & & & \cdot \\ & \cdot & & a_1 & & \cdot \\ & \cdot & & & & a_2 \\ & a_{n-2} & & & & a_{-1} \\ a_{n-1} & a_{n-2} & \cdots & a_1 & & a_0 \end{bmatrix}$$

1. ábra

2.1. LEMMA:

A most megadott két definíció ekvivalens.

BIZONYÍTÁS:

- a. Tegyük fel, hogy egy T mátrix Toeplitz-tipusu
a D.2.1. értelemben, azaz

$$t_{i,j} = t_{i-1, j-1} \quad 2 \leq i, j \leq n .$$

Tekintsük a t_{i_1, j_1} és t_{i_2, j_2} elemeket, amelyekre teljesül:

$$\begin{aligned} /2.3./ \quad & 1 \leq i_1, i_2, j_1, j_2 \leq n \quad \text{és} \\ & i_1 - j_1 = i_2 - j_2 . \end{aligned}$$

Legyen

$$i. \quad i_1 = i_2 .$$

Ekkor /2.3./ alapján $j_1 = j_2$, azaz

$$t_{i_1, j_1} = t_{i_2, j_2} .$$

$$ii. \quad i_1 > i_2 .$$

$$/2.4./ \quad /2.3./ \text{ miatt } 0 < i_1 - i_2 = j_1 - j_2 = k .$$

$$/2.5./ \quad /2.3./ \text{ és } /2.4./ \text{ alapján}$$

$$i_1 = i_2 + k$$

$$j_1 = j_2 + k$$

$$2 \leq i_1, j_1 \leq n .$$

Alkalmazzuk a $t_{i1,j1}$ elemre D.2.1.-t, azaz /2.1./ képletet k -szor egymás után:

$$\begin{aligned} & \text{/2.1./} \\ t_{i1,j1} &= t_{i1-1,j1-1} \stackrel{\text{/2.1./}}{=} t_{i1-1/-1,j1-1/-1} = \\ \text{/2.6./} & \\ &= \dots \stackrel{\text{/2.1./}}{=} t_{i1-k,j1-k} \stackrel{\text{/2.5./}}{=} t_{i2,j2} \end{aligned}$$

iii. $i2 > i1$.

A szimmetria miatt a /2.6./ egyenlőség ebben az esetben is teljesül.

b. Legyen az A mátrix Toeplitz-típusú D.2.2. értelemben, azaz

$$a_{i,j} = a_{i-j} \quad 1 \leq i, j \leq n \quad .$$

Tekintsük az $a_{i,j}$ és $a_{i-1,j-1}$ elemeket, amelyre

$$2 \leq i, j \leq n \quad .$$

Az $1 \leq i, j, i-1, j-1 \leq n$ feltétel most nyilván teljesül, továbbá

$$\stackrel{\text{/2.2./}}{a_{i-1,j-1}} = a_{i-1/-1,j-1/-1} \stackrel{\text{/2.2./}}{=} a_{i-j} = a_{i,j}$$

Q.e.d.

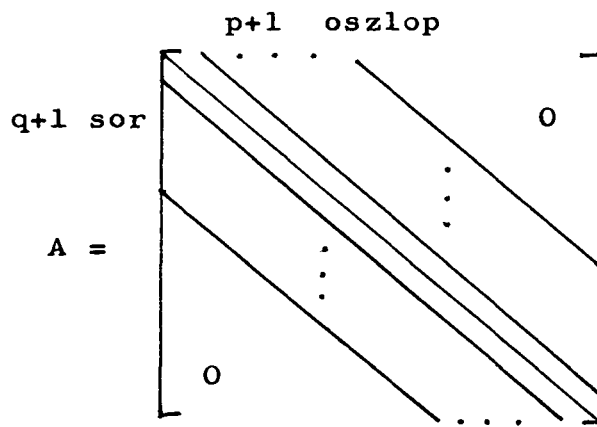
2.3. DEFINÍCIÓ:

Egy A mátrixot szalagmátrixnak nevezünk, ha csak a fődiagonálisában és a vele szomszédos párhuzamos p , ill. q "ferde sorban" tartalmaz nullától különböző elemet, azaz a főátlótól bizonyos távolságra már valamennyi eleme zérus.

Tehát A szalagmátrix, ha minden i, j -re

$$(i-j) < -p \quad \text{vagy} \quad (i-j) > q \quad \text{esetén} \\ a_{i,j} = 0$$

Az A szalagmátrix a következőképpen sematizálható:



2. ábra

2.4. DEFINÍCIÓ:

Egy n -edrendű kvadratikus C mátrixot cirkuláris vagy ciklikus mátrixnak nevezünk, ha elemeire teljesül:

$$c_{i,j} = c_{i-j} \quad / \text{mod } n /$$

vagy a dolgozatban alkalmazott jelöléssel:

$$c_{i,j} = c_{\underline{i-j}} .$$

Tehát $c_{i,j} = c_{i-j+kn} \quad /k=0,1,2 \dots/$ vagy

$$/2.7./ \quad i-j \equiv r-s \Rightarrow c_{i,j} = c_{r,s}$$

A definíció alapján nyilvánvaló, hogy a mátrixot az első sor vagy az első oszlop elemei egyértelműen meghatározzák. A C mátrix a következő alakú:

$$C(c_0, c_1, \dots, c_{n-1}) = \begin{bmatrix} c_0 & c_{n-1} & c_{n-2} & \dots & c_2 & c_1 \\ c_1 & c_0 & c_{n-1} & \dots & & c_2 \\ c_2 & c_1 & c_0 & \dots & & \\ \vdots & & & \ddots & & \\ \vdots & & & & \ddots & \\ c_{n-1} & c_{n-2} & & \dots & & c_0 \end{bmatrix}$$

3. ábra

A definíció alapján az is látszik, hogy minden cirkuláris mátrix egyben Toeplitz-típusu mátrix is. /Az állítás fordítottja nem érvényes./

2.5. DEFINÍCIÓ:

Legyen $\underline{a} = (a_0, a_1 \dots a_{n-1})$ egy n -elemű vektor,

$$w = \exp [2\pi i/n] ,$$

$$i = \sqrt{-1} ,$$

az A pedig egy n -edrendű kvadratikus mátrix, melynek elemei:

$$a_{j,k} = w^{jk} \quad 0 \leq j, k < n .$$

A $\text{DFT}(\underline{a}) = A \cdot \underline{a}$ n -elemű vektort, amelynek j -edik komponense:

$$/2.8./ \quad [\text{DFT}(\underline{a})]_j = \sum_{k=0}^{n-1} a_k w^{jk} \quad 0 \leq j < n$$

az $\underline{a}$ vektor diszkrét Fourier transzformáltjának nevezzük.

2.6. DEFINÍCIÓ:

Legyen $\underline{a} = (a_0, a_1, \dots, a_{n-1})$,

$$w = \exp [2\pi i/n],$$

$$i = \sqrt{-1},$$

az $\text{IDFT}(\underline{a})$ n -elemű vektort, melynek j -edik komponense

$$/2.9./ \quad [\text{IDFT}(\underline{a})]_j = \frac{1}{n} \sum_{k=0}^{n-1} a_k w^{-jk} \quad 0 \leq j < n$$

az $\underline{a}$ vektor inverz diszkrét Fourier transzformáltjának nevezzük.

2.7. DEFINÍCIÓ:

Legyen $\underline{a} = (a_0, a_1, \dots, a_{n-1})$ és

$$\underline{b} = (b_0, b_1, \dots, b_{n-1})$$

két n -elemű vektor. Diszkrét cirkuláris konvolúció-
juk $\underline{a} \circledast \underline{b}$ - egy n -elemű vektor lesz, amelynek
komponenseire teljesül:

$$/2.10./ \quad c_i = \sum_{j=0}^{n-1} a_j b_{i-j} \quad 0 \leq i \leq n-1,$$

/az $i-j$ különbség $\text{mod } n$ tekintendő/.



2.8. DEFINÍCIÓ:

Ha egy mátrixban a mellékdiagonális és minden azzal párhuzamos "ferde sor" csupa megegyező elem-ből áll, azaz ha a mátrix elemei csupán oszlop- és sorindexük összegétől függnnek, akkor a mátrixot Hankel-típusunak nevezzük.

Tehát egy H mátrix Hankel-típusu, ha elemeire teljesül a

$$/2.11./ \quad h_{i,j} = h_{i+j} \quad 1 \leq i, j \leq n$$

egyenlőség.

2.9. DEFINÍCIÓ:

Egy mátrixot erősen nemszingulárisnak nevezünk, ha egyetlen főminora sem szinguláris.

2.10. DEFINÍCIÓ:

$$\begin{aligned} \text{Az } \underline{a} &= (a_0, a_1, \dots, a_{n-1}) & \text{és} \\ \underline{b} &= (b_0, b_1, \dots, b_{n-1}) \end{aligned}$$

vektorok komponensenként képzett szorzata - $\underline{a} * \underline{b}$ -
az a $\underline{c}$ vektor lesz, amelynek komponensei:

$$c_i = a_i \cdot b_i \qquad 0 \leq i < n ,$$

azaz

$$\underline{c} = (a_0 b_0, a_1 b_1, \dots a_{n-1} \cdot b_{n-1}) ,$$

a két vektor komponenseként képzett hányadosa

- $\underline{a} * \underline{b}$ - pedig az a $\underline{d}$ vektor, melyre teljesül:

$$d_i = a_i / b_i \qquad 0 \leq i < n ,$$

vagyis

$$\underline{d} = (a_0 / b_0, a_1 / b_1, \dots a_{n-1} / b_{n-1}) .$$

3. AZ ELJÁRÁS MATEMATIKAI MEGALAPOZÁSÁHOZ SZÜKSÉGES TÉTELEK

Ebben a részben néhány olyan általános tételt ismer-
tetünk, amelyeknek alapvető szerepük van a későbbiekben
megadandó algoritmus elméleti megalapozásában. Ezen tété-
lek egy része jól ismert a cirkuláris- és Toeplitz mát-
rixok elméletéből, ezért ezek bizonyítását nem ismertetjük.

3.1. LEMMA:

A cirkuláris mátrixok szorzása kommutativ és a
szorzat mátrix is cirkuláris. /A bizonyítás megtalál-
ható [17]-ben./

3.2. LEMMA:

Egy nemszinguláris cirkuláris mátrix inverze is
cirkuláris [17].

3.3. LEMMA:

Toeplitz-tipusu mátrixok összege és különbsége
is Toeplitz-tipusu mátrix [17].

3.4. LEMMA:

A cirkuláris konvolúció kommutatív, azaz

$$\underline{a} \circledast \underline{b} = \underline{b} \circledast \underline{a} \quad .$$

BIZONYÍTÁS:

A lemma érvényességének belátásához azt kell megmutatnunk, hogy minden

$$0 \leq i \leq n-1 \quad \text{esetén}$$

$$[\underline{a} \circledast \underline{b}]_i = [\underline{b} \circledast \underline{a}]_i$$

vagyis

$$/3.1./ \quad \sum_{j=0}^{n-1} a_j \underline{b_{i-j}} = \sum_{k=0}^{n-1} b_k \underline{a_{i-k}} \quad .$$

Tekintsük a /3.1./ egyenlőség jobb oldalán álló kifejezést:

$$\sum_{k=0}^{n-1} b_k \underline{a_{i-k}} \quad .$$

Végezzük el a $j=i-k$ helyettesítést.

$$/3.2./ \sum_{k=0}^{n-1} b_k a_{\underline{i-k}} = \sum_{j=i}^{i-(n-1)/} a_j b_{\underline{i-j}} = \sum_{j=0}^i a_j b_{\underline{i-j}} + \sum_{j=i-(n-1)/}^{-1} a_j b_{\underline{i-j}} .$$

Mivel az indexeket moduló n kell vennünk, így az összegzés második tagja:

$$\sum_{j=i+1}^{n-1} a_j b_{\underline{i-j}} \quad \text{alakban írható, s ezért}$$

$$/3.2./ = \sum_{j=0}^i + \sum_{j=i+1}^{n-1} = \sum_{j=0}^{n-1} a_j b_{\underline{i-j}} ,$$

ez pedig éppen a /3.1./ egyenlőség bal oldala.

Mivel i tetszőleges 0 és $(n-1)$ közé eső érték lehet, így a kommutativitás valóban teljesül.

Q.e.d.

3.1. KOROLLÁRIUM:

Legyen C egy n -edrendű cirkuláris mátrix és jelölje $\underline{c}$ a C első oszlopvektorát. Ha $\underline{a}$ egy n -elemű oszlopvektor, akkor

$$C \cdot \underline{a} = \underline{c} \otimes \underline{a} .$$

BIZONYÍTÁS:

Legyen $\underline{r} = C \cdot \underline{a}$, azaz

$$r_k = \sum_{j=0}^{n-1} c_{k,j} a_j \quad 0 \leq k < n$$

Mivel C cirkuláris, ezért $c_{k,j} = c_{\underline{k-j}}$ és így

$$r_k = \sum_{j=0}^{n-1} c_{\underline{k-j}} a_j \stackrel{D2.7}{=} [\underline{a} \circledast \underline{c}]_k \stackrel{L3.4}{=} [\underline{c} \circledast \underline{a}]_k$$

Q.e.d.

3.5. LEMMA:

Tetszőleges $\underline{a}$ vektor esetén

$$/3.3./ \quad \text{IDFT}(\text{DFT}(\underline{a})) = \text{DFT}(\text{IDFT}(\underline{a})) = \underline{a}$$

3.6. LEMMA: /Cirkuláris konvolúciós tétel/

[3]

Legyen $\underline{a}$ és $\underline{b}$ két n -elemű vektor, ekkor

$$/3.4./ \quad \underline{a} \circledast \underline{b} = \text{IDFT}(\text{DFT}(\underline{a}) * \text{DFT}(\underline{b}))$$

azaz két vektor cirkuláris konvolúcióját megkaphatjuk a vektorok diszkrét Fourier transzformáltjai komponensenkénti szorzatának inverz diszkrét Fourier transzformáltjaként.

BIZONYÍTÁS:

Legyen $0 \leq s < n$

$$\begin{aligned} /3.5./ \quad \left[\text{DFT}(\underline{a}) * \text{DFT}(\underline{b}) \right]_s &= \sum_{j=0}^{n-1} a_j w^{sj} \sum_{k=0}^{n-1} b_k w^{sk} = \\ &= \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} a_j b_k w^{s(j+k)} \end{aligned}$$

$$\begin{aligned} /3.6./ \quad \left[\text{DFT}(\underline{a} \otimes \underline{b}) \right]_s &= \sum_{p=0}^{n-1} \left[\underline{a} \otimes \underline{b} \right]_p w^{sp} = \\ &= \sum_{p=0}^{n-1} \sum_{j=0}^{n-1} a_j b_{p-j} w^{sp} \end{aligned}$$

A /3.6./-ban végezzük el a $k = p-j$ helyettesítést és cseréljük fel az összegzés sorrendjét.

$$\begin{aligned} /3.7./ \quad \sum_{p=0}^{n-1} \sum_{j=0}^{n-1} a_j b_{p-j} w^{sp} &= \sum_{j=0}^{n-1} \sum_{k=-j}^{n-1-j} a_j b_k w^{s(j+k)} = \\ &= \underbrace{\sum_{j=0}^{n-1} \sum_{k=-j}^{-1}}_{/3.8./} + \sum_{j=0}^{n-1} \sum_{k=0}^{n-1-j} \end{aligned}$$

/3.8./-ban legyen k helyett az $(n+k)$ az új indexváltozó, így

$$/3.9./ \sum_{j=0}^{n-1} \sum_{k=-j}^{-1} a_j b_k w^{s(j+k)} = \sum_{j=0}^{n-1} \sum_{k=n-j}^{n-1} a_j b_{\underline{k-n}} w^{s(j+k-n)} .$$

Azonban

$$w^{s(j+k-n)} = \exp(2\pi i s(j+k-n)/n) = \exp(2\pi i s(j+k)/n)$$

$$\underbrace{\exp(-2\pi i s n/n)}_{=1} = \exp(2\pi i s(j+k)/n) = w^{s(j+k)} .$$

Ezért a /3.9./ a következő alakban írható fel:

$$\sum_{j=0}^{n-1} \sum_{k=n-j}^{n-1} a_j b_k w^{s(j+k)} ,$$

és a /3.7./ pedig

$$/3.10./ \sum_{j=0}^{n-1} \sum_{k=n-j}^{n-1} + \sum_{j=0}^{n-1} \sum_{k=0}^{n-j-1} = \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} a_j b_k w^{s(j+k)} .$$

A /3.5./ és /3.10./ alapján felírható a következő egyenlőség:

$$/3.11./ \left[\text{DFT}(\underline{a}) * \text{DFT}(\underline{b}) \right]_s = \left[\text{DFT}(\underline{a} \circledast \underline{b}) \right]_s \quad 0 \leq s < n .$$

Ez viszont az L.3.5. miatt

$$\left[\text{IDFT}(\text{DFT}(\underline{a}) * \text{DFT}(\underline{b})) \right]_s = \left[\underline{a} \otimes \underline{b} \right]_s \quad 0 \leq s < n$$

alakban is felírható, ami pedig éppen a cirkuláris konvolúciós tétel érvényességét bizonyítja.

Q.e.d.

3.7. LEMMA:

Egy cirkuláris mátrix inverze létezésének szükséges és elégséges feltétele, hogy a mátrix első oszlopvektorának diszkrét Fourier transzformáltja ne tartsa egyetlen nulla komponens sem.

BIZONYÍTÁS:

i, Szükségesség

Legyen A egy cirkuláris mátrix, $\underline{a}$ pedig az A első oszlopvektora. Tegyük fel, hogy

$$\begin{aligned} & a, \exists A^{-1} \quad \text{és} \\ & b, \exists i \quad 0 \leq i < n \quad \left[\text{DFT}(\underline{a}) \right]_i = 0 \end{aligned}$$

Jelölje E az egységmátrixot, $\underline{e}$ és $\underline{b}$ pedig az E és A^{-1} első oszlopvektorát. Ekkor felírhatjuk a következő egyenlőséget:

$$/3.12./ \quad \underline{e} = A \cdot \underline{b} \stackrel{K3.1}{=} \underline{a} \circledast \underline{b} \stackrel{L3.6}{=} \text{IDFT}(\text{DFT}(\underline{a}) \circledast \text{DFT}(\underline{b})) .$$

Az L.3.5. felhasználásával /3.12./ a következő alakban írható:

$$\text{DFT}(\underline{e}) = \text{DFT}(\underline{a}) \circledast \text{DFT}(\underline{b}) ,$$

az i -edik komponensekre pedig

$$/3.13./ \quad [\text{DFT}(\underline{e})]_i = [\text{DFT}(\underline{a}) \circledast \text{DFT}(\underline{b})]_i .$$

Mivel $\underline{e} = (1, 0, \dots, 0)$, így

$$[\text{DFT}(\underline{e})]_i = \sum_j e_j w^{ij} = 1 ,$$

és így /3.13./ alapján

$$/3.14./ \quad [\text{DFT}(\underline{a}) \circledast \text{DFT}(\underline{b})]_i = [\text{DFT}(\underline{a})]_i \cdot [\text{DFT}(\underline{b})]_i = 1 .$$

Igy azonban a kezdeti feltevéssel $[\text{DFT}(\underline{a})]_i = 0$ - ellentmondásba kerültünk.

ii, Elégségesség

Tegyük fel, hogy az A cirkuláris mátrix első oszlopának diszkrét Fourier transzformáltja nem tartalmaz

nulla komponenst.

Készítsük el a

$$/3.15./ \quad \underline{b} = \text{IDFT}\left(\underline{1} \underset{*}{*} \text{DFT}(\underline{a})\right)$$

vektort. L.3.6. felhasználásával az előbbi egyenlőség

$$/3.16./ \quad \text{DFT}(\underline{b}) = \underline{1} \underset{*}{*} \text{DFT}(\underline{a})$$

alakra hozható. A diszkrét Fourier transzformáció definíciója alapján az $\underline{1}$ vektor helyett írhatunk $\text{DFT}(\underline{e})$ -t és így a /3.16./ új alakja:

$$\text{DFT}(\underline{b}) = \text{DFT}(\underline{e}) \underset{*}{*} \text{DFT}(\underline{a}) \quad ,$$

és ezért

$$/3.17./ \quad \text{DFT}(\underline{e}) = \text{DFT}(\underline{a}) \underset{*}{*} \text{DFT}(\underline{b}) \quad .$$

Az L.3.5. és L.3.6. alapján a /3.17./ fennállása miatt érvényes a

$$/3.18./ \quad \underline{e} = \underline{a} \underset{*}{*} \underline{b} \quad \text{egyenlőség.}$$

Legyen B olyan cirkuláris mátrix, melynek első oszlopvektora $\underline{b}$. L.3.1. alapján $A \cdot B = B \cdot A = C$ és a C cirkuláris mátrix.

Jelölje $\underline{c}$ a C első oszlopát. Ekkor

$$\underline{c} = A \cdot \underline{b} \stackrel{L_{3.1.}}{=} \underline{a} \circledast \underline{b} \stackrel{/3.18./}{=} \underline{e} .$$

Tehát a C olyan cirkuláris mátrix, amelynek első oszlopvektora $\underline{e}$, azaz éppen az egységmátrix, vagyis

$$A \cdot B = B \cdot A = E$$

ami viszont éppen azt bizonyítja, hogy a B mátrix az A inverze.

Q.e.d.

3.2. KOROLLÁRIUM:

Egy cirkuláris mátrix akkor és csak akkor nonszinguláris, ha első oszlopvektorának diszkrét Fourier transzformáltja nem tartalmaz nulla komponenst.

3.3. KOROLLÁRIUM:

Ha A egy nonszinguláris cirkuláris mátrix, melynek első oszlopa $\underline{a}$, akkor az A inverze az a B cirkuláris mátrix lesz, amelynek első oszlopvektora a következő képlettel határozható meg:

$$/3.19./ \quad \underline{b} = \text{IDFT}\left(\underline{1} \underset{*}{*} \text{DFT}(\underline{a})\right) .$$

BIZONYÍTÁS:

Az L.3.2. miatt az inverz mátrix valóban cirkuláris lesz és a /3.19./ képlet érvényességét a 3.7. lemma bizonyításának második részében mutattuk meg.

4. A MEGOLDÁSI MÓDSZER LEÍRÁSA

4.1. Előzmények

Tekintsük a

$$/4.1./ \quad \underline{A} \underline{x} = \underline{y}$$

lineáris egyenletrendszert, melyben A Toeplitz-típusu mátrix /nem feltétlenül szalagmátrix/.

Korábban a felmerülő gyakorlati alkalmazások esetén a /4.1./ típusu egyenletrendszerek megoldására legtöbbször az általános eliminációs - /pl: Gauss, Jordan [27] / vagy iterációs - /pl: Jacobi, Seidel [27] / eljárásokat használták. Ezen algoritmusok műveletigénye az egyenletszám harmadik hatványával arányos /pl: Gauss: $n^3/3$, Jordan: $n^3/2$ /.

A Toeplitz-mátrixok speciális tulajdonságai azonban már régóta azt sugallták, hogy megfelelő algoritmusokkal jelentősen csökkenthető a /4.1./ egyenletrendszer megoldásához szükséges műveletek száma.

A speciális tulajdonságok kihasználásával N. Levinson 1947-ben adott egy, a fent említett eljárásoknál hatékonyabb rekurzív módszert a /4.1./ feladat megoldására [24] .

Lényegében az ő eljárását javította tovább W. Trench és 1964-ben, ill. 1967-ben algoritmust adott a Toeplitz-típusu mátrixok inverzének meghatározására, valamint a /4.1./ típusu egyenletrendszerek megoldására [31,32]. Trench rekurzív algoritmusai azonban csak abban a speciális esetben alkalmazhatók, ha a Toeplitz-típusu mátrixok szimmetrikusak, ill. komplex esetben Hermit-szimmetrikusak, továbbá erősen nonszingulárisak.

Trench algoritmusainak nonszimmetrikus, ill. nem Hermit-szimmetrikus esetre való általánosítását S. Zohar oldotta meg 1969-ben és 1974-ben [36, 37]. Ezen algoritmusok műveletigénye már csak az egyenletszám második hatványával arányos: $3n^2$.

A Toeplitz-típusu szalagmátrixok a Toeplitz-típusu mátrixok speciális változatai. A

/4.2./ $T\underline{x} = y$

Toeplitz-típusu szalagmátrix együtthatóju lineáris egyenletrendszer $\underline{x}$ megoldásvektora az - általános Toeplitz egyenletrendszerek megoldására N. Lewinson, W. Trench és S. Zohar által megadott eljárások, valamint W. Trench által 1974-ben publikált, már speciálisan a Toeplitz-típusu szalagmátrixokra kidolgozott módszer [33] eredmé -

nyeit felhasználó - un. Lewinson-Trench-Zohar algoritmus /a továbbiakban L-T-Z/ alkalmazásával $1,5n^2 + O(n(p+q))$ művelettel kapható meg.

A. Jain 1978-ban publikált egy - az együtthatómátrix cirkuláris felbontásán alapuló - $O(n \log n) + O((p+q)^2)$ műveletszükségletű eljárást a /4.2./ egyenletrendszer megoldására [20]. Ez az algoritmus azonban a Toeplitz-típusu szalagmátrix együtthatóju lineáris egyenletrendszereknek csak egy - bizonyos feltételeket kielégítő - részhalmazára alkalmazható. /A. Jain algoritmusának részletesebb elemzése az 1. mellékletben található./

A dolgozatban a továbbiakban - a témakörben folytatott korábbi vizsgálódásaim eredményeit [4,5,6] összegezve - Jain algoritmusának olyan, $O(n \log n) + O((p+q)^2)$ műveletigényű általánosítását adom meg, amely már tetszőleges Toeplitz-típusu szalagmátrix együtthatóju lineáris egyenletrendszerek megoldására alkalmas.

4.2. Az eljárás elméleti kidolgozása

Adott tehát a

$$\text{/4.2./} \quad T \underline{x} = \underline{y}$$

A következőkben az együtthatómátrix cirkuláris kiterjesztésével a /4.2./ egyenletrendszert két egyenletrendszerre bontjuk. Ezek közül az egyik egy n -edrendű cirkuláris mátrix együtthatóju lineáris egyenletrendszer, amelynek megoldása - a később megadandó tételek alapján - $O(n \log n)$ művelettel meghatározható. A másik pedig egy $O((p+q)^2)$ művelettel megoldható $(p+q) \times (p+q)$ -s Toeplitz egyenletrendszer lesz.

Legyen T_c a T mátrix n -edrendű cirkuláris kiterjesztése, azaz olyan n -edrendű mátrix, amely tartalmazza a T nem nulla elemeit és cirkuláris. /A kiterjeszthetőségnek az a feltétele, hogy $(p+q) \nmid n$ teljesüljön. Ha ez a feltétel nem teljesül, akkor a megoldási algoritmusnak a 6.2. részben ismertetésre kerülő általánosított változatát alkalmazhatjuk. /

A T mátrix n -edrendű cirkuláris kiterjesztésével keletkezett T_c a következő lesz:

$$T_c = \begin{bmatrix} t_0 & t_{-1} & \dots & t_{-p} & 0 & t_q & \dots & t_1 \\ t_1 & & & & & & & \vdots \\ & & & & & & & \vdots \\ & & & & & & & \vdots \\ & & & & & & & t_q \\ & & & & & 0 & & \\ & & & & & & & 0 \\ t_q & & & & \vdots & & & \\ 0 & & & & \vdots & & & \\ & & & & & & & t_{-p} \\ & & & & & & & \vdots \\ & & & & & & & \vdots \\ & & & & & & & \vdots \\ t_{-p} & & & & & & & \\ & & & & & & & 0 \\ & & & & & & & \vdots \\ & & & & & & & \vdots \\ & & & & & & & \vdots \\ t_{-1} & \dots & t_{-p} & 0 & t_q & \dots & t_1 & t_0 \end{bmatrix}$$

5. ábra

A T_c mátrix $\underline{t}$ -vel jelölt első oszlopvektora pedig:

/4.3./ $\underline{t} = (t_0, t_{-1}, \dots, t_q, 0, \dots, 0, t_{-p}, \dots, t_{-1}).$

Legyen T_b egy olyan mátrix, mely kielégíti a

$$T_b = T_c - T$$

Tegyük fel, hogy a T_c nonszinguláris. Ezen feltétel mellett a fenti egyenlet a következő alakra hozható:

$$\underline{x} = T_c^{-1} \underline{y} + T_c^{-1} T_b \underline{x} \quad .$$

Vezessük be a következő jelöléseket:

$$/4.4./ \quad B = T_c^{-1} \quad ; \quad \underline{z} = T_c^{-1} \underline{y} \quad .$$

Ezen jelölések mellett:

$$/4.5./ \quad \underline{x} = \underline{z} + B T_b \underline{x} \quad .$$

A továbbiakban vizsgáljuk meg a /4.5./ jobboldalán szereplő $T_b \underline{x}$ szorzatot.

$$T_b \cdot \underline{x} = \begin{bmatrix} t_q & \dots & t_1 & & & \\ & & & \ddots & & \\ & & & & t_q & \\ & & & & & 0 \\ & & & & & & \ddots \\ & & & & & & & t_{-p} \\ & & & & & & & \vdots \\ & & & & & & & \vdots \\ & & & & & & & t_{-1} \dots t_{-p} \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ \vdots \\ x_{n-1} \end{bmatrix}$$

7. ábra

Az $\underline{u}$ és $\underline{v}$ meghatározása érdekében particionáljuk a B mátrixot és a $\underline{z}$ vektort a következőképpen:

$$B = \begin{bmatrix} B^{ii} & B^{im} & B^{it} \\ B^{mi} & B^{mm} & B^{mt} \\ B^{ti} & B^{tm} & B^{tt} \end{bmatrix} \begin{matrix} p \\ \\ q \end{matrix}$$

$$\underline{z} = \begin{bmatrix} \underline{z}^i \\ \underline{z}^m \\ \underline{z}^t \end{bmatrix} \begin{matrix} p \\ \\ q \end{matrix}$$

q p

9. ábra

Igy a /4.7./ alapján $\underline{x}^i$ -re és $\underline{x}^t$ -re a következő egyenletek kaphatók:

$$/4.8./ \quad \underline{x}^i = B^{ii}\underline{u} + B^{it}\underline{v} + \underline{z}^i$$

$$\underline{x}^t = B^{ti}\underline{u} + B^{tt}\underline{v} + \underline{z}^t$$

Azonban az $\underline{x}^i$ és $\underline{x}^t$ a /4.6./ egyenletek alapján is kifejezhetők, ugyanis az F olyan trianguláris mátrix, amelynek főátlójában minden elem t_q -val egyenlő, s így a $t_q \neq 0$ kezdeti feltevés miatt az F mátrixnak létezik az inverze. Hasonló okok miatt - $t_p \neq 0$ - létezik a G^{-1} is. Ezért a /4.6./ alapján:

$$/4.9./ \quad \underline{x}^t = F^{-1} \underline{u} \quad \text{és} \quad \underline{x}^i = G^{-1} \underline{v} \quad .$$

A /4.8./ és /4.9./ szerint:

$$\begin{aligned} /4.10./ \quad \underline{x}^t &= F^{-1} \underline{u} = B^{ti} \underline{u} + B^{tt} \underline{v} + \underline{z}^t \\ \underline{x}^i &= G^{-1} \underline{v} = B^{ii} \underline{u} + B^{it} \underline{v} + \underline{z}^i \quad . \end{aligned}$$

Az egyenleteket átrendezve kapjuk:

$$\begin{aligned} /4.11./ \quad -B^{tt} \underline{v} + (F^{-1} - B^{ti}) \underline{u} &= \underline{z}^t \\ (G^{-1} - B^{it}) \underline{v} + -B^{ii} \underline{u} &= \underline{z}^i \quad . \end{aligned}$$

Legyen

$$/4.12./ \quad M = \left[\begin{array}{c|c} 0 & F^{-1} \\ \hline G^{-1} & 0 \end{array} \right] \begin{array}{l} q \\ p \end{array} ; \quad N = \left[\begin{array}{c|c} B^{tt} & B^{ti} \\ \hline B^{it} & B^{ii} \end{array} \right]$$

p q

10. ábra

$$\underline{w} = \begin{bmatrix} \underline{v} \\ \text{---} \\ \underline{u} \end{bmatrix} \begin{matrix} q \\ \\ p \end{matrix} ; \quad \underline{s} = \begin{bmatrix} \underline{z}^t \\ \text{---} \\ \underline{z}^i \end{bmatrix} \begin{matrix} q \\ \\ p \end{matrix}$$

11. ábra

és $R = M - N$.

Ezekkel a jelölésekkel a /4.11./ -nek megfelelő mátrixegyenlet:

$$/4.13./ \quad R\underline{w} = \underline{s} \quad .$$

A /4.13./ mátrixegyenlet nem más, mint egy $(p + q)$ egyenletből álló lineáris egyenletrendszer a $\underline{v}$ és $\underline{u}$ vektorokra, vagyis a $v_0, v_1, \dots, v_{p-1}; u_0, u_1, \dots, u_{q-1}$ ismeretlenekre. Mivel a /4.2./ rendszerben T-ről feltettük, hogy nonszinguláris, így ez biztosítja, hogy a kiindulási egyenletrendszernek létezik egyértelmű $\underline{x}$ megoldásvektora. Természetesen így az $\underline{x}$ első p és utolsó q komponense - $\underline{x}^i, \underline{x}^t$ - is egyértelműen meghatározható, ami viszont az eredeti egyenletrendszerből származott /4.10./ alapján azt biztosítja, hogy a /4.13./ egyenletrendszer R együtthtómátrixa sem lehet szinguláris és így létezik egyértelmű megoldás $\underline{v}$ -re és $\underline{u}$ -ra.

Igy a következő főbb lépésekből álló eljárást tudjuk összeállítani az $\underline{x}$ megoldásvektor meghatározására:

I.	T_c	meghatározása	/4.3./	alapján
II.	B	kiszámítása	/4.4./	"
III.	M	meghatározása	/4.12./	"
IV.	N	meghatározása	/4.12./	"
V.	$\underline{z}$	kiszámítása	/4.4./	"
VI.	$\underline{w}$	kiszámítása	/4.13./	"
VII.	$\underline{x}$	meghatározása	/4.7./	"

Ez az algoritmus a 4.2. pontban megadott eredmények alapján alkalmas a /4.2./ Toeplitz-típusu szalagmátrix együtthatóju lineáris egyenletrendszer megoldására.

4.3. Az algoritmus optimalizálása

Ha a /4.2./ kiindulási egyenletrendszer $\underline{x}$ megoldásvektorának meghatározására az előző részben ismertett algoritmust használjuk, és az egyes lépésekhez a gyakorlatban általánosan használt eljárásokat alkalmazzuk, azaz a mátrix invertáláshoz és a /4.13./ lineáris egyenletrendszer megoldásához a Jordan és Gauss eliminációt használjuk és a mátrix-vektor szorzást is

hagyományosan végezzük, akkor a teljes műveletigény jóval a jelenleg használt L-T-Z algoritmus műveletszükséglete fölötti érték lesz. Azonban a T_c mátrix cirkuláris voltának a kihasználásával és a gyors Fourier transzformációs algoritmus használatával a műveletigény jelentősen redukálható.

A továbbiakban megvizsgáljuk, hogy az összeállított eljárás egyes lépéseiben milyen tevékenységeket kell véggeznünk és ezek végrehajtásában milyen hatékonyságnövelő algoritmusok alkalmazhatók.

I. A T_c mátrix meghatározása

Jelölje

$$\underline{a} = (a_0, a_1, \dots, a_p, 0 \dots 0) \quad \text{és}$$

$$\underline{b} = (b_0, b_1, \dots, b_q, 0 \dots 0)$$

a megoldandó /4.2./ egyenletrendszer T együtthatómátrixának első sor-, ill. oszlopvektorát. Ekkor a keresett mátrixot ábrázoló 5. ábra alapján a T_c első oszlopa:

$$/4.14./ \quad \underline{c} = (b_0, b_1, \dots, b_q, 0 \dots 0, a_p, a_{p-1}, \dots, a_1)$$

Mivel a T_c cirkuláris, így a 2.4. definíció értelmében a T_c -t a most megadott első oszlopvektora egyértelműen meghatározza.

A megoldási algoritmusnak ebben az első lépésében csupán $p + q + 1$ értékadást kell elvégeznünk, melynek műveletszükséglete a dolgozatban használt műveletfogalom értelmében 0. /Ez valójában azt jelenti, hogy az értékadás műveletigénye a multiplikatív műveletekéhez képest elhanyagolható./

II. A B mátrix meghatározása

Mivel a B nem más, mint az n -edrendű cirkuláris T_c mátrix inverze, így a 3.3. korollárium alapján a B első oszlopvektorát a következő képlettel kaphatjuk meg:

$$/4.15./ \quad \underline{b} = \text{IDFT} \left(\underline{1} \begin{matrix} * \\ * \end{matrix} \text{DFT}(\underline{t}) \right)$$

ahol $\underline{t}$ -vel jelöltük a T_c mátrix első oszlopát.

Mivel a B cirkuláris, így az első oszlop egyértelműen meghatározza a teljes mátrixot.

Ha a /4.15./ képlettel megadott $\underline{b}$ vektor meghatározásához szükséges diszkrét Fourier transzformációt és

inverzét a gyors Fourier transzformációs algoritmussal számítjuk ki, amely egy n -edrendű vektor /inverz/ diszk-rét Fourier transzformáltját $n \cdot \log n$ művelettel határozza meg [12], akkor a szükséges műveletek száma:

$\text{FDFT}(\underline{t})$ számítására $n \cdot \log n$

$(\underline{1} * \text{FDFT}(\underline{t}))$ számítása n

$\text{FIDFT}(\underline{1} * \text{FDFT}(\underline{t}))$ számítására $n \cdot \log n$

azaz a teljes műveletigény

$$2n \log n + n$$

lesz. Ez az érték - elég nagy n esetén - jóval kisebb az általános mátrix inverziós eljárások $O(n^3)$ műveletszükségleténél.

III. Az M mátrix meghatározása

A /4.12./ szerint definiált M mátrix meghatározásához a G és F Toeplitz-típusú alsó-, ill. felső triangu-láris mátrixok inverzét kell kiszámítanunk. A két mátrix speciális tulajdonságait kihasználva az általános mátrix inverziós eljárásnál hatékonyabb algoritmussal tudjuk



meghatározni az F és a G inverzét a következő tétel eredményeinek a felhasználásával:

4.1. TÉTEL:

Legyen G egy nemszinguláris n -edrendű Toeplitz-típusú alsó trianguláris mátrix, H pedig a G inverze. Jelölje $\underline{g}$ és $\underline{h}$ a G , ill. H első oszlopvektorát. Ekkor

i, H is Toeplitz-típusú alsó trianguláris mátrix lesz.

ii, a $\underline{h}$ -t a következő rekurziós képlettel kaphatjuk meg.

$$h_0 = 1/g_0$$

/4.16/

$$h_m = -h_0 \sum_{k=1}^m g_k \cdot h_{m-k} \quad 0 < m < n$$

BIZONYÍTÁS

Ha E az n -edrendű egységmátrix és $\underline{h}^i$ -vel és $\underline{e}^i$ -vel jelöljük a H és E i -edik oszlopvektorát, akkor érvényes a következő egyenlet:

$$G \underline{h}^i = \underline{e}^i \quad 0 < i \leq n$$

A /4.17./ -et részletesen felírva - kihasználva, hogy a G alsó trianguláris és Toeplitz-tipusu - a /4.18./ egyenletrendszert kapjuk:

$$\begin{array}{rcl}
 g_0 h_{1,i} + & 0 & = 0 \\
 g_1 h_{1,i} + g_0 h_{2,i} + & 0 & = 0 \\
 \cdot & & \\
 \cdot & & \\
 \cdot & & \\
 /4.18./ \quad g_{i-1} h_{1,i} + g_{i-2} h_{2,i} + \dots + g_0 h_{i,i} + & 0 & = 1 \\
 \cdot & & \cdot \\
 \cdot & & \cdot \\
 \cdot & & \cdot \\
 g_{n-1} h_{1,i} + g_{n-2} h_{2,i} + \dots + g_{n-i} h_{i,i} + \dots + g_0 h_{n,i} = 0 .
 \end{array}$$

Mivel G-ről feltettük, hogy nemszinguláris, így a determinánsa, $(g_0)^n$ nem lehet nulla, s ezért a /4.18./ első egyenlete alapján minden $1 \leq i \leq n$ esetén

$$h_{1,i} = 0 .$$

Tegyük fel, hogy $0 \leq j-1 < i \leq n$ esetén

$$h_{1,i} = h_{2,i} = \dots = h_{j-1,i} = 0 ,$$

és tekintsük a $h_{j,i}$ elemet:

a, $j < i$.

A /4.18./ j -edik egyenlete és az indukciós feltevés alapján érvényes a

$$g_0 h_{j,i} = 0$$

egyenlőség. Viszont a g_0 -ról már korábban beláttuk, hogy nem lehet nulla, így tehát

$$\text{/4.19./ } h_{j,i} = 0 \quad 0 < j < i \leq n \text{ esetén.}$$

A /4.19./ feltétel teljesülése éppen azt bizonyítja, hogy a H alsó trianguláris mátrix.

b, $j = i$.

A /4.18./ i -edik egyenlete és /4.19./ alapján

$$g_0 h_{i,i} = 1 \quad \text{vagyis}$$

$$\text{/4.20./ } h_{i,i} = 1/g_0 .$$

c, $j > i$.

A /4.18./ j -edik egyenlete alapján a /4.19./ és

/4.20./ eredmények felhasználásával felírhatjuk a következő egyenlőséget:

$$\varepsilon_{j-i}/\varepsilon_0 + \varepsilon_{j-i-1}h_{i+1,i} + \dots + \varepsilon_0 h_{j,i} = 0.$$

Ezen egyenlet alapján a $h_{j,i}$ -t a kisebb sorindexű i -edik oszlopbeli elemekkel a következő képlettel tudjuk kifejezni:

$$/4.21./ \quad h_{j,i} = (-1/\varepsilon_0) \cdot \sum_{k=1}^{j-i} \varepsilon_k \cdot h_{j-k,i}.$$

Tehát a H mátrix i -edik oszlopának az elemei:

$$/4.22./ \quad h_{j,i} = \begin{cases} 0 & 0 < j < i \leq n \\ 1/\varepsilon_0 & 0 < j = i \leq n \\ -1/\varepsilon_0 \cdot \sum_{k=1}^{j-i} \varepsilon_k h_{j-k,i} & 0 < i < j \leq n \end{cases}.$$

A 2.2. definíció értelmében a H mátrix abban az esetben Toeplitz-típusú, ha tetszőleges $0 < i, j, t, s \leq n$ esetén

$$j - i = t - s \Rightarrow h_{j,i} = h_{t,s}.$$

Legyen $0 < i, j, t, s \leq n$ és

$$1, j-i = t-s < 0$$

akkor a

$$/4.22./ \text{ alapján } h_{j,i} = 0 = h_{t,s} .$$

$$2, j-i = t-s = 0$$

$$h_{j,i} \stackrel{/4.22./}{=} 1/g_0 \stackrel{/4.22./}{=} h_{t,s} .$$

$$3, j-i = t-s = 1$$

alkalmazzuk újra a /4.22./ eredményeit

$$\begin{aligned} h_{j,i} &= (-1/g_0) g_1 \cdot h_{j-1,i} = (-1/g_0) g_1 h_{i,i} = \\ &= (-1/g_0) g_1 / g_0 = (-1/g_0) g_1 \cdot h_{s,s} = (-1/g_0) g_1 h_{t-1,s} = \\ &= h_{t,s} . \end{aligned}$$

4, Tegyük fel, hogy $0 < j-i = t-s = r-1$ esetén

$$h_{j,i} = h_{t,s} \quad \text{és legyen}$$

$$0 < j-i = t-s = r .$$

Alkalmazzuk az indukciós feltevést és a /4.22./-t:

$$\begin{aligned}
 h_{j,i} &= (-1/\varepsilon_0) \sum_{k=1}^{j-i} \varepsilon_k h_{j-k,i} = \\
 &= (-1/\varepsilon_0) \left(\sum_{k=1}^{j-i-1} \varepsilon_k h_{j-k,i} + \varepsilon_{j-i} h_{i,i} \right) = \\
 &= (-1/\varepsilon_0) \left(\sum_{k=1}^{t-s-1} \varepsilon_k h_{t-k,s} + \varepsilon_{t-s} h_{s,s} \right) = h_{t,s} .
 \end{aligned}$$

Igy tehát a H mátrix valóban Toeplitz-típusú és ezért a /4.22./ képletben a $h_{j,i}$ helyett h_{j-i} -t is írhatunk:

$$/4.23./ \quad h_{j-i} = \begin{cases} 0 & 0 < j < i \leq n \\ 1/\varepsilon_0 & 0 < j = i \leq n \\ -1/\varepsilon_0 \cdot \sum_{k=1}^{j-i} \varepsilon_k h_{j-k-i} & 0 < i < j \leq n . \end{cases}$$

Ha a /4.23./ képletben a $j-i$ helyett egy m indexváltozót alkalmazunk:

$$h_0 = 1/\varepsilon_0$$

$$h_m = -h_0 \sum_{k=1}^m \varepsilon_k h_{m-k} \quad 0 < m < n$$

akkor éppen a bizonyítandó /4.16./ rekurziós képletet kapjuk.

Q.e.d.

4.1. KOROLLÁRIUM:

Egy n -edrendű nemszinguláris Toeplitz-típusu felső trianguláris mátrix inverze is Toeplitz-típusu felső trianguláris mátrix lesz, és az inverz mátrix első sorának elemeit a mátrix első sorának elemeiből a 4.1. tételben megadott /4.16./ rekurrenz-képlet alapján számíthatjuk ki.

Mivel az F egy q -adrendű nemszinguláris Toeplitz-típusu felső trianguláris mátrix a G pedig egy p -edrendű nemszinguláris Toeplitz-típusu alsó trianguláris mátrix, így a 4.1. tétel és 4.1. korollárium alapján az inverzeik is hasonló tulajdonságúak, és elemeiket a /4.16./ képlet felhasználásával határozhatjuk meg.

Ha $\underline{h}$ és $\underline{s}$ jelöli a G^{-1} ill. F^{-1} mátrixok első oszlopát, ill. sorát, akkor az M mátrixot a következőképpen ábrázolhatjuk:

$$M = \begin{bmatrix} & & & s_0 & s_1 & \dots & s_{q-1} \\ & & & & & & \vdots \\ & 0 & & & & & s_1 \\ & & & 0 & & & s_q \\ \hline h_0 & & & & & & \\ h_1 & & 0 & & 0 & & \\ \vdots & & & & & & \\ h_{p-1} & \dots & h_1 & h_0 & & & \end{bmatrix}$$

p

q

q

p

12. ábra

Az ábra alapján is jól látszik, hogy az M egy $(p+q)$ - adrendű Toeplitz-típusú mátrix.

Ha a /4.2./ kiindulási egyenletrendszert megoldó "hétlépéses" eljárásnak ebben a III. lépésében az M mátrix meghatározásakor a /4.16./ rekúrziós képletet alkalmazzuk, akkor a mátrix kiszámításának teljes műveletszüksége:

$$p(p+1)/2 + q(q+1)/2 = (p^2 + q^2)/2 + (p+q)/2$$

IV. Az N mátrix meghatározása

A /4.12./ -ben definiált N mátrix a B cirkuláris mátrix 10. ábra szerinti particionálása alapján egyszerűen meghatározható. Azonban a következő tétel alapján az is belátható, hogy a létrejött N mátrix Toeplitz-típusu.

4.2. TÉTEL:

Legyen C egy n-edrendű cirkuláris mátrix és p valamint q olyan pozitív egész számok, hogy $(p+q) \leq n$ teljesüljön. Particionáljuk a C-t a következőképpen:

$$C = \begin{bmatrix} \begin{matrix} C_1 & C_2 \end{matrix} \\ \begin{matrix} C_3 & C_4 \end{matrix} \end{bmatrix} \begin{matrix} p \\ q \end{matrix}$$

$\begin{matrix} q & p \end{matrix}$

13. ábra

és készítsük el a

$$T = \begin{bmatrix} & & & & \\ & c_4 & & & c_3 & \\ & & & & & \\ - & - & - & - & - & - \\ & c_2 & & & c_1 & \\ & & & & & \\ & & & & & \end{bmatrix} \begin{matrix} & & & & q \\ & & & & \\ & & & & \\ & & & & \\ & & & & p \\ & & & & \\ & & & & \end{matrix}$$

p q

14. ábra

$(p+q)$ -adrendű mátrixot. Ekkor a kapott T mátrix Toeplitz-típusú mátrix lesz.

BIZONYÍTÁS:

A T definíciója alapján nyilvánvaló, hogy minden T -beli elemnek van egy - vele megegyező - C -beli megfelelője. Vizsgáljuk meg, hogy mi a hozzárendelési szabály, azaz az egyes T -beli elemeknek mi a C -beli "őse".

Osszuk fel a T -t négy blokkra a következőképpen:

$$T = \begin{bmatrix} 1 & 4 \\ 2 & 3 \end{bmatrix} \begin{matrix} q \\ p \end{matrix}$$

p q

15. ábra

és tekintsünk egy T -beli $t_{i,j}$ elemet.

A T mátrix definíciója alapján:

~ ha a $t_{i,j}$ az 1. blokkban van $/0 < i \leq q, 0 < j \leq p/$,
akkor az eredeti C mátrixban $(i+n-q, j+n-p)$ volt
az indexe, azaz

$$t_{i,j} = c_{i+n-q, j+n-p}$$

Mivel azonban a C cirkuláris, így a C -beli elem
további C -beli elemekkel egyenlő, s ezért érvényes a
következő képlet:

$$\begin{aligned} t_{i,j} &= c_{i+n-q, j+n-p} = c_{\underline{i+n-q-j-n+p}} = \\ &= c_{\underline{i-j+p-q}} \end{aligned}$$

- ha a $t_{i,j}$ a 2. blokkban van $/q < i \leq p+q, 0 < j \leq p/$,
akkor az előzőekhez hasonlóan felírhatjuk a következő egyenlőséget:

$$t_{i,j} = c_{i-q, j+n-p} = c_{\underline{i-q-j-n+p}} = c_{\underline{i-j+p-q}},$$

- = ha $t_{i,j}$ a 3. blokkbeli $/q < i \leq p+q, p < j \leq p+q/$

$$t_{i,j} = c_{i-q, j-p} = c_{\underline{i-j+p-q}},$$

- ha pedig $t_{i,j}$ a 4. blokkban van $/0 < i \leq q, p < j \leq p+q/$

$$t_{i,j} = c_{i+n-q, j-p} = c_{\underline{i-j+p-q}}.$$

Tehát függetlenül attól, hogy $t_{i,j}$ melyik blokkban van, érvényes a

$$/4.24./ \quad t_{i,j} = c_{\underline{i-j+p-q}}$$

képlet.

Legyen $0 < i, j, k, l \leq p+q$ és

$$/4.25./ \quad i-j = k-l.$$

$$\begin{array}{ccccccc} & & /4.24./ & & /4.25./ & & /4.24./ \\ /4.26./ & t_{i,j} & = & c_{\underline{i-j+p-q}} & = & c_{\underline{k-l+p-q}} & = t_{k,l} \end{array}$$

A /4.26./ egyenlőség azonban éppen azt bizonyítja, hogy a T Toeplitz-típusu mátrix.

Q.e.d.

A most bebizonyított 4.2. tétel alapján az N mátrix is Toeplitz-típusu, mert kielégíti a tétel feltételeit. /Az N a /4.12./ alapján a cirkuláris B mátrixból pontosan úgy keletkezik, mint a 4.2. tételben a T mátrix a C-ből./

V. A z vektor meghatározása

A z vektort a $B \cdot \underline{y}$ szorzattal definiáltuk. A B mátrix cirkuláris tulajdonságát kihasználva és a korábbi segédtételek eredményeit felhasználva ebben a lépésben is tudunk egy - a hagyományos n^2 műveletigényü mátrix-vektor szorzásnál - hatékonyabb eljárást alkalmazni.

Legyen b a II. lépésben már meghatározott B mátrix első oszlopvektora. Ekkor

$$\begin{array}{ccccccc} & & K3.1. & & K3.6. & & \\ /4.27./ & \underline{z} = B \cdot \underline{y} & = & \underline{b} \circledast \underline{y} & = & \text{IDFT}(\text{DFT}(\underline{b}) \circledast \text{DFT}(\underline{y})) & . \end{array}$$

Ha ebben a képletben a szükséges DFT-k és IDFT végrehajtására az FFT algoritmust használjuk, akkor a $\underline{z}$ szorzatvektor meghatározásához szükséges műveletigény

$$3n \log n + n$$

lesz.

VI. A $\underline{w}$ kiszámítása

A $\underline{w}$ -t a /4.13./

$$R\underline{w} = \underline{s}$$

lineáris egyenletrendszer megoldásaként kapjuk. A korábbiakban már beláttuk, hogy az R együtthatómátrix nonszinguláris, és így a $\underline{w}$ -t egyszerűen megkaphatjuk a Gauss eliminációs algoritmus alkalmazásával $(p+q)^3/3$ művelet végrehajtásával.

Azonban az eljárásunknak ebben a részében is tudunk egy hatékonyságot javító algoritmust alkalmazni. A III. és IV. lépésben meghatározott M és N mátrixokról beláttuk, hogy $(p+q)$ -adrendű Toeplitz-típusu mátrixok, így a 3.3. lemma alapján az M és N mátrixok különbségeként definiált R mátrix is Toeplitz-típusu lesz.

Igy a /4.13./ egy olyan lineáris egyenletrendszer a $\underline{w}$ -re, melynek együtthatója nemszinguláris Toeplitz-típusu mátrix, s ezért a - 4.1. pontban említett és jelenleg az egész /4.2./ egyenletrendszer megoldására is alkalmazott - L-T-Z algoritmus felhasználásával a megoldásvektor $3(p+q)^2$ művelettel meghapható minden olyan esetben, amikor az R erősen nemszinguláris. /Az együtthatómátrix erősen nemszinguláris volta az L-T-Z algoritmus alkalmazhatóságának szükséges feltétele./

VII. Az $\underline{x}$ megoldásvektor meghatározása

A /4.7./ képlet alapján ebben a lépésben már csak a $B \cdot \underline{c}$ mátrix-vektor szorzást kell elvégeznünk, s a kapott szorzatvektor és a $\underline{z}$ összegeként a kiindulási /4.2./ egyenletrendszer megoldását kapjuk.

A B cirkuláris voltát kihasználva a $B \cdot \underline{c}$ szorzást is egyszerűsíthetjük. Ugyanis

$$\text{/4.28./} \quad \begin{array}{c} \text{K3.1.} \\ B \cdot \underline{c} \end{array} = \underline{b} \circledast \underline{c} \quad \begin{array}{c} \text{L3.6.} \\ = \end{array} \text{IDFT}(\text{DFT}(\underline{b}) * \text{DFT}(\underline{c})) .$$

A kapott képlet alapján - az FFT algoritmus alkalmazásával - a $B \cdot \underline{c}$ szorzat meghatározásának teljes műveletigénye

$$3n \log n + n$$

lesz.

Ezek után már csak n összeadást kell végrehajtanunk az $\underline{x}$ megoldásvektor meghatározásához.

Igy, ha a 4.2. részben kidolgozott "hétlépéses" eljárás egyes lépéseiben a most megadott algoritmusokat alkalmazzuk, akkor teljesülnek a következő tétel állításai:

4.3. TÉTEL:

- i, az eljárás a $T\underline{x} = \underline{y}$ nemszinguláris Toeplitz-típusú szalagmátrix együtthatóju lineáris egyenletrendszer megoldását számítja ki,
- ii, a megoldáshoz szükséges műveletek száma:

$$8n \log n + 3(p+q)^2 + (p^2+q^2)/2 + O(n) ,$$

ill. az eljárás VI. lépésében az L-T-Z algoritmus helyett Gauss eliminációt használva:

$$8n \log n + (p+q)^3/3 + (p^2+q^2)/2 + O(n) ,$$

iii, az eljárás helyigénye $O(n)$

A helyigény azért lesz csak $O(n)$ mert az eljárás során csak cirkuláris és Toeplitz-mátrixokat használunk, s így egyetlen mátrix teljes tárolására sincs szükség, minden esetben elegendő egyetlen oszlop vagy egy sor és egy oszlop elemeinek a tárolása.

A megadott módszer "gyorsaságát", azaz a viszonylag kis műveletigényt az biztosítja, hogy a szükséges diszkrét Fourier transzformációkat és az inverz transzformációkat is a gyors Fourier transzformációs algoritmussal végezzük. Ezen algoritmus használatának szükséges feltétele, hogy a transzformálandó vektor komponenseinek száma kettő hatványa legyen, azaz a megoldandó egyenletrendszer egyenleteinek a számát kettőhatványnak kell választani. Ez azonban a felmerülő gyakorlati alkalmazások esetén nem jelent erős megkötést. /Pl. különböző mintavételeknél a minták számát tetszőlegesen választhatjuk meg./ Ha azonban n mégsem választható meg kettőhatványnak, akkor az eljárás megfelelő kiterjesztésével lehet meghatározni a keresett megoldást. A módszer kiterjesztését a nem kettőhatvány esetre a 6.2. rész tartalmazza.

4.4. További műveletigény csökkentés minimális memória- igény növeléssel

Az előző részben megadott eljárásban - nem túl jelentős memóriaigény növeléssel - a megoldáshoz szükséges műveletek számát mintegy 20 %-kal tovább tudjuk csökkenteni a következőkben ismerttetett módon.

Az eljárás II. lépésében a B inverz mátrix meghatározása a feladat. A B cirkularitása miatt csupán az első oszlopot számítjuk ki a

$$\underline{b} = \text{IDFT} \left(\underline{1} \begin{matrix} * \\ * \end{matrix} \text{DFT}(\underline{t}) \right)$$

képlettel három lépésben:

- | | | |
|------|---|---------------|
| i, | $\text{DFT}(\underline{t})$ | kiszámítása |
| ii, | $\underline{1} \begin{matrix} * \\ * \end{matrix} \text{DFT}(\underline{t})$ | meghatározása |
| iii, | $\text{IDFT}(\underline{1} \begin{matrix} * \\ * \end{matrix} \text{DFT}(\underline{t}))$ | kiszámítása |

Az ii, lépés után kapott érték a 3.5. lemma alapján nem más, mint a $\text{DFT}(\underline{b})$, ugyanis

$$\underline{b} = \text{IDFT} \left(\underline{1} \begin{matrix} * \\ * \end{matrix} \text{DFT}(\underline{t}) \right) \stackrel{\text{L3.5.}}{\Longleftarrow} \text{DFT}(\underline{b}) = \underline{1} \begin{matrix} * \\ * \end{matrix} \text{DFT}(\underline{t})$$

Ezen észrevétel alapján érhetjük el a műveletigény csökkentését a következő módon:

Az eljárás II. lépésének végrehajtása közben tároljuk el egy n -elemű tömbbe az

$$\underline{1} * \text{DFT}(\underline{t}) \quad \left[= \text{DFT}(\underline{b}) \right]$$

vektor elemeit.

Az eljárás V. lépésében a $\underline{z}$ vektort kell meghatároznunk a

$$\underline{z} = \text{IDFT}(\text{DFT}(\underline{y}) * \text{DFT}(\underline{b}))$$

képlet alapján. Mivel azonban a korábbi számítások során már meghatározott és eltárolt $\text{DFT}(\underline{b})$ érték a rendelkezésünkre áll, így a $\underline{z}$ meghatározásának menete során a $\text{DFT}(\underline{b})$ érték kiszámításától eltekinthetünk, s így a szükséges műveletigény az eljárás V. lépésében nem $3n \log n + n$, hanem csak

$$2n \log n + n$$

lesz.

Teljesen hasonlóan tudunk a VII. pont végrehajtása-
kor is - szintén a rendelkezésre álló $DFT(\underline{b})$ segít-
ségével - további $n \cdot \log n$ műveletigény csökkentést el-
érni.

Igy tehát a megoldási algoritmus teljes művelet-
igényét

$$8n \log n + 3(\underline{p} + \underline{q})^2 + (\underline{p}^2 + \underline{q}^2)/2 + O(n) \text{ -ről}$$

$$/4.29./ \quad \underline{6n \log n + 3(\underline{p} + \underline{q})^2 + (\underline{p}^2 + \underline{q}^2)/2 + O(n)}$$

értékre tudtuk redukálni a tárolóigény minimális /n-
szóval történő/ növelésével.

4.5. Az eljárás folyamatábrája

Ebben a részben a $T\underline{x} = \underline{y}$ lineáris egyen-
letrendszer megoldására kidolgozott algoritmus folya-
matábráját adjuk meg a következő jelölések alkalmazá-
sával:

$$\underline{a} = (a_0, a_1, \dots, a_p, 0, \dots, 0) \quad \text{- a } T \text{ első sorát,}$$

$$\underline{d} = (d_0, d_1, \dots, d_q, 0, \dots, 0) \quad \text{- a } T \text{ első osz-}$$

lopát,

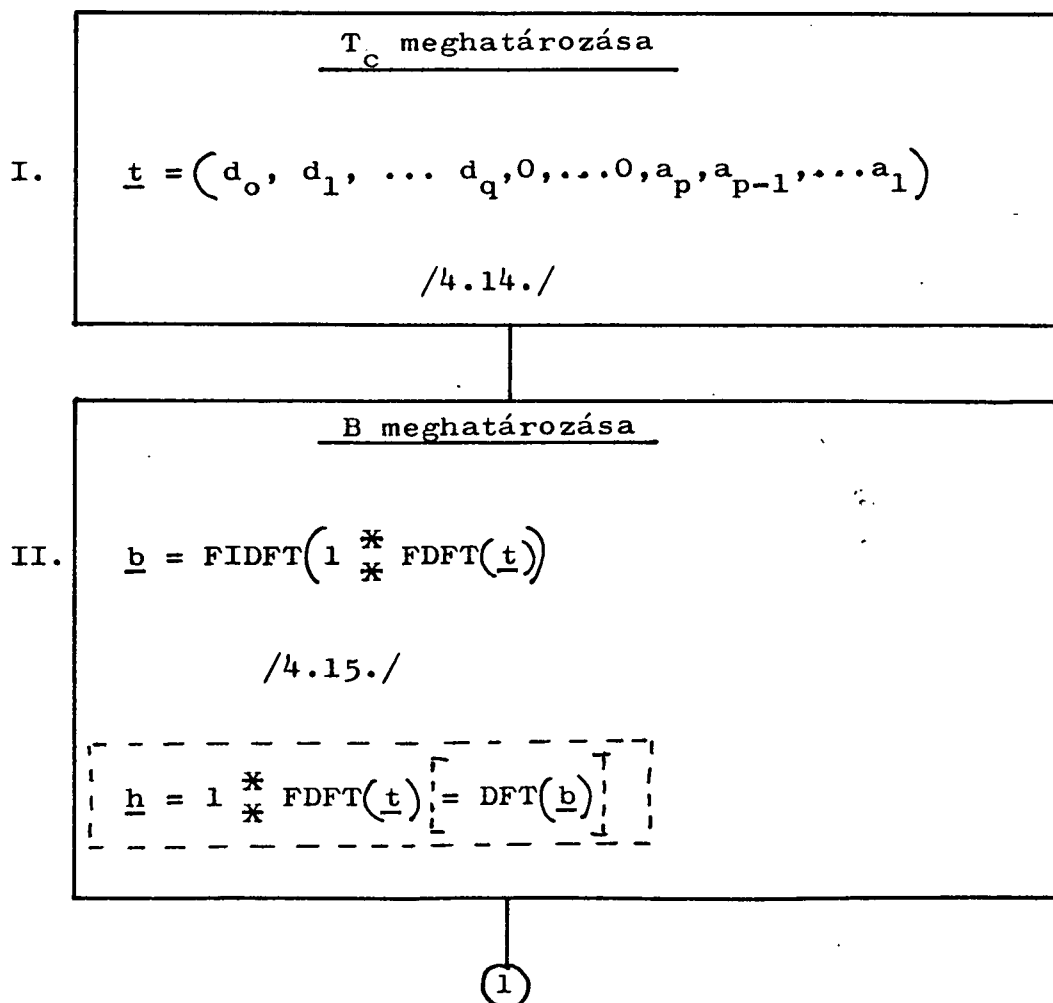
$$\underline{t} = (t_0, \quad . \quad . \quad . \quad t_{n-1}) \quad \text{- a } T_c \text{ első osz-}$$

lopát,

$\underline{b} = (b_0, \dots, b_{n-1})$ - a B első oszlopát,
 $\underline{g} = (g_0, \dots, g_{p-1})$ - a G^{-1} első oszlopát,
 $\underline{f} = (f_0, \dots, f_{q-1})$ - az F^{-1} első sorát,
 $\underline{h} = (h_0, \dots, h_{n-1})$ - munkavektor a $DFT(\underline{b})$
 tárolására.

FOLYAMATÁBRA

Input: $\underline{a}, \underline{d}, \underline{y}$



①

M meghatározása

G^{-1} meghatározása

/G első oszlopa = $(a_p, \dots a_1)/$

III.a

$$g_0 = 1/a_p$$

$$g_m = -g_0 \sum_{k=1}^m a_{p-k} g_{m-k} \quad 0 < m < p$$

/4.16./

III.

F^{-1} meghatározása

/F első sora = $(d_q, \dots d_1)/$

III.b

$$f_0 = 1/d_q$$

$$f_m = -f_0 \sum_{k=1}^m d_{q-k} f_{m-k} \quad 0 < m < q$$

/4.16./

/10. ábra/

N meghatározása

IV.

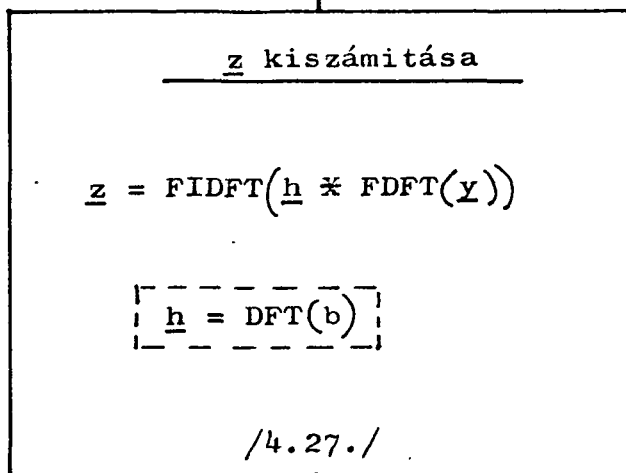
$$n_{i,j} = b_{\underline{i-j+p-q}}$$

/4.24./

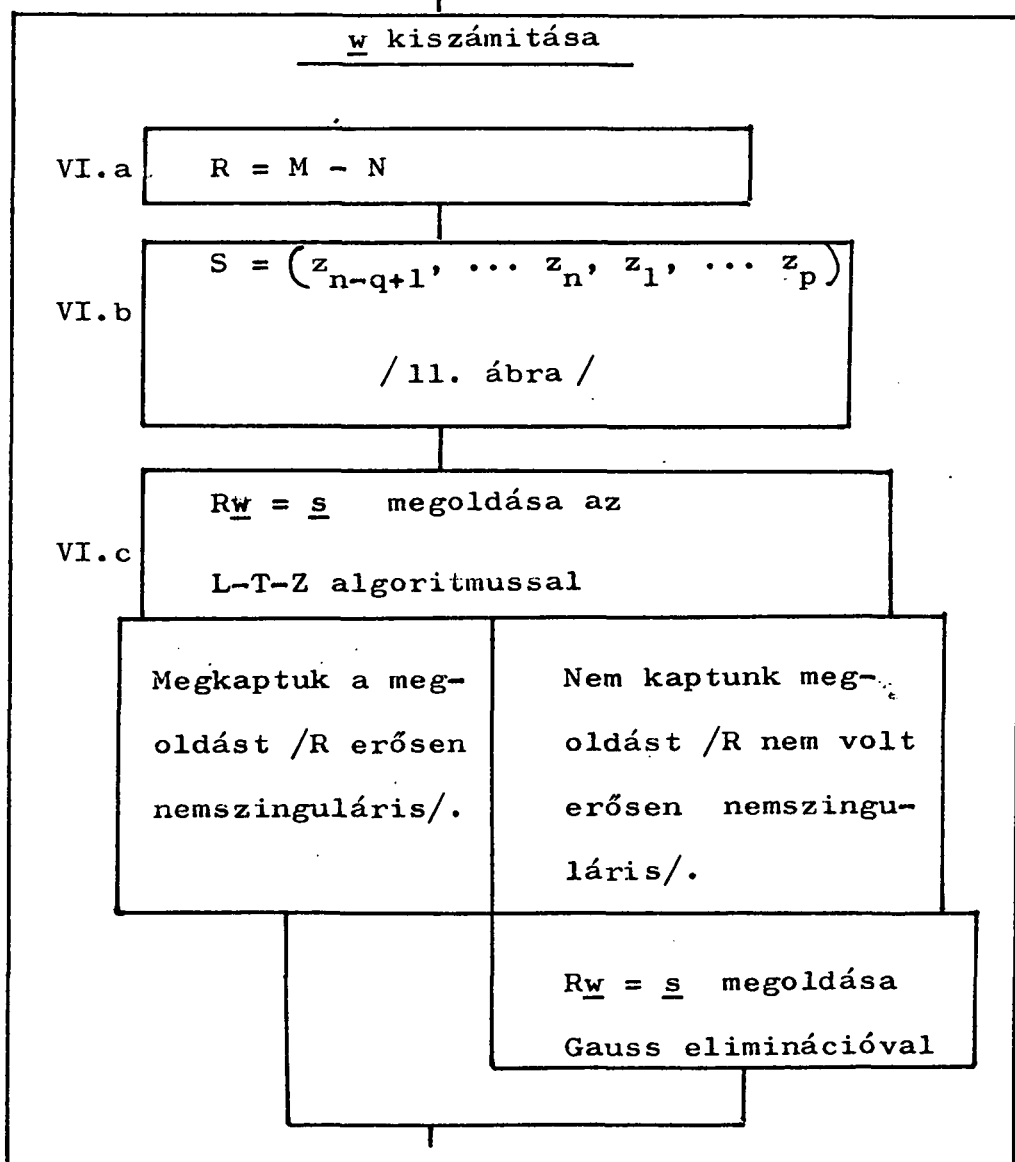
②

2

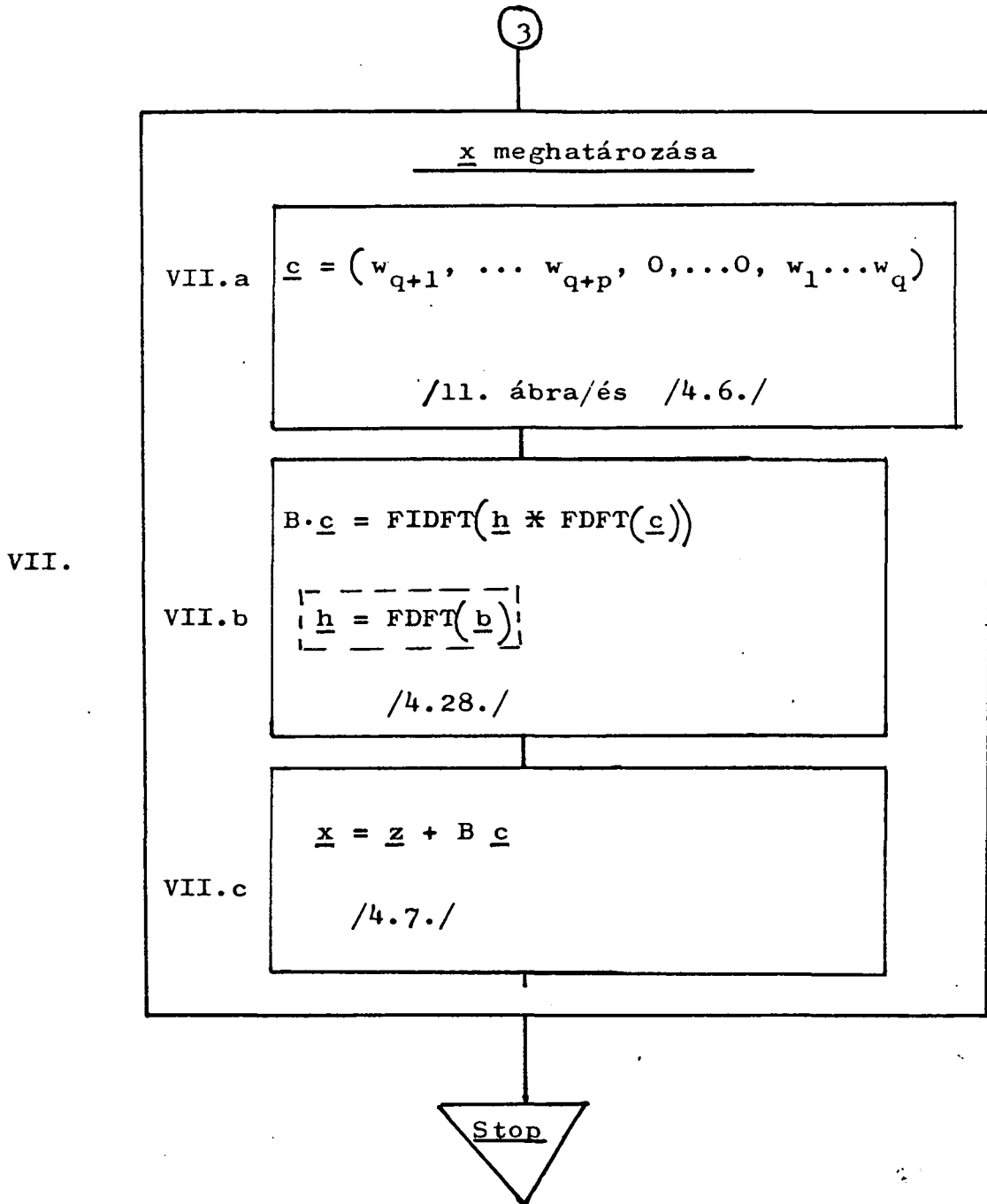
V.



VI.



3



Output: az $\underline{x}$ megoldásvektor

16. ábra

4.6. Az algoritmus bemutatása egy konkrét példán

A most következő részben a kidolgozott megoldási algoritmus menetét egy konkrét példán is bemutatjuk.

Legyen a megoldandó egyenletrendszer

$$\begin{aligned}x_1 + x_2 &= 2 \\ /4.30./ \quad 2x_1 + x_2 + x_3 &= 4 \\ 2x_2 + x_3 + x_4 &= 4 \\ 2x_3 + x_4 &= 3\end{aligned}$$

Az egyenletrendszer egy megoldása:

$$/4.31./ \quad x_1 = x_2 = x_3 = x_4 = 1.$$

Mivel a rendszer determinánsának értéke -1 , így a /4.31./ az egyenletrendszer egyetlen helyes megoldása.

Nézzük most meg, hogy az új eljárás segítségével hogyan kaphatjuk meg a /4.30./ lineáris egyenletrendszer megoldását. A /4.30./ alapján, /az előző részben használt jelöléseket alkalmazva/:

$$T = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 0 \\ 0 & 2 & 1 & 1 \\ 0 & 0 & 2 & 1 \end{bmatrix} ; \quad \underline{y} = \begin{bmatrix} 2 \\ 4 \\ 4 \\ 3 \end{bmatrix}$$

17. ábra

és $p = q = 1$.

A T_c , T_b mátrixok és a $\underline{t}$ vektor pedig a következők lesznek:

$$T_c = \begin{bmatrix} 1 & 1 & 0 & 2 \\ 2 & 1 & 1 & 0 \\ 0 & 2 & 1 & 1 \\ 1 & 0 & 2 & 1 \end{bmatrix} ; \quad T_b = \begin{bmatrix} 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \text{ és } \underline{t} = \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix}$$

18. ábra

B-nek, azaz T_c inverzének az első oszlopát a T_c első oszlopvektorából, $\underline{t}$ -ből a

$$\underline{b} = \text{IDFT}\left(\underline{1} * \text{DFT}(\underline{t})\right)$$

képlet alapján számítjuk ki a következő lépésekben:

$$\underline{t} = \left[(1,0); (2,0); (0,0); (1,0) \right]$$

$$\text{DFT}(\underline{t}) = \left[(4,0); (1,1); (-2,0); (1,-1) \right]$$

$$\left[\underline{h} = \right] \left(\underline{1} * \text{DFT}(\underline{t}) \right) = \left[(1/4,0); (1/2,-1/2); (-1/2,0); (1/2,1/2) \right]$$

/Ezt a vektort eltároljuk egy $\underline{h}$ vektorba, mert erre még szükségünk lesz, hiszen ez nem más, mint a $\text{DFT}(\underline{b})$./

$$\text{IDFT}(\underline{1} * \text{DFT}(\underline{t})) = \left[(3/16,0); (-1/16,0); (-5/16,0); (7/16,0) \right] = \underline{b}$$

és így a B mátrix a következő lesz:

$$B = \frac{1}{16} \cdot \begin{bmatrix} 3 & 7 & -5 & -1 \\ -1 & 3 & 7 & -5 \\ -5 & -1 & 3 & 7 \\ 7 & -5 & -1 & 3 \end{bmatrix}$$

19. ábra

A T_b mátrix alapján

$$F = \begin{bmatrix} 1 \end{bmatrix} \quad \text{és} \quad G = \begin{bmatrix} 2 \end{bmatrix} ,$$

s ezért

$$F^{-1} = \begin{bmatrix} 1 \end{bmatrix} \quad ; \quad G^{-1} = \begin{bmatrix} 1/2 \end{bmatrix} .$$

Igy

$$M = \begin{bmatrix} 0 & 1/2 \\ 1 & 0 \end{bmatrix}$$

20. ábra

A B megfelelő particionálása után:

$$B = \left[\begin{array}{c|c|c|c} 3 & 7 & -5 & -1 \\ \hline -1 & 3 & 7 & -5 \\ \hline -5 & -1 & 3 & 7 \\ \hline 7 & -5 & -1 & 3 \end{array} \right]$$

21. ábra

az

$$N = \frac{1}{16} \begin{bmatrix} 3 & 7 \\ -1 & 3 \end{bmatrix}$$

22. ábra

lesz.

$$R=M-N = \frac{1}{16} \cdot \begin{bmatrix} -3 & 1 \\ 17 & -3 \end{bmatrix}$$

23. ábra

A z vektort a

$$\underline{z} = \text{IDFT}(\text{DFT}(\underline{y}) * \text{DFT}(\underline{b}))$$

képlettel számítjuk ki a következő lépésekben:

$$\underline{y} = [(2,0); (4,0); (4,0); (3,0)]$$

$$\text{DFT}(\underline{y}) = [(13,0); (-2,1); (-1,0); (-2,-1)]$$

$$\text{DFT}(\underline{b}) = [(1/4,0); (1/2,-1/2); (-1/2,0); (1/2,1/2)] -$$

- ezt most nem kell
számolnunk!

$$(\text{DFT}(\underline{y}) * \text{DFT}(\underline{b})) = [(13/4,0); (-1/2,3/2); (1/2,0); (-1/2,-3/2)]$$

$$\text{IDFT}(\text{DFT}(\underline{y}) * \text{DFT}(\underline{b})) = [(11/16,0); (23/16,0); (19/16,0); (-1/16,0)] = \underline{z}.$$

Igy a /4.13./ -nak megfelelő Toeplitz egyenletrendszer a következő lesz:

$$/4.32./ \quad \frac{1}{16} \begin{bmatrix} -3 & 1 \\ 17 & -3 \end{bmatrix} \begin{bmatrix} v \\ u \end{bmatrix} = \frac{1}{16} \begin{bmatrix} -1 \\ 11 \end{bmatrix}$$

Mivel a /4.32./ egyenletrendszer R együtthatómátrixa erősen nonszinguláris /a főminorok értéke: -3 és -8/ így a megoldást az L-T-Z algoritmussal számíthatjuk ki.

A kapott megoldás

$$v = 1 \quad \text{és} \quad u = 2 .$$

Igy

$$c = (2, 0, 0, 1) .$$

A $B \cdot \underline{c}$ szorzatot az

$$\text{IDFT}(\text{DFT}(\underline{b}) * \text{DFT}(\underline{c}))$$

képlet alapján számítjuk ki.

$$\text{DFT}(\underline{c}) = [(3, 0); (2, -1); (1, 0); (2, 1)]$$

$$\text{DFT}(\underline{b}) = [(1/4, 0); (1/2, -1/2); (-1/2, 0); (1/2, 12)]$$

$$/ = \underline{h} /$$

$$\text{DFT}(\underline{c}) * \text{DFT}(\underline{b}) = [(3/4, 0); (1/2, -3/2); (-1/2, 0); (1/2, 3/2)]$$

$$\text{IDFT}(\text{DFT}(\underline{c}) * \text{DFT}(\underline{b})) = \left[(5/16, 0); (-7/16, 0); (-3/16, 0); (17/16, 0) \right] =$$

$$= \underline{B} \cdot \underline{c} \quad .$$

Ezek után a megoldás:

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \underline{z} + \underline{B} \cdot \underline{c} = \frac{1}{16} \begin{bmatrix} 11 \\ 23 \\ 19 \\ -1 \end{bmatrix} + \frac{1}{16} \begin{bmatrix} 5 \\ -7 \\ -3 \\ 17 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Azaz $x_1 = x_2 = x_3 = x_4 = 1 \quad .$

5. A JELENLEG HASZNÁLT ALGORITMUSOK ÉS A BEMUTATOTT ELJÁRÁS ÖSSZEHASONLÍTÁSA

A

$$T\underline{x} = \underline{y}$$

Toeplitz-típusú szalagmátrix együtthatóju lineáris egyenletrendszerek megoldására jelenleg a Gauss eliminációt vagy az L-T-Z algoritmust használják. Hasonlítsuk össze ezeket az algoritmusokat a kidolgozott eljárással a szükséges műveletigény alapján. Az egyes értékek a következők:

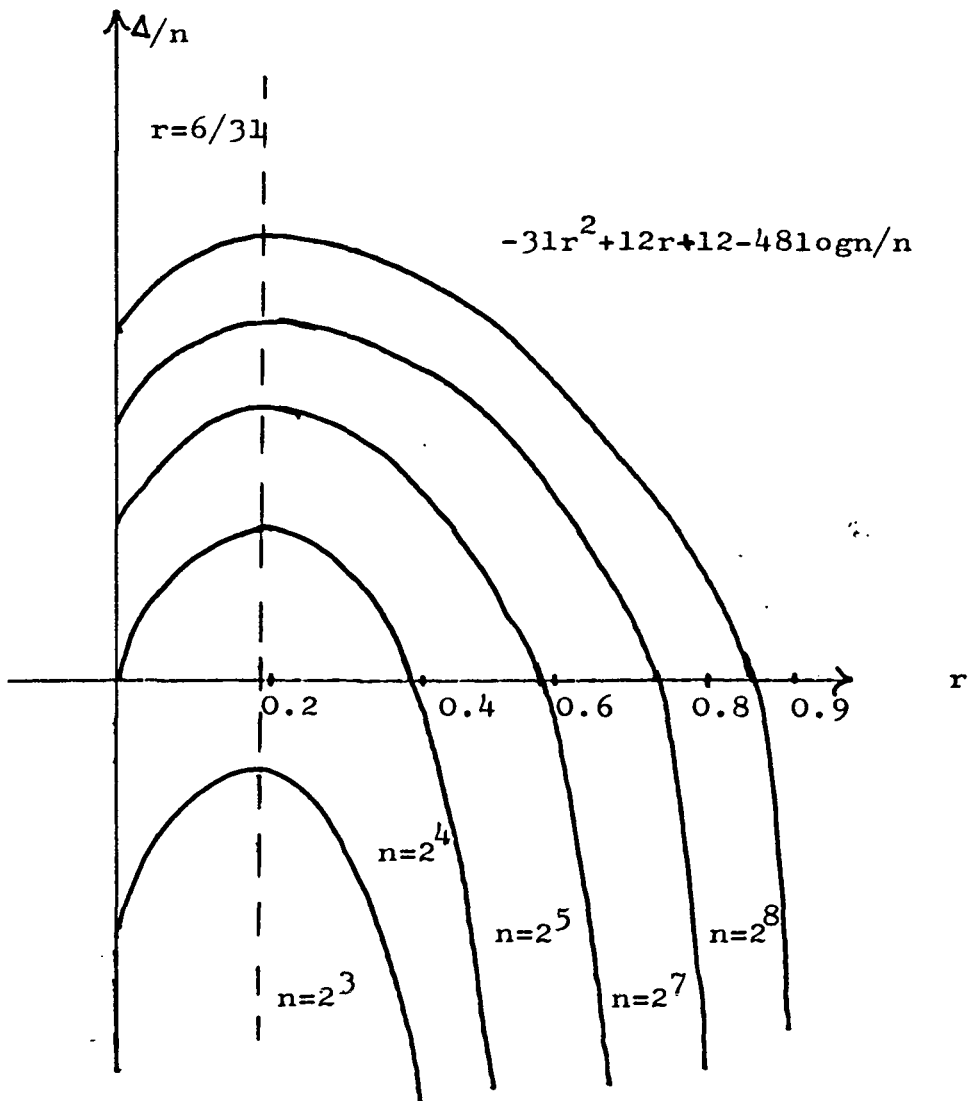
/5.1./ Gauss elimináció : $n(p+q)^2/4$

/5.2./ L-T-Z algoritmus : $1,5n^2 - 1,5(p+q)^2/4 + 3n(p+q)/2 + O(n)$

/5.3./ az új eljárás : $6n \log n + 3(p+q)^2 + (p^2 + q^2)/2 + O(n)$.

Jelöljük d -vel a T együtthatómátrix szalagszélességét $(p+q)$ -t, valamint r -rel a relatív szalagszélességet (d/n) -t és vizsgáljuk meg, hogy milyen n és d értékek mellett lesz az eljárásunk hatékonyabb az L-T-Z algoritmusnál.

A 24. ábrán a megadott eljárásunk és az L-T-Z algoritmus relatív műveletigény-különbségét, $(\sqrt{5.3./-}/5.2./)/n$ -t ábrázoltuk a relatív sáv szélesség függvényében különböző n értékek esetén. Az ábráról jól látszik, hogy az $n=16$ -hoz tartozik az első olyan különbséggörbe, amelynek már vannak pozitív értékei is, azaz $n = 2^4$ esetén már van olyan p és q , amelyekre az eljárásunk hatékonyabb lesz az L-T-Z algoritmusnál.



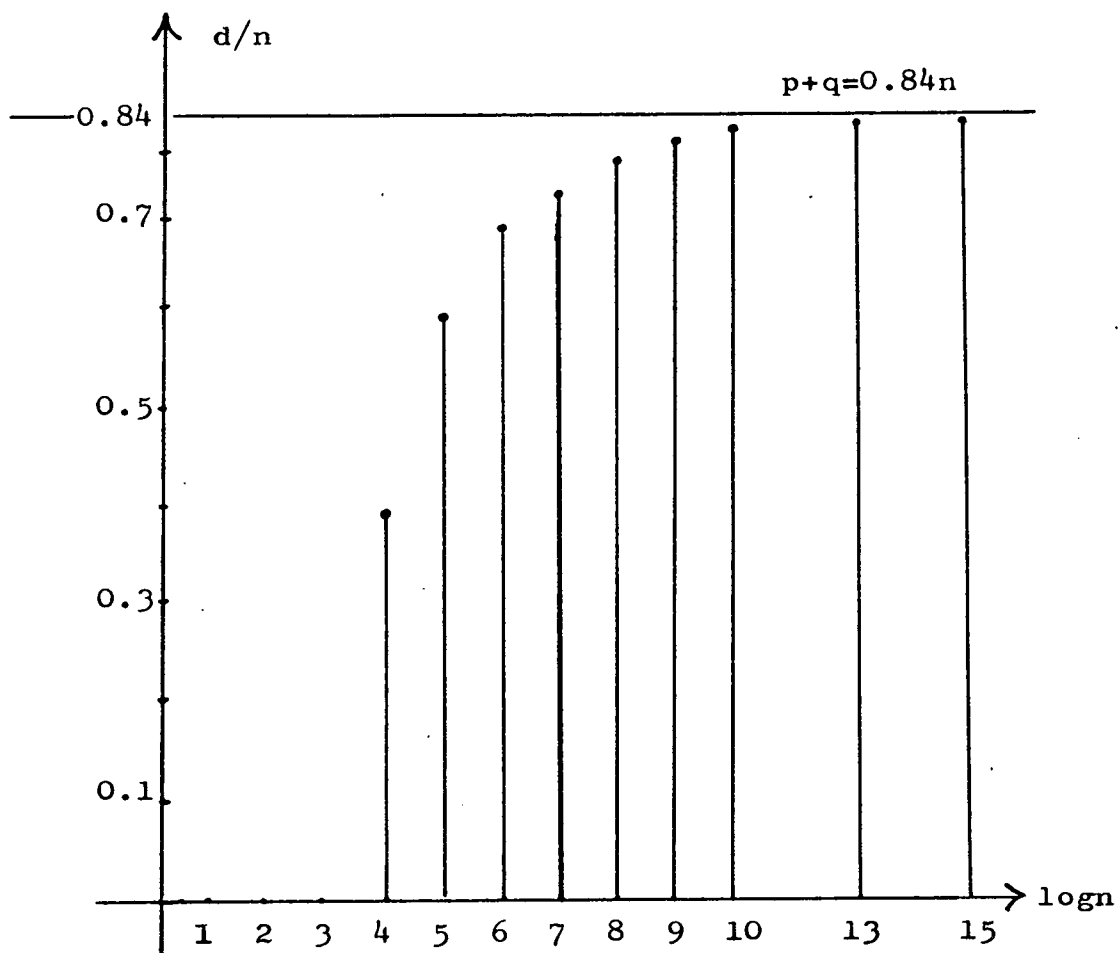
24. ábra

A 25. ábrán az egyes n értékekhez tartozó "jó" relatív sávszélesség-tartományokat ábrázoltuk, azaz az egyes n , ill. $\log n$ pontokhoz tartozó függőleges szakaszok olyan intervallumokat jelölnek, amelyekbe ha belesik a $(p+q)/n$ értéke, akkor az eljárásunkat érdemes alkalmazni. Erről az ábráról jól látszik, hogy $n = 2^4$ - hez tartozik az első függőleges szakasz, mégpedig olyan, hogy az $r \in [0, 0.4]$ pontok tartoznak bele, tehát $n = 16$ esetén, ha $(p+q) \leq 6$, akkor már érdemes az új módszert használni. A 25. ábra függőleges szakaszainak végpontjait az /5.3./ és /5.2./ műveletigény-értékek különbsége alapján származtatott

$$\text{/5.4./} \quad y(n) = \frac{6+2\sqrt{6(21-62 \log n/n)}}{31}$$

függvény értékei szolgáltatják.

Mind az /5.4./ képlet, mind pedig a 25. ábra alapján látható, hogy az n növekedésével a "megfelelő" p és q értékek is nőnek egészen $r = 0,84$ -ig, azaz ha n elég nagy, akkor minden $(p+q) \leq 0,84n$ esetén érdemes a dolgozatban megadott eljárást alkalmazni.



25. ábra

Az /5.3./ és /5.2./ értékek alapján az is látszik, hogy ha n elég nagy és p, q kicsi, akkor igen jelentős lesz a műveletigények eltérése, ugyanis

$$\lim_{\substack{n \rightarrow \infty \\ d \ll n}} \frac{1,5n^2 - 1,5d^2/4 + 1,5nd}{6n \log n + 7d^2/2} = \infty$$

Igy például, ha $n = 2^{12}$ és $p = q = 2^7$, akkor az /5.1./, /5.2./ és /5.3./ érték 2^{26} , $3 \cdot 2^{23}$ és 2^{19} lesz, tehát az eljárásunk a Gauss eliminációnál mintegy 120-szor, az L-T-Z algoritmusnál pedig kb. 50-szer gyorsabb ebben az esetben.

Az eljárásnak - a korábbiaknál kisebb műveletidő szükséglete mellett - további nagy előnye az általános /eliminációs, iterációs/ eljárásokkal szemben, hogy kihasználja az együtthatómátrix Toeplitz tulajdonságát és nem tárolja az egész mátrixot csupán annak első sorát és oszlopát, s így igen komoly tárolóigény csökkenést biztosít. Ez azért jelentős, mert nagyméretű egyenletrendszerek esetén az egész együtthatómátrix már "nem fér el" az operatív memóriában, ezért általában az eliminációs és iterációs eljárások csak disc terület igénybevételeivel hajthatók végre, viszont a disc-hez fordulások miatt lényegesen lassul a megoldás meghatározásának menete. A helyigény az eliminációs és iterációs eljárásoknál $O(n^2)$, míg az új algoritmus esetében ez az érték csak $O(n)$ lesz.

A most megadott eljárás komoly korlátjának tűnhet, hogy szemben a korábbi algoritmusokkal a kiindulási egyenletrendszer megoldhatóságához most nemcsak a T együtthatómátrix nonsingularitása szükséges, hanem a T cirkuláris

kiterjesztésével keletkezett T_c mátrixnak is nemszingulárisnak kell lenni, s így az algoritmus alkalmazhatóságához még egy - általában jelentős műveletigényű - szingularitásvizsgálatot is el kell végeznünk. Azonban a 3.részben bebizonyított 3.7. lemma hatékony eszközt ad a probléma leegyszerűsítésére. Mivel a T_c cirkuláris, így a lemma eredményei alapján egyetlen FFT végrehajtásával el tudjuk végezni a T_c tesztelését.

Itt jegyezzük meg, hogy bár az L-T-Z algoritmus általánosabb eljárás, mivel olyan lineáris egyenletrendszerrek megoldására is alkalmas, melyeknek együtthatómátrixa nem szalagmátrix csupán Toeplitz-típusu, bizonyos esetekben azonban az új algoritmus lesz az általánosabb. Ugyanis az L-T-Z algoritmus alkalmazhatóságának szükséges feltétele, hogy az együtthatómátrix erősen nemszinguláris legyen, míg a megadott eljárásunk alkalmazhatóságához csak a T mátrixnak és a cirkuláris kiterjesztésének, T_c -nek a nemszingularitása szükséges. Így tehát minden olyan esetben, amikor T és T_c nemszinguláris szalagmátrix, azonban nem teljesíti az erős nemszingularitás feltételét, az L-T-Z algoritmus nem alkalmazható, viszont az új eljárással meghatározható a keresett megoldás.

6. TOVÁBBI HATÉKONYSÁGNÖVELÉSI ÉS ÁLTALÁNOSÍTÁSI LEHETŐSÉGEK

6.1. Hatékonyaságnövelés hardware és elméleti uton

A dolgozatban bemutatott eljárás egyes lépéseinek a leírása alapján jól látszik, hogy a megoldás meghatározásához szükséges műveletvégzés szinte kizárólag csak gyors Fourier transzformációk és inverz gyors Fourier transzformációk végrehajtásából áll. Ezért viszont minden, a gyors Fourier transzformációs algoritmus hatékonyságát növelő módszer a transzformációéval majdnem azonos mértékben növeli az eljárásunk hatékonyságát is. Mivel jelenleg a gyors Fourier transzformációt már egyre inkább hardware uton oldják meg [9, 15], s egyre elterjedtebbek a végrehajtási időt nagyságrendekkel csökkentő ún. gyors Fourier transzformációs processzorok [10, 25] és architektúrák [2, 34], ezért ha az eljárásunkban is ilyen módon oldjuk meg a szükséges transzformációk végrehajtását, akkor - a fenti megfontolások értelmében - algoritmusunk hatékonyságában is igen számottevő javulás várható.

A Toeplitz egyenletek megoldása területén elért újabb eredmények felhasználásával a módszerünk haté-



konysága javításának egy másik útja is lehetséges. Sikerült ugyanis J. Jain kettőhatvány elemszámú Toeplitz egyenletrendszer megoldására kidolgozott - az inverz mátrix egy speciális felbontását [23] és a gyors Fourier transzformációs algoritmust felhasználó - módszerének [21] egy olyan általánosítását megadni [7], hogy a kapott eljárás már tetszőleges elemszámú Toeplitz-típusú /nem feltétlenül szalag-/ mátrix együtthatóju lineáris egyenletrendszer megoldására alkalmas. Az eljárás műveletszükségelete a jelenleg használt $3n^2$ műveletigényű L-T-Z algoritmusával szemben $2n^2 + O(n \log n)$. Mivel a Toeplitz-típusú szalagmátrix együtthatóju lineáris egyenletrendszerek megoldására kidolgozott algoritmusunkban a /4.13./ $(p+q)$ -adrendű Toeplitz egyenletrendszer megoldására az L-T-Z algoritmust használtuk, így az újabb eljárás alkalmazása esetén a /4.13./ egyenletrendszert $3(p+q)^2$ művelet helyett $2(p+q)^2 + O((p+q) \log(p+q))$ művelettel tudjuk megoldani, és ez a javulás a teljes algoritmus műveletigényének csökkenésével jár.

6.2. Általánosítási lehetőségek

A dolgozatban megadott megoldási algoritmus alkalmazhatósági körének kiterjesztésére a Toeplitz-

és Hankel mátrixok közötti szoros kapcsolat nyújt további lehetőséget. A [8] dolgozatban megadott eredmények felhasználásával ugyanis a Toeplitz-típusú szalagmátrix együtthatóju lineáris egyenletrendszerekre kidolgozott eljárásunk egyszerűen módosítható úgy, hogy a kapott algoritmus alkalmazható legyen Hankel típusú szalagmátrix együtthatóju lineáris egyenletrendszerek megoldására.

Mivel jelenleg az általános lineáris egyenletrendszerek megoldására alkalmazott programcsomagok általában az egyenletszám harmadik hatványával arányos eliminációs vagy iterációs algoritmusokat használnak, ezért esetleg nagyméretű egyenletrendszerek megoldása esetén érdemes lehet egy "elővizsgálatot" beépíteni a programcsomagba, annak eldöntésére, hogy a megoldandó egyenletrendszer együtthatómátrixa nem Toeplitz vagy Hankel mátrix-e. Mivel ezt az elővizsgálatot $O(n^2)$ egyszerű kivonással el lehet végezni, így az esetlegesen elérhető igen számottevő műveletigény csökkenés miatt ezt a vizsgálatot akkor is érdemes elvégezni, ha csak ritkán találunk Toeplitz vagy Hankel mátrixot.

Ahhoz azonban, hogy tetszőleges alkalmazási területen - ahol esetleg már nem választható meg megfelelően az egyenletek száma - alkalmazható legyen a megoldási módszerünk, meg kell adni az algoritmus kiterjesztését arra az esetre is, ha a megoldandó egyenletrendszer egyenleteinek a száma nem kettőhatvány. Ebben a részben a továbbiakban a kiterjesztési folyamatot és az általánosított algoritmust ismertetjük.

Tekintsük a

$$\text{/6.1./} \quad T\underline{x} = \underline{y}$$

nemszinguláris Toeplitz-tipusu szalagmátrix együtthatóju lineáris egyenletrendszert és tegyük fel, hogy az egyenletek száma nem kettőhatvány. Legyen T rendje n .

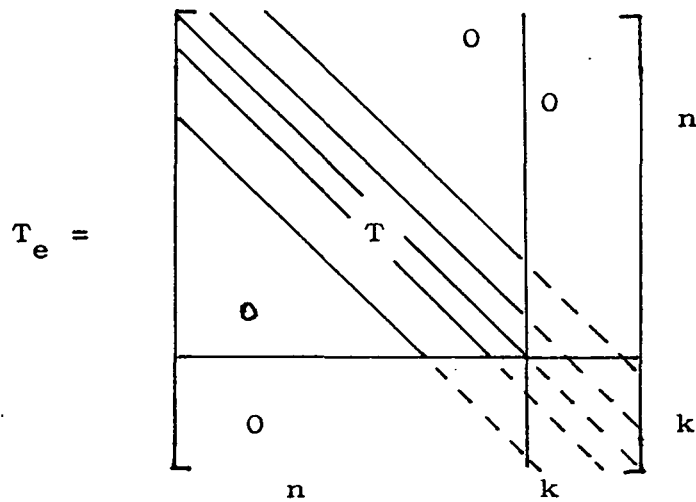
Keressünk egy olyan k számot, amelyre teljesül:

$$\text{i,} \quad \exists t \text{ egész szám:} \quad n + k = 2^t$$

$$\text{ii,} \quad k \geq \max(p, q)$$

iii, k a legkisebb az i. és ii. feltételeket kielégítő számok közül.

Legyen T_e a T mátrix $(n + k)$ -adrendű Toeplitz-típusú kiterjesztése:



26. ábra

A T_e -ben a T nem nulla elemein kívül csak a szaggatott vonallal jelzett sorokbeli elemek lehetnek nem nulla értékűek és a "ferde sorok" mentén továbbra is csupa azonos elem áll.

Az $\underline{x}$ kiterjesztése pedig a következő $(n + k)$ -adrendű vektor legyen:

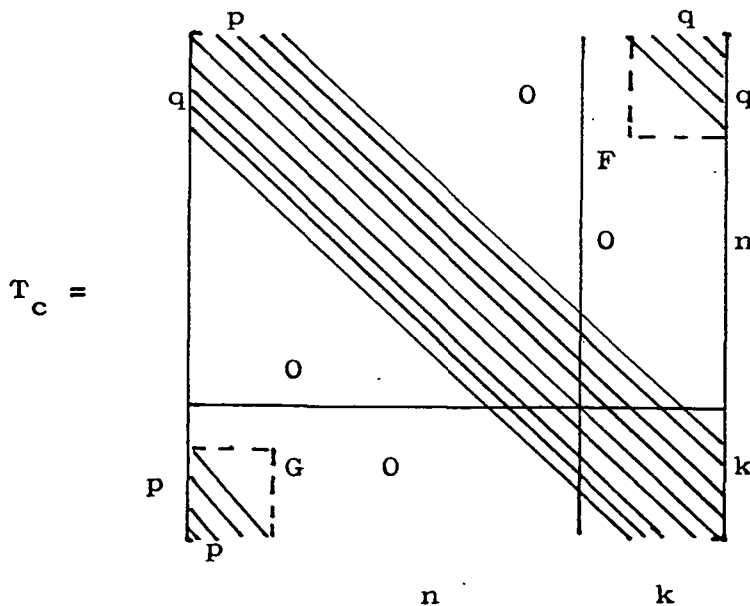
$$/6.2./ \quad \underline{x}_{e_i} = \begin{cases} x_i & 0 \leq i < n \\ 0 & n \leq i < n+k \end{cases}$$

A T_c és T_b mátrixokat T -nek T_e -re való kiterjesztése után a korábbiakhoz hasonlóan definiáljuk /csak most T helyett T_e -re/. Így tehát T_c a T_e mátrix cirkuláris kiterjesztése /most a k definíciója miatt a $p+q < n+k$ feltétel biztosan teljesül/, a T_b -t pedig újra a

$$T_b = T_c - T_e$$

képlettel definiáljuk.

A korábbi jelöléseket használva a T_c a következő alakú lesz:



27. ábra

Az $\underline{y}_e$ legyen a következő $(n + k)$ -elemű vektor:

$$/6.3./ \quad \underline{y}_e = \begin{bmatrix} \underline{y} \\ \vdots \\ \underline{y}_f \end{bmatrix} \begin{matrix} n \\ k \end{matrix},$$

ahol az $\underline{y}_f$ olyan k -elemű vektor, hogy a /6.3./-mal definiált $\underline{y}_e$ -re teljesül a

$$/6.4./ \quad T_c \underline{x}_e = \underline{y}_e$$

egyenlőség.

Tegyük fel, hogy a T_c cirkuláris mátrix nonszinguláris. Így létezik a $B = T_c^{-1}$ inverz mátrix is és a 3.2. lemma alapján a B is cirkuláris.

Particionáljuk a B mátrixot a következőképpen:

$$/6.5./ \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \begin{matrix} n \\ k \end{matrix}$$

$n \qquad k$

A B inverz mátrix létezése miatt a /6.4./ alapján felírhatjuk a következő egyenletet:

$$\text{/6.6./} \quad \underline{x}_e = B \underline{y}_e ,$$

amelynek "blokkosított" alakja:

$$\text{/6.7./} \quad \begin{bmatrix} \underline{x} \\ \underline{0} \end{bmatrix} = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \cdot \begin{bmatrix} \underline{y} \\ \underline{y}_f \end{bmatrix}$$

29. ábra

Elvégezve a kijelölt szorzást, az

$$\text{/6.8./} \quad \underline{x} = B_{11} \underline{y} + B_{12} \underline{y}_f \quad \text{és}$$

$$\text{/6.9./} \quad \underline{0} = B_{21} \underline{y} + B_{22} \underline{y}_f$$

egyenleteket kapjuk.

Tegyük fel, hogy létezik a B_{22}^{-1} inverz mátrix, ekkor /6.9./ alapján:

$$/6.10./ \quad \underline{y}_f = -B_{22}^{-1} B_{21} \underline{y} \quad .$$

Megjegyzés: A B_{22}^{-1} létezésének egy elégséges feltétele, hogy B erősen nonszinguláris legyen, ugyanis a B felbontása alapján látszik, hogy a B_{22} megegyezik a B mátrix bal felső sarkában álló k -adrendű főminorral /a B cirkuláris volta miatt/.

Helyettesítsük be a most kapott $\underline{y}_f$ értéket a /6.8./-ba. Így az

$$/6.11./ \quad \underline{x} = B_{11} \underline{y} - B_{12} B_{22}^{-1} B_{21} \underline{y}$$

képletet kapjuk a keresett $\underline{x}$ megoldásvektorra. Ezen képlet alapján a szükséges műveletek /mátrix szorzás, inverz meghatározás/ elvégzésével a /6.1./ lineáris egyenletrendszer $\underline{x}$ megoldásvektora meghatározható.

A szükséges műveletek száma:

$$/6.12./ \quad n^2(k+1) + 2(n+k)\log(n+k) + 3n\log n + 3k^2 + (n+k) \quad .$$

A számítási algoritmust nem részletezzük, ugyanis az $\underline{x}$ meghatározása lényegesen hatékonyabb lesz, ha a

/6.10./ alapján meghatározott $\underline{y}_f$ vektort közvetlenül a /6.3./-ba helyettesítjük vissza és az $\underline{x}_e$ vektort a /6.6./ képlet alapján határozzuk meg.

Ez azért is előnyös, mert így újra ki tudjuk használni a szorzás egyszerűsítésére a B mátrix cirkuláris voltát. Mivel a B_{22} a B cirkuláris mátrix /6.5./ felbontása alapján Toeplitz-tipusu mátrix, így inverze a Zohar-Trench féle inverziós algoritmus [36] alapján $3k^2$ művelettel meghatározható, feltéve, hogy B_{22} erősen nemszinguláris. Ha B_{22} csak nemszinguláris, akkor Jordan vagy Gauss elimináció segítségével határozzuk meg a B_{22}^{-1} inverz mátrixot $/k^3$ művelettel/. A módosított eljárás algoritmusában is alkalmazhatjuk a 4.4. részben használt összefüggést, azaz a $DFT(\underline{b})$ -t most is tároljuk és ez, ebben az esetben $(n+k)\log(n+k)$ műveletigény csökkenést eredményez.

Mindezek alapján a kiterjesztett algoritmus fő lépései a következők lesznek:

1. Megfelelő k érték keresése

- i. $k + n = 2^t$
- ii. $k \geq \max(p, q)$
- iii. $\min(k)$

műveletigény: $\log(n+k)$.

2. T_c meghatározása

Mivel T_c cirkuláris, csak egy oszlopot kell meghatározunk.

Műveletvégzés: $n + k$ értékadás .

3. $A \ B = T_c^{-1}$ mátrix meghatározása

$$\underline{b} = \text{FIDFT} \left[\underline{1} \begin{matrix} * \\ * \end{matrix} \text{FDFT}(\underline{t}) \right] .$$

Műveletigény: $2(n+k) \log(n+k) + O(n+k)$.

4. Y_f számítása

4.a. B_{22}^{-1} meghatározása

Zohar-Trench algoritmus vagy Jordan elimináció .

Műveletigény: $3k^2$ vagy k^3 .

4.b. $C = B_{22}^{-1} \cdot B_{21}$

mátrix szorzás .

Műveletigény: $k^2 n$.

4.c. $C \cdot y$ kiszámítása

Mátrix-vektor szorzás .

Műveletigény: nk .

4.d. y_f meghatározása

$$y_f = -C \cdot y \quad .$$

5. y_e meghatározása

$$y_e = \begin{bmatrix} y, y_f \end{bmatrix} \quad .$$

Műveletvégzés: $n + k$ értékadás .

6. x meghatározása

$$\underline{x}_e = B \cdot y_e$$

$$\underline{x} = \text{FIDFT} \left[\text{FDFT}(\underline{b}) * \text{FDFT}(y_e) \right] \quad .$$

/FDFT($\underline{b}$) ismert/

Műveletigény: $2(n+k) \log(n+k) + O(n+k)$.

A megadott általánosított algoritmus teljes műve-
letigénye az egyes lépésekben megadott értékek alapján:

$$\frac{4(n+k)\log(n+k) + nk + k^2(3+n) + o(n+k)}{\quad} \quad \text{vagy}$$

$$\frac{4(n+k)\log(n+k) + nk + k^3 + k^2n + o(n+k)}{\quad}$$

a 4. lépéstől függően.

7. AZ ELJÁRÁS SZÁMITÓGÉPES IMPLEMENTÁCIÓJA

A Toeplitz-tipusu szalagmátrix együtthatóju lineáris egyenletrendszereknek a - dolgozatban megadott algoritmus alapján történő - megoldására készült FORTRAN nyelvű program listáját a 2. melléklet tartalmazza.

Az eljárás használatakor egy főmodulban kell megadni az egyenletrendszer paramétereit /T első sorának első $(p+1)$ és első oszlopának első $(q+1)$ elemét, valamint az y vektort/. Ebben a modulban kell aktivizálni a TOPLEQ szubrutint, amely az egyenletrendszer megoldását szolgáltatja. Ez a rutin aktivizálja futása közben az LTZALG nevű szubrutint, amely a /4.13./ - $(p+q)$ egyenletből álló - egyenletrendszer megoldását számítja ki az L-T-Z algoritmus alapján. A rutin futása során ellenőrizzük, hogy az együtthatómátrix kielégíti-e az erős nonsingularitás feltételét, s ha ez nem teljesül, akkor kilépünk az LTZALG rutunból és aktivizáljuk a GAUSEL szubrutint, amely a /4.13./ egyenletrendszer megoldását Gauss eliminációval határozza meg.

A TOPLEQ rutin használja még az FFT diszkrét Fourier transzformációt gyors algoritmussal számító szub-

rutint. Mivel az inverz transzformáció számításában csak kis eltérés van a gyors Fourier transzformációhoz képest, így nem alkalmazunk külön INVFFT-t, hanem az FFT-t képeztük ki úgy, hogy egy paraméter értékétől függően vagy a transzformációt vagy pedig az inverzét hajtja végre.

A következő táblázat - IBM 370/145 gépen futtatott - programok különböző méretű /4.2./ típusu egyenletrendszerek megoldásához szükséges kerekített műveletidőit /CPU idő/ tartalmazza.

N /p=q=12/	Gauss elimináció		L-T-Z algoritmus		a bemutatott eljárás	
	perc	mp	perc	mp	perc	mp
32		4		1		1
128	2	59		9		2
256	23	11		32		3
512			2	01		6
1024			8	03		11

1. táblázat

1. Melléklet

Mint ahogy azt már az "előzményeket" áttekintő 4.1. részben említettük, a dolgozatban megadott eljárás kiindulásául A.Jain 1978-ban közölt [20] algoritmus szolgált.

Jain eredményei azonban a Toeplitz-típusu szalagmátrix együtthatóju lineáris egyenletrendszereknek csak egy részére alkalmazhatók, mivel Jain olyan feltételezéseket használ, amellyel ugyan egyszerű megoldási algoritmust tud kialakítani, azonban így az eljárás veszít az általánosságából.

Ha a dolgozatban használt jelöléseket alkalmazzuk, akkor a következőképpen tudjuk leírni a Jain által tett feltételezéseket:

Tekintsük a T_c inverzének és az y vektornak a szorzatát az inverz mátrix 9. ábra szerinti particionálása mellett:

$$\begin{array}{c}
 \text{/M.1./} \\
 B_{\underline{y}} = \begin{bmatrix} B^{ii} & B^{im} & B^{it} \\ \hline B^{mi} & B^{mm} & B^{mt} \\ \hline B^{ti} & B^{tm} & B^{tt} \end{bmatrix} \begin{matrix} p \\ \\ q \end{matrix} \cdot \begin{bmatrix} \underline{y}^i \\ \underline{y}^m \\ \underline{y}^t \end{bmatrix} \begin{matrix} q \\ p \\ p \end{matrix} = \begin{bmatrix} B^{ii}\underline{y}^i + B^{im}\underline{y}^m + B^{it}\underline{y}^t \\ \hline B^{mi}\underline{y}^i + B^{mm}\underline{y}^m + B^{mt}\underline{y}^t \\ \hline B^{ti}\underline{y}^i + B^{tm}\underline{y}^m + B^{tt}\underline{y}^t \end{bmatrix} \begin{matrix} p \\ \\ q \end{matrix}
 \end{array}$$

30. ábra

Jain, algoritmusában az /M.1./ szorzatvektor első p komponenseként a

$$\text{/M.2./} \quad B^{ii}\underline{y}^i + B^{it}\underline{y}^t$$

értéket használja, azaz elhanyagolja a $B^{im}\underline{y}^m$ értéket. A vektor utolsó q komponensének pedig a

$$\text{/M.3./} \quad B^{ti}\underline{y}^t + B^{tt}\underline{y}^t$$

értéket tekinti, elhagyva a $B^{tm}\underline{y}^m$ -et.

Igy viszont az algoritmus csak abban az esetben ad helyes eredményt, ha a

$$B^{im}_Y = \underline{0}$$

/M.4./

$$B^{tm}_Y = \underline{0}$$

egyenletek egyszerre teljesülnek. Azonban ez egy nagyon erős feltétel és legtöbbször nem teljesül.

/Egy elégséges feltétel /M.4./ teljesülésére, hogy $\underline{Y}^m = \underline{0}$ legyen, ami viszont szemléletesen azt jelenti, hogy a kiindulási egyenletrendszer jobb oldalán az első p és utolsó q értéket kivéve csak nulla állhat/

Azt pedig, hogy az /M.4./ általános esetben nem érvényes, szemléletesen bizonyítja a 4.6. részben bemutatott példa. Ugyanis a példában

$$B^{im}_Y = 1/2 \neq 0$$

$$B^{tm}_Y = -3/2 \neq 0 .$$

A /4.30./ egyenletrendszerre a Jain által adott algoritmust alkalmazva megoldásként

$$x_1 = 9.5 ; \quad x_2 = -4.5 ; \quad x_3 = -10.5 ; \quad x_4 = 23.5$$

adódik, ami a /4.30./ egyenletrendszer egyértelmű megoldhatósága miatt nem lehet jó megoldás.

2. Melléklet

A mellékletben a következő FORTRAN nyelvű szubrutinok találhatók:

1. TOPLEQ - a /4.2./ egyenletrendszert megoldó "fő" szubrutin .
2. FFT - gyors /inverz/ diszkrét Fourier transzformációt végző szubrutin.
3. LTZALG - a Levinson-Trench-Zohar algoritmus alapján készült, S. Zohar által publikált [38] és javított [39] szubrutin, amely Toeplitz mátrix együtthatóju lineáris egyenletrendszerek megoldására szolgál.
4. GAUSEL - lineáris egyenletrendszert Gauss eliminációval megoldó szubrutin.

***** TOPLEQ *****

PAGE 1.

SUBROUTINE TOPLEQ(LW,P,N,Q,Y,LP,LQ,NN,FQ,B,R,S,LT,T)

* * * * *

* THIS SUBROUTINE SOLVES THE * * * * *

* * * * *

* AX=Y * * * * *

* * * * *

* SYSTEM OF LINEAR EQUATIONS, WHEN THE COEFFICIENT * * * * *

* MATRIX /A/ IS A BANDED TOEPLITZ-TYPE MATRIX. * * * * *

* ----- * * * * *

* * * * *

* PARAMETERS: * * * * *

* N - IS THE NUMBER OF EQUATIONS. * * * * *

* LP - IS THE NUMBER OF NON-ZERO ELEMENTS IN THE * * * * *

* FIRST ROW OF THE COEFFICIENT MATRIX. * * * * *

* LQ - IS THE NUMBER OF NON-ZERO ELEMENTS IN THE * * * * *

* FIRST COLUMN OF THE COEFFICIENT MATRIX. * * * * *

* ARRAY P - CONTAINS THE NON-ZERO ELEMENTS OF THE * * * * *

* FIRST ROW OF THE COEFFICIENT MATRIX. * * * * *

* ARRAY Q - CONTAINS THE NON-ZERO ELEMENTS OF THE * * * * *

* FIRST COLUMN OF THE COEFFICIENT MATRIX. * * * * *

* ARRAY Y - INITIALLY CONTAINS THE DATA VECTOR. ON * * * * *

* RETURN, IT CONTAINS THE SOLUTION VECTOR. * * * * *

* * * * *

* ARRAYS FQ, B - ARE WORKING ARRAYS. * * * * *

* ARRAYS R,S,T - ARE WORKING ARRAYS. * * * * *

* LW = LP+LQ-2 * * * * *

* LT = LW*(LW+1) * * * * *

* NN = 2*N * * * * *

* * * * *

DIMENSION P(LW),Q(N),Y(N),FQ(NN),B(NN),R(LW),S(LW),T(LT)

KI=0

M=N

1 KI=KI+1

MM=2*(M/2)

IF(M.NE.MM) GO TO 40

M=M/2

IF(M.NE.1) GO TO 1

***** TOPLEQ *****

PAGE 2.

I. COUNTING TC /FROM T/

```
DO 2 I=2,LP
  Q(N+2-I)=P(I)
DO 3 I=1,N
  FQ(2*I-1)=Q(I)
  FQ(2*I)=0.
```

II. COUNTING B /THE INVERSE OF THE TC MATRIX/

```
CALL FFT(NN,FQ,KI,1)
ZE=10**(-10)
DO 4 I=1,N
  FN=FQ(2*I-1)**2+FQ(2*I)**2
  IF (EN.LE.ZE) GO TO 42
  FQ(2*I-1)=FQ(2*I-1)/EN
  FQ(2*I)=-FQ(2*I)/EN
```

// DFT/B/ ==> FQ //

```
DO 5 I=1,NN
  B(I)=FQ(I)
CALL FFT(NN,B,KI,-1)
DO 6 I=1,N
  B(I)=B(2*I-1)
```

III. COUNTING THE M MATRIX

III.A. CALCULATION THE INVERSE OF THE
G MATRIX /FROM P/

```
LPP=LP-1
DO 7 I=1,LPP
  S(I)=P(LP+1-I)
DO 8 I=2,LPP
  P(I)=0.
P(1)=1./S(1)
DO 9 M=2,LPP
  MM=M-1
  DO 10 K=1,MM
    P(M)=P(M)+S(K+1)*P(M-K)
  P(M)=-P(1)*P(M)
```

***** TOPLEQ *****

PAGE 3.

III.B. CALCULATION THE INVERSE OF THE F MATRIX /FROM Q/

```

C
C
C
C
C
C
LQQ=LQ--1
DO 11 I=1,LQQ
11   S(I)=Q(LQ+1-I)
DO 12 I=2,LQQ
12   Q(I)=0.
Q(1)=1./S(1)
DO 13 M=2,LQQ
    MM=M-1
    DO 14 K=1,MM
14       Q(M)=Q(M)+S(K+1)*Q(M-K)
13       Q(M)=-Q(1)*Q(M)

```

IV. COUNTING N MATRIX /BY B MATRIX/

```

C
C
C
C
IN=LFP-LQQ+1
IF (IN.GT.0) GO TO 15
IN=IN+N
15 RZERO=B(IN)
LRS=LW-1
DO 16 I=1,LRS
    IN=IN+1
    IF (IN.LE.N) GO TO 16
    IN=IN-N
16 R(I)=B(IN)
IN=LFP-LQQ+1
IF (IN.GT.0) GO TO 17
IN=IN+N
17 DO 18 I=1,LRS
    IN=IN-1
    IF (IN.GT.0) GO TO 18
    IN=IN+1
18 S(I)=B(IN)

```

V. CALCULATION THE Z VECTOR

```

C
C
C
C
DO 19 I=1,N
    B(2*I-1)=Y(I)
19 B(2*I)=0.
CALL FFT(NN,B,KI,1)
DO 20 I=1,N
    V=B(2*I-1)*FQ(2*I-1)-B(2*I)*FQ(2*I)
    B(2*I)=B(2*I)*FQ(2*I-1)+B(2*I-1)*FQ(2*I)
20 B(2*I-1)=V
CALL FFT(NN,B,KI,-1)
DO 21 I=1,N
21 B(I)=B(2*I-1)

```

***** TOPLEQ *****

PAGE 4.

VI. CALCULATION OF W VECTOR

VI.A. COUNTING THE R MATRIX /R=M-N/

RZERO=-RZERO

LQ1=LQQ-1

DO 22 I=1,LQ1

R(I)=-R(I)

DO 23 I=1,LFP

R(LQ1+I)=P(I)-R(LQ1+I)

LP1=LFP-1

DO 24 I=1,LP1

S(I)=-S(I)

DO 25 I=1,LQQ

S(LP1+I)=Q(I)-S(LP1+I)

VI.B. CALCULATION THE S VECTOR /FROM Z/

DO 26 I=1,LQQ

IN=N-LQQ

Q(I)=B(IN+I)

DO 27 I=1,LFP

Q(LQQ+I)=B(I)

VI.C. SOLUTION OF THE REDUCED SYSTEM OF /P+Q/
LINEAR TOEPLITZ EQUATIONS

IF (ABS(RZERO).LE.ZE) GO TO 28

T(1)=RZERO

IN=1+LRS

INN=1+2*LRS

DO 29 I=1,LRS

T(I+1)=S(I)

T(IN+I)=R(I)

T(INN+I)=Q(I)

S(I)=T(I)

R(I+1)=T(IN+I)

S(LRS+1)=T(LRS+1)

R(1)=T(1)

T(3*LRS+2)=Q(LW)

C ***** TOPLEQ ***** PAGE 5.

C
C
C
C
C
C
C

VI.C.A. SOLUTION THE TOEPLITZ SET OF /P+Q/
LINEAR EQUATIONS BY THE LEVINSON-
-TRENCH-ZOHAR ALGORITHM

CALL LTZALG(LW,RZERO,R,S,Q,P,Y)
IF(RZERO.NE.0) GO TO 30

C
C
C
C
C
C
C

VI.C.B. SOLUTION THE SYSTEM OF /P+Q/ LINEAR EQU-
ATIONS BY GAUSS ELIMINATION ALGORITHM
BECAUSE OF THE COEFFICIENT MATRIX
DOESN'T STRONGLY NONSINGULAR

DO 31 I=1,LRS
S(I)=T(I+1)
R(I)=T(IN+I)
31 Q(I)=T(INN+I)
Q(LW)=T(3*LRS+2)
28 CALL GAUSEL(T,LW,LW+1,Q,Q,R,S)

C
C
C
C
C
C
C

VII. CALCULATION THE SOLUTION VECTOR

VII.A. COUNTING THE H VECTOR /BY MULTIPLICATION
OF B MATRIX AND C VECTOR /H=B*C/ /

30 DO 32 I=1,N
32 Y(I)=0.
DO 33 I=1,LQQ
33 Y(I)=Q(LPP+I)
DO 34 I=1,LPP
IN=N-LPP+I
34 Y(IN)=Q(I)
DO 35 I=1,N
35 F(I)=B(I)
DO 36 I=1,N
B(2*I-1)=Y(I)
36 B(2*I)=0.
CALL FFT(NN,B,KI,1)
DO 37 I=1,N
V=B(2*I-1)*FQ(2*I-1)-B(2*I)*FQ(2*I)
B(2*I)=B(2*I)*FQ(2*I-1)+B(2*I-1)*FQ(2*I)
37 B(2*I-1)=V
CALL FFT(NN,B,KI,-1)

***** TOPLEQ *****

PAGE 6.

VII.B. CALCULATION THE X VECTOR /BY ADDITION OF
Z AND H VECTORS / $X=Z+B*C$ /

```
DO 38 I=1,N
      B(I)=B(2*I-1)
DO 39 I=1,N
      Y(I)=B(I)+P(I)
```

NORMAL END:

RETURN

END OF ROUTINE WITHOUT SOLUTION:

E1: THE N DOESN'T AN INTEGER-POWER OF 2

```
40 WRITE (6,41)
41 FORMAT(20X,'THE N DOES NOT AN INTEGER-POWER OF 2!')
```

RETURN

E2: THE CIRCULAR MATRIX IS SINGULAR SO,THE
INVERSE MATRIX DOESN'T EXISTS

```
42 WRITE (6,43)
      WRITE (6,44)
43 FORMAT(20X,'THE CIRCULAR MATRIX IS SINGULAR!')
44 FORMAT(20X,'THE INVERSE MATRIX DOES NOT EXISTS!')
```

RETURN

END

***** FFT *****

PAGE 1.

SUBROUTINE FFT(LA,A,M,KK)

*
* SUBROUTINE FOR THE COMPUTATION OF THE /INVERSE/
* DISCRETE FOURIER TRANSFORM OF AN A VECTOR
* BYE THE FAST FOURIER TRANSFORM ALGORITHM.
*

* PARAMETERS: *

* ARRAY A - INITIALLY CONTAINS THE DATA VECTOR.
* ON RETURN, IT CONTAINS THE TRANS-
* FORMED VECTOR.

* LA/2 - IS THE NUMBER OF THE COMPONENTS OF A.

* KK - IS A POINTER. IF KK=1, THAN THE RU-
* OUTINE COMPUTES THE DISCRETE FOURIER
* TRANSFORM OF A, AND IF KK=-1, THAN
* ROUTINE COMPUTES THE INVERSE FOURIER
* TRANSFORM OF THE A-VECTOR.

* M = LOG /LA/-1

* 2

DIMENSION A(LA)

XKK=-KK

N=2**M

XN=N

NV2=N/2

NM1=N-1

J=1

IF(KK.NE.-1) GO TO 2

DO 1 JK=1,N

A(2*JK-1)=A(2*JK-1)/XN

A(2*JK)=A(2*JK)/XN

DO 5 I=1,NM1

IF(I.GE.J) GO TO 3

TR=A(2*J-1)

TI=A(2*J)

A(2*J-1)=A(2*I-1)

A(2*J)=A(2*I)

A(2*I-1)=TR

A(2*I)=TI

K=N/2

IF(K.GE.J) GO TO 5

J=J-K

K=K/2

GO TO 4

J=J+K

C ***** FFI ***** PAGE 2.

C
C
C
C

```
PI=3.141592
DO 7 L=1,N
  LE=2*L
  LE1=LE/2
  XLE1=LE1
  UR=1.
  UI=0.
  WR=COS(PI/XLE1)
  WI=XKK*SIN(PI/XLE1)
  DO 6 J=1,LE1
    DO 4 I=J,N,LE
      IP=I+LE1
      TR=A(2*IP-1)*UR-A(2*IP)*UI
      TI=A(2*IP)*UR+A(2*IP-1)*UI
      A(2*IP-1)=A(2*I-1)-TR
      A(2*IP)=A(2*I)-TI
      A(2*I-1)=A(2*I-1)+TR
      A(2*I)=A(2*I)+TI
    6 X=UR
      UR=X*WR-UI*WI
      UI=UI*WR+X*WI
  7
RETURN
END
```

6

7

***** LTZALG *****

PAGE 1.

SUBROUTINE LTZALG(M,RZERO,R,A,SD,E,G)

*
* THIS SUBROUTINE SOLVES A TOEPLITZ SET
* OF M LINEAR EQUATIONS BY THE
* LEVINSON-TRENCH-ZOHAR ALGORITHM.
* -----

* PARAMETERS:

* RZERO - IS THE MAIN DIAGONAL ELEMENT.
* ARRAY R - CONTAINS THE ELEMENTS OF THE
* FIRST COLUMN OF THE TOEPLITZ
* MATRIX.
* ARRAY A - CONTAINS THE ELEMENTS OF THE
* FIRST ROW OF THE TOEPLITZ
* MATRIX.
* ARRAY SD - INITIALLY CONTAINS THE DATA
* VECTOR. ON RETURN, IT CONTA-
* INS THE SOLUTION VECTOR.
* ARRAYS E,G - ARE WORKING ARRAYS.
* M - IS THE NUMBER OF EQUATIONS.
* N = M-1.

DIMENSION R(M),SD(M),E(M),A(M),G(M)

ZE=10**(-10)

IF(ABS(RZERO).LE.ZE) GO TO 8

N=M-1

DO 1 I=1,N

A(I)=A(I+1)

R(I)=R(I+1)

T=1./RZERO

DO 2 I=1,N

R(I)=T*R(I)

A(I)=T*A(I)

SD(I)=T*SD(I)

SD(M)=T*SD(M)

E(1)=-A(1)

G(1)=-R(1)

FLAMDA=1.-A(1)*R(1)

IF(ABS(FLAMDA).LE.ZE) GO TO 8

C ***** LTZALG ***** PAGE 2.
C
C
C

```
DO 7 I=1,N
  IP=I+1
  TETLAM=SD(IP)
  ETALAM=-A(IP)
  GAMLAM=-R(IP)
  DO 3 J=1,I
    IMJ=IP-J
    TETLAM=TETLAM-R(J)*SD(IMJ)
    GAMLAM=GAMLAM-R(J)*G(IMJ)
3    ETALAM=ETALAM-A(J)*E(IMJ)
  TETLAM=TETLAM/FLAMDA
  DO 4 J=1,I
    IMJ=IP-J
4    SD(J)=SD(J)+TETLAM*E(IMJ)
  SD(IP)=TETLAM
  IF(I-N) 5,7,7
5    GAMLAM=GAMLAM/FLAMDA
  T=ETALAM
  ETALAM=ETALAM/FLAMDA
  FLAMDA=FLAMDA-T*GAMLAM
  IF(ABS(FLAMDA).LE.ZE) GO TO 8
  DO 6 J=1,I
    IMJ=IP-J
    T=E(J)
    E(J)=E(J)+ETALAM*G(IMJ)
    G(IMJ)=G(IMJ)+GAMLAM*T
6    CONTINUE
  E(IP)=ETALAM
  G(IP)=GAMLAM
7  CONTINUE
  RETURN
8  RZERO=0.
  RETURN
  END
```

***** GAUSEL *****

PAGE 1.

SUBROUTINE GAUSEL (A,N,M,Y,X,R,S)

```
*****
*
*      A SUBROUTINE FOR SOLUTION OF THE
*
*      AX=Y
*
*      SYSTEM OF  N  LINEAR EQUATIONS BY
*      GAUSS ELIMINATION  ALGORITHM.
*      -----
*
*  PARAMETERS:
*      N - IS THE NUMBER OF EQUATIONS.
*      M = N+1
*      ARRAY  A - IS THE COEFFICIENT MATRIX.
*      ARRAY  Y - IS THE DATA VECTOR.
*      ARRAY  X - IS THE SOLUTION VECTOR.
*      ARRAYS R,S - ARE WORKING ARRAYS.
*
*****
```

DIMENSION A(N,M),Y(N),X(N),R(N),S(N)

LRS=N-1

DO 1 I=1,N

A(I,M)=Y(I)

DO 2 I=1,LRS

A(I+1,I+1)=A(1,1)

DO 3 J=1,LRS

MJ=N-J

DO 3 I=1,MJ

A(I+J,I)=R(J)

A(I,I+J)=S(J)

DO 5 I=1,N

T=1./A(I,I)

IJ=I+1

DO 4 J=IJ,M

A(I,J)=T*A(I,J)

IF(I.EQ.N) GO TO 6

II=I+1

DO 5 K=II,N

DO 5 J=II,M

A(K,J)=A(K,J)-A(I,J)*A(K,I)

X(N)=A(N,M)

C
C
C
C

***** GAUSEL *****

PAGE 2.

```
DO 8 I=1,LRS
  IN=N-I
  II=IN+1
  W=0.
  DO 7 J=II,N
    W=W+A(IN,J)*X(J)
  X(IN)=A(IN,M)-W
RETURN
END
```

IRODALOMJEGYZÉK

- [1] Aho, V.-Hopcroft, E-
Ullmann, J. The Design and Analysis of
Computer Algorithms

/Addison Wesley 1975./
- [2] Allen, J. Computer architecture for
signal processing

Proc. of IEEE Apr. 1975.
- [3] Beck Gy. A gyors Fourier transzformáció
és egy alkalmazása

Diplomamunka JATE 1978.
- [4] Beck Gy. Gyors algoritmus Toeplitz-ti-
pusu szalagmátrix együtthetőju
lineáris egyenletrendszerek
megoldására

Alkalmazott Matematikai Lapok
1982./1-2.
- [5] Beck Gy. An efficient solution of banded
Toeplitz systems by application
of the fast Fourier transform

MTA SZTAKI Közlemények 28/1982.

- [6] Beck Gy. A fast algorithm for the solution of banded Toeplitz sets of linear equations

Zeitschrift für Angewandte Mathematik und Mechanik
1983. április /megjelenés alatt/
- [7] Beck Gy. Nagyméretű Toeplitz egyenletrendszerek hatékony megoldása a gyors Fourier transzformációs algoritmus felhasználásával

INTERKOMPUTO'82 Nemzetközi Számítástudományi Szimpozium kiadványa 1982. /megjelenés alatt/
- [8] Beck Gy. Hatékony algoritmus lineáris Hankel és Toeplitz egyenletrendszerek megoldására

SZÁMALK Alkotó Ifjúság pályázat
1982.
- [9] Bergland, G. Fast Fourier transform hardware implementations - an overview

IEEE Trans. on Audio and Electroacoustics Jun. 1969.

- [10] Bergland, G. A fast Fourier transform algorithm for a global, highly parallel processor
- IEEE Trans. on Audio and Electroacoustics Jun. 1969.
- [11] Box, G.-Jenkins, G. Time Series Analysis Forecasting and Control
- /Holden-Day San Francisco 1970./
- [12] Brigham, E. The Fast Fourier Transform
- /Englewood Cliffs, Prentice-Hall 1974./
- [13] Clergot, H.-Scharf, L. Connections between classical and statistical methods of FIR digital filter design
- IEEE Trans on Acoust., Speech, and Signal Proc. Oct. 1978.
- [14] Cooley, J.-Tukey, J. An algorithm for the machine calculation of complex Fourier series
- Math. of Comput. Apr. 1965.

- [15] Despain,A. Very fast Fourier transform
algorithms hardware for im-
plementation

IEEE Trans. on Computers
May 1979.
- [16] Farden,D.-Scharf,L. Statistical design of nonrecur-
sive digital filters

IEEE Trans. on Ac., Sp., and
Signal Pr. June. 1974.
- [17] Gray,R. Toeplitz and Circulant Matrices:
A Review

/Stanford University,Satnford,
CA, Tech. Rep., 1970./
- [18] Hunt,B. Digital image processing

Proc. of IEEE Apr. 1975.
- [19] Jain,A. An operator factorisation method
for restoration of blurred images

IEEE Trans.on Computers, Nov.1977.

- [20] Jain,A. Fast inversion of banded Toeplitz matrices by circular decompositions
IEEE Trans. on Ac., Sp., and Signal Proc. Apr. 1978.
- [21] Jain,J. An efficient algorithm for a large Toeplitz set of linear equations
IEEE Trans. on Ac.,Sp., and Signal Proc. Dec. 1979.
- [22] Jenkins,G.- Spectral Analysis and Its Applications
Watts,D.
/Holden-Day: San Francisco 1970./
- [23] Kailath,T.- Inverses of Toeplitz operators,
Vieria,A.-Morf,M. innovations, and orthogonal polynomials
Proc. of IEEE Conference on Decision and Control Dec. 1975.
- [24] Levinson,N. The Wiener RMS /root mean square/
error criterion in filter design
and prediction
J. Math. Phys. Jan. 1947.

- [25] Luk, W.-Phil, M.- Li, H. Microcomputer-based real-time/on-line f.f.t. processor
IEEE Proc. Jan. 1980.
- [26] Marple, L. A new autoregressive spectrum analysis algorithm
IEEE Ac., Sp., and Signal Proc. Aug. 1980.
- [27] Móricz F. Numerikus Analízis II.
/Egyetemi jegyzet Tankönyvkiadó 1975./
- [28] Nerlove, M.- Grether, D.- Carvalho, J. Analysis of economic time series
/Academic Press 1979./
- [29] Preis, D. The Toeplitz matrix: Its occurrence in antenna problems and a rapid inversion algorithm
IEEE Trans. on Antenna and Propagation March. 1972.
- [30] Rózsa P. Lineáris algebra és alkalmazásai
/Műszaki Könyvkiadó 1974./

- [31] Trench, W. An algorithm for the inversion of finite Toeplitz matrices
J. SIAM, Sep. 1964.
- [32] Trench, W. Weighting Coefficients for the prediction of stationary time series from the finite past
J. SIAM, Nov. 1967.
- [33] Trench, W. Inversion of Toeplitz band matrices
Math. of Computation Oct. 1974.
- [34] Wesley, M. Associative parallel processing for the fast Fourier transform
IEEE Trans. on Audio and Electroacoustics June. 1969.
- [35] Wood, L.-Treitel, S. Seismic signal processing
Proc. of IEEE Apr. 1975.
- [36] Zohar, S. Toeplitz matrix inversion: The algorithm of W.F.Trench
J. of ACM Oct. 1969.

- [37] Zohar, S. The solution of a Toeplitz set
of linear equations

J. of ACM Apr. 1974.
- [38] Zohar, S. Fortran subroutines for the so-
lution of Toeplitz sets of linear
equations

IEEE Trans on Ac., Sp., and
Signal Proc. Dec. 1979.
- [39] Zohar, S. Correction to "Fortran subrouti-
nes for the solution of Toeplitz
sets of linear equations"

IEEE Ac., Sp., and Signal Proc.
Oct. 1980.