9. Mitmekanalilise aparaadi pöördülesanne

Lõpmatumõõtmeline ülesanne

Kuna tegelik lahendamine on paratamatult lõplikumõõtmeline, siis vaatleme probleemi lihtsustatult, mõõtmisvigu hindamata.

Aparaadivõrrand

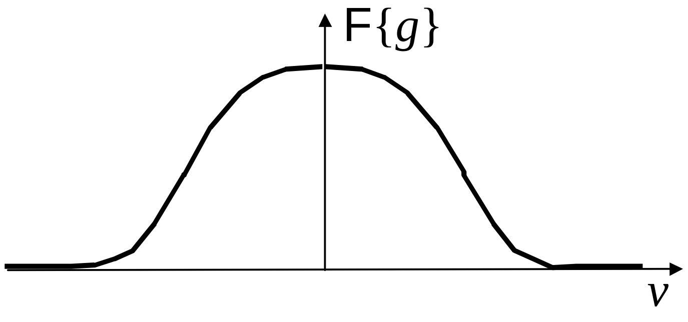
$$y(w) = \int_{v_{\min}}^{v_{\max}} g(w,v)x(v)\,\mathrm{d}v$$

on formaalselt Fredholmi esimest liiki võrrand ja selle lahendamine Hadamard'i mõttes matemaatiliselt ebakorrektselt püstitatud ülesanne. 60-ndatel aastatel loodi regulariseerimise teooria, mis mõõtmisandmete töötlemise ülesannete puhul ei ole rahuldav.

Probleemi olemus on hästi läbi nähtav sidumaparaadi puhul

$$y(w) = \int_{-\infty}^{\infty} g(w-v)x(v)\,\mathrm{d}v \qquad \mathsf{F}\{y\} = \mathsf{F}\{g\}\,\mathsf{F}\{x\}$$

Formaalne lahend on $\mathsf{F}\{x\} = \mathsf{F}\{y\} \,/\, \mathsf{F}\{g\}$. Tüüpolukord:



Analoogia helisalvestise restaureerimise ülesandega.

Olgugi, et lõpmatumõõtmelises teoorias ei ole teada korrektset mõõtmisvigade hindamise meetodit, on lõpmatumõõtmeline mudel matemaatikute hulgas paraku ikka veel üldtunnustatud ja matemaatikute autoriteedile tuginedes ka füüsikute hulgas levinud.

Matemaatiline sissejuhatus 1: ruutvormid

$$R(u) = \sum_{i,j} A_{ij} u_i u_j = \sum_i u_i \left(\sum_j A_{ij} u_j \right) = (\boldsymbol{u}, \boldsymbol{A}\boldsymbol{u}) = \boldsymbol{u}^{\mathrm{T}} \boldsymbol{A}\boldsymbol{u}$$

Sümmetriseerimine $\boldsymbol{A} := (\boldsymbol{A} + \boldsymbol{A}^{\mathrm{T}}) / 2$

Mittenegatiivne ja positiivne määratus ning $\boldsymbol{A}^{-1}$ olemasolu tingimus.

Matemaatiline sissejuhatus 2: mitmemõõtmeline normaaljaotus

Juhuslik vektor $\boldsymbol{v} = \{v_i\}$, $<\boldsymbol{v}> = \{<v_i>\}$, $\boldsymbol{u} = \boldsymbol{v} - <\boldsymbol{v}>$

Dispersiooni- ehk kovariatsioonimaatriks $\boldsymbol{C} = \{c_{ij}\} = \{<u_i u_j>\}$

Normaalne tihedusfunktsioon

$$f(\boldsymbol{u}) = \mathrm{const} \times \exp\left(-\frac{1}{2} R(\boldsymbol{u}) \right)$$

$$R(\boldsymbol{u}) = \boldsymbol{u}^{\mathrm{T}} \boldsymbol{A}\boldsymbol{u} = (\boldsymbol{v} - <\boldsymbol{v}>)^{\mathrm{T}} \boldsymbol{A} (\boldsymbol{v} - <\boldsymbol{v}>) , \qquad \mathrm{const} = \frac{\sqrt{\det \boldsymbol{A}}}{(2\pi)^{n/2}}$$

$$f(v) = \frac{\sqrt{\det \boldsymbol{A}}}{(2\pi)^{n/2}} \exp\left(-\frac{1}{2} (\boldsymbol{v} - <\boldsymbol{v}>)^{\mathrm{T}} \boldsymbol{A} (\boldsymbol{v} - <\boldsymbol{v}>) \right)$$

$$\boldsymbol{C} = \boldsymbol{A}^{-1}$$

Kontrolliks: ühemõõtmeline erijuhtum.

Vektormõõtmise statistiline mudel

$$\boldsymbol{y} = \boldsymbol{G}\boldsymbol{x} + \xi \quad \text{ehk} \quad y_j = \sum_{i=1}^{n} G_{ji} x_i + \xi_j$$

Hüpotees: ξ tõenäosusjaotus on mitmemõõtmeline normaaljaotus.

$$\xi = \boldsymbol{G}\boldsymbol{x} - \boldsymbol{y}$$

Eeldus: $\boldsymbol{G}$, $\boldsymbol{y}$ ja $\boldsymbol{D} = \{<\xi_i \xi_j>\}$ on tuntud suurused

Ülesanne: leida tundmatu parameetri $\boldsymbol{x}$ jaoks parim hinnang $\hat{\boldsymbol{x}}$.

Gauss-Markovi algoritm

$\boldsymbol{x}$ on tundmatu, kuid kindel suurus, $\hat{\boldsymbol{x}}$ on esialgu vabalt varieeritav.

Gaussi vähimruutude meetodi lihtsustatud variandi puhul on ülesandeks valida $\hat{\boldsymbol{x}}$ niiviisi, et hälvete ruutude summa $\sum(\boldsymbol{Gx} - \boldsymbol{y})^2$ oleks vähim võimalik. Täiustatud variandi korral korrutatakse summeeritavad kaaludega, mis on pöördvõrdelised mõõtmisvea ruuduga. Selle ülesande lahendas Gauss ise. Veel enam täiustatud ülesanne võtab arvesse ka mõõtmisvea koordinaatide vahelisi korrelatsioone. Selle ülesande lahendas Markov.

Veel üldisem ja otsesemalt põhjendatud on suurima tõepära ülesanne. Teatavasti taandub see normaaljaotuse korral Gauss-Markovi vähimruutude ülesandeks.

Tõepärafunktsiooni võrrand on vormilt sama, kui tõenäosusjaotuse tihedusfunktsioon, siin aga on varieeritavaks parameetri hinnangu testväärtus $\hat{\boldsymbol{x}}$:

$$L(\hat{\boldsymbol{x}}) = \mathrm{cn} \times \exp\left(-\frac{1}{2}\xi^{\boldsymbol{T}} \boldsymbol{D}^{-1} \xi\right) = \mathrm{cn} \times \exp\left(-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{G}\hat{\boldsymbol{x}})^{\boldsymbol{T}} \boldsymbol{D}^{-1} (\boldsymbol{y} - \boldsymbol{G}\hat{\boldsymbol{x}})\right)$$

Tõepära maksimiseerimiseks tuleb minimiseerida eksponendi argumendi moodul

$$R = (\boldsymbol{y} - \boldsymbol{G}\hat{\boldsymbol{x}})^{\boldsymbol{T}} \boldsymbol{D}^{-1} (\boldsymbol{y} - \boldsymbol{G}\hat{\boldsymbol{x}}).$$

Ekstreemumülesande lahendamise eeskiri: lahendada võrrandisüsteem

$$\frac{\partial R}{\partial \hat{x}_k} = 0, \qquad k = 1..n.$$

Lahenduskäik:

$$R = (\boldsymbol{y} - \boldsymbol{G}\hat{\boldsymbol{x}})^{\mathrm{T}} \boldsymbol{D}^{-1} (\boldsymbol{y} - \boldsymbol{G}\hat{\boldsymbol{x}})$$

$$R = \boldsymbol{y}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{y} - \boldsymbol{y}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{G}\hat{\boldsymbol{x}} - (\boldsymbol{G}\hat{\boldsymbol{x}})^{\mathrm{T}} \boldsymbol{D}^{-1} y + (\boldsymbol{G}\hat{\boldsymbol{x}})^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{G}\hat{\boldsymbol{x}}$$

$$R = \mathrm{const} + \hat{\boldsymbol{x}}^{\mathrm{T}} (\boldsymbol{G}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{G}) \hat{\boldsymbol{x}} - 2 (\boldsymbol{y}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{G}) \hat{\boldsymbol{x}}$$

Tähistame: $\boldsymbol{A} = \boldsymbol{G}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{G}$ ja $\boldsymbol{b} = \boldsymbol{G}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{y}$

Saame minimiseeritava: $S = R - \mathrm{const} = \hat{\boldsymbol{x}}^{\mathrm{T}} A\, \hat{\boldsymbol{x}} - 2 \boldsymbol{b}^{\mathrm{T}} \hat{\boldsymbol{x}}$

Arvutame:

$$\frac{\partial S}{\partial \hat{x}_k} = \frac{\partial}{\partial \hat{x}_k} \left(\sum_{i \neq k} A_{ik} \hat{x}_k \hat{x}_i + \sum_{j \neq k} A_{kj} \hat{x}_j \hat{x}_k + A_{kk} \hat{x}_k^2 - 2 b_k \hat{x}_k \right) = 2 \sum_j A_{kj} \hat{x}_j - 2 b_k = 0$$

Minimiseerimisülesanne sai kuju $\boldsymbol{A}\hat{\boldsymbol{x}} - \boldsymbol{b} = 0$

Markovi lahend: $\hat{\boldsymbol{x}} = \boldsymbol{A}^{-1} \boldsymbol{b} = (\boldsymbol{G}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{G})^{-1} \boldsymbol{G}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{y}$

Algoritm:

$$\begin{aligned} \boldsymbol{H} &:= \boldsymbol{G}^{\mathrm{T}} \boldsymbol{D}^{-1} \\ \boldsymbol{C} &:= (\boldsymbol{HG})^{-1} \\ \boldsymbol{x} &:= \boldsymbol{CHy} \end{aligned}$$

Mõõtmisvigade teisenemine.

Juhuslikku viga sisaldava $\boldsymbol{x}$ tõenäosustiheduse ja selle mittejuhusliku testväärtuse tõepärafunktsiooni avaldis on ühine. Normaalselt jaotatud $f(\boldsymbol{x})$ avaldist võib kirjutada kahel võrdväärsel viisil

$$cn_1 \times \exp\left(-\frac{1}{2} (\boldsymbol{x} - \langle \boldsymbol{x} \rangle)^{\mathrm{T}} \boldsymbol{C}^{-1} (\boldsymbol{x} - \langle \boldsymbol{x} \rangle) \right) = cn_2 \times \exp\left(-\frac{1}{2} \boldsymbol{x}^{\mathrm{T}} \boldsymbol{A}\, \boldsymbol{x} + \boldsymbol{b}^{\mathrm{T}} \boldsymbol{x} \right)$$

Ruutliikmete kordajad vasakus ja paremas astmenäitajas peavad olema võrdsed. See tähendab, et $\boldsymbol{C} = \boldsymbol{A}^{-1}$.

Järeldus: hinnangu $\hat{\boldsymbol{x}}$ mõõtmisvigu kirjeldab $\boldsymbol{C} = (\boldsymbol{G}^{\mathrm{T}} \boldsymbol{D}^{-1} \boldsymbol{G})^{-1}$, mis on sama $\boldsymbol{C}$, kui eespool toodud algoritmi vahetulemus.

Ülesanne: Kirjutada ülaltoodud algoritm välja ühemõõtmelise erijuhu jaoks!

Pöördülesanne tsirkulatsioonaparaadi puhul

Fourier teisendusel on lõplikumõõtmeline analoog. n-mõõtmeline vektor $\boldsymbol{x}$ ja tema Fourier kujutis $\tilde{\boldsymbol{x}}$ on seotud järgnevalt:

$$\tilde{\boldsymbol{x}} = \boldsymbol{F}\boldsymbol{x}$$

$$\boldsymbol{x} = \frac{1}{n}\boldsymbol{F}^{*}\tilde{\boldsymbol{x}},$$

teisendusmaatriksi $\boldsymbol{F}$ elemendid on:

$$F_{kl} = \exp\left(-\frac{2\pi i}{n}kl\right).$$

Praktilisel arvutamisel kasutatakse FFT algoritmi.

Olgu x ja y vektorid, G tsirkulatsioonmaatriks ja g selle esimene veerg, ning

$$\boldsymbol{y} = \boldsymbol{G}\boldsymbol{x}$$

Siis on rangelt algebraliselt

$$\tilde{y}_i = \tilde{g}_i \tilde{x}_i.$$

Kui $\boldsymbol{g}$ Fourier kujutise elementide hulgas pole nulle, taandub võrrandi lahendamine sageduste ruumis n jagamistehtele.

Selleks, et pöördülesande lahendamisel saaks lõplikumõõtmelist Fourier teisendust rangelt kasutada, peavad G ja D olema mõlemad tsirkulatsioon-maatriksid. Ainult sel juhul võib rääkida tsirkulatsioonaparaadist.

Küsimus: millised füüsikalis-tehnilised kitsendused tulenevad sellest nõudest?

Ülesanne: kirjutage välja varasema üldise algoritmi analoog tsirkulatsioonaparaadi jaoks.

10. Gauss-Markovi algoritmi programmeerimine

Kovariatsioonide arvutamine

Mõõdetavad suurused: $x \quad y$

Tähistus: $\Sigma z = \sum_{i=1}^{n} z_i$

Parameetrid:	*Hinnangud*:
Keskväärtused	$\bar{x} \quad \bar{y}$
Dispersioonide hinnangud	$D_x \quad D_y$
Kovariatsiooni hinnang	C_{xy}

Mõõtmise ajal kogutakse 6 summat väärtustest:

1, x, y, xx, yy, xy

Pärast mõõtmisi arvutatakse:

$n = \Sigma 1$

$\bar{x} = \Sigma x / n$

$\bar{y} = \Sigma y / n$

NB: Kui keskmised on apriori nullid, siis neid summasid pole tarvis!

$$D_x = \Sigma xx / n - \bar{x}^2$$

$$D_y = \Sigma yy / n - \bar{y}^2$$

$$C_{xy} = \Sigma xy / n - \bar{x}\bar{y}$$

Kovariatsioonimaatriks võib ajas muutuda. Pikaajalise mõõtmise korral kasutatakse tavaliselt üht kahest strateegiast:

1. Maatriksit uuendatakse mingi kindla arvu üksikmõõtmiste järel.
2. Maatriksit uuendatakse iga üksikmõõtmise järel “unustades” vananenud informatsiooni samm-sammult ja sujuvalt.

Viimasel juhul sobib eksponentkaalude algoritm:

$$keskmine := p \times keskmine + q \times x\,, \qquad p + q = 1$$

```
Program Myra;

Const n = 8;

Type vector = array [1..n] of double;
     matrix = array [1..n] of vector;

Var protokoll : text;
            D : matrix; {myra dispersioonimaatriks}

Procedure wD (silt : string);
   Var i, j : integer;
   Begin
   writeln (protokoll, silt);
   for i := 1 to n do begin
      for j := 1 to n do write (protokoll, 1000 * D [i, j] : 6 : 0);
      writeln (protokoll);
   end;
   writeln (protokoll); writeln (protokoll);
   End;

Procedure mootmine (reg : integer; var kirje : vector);
   {reg = 0 seab aparaadi nullikontrolli rezhiimi,
    reg = 1 seab mootmisrezhiimi}
   Var i : integer;
   Begin
      {MYRA}
      for i := 1 to n do kirje [i] := random - random;
      {korreleeritud myraga kanalid}
      kirje [3] := kirje [3] + kirje [2];
      kirje [4] := kirje [4] - kirje [2];
      {SIGNAAL}
      if reg = 1 then
      for i := 1 to n do kirje [i] := kirje [i] + i * (n - i);
   End;



VAR              y : vector;
                 s : matrix;
 i, j, kord, kordi : integer;
              p, q : double;
            valmis : boolean;

BEGIN
randomize;
valmis := true; {tegelikus programmis kontrollib seda aparatuur}
assign (protokoll, 'D_test.txt'); rewrite (protokoll);
```

```
{MEETOD 1}
kordi := 1000; {D uuendamise tsykli pikkus}
repeat
   for i := 1 to n do for j := 1 to n do s [i, j] := 0;
   for kord := 1 to kordi do begin
      mootmine (0, y);
      for i := 1 to n do for j := 1 to n do
         s [i, j] := s [i, j] + y[i] * y [j] ;
      mootmine (1, y);
      {...
       programmiloik, mis kasutab signaali sisaldavat mootmistulemust
       ja voib kasutada dispersioonimaatriksit
       v. a. esimeses "kordi" tsyklis
       ...}
   end;
   {dispersioonimaatriksi uuendamine}
   for i := 1 to n do for j := 1 to n do D [i, j] := s [i, j] / kordi;
   wD ('1000-kordne dispersioonimaatriks: esimene meetod');
until valmis;


{MEETOD 2}
kordi := 1000; {katse pikkus}
p := 0.99; q := 1 - p;
for i := 1 to n do for j := 1 to n do d [i, j] := 0;
repeat
   mootmine (0, y);
   {dispersioonimaatriksi uuendamine}
   for i := 1 to n do for j := 1 to n do
      d [i, j] := p * d [i, j] + q * y[i] * y [j] ;
   mootmine (1, y);
   {...
    programmiloik, mis kasutab signaali sisaldavat mootmistulemust
    ja voib kasutada dispersioonimaatriksit
    v. a. tsyklites mille number on vaiksem kui 3 / q
    ...}
   kordi := kordi - 1;
until kordi = 0;
wD ('1000-kordne dispersioonimaatriks:   teine meetod');

close (protokoll);
write ('OK'); readln;
END.
```

Kysimused:
1. kuidas saaks arvutuste mahtu kaks korda vahendada?
2. Kuidas lyhendada teise meetodi siirdeaega?

TULEMUSED:

1000-kordne dispersioonimaatriks: esimene meetod

161	-6	-7	9	-2	3	-9	-1
-6	163	167	-171	1	-0	6	-8
-7	167	335	-177	-2	8	7	-18
9	-171	-177	341	2	0	-13	9
-2	1	-2	2	165	-4	-7	-1
3	-0	8	0	-4	154	-6	11
-9	6	7	-13	-7	-6	168	5
-1	-8	-18	9	-1	11	5	160

1000-kordne dispersioonimaatriks: teine meetod

165	2	15	-2	2	12	9	-9
2	181	174	-186	-6	9	-2	16
15	174	334	-187	-12	26	18	12
-2	-186	-187	358	-0	-4	3	-21
2	-6	-12	-0	159	1	-21	-5
12	9	26	-4	1	160	3	2
9	-2	18	3	-21	3	187	-13
-9	16	12	-21	-5	2	-13	178

Gauss-Markovi algoritmi demonstreeriv programm

x - spekter,
y - spektromeetri kirje,
G - aparaadimaatriks, *y* = *Gx*,
D - aparaadimüra dispersiooni- ehk kovariatsioonimaatriks,
C - mõõtmistulemuse dispersiooni- ehk kovariatsioonimaatriks,
H - tehniline abimaatriks,
Q - ilmutava teisenduse *x* » *Qy* maatriks.

$$H := G^T D^{-1};\ C := (HG)^{-1};\ Q := CH;\ x := Qy$$

```
Program Gauss_Markov; {demonstratsiooniprogramm Markov.pas}

Const nmax = 25;
      nspekter = 8;
      nkirje = 16;
      veategur = 0.01;

Type vector = array [1..nmax] of double;
     matrix = array [1..nmax] of vector;

Var protokoll : text;

Procedure wvector (silt : string; v : vector; n : integer);
   Var i : integer;
   Begin
   writeln (protokoll, silt);
   for i := 1 to n do
      if n < 10 then write (protokoll, v [i] : 7 : 2)
                else write (protokoll, v [i] : 5 : 2);
   writeln (protokoll);
   writeln (protokoll);
   End;

Procedure wsigma (silt : string; m : matrix; n : integer);
   Var i : integer;
       v : vector;
   Begin
   for i := 1 to n do v [i] := sqrt (m [i, i]);
   wvector (silt, v, n);
   End;

Procedure wmatrix (silt : string; m : matrix; r, v : integer);
   Var i, j : integer;
   Begin
   writeln (protokoll, silt);
   for i := 1 to r do begin
      for j := 1 to v do write (protokoll, m [i, j] : 7 : 2);
      writeln (protokoll);
   end;
   writeln (protokoll);
   End;
```

```
Function InvDet (var a : matrix; n : integer) : double;
   {InvDet := det (a). Matrix a will be inverted using
   the Gauss-Jordan method CACM58b. If failure, then InvDet = 0}
   Var     i, j, k : integer;
      d, y, w, eps : double;
                 z : array [1..nmax] of integer;
              b, c : vector;
   Begin
   eps := 0;
   for i := 1 to n do for j := 1 to n do eps := eps + abs (a [i, j]);
   eps := 1E-6 * eps / (n * n);
   d := 1;
   for j := 1 to n do z [j] := j;
   for i := 1 to n do begin
      k := i; y := a [i, i];
      for j := i + 1 to n do begin w := a [i, j];
         if abs (w) > abs (y) then begin k := j; y := w end;
      end;
      if abs (y) < eps then begin InvDet := 0; exit end;
      d := y * d; if k <> i then d := - d;
      y := 1 / y;
      for j := 1 to n do begin
         c [j] := a [j, k]; a [j, k] := a [j, i];
         a [j, i] := -c [j] * y;
         b [j] := a [i, j] * y; a [i, j] := b [j];
      end;
      j := z [i]; z [i] := z [k]; z [k] := j;
      a [i, i] := y;
      for k := 1 to n do if k <> i then
      for j := 1 to n do if j <> i then
      a [k, j] := a [k, j] - b [j] * c [k];
   end;
   for j := 1 to n do begin
      k := z [j];
      while k <> j do begin
         for i := 1 to n do begin
            w := a [j, i]; a [j, i] := a [k, i]; a [k, i] := w;
         end;
         i := z [k]; z [k] := z [j]; k := i; z [j] := i;
      end;
   end;
   InvDet := d;
   End;

Function Gauss (sigma : double) : double;
   Var x : double;
       i : integer;
   Begin
   x := 0;
   for i := 1 to 6 do x := x + random - random;
   gauss := sigma * x;
   End;
```

```
{=============================================================}

VAR x0, y, x : vector;
    d, d1, g, h, c, q : matrix;
    i, j, k : integer;
    sum, det : double;

BEGIN
randomize;

{Kahekyyrulise "toelise" spektri moodustamine}
for j := 1 to nspekter do
x0 [j] := 1 / ( 0.2 + sqr (j - 3)) + 2 / (1 + sqr (j - 6));

{Aparaadimaatriksi G moodustamine}
for i := 1 to nkirje do
for j := 1 to nspekter do
g [i, j] := exp (-sqr (j - i/2) / 3);

{Mootmisvea dispersioonimaatriksi D moodustamine}
for i := 1 to nkirje do
for k := 1 to nkirje do
if i = k then d [i, k] := veategur * (1 + i/8)
         else d [i, k] := 0;

{Spektromeetri kirje y arvutamine}
for i := 1 to nkirje do begin
   sum := gauss (sqrt (d [i, i]));
   for j := 1 to nspekter do sum := sum + g [i, j] * x0 [j];
   y [i] := sum;
end;

{Diagonaalse dispersioonimaatriksi D pooramine}
for i := 1 to nkirje do
for k := 1 to nkirje do
if i = k then d1 [i, k] := 1 / d [i, i]
         else d1 [i, k] := 0;
{Kysimus: kuidas arvutada kui D ei ole diagonaalne?}

{Vahemaatriksi H = GT D-1 arvutamine, yldine algoritm}
for j := 1 to nspekter do
for i := 1 to nkirje do begin
   sum := 0;
   for k := 1 to nkirje do sum := sum + g [k, j] * d1 [k, i];
   {NB: g indeksite jarjekord tahendab transponeerimist}
   h [j, i] := sum;
end;
{Kysimus: kuidas arvutada kui D on ette teada diagonaalne?}
```

```
{Spektri kovariatsioonimaatriksi C = (H G)-1 arvutamine}
for i := 1 to nspekter do
for j := 1 to nspekter do begin
   sum := 0;
   for k := 1 to nkirje do sum := sum + h [i, k] * g [k, j];
   c [i, j] := sum;
end;
det := invdet (c, nspekter);


{Ilmutava maatriksi Q = C H arvutamine}
for j := 1 to nspekter do
for i := 1 to nkirje do begin
   sum := 0;
   for k := 1 to nkirje do sum := sum + c [j, k] * h [k, i];
   q [j, i] := sum;
end;

{Spektri hinnangu x = Q y arvutamine}
for j := 1 to nspekter do begin
   sum := 0;
   for i := 1 to nkirje do sum := sum + q [j, i] * y [i];
   x [j] := sum;
end;
{Kui ilmutav maatriks Q pole ise vajalik, siis saab
 arvutada otstarbekamalt. kuidas?}

{Tulemuste protokollimine}
assign (protokoll, 'markov1.txt'); rewrite (protokoll);
wvector ('Kirje', y, nkirje);
wsigma ('Kirje viga', d, nkirje);
wvector ('Moodetud spekter', x, nspekter);
wsigma ('Spektri veahinnang', c, nspekter);
wvector ('Toeline spekter', x0, nspekter);
wmatrix ('Aparaadimaatriks', g, nkirje, nspekter);
wmatrix ('Vea kovariatsioonimaatriks', c, nspekter, nspekter);
close (protokoll);
writeln ('OK'); readln;
END.
```

```
Kirje
 1.22 2.53 3.97 5.41 6.50 7.36 7.36 6.81 5.95 5.26 4.79 4.72 3.95 3.45 2.73 1.94

Kirje viga
 0.11 0.11 0.12 0.12 0.13 0.13 0.14 0.14 0.15 0.15 0.15 0.16 0.16 0.17 0.17 0.17

Moodetud spekter
  -1.11   4.70  -0.61   8.36  -5.93   7.91  -2.53   1.98

Spektri veahinnang
   0.58   1.54   2.55   3.33   3.68   3.45   2.58   1.22

Toeline spekter
   0.32   0.95   5.20   1.23   1.24   2.11   1.06   0.44


Aparaadimaatriks
   0.92   0.47   0.12   0.02   0.00   0.00   0.00   0.00
   1.00   0.72   0.26   0.05   0.00   0.00   0.00   0.00
   0.92   0.92   0.47   0.12   0.02   0.00   0.00   0.00
   0.72   1.00   0.72   0.26   0.05   0.00   0.00   0.00
   0.47   0.92   0.92   0.47   0.12   0.02   0.00   0.00
   0.26   0.72   1.00   0.72   0.26   0.05   0.00   0.00
   0.12   0.47   0.92   0.92   0.47   0.12   0.02   0.00
   0.05   0.26   0.72   1.00   0.72   0.26   0.05   0.00
   0.02   0.12   0.47   0.92   0.92   0.47   0.12   0.02
   0.00   0.05   0.26   0.72   1.00   0.72   0.26   0.05
   0.00   0.02   0.12   0.47   0.92   0.92   0.47   0.12
   0.00   0.00   0.05   0.26   0.72   1.00   0.72   0.26
   0.00   0.00   0.02   0.12   0.47   0.92   0.92   0.47
   0.00   0.00   0.00   0.05   0.26   0.72   1.00   0.72
   0.00   0.00   0.00   0.02   0.12   0.47   0.92   0.92
   0.00   0.00   0.00   0.00   0.05   0.26   0.72   1.00

Vea kovariatsioonimaatriks
   0.34  -0.87   1.36  -1.62   1.59  -1.32   0.86  -0.35
  -0.87   2.38  -3.85   4.73  -4.77   4.00  -2.64   1.10
   1.36  -3.85   6.51  -8.32   8.65  -7.42   4.98  -2.09
  -1.62   4.73  -8.32  11.09 -12.00  10.62  -7.30   3.11
   1.59  -4.77   8.65 -12.00  13.54 -12.44   8.81  -3.84
  -1.32   4.00  -7.42  10.62 -12.44  11.92  -8.75   3.92
   0.86  -2.64   4.98  -7.30   8.81  -8.75   6.68  -3.09
  -0.35   1.10  -2.09   3.11  -3.84   3.92  -3.09   1.49
```

Lisa: operatsioonid kolmnurkmaatriksitega, vaata MATRIX.PAS