

**Rīgas Tehniskā universitāte**

Datorvadības, automātikas un datortehnikas institūts

Datoru tīklu profesora grupa

## **Signālu apstrāde**

**Prof. V.Zagurskis**

**2000.g.**

## Saturs

<b>Wavelets and Signal Processing .....</b>	<b>1-1</b>
Diskrētā laika gadījums .....	1-1
Multiresolution pyramid metode .....	1-2
Subband Coding shēmas .....	1-3
Diskrētā wavelet transformācija .....	1-4
Iterated Filters and Regularity .....	1-4
Mērogošanas funkcijas un wavelets, kuras iegūtas no iterāciju filtriem .....	1-5
Wavelet funkciju pielietojums signālu apstrādē .....	1-9
Piezīmes .....	1-9
<b>Algoritmi ātrai vektorkvantēšanai .....</b>	<b>2-1</b>
Ievads .....	2-1
Tuvākā kaimiņa meklēšana, lietojot K-D kokus: Standartpieeja .....	2-1
Tuvākā kaimiņa meklēšana, lietojot K-D kokus: prioritātes pieeja .....	2-2
Kaimiņattiecību grafi .....	2-2
Empīriskā analīze .....	2-3
<b>Multimediju ievad/izvad termināliekārtu projektēšana .....</b>	<b>3-1</b>
Ievads .....	3-1
Termināla portatīvums .....	3-1
Enerģijas pieejamība .....	3-1
Skaitļošanas sadalīšana .....	3-2
Optimizējošie algoritmi .....	3-2
Optimizējošās arhitektūras .....	3-2
Shēmu veids un tehnoloģija .....	3-3
Bezvadu sakari .....	3-3
Spektrālā efektivitāte .....	3-3
Kļūdu iespējamība .....	3-3
Kopsavilkums .....	3-4
<b>Reālā laika sistēmu ātrā prototipēšana: jauns izaicinājums? .....</b>	<b>3-5</b>
Ievads .....	3-5
Objekta specifikācija/apraksts .....	3-5
Simulācija/Emulācija .....	3-6
Projekta sadalīšana .....	3-7
Īstenošana .....	3-7
Kopsavilkums .....	3-8
<b>Signālu apstrāde ar augstākas kārtas spektru .....</b>	<b>4-1</b>
Uz korelāciju balstīta signālu apstrāde: priekšrocības un ierobežojumi .....	4-3
Augstākas kārtas statistiskie aprēķini un spektrs .....	4-6
Definīcijas un īpašības .....	4-6
Augstākās kārtas spektra izskaitļošana no datiem .....	4-11
Tradicionālo novērtētāju īpašības un asimptotiskā daba .....	4-13
Lineārs process .....	4-15
Koncentrēta spektrālā blīvuma jauda un Polispektrs .....	4-15
Polispektrs .....	4-17
Neminimuma fāžu sistēmu noteikšana .....	4-18
Nelineāri procesi .....	4-19
Pielietošana .....	4-22
Massīva apstrāde .....	4-22
Klasifikācija .....	4-23
Harmoniska atgūšana .....	4-24

Laika aiztures novērtējums .....	4-24
Akla dekonvolūcija un stabilizācija .....	4-25
Interferences anulēšana .....	4-27
Nobeigums .....	4-29
<b>Spontānas runas atpazīšana ar galvena vārda palīdzību adaptīvajā atbildē .....</b>	<b>5-1</b>
Rezumē .....	5-1
Ievads .....	5-1
Dialoga runas sistēma ar multimodala atbildi .....	5-1
TOSBURGII- iespējas paplašināšana .....	5-2
Rezultāta novērtējums .....	5-5
Noslēgums .....	5-6
<b>Rate-Constrained Picture-Adaptive Quantization for JPEG Baseline Coders .....</b>	<b>6-1</b>
Problemas formulēšana .....	6-1
Kvantešanas tabulu radianas algoritma apraksts .....	6-1
Eksperimenta rezultāti .....	6-3
Noslēgums .....	6-3
<b>Interpolācijas JPEG kodēšanas shēmas .....</b>	<b>6-4</b>
Abstrācija .....	6-4
Ievads .....	6-4
JPEG principi .....	6-4
Inerpolācijas JPEG shemas .....	6-5
Datoru simulācija .....	6-6
Nobeigums .....	6-8
<b>Psihovizuālo kroplājumu novērtēšana monohroma attēlam un video kompresijai .....</b>	<b>7-1</b>
Abstrācija .....	7-1
Ievads .....	7-1
Monohromas bildes kroplājumu novērtēšana .....	7-1
Kroplājumu mērīšana plusmam .....	7-2
Kustīga kadra segmentācijas algoritms .....	7-3
Kroplājumu mērīšana kadra pikselīm .....	7-3
Kroplājumu mērīšana kustības pikselīm .....	7-4
Kopīgais video kroplājumu lielums .....	7-5
Subjektīvais tests .....	7-5
Rezultāti .....	7-5
Nobeigums .....	7-6
<b>Augstas veiktspējas vektoru kvantēšanas mikroskāma .....</b>	<b>8-1</b>
Ievads .....	8-1
Arhitektūra .....	8-1
Iekšējā Reizinājumu Masīva (IRM) Shēma .....	8-2
Komparators .....	8-2
Inicializācijas ķēde .....	8-2
Datu Formāta Pārveidotājs .....	8-3
VK Mikroskāma .....	8-4
Veiktspēja .....	8-4
Secinājumi .....	8-4
<b>Būvējot Jūsu sistēmu .....</b>	<b>9-1</b>
Testēšanai un mērīšanai vai rūpnieciskajai automatizācijai .....	9-1
Pielietojumi .....	9-1

Jūsu sistēmas sastāvdaļas .....	9-2
Programmatūra.....	9-2
Iespraudzamās datu iegūšanas plates (DAQ).....	9-2
Signālu saskaņošana .....	9-2
GPIB .....	9-2
VXI .....	9-2
Virtnes un rūpnieciskie sakari.....	9-2
Jūsu instrumentu programmatūras komponentes un analīzes programmatūra .....	9-3
Operētājsistēma.....	9-3
Draiveru programmatūra.....	9-3
Pielikumprogrammu programmatūra.....	9-3
Instrumentu draiveri.....	9-3
Datu analīze ar LabVIEW, LabWindows un Signal Processing komplektu.....	9-3
Skaitliskā analīze un modelēšana ar HiQ .....	9-3
<b>Programmatūras pārskats.....</b>	<b>9-4</b>
National Instruments programmatūras risinājumi .....	9-4
Virtuālo instrumentu programmatūra .....	9-4
Datu analīzes un vizualizācijas programmatūra .....	9-4
Mūsu programmatūras stratēģija .....	9-5
Programmatūras, kas jums ir vislabāk piemērota, izvēlēšanās.....	9-6
Testēšanas un mērīšanas programmatūras arhitektūra .....	9-6
Instrumentu draiveri .....	9-6
Rūpnieciskās automatizācijas programmatūras arhitektūra .....	9-7
Datu analīze un vizualizācija.....	9-7
Automatizācijas rūpnieciskā ievadizvade.....	9-7
<b>DAQ pārskats.....</b>	<b>9-8</b>
Ievads.....	9-8
DAQ Designer .....	9-8
Produkti, kas domāti datu iegūšanai .....	9-8
Plug and Play .....	9-8

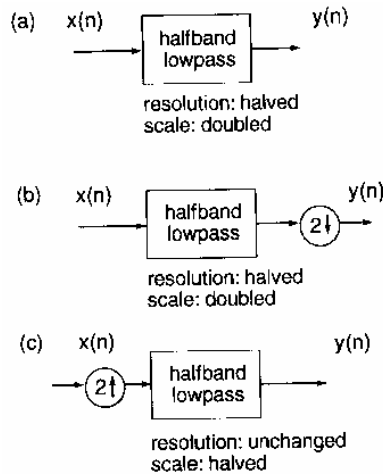
## Wavelets and Signal Processing

### Diskrētā laika gadījums

Šajā nodaļā visi procesi sākotnēji tiek aplūkoti tikai vienīgi no diskrēta laika viedokļa. Tad, izmantojot filtru iterāciju banku konstrukciju, var nonākt atpakaļ pie nepārtraukta laika gadījuma un parādīt, kā konstruēt orthonormālas bāzes wavelets nepārtraukta laika signāliem.

Diskrētā laika gadījumam ir izveidotas divas metodes: subband coding un pyramidal coding vai augstas izšķirtspējas signāla analīze. Metodes paredzētas kodēšanai, kur ir svarīgs minimālais nepieciešamais diskretizācijas pakāpju skaits. Pyramid kodēšana faktiski izmanto arī oversampling, bet tā ir vienkāršāk uztverama, tādēļ tā tiek aplūkota pirmā.

Vēl būtiska ir jēdzienu mērogs un izšķirtspēja izpratne no wavelets viedokļa. Mērogs ir saistīts ar signāla izmēru, bet izšķirtspēja ir piesaistīta detaļu daudzumam signālā.

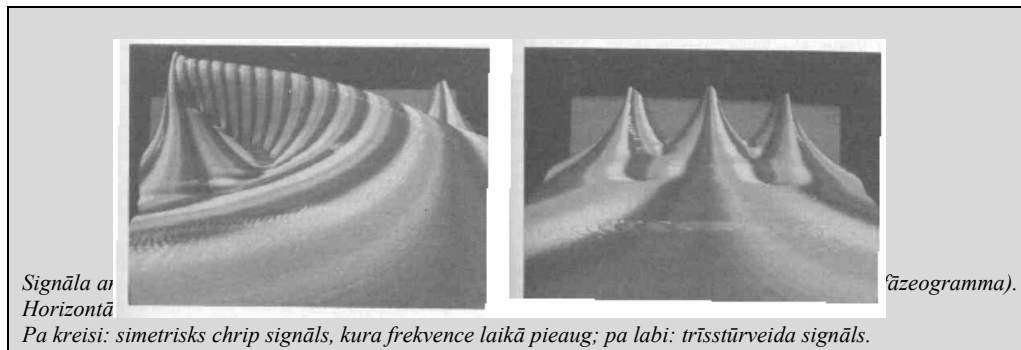


**9. attēls. Izšķirtspējas un mēroga izmaiņas diskrētā laikā (ar divkāršiem reizinātājiem). (a) Pusjoslas zemfrekvenču filtrēšana samazina izšķirtspēju divas reizes (mērogs ir nemainīgs). (b) Pusjoslas zemfrekvenču filtrēšana, kas seko divkāršai subsampling divkāršo mērogu (un samazina uz pusi izšķirtspēju, salīdzinot ar (a)). (c) Upsampling 2 reizes, kas seko pēc pusjoslas zemfrekvenču filtrēšanas samazina uz pusi mērogu (izšķirtspēja ir nemainīga).**

Mēroga parametra jēdziens diskrētā wavelets analīzē: lieliem mērogiem globālā ziņā izplestas wavelets aplūko subsampled signālu, bet maziem mērogiem noslēgtas wavelets analīzē signālā smalkas detaļas.

### Sarežģītas wavelet transformācijas 3-D attēlojumi

Šeit ir parādīti divi piemēri: simetrisks chirp un trīsstūra funkcija blakus. Abos gadījumos pārtraukuma punkti ir skaidri identificējami mazos mērogos (figūras augšējā daļa). Chirp satur divus šādus punktus (sākumu un beigas), bet trīsstūrim ir trīs šādi punkti.



### Multiresolution pyramid metode

Ieejas signāls ir uzdots kā punktu masīvs  $x(n)$ ,  $n \in \mathbb{Z}$ . Var iegūt zemākas izšķirtspējas signālu, izmantojot zemfrekvences filtrēšanu ar pusjoslas zemfrekvenču filtru, kuram ir impulsu reakcija  $g(n)$  (svaru funkcija kā masīvs). Pēc Naikvista likuma var samazinātās nolases divkārtot, tā divkārtot analīzes mērogu. Tas iespaido signāla  $y(n)$  rezultātu, kas uzdots kā

Mērogs mainās sakarā ar divkārtu subsampling, jo nobīde pa diviem oriģināla signālā  $x(n)$  izsauc nobīdi pa vienu

$$y(n) = \sum_{k=-\infty}^{+\infty} g(k) x(2n - k)$$

$y(n)$ .

Balstoties uz  $x(n)$  zemfrekvenču filtra un samazināto nolašu versiju, nepieciešams atrast oriģināla  $x(n)$  aproksimāciju  $a(n)$ . Tas tiek veikts pirmām kārtām upsampling  $y(n)$  divas reizes (t.i., starp katru nolasi ievietojot vienu nulli), jo salīdzināšanai ir nepieciešams signāls oriģināla mērogā.

$$y'(2n) = y(n), \quad y'(2n+1) = 0$$

Tad  $y'(n)$  tiek interpolēts ar filtru, kura atbildes reakcija  $g'(n)$  (kompensētā svaru funkcija), lai iegūtu aproksimāciju  $a(n)$ ,

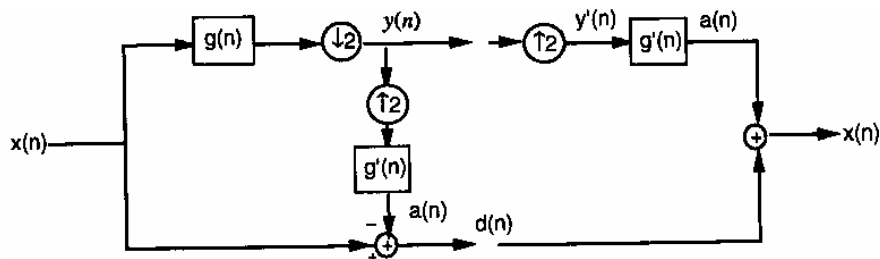
$$a(n) = \sum_{k=-\infty}^{\infty} g'(k) y'(n - k)$$

Nemot vērā, ka, ja  $g(n)$  un  $g'(n)$  ir ideāli pusjoslas filtri (frekvenču caurlaides spēja normalizētajā frekvenču diapazonā  $-\pi/2, \pi/2$  ir 1, bet vienāda ar 0 citur), tad Furjē transformācijai  $a(n)$  jābūt vienāda ar Furjē transformāciju  $x(n)$  frekvenču diapazonā  $-\pi/2, \pi/2$  un vienāda ar 0 citur. Tātad  $a(n)$  jābūt ideāli  $x(n)$  pusjoslas zemfrekvences aproksimācijai.

Protams,  $a(n)$  nebūs vienāds ar  $x(n)$ . Tādēļ jāatrod starpība starp  $a(n)$  un  $x(n)$ ,

$$d(n) = x(n) - a(n)$$

Tas ir acīm redzami, ka  $x(n)$  var atjaunot, saskaitot  $d(n)$  ar  $a(n)$ ; viss process attēlots 10. attēlā.



**10. attēls Piramīdas shēma. Zemfrekvenču subsampled aproksimācijas  $y(n)$  iegūšana, no kuras aproksimācija  $a(n)$  no  $x(n)$  tiek iegūta ar upsampling un interpolācijas palīdzību. Tad tiek aprēķināta starpība  $d(n)$  starp aproksimāciju un oriģinālu  $x(n)$ . Signāls  $x(n)$  pinībā tiek iegūts, atpakaļ pieskaitot  $a(n)$ .**

Ideāla pusjoslas zemfrekvenču filtra gadījumā ir skaidrs, ka  $d(n)$  satur frekvences virs  $\pi/2$  no  $x(n)$ , un tā  $d(n)$  tikpat labi var samazināt nolases divas reizes, nezaudējot informāciju.

Origiņālā signāla  $x(n)$  sadalīšana rupjā aproksimācijā  $a(n)$  plus papildus detaļas  $d(n)$  ir konceptuāli svarīgi. Tā kā tiek pielietota izšķirtspējas maiņa (zemfrekvences filtrēšanai seko divkārtšā subsampling, kas rada signālu ar uz pusi mazāku izšķirtspēju un divkārtšu mērogu, salīdzinot ar oriģinālu), iepriekšminētā metode ir daļa no tā, ko sauc par Multiresolution Signal Analysis datoru redzē.

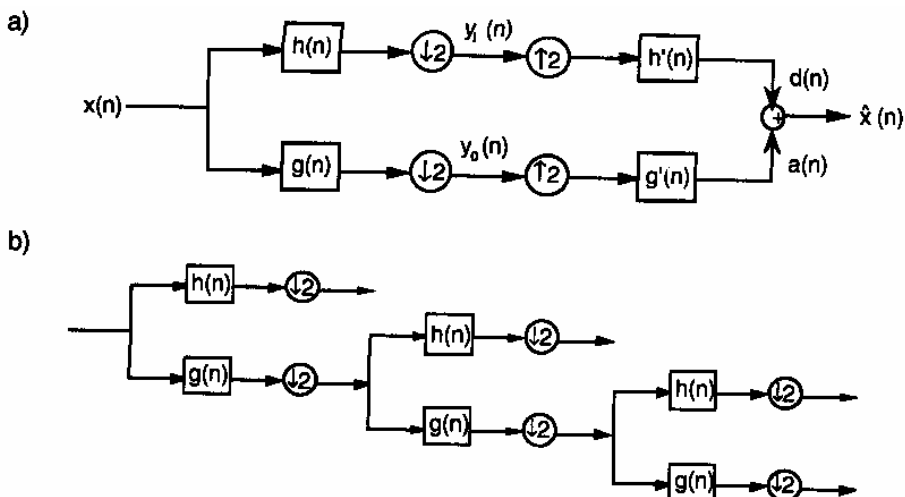
Shēma var tikt izmantota  $y(n)$  iterācijām, veidojot zemākas izšķirtspējas signālu zemāku mērogu hierarhiju. Tādēļ, ka ir šāda hierarhija, un signāli kļūst aizvien īsāki un īsāki, šādas shēmas sauc par signālu vai attēlu piramīdām.

### Subband Coding shēmas

Piramīdas shēmā ir liela redundance, tādēļ turpmāk apskatīsim shēmu, kur tādas nav. Zemfrekvenču subsampled aproksimācija ir iegūta tieši kā iepriekš aprakstīts, bet signāla starpības vietā tiek aprēķinātas "pievienotās detaļas", kā  $x(n)$  augstfrekvences nofiltrētā daļa (izmantojot filtru ar impulsu reakciju  $h(n)$ ), kura izriet no divkārtšā subsampling. Intuitīvi ir skaidrs, ka "pievienotās detaļas" zemfrekvences aproksimācijai ir augstfrekvences signāls, un ir acīm redzams, ka, ja  $g(n)$  ir ideāls pusjoslas zemfrekvenču filtrs, tad ideāls pusjoslas zemfrekvences filtrs  $h(n)$  novedīs pie izcilas oriģināla signāla attēlošanas divās samazinātās nolases versijās.

Tas ir precīzi viens wavelet dekompozīcijas solis, izmantojot  $\sin(x)/x$  filtrus, jo oriģinālais signāls tiek attēlots zemfrekvenču aproksimācijā (divkārtšā mērogā) un pievienotais detaļu signāls (arī divkārtšā mērogā). Šajā gadījumā, izmantojot ideālos filtrus, diskrētā versija ir identiska nepār-trauktajai wavelet transformācijai.

Vēl būtiskāk ir tas, ka nav nepieciešams izmantot ideālus filtrus, bet tomēr  $x(n)$  var tikt atjaunots no tā divām filtrētajām un samazinātās nolases versijām, kuras tagad tiek pārsauktas par  $y_0(n)$  un  $y_1(n)$ . Lai to izdarītu, abas tiek upsamplē un filtrētas pēc  $g'(n)$  un  $h'(n)$  atbilstoši un rezultātā saskaitītas kopā, kā parādīts attēlā 11a. Pretēji piramīdas gadījumam atjaunotais signāls (kuru tad nodēvējam par  $x(n)$ ) nav identisks ar  $\bar{x}(n)$ , lai gan filtriem ir daži specifiski ierobežojumi. Filtriem ar šādiem ierobežojumiem piemīt perfekta atjaunošanas īpatnība.



**11. attēls. Subband Coding shēma. (a) divas subsampled aproksimācijas: viena atbilstoši zemām un otra augstām frekvencēm. (rupjas diskretizācijas un detalizēta signāla shēmas). Atjaunotais signāls tiek iegūts, atkārtoti interpolējot aproksimācijas un summējot tās. Filtri, kuri novietoti pa kreisi, veido analīzes filtrēšanas bloku, bet pa labi - sintēzes filtrēšanas bloku. (b) Bloku diagramma (filtrēšanas mezglu koks) diskrētai wavelet transformācijai papildināta ar diskrēta laika filtriem un divkārtšu subsampling**

Vienkāršākais analīzes gadījums ir tad, kad analīzes un sintēzes filtri ir identiski un sasniedzama pilnīga atjaunošana (t.i.,  $\bar{x}(n) = x(n)$ ). Subband analīze/sintēze atbilst orthonormālas bāzes dekompozīcijai, bet signāla rekonstrukcija savukārt iegūstama, summējot ortogonālās projekcijas. FIR filtru ieviešana. Gan uz zemfrekvenču, gan augstfrekvenču filtriem var attiecināt (12)

$h(L - 1 - n) = (-1)^n g(n)$  kur  $L$  ir filtra garums (pāra skaitlis). Jāņem vērā, ka modulācija  $(-1)^n$  pārveido zemfrekvenču filtru par augstfrekvenču.

Filtrēšanas modulis attēlā 11a, kurš aprēķina convolutions, kuras izriet no subsampling by two, iegūst secību  $x(n)$  un secības  $\{g(-n+2k), h(-n+2l)\}$  iekšējos produktus (laika pretējo virzienu iegūst no convolution, kura apgriež vienu no secībām). Tā

$$y_0(k) = \sum_n x(n) g(-n + 2k)$$

$$y_1(k) = \sum_n x(n) h(-n + 2k)$$

Iepriekšminētās izteiksmes uzdod kritēriju, kurš nosaka vai punktu skaits bāzes funkcijai ir pietiekošs vai nepietiekošs funkcijas atjaunošanai.

$$x(n) = \sum_{k=-\infty}^{\infty} [y_0(k) g(-n + 2k) + y_1(k) h(-n + 2k)] \quad (13)$$

**Jāņem vērā, ka viss iepriekšminētais attiecas uz lineāru apstrādi. Ja nelineāra apstrāde tiek pievienota, oversampled daba piramīdas shēmai faktiski noved pie lielākas robustness.**

## Diskrētā wavelet transformācija

Lai sasniegtu smalkāku frekvenču izšķirtspēju pie zemākām frekvencēm (kā tas iegūts ar nepārtrauktu wavelet transformāciju), tiek iterēta shēma tikai zemākajā joslā. Ja  $g(n)$  ir labs pusjoslas zemfrekvenču filtrs,  $h(n)$  ir labs pusjoslas augstfrekvenču filtrs pēc (12). Tad viena iterācija pirmās apakšjoslas shēmai rada jaunu apakšjoslu, kura atbilst frekvenču spektra zemākam ceturksnim. Katra turpmākā iterācija daļa uz pusēm apakšjoslas platumu (palielina tās frekvenču izšķirtspēju divas reizes), bet sakarā ar divkārtu subsampling, tās laika izšķirtspēja arī tiek dalīta uz pusēm. Shematiski tas ir ekvivalenti 11b attēlam.

Būtiska diskrētā algoritma priekšrocība ir tā relatīvi zemā sarežģītība. Sarežģītība ir lineāra atkarībā no ieeju samples skaita ar konstantu faktoru, kurš atkarīgs no filtra garuma.

$$C_{total} = C_0 + \frac{C_0}{2} + \frac{C_0}{4} + \dots < 2C_0$$

$C_0$  operāciju skaits uz ievades katru ievades nolasi ( $C_0$  kārtā parasti ir vienāda ar  $L$ ). Tas parāda diskrētās wavelet transformācijas algoritma efektivitāti.

## Iterated Filters and Regularity

Ir būtiska starpība starp diskrēto shēmu un nepārtraukta laika wavelet transformāciju. Diskrētā laika gadījumā galvenā loma wavelet transformācijā ir augstfrekvenču filtram  $h(n)$  un subsampled zemfrekvenču filtru kaskādei, kuri seko augstfrekvenču filtram. Šie filtri rupji ņemot atbilst oktāvas joslas filtriem. pretēji nepārtrauktai wavelet transformācijai nav precīzi savstarpēji mērogojamas versijas. Šajā gadījumā, tā kā tiek izmantots diskrētais laiks, mērogu nav viegli definēt, tādēļ nepieciešama gan interpolācija, gan laika izvēršana.

Vajadzētu atrast ekvivalentu filtru, kurš atbilstu apakšējam zaram attēlā 11b, tas ir iterētais zemfrekvenču filtrs. Ērti izmantot filtru  $z$ -transformāciju, t.i.,

$G(z) = \sum g(n) z^{-n}$  sekojošā veidā. Var viegli pārbaudīt, ka subsampling by two, kura seko pēc filtrēšanas ar  $G(z)$  ir ekvivalenta filtrēšanai ar  $G(z^2)$ , kura seko subsampling ( $z^2$  iesprauž nulles starp impulsa reakcijas samples, kuras nonem tai sekojošā subsampling). Tas ir – pirmās divas zemfrekvenču filtrēšanas pakāpes var tikt aizvietotas ar  $z$ -transformāciju  $G(z)$   $G(z^2)$  sekojošas by subsampling by 4. Vispārīgā gadījumā, nosaucot  $G^1(z)$  par ekvivalentu filtru zemfrekvenču filtrēšanas i līmeņiem un subsampling by two (t.i., kopējā subsampling by  $2^l$ ), iegūstam

$$G^l(z) = \prod_{i=0}^{l-1} G(z^{2^i}) \quad (14)$$

Ja  $i$  bezgalīgi pieaug, filtrs kļūst bezgalīgi garš. Tiek ņemta vērā funkcija  $f(x)$ , kura ir gabaliem konstanta intervālos no garuma  $1/2^l$  un ir vērtība  $2^{l/2} g^l(n)$  intervālā  $[n/2^l, (n+1)/2^l]$ . Tas ir,  $f(x)$  ir kāpņveida funkcija ar vērtību uzdotu ar  $g_l(n)$  un intervāliem, kuri samazinās pēc  $2^{-l}$ . Var pārbaudīt, ka funkcija tiek nodrošināta intervālā  $[0, L-1]$ , kur  $L$  ir filtra garums. Ja  $i$  tuvojas bezgalībai,  $f(x)$  var konverģēt uz nepārtrauktu funkciju  $g_c(x)$ , vai funkciju ar bezgalīgi daudz pārtraukumiem, pat daļveida funkciju, vai nekonverģē. (skatīt 5. tabulu).



**5. tabula:**

**Regulāri mērogošanas filtri**

Funkcijas attēlošana tiek realizēta divās pakāpēs:

- 1) funkcijas attēlošana ar wavelet funkcijām;
- 2) papildus funkciju pievienošana lielākas precizitātes iegūšanai.

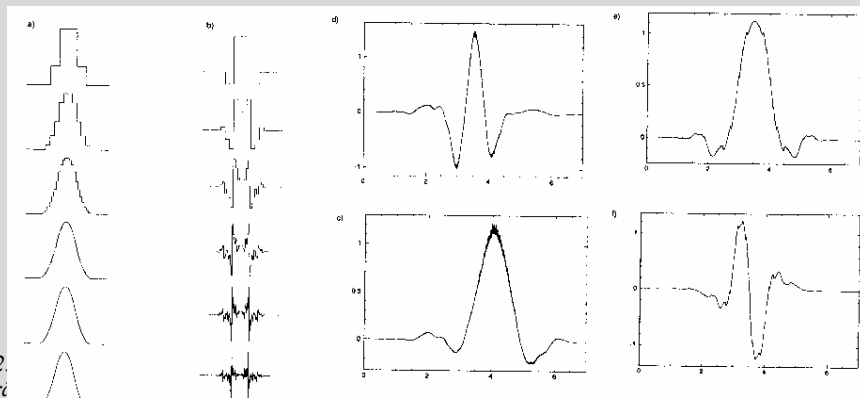


Fig. 12. (a) Wavelet funkcija. (b) Iterāciju rezultāti. (c) Iegūtais 8-tap zemfrekvenču filtrs ar 8 iterāciju kārtām. (d) Atbilstošā nepārtrauktā laika wavelet. (e) Johnston 8-tap zemfrekvenču filtrs pēc 8 iterācijām. (f) Atbilstošā nepārtrauktā laika wavelet.

**Mērogošanas funkcijas un wavelets, kuras iegūtas no iterāciju filtriem**

Šeit nepieciešams atsaukties uz  $g_c(x)$  kā rezultējošo funkciju uz kuru  $f(x)$  konverģē. Tā kā tas ir zemfrekvenču filtru produkts, rezultējošā funkcija pati par sevi ir zemfrekvenču un tiek saukta par "mērogošanas funkciju", jo tā tiek izmantota, lai pārietu no smalkāka mēroga uz rupjāku mērogu. No (14) izteiksmes izriet, ka  $g_c(x)$  apmierina sekojošu divu mērogu vienādojumu:

$$g_c(x) = \sum_{n=-\infty}^{\infty} g(n) g_c(2x-n) \quad (15)$$

13. attēls parāda divus šādus piemērus. Otrā pamatā ir 4-tap Daubechies filtrs, kurš ir regulārs un orthogonāls. Tomēr no 11b attēla ir skaidrs, ka bandpass filtrs tiek iegūts tādā pašā veidā, izņemot rezultējošo augstfrekvenču filtru. Savukārt ar līdzīgu struktūru (15) izteiksmei wavelet  $h_c(x)$  tiek iegūta

tā arī  $h_c(x) = \sum_{n=-\infty}^{\infty} h(n) g_c(2x-n)$  apmierina divu mērogu vienādojumu. (16)

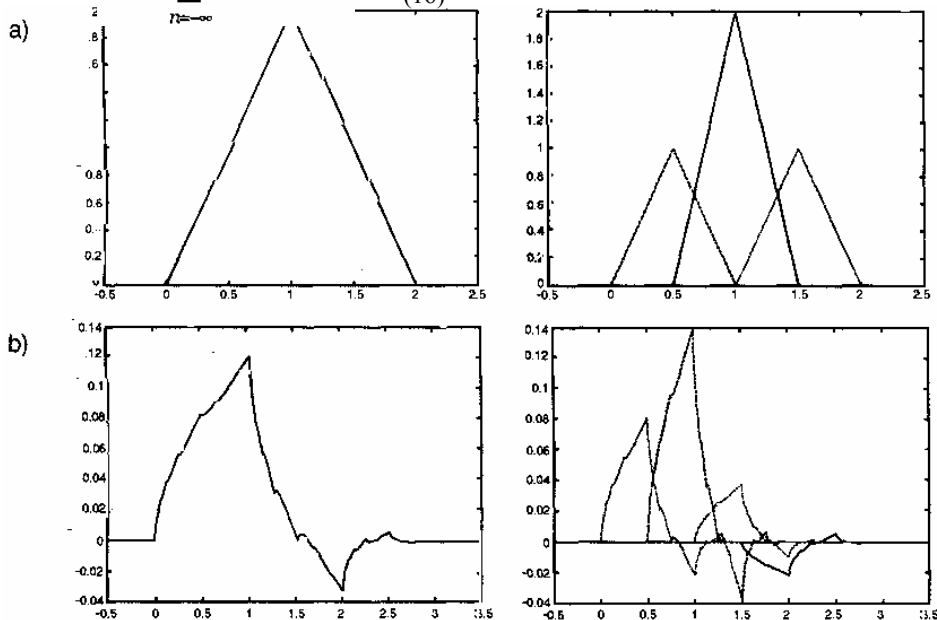


Fig. 13. Mērogošanas funkcijas, kuras apmierina divu mērogu starpību vienādojumus. (a) hat funkcija. (b)  $D_4$  wavelet, kura iegūta no regulāra 4-tap Daubechies filtra.

Ja filtri  $h(n)$  un  $g(n)$  veido orthonormālu kopu ar pāra nobīžu ievērošanu, tad funkcijas  $g_c(x-1)$  un  $h_c(x-k)$  veido orthonormālu kopu. Tā kā tās apmierina divu mērogu starpību vienādojumu, tad kopa  $h_c\{2^{-i}x-k\}$ ,  $i, k \in \mathbf{Z}$ , veido orthonormālu bāzi kvadrāta integrējamām funkcijām.

**6. tabula:**

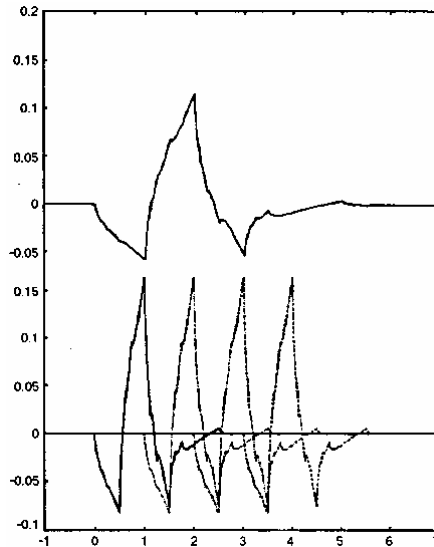
**Multiresolution analīze**

Šī multiresolution aproksimācijas koncepcija sniedz wavelet dekompozīcijas pamatideju. Signālu telpa tiek sadalīta ar plaknēm:

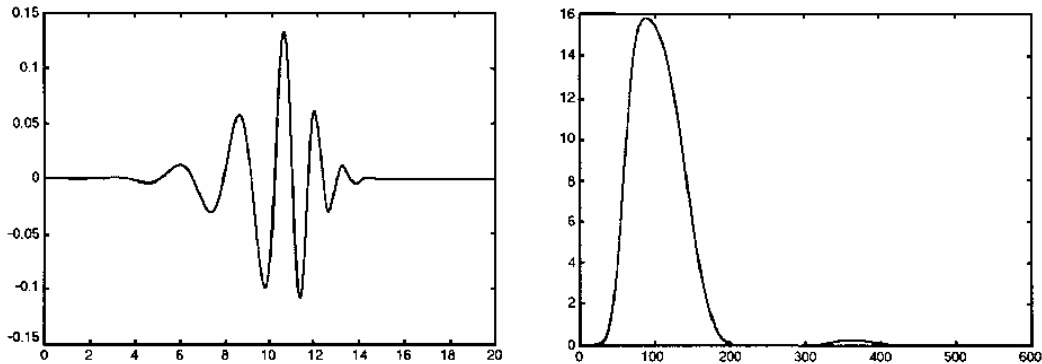
$$\dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \dots$$

No  $n$ -dimensiju telpas tiek izvēlēta 3-dimensiju telpa, un tālā visi procesi var tikt saistīti ar to.

14. attēls parāda 4-tap Daubechies wavelet divos mēros un ar divām nobīdēm.

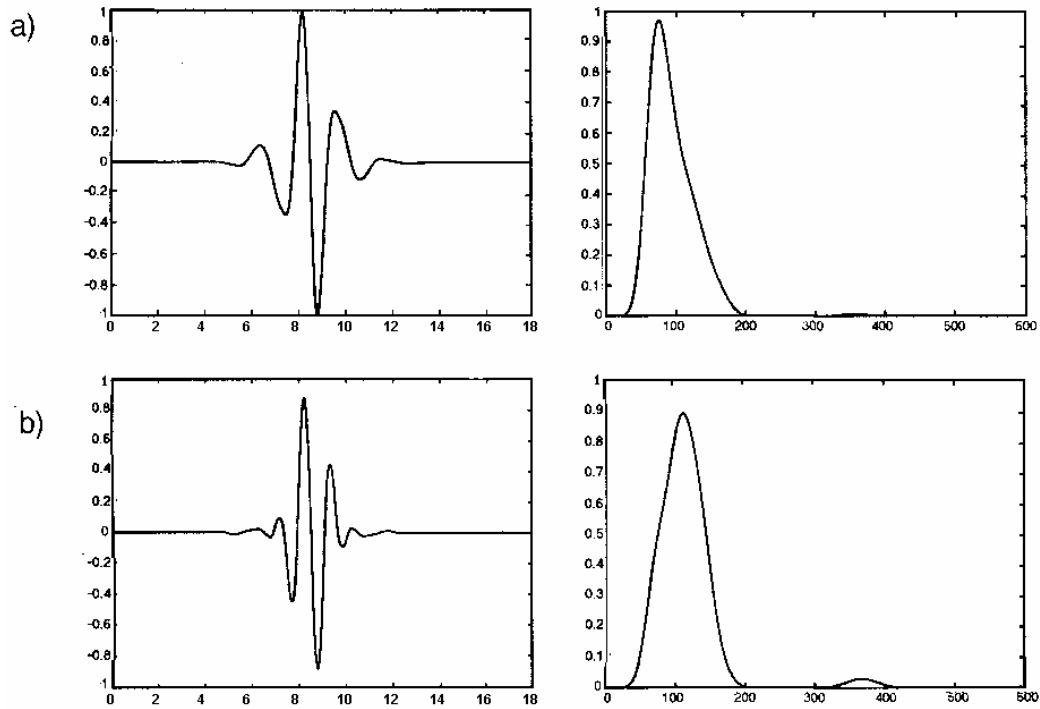


14. attēls.  $D_4$  wavelet un nobīžu divi mērogi.



15. attēls Orthonormāla wavelet, kura ģenerēta, izmantojot 18 līmeņu regulāru filtru . Funkcija attēlota pa kreisi, bet tās spektrs – pa labi.

16. attēls parāda biorthogonālu kopu ar lineārām fāzes wavelets, kur analīzes wavelets ir orthogonālas pret sintēzes wavelets. Šīs ir iegūtas no biorthogonāla lineāra fāzes filtra moduļa ar garumu 18 regulāri filtri.



**16. attēls. Biorthogonālas wavelets, kuras ģenerētas, izmantojot 18-tap regulārus filtrus. (a) Analīzes wavelet, (b) Sintēzes wavelet. Funkcija attēlota pa kreisi, bet tās spektrs – pa labi.**

Iepriekšējos piemēros tika parādīts, kā regulāri filtri var tikt izmantoti, lai ģenerētu wavelet bāzes. Pretējais apgalvojums arī ir spēkā. Tas nozīmē, ka orthogonālu mērogošanas filtru kopas var tikt izmantotas, lai ģenerētu izcilus atjaunošanas filtru moduļus.

Wavelet vairāku dimensiju paplašinājuma koncepcija, kura noderīga attēlu kodēšanā parādīta 7. tabulā.

**7. tabula:**

**Daudzdimensiju filtrācija**

Lai pielietotu wavelet dekompozīcijas daudzdimensionāliem signāliem (t.i., attēliem) nepieciešami multidimensionāli wavelets paplašinājumi. Acīm redzams risinājums ir "atdalāmas wavelets", kuras iegūtas no viendimensiju wavelets un mērogošanas funkciju produktiem. Vienkāršības labad aplūkosim divu dimensiju gadījumu. Izmantosim mērogošanas funkciju  $g_c(x)$  (15) un wavelet  $h_c(x)$  (16).

Var konstruēt sekojošas divu dimensiju funkciju: (katrs modulis ar savu funkciju):

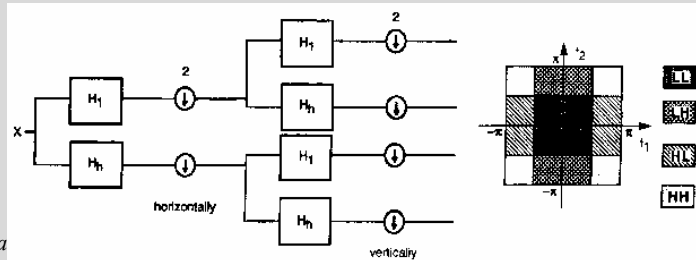
$$g_c(x,y) = g_c(x) \cdot g_c(y)$$

$$h_c^{(1)}(x,y) = g_c(x) \cdot h_c(y)$$

$$h_c^{(2)}(x,y) = h_c(x) \cdot g_c(y)$$

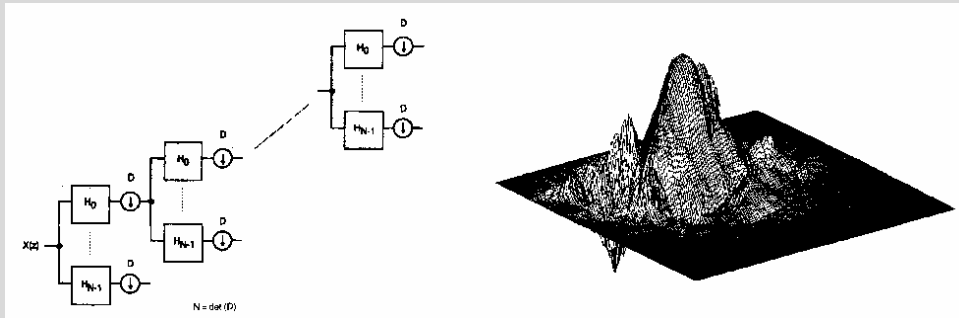
$$h_c^{(3)}(x,y) = h_c(x) \cdot h_c(y)$$

kuras ir orthogonālas viena otrai attiecībā uz veselu skaitu nobīžu. Funkcija  $g_c\{x,y\}$  ir sadalāma divu dimensiju mērogošanas funkcija (t.i., zemfrekvenču filtrs), savukārt funkcijas  $h_c^{(i)}(x,y)$  ir "wavelets". Kopa  $\{h_c^{(i)}(2^j x - k, 2^j y - l), i = 1,2,3 \text{ un } j,k,l \in \mathbf{Z}\}$  veido orthonormālu bāzi kvadrāta integrējamām funkcijām pēc  $\mathbf{R}^2$ . Šis risinājums atbilst divu dimensiju filtru bankai ar divkārtu subsampling katrā dimensijā, t.i., vispārīgi četrkārtu subsampling (skatīt 17. attēlu).



17. attēls. Atda

bojas autonomi – katrs ar savu funkciju  $n(t)$ . Frekvencu plaknes sadausana paraata labaja puse: Hl un Hh norāda zemfrekvenču un augstfrekvenču filtru atbilstoši.



18. attēls. Neatdalāmas filtru bankas iterāciju pakāpes.

Šī konstrukcija noved neatdalāmām wavelets.

19. attēls. Divu dimensiju neatdalāma mērogošanas funkcija

Izšķirtspējas maiņa ir  $2 (\sqrt{2})$  katrā dimensijā

## Wavelet funkciju pielietojums signālu apstrādē

Wavelet transformācija tika radīta kā alternatīva STFT, lai tās galvenais pielietojums būtu nestacionāru signālu analīze. Lai arī konceptuāli CWT ir klasiska konstanta-Q analīze, tās vienkāršā definīcija (balstīta uz vienu funkciju nevis vairākiem filtriem) pieļauj spēcīgus analītiskus atvasinājumus.

Wavelet dekompozīcijas pielietojums skaitliskajā analīzē, risinot diferencu vienādojumus šķiet ļoti progresīvs, jo tai piemīt "zooming" īpašība, kuras izmantošana, pretstatā Furjē transformācijai, pieļauj ļoti labu nenoteiktību attēlošanu.

## Piezīmes

*robustness* ... skaitļošanas sistēmas spēja atjaunot darbu pie kļūdainas situācijas, kura rodas ārēju vai iekšēju apstākļu dēļ. Šāda sistēma pieļauj kļūdas ieejas datus vai nepilnības kaut kādās šīs sistēmas sastāvdaļās.

*orthogonal functions* ...  $f_1(x), f_2(x), \dots, f_n(x)$  ir funkciju kopa, kuras definētas intervālā  $(a, b)$ ,  $w(x)$  – uzdots svaru funkcija intervālā  $(a, b)$ . Tāda veida funkcijas  $f_i(x)$  pie  $i=1, 2, \dots, n$  ir orthogonālas attiecībā pret intervālu  $(a, b)$  un

svaru funkciju  $w(x)$ , ja tiek apmierināts nosacījums  $\int_a^b w(x) f_i(x) f_j(x) dx = 0$  pie  $i \neq j, j=1, 2, \dots, n$ . Ja  $i=j$  tad

$\int_a^b w(x) f_i^2(x) dx = 1$  pie  $i=1, 2, \dots, n$ , tad tādas funkcijas sauc par orthonormālām.

*orthonormal basis* ... orthonormētu funkciju kopa, kuras izmanto locekļu aprēķināšanai, kuri ietilpst kāda pārveidojuma izteiksmē (piemēram, Furjē). Efektīvai pārveidojuma veikšanai nepieciešams, lai funkcijas būtu orthogonālas, bet ne obligāti normētas. Pārveidojuma elementu aprēķināšana ir orthonormāla vai orthogonāla analīze.

*oversampling* ... analogsignāla nolase diskrētos laika momentos jeb iztveršana, kura stipri pārsniedz vidējo vērtību (piemēram, 20 reizes).

*subsampling* ... tas pats, tikai nedaudz zem vidējās vērtības.

*upsampling* ... tas pats, tikai nedaudz virs vidējās vērtības.

# Algoritmi ātrai vektorkvantēšanai (Algorithms for Fast Vector Quantization)

## Ievads

Tuvo kaimiņu problēma ir atrast punktu, kas ir tuvākais meklējamajam punktam starp  $n$  punktu kopumu  $k$ -dimensiju telpā. Vistuvākā kaimiņa atrašana ir ļoti nozīmīga problēma daudzās programmās. Viena svarīga programma ir vektoru kvantēšana – tehnika, ko lieto runas un attēlu saspiešanai. Izmantojot nolases, ko iegūst no signāla, tiek veidoti vektori ar garumu  $k$ . Balstoties uz vektoru apmācošo izlasi, vispirms tiek izskaitļota kodēto vektoru kopa. Tālāk tehnika nokodē katru jauno vektoru ar tā tuvākā kaimiņa indeksu no šīs vektoru kopas. Vektora kvantētāja koeficients  $r$  norāda bitu skaitu, kas tiek lietoti, lai kodētu attiecīgo nolasi. Šis lielums ir saistīts ar  $n$  – kodēto vektoru skaitu, sekojoši:

$$n = 2^k$$

Ja  $r$  ir fiksēts lielums, vektora kvantētāja veikspēja uzlabojas, pieaugot dimensijai, bet, diemžēl, kodēto vektoru skaits aug eksponenciāli ar dimensiju. Principā ir divas pieejas, lai tiktu galā ar šo sarežģītību. Pirmā pieeja ir izmantot kodu ierakstu struktūru, lai tuvākais kaimiņš vai tā aproksimācija varētu tikt ātri atrasta. Veiktspējas pasliktināšanās  $m$  var notikt tāpēc, ka struktūras izmantošana noved pie neoptimālas kodu ierakstīšanas. Otra pieeja ir iepriekš sagatavot nestrukturētu kodu ierakstu, tādā veidā tuvākā kaimiņa meklēšanas sarežģītību samazinot. Lai kā, praktisko algoritmu, kas tiek izmantoti šajā pieejā, sarežģītība strauji aug, palielinoties dimensijai un to darbības laiks ir tikai nedaudz labāks kā secīgai koeficienta meklēšanai apmēram ar 1 bitu uz nolasi vai mazāk (t.i.,  $n \leq 2^k$ ).

Šajā rakstā mēs parādam, ka, ja kāds ir gatavs mazināt prasības patiesā tuvākā kaimiņa atrašanai, tad ir iespējams sasniegt nozīmīgus uzlabojumus darbības ilgumā, vienlaicīgi zaudējot tikai nedaudz vektora kvantētāja veikspējā. Mēs piedāvājam trīs algoritmus tuvākā kaimiņa meklēšanai:

- (1)  $k$ -d koka standarta meklēšanas algoritms, kas ir līdzīgs variantam, ko piedāvāja Spraulli, bet ar uzlabojumiem – attāluma aptuvenais novērtējums tiek aizstāts ar precīzu attāluma vērtību.
- (2) jauns meklēšanas algoritms, kuru mēs saucam par  $k$ -d koka prioritātes meklēšanu, kurā meklēšanas koka šūnas tiek pārbaudītas prioritārā secībā. Šo prioritāti nosaka attāluma lielums no meklējamā punkta.
- (3) kaimiņattiecību grafa meklēšanas algoritms, kurā ir izveidots orientētais grafs punktu kopai un tā šķautnes savieno tuvējos punktus.

Mēs veicām vairākus šo algoritmu eksperimentus ar punktu kopām no dažādiem sadalījumiem un dimensiju diapazonu no 8 līdz 16. Šiem eksperimentiem mēs izmantojām 1 bitu uz nolasi. Mēs pētījām šo algoritmu darbības ilgumus, veicot dažāda veida mērījumus (pārbaudīto punktu skaits, darbību ar peldošo komatu skaits). Tāpat mēs dažādos veidos novērtējām šo algoritmu veikspēju (relatīvā kļūda, attiecība signāls-troksnis, iespējamība neatrast patieso tuvāko kaimiņu). Mūsu pētījumi parādīja, ka daudziem sadalījumiem augstākajās dimensijās pēdējie divi algoritmi nodrošina krasu darbības ilguma samazinājumu salīdzinājumā ar pirmo algoritmu, pie ļoti niecīga veikspējas zuduma.

### **Tuvākā kaimiņa meklēšana, lietojot K-D kokus: Standartpieeja**

Bentlijs ieviesa k-d koku kā binārās meklēšanas koka vispārinājumu augstākās dimensijās. Katrs k-d koka iekšējais mezgls ir saistīts ar hipertaisnstūri un hiperplakni ortogonāli vienai no koordinātu asīm, kura sadala hipertaisnstūri divās daļās. Šīs divas daļas tālāk tiek saistītas ar diviem apakšmezgliem. Telpas sadalīšanas process turpinās līdz datu punktu skaits hipertaisnstūrī nokrītas zem kaut kādas uzdotas sliekšņa vērtības. Šie hipertaisnstūri tiek saukti par kausiem un attiecīgie mezgli – par koka lapu mezgliem. Datu punkti glabājas tikai lapu mezglos, ne iekšējos mezglos.

Šeit ir dots algoritms, lai atrastu tuvāko kaimiņu, lietojot optimizētus k-d kokus, kas aizņem  $O(\log n)$  laiku šajā gadījumā, pie zināmiem datu un meklējamo punktu sadalījuma apstākļiem. Optimizētā k-d koka iekšējie mezgli sadala datu punktu kopu, kas atrodas atbilstošā hipertaisnstūrī, divās vienādās daļās pa dimensiju, kurā datu punktiem ir maksimāla izplatība. Algoritms darbojas vispirms caurejot koku, lai atrastu datu punktus, kas atrodas kausā, kas satur meklējamo punktu. Tad tas pārbauda apkārtesošos kausus, vai tie nekrusto sfēru  $S$ , kuras centrs ir meklējamais punkts un kurai rādiuss ir vienāds ar attālumu starp meklējamo punktu un tuvāko līdz šim apmeklēto datu punktu. Atbilstošā hipertaisnstūra apakšējā un augšējā robeža (*robežmatrica*) var tikt atklāti glabāta katrā no k-d koka iekšējiem mezgliem, tādā veidā palīdzot īsināt meklēšanu, jo kausi kas nekrusto sfēru  $S$ , netiek pārbaudīti. Spraulli pierādīja, ka nebija vajadzības atklāti glabāt robežmatricu un piedāvāja alternatīvu variantu, lai ietaupītu vietu un paātrinātu meklēšanu. Spraulli variants pārbaudīja kausus, vai tie nekrusto mazāko hiperkubu, ko ierobežo sfēra  $S$ . Mēs tālāk pilnveidojām izpildījumu tā, ka kausi tiek pārbaudīti tikai tādā gadījumā, ja tie krusto sfēru  $S$  (šo mēs saucam par *attāluma* uzlabojumu un attiecinām uz algoritmu, kas iekļauj šo uzlabojumu kā standartpieeju). Tas dod ļoti nozīmīgu ietaupījumu augstākās dimensijās, jo ir liela atšķirība starp sfēras tilpumu un tajā ierobežotā hiperkuba tilpumu šajās dimensijās. Piemēram, dodot 65536 punktus 16 dimensijās no nekorelēta Gausa avota, apmeklēto punktu skaits ar un bez šī uzlabojuma ir attiecīgi 14500 un 50000. Šīs vidējās vērtības bija izskaitļotas 25000 meklējamo punktu, arī no tā paša avota. Šiem un visiem citiem eksperimentiem, kas ir aprakstīti šajā rakstā, mēs lietojām optimizētus k-d kokus ar vienu datu punktu uz kausu un mērījām attālumus Eiklīda standartā.

Īsumā, algoritms, kas iekļauj attāluma uzlabojumu, darbojas sekojoši. Katrā apmeklētajā lapas mezglā mēs rēķinām attāluma kvadrātu starp meklējamo punktu un datu punktu kausā un precizējam tuvāko kaimiņu gadījumā, ja šis ir tuvākais punkts no līdz šim apmeklētajiem. Katrā iekšējā mezglā, ko mēs apmeklējam, vispirms mēs meklējam apakškoku, kura atbilstošais hipertaisnstūris ir tuvāk meklējamajam punktam. Vēlāk mēs meklējam tālāko apakškoku un skatāmies, vai attāluma kvadrāts starp meklējamo punktu un tuvāko līdz šim apmeklēto punktu pārsniedz attāluma kvadrātu starp meklējamo punktu un atbilstošo hipertaisnstūri. Lai atvieglotu attāluma kvadrāta starp meklējamo punktu un atbilstošo hipertaisnstūri skaitļošanu, mēs saglabājam mainīgo *cpdistS*, kurš seko šim lielumam. Tāpat mēs saglabājam arī masīvu *cp*, lai sekotu attālumam starp meklējamo punktu un hipertaisnstūri pa katru dimensiju. Tā kā k-d koks ir šķērsots, tad ir iespējams modificēt *cpdist2* un atbilstošo *cp* elementu konstantā laikā katram apmeklētajam mezglam.

### **Tuvākā kaimiņa meklēšana, lietojot K-D kokus: prioritātes pieeja**

K-d koka standarta algoritms parasti sastop tuvāko kaimiņu labu laiku pirms meklēšanas izbeigšanas. Šo papildus meklēšanu var uzskatīt kā maksu, kas jāmaksā, lai tiktu garantēts, ka tuvākais kaimiņš tiks atrasts. Ja mēs būtu ar mieru ziedot šo garantiju, tad sarežģītība varētu tikt samazināta, pārtraucot meklēšanu pirms tās izbeigšanās (pēc fiksēta apmeklēto punktu skaita). Šajā gadījumā ir vēlams veikt meklēšanu tā, ka kausi, kas drīzāk varētu saturēt tuvāko kaimiņu, tiktu apmeklēti ātrāk. Principā šeit tiek ieteikts k-d koka standarta algoritma variants, bet šajā gadījumā k-d koka kausi tiek apmeklēti attāluma no meklējamā punkta pieaugošā secībā.

Algoritms satur apakškoku prioritātes rindu, kur apakškoka prioritāte ir apgriezti proporcionāli saistīta ar attālumu starp meklējamo punktu un hipertaisnstūri, kas atbilst apakškokam. Sākotnēji mēs ievietojam k-d koka sakni prioritātes rindā. Tad vairākkārt realizējam sekojošu procedūru. Vispirms mēs iegūstam apakškoku ar augstāko prioritāti rindā. Tad mēs izejam cauri šim apakškokam, lai atrastu kausu, kas ir tuvākais meklējamajam punktam. Mēs precizējam tuvāko kaimiņu, ja datu punkts kausā ir tuvākais no līdz šim apmeklētajiem punktiem. Tā kā mēs ejam cauri apakškokam, tad katram mezglam  $u$ , ko mēs apmeklējam, mēs ievietojam  $u$  brāli prioritātes rindā. Algoritms beidzas, kad prioritātes rinda ir tukša vai ātrāk, ja attālums no meklējamā punkta līdz hipertaisnstūrim, kas atbilst augstākās prioritātes apakškokam, ir lielāks kā attālums līdz tuvākajam datu punktam.

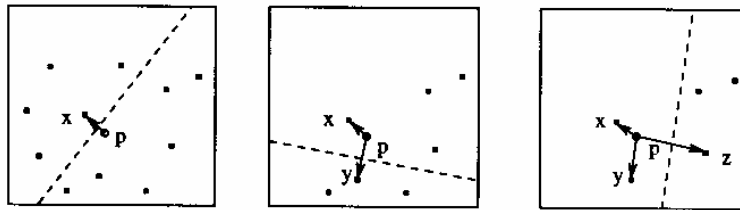
Lai apstiprinātu šī algoritma pareizību, vispirms ievērojiet, ka mēs varam uzskatīt apakškoku kā kausu kopu, kas ir ietverta ar apakškoku saistītajā hipertaisnstūrī. Tālākā prasība ir, lai katras augstākminētās procedūras atkārtotās sākumā saglabātos sekojoša invariantitāte: apakškoki prioritātes rindā ir savstarpēji atdalīti (tiem nav kopīga kausa) un

tos vieno vēl neapmeklēto kausu kopa. Šī prasība balstās uz novērojumu, ka pie katras atkārtotās apakškoks T tiek noņemts no prioritātes rindas un aizstāts ar (0 vai vairākiem) apakškokiem, kuru savienība ir kopa no visiem apakškokā T ietilpstošajiem kausiem mīnus apakškoka T kauss, kas tiks apmeklēts nākamais. Algoritma pareizība ir viegli izsecināma no augstākminētās prasības un fakta, ka apakškoks ar visaugstāko prioritāti satur kausu, kas ir tuvākais meklējamajam punktam.

### Kaimiņattiecību grafi

Šeit mēs dodam īsu pārskatu pieejai tuvākā kaimiņa meklēšanai, kas balstīta uz kaimiņattiecību grafiem. Kaimiņattiecību grafs ir saistītais grafs (orientēts vai neorientēts), kura virsotnes ir datu punktu kopas, tādas, ka divi punkti ir viens otram blakus, ja tie apmierina kaut kādu vietējo kritēriju. Piemēram, Delaunay triangulācija ir neorientēts kaimiņattiecību grafs, kurā divi punkti ir blakusesoši gadījumā, ja ir sfēra, kas iet caur diviem punktiem, un kura nesatur citus punktus tās iekšpusē.

Uzdodot kaimiņattiecību grafu, mēs varam meklēt meklējamā punkta tuvāko kaimiņu, lietojot rijīgo stratēģiju. Mēs uzsākam meklēšanu ar datu punktu  $p$  no  $k$ -d koka kausa, kas satur meklējamo punktu. Mēs vairākkārt realizējam sekojošus soļus. Mēs izvēršam punktu  $p$ , ar ko mēs gribam teikt, ka mēs rēķinām attālumu līdz meklējamajam punktam visiem tiem punkta  $p$  kaimiņiem, kas vēl nav bijuši izvērsti. Starp tādiem kaimiņiem mēs izvēršam punktu, kas ir tuvākais meklējamajam punktam. Šādā veidā mēs turpinām izvērst punktus līdz mēs nonākam pie punkta, kura visi kaimiņi ir jau izvērsti (saka, ka meklēšana ir nonākusi strupceļā), vai arī algoritma apmeklēto punktu skaits pārsniedz kaut kādu iepriekš uzdotu vērtību. Tādā gadījumā mēs pārtraucam meklēšanu un izvadām vistuvāko datu punktu no apmeklētajiem.



1. zīm. Kaimiņattiecību grafs

Kaimiņattiecību grafs, ko mēs lietojam tuvākā kaimiņa meklēšanai ir diezgan līdzīgs relatīvajam kaimiņattiecību grafam (relative neighborhood graph – RNG). RNG-ā divi punkti  $p$  un  $r$  ir blakusesoši tādā gadījumā, ja nav punkta, kas vienlaicīgi ir tuvāks abiem punktiem salīdzinājumā ar attālumu starp šiem punktiem. Modificētais grafs, ko mēs veidojam, ir ekvivalents grafam, kas tika lietots kā starprezultāts to RNG konstruēšanā. Tas ir balstīts uz sekojošu izsvītrošanas likumu. Katram datu kopas  $p$  punktam mēs skatāmies atlikušos punktus attāluma no punkta  $p$  pieaugošā secībā. Mēs izņemam no šīs virknes tuvāko punktu  $x$ , radam orientētu šķautni no  $p$  uz  $x$  un izņemam no tālākas izskatīšanas visus punktus  $s$ , kas atbilst šādam nosacījumam:  $\text{dist}(p, s) > \text{dist}(x, s)$ . Šis process tiek atkārtots līdz visi punkti ir izsvītroti. 1. zīmējums parāda šī procesa piemēru, attiecinātu uz punktu  $p$ , trijos posmos; katrā posmā jaunā šķautne ir orientēta no  $p$  uz vienu no tā kaimiņiem. Punkta  $p$  trīs kaimiņi kaimiņattiecību grafā ir  $x$ ,  $y$  un  $z$ . Šis variants, saukts par RNG\*, var tikt izskatīts  $O(n^2)$  laikā, kur  $n$  ir punktu skaits.

Lai arī rijīgā algoritma sliktākais gadījums var būt diezgan sliktas, mūsu eksperimentālie pētījumi parādīja, ka augstās dimensijās meklēšana ātri beidzas, atrodot tuvāko kaimiņu, un tikai retums tās nonāk strupceļā pirms tuvākā kaimiņa atrašanās.

### Empīriskā analīze

Pirms eksperimenta uzsākšanas mēs optimizējām vairākus koda aspektus visiem trim algoritmiem. Mēs minēsim divus no tiem. Pirmā, labi zināmā *daļējā distances* optimizācija bija īstenota visiem trim algoritmiem: kad mēs skaitļojam attāluma kvadrātu starp meklējamo punktu un datu punktu, summējot katras dimensijas devumu, mēs izejam no cikla, kad uzkrātā kvadrātu summa kļūst pārāk liela. Šī optimizācija samazināja kopējo darbību skaitu ar peldošo komatu pie tikai neliela salīdzinājumu skaita pieauguma. Otra lieta, RNG\* meklēšanai mēs noglabājam daļējā attāluma skaitļošanas rezultātus tā, lai tie varētu tikt atkal lietoti, ja tas pats punkts tiktu sastapts, izvēršot vairākus atšķirīgus punktus.

Mēs izpētījām, kā vektora kvantētāja veikspēja mainās, mainoties algoritma sarežģītības funkcijai. Mēs fokusējāmies uz ātrumu 1 bits uz nolasi dimensiju diapazonā no 8 līdz 16. Katram no trim algoritmiem meklēšana tiek pārtraukta, ja algoritma apmeklēto punktu skaits sasniedz noteiktu sliekšni, un tādā gadījumā līdz šim tuvākais no apmeklētajiem punktiem tiek ņemts kā algoritma izejas lielums. Mainot šo sliekšni (sliekšņa vērtību), katra algoritma sarežģītība var tikt mainīta. Mēs katram eksperimentam lietojām 25000 meklējamo punktu un atzīmējām sekojošo katram meklējamajam punktam pie katra sliekšņa ( $d_n$  šeit ir attālums starp meklējamo punktu un datu punktu, ko izveda algoritms,  $d_n$  ir attālums starp meklējamo punktu un tā tuvāko kaimiņu):

- Datu punktu skaits, kurus apmeklēja algoritms. Datu punkts tiek uzskatīts par apmeklētu, ja algoritms ir sasniedzis tā koordinātes. Katrs datu punkts lielākoties tiek skaitīts vienreiz šajā kopsummā. Ja algoritms beidz

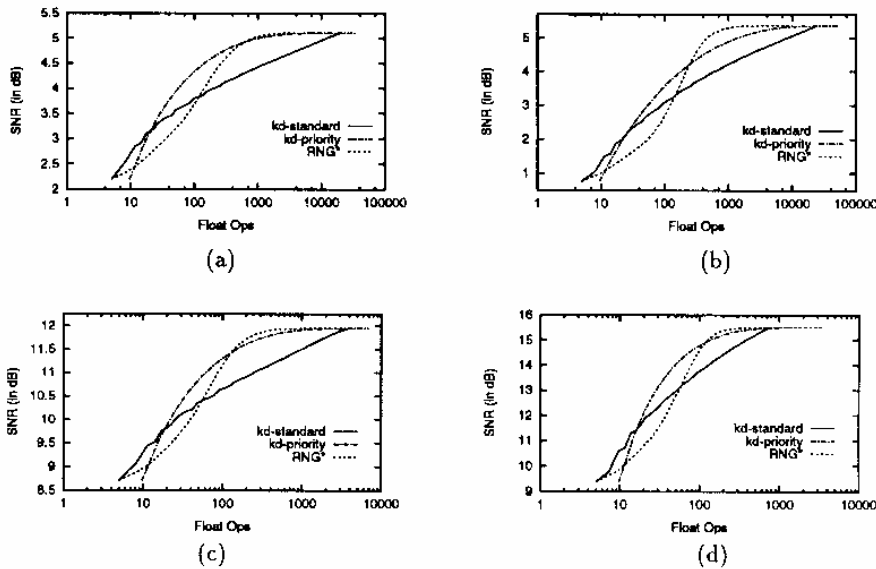
savu darbību pirms ir sasniegts sliekšnis, tad šis daudzums ir tāds pats kā līdz algoritma izbeigšanai apmeklēto punktu skaits, pretējā gadījumā tā ir sliekšņa vērtība.

- Operāciju ar peldošo komatu skaits uz nolasi. Šeit mēs ietvērām visas peldošā komata saskaitīšanas, atņemšanas, reizināšanas un salīdzināšanas (izņemot salīdzināšanu ar nulli), kuras veica algoritms. Netika ietvertas tikai tās operācijas, kas bija iesaistītas attālumu rēķināšanai starp punktiem. Operāciju ar peldošo komatu kopsumma tiek dalīta ar dimensiju, lai iegūtu to vidējo skaitu uz nolasi.
- Vai patiesais tuvākais kaimiņš ir jau atrasts.
- Kļūdas faktors, kurš ir definēts kā  $(d_e - d_n)/d_n$ .
- Kropļojums uz nolasi, kurš ir definēts kā  $(d_e)^2/k$ , kur k ir dimensija.

Trīs veikspējas mērījumi tika izrēķināti pie katra sliekšņa – signāls-troksnis līmenis (the signal-to-noise ratio – SNR), vidējais kļūdas faktors un kļūmes iespējamība (iespējamība neatrast tuvāko kaimiņu). SNR ir definēts kā  $10 \log_{10}(V/D)$ , kur V ir nolašu neatbilstība un D ir vidējais kropļojums uz nolasi. Visas šīs vidējās vērtības ir ņemtas no visas meklējamo punktu kopas. No šiem veikspējas mērījumiem SNR ir visnozīmīgākais vektoru kvantēšanā. Pārējie divi mērījumi šeit galvenokārt ir, lai palīdzētu labāku algoritmu empīrisko saprašanu. Divi sarežģītības mērījumi tika veikti pie katra sliekšņa – vidējais apmeklēto punktu skaits un vidējais operāciju ar peldošo komatu skaits uz nolasi. Apmeklēto punktu skaits ir noderīgs, bet, kopš algoritmam ir dažādi papildus uzdevumi, operāciju ar peldošo komatu skaits ir daudz tiešāk saistīts ar algoritmu sarežģītību. Mūsu pētījumi parādīja, ka operāciju ar peldošo komatu skaits ir saprātīgs meklēšanas laika mērījums un var tikt izmantots algoritmu salīdzināšanai.

Mēs veicām eksperimentus, lietojot Gausa un Laplasa avotus. Tika lietoti gan nekorelēti, gan korelēti avoti. Korelētajiem avotiem mēs lietojām 0,9 kā korelācijas koeficientu. Visiem avotiem bija nulles vidus un vienu vienību liela dispersija.

16. dimensijā mēs lietojām kodu ierakstus, kas sastāv no 65536 kodu vektoriem, kas iegūti no uz k-d koku balstītā Equitz-a algoritma. Mēs dažādos veidos paātrinājām Equitz-a algoritmu, un nekorelētajiem avotiem tā vietā, lai būvētu pārdomātus k-d kokus, kā tas ir ierasts, mēs sadalījām hipertaisnstūrus atbilstoši iekšējiem mezgliem tā, ka katra daļa saturēja nejašu punktu skaitu. Tas noveda pie labākas kvalitātes kodu ierakstiem. Apmācošās izlases izmērs, kas tika lietots bija 32 reizes lielāks par kodu ierakstu izmēru. Mūsu eksperimentos apmācošā kopa un pārbaudāmā kopa bija atšķirīgas.



2. zīm. SNR pret vidējo operāciju ar peldošu komatu uz nolasi, (a) nekorelētam Gausa, (b) nekorelētam Laplasa, (c) korelētam Gausa, (d) korelētam Laplasa avotam.

1. tabula Peldošā komata operācijas uz nolasi, lai sasniegtu SNR 0,1 dB robežās no SNR-MAX

SADALĪJUMS	SNR-MAX	KD-STANDARTA	KD-PRIORITĀTES	RNG*
Nekorelēts Gausa	5.11	12000	1100	850
Nekorelēts Laplasa	5.36	18500	4500	850
Korelēts Gausa	11.94	2500	550	300
Korelēts Laplasa	15.51	650	400	200

2. tabula Peldošā komata operācijas uz nolasi, lai sasniegtu SNR 0,01 dB robežās no SNR-MAX

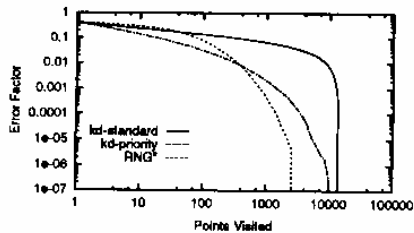


SADALĪJUMS	SNR-MAX	KD-STANDARTA	KD-PRIORITĀTES	RNG*
Nekorelēts Gausa	5.11	19000	5000	2000
Nekorelēts Laplasa	5.36	24000	15000	2000
Korelēts Gausa	11.94	3700	1700	600
Korelēts Laplasa	15.51	800	950	450

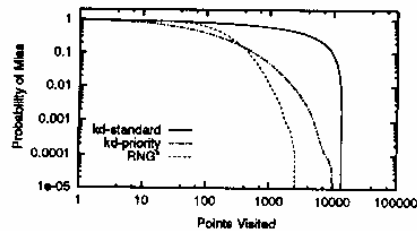
2. zīmējums parāda SNR variācijas ar vidēju operāciju ar peldošo komatu skaitu uz nolasi attiecīgi nekorelētam Gausa, nekorelētam Laplasa, korelētam Gausa un korelētam Laplasa avotiem. Uzmanīgākai augstas veiktspējas apgabalu izpētei (teiksim, mazāk kā 0,1 dB pasliktināšanās no veiktspējas, kas iegūta no pilnībā izsmelšanas meklēšanas, SNR-MAX), mēs izveidojam divas tabulas. Pirmā tabula salīdzina trīs algoritmus pēc vidējā operāciju ar peldošo komatu skaita uz nolasi, kas nepieciešamas, lai sasniegtu SNR 0,1 dB robežās no SNR-MAX. 2. tabula rāda kas vajadzīgs, lai sasniegtu SNR 0,01 dB robežās no SNR-MAX. Vietas trūkuma dēļ, mēs apkopojām galvenos novērojumus 16. dimensijai augstas veiktspējas apgabalā:

- RNG\* meklēšana ir ātrākā no trim algoritmiem, aiz tās seko k-d koka prioritātes algoritms. Abi šie algoritmi bieži vien dod ievērojamus paātrinājumus, dažreiz ar koeficientu pāri par 10, salīdzinot ar k-d koka standarta algoritmu.
- RNG\* meklēšanas sarežģītība mainās no tikai nedaudz labākas kā k-d koka prioritātes algoritmam līdz daudz labākai, dažreiz sasniedzot paātrinājumu ar koeficientu pāri par 5.
- Visi trīs algoritmi sasniedz ievērojamus paātrinājumus pār pilnībā izsmelto meklēšanu ar nenozīmīgiem veiktspējas zudumiem (mazākiem par 0,01 dB). RNG\* meklēšana sasniedz milzīgu paātrinājumu ar koeficientu pāri par 100 salīdzinot ar pilnībā izsmelto meklēšanu. Pat k-d koka standarta algoritms sasniedz paātrinājumu ar koeficientu pāri par 8.
- Divi k-d koka algoritmi pieprasa vienādu atmiņu, kamēr RNG\* meklēšana var pieprasīt divreiz vairāk (atkarībā no avota un izpildes). Attiecībā uz noteiktu dimensiju un ignorējot vietu, kas nepieciešama datu punktiem, atmiņas prasības visiem trim algoritmiem ir  $O(n)$ . K-d koka algoritma proporcionalitātes nemainīgums lielā mērā nav atkarīgs no dimensijas, kamēr RNG\* meklēšanai tas rāda mērenu eksponentveida augšanu. Vienveidīgam sadalījumam empīriskie pētījumi rāda, ka RNG\* pakāpe aug aptuveni par  $2,90 (1,24^k)$  asimptotiskā gadījumā, un par  $1,46 (1,20^k)$ , kad punktu skaits aug kā  $2^k$ . RNG\* izveides izmaksas ir  $O(n^2)$ , kamēr k-d koka izveides izmaksas ir  $O(n \log n)$ . Šī milzīgā atšķirība ir tāpēc, ka punktu skaits ir tik liels.

Nekorelētam Gausa avotam 3. zīmējums parāda, kā vidējais kļūdas faktors un iespējamība neatrast tuvāko kaimiņu mainās atkarībā no vidējā apmeklēto punktu skaita. Šie grafiki parāda, ka abi šie lielumi nokrīt daudz ātrāk RNG\* meklēšanas gadījumā, nekā k-d koka prioritātes algoritmam. Tas vedina uz domām, ka daudzās programmās, kur šie lielumi ir būtiski, RNG\* meklēšanai varētu būt ļoti nozīmīgas priekšrocības salīdzinājumā ar k-d koka prioritātes algoritmu.

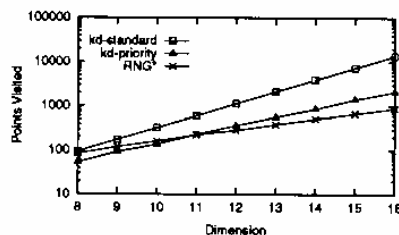


(a)



(b)

3. zim. Nekorelēts Gausa avots. (a) Vidējais kļūdas faktors pret vidējo apmeklēto punktu skaitu. (b) Iespējamība neatrast tuvāko kaimiņu pret vidējo.



4. zīm. Nekorelēts Gausa avots: vidējais apmeklējamo punktu skaits, kas nepieciešams, lai sasniegtu kļūdas faktoru 0,001.

Tāpat mēs veicām eksperimentu nekorelētam Gausa avotam dažās citās dimensijās robežās no 8 līdz 16, lietojot nejaušus kodu ierakstus. Katrā gadījumā mēs lietojām 1 bitu uz nolasi. Mēs atklājām, ka visās šajās dimensijās k-d koka prioritātes algoritms un RNG\* meklēšana ir daudz ātrāki nekā k-d koka standarta algoritms. Dimensijās, kas ir zem 11, mēs konstatējām, ka k-d koka prioritātes algoritms ir ātrāks nekā RNG\* meklēšana, kamēr dimensijās virs 11 RNG\* meklēšana ir ātrāka augstas veiktspējas apgabalos. Tas ir redzams 4. zīmējumā, kurš dažādās dimensijās parāda vidējo punktu skaitu, ko ir apmeklējuši trīs algoritmi, lai sasniegtu vidējo kļūdas faktoru 0,001.

## Multimediju ievad/izvad termināliekārtu projektēšana (System Design of a Multimedia I/O Terminal)

Nākotnes personālo sakaru sistēmu panākumu atslēga varētu būt gala lietotāja iekārtas vai termināli, kas dos lietotājiem neierobežotu pieeju noteiktiem multimediju informācijas serveriem. Šajā rakstā ir atspoguļoti multimediju ievad/izvad terminālu projektēšanas pamatrisinājumi, kas uz šo brīdi ir jau ieviesti. Portatīvuma prasības uzliek nopietnus ierobežojumus enerģijas patēriņam, prasot tā samazināšanu. Tas ietver skaitļošanas un optimizācijas algoritmu, arhitektūru, shēmu veidu un tehnoloģiju sadalīšanu. Klāt pie enerģijas patēriņa samazināšanas un savienojumu caurlaidspējas prasībām, portatīva operācija rada papildus problēmas sistēmas projektēšanā, jo jātiek galā ar daudzām kļūdām attiecīgajos bitos, kas rodas bezvadu sakaru gadījumā.

### Ievads

Pēdējo gadu laikā personālo sakaru lauks ir piedzīvojis strauju augšupeju tieši pēc pakalpojumu skaita un tehnoloģiju veidiem, kuri ir kļuvuši viegli pieejami. Balsis tipa sakaros tādas sistēmas kā mobilā analogā telefonija, radio peidžeri un bezvadu telefoni, ir kļuvušas ikdienišķas, par spīti to ierobežotajai dabai un dažreiz diezgan sliktai pārraides kvalitātei. Portatīvajā skaitļošanā "notebook" datoriem, pateicoties multi-MIPS uz RISC bazētu portatīvu darbstaciju ātru parādīšanos, ir iespējas, kas ievērojami pārsniedz galda tipa datorus, kādi bija pieejami pirms pieciem gadiem. Lai kā, neskatoties uz pašreiz pieejamo tehnoloģiju lielo daudzumu, joprojām par maza ir šo dažādo pakalpojumu, skaitļošanas apvienošanas un sakaru integrācija vienā portatīvajā mezglā. Tādā veidā mūsu personālās sakaru sistēmas nākotnes vīzija tiek centrēta uz tādu pakalpojumu integrāciju, kas nodrošinātu visuresošu pieeju datiem, izmantojot specializētu, bezvadu multimediju terminālu.

Kā jau bija pārrunāts, mūsu skats uz bezvadu terminālu ir, ka tas būs pilna duplexa komunikācijā ar tīkla bāzes staciju, kura kalpos kā vārti starp vadu un bezvadu vidēm. Tādā veidā, caur bāzes staciju, lietotājs iegūs vēlamos pakalpojumus, izmantojot ātrdarbīgus sakarus, tajā skaitā sazinoties arī ar citu personu, kas ir pievienota tīklam. Lai kā, šī doma var tikt izvērsta līdz lietotājam, kas sazināsies ne tikai ar citu personu, bet ar tīkla "serveriem". Kopš nākotnes optisko šķiedru tīkla datu caurlaidspējai ir viegli pārsniegt 10 gigabitus/sekundē, šie centralizētie serveri lietotājam var nodrošināt plašu informācijas pakalpojumu dažādību. Dažas personālās sakaru sistēmas pamatpazīmes iespējams, ka ietvers:

- Pieeju lielām komerciālajām datu bāzēm, kas saturēs dažāda veida informāciju, tādu kā ziņas, finanšu informācija, tirdzniecības dati utt.
- Pieeju ciparu video datu bāzēm, kas satur gan izklaidējošus, gan izglītojošus materiālus.
- Vienkāršotu ieejas mehānismu, tādu kā balsis atpazīšana un rokraksta atpazīšana, kas koordinēs pieeju augstāk minētajām funkcijām.

Šis raksts fokusēsies uz kritiskām multimediju termināla projektēšanas problēmām. Tas ietvers:

- Portatīvums, kurš uzliek ierobežojumu izmēram, svaram un enerģijas patēriņam.
- Bezvadu sakari, ar iespēju optimizēt pārraides ātrumu, atkarībā no prasībām.

Veiktspējas optimizēšana katrā no šiem laukiem ir kritiska, lai apmierinātu visas sistēmas veiktspējas prasības, nodrošinot mazu, vieglu terminālu personāliem sakariem.

### Termināla portatīvums

Termināla portatīvuma prasība uzliks stingrus izmēra, svara un enerģijas patēriņa ierobežojumus. Tālāk tiks apskatītas dažādas stratēģijas, kas tiek lietotas, lai samazinātu enerģijas patēriņu sistēmas līmenī.

### Enerģijas pieejamība

Viens no primārajiem mērķiem portatīvas sistēmas projektēšanā ir enerģijas samazināšana, kas ir vajadzīga, lai palielinātu darbības laiku un reizē izmantotu mazāka izmēra un svara baterijas. Diemžēl uzlabojumi bateriju tehnoloģijā pēdējo 30 gadu laikā ir bijuši diezgan pieticīgi, ar jaudas (Vatstundas/mārciņu) pieaugumu Niķeļa-Kadmija baterijām tikai divas reizes, sasniedzot šobrīd vērtību 20 Wh/1b. Vienīgā piemērotā jaunā tehnoloģija, kura pārspēj NiCd, dažos pēdējos gados ir Niķeļa-Metāla Hidrīda, kas nodrošina ne tikai paaugstinātu ražību (30-35 Wh/1b), bet ir arī apkārtējai videi droša. Lai kā, tuvāko 5 gadu plāni, pat šai jaunajai tehnoloģijai, paredz uzlabojumu tikai par kādiem 30%.

Tādā veidā mēs varam saprātīgi novērtēt enerģijas pieejamību nākotnes portatīvajām sistēmām. Piemēram, pieņemot, ka 4 mārciņas ir augšējā robeža portatīvās sistēmas svaram,  $\frac{1}{2}$  no kura ir baterijas; 8 stundu darba laikā vidējai enerģijai, kas lietota visām funkcijām, jābūt mazākai kā 8 vatiem. Daudzums, kas var tikt izmantots skaitļošanai, ir pat mazāks, kā tas ierobežotais daudzums, tajās programmās, kas prasa augstas kvalitātes video displeju. Lai arī šodienas displeji izlietotu visu šo enerģijas daudzumu, šķiet saprātīgi sagaidīt, ka nākamajos 5 gados tiks sasniegts līmenis, kad displejs patērēs dažus vatus, atstājot tikai 2-3 vatus, lai īstenotu visu skaitļošanu un ievad/izvad atbalstu mūsu relatīvi smagajā portatīvajā ierīcē.

Varētu cerēt, ka uzlabojumi izgatavošanas tehnoloģijā varētu būt risinājums, lai nodrošinātu skaitļošanu, patērējot nelielu enerģijas daudzumu. Diemžēl pēdējo laiku "progress" mikroprocesoru projektēšanā ir parādījies, ka vienas mikroshēmas enerģijas patēriņš pārsniedz 30 vatus, savukārt, pilna augstāku frekvenču ekspluatācija noved pie dramatiskas enerģijas patēriņa pieauguma, un nevis pie nepieciešamās šīs enerģijas samazināšanas. Lai kā, ļoti zemu spriegumu operācijas kopā ar arhitektūras projektēšanas stratēģiju var samazināt enerģijas prasības. Tomēr arī tas nebūs pietiekami, lai padarītu iespējamu portatīvās elektronikas visai zraujošāko nākotnes programmu darbību; tās papildus prasīs nākotnes visuresošo bezvadu sakaru tīklu lietošanu.

### Skaitļošanas sadalīšana

Zemas enerģijas ieviešanas problēma, universāla apstrāde (X-serveris, vārda apstrāde utt.) pirmām kārtām ir saistīta ar apakšsistēmu enerģijas pazemināšanas stratēģiju ar ierobežotu sprieguma mērogošanas daudzumu. Tas ir efektīvi zināmu laiku, bet par cik ir grūti noteikt, kad programma patiešām darbojas tukšgaitā, ir vajadzīgas modifikācijas arī programnodrošinājumā un operētājsistēmā, lai pilnībā izmantotu procesora bezdarbību. Diemžēl par primāro datora, kas lieto vienu no standartprocesoriem, veikspējas noteikšanas faktoru ir kļuvusi frekvence, kas noved pie arhitektūrām, kuras ir antagoniskas zema enerģijas patēriņa operācijām. No otras puses, paralēlas arhitektūras, kas darbojas pie samazinātas frekvences, var sasniegt to pašu veikspējas efektivitāti (instrukciju skaita sekundē izteiksmē), pie būtiski samazinātiem enerģijas līmeņiem. Cita stratēģija, lai samazinātu universālai skaitļošanai nepieciešamo enerģiju, ir efektīvi "noņemt" skaitļošanu no portatīvās iekārtas. Tas tiek paveikts, nodrošinot sakaru iespējas skaitļošanas resursiem vai nu fiksētajai bāzes stacijai otrā bezvadu savienojuma galā, vai skaitļošanas serveriem, kas ir pieejami tīklā. Šī ir vienkārša labi zināmās klienta-serveris arhitektūras pielietošana, kurā portatīvā iekārta deģenerējas līdz ievad/izvad serverim (piemēram, bezvadu X-termināls). Pamatsratēģija, ko mēs lietojam teksta/grafikas atbalstam, ir pārraidīt datus pa "uplink", veikt apstrādi (X-serveris kā arī klienta programmas) bāzes stacijā vai attālinātā skaitļošanas serverī, kas ar vadiem ir pievienots šai stacijai, un pārraidīt pieaugošos "bitmapus" no X-servera caur "down-link". Tādējādi vairums aprēķinu teksta/grafikas atbalstam ir noņemti no termināla, atstājot tikai teksta/grafikas kadru buferi. Līdzīgi noteikta lietojumprogramma specifificē uzdevumu, tādu kā runas un rokraksta atpazīšana, pārvietošanos uz šo apstrādes mezglu. Ar PCM metodi iegūtās runas nolases (64 Kbit/s) tiek pārraidītas atpakaļ un uz priekšu caur bezvadu savienojumu. Kopš video ir pieejams caur bezvadu tīklu ierobežotā ātrumā (<2 Mbit/s uz lietotāju), video tiek pārraidīts kompresētā formātā un tādejā dekompresija ir jāveic terminālam. 1a zīmējums parāda "master" kontrolera mikroshēmu, kas konstruēta, lai veiktu interfeisa funkcijas ar dažādām ievad/izvad iekārtām, ieskaitot seriālo portu (bezvadu puse), "pen" tipa ieeju, runas kodeku un teksta/grafikas displeju. Kontrolers darbojas ar 1,5 V spriegumu (1,2 μm CMOS tehnoloģija, 9,1 x 9,1 mm) un lieto līmeņa pārveidošanu, lai sazinātos ar ievad/izvad iekārtām, kas darbojas pie augstākiem spriegumiem (5 V).

### Optimizējošie algoritmi

Operāciju skaita samazināšana ir būtiska, lai samazinātu enerģijas patēriņu. Lai secīgi ilustrētu, kā algoritmus var optimizēt mazam enerģijas patēriņam, apskatīsim video dekompresijas moduli. Dažādi algoritmi, ieskaitot kodešanu un vektoru kvantēšanu, tiek plaši pētīti un lietoti attēla saspiešanai. Uz transformāciju bāzēta kompresija, lietojot diskreto kosinusa transformāciju (DCT), ir sevišķi populāra un tā formē pamatu vairumam attēlu kompresijas standartiem (JPEG un MPEG). Viena DCT iezīme ir, ka tā ir simetriska, t.i., kodēšanas un dekodēšanas ir vienādas sarežģītības, kā tas redzams 1. tabulā. Cita pieeja ir lietot vektoru kvantēšanu, kurai ir sarežģītāks kodēšanas, bet ļoti vienkāršs dekodēšanas, padarot to ļoti piemērotu sistēmām ar vienu koderu un vairākiem dekoderiem. Šajā gadījumā vektoru kvantēšanas risinājums nodrošina dekompresiju reālā laikā, lietojot ļoti maz skaitļošanas un enerģijas.

#### 1. tabula. Kompresijas algoritmu sarežģītība.

Metode	Skaitļošanas prasības	
	koders	dekoders
DCT ātrais algoritms	4 reizināšanas 8 saskaitīšanas 6 pieejas atmiņai	4 reizināšanas 8 saskaitīšanas 6 pieejas atmiņai
Vektora kvantēšana (Koka meklēšana – diferenciālais CB)	8 reizināšanas 8 saskaitīšanas 8 pieejas atmiņai	1 pieeja atmiņai

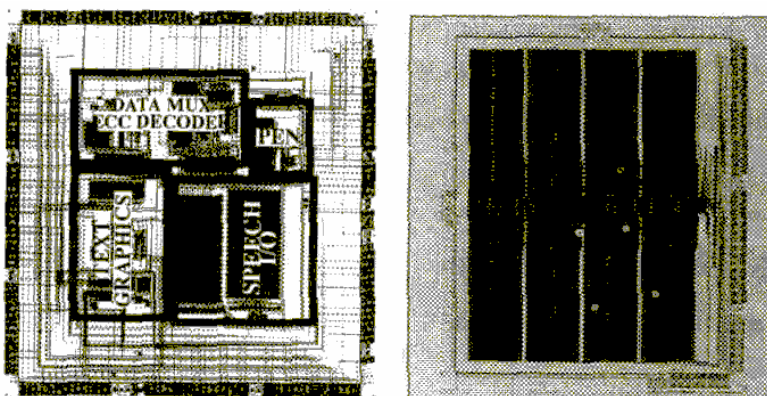
Optimizējošās arhitektūras

Galvenais devums enerģijas patēriņā CMOS ķēdēs tiek attiecināts uz parazītisko kapacitāšu uzlādešanu un izlādešanu, kas notiek loģisko pāreju laikā. Vidējā CMOS pārslēgšanās enerģija ir vienāda ar  $C_{avg} \cdot V_{dd}^2$ , kur  $C_{avg}$  ir vidējā kapacitāte, kas tiek pārslēgta takts laikā un  $V_{dd}$  ir saņemtais spriegums. Sprieguma enerģijas kvadrātiskā atkarība norāda, ka darbība pie iespējami zemākiem spriegumiem ir visvēlamākā, lai optimizētu skaitļošanas enerģiju; diemžēl, piegādājamā sprieguma samazināšana notiek uz skaitļošanas veiktspējas reducēšanās rēķina.

Viens veids kā kompensēt šīs pieaugošās aizkaves ir pielietot arhitektūras, kas samazina darbību ātruma prasības, tajā pašā laikā saglabājot nemainīgu veiktspēju. Lai ilustrētu šo variantu apskatīsim teksta/grafikas kadru buferi. Kadru buferis tiek aizrakstīts ar X-servera caur RF savienojumu un tiek nolasīts uz displeja. Kamēr bufera aizpildīšana no X-servera ir asinhrona un lēna, nolasīšana ir sinhronizēta ar displeja kadru frekvenci (60 Hz). Tā kā nolasīšanas darbības notiek daudz biežāk kā ierakstīšanas, sistēma tiek optimizēta nolasīšanas darbībām. Tradicionālā pieeja ir nolasīt datus sērijveidā; lai kā, bet vairāki pikseļi var tikt nolasīti arī paralēli (mūsu sistēmā 32 pikseļi), tādējādi samazinot ātruma prasības atmiņai. Šādā veidā mēs varam darbināt kadru buferi, izmantojot piegādāto spriegumu diapazonā 1,2 V – 1,5 V (un tādējādi arī zemu enerģijas līmeņus), tajā pašā laikā apmierinot augstās atjaunošanās frekvenču prasības. Ievērojiet, ka X-servera rakstīšana ir nejausa, un tādējādi serverim jābūt modificētam, lai padarītu rakstīšanu secīgu (tādā veidā ļaujot zema enerģijas patēriņa īstenošanu).

Shēmu veids un tehnoloģija

Izvēloties un optimizējot pamatshēmu pieeju un topoloģiju, ir daudz iespējas dažādu loģisko un aritmētisko funkciju īstenošanā. Mēs esam izveidojuši šūnveida bibliotēku, kas ir optimizēta zema enerģijas patēriņa operācijām. Galvenās šīs bibliotēkas pazīmes ir: darbības pie zemiem spriegumiem (1,2 – 1,5 V), samazināts tranzistoru skaits, lai īstenotu dažādas funkcijas, minimāla (cik vien iespējams) izmēra tranzistoru lietošana, vidēja pārslēgšanās darbību samazināšana un izkārtojuma optimizācija, lai samazinātu attālumu starp savstarpēji saistītiem elementiem un reducētu to izsuktās parazītiskās kapacitātes. Piemēram, SRAM lieto pašnoteicošu elektrisko shēmu, lai samazinātu kļūdu aktivitāti, aktivizē tikai vienu bloku konkrētā brīdī, lai samazinātu pārslēgšanās kapacitātes, lieto samazinātu svārstību spriegumu, darbojas ar piegādājamo 1,5 V spriegumu un patērē ap 1 mW pie 10 MHz darbības. 1b zīmējumā redzama SRAM mikroskāmas fotogrāfija. Sapakošanai mēs lietojam multi-mikroskāmu moduļus un mikroskāmas uz platēm, kuras radikāli samazina saistīto mikroskāmu savstarpējo savienojumu kapacitātes.



1a zīm. “Master” kontrolera mikroskāma ar “Pen”, runas un displeja interfeisu.

1b zīm. 64 Kbit SRAM teksta/grafikas kadru buferim.

Bezvadū sakari

Bezvadū sakari ietver pieejamo sakaru caurlaidspējas ierobežojumu un vajadzību projektēt augstām bitu kļūdu normām. Šie divi ierobežojumi tiks apskatīti tālāk.

Spektrālā efektivitātē

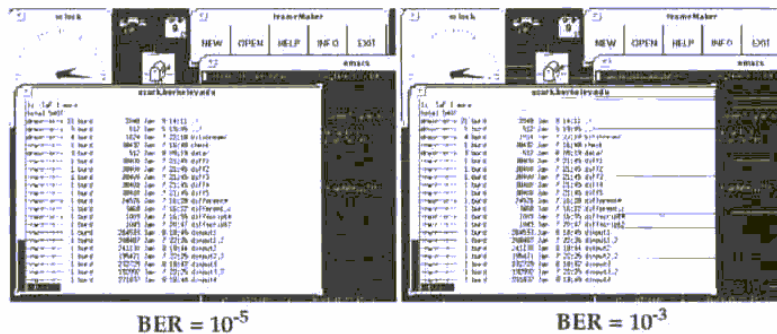
Lai atbalstītu multimediju datu dažādus tipus, tādus kā video (pārraidītu saspiegtā veidā), tekst/grafika (pārraidītus pieaugošu "bitmapu" veidā) un runa, mums vajag vidēji ap 2 Mbit/s uz lietotāju, un tādēļ spektra lietošana ir ļoti būtiska. Mūsu pieeja šīs problēmas risinājumam ir izmantot šunveida tīklošanas tehniku, lai sasniegtu telpisku vairākkārtēju lietošanu. Atšķirībā no pierastā izmēra šūnām, kurām ir vairāku kilometru lieli rādiusi, mēs lietojam ļoti mazas pikošūnas (ar rādiusu ap 5 m). Pikošūnu lietošana nozīmē, ka mums vajadzēs ap 100 MHz RP spektru, lai uzturētu lietotāju augsto blīvumu (~8-10 lietotāju 5 m rādiusā), kamēr pierasto izmēru šūnu lietošana prasītu vairāk kā GHz RF spektru.

Ir vēl viena nozīmīga priekšrocība pikošūnu bezvadu sistēmām: tā kā pārraidāmās jaudas ir kritušās, jo šūnas ir pārvietotas tuvāk viena otrai, lai samazinātu interferenci, tad enerģija, ko patērē portatīvajā raidītājā, arī ir atbilstoši samazināta. Ja eksistējošās šunveida sistēmas izlieto 1 vatu pārraidīšanas jaudas priekš RF savienojumiem 5 jūdžu šūnās, pikošūnu sistēmai ar 5 metru šūnām nepieciešams mazāk kā viens milivats, lai uzturētu savienojumu.

Kļūdu iespējamība

Bezvadu sakari dabiski sevī ietver trokšņainu pārraidīšanas vidi. Salīdzinoši, vadu "ethernet" kļūdu labošanas efektivitātē ir daudz zemāka kā  $10^{-12}$ , pateicoties ļoti tīrai kanāla dabai. Mūsu sistēmām, ar vienlaicīgi uz pilnu jaudu komunicējošiem visiem lietotājiem, mēs esam novērtējuši sliktāko kļūdu labošanas efektivitātes gadījumu  $10^{-3}$ .

Uzdodot šo parametru, ir ņemts vērā teksta/grafikas shēmu, kuras pārraida pieaugošos "bitmapus" caur RF savienojumu, robustums. Protokols, kas tiek lietots, ir domāts, lai pārraidītu bāzes adresi, garumu un "bitmapu" datus. Kļūdas parādās 2 veidos; kļūdas "bitmapu" datus, kuras parādās kā punkti uz ekrāna, kā tas redzams 2b zīmējumā, un kļūdas kontroles informācijā (adrese un garums), kuras parādās kā līnijas uz ekrāna. Mūsu pieeja ir aizsargāt kontrolinformāciju un atstāt "bitmapu" datus neaizsargātus. Mēs redzam, ka pat ļoti zema kļūdu labošanas efektivitāte ( $10^{-3}$ ), mēs joprojām iegūstam "pieņemamu" veiktspēju. Uz caurlaidspējas rēķina var iegūt labāku kvalitāti, nodrošinot papildus kļūdu aizsardzību kontroles datiem.



2. zīm. Pieeja zema kļūdu labošanas efektivitātei: pārraidi vairāk "datus" un minimālu "kontroli".

Priekš video ir trīs pamatiezīmes, kas tiek lietotas, lai padarītu to robustu pie zemas kļūdu labošanas efektivitātes:

- Ping-pong kadru bufera shēma, kas nodrošina imunitāti pret kļūdām, t.i., iepriekšējais kadrs tiek atkārtots gadījumā, ja pakete ir pazudusi.
- Vektora kvantēšanas dekodēšana, kura lokalizē kļūdas telpā, t.i., kļūdas kodvārdā, kas tiek pārraidīts, parādās kā samaitāti 4x4 bloki uz ekrāna un nesamaitā visu ekrānu.
- Intra-kadra kompresija, kura lokalizē kļūdas laikā, t.i., kļūdas neuzkrājas, laikiem ejot.

Kopsavilkums

Nākotnes personālo sakaru sistēmu panākumu atslēga varētu būt gala lietotāja iekārtas vai termināli, kas dos lietotājiem neierobežotu pieeju noteiktiem multimediju informācijas serveriem. Šo terminālu prasības pēc portatīvuma uzliek smagus ierobežojumus kopējam enerģijas patēriņam. Lai minimizētu enerģijas patēriņu, ir nepieciešams izmantot sakarus starp terminālu un skaitļošanas resursiem bāzes stacijā un ātrdarbīgu tīklu, padarot terminālu par ievad/izvad iekārtu. Citi paņēmieni zema enerģijas patēriņa sasniegšanai ietver optimizējošus algoritmus minimālai pārslēgšanās aktivitātei, paralēlu arhitektūru pielietojumus, kas dod iespēju lēnām un zema sprieguma darbībām, atmiņas struktūru optimizēšanu, lai samazinātu kapacitāti un zema enerģijas patēriņa šūnu bibliotēku izmantošanu. Terminālu ciparaparātūra darbojas ar 1,5 V piegādāto spriegumu un lieto līmeņa pārveidošanas shēmas, lai sazinātos ar ievad/izvad iekārtām, kas darbojas pie augstākiem spriegumiem.

**Klāt pie enerģijas patēriņa samazināšanas un caurlaidspējas prasību apmierināšanas, bezvadu darbības uzliek papildus ierobežojumu, proti, troksni. Mūsu pieeja, lai nodrošinātu robustumu pret kļūdām, ir pārraidīt ievad/izvad informāciju un minimālus “kontroles” datus caur sistēmas līmeni.**

# Reālā laika sistēmu ātrā prototipēšana: jauns izaicinājums?

## (Fast Prototyping of Real Time Systems: A New Challenge?)

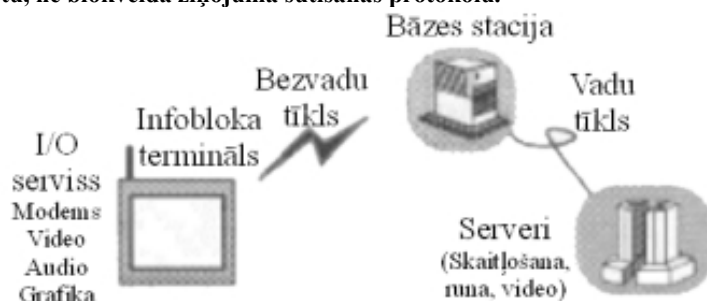
Tā kā elektroniskās sistēmas kļūst sarežģītākas un daudzfunkcionālākas, automatizētās projektēšanas sabiedrības uzmanība pamazām novirzās no mikroshēmu prototipēšanas problēmas un sāk fokusēties uz sistēmas projektēšanas tehnoloģiju attīstību. Viens gan var izbrīnīt, ar ko sistēmas prototipēšana atšķiras no tradicionālās mikroshēmu projektēšanas. Šajā rakstā mēs demonstrēsim, ka patiešām ir dažas fundamentālas atšķirības, kuras ietekmēs projektēšanas metodoloģiju gan specifikācijas un simulācijas, gan ieviešanas līmeņos. No otras puses, mēs arī pierādīsim, ka liela daļa no mikroshēmu projektēšanas metodoloģijas ir tikpat noderīgas arī sistēmas līmenī.

### Ievads

Kad diskutējam par sistēmu projektēšanu, prātā nāk viens jautājums: kas tad īsti tiek saprasts ar terminu sistēma un ar ko tas atšķiras, piemēram, no integrētas shēmas. Lai atbildētu uz šo jautājumu ir jāsaprot atšķirība starp sistēmu un tradicionālo integrēto shēmu prototipēšanu. Mēs esam pieņēmuši sekojošu sistēmas definīciju, kurai, kā kļūs redzams no pārējā šī raksta, ir pamatīga ietekme uz projektēšanas metodoloģiju un instrumentiem: *sistēma ir pašietverošs objekts (entīcija), kurš tiek izveidots no dažādiem apakškomponentiem ar heterogēnām īpašībām, kas savstarpēji komunicējas, lietojot dažādus protokolus.*

Šai definīcijai ir daži nozīmīgi trūkumi. Pirmkārt, nekas nav minēts par aparatūras platformu. Sistēma var aptvert tikai vienu integrēto shēmu, spiesto shēmu plati, karšu kopu vai vairākas kastes. Otrkārt, uzsvars sistēmu projektēšanā ir uz neviendabību. Tā savāc kopā lielu skaitu dažādu arhitektūru un komponentu, daži no tiem ir realizēti aparatūrā, daži programmatūrā, un tie lieto atšķirīgus komunicēšanās un sinhronizēšanās līdzekļus. Tas ir kontrastā tradicionālai mikroshēmu prototipēšanai, kura tiek balstīta uz vienas arhitektūras paraugu un lieto vienu komunicēšanās mehānismu, kas bieži ir sinhrons, statiski aprakstīts protokols.

1. zīmējums parāda kompleksas sistēmas piemēru. Tas demonstrē sistēmas pamatīpašības plašā veidā. Vispārējā skaitļošanas sistēma sastāv no daudzām komponentēm ar mainīgām arhitektūrām (skaitļošanas serveri, bāzes stacijas vai termināli), kuras savstarpēji komunicējas, lietojot dažādus protokolus (bezvadu vai vadu). Katrs komponents pats ir atkal sistēma. Piemēram, informācijas termināls sastāv no daudziem elementiem tādiem kā radio modems vai video dekompresēšanas iekārta. Šajā gadījumā iekārtas sarunājas viena ar otru, lietojot buferētu, ne blokveida ziņojuma sūtīšanas protokolu.



1. zīm. Infobloks: komunikācijas un skaitļošanas sistēma ar bezvadu pieeju.

Šī neviendabība tiks atspoguļota viscaur projektēšanas gaitai, kas tiks apskatīta nākamajās daļās. Mēs konsekventi apspriedīsim jautājumus, kas attiecas uz specifikāciju, simulāciju/emulāciju, projektēšanas sadalīšanu un īstenošanu.

### Objekta specifikācija/apraksts

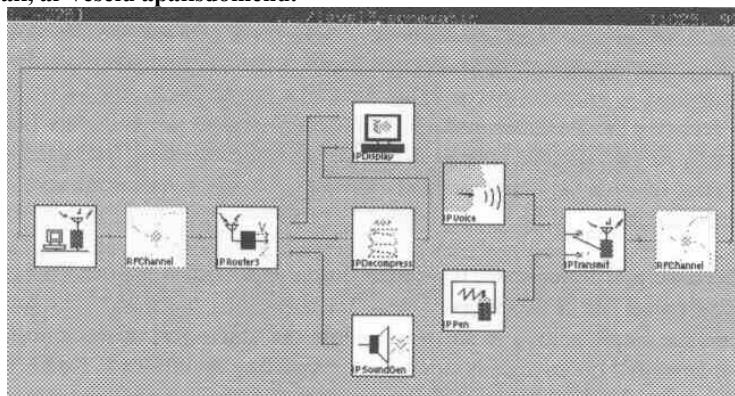
Kā jau minēts, tipiska sistēma sapludina daudzus objektus, kuru pamatsemantika ir ļoti atšķirīga. Turklāt šie moduļi tiecas viens ar otru sazināties, lietojot dažādus mehānismus. Uzskatīt, ka visi šie domēni ar to īpašībām var tikt adekvāti modelēti vienā vidē, ir pārāk cerīga domāšana. No VHDL pieredzes mēs esam iemācījušies, ka nozīmīga projektēšanas analīze un sintēze ir iespējama tikai tad, kad domēna semantika un tā interfeiss ar apkārtējo vidi tiek labi saprasti.

Tādēļ, mums ir ticība, ka heterogēnas specifikas vide ir svarīgākais kompleksu sistēmu aprakstīšanā. Tādā vidē katrs objekts tiek modelēts, lietojot atbilstošu domēnu, kurš atbilst tā pamatkonceptijai. Piemēram, vairums signālu apstrādes algoritmu tiek adekvāti modelēti, lietojot (statisko vai dinamisko) datu plūsmu. Galīgā automāta modelis ir ļoti piemērots globālam kontrolerim, savukārt, notikuma vadāmais modelis ir piemērots komunikācijas kanālam ar tā gadījumrakstura dabu. Sistēmas hierarhiskā daba, bez tā, vēl pieprasa, lai šie

domēni varētu tikt ievietoti cits citā. Piemēram, mūsu pieredze ir parādījusi, ka komunikēšanās procesu domēns bieži vien ir vislabāk piemērots sistēmas aprakstīšanai visaugstākajā līmenī. Sistēmas individuāli komponenti tiek modelēti kā vienlaicīgu procesu kopa, kas savstarpēji komunicējas, nododot ziņojumu. Katrs no komponentiem var tikt aprakstīts atbilstošā semantiskajā modelī (datu-plūsmas, secīgā koda, utt.) vai var būt kā vesela sistēma.

Hierarhiskas, heterogēnas sistēmas modelēšanas ideja formē Ptolemeja sistēmas pamatkonceptiju. Tā tiek ieviesta kā noslēgts objektorientēts karkass, kura iekšienē līdzās var pastāvēt un mijiedarboties dažādi skaitļošanas modeļi.

Piemēram, 2. zīmējums parāda infobloka sistēmas augsta līmeņa aprakstu Ptolemeja vidē. Tas modelē bāzes staciju, bezvadu kanālu un bezvadu terminālu kā procesu kopu. Atkarībā no abstrakcijas līmeņa, katrs process pats par sevi tiek modelēts dažādos veidos, vai nu kā abstrakts modelis, kas attēlo tikai ieejas/izejas uzvedību, vai, detalizētāk, ar veselu apakšdomēnu.



2. zīm. Infobloka komunikēšanās procesu modelis.

### Simulācija/Emulācija

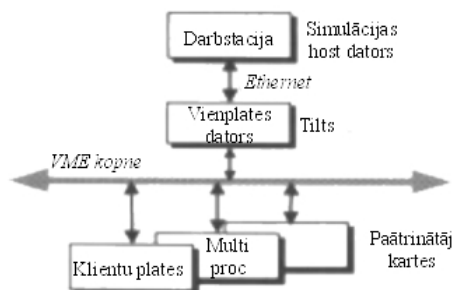
Heterogēnā modelēšana rezultātā noved pie ātrākas un efektīvākas simulācijas: tā kā katra apakšsistēma tiek definēta tās pašas domēnā ar labi saprotamu semantiku, var tikt lietots simulācijas dzinējs, kurš izmanto tā paša specifiskā domēna īpašības un izvairās no “kalpot visiem” simulācijas pieejas. Kā piemērs, mikroprocesora arhitektūras efektīva simulācija ir iespējama, pieņemot vienkārtēju vērtējumu takts cikla režīmā. Simulatori, kurš balstīts uz šo koncepciju ir 100 reizes ātrāks kā VHDL simulatori, kurš spējīgs izdarīt tieši to pašu, bet kuram jāpanes papildus slodze.

Komunikācijas semantikas skaidrai definīcijai, gan iekš, gan starp domēniem, ir vēl viena būtiska priekšrocība. Kompleksas sistēmas, tādas kā 1. zīmējumā, simulācija un analīze, prasa milzīgu daudzumu skaitļošanas ciklu, piemēram, lai noteiktu tīkla vājo vietu vai, lai analizētu kanāla kļūdu sekas. Bieži vien mēs vēlētos paātrināt simulāciju, sadalot to uz vairākām darbstacijām vai arī pārnesot uz faktisko aparatūru, kad tas ir iespējams. Piemēram, tā kā video serveris ir izpildīts uz eksistējoša RAID servera, video plūsma var tikt piegādāta tieši no tā servera pa tīklu, tā vietā, lai lietotu uz failu balstītu pieeju.

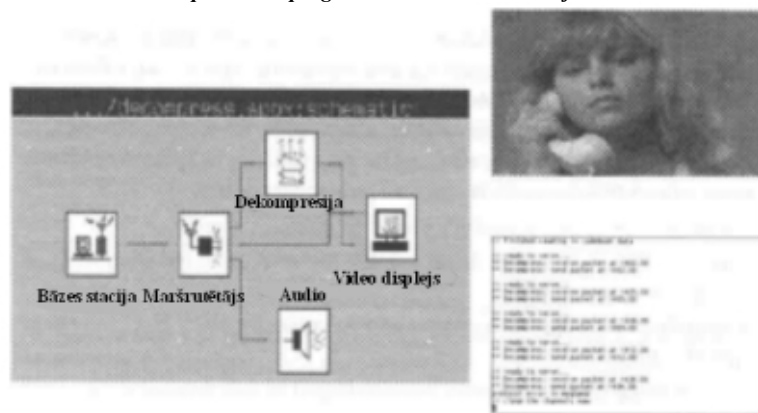
Tādu aparatūras-programmatūras līdzsimulāciju ir vieglāk panākt, kad komunikācijas mehānisms ir labi saprotams. Komunikāciju procesu domēnā, piemēram, sistēmas mezgls var tikt pārcelts no simulatora uz faktisko aparatūru, aizstājot mezgla simulācijas modeli ar procesu, kurš uzvedas tieši tāpat kā oriģinālais mezgls no ievad/izvad perspektīvas, bet kam nav nekādu citu iekšējo funkciju, kā tikai pārraidīt un uztvert ziņojumus uz/no paātrinātāj aparatūras.

Šī koncepcija tiek demonstrēta sekojošā piemērā, kurš analizē infobloka sistēmas bāzes stacijas-termināla saiti. Bāzes stacija, maršrutētājs un audioapakšsistēma tiek realizētas programmas ceļā, kamēr termināla video daļa tiek realizēta uz paātrinātāj kartes, kas galvenokārt sastāv no TMS320C30 (4. zīmējums). Saite starp host darbstaciju un paātrinātāj karti ir diezgan sarežģīta un darbojas no Ethernet tīkla uz VME-kopni caur tiltu, kas ir realizēts vienplātes mikroprocesorā (3. zīmējums). Ir vērts pieminēt, ka koda migrācija no programmatūras domēna uz aparatūras paātrinātāju ir pilnībā automātiska un visi nepieciešamie programmatūras procesi (host darbstacijā, tiltā un akseleratora atrašanās vietā) tiek sintezēti, izmantojot noteiktu komunikācijas protokolu un tā īpašības. 4. zīmējums rāda simulācijas modeli. Attālā procesa ģenerētie ziņojumi, kas kontrolē simulāciju, tiek parādīti atsevišķā logā.





3. zīm. Aparatūras-programmatūras līdzsimulācijas vide.



4. zīm. Infobloka aparatūras-programmatūras līdzsimulācija.

Aparatūras-programmatūras līdzsimulācijas pieejas rezultāts var uzlabot prototipēšanas projektu metodoloģiju. Šīs koncepcijas realizācijas lielākais kavēklis ir mūsdienu datortīklu ilgais latentums. Taču, lai kā, pašreiz tiek strādāts pie zema latentuma tīklu izveides.

#### Projekta sadalīšana

Viena no pamatproblēmām sistēmas projektēšanas procesā ir projekta sadalīšana apakškomponentos un atbilstošu arhitektūras un aparatūras platformu izvēle katram no tiem. Šis uzdevums bieži vien tiek uzskatīts par intuitīvu, taču patiesā projektēšanas pieredze ir parādījusi pretējo. Piemēram, infobloka projektēšanā, pārvietojot funkcionalitāti no termināla uz bāzes staciju vai serveriem, var samazināt enerģijas patēriņu roku (hand-held) termināliem, bet tas var arī novest pie caurlaidspējas problēmām vai samazinātas kvalitātes. Galvenā problēma ar sistēmas līmeņu sadalīšanu ir informācijas trūkums par to, kā augsta līmeņa lēmums ietekmēs projekta izmaksas un sistēmas veiktspēju. Lai izņemtu, vai konkrētajam modulim vajag būt realizētam programmas vai aparatūras ceļā, kādam ir vajadzīga aptuvena ideja attiecībā uz veiktspēju, cenu vai enerģijas patēriņu abām alternatīvām. Tā kā faktiskā arhitektūra varētu būt nezināma tajā laikā vai arī pilnīga projekta sintēze katrai alternatīvai var aizņemt pārāk daudz laika, ir vajadzīgs efektīvs paņēmieni, lai ģenerētu drošu darbības modeli modulim, ar uzdotu arhitektūras šablonu.

Kāds varētu cerēt, ka ar vienotas novērtēšanas vai veiktspējas analīzes instrumentu varētu būt pietiekami, lai aptvertu visu arhitektūras un algoritmisko telpu. Diemžēl tā nav. Tā kā novērtēšanas funkcija ir faktiskā sintēzes/kompilācijas procesa vienkāršots variants, tā mehāniski tiks ietekmēti ar arhitektūras dabu, lietojumprogrammu īpašībām un izmaksu faktoru, kurš tiks optimizēts. Kopumā var tikt identificētas četras instrumentu klases, kas analizē darbību:

- *Determinētās* novērtēšanas funkcijas, balstītas uz vienkāršotu sintēzes procesu.
- *Stohastiskās* novērtēšanas funkcijas, kuras iegūst statistisko izteiksmi, analizējot projektu piemēru lielu kopu.
- *Etalontestēšana*. Šī pieeja ir īpaši efektīva melnās kastes sintēzes/kompilācijas līdzekļiem.
- *Profilēšana* (uz simulāciju balstīta), lai paredzētu datu atkarības ietekmi.

Tas, kādu no klasēm lietot, ir atkarīgs no specifikācijas parauga, mērķarhitektūras un plānošanas procesa dabas. Publicētie katras klases rezultāti rāda, ka vidējā prognozējumu precizitāte 5-10 procentu robežās noteikti ir sasniedzama. Efektīvi ģenerētas veiktspējas modeļu pieejamība ir pirmais solis uz vai nu interaktīvu vai automātisku sistēmu sadalīšanas un projekta stila izvēles vidi.

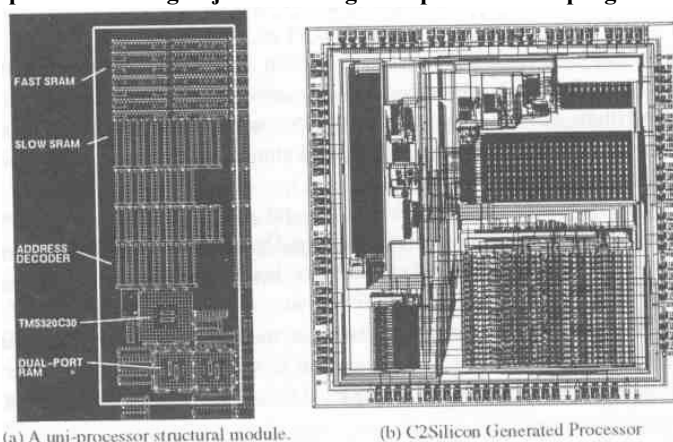
## Īstenošana

No īstenošanas perspektīvas mēs varam iedalīt komponentus divās klasēs: *Datu apstrādes moduļi*, kuri īsteno dažādu lietojumprogrammu specifiskus algoritmus un *Savstarpēji saistīti moduļi*, kuri īsteno komunikāciju protokolus, kas savieno datu apstrādes moduļus.

Kā redzams no iepriekšējās diskusijas, nekāds vienotas arhitektūras šablons neaptver visus iespējamus datu apstrādes moduļus. Tādēļ ir piedāvāts moduļu ģeneratoru miksejums. Šajā līmenī mēs varam izmantot projekta pieejas un metodoloģijas, ko lietoja individuālajā projektā: piemēram, datubāzes un projekta vadības instrumenti, ko lietoja priekš ASIC, tiek viegli adaptēti sistēmas līmenī. Projekta atkārtotas lietošanas un moduļprojektu koncepcijas, kas pirmām kārtām padarīja iespējamu sarežģīto ASIC projektu, ir tikpat labi paplašināmas.

Piemēram, parametrizētu aparatūras moduļu bibliotēka var tikt nodrošināta biežāk lietotajiem objektiem, tādiem kā mikroprocesoru vai signālprocesoru apakšsistēmas, atmiņas moduļi, datu iegūšanas moduļi un kopnes interfeisi. Parametri var būt procesoru skaits multiprocessoru modulī, atmiņas apjoms vai portu skaits makroelementa interfeisā. Bez savienojumu saraksta ģenerēšanas, moduļa ģenerators var arī definēt vietu, izsaukt tīkla karti vai MCM maršrutētājus un pat var sintezēt nedaudz no saistošās loģikas. SIERA sistēmā tiek lietota struktūras aprakstīšanas valoda (saukta par *sdl*) ar jaudīgām parametrizācijas iespējām, lai aprakstītu moduļa topoloģiju. Tāda moduļa piemērs ir parādīts 5a. zīmējumā, kurš attēlo vienprocesora modeli. SIERA moduļa bibliotēka satur tādus elementus kā VME slave modulis, statiskais RAM bloks un multiprocessoru apakšsistēma.

Citi moduļu ģeneratori sāk no uzvedības apraksta. Esošie ASIC sintēzes instrumenti, tādi kā C2Silicon un HYPER ir viegli pieejami kā sistēmas līmeņa izveides bloku ģeneratori. Moduļa piemērs, kas īsteno robota kontroleri un ir izveidots, lietojot C2Silicon un LAGER sistēmas, ir parādīts 5b. zīmējumā. Šie bloki viegli iekļaujas globālajā sistēmā, ja ir pieejams melnās kastes modelis, kas specificē ieejas/izejas interfeisu. Kāds var iet pat soli tālāk: kombinējot programmas kompilatoru ar vēl neesošā procesora plati vai parametrizēto procesora moduli, rezultātā dodot “*programmas*” moduļa ģeneratoru. Tas demonstrē, ka nav konceptuālas atšķirības starp aparatūru un programmatūru no sistēmas perspektīvas. Tādi jēdzieni kā modularitāte un atkārtota lietojamība ir pamatoti abos gadījumos. Atslēga uz aparatūras un programmatūras integrāciju ir



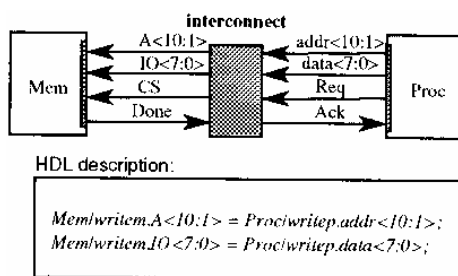
skaidri saprotama no interfeisa semantikas.

5. zīm. Sistēmas līmeņa moduļa ģenerators: (a) – vienprocesora strukturāls modulis, (b) – C2Silicon ģenerēts procesors.

Pēdējam būs noteikta ietekme uz savstarpēji saistīto moduļu dabu. Atkarībā no izvēlēta protokola un aparatūras platformas, savstarpēji saistītais modulis var atšķirties no vadiem, saistītās loģikas kompozīcijas, kompleksās aparatūras moduļa, kas satur bufera elementus un sarežģītu kontroli vai vienkāršu programmatūras daļu, kas īsteno semaforu un darbojas procesora modulī. Tādējādi saistītais ģenerators var mainīties no ļoti vienkārša līdz sarežģītam. Piemēram, tipiskā integrētā shēma komunikācijas protokols ir netiešs un balstīts uz frekvences vadīta, statistiski saplānota notikuma sakārtošanu. Ziņojuma sūtīšanas protokols var tikt viegli pārklāts pār VME slave moduli, kas iepriekš tika minēts un arī īstenots kā RPC signāls. Notikuma vadības protokoli pieprasa asinhrona kontroles moduļa izveidi. Šie protokoli var būt diezgan sarežģīti, kā tas ir redzams 6. zīmējumā, kurš parāda, kā protokols pārvalda rakstīšanu no procesora uz atmiņas moduli.

## Kopsavilkums

Centrālā koncepcija sistēmu prototipēšanā ir heterogenitāte, kas tiek atspoguļota visos projektēšanas procesa līmeņos. Katras apakšsistēmas semantikas un attiecību ar vidi skaidra saprašana rezultātā dod modulāru, hierarhisku projektēšanas procesu, kas nav nemaz tik atšķirīgs no vispārtzītās mikroshēmu prototipēšanas metodoloģijas.



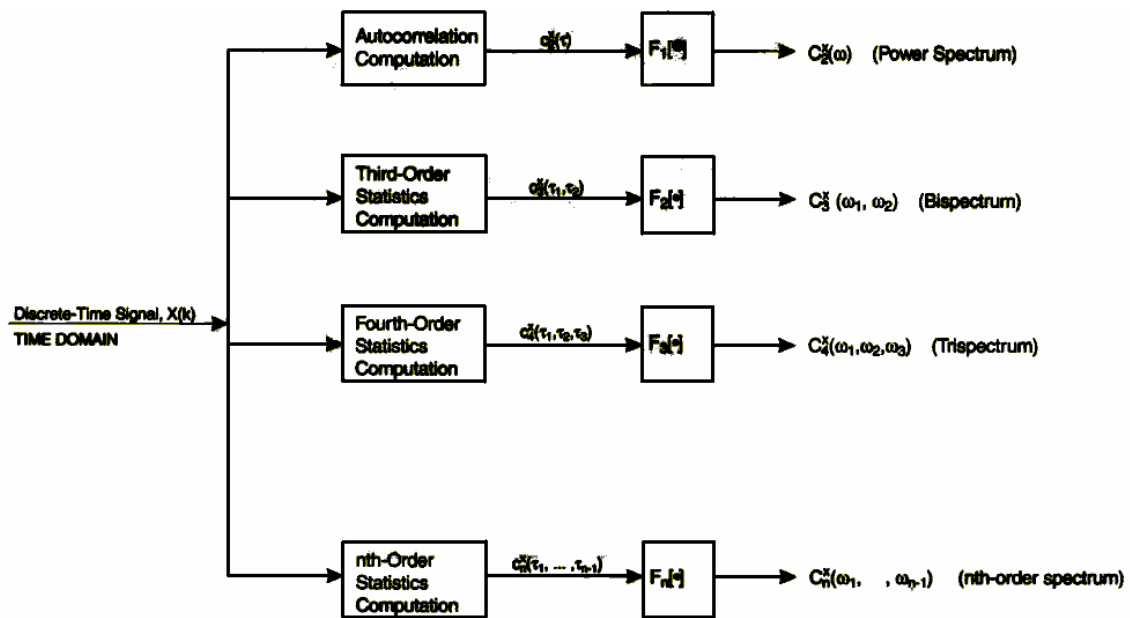
6. zīm. Interfeisa procesors – atmiņa (rakstīšanas darbība) protokola specifikācija.

## Signālu apstrāde ar augstākas kārtas spektru (Signal Processing with Higher-Order Spectra)

Diskrēta laika determinētu vai stohastisku signālu enerģijas spektrālā blīvuma vai vienkārši enerģijas spektra novērtējums ir noderīgs līdzeklis ciparsignālu apstrādē jau vairāk kā trīsdesmit gadus. Enerģijas spektra novērtējuma paņēmieni ir pamatā uzlabotu radaru, sonaru, sakaru, runas, biomedicīnas, ģeofizikas un citu datu apstrādes sistēmu radīšanā. Pieejamie enerģijas spektra novērtēšanas paņēmieni var tikt apskatīti atsevišķu klašu rindā, proti, izmantojot vispārpieņemtās (vai “Furjē tipa”) metodes, Kapona maksimālās varbūtības metodi ar tās modifikācijām, maksimālās entropijas, minimālās enerģijas un minimālās šķērs-entropijas metodes, kā arī metodes, kas balstītas uz autoregresīviem (AR), vidējās kustības (MA) un ARMA modeļiem; un harmoniskās sadalīšanas metodes tādas kā Pronija, Pisarenko, MUSIC, ESPRIT. Notikumi šajā laukā ir noveduši līdz signālu modelēšanai un līdz multidimensionālu, multikanālu un masīvu apstrādes problēmām. Katram no iepriekšminētajiem paņēmieniem ir noteiktas priekšrocības un ierobežojumi ne tikai pēc darbības novērtējuma, bet arī pēc skaitļošanas sarežģītības, un, tādēļ, atkarībā no signāla vides, ir jāizvēlas vispiemērotākais no paņēmieniem.

Enerģijas spektra novērtēšanā attiecīgais signāls tiek apstrādāts tādā veidā, ka tiek aprēķināts enerģijas sadalījums starp tā frekvenču komponentēm. Tāpat, fāžu attiecības starp frekvenču komponentēm tiek slāpētas. Informācija, kas ietverta enerģijas spektrā, būtībā ir tā, kura ir autokorelācijā; tas būtu pietiekami pilnīgam Gausa signāla statistiskam aprakstam. Tomēr ir praktiskas situācijas, kad mums ir jāskatās aiz signāla enerģijas spektra (autokorelācijas), lai iegūtu informāciju attiecībā uz novirzēm no Gaussianity un fāzes saistību klātbūtni.

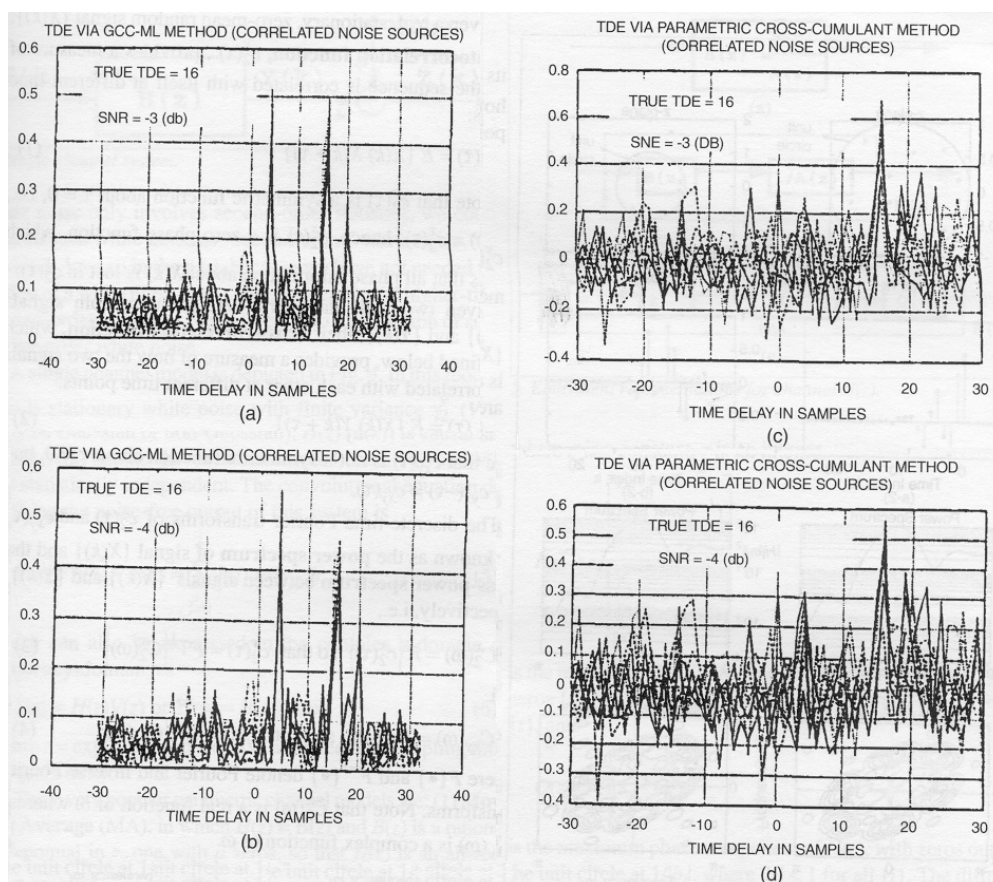
Augstākās kārtas spektrs (zināms arī kā polispektrs), kas ir definēts kā signāla augstāko kārtu statistisks skaitļojums (“kumulants”), satur tādu informāciju. Augstākās kārtas spektra īpašie gadījumi ir trešās kārtas spektrs, saukts arī par bispektru, kas pēc definīcijas ir trešās kārtas statistisko skaitļojumu Furjē transformācija, un trispektrs (ceturtās kārtas spektrs), kas ir stacionāra signāla ceturtās kārtas statistisko skaitļojumu Furjē transformācija. Enerģijas spektrs faktiski ir augstākās kārtas spektra klases loceklis, t.i., tas ir otrās kārtas spektrs. 1. zīmējums parāda uzdotā diskrēta laika signāla augstākās kārtas spektra klasifikācijas plānu. Kaut arī signāla augstākās kārtas statistiskie skaitļojumi un spektrs var tikt definēti momentu un kumulantu veidā, momenta un momenta spektrs var būt ļoti noderīgs determinētu signālu analīzē (īslaicīgu un periodisku), turpretim, kumulants un kumulants spektrs ir ļoti svarīgs stohastisku signālu analīzē.



1. zīm. Diskrēta signāla  $X(k)$  augstākās kārtas spektra klasifikācijas plāns.  $F[\cdot]$  apzīmē  $n$ -dimensionālu Furjē transformāciju.

Ir vairākas motivācijas tam, ka signālu apstrādē tiek lietots augstākās kārtas spektrs. Tās iekļauj paņēmienus, lai: (1) slāpētu papildus Gausa trokšņus no nezināma enerģijas spektra; (2) identificētu neminimuma fāzes sistēmas vai atjaunotu neminimuma fāzes signālus; (3) iegūtu informāciju, kas radusies no novirzēm no Gaussianity; un (4) signālos atklātu un raksturotu nelineāras īpašības, kā arī identificētu nelineāras sistēmas.

Pirmā motivācija ir balstīta uz īpašību, ka tikai Gausa signāliem viss kumulanta spektrs ar kārtu, kas lielāka par divi, ir vienāds ar nulli. Ja ne-Gausa signāls tiek uzvertts kopā ar Gausa troksni, transformācija augstākās kārtas kumulantā domēnā likvidēs troksni. Šādai signālu apstrādei noteikti būs priekšrocības, lai atklātu vai/un novērtētu signāla parametrus no novēroto datu kumulantā spektra. Dažos gadījumos kumulants spektrs var kļūt par domēniem ar augstu signāls-troksnis attiecību (SNR), kuros var veikt uztveršanu, parametru novērtēšanu vai pat visa signāla atjaunošanu. 2. zīmējums attēlo laika aizkaves parametru novērtējuma rezultātus, kas iegūti ar šķērs-korelācijas (t.i., ar otrās kārtas statisko skaitļošanu) metodi un paņēmieni, kas balstīts uz trešās kārtas kumulantiem. Tiek pieņemts, ka interesējošais signāls nav Gausa, turpretim, papildus troksnis ir Gausa un telpiski korelēts. No 2. zīmējuma ir skaidri redzams, ka trešās kārtas kumulanti slāpē Gausa trokšņa efektu un tādā veidā nodrošina labāku laika aizkavju kalkulāciju, sevišķi pie zemiem SNR.



2. zīm. Laika aizkaves kalkulācija telpiski korelēta Gausa trokšņa klātbūtnē. Ir pieņemts, ka  $X(k)$  un  $Y(k)$  ir divas pieejamās sensoru mērsistēmas, kas apmierina  $X(k)=S(k)+W_1(k)$  un  $Y(k)=AS(k-D)+W_2(k)$ , kur  $S(k)$  ir nezināms ne Gausa signāls,  $D$  ir laika aizkave un  $W_1(k)$ ,  $W_2(k)$  ir telpiski korelēti Gausa trokšņi. Problēma ir aprēķināt  $D$ . Šķērskorelācijas metode, kas balstīta uz maksimālās varbūtības logu (ML), aprēķina  $S(k)$  autokorelāciju, kura sasniedz maksimālo vērtību pie  $D(D=16)$ , un telpisko korelāciju starp  $W_1(k)$  un  $W_2(k)$ , kā parādīts (a) un (b). No otras puses šķērskumulantā metode slāpē trokšņa efektu un tādā veidā padara  $D$  aprēķināšanu drošāku, kā redzams (c) un (d).

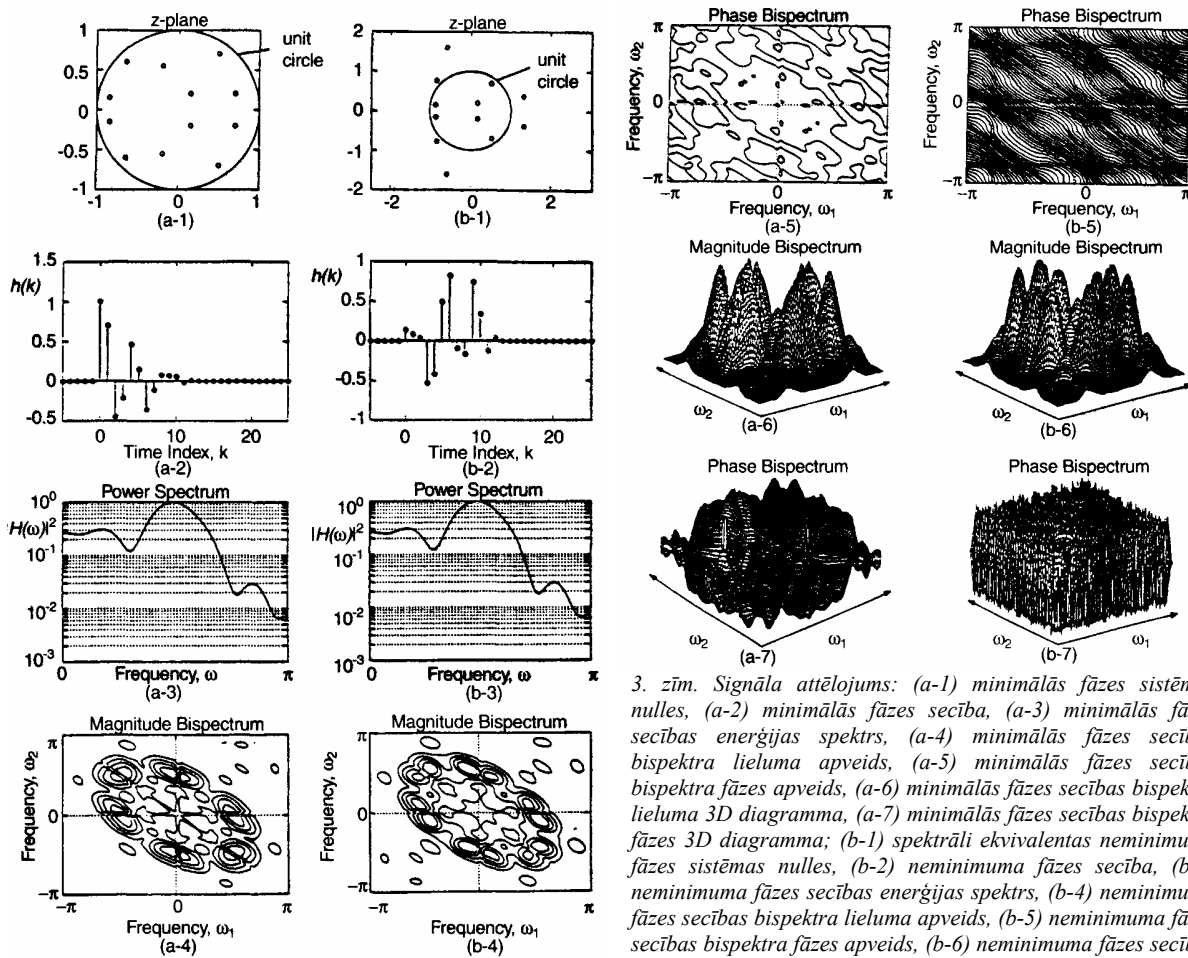
Otrā motivācija ir balstīta uz faktu, ka polispektrs (kumulantais un moment) saglabā signālu patieso fāzes raksturu. Laika sērijas datu modelēšanai signālu apstrādē otrās kārtās statistiskie aprēķini ir gandrīz vienīgi, kas tiek lietoti, tāpēc, ka tie parasti ir vismazāko laukumu optimizācijas kritērija rezultāts. Tomēr autokorelācijas domēns slāpē fāzes informāciju. Precīza fāzes atjaunošana autokorelācijas (vai enerģijas spektra) domēnā var tikt sasniegta tikai, ja signālam ir minimuma fāze. No otras puses, ne minimuma fāzes signāla atjaunošana vai sistēmas identifikācija var tikt sasniegta augstākas kārtas spektra domēnos polispektra spējas saglabāt gan lielumu, gan ne minimuma fāzes informāciju rezultātā. 3. zīmējums parāda divus dažādus signālus, kuriem ir identiska autokorelācija, bet atšķirīgi trešās kārtas statistiskie aprēķini. Tātad šiem diviem signāliem ir identisks enerģijas spektrs un atšķirīgs bispektrs.

Trešā motivācija ir balstīta uz novērojumu, ka vairums “reālās pasaules” signālu ir ne Gausa un tādā veidā tiem nav augstākās kārtas spektra. Kā rāda 1. zīmējums, ne Gausa signāls var tikt sadalīts tā augstāko kārtu spektrālās funkcijās, kur katra no tām var ietvert dažādu informāciju par signāliem. Tas var būt ļoti noderīgi signālu klasifikācijā, kur noteiktas klasifikācijas pazīmes var tikt iegūtas no augstākās kārtas spektra domēniem.

Galū galā, augstākās kārtas spektra priekšastādīšana ir pilnīgi dabiska, kad mēs mēģinām analizēt sistēmas nelinearitāti, kas darbojas ar nejausu ieejas signālu. Kopējās attiecības stacionāri nejausai datu nodošanai caur lineārām sistēmām ir studētas diezgan plaši jau daudzus gadus. Principā, vairums šo attiecību tiek balstītas uz enerģijas spektram (autokorelācija) piekapiotiem kritērijiem. No otras puses, kopējās attiecības nav pieejamas stacionāri nejausai datu nodošanai caur lineārām sistēmām. Tā vietā, katrs nelinearitātes veids tiek pētīts kā speciāls gadījums. Polispektrs var spēlēt galveno lomu sistēmas nelinearitātes tipa atklāšanā un raksturošanā no tās izejas datiem. Dažādas signālu apstrādes metodes ir izstrādātas nelinearitāšu atklāšanai un raksturošanai laika sērijās, lietojot augstākās kārtas spektru.

Šī raksta organizācija ir sekojoša. Vispirms mēs apspriežam uz korelāciju balstītu signālu apstrādes metožu priekšrocības un ierobežojumus. Tad mēs sniedzam definīcijas, īpašības un aprēķinus augstākās kārtas statistikai un spektram ar uzsvaru uz bispektru un trispektru. Pēc tam mēs aprakstam lineāru procesu polispektra parametriskās un neparametriskās izteiksmes. Nākamā daļa

attiecas uz nelineāru procesu polispektru. Rakstu mēs nobeidzam ar diskusiju par augstākas kārtas spektra lietojumprogrammām signālu apstrādē.



3. zīm. Signāla attēlojums: (a-1) minimālās fāzes sistēmas nulles, (a-2) minimālās fāzes secība, (a-3) minimālās fāzes secības enerģijas spektrs, (a-4) minimālās fāzes secības bispektra lieluma apveids, (a-5) minimālās fāzes secības bispektra fāzes apveids, (a-6) minimālās fāzes secības bispektra lieluma 3D diagramma, (a-7) minimālās fāzes secības bispektra fāzes 3D diagramma; (b-1) spektrāli ekvivalentas neminimuma fāzes sistēmas nulles, (b-2) neminimuma fāzes secība, (b-3) neminimuma fāzes secības enerģijas spektrs, (b-4) neminimuma fāzes secības bispektra lieluma apveids, (b-5) neminimuma fāzes secības bispektra fāzes apveids, (b-6) neminimuma fāzes secības bispektra lieluma 3D diagramma, (b-7) neminimuma fāzes secības bispektra fāzes 3D diagramma.

### Uz korelāciju balstīta signālu apstrāde: priekšrocības un ierobežojumi

Šajā daļā mēs apskatām nozīmīgus rezultātus no uz korelāciju balstīta signālu apstrādes, lai uzsvērtu tās priekšrocības un ierobežojumus. Visi mūsu rezultāti ir diskrēta laika nejaušam signālam tāpēc, ka mēs esam ieinteresēti ciparsignālu apstrādes lietojumprogrammās un tiek pieņemts, ka dati, ar kuriem mēs strādājam, ir semplēti (ir veiktas nolases).

Uzdotot reālu, stacionāru, nulles vidus nejaušu signālu  $\{X(k)\}$ , tā **autokorelācijas funkcija**,  $c_x^2(\tau)$ , nodrošina paša signāla secības korelācijas dažādu laika punktu mērīšanu:

$$c_x^2(\tau) = E \{X(k) X(k + \tau)\} \quad (1)$$

Ievērojiet, ka  $c_x^2(\tau)$  ir simetriska funkcija pie  $\tau = 0$ , t.i.,  $c_x^2(-\tau) = c_x^2(\tau)$ ; tādējādi,  $c_x^2(\tau)$  ir nulles fāzes funkcija, kas nozīmē, ka visa fāzes informācija par  $X(k)$  ir pazaudēta iekš  $c_x^2(\tau)$ .

Uzdotot divus reālus, stacionārus, nulles vidus nejaušus signālus  $\{X(k)\}$  un  $\{Y(k)\}$ , to **šķērskorelācijas funkcija**, kura tiek definēta zemāk, nodrošina divu signālu savstarpējās korelācijas dažādos laika punktos mērīšanu:

$$c_{xy}(\tau) = E \{X(k) Y(k + \tau)\} \quad (2)$$

Ievērojiet, ka  $c_{xy}(\tau)$  nav simetriska funkcija pie  $\tau = 0$ , bet, ka  $c_{yx}(-\tau) = c_{xy}(\tau)$ .

Diskrēta laika  $c_x^2(\tau)$  un  $c_{xy}(\tau)$  Furjē transformācijas ir zināmas kā signāla  $\{X(k)\}$  **enerģijas spektrs** un šķērsenerģijas spektrs starp signāliem  $\{X(k)\}$  un  $\{Y(k)\}$ , attiecīgi, t.i.,

$$C_x^2(\omega) = F\{c_x^2(\tau)\} \text{ tā, ka } c_x^2(\tau) = F^{-1}\{C_x^2(\omega)\} \quad (3)$$

un

$$C_{xy}(\omega) = F\{c_{xy}(\tau)\} \text{ tā, ka } c_{xy}(\tau) = F^{-1}\{C_{xy}(\omega)\} \quad (4)$$

kur  $F\{\cdot\}$  un  $F^{-1}\{\cdot\}$  apzīmē Furjē un inverso Furjē transformācijas. Ievērojiet, ka  $C_x(\omega)$  ir reāla funkcija no  $\omega$ , turpretim  $C_{xy}(\omega)$  ir kompleksa funkcija no  $\omega$ .

Ja diskrētām nejaušām signālam nenotiek korelācija no viena punkta uz nākamo, t.i.,  $E\{X(i)X(j)\} = 0$  visiem  $i \neq j$ , tad mēs attiecamies pret šo signālu kā pret (diskrēta laika) balto troksni. Baltajam troksnim ir  $\gamma_x^2$  visām  $\omega$  vērtībām; šajā gadījumā mēs apvienojam balto troksni ar signālu, kura enerģijas spektrs ir "līdzens" visām frekvencēm. Dēļ tā, ka parasti lietotais baltā trokšņa apraksts ietver tikai otrās kārtas statistiskos aprēķinus, mēs varētu attiekties pret tādu balto troksni kā "otrās kārtas" balto troksni. Mēs tā neesam darījuši tāpēc, ka mums nebija vajadzīgs baltā trokšņa priekšstats, kas tiek raksturots ar tā augstākas kā otrā kārtas statistiskajiem aprēķiniem. Mēs atgriezīsimies pie diskusijas par tādu augstāku kārtu balto troksni.

Viena kanāla modelis tiek attēlots 4. zīmējumā. Šajā modeli  $V(k)$  ir stacionārs baltais troksnis ar ierobežotu dispersiju  $\gamma_v^2$  ( $V(k)$  var būt Gausa vai ne Gausa);  $H(z)$  [h(k)] ir cēlonisks (kauzāls) un stabils;  $N(k)$  ir Gausa (balts vai krāsains); un  $V(k)$  un  $N(k)$  ir statistiski neatkarīgi. Vienādojums, kas apraksta šīs sistēmas beztrokšņa izeju, ir

$$Y(k) = h(k) * V(k) = \sum_{i=0}^k h(i) V(k-i) \quad (5)$$

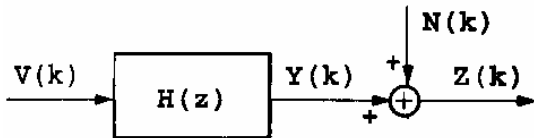
kurš arī var būt izteikts kompleksajā  $z$  domēnā vai frekvences domēnā, kā

$$Y(z) = H(z)V(z) \text{ or } Y(\omega) = H(\omega)V(\omega) \quad (6)$$

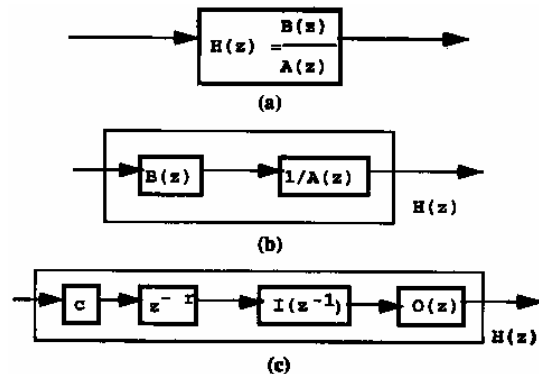
kur  $z = \exp(j\omega T)$ , un mēs esam pieņēmuši, ka kvantēšanas periods  $T$  ir vienāds ar vienu vienību.

Ir trīs ļoti populāri parametrisko kanālu modeļi: (1) Vidējās kustības (MA – moving average), kurā  $H(z)=B(z)$  un  $B(z)$  ir racionāls polinoms  $z$ -tā, tas ar  $q$  nullēm, tā kā  $H(z)$  ir visu nulļu modelis; (2) Autoregresīvais (AR), kurā  $H(z)=1/A(z)$  un  $A(z)$  ir racionālais polinoms  $z$ -tā, tas ar  $p$  nullēm, tā kā  $H(z)$  ir visu polu modelis; un (3) Autoregresīvais – vidējās kustības (ARMA), kurā  $H(z)=B(z)/A(z)$ , kur  $A(z)$  un  $B(z)$  ir jau attiecīgi aprakstīti pie AR un MA modeļiem; tādējādi ARMA modelim ir  $q$  nulles un  $p$  poli. MA modelim ir galīga impulsa reakcija (FIR), turpretim AR un ARMA modeļiem ir bezgalīga impulsu reakcijas (IIR).

Kad visas  $H(z)$  nulles atrodas kompleksās  $z$  plaknes vienības riņķa iekšpusē, tad saka, ka ir  $H(z)$  minimālā fāze. Kad visas  $H(z)$  nulles atrodas ārpus vienības riņķa, tad saka, ka  $H(z)$  ir maksimālā fāze. Kad dažas no  $H(z)$ -ta nullēm atrodas vienības riņķa iekšpusē, dažas ārpusē, tad saka, ka  $H(z)$  ir sajaukta fāze vai neminimuma fāze. Daudzas reālās pasaules sistēmas (kanāli) ir neminimuma fāzē.



4. zīm. Vienkanāla sistēma.



5. zīm. Ekvivalentie attēlojumi/vai kanāls H(z).

Sistēmas linearitātes dēļ, 4. zīmējumā ir iespējami dažādi ARMA modeļa,  $H(z)=B(z)/A(z)$ , attēlojumi, ieskaitot tos, kas attēloti 5. zīmējumā. 5a gadījums apskata ARMA modeli kā modeļu AR un MA kaskādi; 5c ir sekojoši balsītis uz neminimuma fāzes sistēmu tās polu un nulļu veidā:

$$H(z) = B(z)/A(z) = c z^{-r} I(z^{-1}) O(z) \quad (7)$$

kur  $c$  ir konstante,  $r$  ir vesels skaitlis,

$$I(z^{-1}) = \frac{\prod_{i=1}^{L_1} (1 - a_i z^{-1})}{\prod_{i=1}^{L_2} (1 - c_i z^{-1})} \quad (8)$$

ir  $H(z)$  minimālās fāzes komponente, ar poliem  $\{c_i\}$  un nullēm  $\{a_i\}$  vienības riņķa iekšpusē, t.i.,  $|c_i| < 1$  un  $|a_i| < 1$  visiem  $\{i\}$ , un

$$O(z) = \prod_{i=1}^{L_2} (1 - b_i z) \quad (9)$$

ir  $H(z)$  maksimālās fāzes komponente, ar nullēm ārpus vienības riņķa pie  $1/|b_i|$ , kur  $|b_i| < 1$  visiem  $\{i\}$ . Katrs no dažādajiem  $H(z)$  attēlojumiem ir noderīgs savā veidā.

Kompleksais kepstrs ir plaši pazīstams ciparsignālu apstrādē. Lai iegūtu komplekso kepstru  $h(k)$ , jāsāk ar pārvietošanas funkciju  $H(z)$ , jāiegūst logaritms  $H(z) = \ln H(z)$  un tad jāņem  $H(z)$  inverso  $z$  transformāciju.

Kad  $H(z)$  ir sadalīta, kā tas ir 7. vienādojumā, tās minimuma fāzes un maksimuma fāzes komponentēs, t.i., attiecīgi  $i(k)$  un  $o(k)$ , tad

$$h(k) = c \cdot i(k) * o(k) * \delta(k - r). \quad (10)$$

Ir labi zināms, ka  $i(k)$  un  $o(k)$  var tikt rekursīvi aprēķināti  $h(k)$  izteiksmē, lietojot sekojošas formulas:

$$i(k) = \begin{cases} 0, & k < 0 \\ \exp(\hat{h}(0)), & k = 0 \\ \frac{1}{k} \left[ \sum_{j=1}^k A^{(j)} i(k-j) \right] & k > 0 \end{cases} \quad (11)$$

un

$$o(k) = \begin{cases} 0, & k > 0 \\ 1, & k = 0 \\ \frac{1}{k} \left[ \sum_{j=k}^k B^{(-j)} o(k-j) \right] & k < 0 \end{cases} \quad (12)$$

Kurā "A" un "B" koeficienti kepstra koeficienti ir saistīti ar minimuma aizkaves un maksimuma aizkaves nullēm, t.i., attiecīgi,

$$A^{(i)} = \sum_{j=1}^{L_1} a_j^i - \sum_{j=1}^{L_2} c_j^i \quad (13)$$

un

$$B^{(i)} = \sum_{j=1}^{L_2} b_j^i \quad (14)$$

Ja kepstra koeficientus var izrēķināt, tad  $i(k)$  un  $o(k)$  var tikt atjaunoti, attiecīgi, no 11. un 12. vienādojuma, pēc kuriem var tikt atjaunots  $h(k)$  no 10. vienādojuma.

Tāpat ir labi zināms, ka

$$c_2^y(\tau) = c_2^y(\tau) + c_y^N(\tau) = \gamma_2^y \sum_{i=0}^{\infty} h(i) h(i + \tau) + c \quad (15)$$

$$C_2^z(z) = \gamma_2^y H(z) H(z^{-1}) + C_2^N(z) \quad (16a)$$

vai

$$C_2^z(\omega) = \gamma_2^y |H(\omega)|^2 + C_2^N(\omega), \quad (16b)$$

un

$$c_{vz}(\tau) = \gamma_2^y h(\tau). \quad (17)$$

No 16. vienādojuma mēs redzam, ka visa fāzes informācija par  $H(\omega)$  ir zaudēta spektrā (vai autokorelācijā); tādējādi mēs sakām, ka korelācija vai spektrs ir fāzes akli. Piebildīsim, no 17. vienādojuma, ka, ja mums ir pieeja gan sistēmas ieejas, gan izejas signāliem, tad mēs varam atjaunot pareizo fāzi IR,  $h(k)$ . Daudzās svarīgās signālu



apstrādes lietojumprogrammās mums ir pieejā tikai sistēmas izejas signālam; tādējādi, mēs nevaram lietot 17. vienādojumu šajās lietojumprogrammās.

Pieņemsim, ka  $H(z)$ -tam ir nulles gan vienības riņķa iekšpusē, gan ārpusē. Kad šīs nulles tiek atspoguļotas to papildvietās [t.i., (dažas vai visas no) tām, kas ir vienības riņķa iekšpusē, tiek atspoguļotas vienības riņķa ārpusē,

$$\gamma_2^N H_{MP}(z) H_{MP}(z^{-1}) + C_2^N(z) = \gamma_2^N H(z) H(z^{-1}) + C_2^N(z).$$

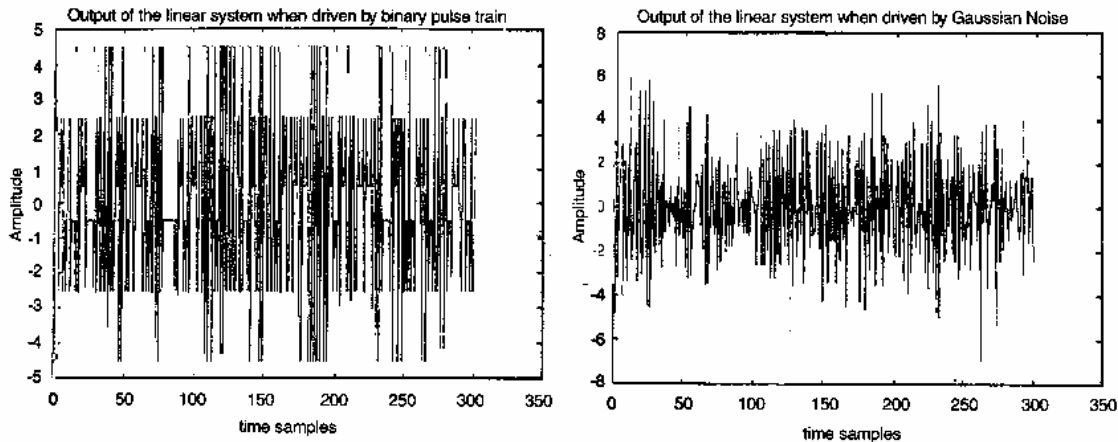
un (dažas vai visas no) tām, kas ir vienības riņķa ārpusē, tiek atspoguļotas vienības riņķa iekšpusē], mēs redzam, ka enerģijas spektrs paliek nemainīgs, piemēram,  $C_2(z^{-1}) = C_2(z)$ . Kad visas no  $H(z)$ -ta nullēm, kas ir ārpus vienības riņķa, tiek atspoguļotas vienības riņķa iekšpusē tā, ka rezultējošā pārvietošanas funkcija ir minimuma fāze, t.i.,  $H(z) \rightarrow H_{MP}(z)$ , mēs atkal iegūstam, ka

Minimālās fāzes pārvietošanas funkcija  $H_{MP}(z)$  tiek saukta par “spektrāli ekvivalentu minimālo fāzi” (SEMP) ekvivalentu  $H(z)$ . 1. tabula dod triju MA sistēmu vienkāršu piemēru, kurām ir vienādas autokorelācijas un spektrs. Tā ilustrē to, ka ir neiespējami atjaunot pareizo fāzes modeli tikai no autokorelācijas vai enerģijas spektra informācijas.

1. tabula. Minimuma, maksimuma un jauktās fāzes sistēmas – ar identisku enerģijas spektru (vai autokorelācijām):  $0 < a < 1$ ,  $0 < b < 1$ .

Sistēma	Minimālā fāze	Maksimālā fāze	Jaukta fāze
$H(z)$	$(1 - az^{-1})(1 - bz^{-1})$	$(1 - az)(1 - bz)$	$(1 - az)(1 - bz^{-1})$
Autokorelācijas			
$c_2(0)$	$1 + a^2b^2 + (a + b)^2$		
$c_2(1)$	$-(a + b)(1 + ab)$		
$c_2(2)$	$ab$		

6. zīmējums attēlo divas laika sērijas, kurām ir viens un tas pats enerģijas spektrs. Viena ir Gausa un otra ir ne Gausa. Augšējā laika sērija bija iegūta iedarbojoties neminimuma fāzes MA sistēmai  $H(z) = (1 - 1/2z^{-1})(1 - 2z^{-1})$  ar  $a = \pm 1$  bināro impulsu rindu (vienības dispersiju, un – 2 ceturtais kārtas kumulantiem), turpretim apakšējā laika sērija bija iegūta iedarbojoties uz to pašu sistēmu ar Gausa secību (vienības dispersiju). Piebildīsim, ka šīs laika sērijas izskatās diezgan atšķirīgas. Ja mēs lietojam tikai korelācijas informāciju ar tās palīgmodeli SEMP, tad no datiem mēs iegūsim nepareizu informāciju, kad faktiskais modelis ir neminimuma fāze. Nākamajā daļā mēs parādīsim, ka varam iegūt no datiem pareizu informāciju, kad mēs strādājam ar augstākas kārtas statistiskiem aprēķiniem.



6. zīm. Laika sērijas ar identisku enerģijas spektru. Viens ir Gausa; otrs ir ne Gausa.

Augstākas kārtas statistiskie aprēķini un spektrs

a. Definīcijas un īpašības

Šajā daļā mēs sniedzam definīcijas, īpašības un augstākās kārtas statistikas aprēķinus, t.i., momentus un kumulantus, un to atbilstošos augstākās kārtas spektrus. Apraksts ir dots gan stohastiskiem, gan determinētiem signāliem. Lai kā, taču raksta uzsvars ir likts uz otrās, trešās un ceturtais kārtas statistiskajiem aprēķiniem un tiem atbilstošajām Furjē transformācijām: enerģijas spektra, bispektra un trispektra.

Ja  $\{X(k)\}$ ,  $k = 0, \pm 1, \pm 2, \pm 3, \dots$  ir reāls, stacionārs diskrēta laika signāls un tam eksistē momenti līdz  $n$ , tad

$$m_n^x(\tau_1, \tau_2, \dots, \tau_{n-1}) \triangleq \tag{18}$$

$$E \{X(k) X(k + \tau_1) \dots X(k + \tau_{n-1})\}$$

attēlo n-tās kārtas stacionāra signāla momenta funkciju, kura ir atkarīga tikai no laika starpības  $\tau_1, \tau_2, \tau_{n-1}, \tau = 0 \pm 1, \dots$  visiem  $i$ . Skaidrs, ka otrās kārtas momenta funkcija  $m^x_2(\tau_1)$  ir autokorelācija no  $\{X(k)\}$ , turpretim  $m^x_3(\tau_1, \tau_2)$  un  $m^x_4(\tau_1, \tau_2, \tau_3)$  ir, attiecīgi, trešās un ceturrtās kārtas momenti.  $E\{\bullet\}$  apzīmē statistisko gaidīšanu.

n-tās kārtas ne Gausa stacionāra, nejauša signāla  $X(k)$  kumulanta funkcija var tikt uzrakstīta (tikai priekš  $n=3, 4$ ) kā:

$$c_n^x(\tau_1, \tau_2, \dots, \tau_{n-1}) = m_n^x(\tau_1, \tau_2, \dots, \tau_{n-1}) - m_n^G(\tau_1, \tau_2, \dots, \tau_{n-1}) \quad (19)$$

kur  $m_n^x(\tau_1, \dots, \tau_{n-1})$  ir n-tās kārtas  $X(k)$  momenta funkcija un  $m_n^G(\tau_1, \tau_2, \dots, \tau_{n-1})$  ir n-tās kārtas ekvivalentā Gausa signāla momenta funkcija, kurai ir tāda pati vidējā vērtība un autokorelācijas secība kā  $X(k)$ . Skaidrs, ja ir Gausa, tad  $m_n^x(\tau_1, \dots, \tau_{n-1}) = m_n^G(\tau_1, \dots, \tau_{n-1})$  un tādā veidā  $c_n^x(\tau_1, \tau_2, \dots, \tau_{n-1}) = 0$ . Tomēr ievērojiet, ka, lai arī 19. vienādojums ir patiesi tikai kārtām  $n=3$  un  $4$ ,  $c_n^x(\tau_1, \tau_2, \dots, \tau_{n-1}) = 0$  visiem  $n$ , ja  $X(k)$  ir Gausa. Kumulantu īpašības ir apkopotas 2. tabulā.

**2. tabula. Kumulanta īpašības**

<p>Kumulanti var tikt aplūkoti kā operatori <math>cum(x_1, \dots, x_n) = cum(x_{i_1}, \dots, x_{i_n}) \dots, x_n</math> lanta īpašībām.</p> <p>[CP1] Ja <math>\lambda_i, i = 1, 2, \dots, n</math> ir konstantes, evērojiet, ka tikai trešās un ceturrtās kārtas kumulantiem, <math>n=3, 4</math>)</p>
<p>[CP2] Kumulanti ir simetriski to argumentos, t.i.,</p> <p>kur <math>(i_1, \dots, i_n)</math> ir <math>(1, \dots, n)</math> permutācija; tas nozīmē, ka mēs varam mainīt kumulantu argumentus kā vien mēs gribam, nemainot kumulanta vērtību, piemēram, <math>c^x_4(\tau_1, \tau_2, \tau_3) = c^x_4(\tau_3, \tau_1, \tau_2) = c^x_4(\tau_2, \tau_3, \tau_1)</math>, utt.</p>
<p>[CP3] Summas kumulanti ir vienādi ar kumulantu summu; no tā arī cēlies nosaukums "kumulants". Piemēram, pat tad, kad <math>x_0</math> un <math>y_0</math> nav statistiski neatkarīgi, it patiesi, ka</p> $cum(x_0 + y_0, z_1, \dots, z_n) = cum(x_0, z_1, \dots, z_n) + cum(y_0, z_1, \dots, z_n)$
<p>[CP4] Kumulanti ir akli attiecībā uz konstanšu pieskaitīšanu, t.i., ja <math>\alpha</math> ir konstante, tad</p> $cum(\alpha + z_1, z_2, \dots, z_n) = cum(z_1, z_2, \dots, z_n)$
<p>[CP5] Statistiski neatkarīgu lielumu summas kumulanti ir vienādi atsevišķu lielumu kumulantu summai, t.i., ja nejauši mainīgie <math>\{x_i\}</math> ir neatkarīgi no nejaušiem mainīgajiem <math>\{y_i\}, i = 1, 2, \dots, k</math>, tad</p> $cum(x_1 + y_1, \dots, x_n + y_n) = cum(x_1, \dots, x_n) + cum(y_1, \dots, y_n)$
<p>Ievērojiet, ka, ja <math>x_i</math> un <math>y_i</math> nebūtu neatkarīgi, tad pēc [CP3] <math>cum(x_1 + y_1, \dots, x_n + y_n)</math> labajā pusē būtu <math>2n</math> nosacījumi. Statistiskā neatkarība samazina šo skaitu līdz diviem nosacījumiem.</p>
<p>[CP6] Ja <math>k</math> nejaušu mainīgo apakškopa <math>\{x_i\}</math> ir no pārējiem neatkarīga, tad</p> $cum(x_1, \dots, x_n) = 0$
<p><b>Piemērs.</b> Pieņemsim <math>Z(k) = aY(k) + V(k)</math>, kur <math>Y(k)</math> un <math>V(k)</math> ir statistiski neatkarīgi un <math>a</math> ir konstante; tad, no [CP5] un [CP1], <math>c^z_n(\tau_1, \dots, \tau_n) = a^n c^y_n(\tau_1, \dots, \tau_n) + c^v_n(\tau_1, \dots, \tau_n)</math>. Turklāt, ja <math>V(t)</math> ir statistiski neatkarīgs process, tad, no [CP6], <math>c^v_n(\tau_1, \dots, \tau_n) = 0</math>, tā kā <math>c^z_n(\tau_1, \dots, \tau_n) = a^n c^y_n(\tau_1, \dots, \tau_n)</math>. Ievērojiet, ka <math>c^v_n(\tau_1, \dots, \tau_n) = 0</math> tikmēr, kamēr <math>V(k)</math> ir Gausa. Tas ir tāpēc, ka jebkāda veida Gausa procesa visi augstākās kārtas kumulanti ir nulles; vienalga, vai tie ir krāsaini vai balti procesi.</p>

Sekojošas saistības starp  $\{X(k)\}$  momentu un kumulantu secībām eksistē kārtām  $n=1, 2, 3, 4$ .

**Pirmās kārtas kumulanti:**

$$c_1^x = m_1^x = E\{X(k)\} \quad (\text{mean value}) \quad (20)$$

**Otrās kārtas kumulanti:**

$$c_2^x(\tau_1) = m_2^x(\tau_1) - (m_1^x)^2 \quad (\text{covariance sequence}) \quad (21)$$

$$= m_2^x(-\tau_1) - (m_1^x)^2 = c_2^x(-\tau_1)$$

kur  $m^x_2(\tau_1)$  ir autokorelācijas secība. Tādā veidā, mēs redzam, ka otrās kārtas kumulanta secība ir *kovariācija*, kamēr otrās kārtas momenta secība ir *autokorelācija*.

*Trešās kārtas kumulanti:*

$$c^x_3(\tau_1, \tau_2) = m^x_3(\tau_1, \tau_2) \quad (22)$$

$$- m^x_1 \left[ m^x_2(\tau_1) + m^x_2(\tau_2) + m^x_2(\tau_1 - \tau_2) \right] + 2(m^x_1)^3 \quad \text{kur } m^x_3(\tau_1, \tau_2) \text{ ir trešās kārtas momenta secība. Tas izriet, ja mēs kombinējam 18. un 19. vienādojumu.}$$

*Ceturtais kārtas kumulanti:*

Kombinējot 18. un 19. vienādojumu, mēs iegūstam

$$c^x_4(\tau_1, \tau_2, \tau_3) = m^x_4(\tau_1, \tau_2, \tau_3)$$

$$- m^x_2(\tau_1) \cdot m^x_2(\tau_3 - \tau_2) - m^x_2(\tau_2) \cdot m^x_2(\tau_3 - \tau_1)$$

$$- m^x_2(\tau_3) \cdot m^x_2(\tau_2 - \tau_1) - m^x_1 [m^x_3(\tau_2 - \tau_1, \tau_3 - \tau_1)$$

$$+ m^x_3(\tau_2, \tau_3) + m^x_3(\tau_2, \tau_4) + m^x_3(\tau_1, \tau_2)] \quad (23)$$

$$+ (m^x_2)^2 [m^x_1(\tau_1) + m^x_2(\tau_2) + m^x_2(\tau_3) + m^x_2(\tau_3 - \tau_1)$$

$$+ m^x_2(\tau_3 - \tau_2) + m^x_2(\tau_2 - \tau_1)] - 6(m^x_1)^4.$$

Ja signāls  $\{X(k)\}$  ir nulle  $m^x_1 = 0$ , tad, ņemot vērā 21. un 22. vienādojumu, otrās un trešās kārtas kumulanti ir, attiecīgi, identiski otrās un trešās kārtas momentiem; taču, lai ģenerētu ceturtais kārtas kumulantus, mums ir vajadzīgas zināšanas par ceturtais un otrās kārtas momentiem 23. vienādojumā, t.i.,

$$c^x_4(\tau_1, \tau_2, \tau_3) = m^x_4(\tau_1, \tau_2, \tau_3) - m^x_2(\tau_1) \cdot m^x_2(\tau_3 - \tau_2)$$

$$- m^x_2(\tau_2) \cdot m^x_2(\tau_3 - \tau_1) - m^x_2(\tau_3) \cdot m^x_2(\tau_2 - \tau_1). \quad (24)$$

Ievietojot 21-23 vienādojumus  $\tau_1 = \tau_2 = \tau_3 = 0$  un pieņemot  $m^x_1 = 0$ , mēs iegūstam

$$\gamma^x_2 = E \{x^2(k)\} = c^x_2(0) \quad (\text{variancē})$$

$$\gamma^x_3 = E \{x^3(k)\} = c^x_3(0,0) \quad (\text{skewness}) \quad (25)$$

$$\gamma^x_4 = E \{x^4(k)\} - 3[\gamma^x_2]^2 = c^x_4(0, 0, 0) \quad (\text{kurtosis}).$$

Normalizētais kurtosis tiek definēts kā  $\gamma^x_4/[\gamma^x_2]^2$ .

1-D daļa no n-tās kārtas kumulanta tiek iegūta sasaldējot (n-2) no tā n-1 indeksiem. Ir iespējami daudzi 1-D daļu tipi, ieskaitot radiālos, vertikālos, horizontālos, diagonālos un diagonāli nobīdītos. Diagonālā daļa tiek iegūta, iestādot  $\tau_i = \tau$ ,  $i=1, 2, \dots, n-1$ . Visas šīs 1-D daļas ir ļoti noderīgas kumulantu lietojumprogrammās signālu apstrādē.

Loģisks jautājums būtu "Kāpēc mums ir vajadzīgi ceturtais kārtas kumulanti, t.i., vai ar trešās kārtas kumulantiem nav pietiekami?" Ja nejaus process ir simetriski sadalīts, tad tā trešās kārtas kumulants ir vienāds ar nulli; tādējādi, tādām procesam mums ir jālieto ceturtais kārtas kumulanti. Piemēram, Laplasa, vienvēidīgais, Gausa un Bernulli-Gausa sadalījumi ir simetriski, turpretim, eksponenciālie, Rayleigh un K-sadalījumi ir nesimetriski. Pie tam, dažiem procesiem ir ļoti mazi trešās kārtas kumulanti un daudz lielāki ceturtais kārtas kumulanti; tādējādi, tādiem procesiem mēs varētu lietot arī pēdējos. Galu galā, dažās specifiskās lietojumprogrammās (piemēram, harmoniku izguve un kubiskā fāzes savienošana pāros) trešās kārtas kumulanti ir vienādi ar nulli, turpretim ceturtais kārtas kumulanti ir nenulles.

Augstākās kārtas spektri tiek definēti vai nu kumulantu (piemēram, kumulants spektrs) vai momentu (piemēram, momenta spektrs) veidā. Kā vēlāk tiks izskaidrots, stohastisku signālu gadījumā ir noteiktas priekšrocības kumulantu izmantošanā; kamēr determinētiem signāliem ir labāk lietot momentus. Vienkārši noteikti augstākās kārtas spektri ir augstāko kārtu statistiku multidimensionālas Furjē transformācijas. Tādā veidā, enerģijas spektrs, bispektrs un trispektrs kumulantu veidā tiek definēti sekojoši:

*Enerģijas spektrs:*

$$C^x_2(\omega) = \sum_{\tau=-\infty}^{+\infty} c^x_2(\tau) \exp[-j(\omega\tau)], \quad (26)$$

$|\omega| \leq \pi$ , kur  $c^x_2(\tau)$  ir  $\{X(k)\}$  kovariācijas secība. 26. vienādojums ir zināms arī kā Wiener-Khintchine teorēma. No 21. un 26. vienādojuma, mums ir

$$c^x_2(\tau) = c^x_2(-\tau)$$

$$C^x_2(\omega) = C^x_2(-\omega) \quad (27)$$

$$C^x_2(\omega) \geq 0 \quad (\text{real, nonnegative function})$$

Kaut arī enerģijas spektrs bija aplūkots jau iepriekš, tā definīcija un īpašības šeit tiek atkārtotas, lai tās būtu viegli salīdzināt ar bispektru un trispektru.

*Bispektrs:  $n=3$*

$$C_3^x(\omega_1, \omega_2) = \sum_{\tau_1=-\infty}^{+\infty} \sum_{\tau_2=-\infty}^{+\infty} c_3^x(\tau_1, \tau_2) \exp[-j(\omega_1\tau_1 + \omega_2\tau_2)] \quad (28)$$

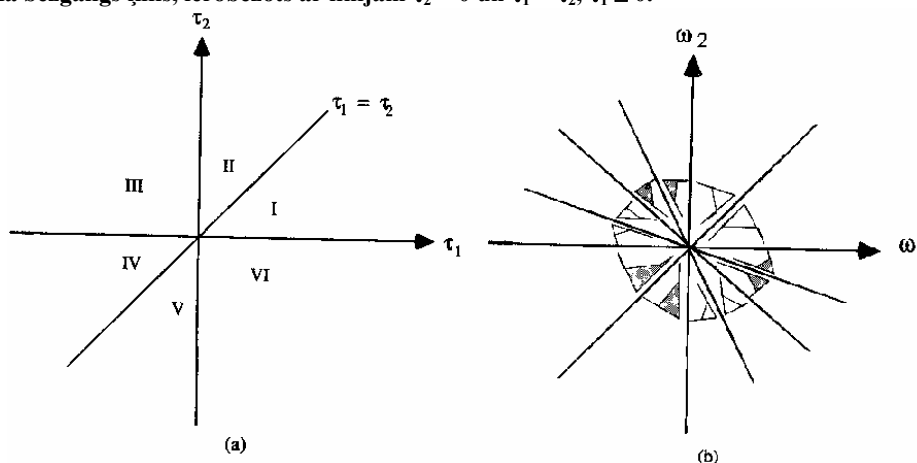
$$|\omega_1| \leq \pi, |\omega_2| \leq \pi, |\omega_1 + \omega_2| \leq \pi$$

kur  $c_3^x(\tau_1, \tau_2)$  ir  $\{X(k)\}$  trešās kārtas kumulantu secība, aprakstīta ar 22. vienādojumu.

Nozīmīgi simetrijas nosacījumi seko no momentu īpašībām un 22. vienādojuma:

$$\begin{aligned} c_3^x(\tau_1, \tau_2) &= c_3^x(\tau_2, \tau_1) = c_3^x(-\tau_2, \tau_1 - \tau_2) \\ &= c_3^x(\tau_2 - \tau_1, -\tau_1) = c_3^x(\tau_1 - \tau_2, -\tau_2) \\ &= c_3^x(\tau_1, \tau_2 - \tau_1) \end{aligned} \quad (29)$$

Rezultātā, zinot trešās kārtas kumulantus jebkurā no sešiem sektoriem, no I līdz VI (kā tas redzams 7a. zīmējumā), mums ir iespēja atrast visu trešās kārtas kumulantu secību. Šie sektori iekļauj to robežas, piemēram, sektors I ir kā bezgalīgs ķīlis, ierobežots ar līnijām  $\tau_2 = 0$  un  $\tau_1 = \tau_2, \tau_1 \geq 0$ .



7. zīm. (a) Trešās kārtas kumulantu simetriskie apgabali. (b) Bispektra simetriskie apgabali.

Bispektra definīcija 28. vienādojumā un trešās kārtas kumulantu īpašības 29. vienādojumā dod

$$\begin{aligned} C_3^x(\omega_1, \omega_2) &= C_3^x(\omega_2, \omega_1) \\ &= C_3^{x*}(-\omega_2, -\omega_1) = C_3^x(-\omega_1 - \omega_2, \omega_2) \\ &= C_3^x(\omega_1, -\omega_1 - \omega_2) = C_3^x(-\omega_1 - \omega_2, \omega_1) \\ &= C_3^x(\omega_2, -\omega_1 - \omega_2). \end{aligned} \quad (30)$$

Tādā veidā, ar zināšanām par bispektru triangulārā apgabalā  $\omega_2 \geq 0, \omega_1 \geq \omega_2, \omega_1 + \omega_2 \leq \pi$ , kas ir parādīts 7b. zīmējumā, ir pietiekami pilnīgai bispektra aprakstīšanai. Reāliem procesiem, bispektram ir 12 simetriski apgabali.

*Trispektrs:  $n=4$*

$$C_4^x(\omega_1, \omega_2, \omega_3) = \sum_{\tau_1=-\infty}^{+\infty} \sum_{\tau_2=-\infty}^{+\infty} \sum_{\tau_3=-\infty}^{+\infty} c_4^x(\tau_1, \tau_2, \tau_3) \exp[-j(\omega_1\tau_1 + \omega_2\tau_2 + \omega_3\tau_3)] \quad (31)$$

$$|\omega_1| \leq \pi, |\omega_2| \leq \pi, |\omega_3| \leq \pi, |\omega_1 + \omega_2 + \omega_3| \leq \pi$$

kur  $c_4^x(\tau_1, \tau_2, \tau_3)$  ir ceturtais kārtas kumulanta secība, kas uzdots ar 23. vienādojumu. No 31. vienādojuma ceturtais kārtas kumulantu definīcijas daudzas simetrijas īpašības var tikt iegūtas priekš trispektra, līdzīgi tām no bispektra.

Bispektrs un trispektrs vispārīgi ir kompleksas funkcijas, t.i., tām ir lielums un fāze:

$$C_n^x(\omega_1, \omega_2, \dots, \omega_{n-1}) = \left| C_n^x(\omega_1, \omega_2, \dots, \omega_{n-1}) \right| \cdot \exp j\psi_n^x(\omega_1, \omega_2, \dots, \omega_{n-1}) \quad (32)$$

$n = 3, 4, \dots$

Kumulanta spektrs ir noderīgāks nejašu signālu apstrādē, nekā momenta spektrs tāpēc, ka: (a)  $n > 2$  kārtas kumulantu spektri ir nulle, ja signāls ir Gausa, un, tādā veidā, nenulles kumulantu spektri nodrošina ne Gausa signālu mērīšanu; (b) kumulanti nodrošina piemērotu mēru statistiskās atkarības noteikšanai laika sērijās; (c) tāpat kā baltā trokšņa kovariācijas funkcija ir impulsa funkcija un tās spektrs ir līdzens, baltā trokšņa augstāko kārtu kumulanti ir daudzdimensionālas impulsa funkcijas un šī trokšņa polispektrs ir daudzdimensionāli līdzens; un (d) divu statistiski neatkarīgu nejašu procesu kumulanti ir vienādi atsevišķai nejašu procesu kumulantu summai, turpretim, viss tas pats nav patiess augstāku kārtu momentiem. Šī pēdējā īpašība ļauj mums viegli strādāt ar kumulantu kā ar operatoru.

Normalizēts augstākās kārtas spektrs vai  $n$ -tās kārtas koherences indekss ir funkcija, kas kombinē  $n$ -tās kārtas kumulanta spektru ar signāla enerģijas spektru ( $n=2$ ). Trešās un ceturtais kārtas koherences indeksi ir, attiecīgi, definēti ar

$$P_3^x(\omega_1, \omega_2) = \frac{C_3^x(\omega_1, \omega_2)}{\sqrt{C_2^x(\omega_1)C_2^x(\omega_2)C_2^x(\omega_1 + \omega_2)}} \quad , \quad (\text{biocoherency}) \quad (33a)$$

$$P_4^x(\omega_1, \omega_2, \omega_3) = \frac{C_4^x(\omega_1, \omega_2, \omega_3)}{\sqrt{C_2^x(\omega_1)C_2^x(\omega_2)C_2^x(\omega_3)C_2^x(\omega_1 + \omega_2 + \omega_3)}} \quad , \quad (\text{tricoherency}) \quad (33b)$$

Šīs funkcijas ir ļoti noderīgas, lai atklātu un raksturotu laika sēriju nelinearitātes, un, lai *atšķirtu lineārus procesus no nelineāriem*. Faktiski, saka, ka signāls ir lineārs  $n$ -tās kārtas ne Gausa process, ja  $n$ -tās kārtas koherences lielums  $|P_n^x(\omega_1, \dots, \omega_{n-1})|$  ir konstants pie visām frekvencēm; pretējā gadījumā saka, ka signāls ir nelineārs process.

Lineāra ne Gausa signāla augstākās kārtas spektrs  $C_n^x(\omega_1, \omega_2, \dots, \omega_{n-1})$  vienmēr var tikt uzrakstīts formā

$$C_n^x(\omega_1, \dots, \omega_{n-1}) = \gamma_n^x H(\omega_1) \dots H(\omega_{n-1}) H^*(\omega_1 + \dots + \omega_{n-1}) \quad (34)$$

kur  $\gamma_n^x$  ir skalāra konstante un  $H(\omega)$  ir lineāras, no laika neatkarīgas (LTI) sistēmas frekvences pārvietošanas funkcija. Ja ne Gausa signāls  $X(k)$  tiek ģenerēts ierosinot LTI sistēmu  $H(\omega)$  ar ne Gausa balto troksni  $V(k)$ , kā tas būs parādīts vēlāk, tad 34. vienādojums ir patiess visām kārtām. Speciālie gadījumi ietver

$$C_2^x(\omega) = \gamma_2^x |H(\omega)|^2 \quad (\text{Power Spectrum}) \quad (35a)$$

$$C_3^x(\omega_1, \omega_2) = \gamma_3^x H(\omega_1) \cdot H(\omega_2) \cdot H^*(\omega_1 + \omega_2) \quad (\text{Bispectrum}) \quad (35b)$$

$$C_4^x(\omega_1, \omega_2, \omega_3) = \gamma_4^x H(\omega_1) \cdot H(\omega_2) \cdot H(\omega_3) \cdot H^*(\omega_1 + \omega_2 + \omega_3). \quad (\text{Trispectrum}) \quad (35c)$$

Lai kā, ir arī ne Gausa signāli, kas var būt lineāri bispektra domēnā, t.i.,

$$C_3^x(\omega_1, \omega_2) = T(\omega_1) T(\omega_2) T^*(\omega_1 + \omega_2)$$

bet nelineāri trispektra vai augstāku kārtu spektru domēnos.

Kaut arī gan lineāra, gan nelineāra procesa enerģijas spektrs vienmēr var tikt uzrakstīts 35a. vienādojuma formā, augstākās kārtas spektri var būt 34. vienādojuma formā tikai tad, ja process ir lineārs (tam ir plakana lieluma koherence). Kombinējot 35a. vienādojumu ar 35b. un 35c., mēs redzam, ka lineāra procesa enerģijas spektrs var tikt atjaunots no tā bispektra vai trispektra ( $H(0) \neq 0$ ), t.i.,

$$C_3^x(\omega, 0) = C_2^x(\omega) H(0) \cdot \begin{pmatrix} \gamma_3^x \\ \gamma_2^x \end{pmatrix} \quad (36a)$$

$$C_4^x(\omega_1, \omega_2, 0) = C_3^x(\omega_1, \omega_2) H(0) \cdot \begin{pmatrix} \gamma_4^x \\ \gamma_3^x \end{pmatrix} \quad (36b)$$

kā arī bispektrs var tik atjaunots, ja ir trispektrs, lietojot

$$C_4^x(\omega_1, \omega_2, 0) = C_3^x(\omega_1, \omega_2)H(0) \begin{pmatrix} \gamma_4^y \\ \gamma_3^y \end{pmatrix} \quad (37)$$

Vienādojumi 36a un 36b var būt ļoti noderīgi, lai iegūtu signāla enerģijas spektra labākus vērtējumus no tā bispektra/ trispektra, kad papildus troksnis ir Gausa.

Ja  $W(k)$  ir stacionārs ne Gausa process ar  $E\{W(k)\} = 0$  un  $n$ -tās kārtas kumulantu secību

$$C_n^w(\tau_1, \dots, \tau_{n-1}) = \gamma_n^w \cdot \delta(\tau_1, \dots, \tau_{n-1}) \quad (38a)$$

kur  $\delta(\tau_1, \dots, \tau_{n-1})$  ir delta funkcija, tad saka, ka  $W(k)$  ir  $n$ -tās kārtas baltums. Ņemot 38a. vienādojuma  $(n-1)$ -dimensionālu Furjē transformāciju, mēs iegūstam

$$C_n^w(\omega_1, \dots, \omega_{n-1}) = \gamma_n^w \quad (38b)$$

kurš ir līdzens spektrs visām frekvencēm. Baltā procesa augstākās kārtas spektra speciālie gadījumi ietver

$$\begin{aligned} C_2^w(\omega) &= \gamma_2^w \quad (\text{Power Spectrum}) \\ C_3^w(\omega_1, \omega_2) &= \gamma_3^w \quad (\text{Bispectrum}) \\ C_4^w(\omega_1, \omega_2, \omega_3) &= \gamma_4^w \quad (\text{Trispectrum}) \end{aligned} \quad (39)$$

kur  $\gamma_2^w$ ,  $\gamma_3^w$  un  $\gamma_4^w$  ir, attiecīgi,  $\{W(k)\}$  dispersija, asimetrija un kurtosis.

Determinētu signālu augstāko kārtu momentu spektra definīcijas ir apkopotas 3. tabulā.

### 3. tabula. Determinētu signālu augstāko kārtu momenta spektrs.

Mēs dodam determinētu signālu

$$R_4^{\hat{x}}(\lambda_1, \lambda_2, \lambda_3) = \frac{1}{N} X(\lambda_1)X(\lambda_2)X(\lambda_3)X^*(\lambda_1 + \lambda_2 + \lambda_3 + \omega_3) \quad \text{īcijas.}$$

- ◆ ENERĢIJAS SIGNĀLI  
Pieņemsim, ka  $\{x(k)\}$

signāls. Tad

Furjē transformācija:

Enerģijas spektrs:

$$M_2^x(\omega_1, \omega_2) = X(\omega)X^*(\omega)$$

Bispektrs:

Trispektrs:

- ◆ PERIODISKI SIGNĀLI

Pieņemsim, ka ir periodisks signāls ar periodu  $N$ . Tad

Furjē rindas:

Enerģijas spektrs:

Bispektrs:

Trispektrs:

### b. Augstākās kārtas spektra izskaitļošana no datiem

Problēma, ar ko nākas saskarties praksē ir novērtēt augstākās kārtas spektru procesam, kad ir dots galīgs skaits mērījumu. Divas no vispopulārākajām tradicionālajām pieejām ir *netiešā* un *tiešā* metodes, kuras var uzskatīt kā tiešas aproksimācijas augstākās kārtas spektra definēšanai. Tā kā šīs aproksimācijas ir vienkāršas, dažreiz nepieciešamā izskaitļošana var būt dārga, par spīti tam, ka tiek izmantoti ātro Furjē Transformāciju (FFT) algoritmi. Mēs šeit aprakstam abas tradicionālās metodes kā bispektrālās novērtētājas.

**Netiešā metode**

Lai  $\{X(1), X(2), \dots, X(N)\}$  ir dotā datu kopa. Tad mums ir sekojošais:

1. Segmentē datus  $K$  ierakstos no  $M$  paraugiem katru, t.i.,  $N = KM$ .
2. Atņem vidējo vērtību no katra ieraksta.
3. Pieņemot, ka  $\{x^{(i)}(k), k = 0, 1, \dots, M - 1\}$  ir datu kopa uz segmentu  $i = 1, 2, \dots, K$ , iegūst novērtējumu trešā-momenta secībai

$$r^{(i)}(m, n) = \frac{1}{M} \sum_{\ell=s_1}^{s_2} x^{(i)}(\ell) x^{(i)}(\ell + m) x^{(i)}(\ell + n)$$

kur  $i = 1, 2, \dots, K$

$$s_1 = \max(0, -m, -n)$$

$$s_2 = \min(M-1, M-1-m, M-1-n).$$

4. Vidējais  $r^{(i)}(m, n)$  par visiem segmentiem

$$\hat{c}_3^x(m, n) = \frac{1}{K} \sum_{i=1}^K r^{(i)}(m, n)$$

5. Ģenerē bispektrālo novērtējumu

$$\hat{C}_3^x(\omega_1, \omega_2) = \sum_{m=-L}^L \sum_{n=-L}^L \hat{c}_3^x(m, n) W(m, n) \exp[-j(\omega_1 m + \omega_2 n)]$$

kur  $L < M - 1$  un  $W(m, n)$  ir 2-dimensionāla loga funkcija. Piezīmēsim, ka katra bispektrālā novērtējuma izskaitļošana var būt būtiski reducēta, ja  $r^{(i)}(m, n)$  kalkulācijām tiek ņemtas vērā simetrijas īpašības trešās pakāpes koncentrētai trešās pakāpes spektrālā blīvuma jaudai (*Vien.29*) un arī, ja bispektra, kas parādīts vien.30 simetrijas īpašības tiek ņemtas vērā, izdarot skaitļošanu.

Kā gadījumā ar tradicionālo pakāpes spektra novērtēšanu, lai sasniegtu labāku novērtējumu, būtu jāizmanto piemēroti logi. Logu funkcijām būtu jāapmierina sekojoši ierobežojumi:

- a)  $W(m, n) = W(n, m) = W(-m, n - m) = W(m - n, -n)$  (simetrijas īpašības trešās pakāpes koncentrētai trešās pakāpes spektrālā blīvuma jaudai);
- b)  $W(m, n) = 0$  ārpus  $\hat{c}_3^x(m, n)$  atbalsta;
- c)  $W(0, 0) = 1$  (normalizējošs stāvoklis);
- d)  $W(\omega_1, \omega_2) \geq 0$ , visiem  $(\omega_1, \omega_2)$ .

Funkciju klase, kas apmierina ierobežojumus priekš  $W(m, n)$  ir:

$$W(m, n) = d(m)d(n)d(n - m) \quad (40)$$

kur

$$d(m) = d(-m)$$

$$d(m) = 0, m > L$$

$$d(0) = 1$$

$$D(\omega) \geq 0, \text{ for all } \omega.$$

Vienādojums (40) atļauj rekonstruēt 2-dimensinālo logu funkcijas priekš bispektrālā novērtējuma, lietojot standarta 1-dimensinālos kavējuma logus. Divas labas izvēles no 1-dimensināliem logiem ir:

$$d_0(m) = \begin{cases} \frac{1}{\pi} \left| \sin \frac{\pi m}{L} \right| + \left(1 - \frac{|m|}{L}\right) \left(\cos \frac{\pi m}{L}\right), & |m| \leq L \\ 0, & |m| > L. \end{cases} \quad \text{a) Optimālais logs (minimuma bispektruma noslieces supremum):} \quad (41a)$$

b) *Parzen* logs:

$$d_p(m) = \begin{cases} 6 \left(\frac{|m|}{L}\right)^2 + 6 \left(\frac{|m|}{L}\right)^3, & |m| \leq \frac{L}{2} \\ 2 \left(1 - \frac{|m|}{L}\right), & \frac{L}{2} \leq |m| \leq L \\ 0, & |m| > L. \end{cases} \quad (41b)$$

*Tiešā metode*

Lai  $\{X(1), X(2), \dots, X(N)\}$  ir pieejamā kopa no novērojumiem bispektrālam novērtējumam. Pieņemsim, ka  $f_s$  ir diskretizēšanas frekvence un  $\Delta_0 = f_s/N_0$  ir nepieciešamā atstarpe starp frekvenču paraugiem bispektrālā domenā gar horizontālo un vertikālo virzieniem, tādā veidā,  $N_0$  ir kopējais frekvenču paraugu skaits. Tad mums ir sekojošais:

**a) Segmentē datus K segmentos no M paraugiem katru, t.i.,  $N = KM$  un atņem vidējo vērtību no katra segmenta. Ja nepieciešams, tas atņem nulles katra segmenta beigās, lai iegūtu tradicionālo M garumu priekš FFT.**

b) Pieņemot, ka  $\{x^{(i)}(k), k = 0, 1, 2, \dots, M-1\}$  ir datu segments  $\{i\}$ , ģenerē DFT koeficientus

$$Y^{(i)}(\lambda) = \frac{1}{M} \sum_{k=0}^{M-1} x^{(i)}(k) \exp(-j2\pi k\lambda/M), \lambda = 0, 1, \dots, \frac{M}{2}$$

$$i = 1, 2, \dots, K.$$

c) Vispārīgi,  $M = M_1 \times N_0$ , KUR  $M_1$  ir pozitīvs vesels skaitlis (pieņemot par nepāri), t.i.,  $M_1 = 2L_1 + 1$ . Tā kā  $M$  ir pāris un  $m_1$  ir nepāris, mēs kompromitējam vērtību  $n_0$  (TUVĀKAIS VESELAIS SKATLIS). FORMĒ BISPEKTRUMA NOVĒRTĒJUMU PAMATOJOTIES UZ MATRICAS DFT KOEFICIENTIEM

$$\hat{b}_i(\lambda_1, \lambda_2) = \frac{1}{\Delta_0^2} \sum_{k_1=-L_1}^{L_1} \sum_{k_2=-L_1}^{L_1} Y^{(i)}(\lambda_1+k_1) Y^{(i)}(\lambda_2+k_2) Y^{(i)*}(\lambda_1+\lambda_2+k_1+k_2)$$

virš trīstūra apgabala  $0 \leq \lambda_2 \leq \lambda_1$ ,  $\lambda_1 + \lambda_2 = f_s/2$ . Īpašā gadījumā, kad vidējošana netiek izdarīta bispektruma domenā,  $M_1 = 1$ ,  $L_1 = 0$  un tāpēc:

$$\hat{b}_i(\lambda_1, \lambda_2) = \frac{1}{\Delta_0^2} Y^{(i)}(\lambda_1) Y^{(i)}(\lambda_2) Y^{(i)*}(\lambda_1 + \lambda_2)$$

d) Bispektra novērtējums no dotajiem datiem ir vidējais no K gabaliem

$$\hat{C}_3^x(\omega_1, \omega_2) = \frac{1}{K} \sum_{i=1}^K \hat{b}_i(\omega_1, \omega_2)$$

kur

$$\omega_1 = \left( \frac{2\pi f_s}{N_0} \right) (\lambda_1) \text{ and } \omega_2 = \left( \frac{2\pi f_s}{N_0} \right) (\lambda_2).$$

**c. Tradicionālo novērtētāju īpašības un asimptotiskā daba**

Aplūkosim bispektrālu gadījumu. Pieņemsim, ka  $C^x(\omega)$  un  $C^x(\omega_1, \omega_2)$  ir attiecīgi stacionārā nulles signāla patiesie jaudas spektrs un bispektrs. Lai  $\hat{C}_3^x(\omega_1, \omega_2)$  ir konsekvents novērtējums, kas izskaitļots pēc tiešās vai netiešās metodes, izmantojot vienkārša signāla ar garumu  $N$  realizāciju. Atrisinājuma rezultāts, izmantojot šīs metodes nozīmē, ka priekš pietiekoši liela ierakstu skaita  $M$  un kopējā garuma  $N$ , abas – tiešās un netiešās pieejas nodrošina apmēram objektīvus novērtējumus:

$$E\{\hat{C}_3^x(\omega_1, \omega_2)\} \equiv C_3^x(\omega_1, \omega_2) \quad (42)$$

ar asimptotiskām neatbilstībām

$$\begin{aligned} \text{var}\{\text{Re}[\hat{C}_3^x(\omega_1, \omega_2)]\} &\equiv \text{var}\{\text{Im}[\hat{C}_3^x(\omega_1, \omega_2)]\} \\ &\equiv \frac{1}{2} \sigma_2^2(\omega_1, \omega_2), \end{aligned} \quad (43a)$$

kur

$$\sigma_2^2(\omega_1, \omega_2) = \begin{cases} \frac{VL^2}{MK} C_2^x(\omega_1) C_2^x(\omega_2) C_2^x(\omega_1 + \omega_2) & \text{(indirect)} \\ \frac{N_0^2}{MK} C_2^x(\omega_1) C_2^x(\omega_2) C_2^x(\omega_1 + \omega_2) & \text{(direct)} \end{cases}$$



(43b)

kad  $0 < \omega_2 < \omega_1$ , kur  $K$  ir ierakstu skaits,  $M$  – paraugu skaits uz ierakstu un  $V$  ir kopējā enerģija bispektra logam, kas ir vienota taisnstūra logam;  $L$  ir definēts solī (5) pie netiešās metodes un  $N_0 = (M/M_1)$  ir definēts tiešās metodes aprakstā. Vienādojumos 43a un 43b ir redzams, ka, ja taisnstūra logs tiek lietots ar netiešo metodi un  $L = N_0$ , tad divas tradicionālās metodes dod apmēram tos pašus novērtējumus.

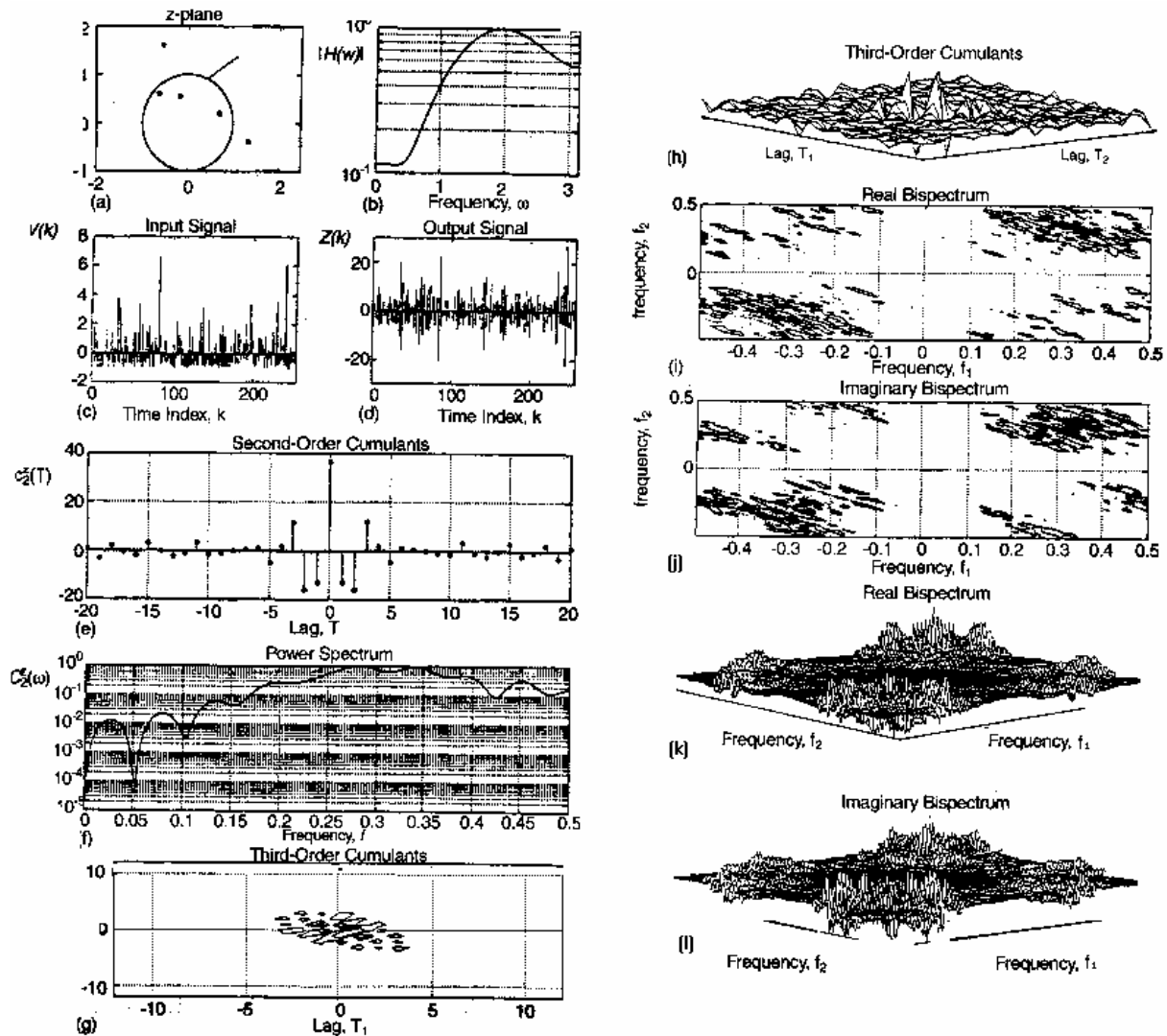
Šeit ir parādīts, ka pie lieliem  $M$  un  $N$ , ir novērojama kļūdas bikoherence,

$$\frac{\hat{C}_3^x(\omega_1, \omega_2) - C_3^x(\omega_1, \omega_2)}{\sigma_3(\omega_1, \omega_2)} \sim N_c(0, 1) \quad (44)$$

kas ir apmēram kompleksais Gausa variants ar vidējo nulli un netabilstību vienādu ar vienu. Cits, tikpat svarīgs rezultāts, kas seko no asimptotisko rezultātu kopas ir, ka šie statistiskie rezultāti var tikt interpretēti kā neatkarīgi un nejauši ņemti mainīgie sieta robežās, ja sieta platums ir lielāks vai vienāds ar bispektrālo caurlaides joslu, t.i.,  $\hat{C}_3^x(\omega_j, \omega_k)$  un  $\hat{C}_3^x(\omega_r, \omega_s)$  ir neatkarīgi, ja  $j \neq r$  vai  $k \neq s$  ja  $|\omega_{j+1} - \omega_j| \geq 2\pi\Delta_0$  vai  $|\omega_{r+1} - \omega_r| \geq 2\pi\Delta_0$ , kur

$$\Delta_0 \triangleq \begin{cases} \frac{1}{L} & \text{(indirect)} \\ \frac{1}{N_0} & \text{(direct)} \end{cases} \quad (45)$$

Asimptotiskā neatkarība un Gausa teorēma apgalvo, ka tādā veidā apstrādāta bikoherences statistika ir apmēram ne-centrālā statistika ar pielaidi 2 grādi. Šeit ir pateikts, ka šī aproksimācija ir spēkā paraugiem, kuriem  $N = 256$ , ja  $\Delta_0 \cong \sqrt{N}$ . Ja procesam ir nulles bispektrs  $C_3^x(\omega_1, \omega_2) = 0$ , tad  $ch_3(\omega_1, \omega_2)$  ir centrālais variants ar 2 grādu pielaidi.



8. (a) Neminimuma fāzes sistēmas nulles, (b) Neminimuma fāzes sistēmas lieluma atbildes reakcija, (c) ievada nejauša secība (vidējās nulles vienas puses rādītājs izdalīts ar neatbilstību (1), (d) izvada secība, (e) otrās pakāpes izvada koncentrēta spektrālā blīvuma jauda, (f) izvada jaudas spektrs, (g) izvada trešās pakāpes koncentrētas spektrālā blīvuma jaudas kontūrs, (h) izvada trešās pakāpes koncentrētas spektrālā blīvuma jaudas 3D grafiks, (i) izvada bispektruma reālās daļas kontūrs, (j) izvada bispektruma imaginārās daļas kontūrs, (k) izvada bispektruma reālās daļas 3D grafiks, (l) izvada bispektruma imaginārās daļas 3D grafiks.

No vienādojumiem 43a un 43b ir redzams, ka bispektra neatbilstība, kas saistīta ar tradicionāliem novērtējumiem var tikt reducēta: (i) palielinot ierakstu skaitu, (ii) samazinot loga atbalsta apgabala izmēru koncentrētas spektrālā blīvuma jaudas domēnā ( $L$ ) vai arī palielinot loga nogludināšanas frekvenci ( $M_1$ ) un (iii) palielinot ierakstu skaitu  $M$ . Kaut gan, ierakstu skaita ( $K$ ) palielināšana ir prasīga uz kompjūtera laiku un var radīt potenciālas nestacionaritātes. No otras puses, bieža domēna vidējošana lieliem taisnstūriem ar izmēru ( $M_1$ ) vai izmantojot koncentrētas spektrālā blīvuma jaudas logus ar mazām vērtībām  $L_0$ , samazina frekvences izšķirtspēju un var palielināt troksni. "Īsa garuma" datu gadījumā, lielums  $K$  ir jāpalielina, izmantojot pārklājošos ierakstus.

Tradicionālām metodēm ir priekšrocība būt viegli izpildītām (var tikt lietoti FFT algoritmi) un tas nodrošina labus rezultātus pietiekoši gariem datiem. Kaut gan, tāpēc ka FFT transformācijām piemīt "nenoteiktības princips", šo metožu spēja atrisināt harmoniskās komponentes daudzspektru domēnos ir ierobežota. No šejienes izriet, ka tas rada problēmu, nosakot kvadrārisku fāžu saiti tuvu novietotos frekvenču pāros.

Viena no atslēgas priekšrocībām tradicionālajiem bispektra (vai augstākās kārtas spektra) novērtējumiem ir to asimptotiskās Gausa īpašības, kas parādītas ar vienādojumiem 44 – 46. Asimptotiskie rezultāti kalpo kā tilts starp Varbūtīgā Proporciju Testa (LRT) detektoriem un Maksimālās Varbūtības (ML) Teoriju no vienas puses un augstākās kārtas spektru (HOS) no otras puses. Detektēšanas, parametru novērtēšanas un klasifikācijas shēmas var tikt attīstītas nozīmīgām signālu apstrādes aplikācijām, izmantojot asimptotiskās statistiskās HOS novērtējumu īpašības un LRT vai ML teoriju.

## Lineārs process

a. Koncentrēta spektrālā blīvuma jauda un Polispektrs

**Šajā gadījumā sistēma Zīm.4 tiek uzskatīta par kauzālu un eksponenciāli stabilu un  $\{V(k)\}$  tiek uzskatīta par neatkarīgu, identiski izdalītu (n.i.d.) un ne-Gausa funkciju, t.i.,**

$$c_n^y(\tau_1, \dots, \tau_{n-1}) = \begin{cases} \gamma_n^y, & \tau_1 = \tau_2 = \dots = \tau_{n-1} = 0 \\ 0, & \text{otherwise} \end{cases} \quad (47)$$

kur  $\gamma_n^y$  nozīmē n-tās pakāpes  $V(k)$  koncentrētu spektrālā blīvuma jaudu. Papildus troksnis  $N(k)$  tiek uzskatīts par Gausa funkciju, bet nedrīkst būt baltais troksnis. To vispārinājumi uz vienādojumiem 15 un 16 ir

$$c_n^z(\tau_1, \dots, \tau_{n-1}) = \gamma_n^y \sum_{k=0}^{\infty} h(k)h(k + \tau_1) \dots h(k + \tau_{n-1}) \quad (48)$$

un

$$C_n^z(\omega_1, \dots, \omega_{n-1}) = \gamma_n^y H(\omega_1)H(\omega_2) \dots H(\omega_{n-1})H\left(-\sum_{i=1}^{n-1} \omega_i\right) \quad (49a)$$

kuri arī var tikt izteikti kā multi dimensionālas z-transformācijas kā

$$C_n^z(z_1, \dots, z_{n-1}) = \gamma_n^y H(z_1)H(z_2) \dots H(z_{n-1})H\left(-\prod_{i=1}^{n-1} z_i^{-1}\right) \quad (49b)$$

Ievērojiet, kad  $n = 2$ , vienādojumi 48 un 49 reducējas uz vien.15 [ $c_n^N(\tau)$  saskatīšanas subjekts] un vien.16 [ $C_n^N(z)$  saskatīšanas subjekts]. Vienādojumi 49a un 49b ir bijuši sākumpunkti daudziem bezparametru un parametru polispektrāliem paņēmieniem, kas ir tikuši attīstīti dažu pēdējo gadu laikā. Viens, ļoti svarīgs pielietojums vien.48 ir izskaitļot koncentrētu spektrālā blīvuma jaudu sistēmas modeļiem. Šī procedūra ietver: (1) modeļa IR noteikšanu,  $h(k)$   $k = 0, 1, 2, \dots, K$ ; (2)  $\tau_j$  vērtību nolabošanu un vien.48 izskaitļošanu; un (3) soļa (2) atkārtošana visām  $\tau_j$  vērtībām mūs interesējošā domēnā (jāatceras izmantot koncentrētas spektrālā blīvuma jaudas simetrijas īpašības, lai samazinātu skaitļošanas laiku). Tas notiek tā, kā parādīti rezultāti Zīm.8g un iegūts lielums  $h$ . Svarīgs vien. 49a pielietojums ir izskaitļot polispektru sistēmas modeļiem. Tas notiek tā: (1) nosaka modeļa IR,  $h(k)$   $k = 0, 1, 2, \dots, K$ ; (2) izskaitļo DFT no  $h(k)$ ,  $H(\omega)$ ; (3)  $\omega_j$  vērtību nolabošana un vien.49a izskaitļošana un (4) atkārto soli (3) visām  $\omega_j$  vērtībām mūs interesējošā domēnā (jāatceras izmantot polispektra simetrijas īpašības, lai samazinātu skaitļošanas laiku). Protams, cits veids, kā izskaitļot polispektru ir vispirms izrēķināt koncentrētu spektrālā blīvuma jaudu, kā tikko izskaidrots un tad izrēķināt to daudzdimensionālo DFT.

Vispārinājums vien.48 un 49 gadījumam ar krāsainu ne-Gausa ievadu ir daudz vieglāk vizualizējams polispektrālā domēnā un ir:

$$C_n^z(\omega_1, \dots, \omega_{n-1}) = \gamma_n^y H(\omega_1) H(\omega_2) \dots H(\omega_{n-1}) H\left(-\sum_{i=1}^{n-1} \omega_i\right) \quad (50)$$

Vispārinājums vien.48 krāsainam troksnim ir arī dots Mendeļa (Mendel 1991) teorēmā, tas ir,  $a(n-1)$  – slēgta konvolūcija starp  $cvn(\tau_1^{2x}, \dots, \tau_{n-1}^{2x})$  un

$$c_n^h(\tau_1^{2x}, \dots, \tau_{n-1}^{2x}) = \sum h(k) h(k + \tau_1) \dots h(k + \tau_{n-1}).$$

*Piemērs 1.* Pieņemsim, ka  $h(k)$  ir impulsa reakcija (IR) kauzālai MA sistēmai. Tādai sistēmai ir galīga IR un tiek aprakstīta ar sekojošu modeli:

$$Y(k) = \sum_{i=0}^q b(i)V(k-i) \quad (51)$$

MA parametri ir  $b(0), b(1), \dots, b(q)$ , kur  $q$  ir MA modeļa kārtā un  $b(0)$  ir parasti uzskatāma par vienādu ar viens [mērogošana ir saistīta ar  $V(k)$  statistiku]. Ir labi zināms, ka šim modelim  $h(i) = b(i)$ ,  $i = 0, 1, \dots, q$ ; tādēļ, kad  $\{V(k)\}$  ir i.i.d., varam atrast no vien.48, ka

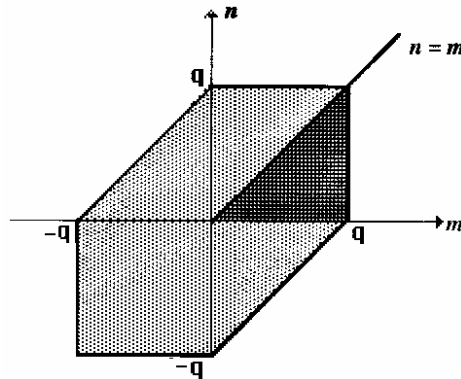
$$c_3^y(\tau_1, \tau_2) = \gamma_3^y \sum_{l=0}^q b(l)b(l+\tau_1)b(l+\tau_2) \quad (52)$$

un

$$c_4^y(\tau_1, \tau_2, \tau_3) = \gamma_4^y \sum_{l=0}^q b(l)b(l+\tau_1)b(l+\tau_2)b(l+\tau_3). \quad (53)$$

Interesants ir jautājums “Kurām  $\tau_1$  un  $\tau_2$  vērtībām  $c_3^y(\tau_1, \tau_2)$  nav nulle?”. Protams, līdzīgs jautājums var tikt uzdots priekš  $c_4^y(\tau_1, \tau_2, \tau_3)$ . Atbilde uz šo jautājumu ir redzama Zīm.9. Atbalsta domēns priekš  $c_3^y(\tau_1 = m, \tau_2 = n)$  ir sešu pušu ēnots apgabals. Tas ir pateicoties MA sistēmas FIR dabai. Tumši ēnots trijstūra reģions pirmajā kvadrantā ir principā zudis apgabals. Šajā stacionārajā gadījumā, mums ir tikai jānosaka trešās pakāpes koncentrētas spektrālā blīvuma jaudas vērtības šajā apgabalā,  $R$ , kur

$$R = \{m, n: 0 \leq n \leq m \leq q\} \quad (54)$$



9. Atbalsta domēns priekš  $c_3^y(m, n)$  priekš  $MA(q)$  sistēmas. Tumši ēnotais  $c_3^y$  apgabals ir pamatā pilnveidots Zīm. 54.

Ievērojiet no Zīm.9, ka trešās pakāpes koncentrēta spektrālā blīvuma jauda ir vienāda ar nulli vienam vai abiem tā argumentiem, kas vienādi ar  $q + 1$ . Tas nozīmē, ka ir jābūt iespējamam noteikt MA modeļa pakāpi  $q$ , veicot pārbaudi statistiskā ziņā, trešās pakāpes koncentrētas spektrālā blīvuma jaudas  $c_3^y(q + 1, 0)$  mazumam.

**Tabula 1, kas parādīta kā vienkāršs piemērs trīs MA sistēmām, kam ir tieši vieni un tie paši spektri un autokolerācijas; tādēļ, tās nevar atrisināt, izmantojot otrās pakāpes statistiku. Tabula 4 turpina šo piemēru. Tā dod trešās pakāpes koncentrētu spektrālā blīvuma jaudu katrai  $MA(2)$  sistēmai priekš  $R = \{m, n: 0 \leq n \leq m \leq 2\}$ . Ievērojiet, ka trešās pakāpes koncentrēta spektrālā blīvuma jauda trīs sistēmām ir pietiekoši atšķirīga, tāpēc mums ir pieņemams, ka ir iespējams atrisināt tās, lietojot šo koncentrēto spektrālā blīvuma jaudu. Mēs pierādīsim, ka tas ir patiešām gadījums zemāk.**

Tabula 4. Minimuma un maksimuma un jauktās fāžu sistēmas ar identisku jaudas spektru (vai autokolerācijām), bet ar atšķirīgu trešās pakāpes statistiku  $0 < a < 1, 0 < b < 1$ .

Trešās pakāpes cumulanti			
$c_3^y(0,0)$	$1 - (a + b)^3 + a^3b^3$	$1 - (a + b)^3 + a^3b^3$	$(1 + ab)^3 - a^3 - b^3$
$c_3^y(1,0)$	$-(a + b) + ab(a + b)^2$	$(a + b)^2 - (a + b)a^2b^2$	$a^2(1 + ab) - (1 + ab)^2b$
$c_3^y(1,1)$	$(a + b)^2 - (a + b)a^2b^2$	$-(a + b) + ab(a + b)^2$	$-a(1 + ab)^2 + (1 + ab)b^2$
$c_3^y(2,0)$	$ab$	$a^2b^2$	$-a^2b$
$c_3^y(2,1)$	$-(a + b)ab$	$-(a + b)ab$	$ab(1 + ab)$
$c_3^y(2,2)$	$a^2b^2$	$ab$	$-ab^2$

**b. Polispektrs**

Sāksim ar bispektru

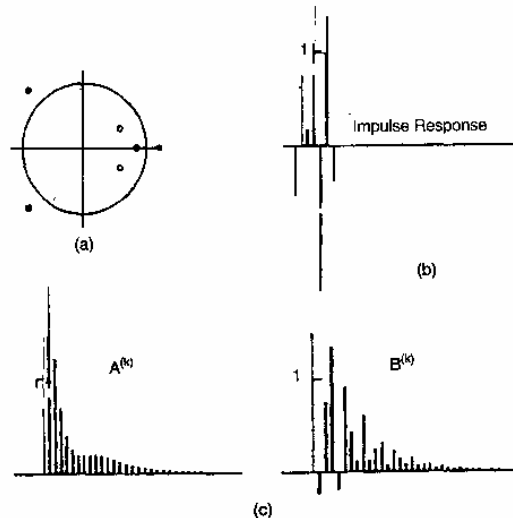
$$C_3^y(z_1, z_2) = \gamma_3^y H(z_1) H(z_2) H(z_1^{-1}, z_2^{-1}), \tag{55}$$

Šeit ņem logaritmu no  $C_3^y(z_1, z_2)$ ,  $\hat{C}_3^y(z_1, z_2) = \ln C_3^y(z_1, z_2)$  un ņem inverso 2D z-transformāciju no  $\hat{C}_3^y(z_1, z_2)$ , lai iegūtu **komplekso bispektru**  $b_y(m, n)$ . Piezīmējam, ka šie soļi ir paralēli soļi, kuri iepriekš saistīja komplekso spektra daļu  $h(k)$ ,  $\hat{h}(k)$ . Arī piezīmējam, ka salīdzināmie rezultāti ir arī iegūti ar to palīdzību priekš kompleksā trispektra,  $t_y(m, n, l)$ . Vispārīgi,  $b_y(m, n)$ ,  $t_y(m, n, l)$  un augstākas kārtas kompleksie spektri ir zināmi kā "polispektrs".

Kad  $H(z)$  ir parādīts kā vien. 7 – 9 (skatīt arī zīm.5), tad tiek parādīts, ka: (1)  $b_y(m, n)$  ir nulles vērtības tikai, kad  $m = n = 0$ , veselu skaitļu vērtības gar  $m$  un  $n$  asīm, un šo vērtību pārklāšanās vietā gar 45 grādu līniju  $m = n$ , t.i.,

$$b_y(m, n) = \begin{cases} \ln \gamma_3^y c_l & m = 0, n = 0 \\ -1/nA^{(n)} & m = 0, n > 0 \\ -1/mA^{(m)} & n = 0, m > 0 \\ 1/mB^{(-m)} & n = 0, m < 0 \\ 1/nB^{(-n)} & m = 0, n < 0 \\ -1/nB^n & m = n > 0 \\ 1/nA^{(-n)} & m = n < 0 \\ 0 & otherwise \end{cases} \tag{56}$$

kur "A" un "B" lineārās spektra daļas koeficienti tiek definēti vien.13 un 14 attiecīgi; un (2)  $t_y(m, n, l)$  ir nulles vērtības tikai, kad  $m = n = l = 0$ , veselu skaitļu vērtības gar  $m$ ,  $n$  un  $l$  asīm, un šo vērtību pārklāšanās vietā gar 45 grādu līniju  $m = n = l$  tikai īpašām  $t_y(m, n, l)$  vērtībām, kuras ir salīdzināmas ar vien.56



10. Jauktās fāzes MA sistēma ar spilgti izteiktām rezonansēm: (a) nulles atrašanās vieta, (b) impulsa reakcija un (c) lineārās spektra daļas koeficienti [definēti vien. 13 un 14 attiecīgi]

Piemērs 2. Nekauzāls MA(3,3) modelis ar spilgti izteiktām rezonansēm ir attēlots zīm 10. Tā transformācijas funkcija ir

$$H(z) = \frac{(1 - 0.869z)(1 + 1.1z + 0.617z^2)}{(1 - 0.85z^{-1})(1 - 1.2z^{-1} + 0.45z^{-2})} \tag{57}$$

Zīm.10 ir attēloti arī “A” un “B” lineārās spektra daļas koeficienti, no kuriem tas ir vienkāršots, lietojot vien.56, lai izskaitļotu bispektra koeficientus  $b_y(m, n)$ . “A” un “B” lineārās spektra daļas koeficientiem ir dažas ļoti interesantas un svarīgas īpašības, kuras ir apkopotas Tabulā 5.

Tabula 5. Lineārās spektra daļas koeficientu īpašības

Šeit mēs dodam lineārās spektra daļas koeficientu īpašības.

Lai  $h(k)$  ir determinēts enerģijas signāls ar Furjē transformāciju

$$H(\omega) = |H(\omega)| \exp(j\phi(\omega))$$

un kompleksu spektru

$$c_h^{(m)} = F^{-1}[\ln |H(\omega)|] = \begin{cases} \frac{A^{(m)}}{m}, & m > 0 \\ -\frac{B^{(-m)}}{m}, & m < 0 \end{cases}$$

kur  $F^{-1}[\bullet]$  apzīmē inverso Furjē transformāciju

Lineārā spektra daļa no  $|H(\omega)|$  ir funkcija  $\{A^{(m)} + B^{(m)}\}$ , t.i.,

$$p_h^{(m)} = F^{-1}[\ln |H(\omega)|] = \begin{cases} \frac{1}{m} [A^{(m)} + B^{(m)}], & m > 0 \\ \frac{1}{m} [A^{(-m)} + B^{(-m)}], & m < 0 \end{cases}$$

turpretim lineārā spektra daļa no  $e^{j\phi(\omega)}$  ir funkcija  $\{A^{(m)} + B^{(m)}\}$ , t.i.,

$$f_h^{(m)} = F^{-1}[\ln \exp(j\phi(\omega))] = \begin{cases} -\frac{1}{m} [A^{(-m)} - B^{(-m)}], & m < 0 \\ -\frac{1}{m} [A^{(m)} - B^{(m)}], & m > 0 \end{cases}$$

Lineārās spektra daļas koeficienti  $A^{(m)}$ ,  $B^{(m)}$  var tikt izskaitļoti tieši no augstākas kārtas koncentrētas spektrālā blīvuma jaudas bez fāzes atsaistīšanas algoritmu lietošanas, bet risinot lineāru vienādojumu sistēmu.

### c. Neminimuma fāžu sistēmu noteikšana

Vienādojums 17 demonstrē, ja mums ir pieeja abiem – sistēmas ievadam un izvadam, tad mēs varam rekonstruēt tās IR,  $h(k)$ , lietojot šķērs-korelācijas funkciju  $r_{vz}(\tau)$ . Dažās signālu apstrādes aplikācijās (piem., nolīdzināšanas, dekonvolūcijas atstarojumu seismoloģijā) mums ir tikai pieeja sistēmas izvadam. Vai ir iespējams noteikt  $h(k)$  (vai mērogotu  $h(k)$  versiju) tikai no izvada mērījumiem? Ja signāli nav Gausa signāli, tad tas ir patiešām iespējams rekonstruēt pareizas fāzes IR tikai no izvada datiem, lietojot augstākas kārtas statistiku. Viņu sākotnējā publikācija ir radījusi lielu skaitu jaunas metodes, kā noteikt neminimuma fāžu sistēmas tikai no trokšņainiem izvada mērījumiem.

**Atsaucoties uz zīm.4 problēma ir: doti laikā ierobežotu trokšņainu mērījumu  $Z(k)$ ,  $k = 1, 2, \dots, N$  novērtējuma  $H(z)$  parametri, kad  $H(z)$  ir tāds, kā parādīts zīm.5, t.i.,**

$$H(z) = B(z)/A(z) = \left[ \frac{\sum_{j=0}^q b_j z^{-j}}{\sum_{i=0}^p a_i z^{-i}} \right] \quad (58)$$

**Parametri, kuri ir jānosaka ir  $b(1), \dots, b(q)$ ,  $a(1), \dots, a(p)$ ;  $a(0) = b(0) = 1$  mērogošanas nolūkiem un pakāpes  $p$  un  $q$  tiek uzskatītas par zināmām.**

**Diskusijai par daudz reālāku gadījumu, kad pakāpes  $p$  un  $q$  arī ir jānosaka.**

Šī izvada mērījumu noteikšanas problēma parādās daudzās nozarēs, ieskaitot komunikācijas un atstarojumu seismoloģiju. Saskaņā ar modeli,  $V(k)$  ir “ziņojums”,  $h(k)$  ir “kanāls” un  $Z(k)$  ir “izkropļots ziņojums” (izkropļots pateicoties starpsimbolu interferencei). Precīzs kanāla modelis ir vajadzīgs, lai varētu atjaunot ziņojumu uztvērējā. Šis modelis tiek lietots daudzās nolīdzināšanas shēmās. Atstarojumu seismoloģijā  $V(k)$  ir Zemes “atstarojumu secība” (t.i., Zemes “ziņojums”),  $h(k)$  ir “seismiskais avota vilnis” un  $Y(k)$  ir “seismogramma”. Precīzs avota viļņa modelis ir nepieciešams, lai novērtētu Zemes atstarošanās spēju caur dekonvolūciju.

Kad skaitītāja parametri vien.58 visi ir vienādi ar nulli, izņemot  $b(0)$ , mums ir visu polu modelis, kura gadījumā mēs esam AR parametru novērtējuma sfērā. Kad visi saucēja parametri ir vienādi ar nulli, izņemot  $a(0)$ , mēs esam MA parametru novērtējuma sfērā. Šie divi speciālie gadījumi ir bijuši plaši izpētīti ne tikai viņu pašu interesēs, bet arī tāpēc,

ka dažas metodes ARMA parametru novērtēšanai turpinās divos soļos, vispirms novērtējot AR parametrus ARMA modelim un tad novērtējot MA parametrus ARMA modelim, izmantojot tikko noteiktos AR parametrus.

Kā atgādinājums, mēs lietojam augstākas kārtas statistiku, lai atrisinātu šīs identifikācijas problēmas, tāpēc ka otrās kārtas statistika ir nejūtīga uz fāzi [tādē], viņi var dot tikai izvirzījumu minimuma vai maksimuma fāzes modeļiem, t.i., SEMP modeļiem] un augstākas kārtas statistika ir nejūtīga pret pieaugošu Gausa troksni.

Šeit demonstrē drīzāk atzīmējamu rezultātu, ka ir iespējams noteikt parametrus MA modelim slēgtā formā. Nebija nekādu rezultātu šajā sakarā pirms augstākas kārtas statistikas ieviešanas signālu apstrādē.

*Piemērs 3.* Šis ir turpinājums Piemēram 1. Agrāk, mēs rādījām, ka ja mums ir pieeja abiem – sistēmas ievadam un izvadam, tad mēs varam rekonstruēt pareizas fāzes IR,  $h(k)$ , lietojot vien.17. Vai šeit ir salīdzināms rezultāts ar vienkāršu slēgtas formas formulu, rekonstruējot pareizas fāzes IR,  $h(k)$  lietojot tikai izvada mērījumus?

q-tās pakāpes MA sistēmas IR var tikt izrēķināts tikai no sistēmas izvada koncentrētās spektrālā blīvuma jaudas, lietojot sekojošu vienkāršu slēgtas formas formulu [apgalvots šeit attiecībā uz trešās pakāpes koncentrētu spektrālā blīvuma jaudu; vispārīgākus patvaļīgas pakāpes koncentrētai spektrālā blīvuma jaudai var atrast vien.13].

$$h(k) = c_3^2(q, k) c_3^2(q, 0) \quad k = 0, 1, \dots, q \quad (59)$$

Pierādījums šim interesantajam rezultātam ir dots Tabulā 6.

Tabula 6.  $C(q, k)$  formulas atvasinājums

Šeit mēs atvasinām  $C(q, k)$  formulu  $c_3^2(q, 0) = \gamma_3^2 h(q)$  blīvuma jaudai. Mēs sākam ar (Vien.48) ņemot  $k = 3$ , t.i.,

$$c_3^2(\tau_1, c_3^2(q, k) = \gamma_3^2 h(q) h(k) h(k + \tau_2) \quad (1)$$

kurā  $h(0) = 1$ , normalizācijas mērķiem. Uzstādam  $\tau_1 = q$  un  $\tau_2 = k$  vien.(1) un izmantojam faktu, ka MA(q) sistēmai  $h(j) = 0 \quad j > q$ , tas izskatās

$$(2)$$

Nākošais, uzstādam  $\tau_1 = q$  un  $\tau_2 = 0$  vien.(1), tas izskatās

$$(3)$$

Dalot (2) ar (3) mēs iegūstam  $C(q, k)$  formulu (Vien.59).

Uz vienādojumu 59 bieži atsaucas kā uz "**C(q, k) formulu**". Šeit piedāvā neprecīzu šīs formulas atvasinājumu priekš 1D un 2D nepārtrauktiem laika procesiem. Piezīmējiet, ka vien.59 izmanto tikai 1D daļu no izvada koncentrētās spektrālā blīvuma jaudas gar labo pusi tumši ēnotam labajam trijstūrim zīm.8. Piezīmējiet arī, ka tam nepieciešams precīzi zināt MA pakāpi q. No teorētiskā redzes viedokļa tas ir interesanti un svarīgi, bet nepraktiski no reālā skaitļošanas redzes viedokļa. Tas ir tāpēc, ka praksē izvada koncentrētai spektrālā blīvuma jaudai ir jābūt novērtētai un vien.59 nedod nekādu filtrēšanu, lai samazinātu koncentrētās spektrālā blīvuma jaudas novērtēšanas kļūdas. Par laimi, citas, uz koncentrētu spektrālā blīvuma jaudu bāzētas metodes ir tikušas izstrādātas, kas piedāvā šādu filtrēšanu.

### Nelineāri procesi

**Augstākas kārtas monemtāns un kumulatīvs spektrs vai polispektrs dod iespēju noteikt un novērtēt nelinearitāti stohastiskajos signālos.** Šie stohastiskie signāli parasti rodas tad, kad nelineāra sistēma operē ar gadījuma ievad signāliem. Kopēja mijiedarbība patvaļīgiem, nekustīgiem, gadījuma datiem, kuri iet caur patvaļīgu lineāru sistēmu tika diezgan intensīvi izpētīta pēdējos gados. Principā, visas tās mijiedarbības pamatojas uz autokorelāciju, spektra pakāpi un kross-korelāciju. No citas puses, kopējas mijiedarbības neiespējamas patvaļīgiem, nekustīgiem, gadījuma datiem, kuri iet caur patvaļīgu lineāru sistēmu. Tā vietā, katrs nelinearitātes veids tika izpētīts kā atsevišķs gadījums. Polispektri var spēlēt galveno lomu sistēmā nelinearitātes tipa noteikšanā un raksturošanā pa tās izejas secību. Bez tam kross-polispektri var tikt izmantoti pie nelineāro sistēmu identifikācijas pie ievad un izvad datu novērošanas.

Praksē gadās situācijas, kad divu harmonisko komponentu mijiedarbība ietekme jaudas palielināšanu vai frekvenču starpību. Piemēram, paņemsim signālu

$$X(k) = A_1 \cos(\lambda_1 k + \theta_1) + A_2 \cos(\lambda_2 k + \theta_2),$$

kurš iet caur sekojošu, parastu, kvadrātisku, nelineāru sistēmu:

$$Z(k) = X(k) + \varepsilon X^2(k); \quad (60)$$

kur  $\varepsilon$  - konstante, kas nav vienāda ar 0. Signāls  $Z(k)$  satur kosinusoidālās

attiecības apgabalos  $(\lambda_1, \theta_1)$ ,  $(\lambda_2, \theta_2)$ ,  $(2\lambda_1, 2\theta_1)$ ,  $(2\lambda_2, 2\theta_2)$ ,  $(\lambda_1 + \lambda_2, \theta_1 + \theta_2)$  un  $(\lambda_1 - \lambda_2, \theta_1 - \theta_2)$ . Šis fenomens, kurš noved pie dažu vai visu fāžu attiecību palielināšanu, kuras ir tieši tādas pašas kā frekvenču attiecības saucās kvadrātiska fāžu saite.

Pie konkrētas pielietošanas vajag noskaidrot vai harmoniski saistīto pozīciju maksimumi faktiski ir fāžu saistīti jaudas spektrā. Kad jaudas spektrs samazina visas fāžu saites – uz šo jautājumu atbildēt nevar. Trešās kārtas komulantes (bispektrs), kā zināms, ir spējīgas noteikt un noraksturot kvadrātisko fāžu saiti.

Izrēķināsim procesu

$$X(k) = \sum_{i=1}^6 \cos(\lambda_i k + \varphi_i) \quad (61)$$

kur  $\lambda_2 > \lambda_1 > 0$ ,  $\lambda_5 > \lambda_4 > 0$ ,  $\lambda_3 = \lambda_1 + \lambda_2$ ,  $\lambda_6 = \lambda_4 > \lambda_5$ ,  $\varphi_1, \varphi_2, \dots, \varphi_3$  – neatkarīgas, vienmērīgi novietotas gadījuma vērtības intervālā  $(0, 2\pi)$ , un  $\varphi_6 = \varphi_4 + \varphi_5$ . Formulā 61, tajā laikā kad  $\lambda_1, \lambda_2, \lambda_3$  un  $\lambda_4, \lambda_5, \lambda_6$  atrodas uz harmoniski saistītām pozīcijām, vienīgā komponente pie  $\lambda_6$  ir fāžu saites rezultāts starp komponentēm pie  $\lambda_4$  un  $\lambda_5$ , papildus komponente  $\lambda_3$  ir neatkarīga harmoniska komponente. Procesa jaudas spektrs sastāv no impulsiem pie  $\lambda_i$ ,  $i = 1, 2, \dots, 6$ , kā pārādīts 11.zīm. Paskatoties uz spektru uzreiz nevar pateikt vai harmoniski saistītie komponenti faktiski ir iekļauti kvadrātisko fāžu savienošanas saitēs. Trešās pakāpes komulante  $c^x_3(n, 1)$  no  $X(k)$  var viegli iegūt no

$$c^x_3(n, \ell) = \frac{1}{4} \{ \cos(\lambda_5 n + \lambda_4 \ell) + \cos(\lambda_6 n + \lambda_4 \ell) + \cos(\lambda_4 n + \lambda_5 \ell) + \cos(\lambda_6 n - \lambda_5 \ell) + \cos(\lambda_4 n - \lambda_6 \ell) + \cos(\lambda_5 n - \lambda_6 \ell) \} \quad (62)$$

Vajag atzīmēt, ka formulā 62 parādas tikai fāžu savienošanas komponentes. Tāad bispektrs, kurš ir aprēķināts pirmā kvadrātā trīsstūra rajonā, zīm.7. rāda ka tikai impulss pie  $(\lambda_5, \lambda_4)$  ir fāžu saite. Ja nav fāžu saites trešās pakāpes, tad komulante un arī bispektrs ir vienādi ar 0. Tāpēc fakts kā tikai fāžu saites komponentes iekļautas trešās pakāpes komulantē, rāda ka bispektrs ir ļoti svarīgs instruments kvadrātisko fāžu saišu novērošanai.

Jebkura no esošajām bispektra novērošanas metodēm var tikt izmantota kvadrātisko fāžu saišu analīzei. Parasti, katrai no šīm metodēm ir savas priekšrocības un ierobežojumi. Piemēram, nosacījuma metodes labāk izmantot fāžu saites kvantitatīvai novērtēšanai, bet parametriskas metodes (AR un ARMA) labāk izmantot ka detektoru.

Nosacīto metožu izmantošana bispektra novērtīšanai kopā ar bikoherento indeksu  $|P^x_3(\omega_1, \omega_2)|$  ir saistīta ar kvadrātisko fāžu saišu novērošanu un kvantitatīvu novērtēšanu. Kad bikoherents indekss pieņem vērtību tuvāko vieniniekam, to rāda gandrīz 100% fāžu saišu pakāpi. No otras puses  $|P^x_3(\omega_1, \omega_2)|$  tuvāka nullei, rāda fāžu saišu neesamību.

Viena no metodikas priekšrocībām ir iespēja izmantot to labai kvantitatīvai novērtēšanai, kura dod fāžu saišu pakāpes precīzus rezultātus. Piemēram, nosacīta bikoherentā novērtējums signālam

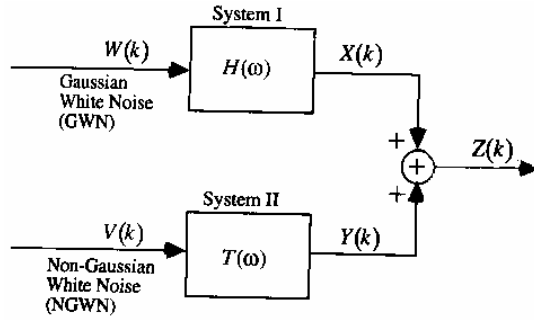
$$X(k) = \sum_{i=1}^4 \cos(\lambda_i k + \varphi_i) \quad (63)$$

kur  $\lambda_3 = \lambda_1 + \lambda_2$ ,  $\varphi_3 = \varphi_1 + \varphi_2$ ,  $\lambda_4 = \lambda_1 + \lambda_2$  un  $\varphi_1, \varphi_2, \dots, \varphi_4$  - gadījuma vērtības, kuras vienlīdzīgi sadalītas intervālā  $[0, 2\pi]$ , parādīs maksimums pie  $(\lambda_1, \lambda_2)$  ar vērtību piemēram 0,5, kas rāda tikai 50% fāžu koherences pakāpes. Iespējams ierobežojums nosacījuma metožu izmantošanā, kvadrātiskas fāžu saites analīzei ir augsta bispektra novērtēšanas variācijas pakāpe un zema izrēķināšanas kvalitāte, kad harmoniski saistīti fāžu pāri atrodas tuvu viens no otra. Vajag atzīmēt, ka efekts atkarībā no šiem aprobežojumiem samazinās ar datu segmenta daudzuma palielināšanu.

Kvadrātiska fāžu saite, izmantojot AR modeļus bispektru novērtēšanai. Kā motīvs AR modeļu izmantošanai var kalpot augstas precizitātes pakāpes aprēķini un šo rezultātu zema variācijas pakāpe. Mēs nevaram pielietot tikai vienu datu segmentu (t.i. tikai viens nemainīgo vērtību kopums priekš  $\varphi_1, \varphi_2, \varphi_3, \varphi_4$  formulā 63), kvadrātiskai fāžu saites noteikšanai starp sinusoīdu pāriem. Datu segmentācijas reģistrēšana vajadzīga atbilstošu novērtījumu iegūšanai trešās pakāpes komulantēm. AR metodes priekšrocības parādas situācijās, kur nosacījuma metodes praktiski nevar aprēķināt tuvu izvietotus fāžu saites. No citas puses, AR modeļu izmantošanas ierobežojumi saistīti ar to, ka nevar precīzi novērtēt fāžu koherences pakāpi, kad fāžu saite ir harmoniski saistīto fāžu pāru rezultāts.

Šeit ir parādīts, ka trisspektrs var tikt izmantots kubiskas fāžu saites aprēķināšanai un, ja signāls sastāv kā no kvadrātiskas, tā arī no kubiskas fāžu saites, tad šī signāla bispektrs nav redzams kubiski saistītām komponentēm, un izrēķināt kvadrātiski saistītas komponentes var tad, kad šī signāla trisspektrs nav redzams kvadrātiski saistītām komponentēm un var aprēķināt kubiski saistītas komponentes. Tāad, augstākas kārtas spektri var tikt izmantoti dažādu nelinearitātes veidu aprēķiniem.





12. Signāls  $Z(k)$ , kura spektra un bispektra jauda modelējas ar dažādām lineārām sistēmām. Procesi  $W(k)$  un  $V(k)$  aprēķinās neatkarīgi.

Lineārais modelis, kurš paskaidro signāla bispektru parasti ir atšķirīgs no lineāra modeļa, kurš paskaidro jaudas spektru. Aprēķināsim procesu  $Z(k)$ , uz zīm. 12. Var redzēt, ka  $Z(k)$  - divu signālu summa, viens ir lineāras sistēmas izeja, kurš izmanto "balto Gausa skaņu"  $\{W(k)\}$  un cits - lineāras sistēmas izeja, kurš neizmanto "balto Gausa skaņu"  $\{V(k)\}$ . Tālāk  $\{V(k)\}$  un  $\{W(k)\}$  ir neatkarīgi, un tas nozīmē, ka  $X(k)$  un  $Y(k)$  arī statistiski neatkarīgas; līdz ar to, signāla bispektrs  $Z(k)$  ir signāla bispektra summa  $X(k)$  un  $Y(k)$ . tā kā  $X(k)$  "Gausa", tā bispektrs ir vienāds ar 0 un tātad bispektrs  $Z(k)$  ir bispektrs  $Y(k)$ , kurš ir lineāras sistēmas rezultāts. Sistēma II zīm. 12 modelē bispektru  $Z(k)$ , citiem vārdiem

$$C_3^x(\omega_1, \omega_2) = \tag{64}$$

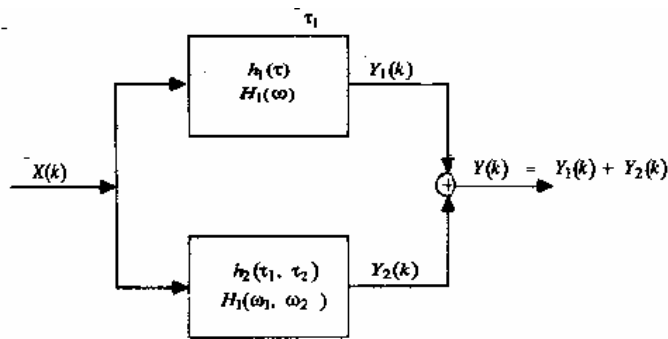
$$C_3^y(\omega_1, \omega_2) = \gamma_3^y T(\omega_1) T(\omega_2) T^*(\omega_1 + \omega_2).$$

Signāla  $Z(k)$  jaudas spektrs ir  $X(k)$  un  $Y(k)$  jaudas spektru summa. Kaut gan signāla  $X(k)$  iedarbe ir ieverojama, otra sistēma nemodelē jaudas spektru. Mūsu gadījumā:

$$C_2^z(\omega) = C_2^x(\omega) + C_2^y(\omega) = \gamma_2^x |H(\omega)|^2 + \gamma_2^y |T(\omega)|^2. \tag{65}$$

Ir skaidrs, ka ja  $T(\omega)$  un  $H(\omega)$  piedāvā divus dažādus AR modeļus, signāla  $Z(k)$  jaudas spektrs ir ARMA, bet bispektrs - AR veida.

Nelineāri procesi var arī tikt apskatīti kā kopēja Voltera sistēma. Šeit mēs apskatīsim tikai populāru otras kārtas Voltera modeli.



13. Otrās kārtas Voltera sistēma: lineāra un kvadrātiska daļas paralēlaja struktūrā.

Pieņemsim, signāls ir apskatīts kā otras kārtas Voltera modelis.

$$Y(k) = \sum_{\tau_1} h_1(\tau_1) X(k - \tau_1) + \sum_{\tau_1} \sum_{\tau_2} h_2(\tau_1, \tau_2) X(k - \tau_1) X(k - \tau_2) \tag{66}$$

kur  $X(k)$  - stacionārs gadījuma process ar vērtību 0. Problēmas apraksts: novērtēt impulsa reakciju,  $h_1(\tau)$  un kodols  $h_2(\tau_1, \tau_2)$ . Piezīmēsim, ka formula 61 var tikt apskatīta ka paralels lineāras sistēmas  $\{h_1(\tau)\}$  un kvadrātiskas sistēmas  $\{h_2(\tau_1, \tau_2)\}$  slēgums, kā parādīts zīm. 13. Atzīmēsim, ka ieejas signāls  $X(k)$  - stacionārs Gausa process ar vērtību 0. Šeit parādīta sekojoša fundamentālā saite.

$$H_1(\omega) = \frac{C_{xy}(\omega)}{C_x^2(\omega)} \quad (67)$$

un

$$H_2(-\omega_1, -\omega_2) = \frac{C_{xy}(\omega_1, \omega_2)}{2C_x^2(\omega_1)C_x^2(\omega_2)} \quad (68)$$

kur  $C_{xy}(\omega_1)$ ,  $C_{xy}(\omega_1, \omega_2)$  - krosspektrs un atbilstošs krosbispektrs. Tātad, mums ir sistēma ieeja un izeja, zīm. 13, un formulas 67 un 68 piedāvā sistēmas matemātisko vienādojumu. Svarīgi atcereties, ka formulam 67 un 68 ir jēga tikai Gausa ieejas signālam.

## Pielietošana

Pēdējos 25 gados komulantes un polispektri iegūva plašu pielietošanu. Pielietošanu var sadalīt trijos galvenajos kategorijās: saistīti ar fiziku, 1D in 2D/3D. **Pielietojumus, kas saistīti ar fiziku** izmanto novērtējot: okeānu (iekšēju viļņi, troksni, viļņus dziļuma aprēķināšanai, viļņu mijiedarbe, viļņu saite, viļņu paradības, viļņu pretestība kuģiem un jūras virsmas temperatūras anomālijas), zemi (brīvas svārstības), atmosfēru (spiedienu turbulenci), starplanētu novērošanu (stencilācija), vēju (turbulence, plūsma), plazmas (viļņu mijiedarbība, nelineāras paradības), elektromagnētisms (zemfrekvences dati), kristalogrāfija (struktūra). **ID (identifikācija) pielietošana** operē ar: diagnostika (virsmas gludums, mehānismu defekti, mehānisko sistēmu trokšņi), vibrācijas analīze (signāla veida noteikšana, kvantitatīvais novērtējums, sitiena noteikšana), runa (skaņas augstuma noteikšana, runas/nerunas aprēķini), troksnis (novēršana, kuģu radiolokācija, dzinēju troksnis, bioelektrība), sistēmas identifikācija (kanāli ar maksimālo fāzi, ieeja/izeja), reģistrācija, fāzes stabilizācija, FM signāli, seismiska dekonvolūcija, avota diapazons, pazūšanu novēršana, ekonomiskas laika rindas, smadzeņu skanēšana (viļņu saite uz elektroencefologrammas) un bioloģiskie ritmi. Tie operē arī ar plašu problēmu loku, kas saistītas ar harmonisko atjaunošanu, nelineārām sistēmām (Voltera, bilineāri, fāžu saite), masīvu apstrāde, eholots, radars. **2D/3D pielietojumi** saistīti ar attēliem (modelēšana, konstruēšana, atjaunošana, kodēšana, kustības novērtēšana, secības analīze), tekstura (izšķirtspēja), tomogrāfija (3D atruma apgabals), ultraskaņas atstarojuma inversā filtrācija, fotonno ierobežoto datu attēlojums. Šinī daļā mēs dosim īsus piemērus, ka augstākas kārtas spektrus pielieto sekojošās nozarēs: masīva apstrāde, klasifikācija, harmoniska atjaunošana laika aiztures novērtēšana, pazūšanu novēršana un interferences anulēšana.

### a. Masīva apstrāde

Massīva apstrāde spēlē ļoti nozīmīgu lomu signāla uzlabošanā pie interferences. Massīva apstrādes problēma satur: pieņemšanas virziens (DOA), avotu skaitu noteikšana, radiosignāla formēšana, signāla avota novērtēšana, avota klasifikācija, sensoru kalibrēšana utt.

Tik daudz jaunu un interesantu masīvu apstrādes algoritmu parādījās pēdējo 10 gadu laikā. Priekš ka tad ir vajadzīga augstākas pakāpes statistisko datu pielietošana masīvu apstrādes problēmu risināšanai? Ir dažas atbildes uz šo jautājumu, ieskaitot: (1) viens no vispopulāriem un svarīgiem avotiem, kas formē radiosignālu, saucamais kapona radiosignāla formētais ar minimālo dispersijas izkropļojumu (MVDR), kurš bija signāla uzlabošanas sakuma punkts un plaša DOA izmantošanas apgabals, prasa ļoti specifisku un detalizētu informāciju par tā saucamo masīva vadības vektoru (piemēram, avota vadība rakurs, masīva ģeometrija, uztverēja raksturojums), informāciju, kura bieži nav pieejama vai pieejama, bet ar nepietiekamu precizitātes pakāpi; tehnika, kura ir aplūkota zemāk, parāda, ka komulanti var tikt izmantoti nezināma vadības vektora novērtēšanai, pēc kā var izmantot MVDR radiosignāla formētāju; (2) kad additīvs troksnis ir krāsains vai Gausa, otras kārtas statistika, kas balstās uz DOA algoritmu ar augstu paplašinājumu, tādu, kā mūzika neizpilda savas funkcijas pietiekami labi; tajā laikā kad mūzikas algoritms, kas balstīts uz komulanti izpilda: (3) otras kārtas statistisko datu vairākums, kura pamatā strādā radiosignāla formētāji, pieņem, ka iegūtais signāls ir nekoherents, kas neietilpst tāda svarīga elementa kā daudzstaru sadalīšana noteikumos, radiosignāla formētāji, kas balstas uz komulantēm var strādāt ar daudzstaru sadalīšanu. Šeit tiek izmantoti iegūta signāla komulanti vadības vektora novēršanai priekš šaurjoslas ne Gausa vajadzīgo signālu pie nosacījuma ka ir novirzīts Gausa traucējumu avots ar nezināmo kovariācijas struktūru. Tas nozīmē, ka nezinām masīvu dažveidību vai DOA informāciju par prasāmo signālu.

Prasāmais signāls var tikt izteikts kā runas eholots radarā vai komunikācijas signāls. Piemērs, kas parādīts tālāk, apskata komunikatīvo scēnu un prasa ceturtās kārtas komulantes izmantošanu, tā kā trešās kārtas komulante komunikatīva signāla vienāda ar 0.

Izrēķināsim masīvu no  $M$  elementiem ar devēju reakciju gadījuma raksturojumu un to izvietojumiem. Pieņemsim, ka signāls  $\{i_j(t), j = 1, 2, \dots, J\}$  - interferenciāls  $J$  Gausa signāls, un ne Gausa prasāmais signāls -  $d(t)$  ar frekvenci  $\omega_0$ . Pieņemsim, ka avoti atrodas lielā attālumā no masīva, tāpēc ir iespējama plakana viļņu fronta aproksimācija. Pieņemsim arī, ka additīvs troksnis ir Gausa troksnis ar nezināmo kovarnāciju. Tādējādi, aprēķinu  $\mathbf{M}$  var sagrupēt sekojošā modelī.

$$\underline{r}(t) = \underline{a}(\theta_d)d(t) + A_1(\theta)\underline{i}(t) + \underline{n}(t) \quad (69)$$

Kur  $\theta_d$  - prasāma signāla DOA;  $\underline{a}(\theta_d)$  - masīva vadības vektors prasāmajam signālam;  $A_1(\theta)$  - masīva vadības matrica interferences avotiem  $\underline{i}(t)$ ;  $\theta$  -  $J \times 1$  vektors DOA trokšņu avotiem;  $\underline{r}(t)$  -  $M \times 1$  iegūto signālu vektors, un  $\underline{n}(t)$  -  $M \times 1$  Gausa trokšņa vektors. Kā  $R$  mēs apzīmējam kovariācijas matricu priekš  $\underline{r}(t)$ . Izeja MVDR formētāja signāla var tikt izteikta kā

$$y(t) = \underline{w}^H \underline{r}(t) = \left[ \beta_1 R^{-1} \underline{a}(\theta_d) \right]^H \underline{r}(t) \quad (70)$$

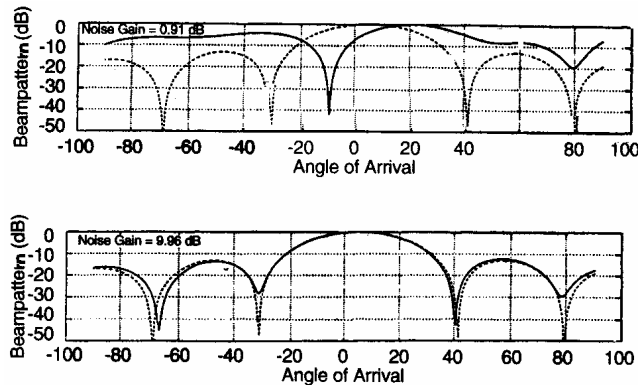
kur konstanta  $\beta_1$  atbilst atbildei uz prasāmo signālu un  $\underline{w}$  rāda procesa svara vektoru. Skaidrs, ka MVDR radiosignāla formētājs prasa  $\underline{a}(\theta_d)$  vērtības. Nezinot masīva sastāvdaļas nav iespējams atrast  $\underline{a}(\theta_d)$  arī gadījuma, ja ir zināma vērtība ( $\theta_d$ ). Aprēķināsim sekojošo ceturtās kārtas komulantes vektoru  $\underline{c} = \text{col}[c_1, c_2, \dots, c_M]$ , kur  $c_j = \text{cum}\{r(t), r^H_1(t), r^H_1(t), r_j(t)\}$ ,  $j = 1, 2, \dots, M$ . Zinot komulantu vērtības, kuras dotas 2.tab un uztvērēja modeli (formula 69) parādīsim, ka

$$\underline{c} = \beta_2 \underline{a}(\theta_d) \quad (71)$$

kur  $\beta_2$  - cita mēroga faktors. Izmantojot šo rezultātu, tagad ir iespēja atrast  $\underline{a}(\theta_d)$  no esošajiem datiem, izmantojot ceturtās kārtas komulanti. Atzīmēsim, ka  $C$  ir prasāma signāla vadības vektora projekcija uz mēroga faktoru. Samainot formulu 71 uz 70, mēs dabūsim sekojošo MVDR radiosignāla formētāju, kas balstīts uz komulantu:

$$y(t) = \underline{w}_{\text{cum}}^H \underline{r}(t) = \left[ \beta_3 R^{-1} \underline{c} \right]^H \underline{r}(t) \quad (72)$$

kur  $\beta_3 = \left( \underline{c}^H R^{-1} \underline{c} \right)^{-1}$



14. Procesora radiosignāla un balta trokšņa piemērs vienīgā realizācija priekš  $SNR = 20\text{dB}$  a) MVDR  
b) MVDR, kas balstīts iz komulantu

Daudzi modelēšanas rezultāti doti ne tikai nekoherentiem avotiem, bet arī koherentiem gadījumiem (kas prasa neieverojamu modeļa izmaiņu formulā 69). Šeit mēs piedāvāsim rezultātus vienkāršotai scenārijai. Masīvs ir lineārs ar 10 vienādas formas izotronu elementiem ar vienādām viļņu formas pusēm. Rakstāmais garums = 1000 kadri. Šo masīvu izmanto optimālai BPSk signāla uztveršanai, kurš attēlo masīvu piecugrādu skalā laika un telpas baltā trokšņa klātbūtnē, kurš vienāds jaudai, cirkulāri simetriska trokšņa devējam, kura  $SNR = 20\text{dB}$ . Zīm. 14 salīdzina MVDR radiosignāla formētāju (kurš piesavina DOA primāro vērtību) un pamatots uz radiosignāla formētāja MVDR komulantes. Uzbūvēsim radiosignāla formētāja vērtību  $P(\theta) = |\underline{w}^H \underline{a}(\theta_d)|^2$ . Arī grafikā redzami baltā

trokšņa vērtības. Visas vērtības normāli sadalītas ar maksimālo vērtību 0 dB. Salīdzināšanai, radiosignāla formētāja optimālas vērtības, kuri domāti precīzo statistisko datu izmantošanai un vadības vektora informācijai formulā 70, attēloti zīmējuma punktier līnijas veidā. Atzīmēsim, ka pat tad, kad radiosignāla formētājs piedāvā vadības vektora informācijas veida, viņš neattēlo pilno informāciju tā, kā to dara radiosignāla formētājs, kurš balstīts uz komulantes. Radiosignāla formētājs, kurš balstīts uz komulantes piedāvā lielisko informāciju pat bez DOA avota zināšanas. Atzīmēsim arī, ka priekš 100 Monte-Karlo darbībām matemātiskas gaidīšanas vērtība un standartnovirziens baltā trokšņa vērtībai radiosignāla formētājam  $MVDR = 0,179 \pm 1,360$ , tad kad salīdzināšanai, radiosignāla formētājam kas balstas uz MVDR komulantes  $= 9,954 \pm 0,015$ .

Teorētiski - trokšņa vērtība šīm scenārijam = 10.

Līdz šim bija parādīts, ka masīva simbolu apstrādei ir 2 papild iemesli vairāk augstākas kārtas spektru izmantošanai: (1) HOS var palielināt masīva diafragmas efektivitāti, un (2) HOS var novērst ne tikai additīva Gausa trokšņa efektu, bet arī var novērst additīva ne Gausa trokšņa efektu. Tas viss beidzās ar kroskorelācijas aprēķinu, kurā ir vajadzīga augstākas kārtas tehnoloģijām, kuras izmanto ceturtās kārtas komulantes.

### b. Klasifikācija

Modeļu vai signālu klasifikācija var tikt veikta tieši ar modeļu vai signālu izvēli, vai ar atribūtiem, kas ir ar tiem saistīti. Grūti strādāt ar additīvo, krāsu Gausa troksni ar parasto pieeju palīdzību. Jauna pieeja, kura neapskata additīvo krāsu vai balto Gausa troksni, strāda ar komulantes vai polispektra vektoru un paplašina klasifikāciju, kas balstīta uz korelācijas līdz klasifikācijas, balstītas uz HOS. Tas pamatojas uz svarīga fakta, ka komulantu un polispektra novērtējums ir asimetrisks Gausa troksnim. Tādējādi, to var izteikt ar vienādību:

$$\text{Novērtējums HOS} = \text{HOS} + \text{kļūda} \quad (73)$$

kur "kļūda" - asimetriskais Gausa troksnis. Tas paplašina tradicionālo klasifikāciju vai noteikšanas procedūru. Līdz šai formulai darbs ar augstākas kārtas statistisko datu vektoru un sava pamata izmanto atribūtus, kuri ir iegūti no sākotnejiem datiem. Tādējādi, augstākas kārtas statistiskie dati pieļauj jaunus atribūtus, izmantojamus modeļu un signālu klasifikācijas problēmas.

### c. Harmoniska atgūšana

Dažu harmoniku, kā arī šo harmoniku frekvenču un amplitūdu novērtējums, trokšņa novērtējumam bieži sastopas dažādos signāla apstrādes pielietojumos, tādos kā ieejas virziena novērtējums signāliem ar avota lineāro masīvu ar šauru frekvences diapazonu, un harmoniskas atgūšanas problēmas risināšanai. Pēdējā uzdevuma reķināšanai sāksim ar modeli:

$$Y(k) = \sum_{j=1}^p a_j \cos(k\omega_j + \phi_j) + N(k) = X(k) + N(k) \quad (74)$$

kur  $\phi_j$  apzīmē gadījuma fāzes, kuri ir vienādi un kuri vienmērīgi sadalīti intervālā  $[0, 2\pi]$ ;  $\omega_j$  - nezināmas determinētas frekvences; un  $a_j$  - nezināmas, determinētas amplitūdas. Additīvs troksnis  $N(k)$ . Pieņemsim, balts vai krāsains Gausa troksnis ar nezināmo spektra blīvumu. Uzdevums ir novērtēt signālu  $p$  daudzumu, stūru frekvences  $\omega_j$  un amplitūdu  $a_j$ . Uzdevums tika plaši apskatīts. Kad additīvs troksnis ir ilgs, tad augstas izšķirtspējas metodes, balstītas uz otrās kārtas statistiskiem datiem, tādiem kā mūzika vai minimālā norma kombinējas ar SVD ( $p$  noteikšanai), kas dod labus rezultātus. Kad troksnis ir nokrāsots, šīs metodes nestrādā. Tiem ir tendence pārvērtēt sinusoīdu daudzumu, pieņemot nokrāsoto troksni par papild sinusoīdu. Augstākas kārtas statistiskiem datiem nav problēmu ar jebkura veida Gausa troksni; tāpēc tie tika izmantoti šā uzdevuma risināšanai.

Trešās kārtas komulante priekš  $Y(k) = 0$ , tāpēc šim pielikumam nepieciešami izmantot ceturtās kārtas komulanti. Komulante priekš  $Y(k)$  ceturtās kārtas vispārīgā gadījumā ir trīs laika intervālu funkcija, kā likums šīs komulantes diagonāls nogriezum:

$$c_4^y(\tau, \tau, \tau) \triangleq c_4^y(\tau) = -\frac{3}{8} \sum_{j=1}^p a_j^4 \cos(\omega_j \tau) \quad (75)$$

Labi zināms, ka  $Y(k)$  autokorelācija ir:

$$c_2^y(\tau) = \frac{1}{2} \sum_{j=1}^p a_j^2 \cos(\omega_j \tau)$$

(76)

Salīdzinot vienādojumus 75 un 76 var redzēt, ka (izņemot mēroga faktora starpību)  $c_{yy}^y(\tau)$  var tikt attēlota ka sekojoša signāla autokorelācijas funkcija, kurš saistīts ar  $Y(k)$ .

$$Y_1(k) = \sum_{j=1}^p a_j^2 \cos(k\omega_j + \phi_j) + N(k) \quad (77)$$

Tas nozīmē, ka esošie augstās izšķirtspējas metodes, tādas kā mūzika un minimālā norma, var tikt pielietota izmainot korelācijas lielumu uz lielumu  $c_{yy}^y(\tau)$ . Var apskatīt detalizēti, kā tas viss notiek: vispusīgai metožu salīdzināšanai, kuri balstīti uz korelācijas un komulantes, harmoniku daudzuma noteikšanai, kad harmoniku amplitūda samazinās, kad vienas harmonikas frekvence pielīdzinās citām, un kad atšķirīgs datu daudzums izmantojas diviem nokrāsotā Gausa trokšņa harmonikām.

#### d. Laika aiztures novērtējums

Viens no svarīgiem laika aiztures novērtējuma metožu - dzirdēšanas un izvietošanas apgabala aprēķins. Pieņemsim, ka  $[X(k)]$  un  $[Y(k)]$  - ir divi telpiski sadalīti devēji, kuri atbilst vienādojumiem.

$$X(k) = S(k) + W_1(k) \quad (78)$$

$$Y(k) = AS(k - D) + W_2(k) \quad (78b)$$

kur  $\{S(k)\}$  - nezināms signāls,  $\{S(k-D)\}$  - pārnesta, un iespējams  $\{S(k)\}$  un  $\{W_1(k)\}$ ,  $\{W_2(k)\}$  - nezināmi skaņas avoti. Uzdevums ir novērtēt laika aizturi  $D$  pēc devējiem  $X(k)$  un  $Y(k)$ . Tādas situācijas rodas tādos pielietojumos, kā eholoti, radāri, biomedicīna, geofizika utt. Laika aiztures rēķināšanas pamatpielietojums -  $\{X(k)\}$  nobīdīšana attiecībā pret  $\{Y(k)\}$ , un līdzības apskatīšana starp tiem. Labāka pozīcija rodas pie  $D$  nobīdīšanas. Pie signāla apstrādes "līdzību apskatīšana" pariet "kroskorelācijas apskatīšana" starp  $\{X(k)\}$  un  $\{Y(k)\}$ .

$$\begin{aligned} c_{xy}(\tau) &= E\{X(k)Y(k + \tau)\} \\ &= Ac_2^s(\tau - D), \quad -\infty < \tau < \infty \end{aligned} \quad (79)$$

kur  $\{W_1(k)\}$  un  $\{W_2(k)\}$  stacionāri signāli, kuri nav vienādi ar 0, neatkarīgi viens no otra un ar  $\{S(k)\}$ .

Ievērosim, ka

$$c_2^s(\tau) = E\{S(k)S(k + \tau)\} \quad (80)$$

kovariācijas secība  $\{S(k)\}$ .

$c_{xy}(\tau)$  vērtība formulā 79 pieņem maksimālu vērtību pie  $\tau = D$ . Kā likums, praktiskajā modelēšana aprobežoto reģistrēto datu daudzuma dēļ, trokšņu avotu atkarības dēļ,  $c_{xy}(\tau)$  neobligāti parāda maksimumu pie laika aiztures  $D$ . Dažādas funkcijas tika pielīdzināti kroskorelācijas funkcijām, lai pierādīt laika aiztures kvalitāti.

Praktiskajos pielietojumos uzdevums, kurā signāls  $\{S(k)\}$  var tikt uzskatīts kā ne Gausa stacionārs gadījuma trokšnis, un trokšņa avoti - neatkarīgi, stacionāri, Gausa; līdzība starp  $\{X(k)\}$  un  $\{Y(k)\}$  var tikt veikta uz augstākas kārtas spektra pamata, tāda kā bispektrs. Atzīmēsim, ka signālu pašizstarošana no sarežģītām mehāniskām sistēmām sastāv no kvaziperiodiskām komponentēm un tāpēc var tikt pieskaitīta pie ne Gausa trokšņa. Galvenā priekšrocība augstākas kārtas spektra izmantošanai laika aiztures novērtēšanai pie augstākminētiem aprobežojumiem - tas ir tas, ka tie ir brīvi no Gausa trokšņa. Pieņemsim, ka  $\{S(k)\}$  - arī nevienāda ar nulli vērtība.

$$c_3^s(\tau, \rho) = E\{X(k)X(k + \tau)X(k + \rho)\} = c_3^s(\tau, \rho) \quad (81a)$$

$$c_{xyx}(\tau, \rho) = E\{X(k)Y(k + \tau)X(k + \rho)\} = c_3^s(\tau - D, \rho) \quad (81b)$$

kur

$$c_3^s(\tau, \rho) = E\{S(k)S(k + \tau)S(k + \rho)\}$$

tāpēc ka trešās kārtas Gausa procesa komulante nav vienāda ar 0. Ieguvot bispektru trešās kārtas komulantei formulās 81a un 81b, mums ir:

$$C_3^s(\omega_1, \omega_2) = C_3^s(\omega_1, \omega_2)$$

$$C_{xyx}(\omega_1, \omega_2) = C_3^2(\omega_1, \omega_2) \exp\{j\omega_1 D\}. \quad (82(b))$$

Pieņemot, ka  $C_3^S(\omega_1, \omega_2)$  nav vienāds ar 0, sekošo vienādību var ņemt vērā:

$$I(\omega_1, \omega_2) = \frac{C_{xyx}(\omega_1, \omega_2)}{C_3^S(\omega_1, \omega_2)} = \exp\{j\omega_1 D\}. \quad (83)$$

Vienīgais aprēķina ceļš laika aizturai  $D$  - tas ir funkcijas iegūšana.

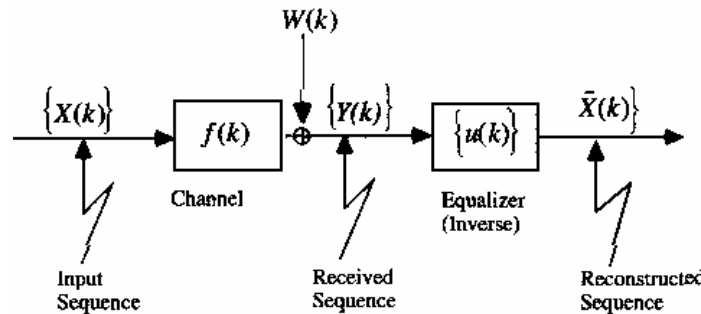
$$I(\tau) = \int_{-\pi}^{+\pi} \int_{-\pi}^{+\pi} \exp\{-j(\tau - D)\omega_1\} d\omega_1 d\omega_2 \quad (84)$$

kur maksimums pie  $\tau = D$ . Ir, protams, daudz veidu, kuri tika izveidoti uz trešās kārtas komulantes parametriskas modelēšanas pamata. Svarīgi atzīmēt, ka formula 83 - brīva no Gausa trokšņa, un tāpēc var gaidīt labākus novērtēšanas rezultātus, izmantojot  $I(\omega_1, \omega_2)$ . Zīmējums 2 parāda laika aiztures novērtēšanas rezultātus pie Gausa trokšņa esamības, kurš iegūts uz otrās un trešās kārtas statistisko datu metožu pamata. No zīmējuma redzams, ka metode, balstīta uz spektru, parāda labāku spēju skaņas samazināšanā.

### e. Akla dekonvolūcija un stabilizācija

Akla dekonvolūcija, vai stabilizācijas problēma, nodarbojas ar secīgas ieejas uzbūvi pēc iegūtas lineārās sistēmas izejas un statistiskas informā. Aklas dekonvolūcijas algoritmi - adaptētie filtrējošie algoritmi, izstrādāti tādā veidā, ka tiem nav vajadzīgs ārējais vēlamais raksturojums lai nogenerēt kļūdas signālu adaptēta stabilizēta filtra izejā. Citiem vārdiem, algoritms neredz vēlamo raksturojumu. Kā līkums, algoritms pats generē vēlamo raksturojuma novērtējumu, izmantojot nelineāru secības transformāciju, kura iekļaujas adaptācijas procesā. Ir trīs galvenas aklas stabilizācijas algoritmu grupas, kuras ir atkarīgas no tā, kur tika izmantota nelineāra transformācija:

- (i) "Bussganga" algoritmi, nelinearitāte adaptējošā stabilizējošā filtra izejā.
- (ii) Polispektra algoritmi, nelinearitāte adaptējošā stabilizējošā filtra ieejā.
- (iii) Algoritmi, kur nelinearitāte tiek apskatīta stabilizējoša filtra iekšienē, t.i. nelineāri filtri (piemēram, Voltera).



15. Komunikatīvas sistēmas blok-diagramma ar additīvo troksni.

Aprēķināsim laika diskreto lineāro transmisīvo kanālu ar impulsa raksturojumu  $f(k)$ , kura ir nezināma un var mainīties laikā. Ienākošie dati  $X(k)$ , pieņemsim, ir neatkarīgi un vienlīdzīgi sadalīti gadījuma mainīgie ar ne Gausa blīvuma sadalījuma funkciju ar vērtību 0 un dispersiju,  $\{X^2(k)\} = \gamma^2$ . Sākumā, troksnis nebūs ņemts vērā kanāla izejā. Tātad, iegūta secība  $Y(k)$ , (sk. zīm. 15):

$$Y(k) = f(k) * X(k) = \sum_i X(k-i) f(i). \quad (85)$$

Uzdevums ir atjaunot  $X(k)$  no iegūtas secības  $Y(k)$ , vai, kas ir viens un tas pats, identificēt inverso kanāla filtru (stabilizatoru)  $U(k)$ .

No zīm. 15 var redzēt, ka stabilizatora izejas secība  $\tilde{X}(k)$

$$\tilde{X}(k) = u(k) * Y(k) = u(k) * f(k) * X(k).$$

Iegūsim

$$\tilde{X}(k) = X(k - D)e^{j\theta} \quad (87)$$

kur  $D$  - aiztures konstante un  $\theta$  - fāžu nobīdes konstante un prāsa, lai:

$$u(k)*f(k) = \delta(k - D) e^{j\theta}, \quad (88)$$

kur  $\delta(k)$  Kronekera delta-funkcija. Izmantojot Furje transformāciju (formula 88) mēs iegūsim:

$$U(\omega)F(\omega) = e^{j(\theta - \omega D)}.$$

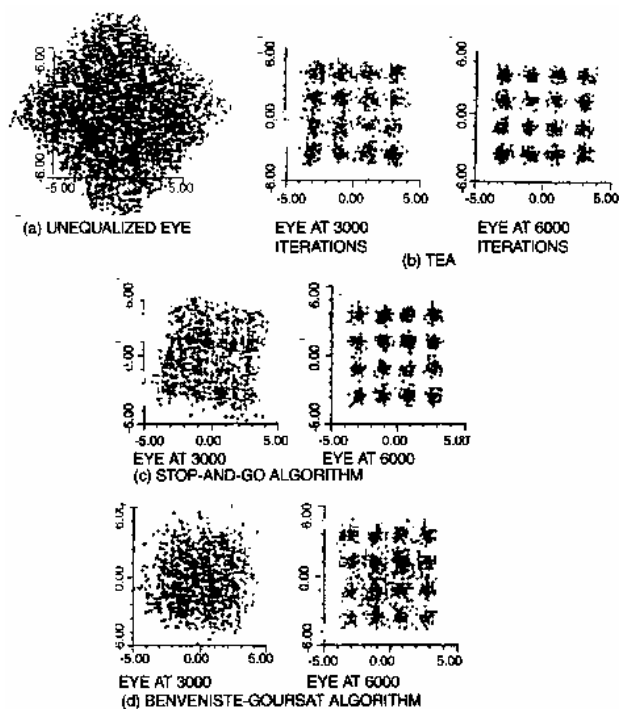
Tātad, stabilizatora uzdevums - aizsniegt transformācijas funkciju.

$$U(\omega) = \frac{1}{F(\omega)} e^{j(\theta - \omega D)} \quad (89)$$

Parasti,  $D$  un  $\theta$  - nezināmas. Kā līkums, aiztures  $D$  konstante neiedarbojas uz oriģinālo ieejas secības rekonstrukciju  $X(k)$ . Fāžu konstante  $\theta$  var tikt aizmesta ar izšķirt ierīces palīdzību.

Aklas stabilizācijas algoritmi balstīti uz augstākas kārtas statistiskiem datiem, izpilda stabilizējoša filtra ieejas transformāciju. Šī nelineāra transformācija ir it kā nelinearitātes atmiņa un identiska augstākas kārtas komulantes generācijai iegūtiem kanāla datiem. Viens no pirmiem akliem stabilizatoriem, balstītiem uz HOS un kuri parādījās literatūrā, bija Tri-Kepstra (TEA) stabilizācijas algoritmi. Viņi novērtēja stabilizējošā impulsa raksturojumu, izmantojot ceturtais kārtas komulantes komplekso kepstru (tri-kepstrs) sinhronam, diskrētam iegūtajam signālam. Divi TEA paplašinājumi tika apskatīti literatūrā. Pirmais - jaudas Kepstrs un stabilizācijas algoritma trīkoherence (POTEA), kurš ieguva Furjē vērtību stabilizatoram, izmantojot autokorelāciju un Furjē fāzu, izmantojot ceturtais kārtas komulantes un trīkoherences Kepstru. Otra pieeja TEA paplašināšanai - multikanālu gadījums, kurš izmanto apskatāmā signāla kroskorelāciju. Tas viss bija apzīmēts kroskorelācijas stabilizācijas algoritmā.

Labi pierēģistrēti literatūrā ka stabilizatori, pamatoti uz polispektra, daudz ātrāk sasniedz konverģenci, nekā Bussganga algoritmi, tāpēc ka ir vairāk aprēķinu uz vienu iterāciju. Zīm. 16 ilustrē 16-QAM korelāciju diagrammas (kvadrātiska amplitūdas modulācija), kuras ir izkropļotas ar kanālu un pēc tam nostabilizētiem ar TEA algoritmu un arī ar diviem citiem algoritmiem, kuri pieder pie Bussganga algoritmu grupas un kurus sauc Stop-And-Go un Binvenista-Gorsata algoritms. Skaidrs, ka algoritms TEA vairāk precīzs, nekā divi citi.



16. Diskrētas diagrammas. (a) Diskrēts signāls. (b) Izmantojot TEA. (c) Izmantojot Stop\_and\_Go algoritmu. (d) Izmantojot Benvenista-Gousata algoritmu.

#### f. Interferences anulēšana

Kad vēlamais signāls (SOI) izkropļojas ar palīgsignāla additīvo interferenci, kurš pieder šim signālam, korelēts tikai ar interferenci, interferences ierobežošana var tikt sasniegta ar adaptēta trokšņa izņēšanas procedūrām. Atbilstošs signāls formējas ar adaptīvo aizpildītāju lai saskaņot nevēlamo interferenci par tik, par cik tas ir iespējams, un izejas filtru atņem no ieejas, kurš sastāv no SOI un interferences, lai noformēt sistēmas izeju. Adaptīva trokšņu novērstāja uzdevums (ANC) - noformēt sistēmas izeju, kura būs labāka, nekā SOI. ANC pielietojumi iekļauj sevī dažāda veida interferenču novēršanu, piemēram, komunikācijās, balss iekārtās, interference antenās, telefonu aprīkojumā.

Tradicionāls šķēršais ANC, kurš apzīmēts šinī darbā kā ANC-SOS algoritms, izmanto LMS algoritmu un otrās kārtas statistiskus datus (SOS). Izmantojot praktiskā algoritmu ANC-SOS, mēs parasti saskeramies ar divām galvenām problēmām. Pirmā - ANC-SOS filtra matrica tieši iedarbojas ar nekorelācijas troksni uz primāro un ar to saistīto ieeju. Otrā - ANC-SOS algoritms ir atkarīgs no uzdevuma nostādnes, piemēram, tas ir ļoti jūtīgs pret saistīta signāla statistiskiem datiem un izvēlēto soļa lielumu.

Peņemsim, ka  $\{X(k)\}$  un  $\{Z(t)\}$  apzīme daudzuma novērtējumu primārajam un ar to saistītiem devējiem, atbilstoši:

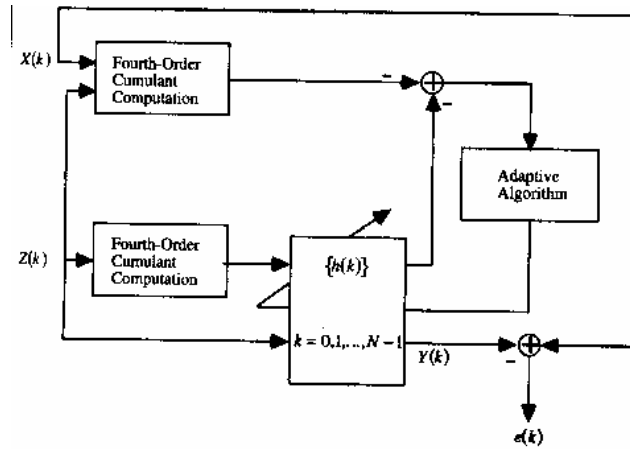
$$X(k) = S(k) + I(k) + N_p(k) \quad (90)$$

$$Z(k) = W(k) + N_r(k) \quad (91)$$

kur  $\{S(k)\}$  - mūs interesejošs signāls (SOI),  $\{I(k)\}$  - interference (šaurdiapazona vai plašdiapazona),  $\{W(k)\}$  - saistīts signāls, nokorelēts ar interferenci, un  $\{N_p(k)\}$  un  $\{N_r(k)\}$  nenokorelēti ar trokšņa devējiem. Mēs pieņemsim, ka SOI ir jebkurš signāls ar vērtību 0, piemēram, determinēts vai gadījuma, vai to kombinācija, un nenokorelēts ar interferenci vai ar saistīto signālu. Saistīts signāls - stacionārs, ar vērtību 0, ne Gausa gadījuma process. Trokšņi  $\{N_p(k)\}$  un  $\{N_r(k)\}$  - ilgstošs vai nokrāsots Gausa troksnis ar vērtību 0, nenokorelēts viens ar otru un neatkarīgs no SOI, interferences un saistīta signāla.

Jauns ANC, balstīts uz ceturtais kārtas statistiskiem datiem (ANC-FOS) rāda, ka ANC-FOS filtrs nav atkarīgs no ilgstošiem vai nokrāsotiem Gausa nenokorelētiem trokšņiem un nejūtīgs saistīta signāla statistiskiem datiem un soļa lielumu. Zīm. 17 ilustrē blok-diagrammu ANC-FOS algoritma.





17. Adaptēta trokšņa novēršanas konfigurācija, izmantojot ceturtais kārtas statistiskus datus (ANC-FOS).

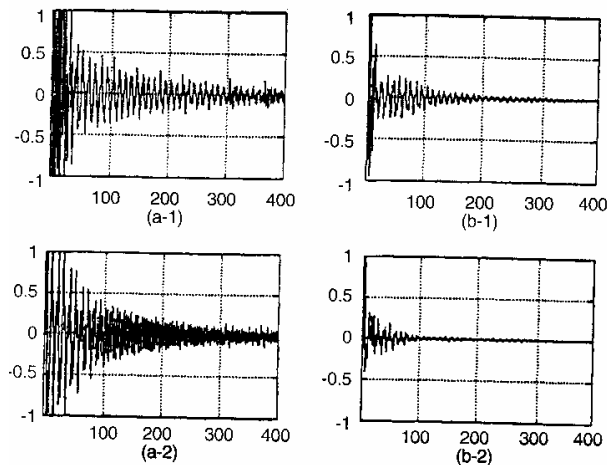
Mēs izrēķinājam tipisko piemēru salīdzinot ANC-FOS algoritmu un ANC-FOS algoritmu pie interferences novēršanas. Salīdzināšana veikta uz kļūdas starp SOI un tā rekonstrukcijas priekš katra ANC algoritma pamata. Pieņemsim, ka SOI determinēts BPSK, kuram ir divi stāvokļi un kurš ir attēlots formulā:

$$S(k) = \begin{cases} \cos(2\pi f_s T k), & \text{for } s_1 \\ -\cos(2\pi f_s T k), & \text{for } s_0 \end{cases} \quad (92)$$

kur  $f_s T = 0,43$  un viena stāvokļa ilgums = 20 izvēlnes. Saistīts signāls  $\tau$ , pieņemsim, ir sinusoidālo viļņu summa ( $W(k) = (W_1(k) + W_2(k))$ )

$$W_i(k) = A_i \sin(2\pi f_i T k + \phi_i), \quad i = 1, 2 \quad (93)$$

kur  $A_i$ 's un  $f_i$ 's apzīmē amplitūdu un frekvenci, atbilstoši.  $\phi_i$ 's - neatkarīgi, gadījuma mainīgi vienlīdzīgi sadalīti intervālā  $[-\pi, \pi]$ . Ievērosim, ka  $f_1 T = 0,1$  un  $f_2 T = 0,25$ . Katrs interferences signāls  $\{I_i(k), i = 1, 2\}$  generējas starp trīm MA sistēmām, kas uzbudināti ar saistīto signālu  $\{W_i(k), i = 1, 2\}$ . Atbilstoši MA koeficienti =  $[1, 0,1, -0,3]$ ,  $[1, 0,5, -0,1]$  un  $[1, -0,2, 0,2]$ , atbilstoši. Mēs pieņemsim ka  $I(k) = I_1(k) + I_2(k)$  ar  $A_1 = \sqrt{2}$ ,  $A_2 = 0,5$ , vai  $A_1 = 0,5$  un  $A_2 = 0,1$ . Zīm 18 parāda rezultātus, kuri iegūti no ANC-SOS un ANC-FOS. No zīm. redzams, ka visātrāk sasniedzas konverģence ar ANC-FOS algoritmu.



18. Kļūda starp SOI un to rekonstrukcijai. Pirmā kolonna parāda rezultātus, kuri ir iegūti ar ANC-SOS algoritma palīdzību ( $N = 32$ ), kad (a - 1)  $A_1 = \sqrt{2}$  un  $A_2 = 0,5$  un (a - 2)  $A_1 = 0,5$  un  $A_2 = 1$ . Otrā kolonna parāda rezultātus, kuri ir iegūti ar ANC-FOS algoritma palīdzību ( $N = 16$ ), kad (b - 1)  $A_1 = \sqrt{2}$  un  $A_2 = 0,5$  un (b - 2)  $A_1 = 0,5$  un  $A_2 = 1$ .

## Nobeigums

Pēdējo 20 gadu laikā spektra novērtēšanas tehnika kļuva par pamatu progresīvo komunikāciju, sonaru, radāru, balsu, biomedicīnu, ģeofizisko un vizuālo sistēmu veidošanā. Šī tehnika izmanto tikai otras kārtas statistiskus datus, kas runa par to, ka visi signāli tiek pamatoti uz Gausa. Vairākums signālu reālajā pasaulē nav Gausa. Tāpēc nav brīnums, ka spektrālā tehnika satikas ar ievērojamām grūtībām.

Ir daudz vairāk informācijas par gadījuma ne Gausa vai determinētiem signāliem, nekā par to korelāciju vai spektriem. Augstākas kārtas spektri (piemēram, polispektri, kuri tiek novērtēti atkarībā no augstākas kārtas statistiskiem datiem signālam, satur šo papild informāciju. Dotā grāmatā tika piedāvāts augstākas kārtas spektra analīzes apskats un tā pielietošana praksē.

Signāla apstrādes algoritmi, kas balstās uz augstākas kārtas spektra, tagad ir pieejami izmantošanai komerc- un kara pielietojumos. Lēto, atro ierīču parādīšanas un lielāka datoru pieejamība ļauj mums gūt vairāk informācijas no signāliem, ar ko mēs bijām spējīgi pagatnē, tas arī ļauj izdarīt pareizus secinājumus.

## Spontānas runas atpazīšana ar galvena vārda palīdzību adaptīvajā atbildē

(Noisy Spontaneous Speech Understanding Using Noise Immunity  
Keyword Spotting With Adaptive Speech Response Cancellation)

### Rezumē

Mēs pilnveidojam balss atpazīšanas sistēmu izmantojot noturību pret trokšņiem galvenajos vārdos, kuri atrodas adaptīvajā atbildes rūnās. Rūnas saprašanai mēs izmantojam galvenu vārdus. Lai atvieglot sakarus ar domājošo datoru mēs piedāvājam jaunu sistēmas komponentu: atsauksmi uz mākslīgas rūnas saasinājumiem. Mēs paplašinājām apmācības robežas pret trokšņiem noturībai, kas lava atpazīnāt balsi dialoga, kas ļauj atpazīt balsi dialoga ja viņa ir piesārņota ar neanulētu sintētisko(mašīnas) atbildi. Atcelšana ļauj datora izmantojam patraukt dialogu sistēmā lai labāk kontaktētos ar domājošo datoru. Mēs apvienojam vadību izmantojam uzskatītajam dialoga un atbilda multimodala ģenerators, lai veidotu dialoga sistēmu saprotamu izmantotajam pie ātras edieņas(fast-food) pasutinās. Eksperimenta rezultāti uzradīja adaptīvas teksta atcelšanas efektivitāte un trokšņa traucējumu diapazona paplašināšanos.

### Ievads

Tiek piedāvāta vairākas ātri darbojošas balss atpazīšanas sistēmas (HMM) ,kuras darbojas ar vāraķiem neatkarīgajiem izmantotajam ar lielu vārdu krājumu. Diemžēl viņu darba sfēras samazinājos pie trokšņu fona un teksta ātruma paļielinašanas. Vairākus sistēmu interpreti gramatiski pareizos vārdus, ka rezultāta viņi nevar strādāt, ja teksts nav adaptēs(reāls balss). Piedāvātas balss atpazīšanas sistēmas, karas tiek piedāvātas Eiropa programma projekta DARRA, ir ar trūkumiem, jo reālai runai ir dažādas vārda nozīmes. Ņemot vērā norodas uzrukumiem, mēs rādijam eksperimentālo dialoga runas paraugu TOSBURG(dialoga sistēma orientēta uz dialoga sapratni un atbildēšanu), šajā gadījumā tiek izmantota galvena vārda saprašana. Sistēma sastāv no vārdu korektetaja, semantisko vārdu analīze, dialoga vadība un multimodala atbilžu ģenerators.

Šajā darbā mūsu pieeja balsu atpazīšana dialoga režīma tiek piedāvāt pirmo reizi. Zemāk minēta balss sistēmas atpazīšanas problēmas (TOSBURGII) .Tiek piedāvāts nākamās sistēmas TOSBURG versijas; dialoga sistēma balss atpazīšanai TOSBURGII, kas reaģē uz balss izmantotāja patraukumiem. Ir aprakstītas jaunas sistēmas TOSBURGII iespējas; mākslīgais balss atpazīšanas atcelšana un apciešana uz trokšņu atšķiršanas.

### Dialoga runas sistēma ar multimodala atbildi

Tieša pieeja balss atpazīšana dialoga sistēmā.

- Lai pielietot šo balss atpazīšanas sistēmu reālajā pasaule mēs nepieciešams uzlabot noturību pret kļūdams. Bez tam paaugstināt spejas izšķirt (saprast spontānu runu) bez gramatiskajiem ierobežojumiem un izrunas veidiem. Balss atpazīšanas sistēma jaspei izšķirt (saprast) nepareiza izrunu, pauzes un ne sarunu valodas vārdus. Diemžēl, augstāk minētas runas pavadības ir grūti uzstādiēt priekša izmantojot gramatiskos noteikumus, sistēma ērta forma. Tāpēc mēs uzskatam, kā šie vardi var tikt ieguti izmantojot gramatika vardu sadaļa. Mēs varām izvairīties novadīt dialogu. Sakumā mēs uzbūvējam experimentalo paraugu runas atpazīšanas sistēmai TOSBURGs, kurš ļauj samazināties ar dažādiem lietotājiem. Sistēmai ir sekojošas īpatnības:
  - Sistēma saprot lietotājiem runu bāzejerties uz pamatvārdiem
  - Dialoga vadīšana, kas ļauj sadarboties ar intelektuālo datoru
  - Multimodala atbildi

Šajā sistēmā kontrols-vards nosaka kandidāta vārda kurš pēc tam tiek analizēts gramatiski un semantiski. Dialoga menedžeris interpreti kandidāta vārda izmantojot iepriekš uzkrāto vardu krājumu un dot semantisko atbilde. Atbilžu ģeneratoru izmanto lai radītu multimodālai atbildi, kurš ietver sevi sintezētu runu un vizuālo atbild 1.zīm.



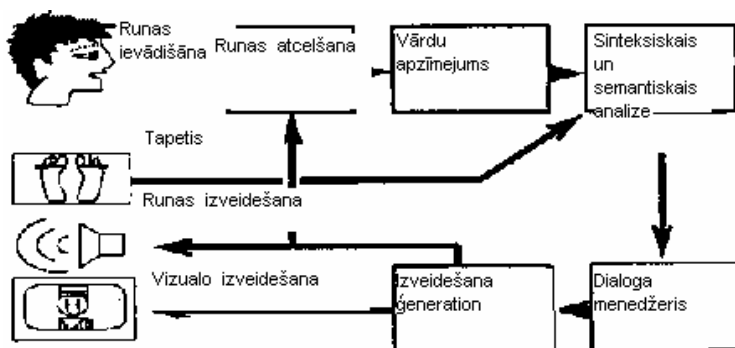
1.zīm. Piemērs vizuāla radīšana multimodāla dialoga sistēmai.

Tā ka izmantotajam atbilde tiek dota ne tikai mutiski bet arī ka grafiska informācija un teksts (zīm.1.), izmantotajās var apstiprināt atbildi sistēmai vizuāli līdz sistēmas darba beigām. Tāda gadījuma lietotājs ignorē ziņojumu un pārtrauc sistēmas atbildi. Diemžēl mūsu eksperimentālais paraugs nevar pieņemt un saprast lietotāju ziņojumu kamēr sistēma nav ģenerējusi atbildi. Atbildes runa noraidīta caur mikrofonu izkropļojas tā kā rezultāta samazinās saņemtas informācijas saprašana apjoms. Lai lietotais varētu noteikt vai sākt dialogu, mēs ieviesām grafisko lūpu attēlu pēc kuru formas var spriest : runāt vai nē.

### TOSBURGII- iespējas paplašināšana

Mēs paplašinājām eksistējošās sistēmas iespējas, ar mūsu sistēmu TOSBURG lai izmantotu iespēju pārtraukt runu brīdi kad runa lietotais. Šī sistēma ļauj sadarboties runas pārtraukšanas momentā. Šajā gadījumā sistēma ģenerē sintētisko balss atsaukšanu izmantojot trokšņa kontroles aktīvo tehnoloģiju. Mēs arī paplašinājām trokšņu traucējumu noturības robežas lai labāk izšķirto galvenos vārdus izmantojot sintētiskas atbildes komponentu. Mūsu dialoga sistēma apvieno galveno vardu korekciju semantisko vārda analīze, dialoga menedžeris multimodālais atbildes ģenerators. Katra atbilde var tiek sadalīt tā atsevišķi un novērtēta pēc 49 japāņu galvenie vārdiem..

Sistēma konfigurācija norādīta zim. 2.

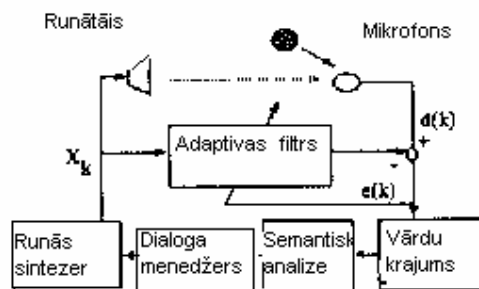


2.zīm. Sistēma TOSBURGII konfigurēšana.

Sistēma strādā reālajā laikā izmantojot 3 pamata darba stacijas un 5 DSP akseļatorus. Lietotājs sāk dialogu ar sistēmu stāvot uz paklāja kas reaģē uz svaru izmaiņām. Sistēma ģenerē atbildi pavadot to ar vizuālo attēlu uz ekrāna teksta veida, zīmējumiem, animācijas bildēm un sejas izteiksmi.

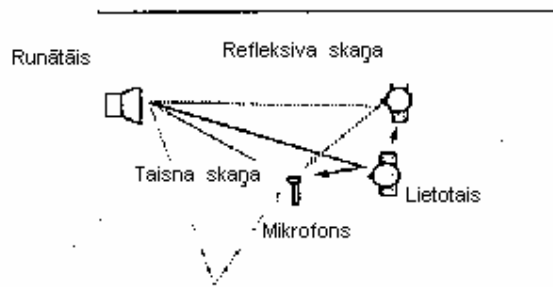
Nākamajās nodaļās plaši aprakstītas sintētiskas runas atbildes sistēmas un traucējumu noturības robežu paplašināšana.

Saskaroties ar sistēmas trokšņu piesārņojumu mēs piedāvājam atbildes runas adaptīvo sistēmu kura ļauj lietotājam sadarboties ar mašīnu.



3.zīm. Dialoga sistēma atcelšana.

Zīm. 3 attēlota diagrama- bloks dialogā runai ar rūnas atcelšanas sistēmu. Pie atbildes runas atcelšanas tiek izmantots LMS algoritms., kā novērtē atbildes impulsu starp skaļruni un mikrofoni ņēmot vēra lietotāja ķermeņa kustības sārūnas laika kā noradīts zīm. 4. Mēs āri piedāvāam vairākas metodes atbildes runas izlabošana, kas būvētas uz ciparu signālus saturošiem galvenajiem vārdiem lietotāju un atbildes runas laika. zīm. 4 Atstarotās skaņas piemēri.



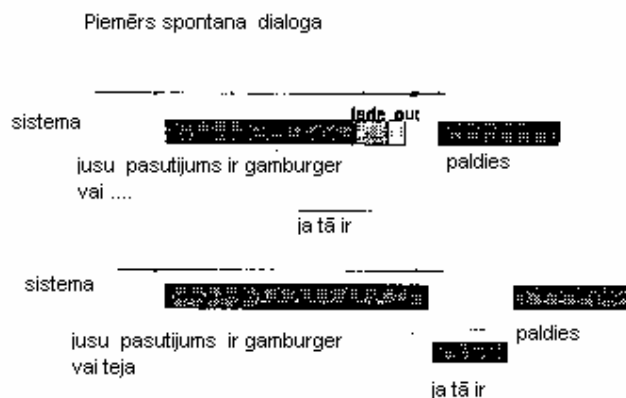
4.zīm. Piemērs skaņa izmaiņas .



5.zīm. Piemērs runāšanas atcelšana.

Lai sasniegtu stabilitāti un ātrdarbību sistēma izmantots NLMS algoritms, un tiek izmantots adaptīvais atslēgums un spektrālā sadalīšana. Adaptīvais atslēgums novērš nestabilitāti ātras signāla izmaiņas gadījumā. Bez tam spektrālā sadalīšana tiek izmantos pirmās pakāpes signālos paātrinot salīdzināšanu bez izskaitļojumu palielināšanas. Kā runas atcelšana piemērs sistēma tiek piedāvāts dialogs reālajā laikā izmantojot akceleratoru DSP , sistēma parādīja sevi kā draudzīgu attiecība pret lietotāju.

Tā kā runas atcelšana lietotais nosaka viennozīmīgi tad nav vajadzības turpmāk izmantot lūpu grafisko attēlu.(kā norādīts zīm. 1). Zīmējumā 6 norādīta sistēmas darba salīdzinājumu tabula sistēmas darbam reālajā laikā un parastā dialogā. Runas pārtraukuma gadījuma lietotais var ātrāk pieņemt lēmumu. Sistēma TOSBURG tapāt pārtrauc lietotāja runu ja runa satur pauzes.

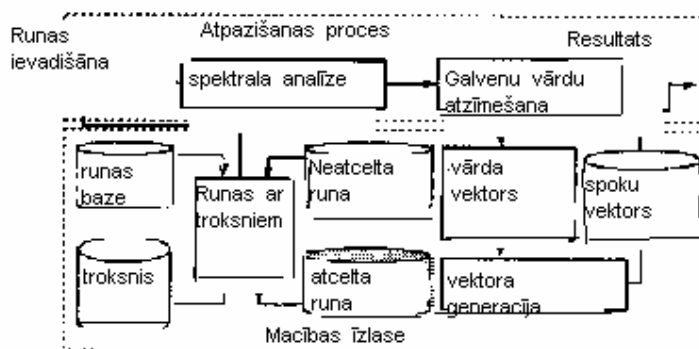


6.zīm. Piemērs runāšanas atcelšana.

Traucējumu noturības robežu paplašināšana kas ļauj atpazīt neatcelto mašīnas runas komponentu.

- runas ievadišana
- galveno vardu noteikšana
- datu bāze
- spektrāla runas sintēze
- ģenerējošajās vektors
- attēla atpazīšanas vektors

Atbildes runas atcelšana ir efektīva bet nevienmēr pilnīga. Neatceltais balss komponents izkropļo teikta jēgu. Tāpēc mēs paplašinājam trokšņu traucējumu diapazonu lai palielinātu galveno vardu atpazīšanu gadījumā ja runu patrauc lietotais. Zīm. 7 norādīta bloks diagramma galveno vardu atpazīšanai sistēmas apmācības procesa ar trokšņu traucējumiem. (Zīm. 7 bloks diagramma galveno vardu atpazīšana ar atzīmēšanu traucējumus drošā vidē.)



7.zīm. Bloks diagramma

Mācības procesā modelējas vide ar trokšņu traucējumiem kad ievada runu un ir ne tikai fons, bet arī ātri neskaidri teikti vārdi. Dati ar neskaidru izrunu tiek savākti un tiek izmantoti sistēmas atbilde (šajā gadījumā tiek ņemta vērā lietotāja ķermeņa kustības). Vārda attēls automātiski realizējas no daudzkārt agrāk saņemtiem līdzīgiem vārdiem. Šo metodi izmanto lai radītu jaunu vārdu modificētas matricas. Sistēmas apmācības process atkārtojas līdz brīdim kad samazinās trokšņu fons.

Noslēgumā.

Apmācības process sāk ar spontāno balsi, pēc tam izmanto vārdu izvēles komponentu atbildes teksta līdz pilnas sapratnes sasniegšanai. Datu bāze tiek izmantota sistēmas apmācībai pie runas ievadišanas (tiek izmantots mākslīgie trokšņa traucējumi).

Rezultāta novērtējums

Lai salīdzināt eksistējošo metožu un mūsu piedāvāto balss izšķiršanu tika veikts eksperiments kura 350 vīriešiem bija jāizdara pāsutejums fast-food sava runa izmantojot 49 galvenos vārdus. Tabulā parādīti eksperimenta rezultāti. Tabula 2 salīdzināts galveno vardu atpazīšanas līmenis viltus trauksme un atpazīšanas līmenis kopuma. Galveno vardu atpazīšanas līmenis neņemot vērā trokšņu līmeni 25 db sasniedza 58,8%, viltu trauksme 27,8% attiecīgi. Pielietojot jaunu metodi (ņemot vērā noturību pret traucējumiem) atpazīšanas līmenis pieaug līdz 89,6%.

Šo izmēģinājumu rezultāti parāda jaunās balss atpazīšanas metodes efektivitāti trokšņainā vidē. Šādi rezultāti tika sasniegti izmantojot galveno vardu izcelšanu lai

sasniegtu labāku savstarpējo sapratni starp sistēmu un lietotāju. Lai paātrinātu jaunas metodes ieviešanu mēs uzlabojam DSP akseleatoru kurš ļauj izmantot darba staciju Tab3 eksperimenta rezultāti.

1.tab.

Runātais	Training: 34 males Test: 5 males
vārdnīca	350 sentences including 49 keywords
troksnis	Computing room noise
vārda vektors	192 (16ch x 12 frames)

2.tab.Experimenta rezultats

Condition	Without cancellation		With Cancellation			
			Without NIL		With NIL	
SNR	$\infty$ dB	15 dB	$\infty$ dB	15 dB	$\infty$ dB	15 dB
Līneru atrašana	64.4%	58.8%	92.5%	85.3%	94.9%	89.6%
FA/H/W	25.5	24.1	15.7	14.9	16.1	15.4
Atpazīšanas līmenis	37.9%	27.8%	63.8%	55.6%	66.0%	58.4%

Noslēgums

Mēs aprakstījām dialoga runas sistēmu reāla laika vienība TOS kurš ir aparāts ar paplašinājām iespējām. Šī sistēma nodrošina spontānu cilvēka runas un mašīnas sapratnes procesu uz trokšņu fona izmantojot vārdu mēklešanas metode. Eksperimentālie rezultāti uzrādīja sistēmas draudzīgumu un efektivitāti reālajā laikā. Šobrīd mēs uzlabojam sistēmas darbību ņemot vērā evolūcijas procesus notiekušos reālajā pasaule.

Rate-Constrained Picture-Adaptive Quantization for JPEG Baseline Coders

## Problemas formulēšana

Nemot no tabulas JPEG pamat kodus un ievadot attēlu sadaļoto  $8^{**}$  pix nosakoto datu parraides līmeni ka attēla kodešanu,mes nosaka kvantešanas soļu izmēru  $\{Q_k: k=0, \dots, 63\}$  minimalizejot pilnība attēla izkropļojumus

$$D = \sum_{n=1}^N \sum_{k=0}^{63} D_{n,k}(Q_k) \quad (1)$$

Parraidu atrums:

$$R = \sum_{n=1}^N R_n(Q_0, \dots, Q_{63}) \leq B, \quad (2)$$

kur  $D_{n,k}(Q_k)$  ir koeficient DCT izkropļojums,  $Q_k$ -kvantešanas soļus,  $R_n(Q_0, \dots, Q_{63})$ - ir bināra kodu numeros. Vienkaršonai mēs mēram DCT izkropļojumu koeficientus un kvadrat kļudu starp originalu ņielu un kvantejamo. Ka alternativu var izmantot (weight distortion measure).

Pieļaujām, ka  $J_n$  ir n-bloka gārums. Noradam tekošo garumu, indeksu ņimeņa robežas un koda garumu, kas tiek noteikt ka  $r_{nj}, s_{nj}; j=1, \dots, J_n$ .

Divejado kodu numuru ņielums var tikt ieguti n-bloka un izkaitļoti

$$R_n(Q_0, \dots, Q_{63}) = L_n^{(dc)} + \sum_{j=1}^{J_n} L(r_{nj}, s_{nj}) + L_{EOB}, \quad (3)$$

kur  $L_n^{(dc)}$  ir bināra koda lielums Dc kodešanas un L ir beigu koda lielums bloka.

Lai minimizeto problemu var izmainit daļiņu sadales ņimeni. Diemžel, attiecīga kodešana nav vienkarta daleņu suma kodeta ar individualo DC T koeficientu palīdzību, tapec ir gruti atrast optimālo risinājumu ar klasiska algoritma palīdzību. Nesen tika piedavata gradientas līdzības metode lai risinātu kvantešanas tabulas lokālas optimizācijas uzdevumus. Problema balstas uz to ka DCT koeficienti ir Laplasa koeficienti. Diemžel Laplasa sadalīšana tiek izmantota tikai DCT koeficienta tipiskajā statistikā un nevar atvert nejauo attēlu ievadianu. Nakošajā nodaļā mēs piedavajam rekursivo algoritmu risinājuma meklešana lai atrastu attēlu daļiņas ar statistika attēlu ievadišana.

Kvantešanas tabulu radianas algoritma apraksts



Mēs izmantojam rekursīvo meklešanas metodi lai atrastu pieņemamu (bet iespējams ne optimālo) lēmumu kvantešanas radišanai. Mūsu pieja ir vienkārša versija atlasa kvantešanas algoritmiem. Šajā darbā mēs pielietojam optimālo kvantešanas algoritmu saīsināšanai. Šī algoritma ideja aprakstīta sekojoši: sākot no kvantešanas tabulas inicializāciju ar liela izmēra soļi pie datu parrāides ar zemu daļiņu izlādi un augstu izkropļojumu pamazām samazina saļa izmērs katra jauna piegājiena, kamēr tiek sasniegts iepriekš noteiktais daļiņu lielums. Katra iterācija mēs sasniegam zemāku izkropļojumu līmeni un pie tam sasniegtam augstāku parrāides līmeņa atrumu. Novērojot skaitliski mēs meģinājam atjaunot kvantešanas tabulu veida, lai samazinātu izkropļojumus un palielinātu daļiņu izmērus, palielinot soļa lielumu ievērojot kvantešanas tabulu. Tada veida mēs atradam lielumu  $k$  un  $q$ , kas risina maksimizācijas problēmu :

$$\max_k \max_q \frac{-\Delta D |_{Q_k \rightarrow q}}{\Delta R |_{Q_k \rightarrow q}}, \quad (4)$$

kurš  $\Delta D |_{Q_k \rightarrow q}$  un  $\Delta R |_{Q_k \rightarrow q}$  ir daļiņas līmeņa kopeja lieluma izmaiņas 'k' kvantešanas tabulas  $Q_k$  ievē, apzīmēt ka  $q$ .

$$\Delta D |_{Q_k \rightarrow q} = \sum_{n=1}^N [D_{n,k}(q) - D_{n,k}(Q_k)] \quad \text{un} \quad (5)$$

$$\Delta R |_{Q_k \rightarrow q} = \sum_{n=1}^N [R_n(Q_0, \dots, q, \dots, Q_{63}) - R_n(Q_0, \dots, Q_k, \dots, Q_{63})]. \quad (6)$$

Lai vienkāršotu skaitļošanu mēs sadalam maksimizācijas problēmu divās daļās kur pirmā daļa izskaitļojas:

$$\lambda_k = \max_q \frac{-\Delta D |_{Q_k \rightarrow q}}{\Delta R |_{Q_k \rightarrow q}} \quad (7)$$

visas nozīmes otra daļa atrisinās

$$\lambda = \max_k \lambda_k. \quad (8)$$

Algoritms izmanto tabulas  $\lambda_k$  un  $Q_k$  kur  $Q_k$  ir resinājums (7). Katra atkartojuma, algoritms meklē  $k$  lai maksimizēt  $\lambda_k$  un atjaunot attiecīgo kvantešanas ievē  $Q_k$  kvantešanas tabulā. Katra cikla ievē ar numuru  $k$  uz tabulas  $\{\lambda_k\}$  un  $\{Q_k\}$  tiek atjaunoti saskaņā ar (7) lai sagatavotu jaunu atkartojumu. Lai saglabātu koeficienta DC kvalitāti mēs uzstādam  $Q_0$  tad pašu lielumu, kadu dot JPEG kodešana. Ša soļa izmērs ( $Q_0=16$ ) nodrošina mazākus kvantešanas zudumus koeficientiem  $D_c$ . Tada veida kvantešanas tabulas radišanu nosaka soļa izmērs AC. Tātad algoritms ir dots zemāk:

1. Kvantešanas tabulas inicializācija

$$Q_k = \begin{cases} 16 & \text{for } k = 0, \\ Q_{\max} & \text{for } k = 1, \dots, 63. \end{cases} \quad (9)$$

2. Inicializācijas tabulas koeficientiem  $\lambda_k$  un  $Q_k$  nosakot  $q \in \{1, \dots, Q_{\max}\}$  un risināt (7) kur  $k=1, \dots, 63$ .

3. Meklēšana  $\{1, \dots, 63\}$  lai atrast (8).

4. Kvantešanas tabulas atjaunošanas  $Q_{k^*} = Q_k^*$

5. Atjaunošanas  $\lambda_k$  un  $Q_k$  lai atrast  $q \in \{1, \dots, Q_{k-1}\}$  (7) solis  $k=k^*$ .

6. Soļu atkartojums 3...5 tik ilgi kam 3...5 tik ilgi kamēr  $R(Q_0, \dots, Q_{63}) \leq B$ .

Maksimālais soļa lielums JPEG kodešana tiek noteikts ka 256. Reāli mēs saņemam kvantešanas tabulu ar soļa izmēra 128, tas novēd pie tā paa rezultāta, jo tas ir maksimāli pieļaujama solis, bet šajā gadījumā samazina atkartojumu skaiti. Tapēc mēs izmantojam skaitli 128 256 vieta ka  $Q_{\max}$  koeficientu.

Tas tiek viegli noteikts

$$R_n(Q_0, \dots, Q_{63}) = \sum_{k=1}^{63} R_n^{(k)}(Q_k) \quad \text{for } n = 1, \dots, N, \quad (10)$$

Šajā gadījumā augstāk aprakstītais algoritms izmantoto visoptimālākos punktus iegūtus ar mazākiem izkropļojumiem, izmantojot dažādas kvantešanas tabulas.

Šie noteikumi ne vienmēr tiek ievēroti tāda veida kvantešanas tabulas izmantošana nav optimāla. Pie katras algoritma atkartošana samazina izkropļojumu un palielina daļiņu lielums.

Precizākai izskaitļošanai nepieciešams izmantot 63 lks lai atjaunotu katrā iterācijā  $\lambda_k$  ta ka  $Q_k$  modernizācija var izmainīt  $\Delta R|_{Q_k \rightarrow q}$  ja  $k \neq k^*$ .

## Eksperimenta rezultāti

Datora modeļošana tiek izmantota šajā darbā uz JPEG bāzes sistēmas pie meln –balta attēla ievadīšanas, lai novērtētu piedāvāto paraugu.

Katram paraugam mēs salīdzinām divu kvantēšanas tabulu un (1) JPRG kvantēšanas tabulu un (2) kvantēšanas tabulas iegūtas ar augstāk minēto algoritma palīdzību. Mēs iegūvam salīdzinājumu tabulu.

Tab.1 Attēlu salīdzināšana tabula

Attēls	PSNR(JPEG)	PSNR(at.ad.)	Izlabošana	Limenis
Lenna(512*512)	35,8db	36,6db	0,8db	0,64bpp
Barbara(720*576)	32,8db	34,5db	1,7db	0,98bpp
Baboon(512*512)	28,2db	29,7db	1,5db	1,39bpp

Pirma tabula parāda attēla nodešanas rezultātus (testa attēli Lenna, Barbara, Barboona. Lenas attēls ir kodēts ar JPEG attēls adaptīvas kvantēšanas tabulu, ka norādīts at.3,4 attiecīgi. Mēs pievēršam jūsu uzmanību tam ka divu kodešannu rezultātu salīdzināšana parāda, ka kodēta attēlu kvalitāte pie vienāda līmeņa sistēma PSNR ir augstāka ne ka JPEG kodešana.



Ķim. 3(a)



Ķim. 3(b)

## Noslegums

Mēs piedāvājam kvantēšanas tabulu iegūšanas algoritmu JPEG sistēma DCT kodos. Šis algoritms ir efektīvs pie apmaiņas starp izkropļojumiem un līmeņu dažādības. Si informācija tiek balstīta uz attēlu ievadīšanas statistiku. Kvantēšanas tabula piemērojas elementiem kas ir attēla sastava. Ka tika demonstrēts eksperimentslojos modeļos attēls ar piedāvāto algoritmu ļauj kodēt attēlu izkropļojumiem ne ka izmantojot empiriskas kvantēšanas tabulas. Bēz tam, šī metode ļauj lietotajam kontrolēt attēla kvalitāte JPEG kodešana. Ši īpatnība ir būtiska ciparu ierakstu aparatūrā.

## Interpolācijas JPEG kodēšanas shēmas (A JPEG-Based Interpolative Image Coding Scheme)

### Abstrākcija

Nesen JPEG atzina par vienu no labākajiem kodēšanas shēmu. Pie nelielas bitu frekvences JPEG apmierināja daudzos praktiskajos pielikumos. Šajā stasta tiek piedāvāta nepazīstama kodēšanas shēma, kura balstas uz JPEG tehnoloģijas. Ja salīdzināt šo shēmu ar standarto JPEG, tad izrādās, ka pirmā ir daudz efektīvāka, jo bloķēšanas efekts ir samazināts līdz minimumam.

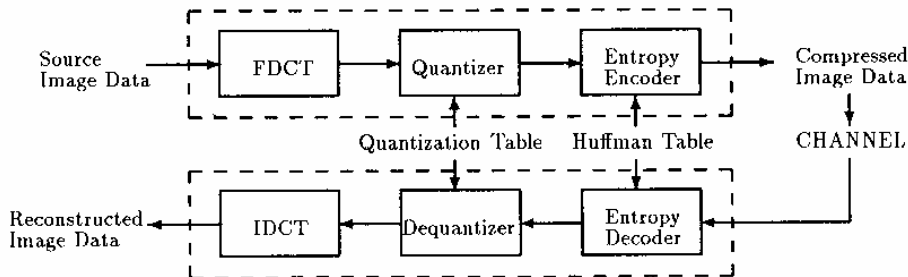
### Ievads

Efektīvas video/bildes kodēšanas shēmas pievērš lielu uzmanību no akademiķu un industriālo ļaužu puses. Šajā virziena darbs iet jau 20 gadus un rezultāti parādās standarti: JPEG, MPEG un H.261. JPEG formāts pirmkārt domāts kadru freimam, MPEG un H.261 domāti video plūsmam. Šajā raksta uzmanība tiks pievērsta galvenokārt JPEG tehnoloģijai, kas izmanto modifikācijas interpolācijas shēmu.

Šī laikraksta uzbūvē ir šāda. Sekcija Nr.2 – īsi tiks aprakstīts JPEG principi. Pēc tam uz praktiskā piemēra apskatīsim bloķēšanas efekta iespaidus, kas rodas kad izmantojam JPEG, pie zemas bitu frekvencēm, kas ir ļoti manāms cilvēka acim, pie kā pik/pik un signāls % troksnis attiecība paliek relatīvi laba. Tāpēc 3. sekcijā mēs piedāvājam interpolācijas shēmu, kas ļauj samazināt bloķēšanas iespaidus. 4. sekcijā doti daži datoru simulācijas rezultāti, lai varētu salīdzināt jauno shēmu ar standarto JPEG shēmu.

### JPEG principi

Principiāla JPEG kodēšanas blokshēma ir uzrādīta Zīm.1.



Zīm. 1

Te ir kodēšanas un dekodēšanas shēmas. Kodēšanas shēma sastāv no 3 galvenām daļām: Diskrēta Kosinoidāla priekšapstrādes Transformācija (FDCT), kvantizācija un entropijas kodēšana. Dekodēšanas shēma ta pati sastāv no trim daļām, kas veic kodēšanas apgrieztas darbības: inversija kosinoidāla transformācija, entropijas dekodēšana, dekvantizācija. DCT apstrāda katru 8x8 bloku pēc kārtas. Katra bloka DC komponenta ir nokodēta diferencētā impulsa kodēšanas formātā. Tas K-ta un (k-1) bloka starpība. Visi AC komponenti ir nokodēti impulsa modulācijā.(PCM)

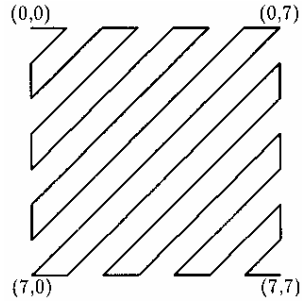
$$Q = \begin{pmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{pmatrix}$$

Kvantizāciju un dekvantizāciju izpilda ar priekšdeterminēto kvantizācijas tabulu.

A parametrs norāda skalas tipu kvantizācijas tabulā.

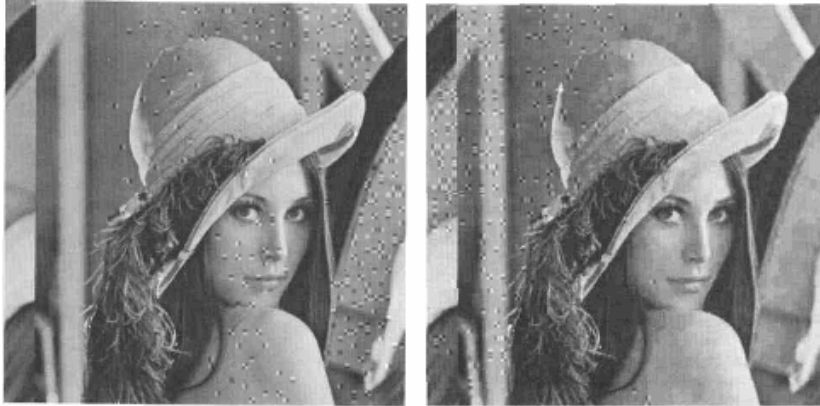
$$\alpha = \begin{cases} 10 - q; & \text{if } q \leq 9 \\ 1/2^{(q-9)}, & \text{otherwise} \end{cases}$$

Pirms ta, kamer saksies kvantizācija, visi 8x8 bloka DCT koeficienti tiks pargrupēti zigzagveidā., ka ir uzrādīts Zīm.2.



Zīm.2

Pēc tam katro komponentu, kura atšķiras no nulls kode paru veidā. Kuras pirma daļa rāda kvantizācijas vērtību un otrā rāda cik daudz komponentu ir ar nulls vērtību. Tāda veidā var izvarīties no nulls komponentem. JPEG kodēšanas kvalitāte var kontrolēt tikai ar Q( vai a) parametru un visi pārēji ir fiksēti. Piredze rāda, ka JPEG kodēšanas shema dot labo rezultātu pie diezgan augstas izšķirspējas kvalitātes un mērenas bitu frekvences (apm. 1.0 bpp). Ja būtu augstāka bitu frekvence ( teiuksīm 0.4 bpp) , ta sak ļoti itekmēt blokēšanas efekta rašanai un AC koeficients drizāk bus 0. Atšķirību var redzēt zīm.3(a) un 3(b)

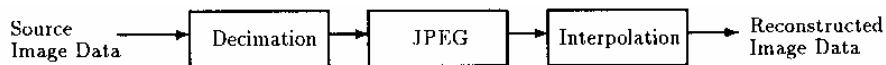


Zīm. 3(a)

Zīm. 3(b)

## Inerpolācijas JPEG shemas

Te būs apskatīta metode, kas balstas uz standartam JPEG shemam un ļauj samazināt blokēšanas efekts pie ļoti zēmam bitu frekvencem.



Zīm.4

Te ir piedāvata principiāla šīs metodes blokshēma. Galvēna idēja ir šāda: No šakuma oriģināla bilde iet caur decimācijas bloku, kuru taisa šaurjoslas filtrs. Decimāla Bilde ar pasliktinātu izšķirspēju tiks nokodēta standartā JPEG kodā. Gālējo rezultātu ar pilno izšķirspēju iegustam

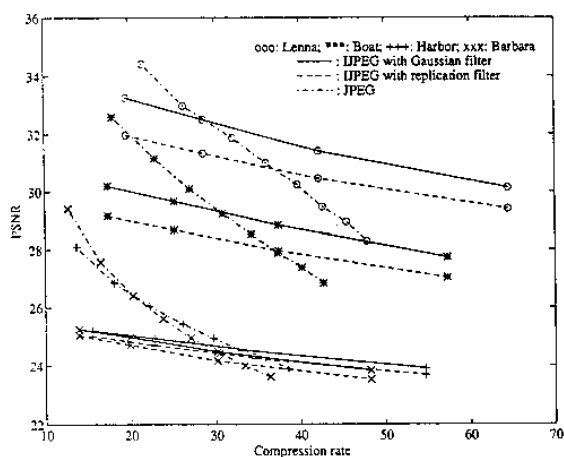
**Laižot cauri inerpolācijas filtru. Dažas piezīmes, kas attiecas uz šo shēmu. 1. Šeit inerpolācija ir pats svarīgais moments. Tads gājiens ļauj ļoti pamazināt elementu skaitu un saglabāt kvalitāte. 2. Decimāla bilde var būt nokodēta ar lielāko izšķirspēju, ierakstot skalas faktoru kvantizācijas**

tabulā. Ļoti zēma bitu frekvence ir garantēta ar decimāciju, pēc tam kompresijas frekvence vajag reizināt NxN reizes, kur N ir decimācijas frekvence. 3. Salīdzinājūmā ar standartam JPEG shemam , šī shēma ļauj efektīvi samzināt blokēšanas efekts, kas ir sasniegts ar decimāciju un inerpolācijas pielietošanu. 4. Ir skaidrs, ka ir jēga izmantot tikai zēmas frekvences decimāciju.

Šī metodei ir liela atšķirība no sub-band shēmam. Viegli saprast, ka šaurjoslas filtrs dos labako korelāciju starp pikseliem decimālai bildei.

## Datoru simulācija

Lai salīdzināt piedāvāto shemu ar standarto JPEG shemu, tika taisīta datoru simulācija. Daži rezultāti ir Zīm.5



Zīm.5

Šeit ir bilžu Boat, Harbor, Barbara, Lenna novērtējums pie dažādiem bitu frekvencem. Eksperimentā tiek izmantots 2x2 zemfrekvences filtrs decimācijas blokā. Divi interpolātori atrodas interpolācijas blokā. Pirmais ir kopiju filtrs, otrs ir Gausa filtrs, strādājošs pēc matricas. Izrādas, ka Gausa filtrs nodrošina labāko PSNR attiecību. Nekā kopiju interpolators. Mēs arī ierādījām, ka pie relatīvi zemas kompresijas frekvences, PSNR koeficients ir mazāks nekā standartam JPEG shemam. Bet pie pietiekami lielām kompresijas frekvencēm izrādas, ka PSNR ir lielāks, nekā JPEG'am.

Lai varētu spriest par vizuālam rezultātam ir doti Zīm. 6(a) un Zīm.6 (b)



Zīm.6 (a)

Zīm. 6(b)

, kur ir 2 nokodetas Lenna bildes (abas 0.28 bpp) izmantojot interpolācijas shemas (ar dažādiem interpolācijas filtram). Salīdzinot bildes 6.(a,b) un 3. (a,b), var secināt ka pirmie daudz palabojusies. Gandrīz nav manai bloķēšanas efekti, neskatoties uz to ka dažādas detaļas var tikt pazaudēti.

## Nobeigums

Ši laikraksta, tika piedāvātas interpolācijas koddēšanas shemas, kas kā bāzi izmantoja JPEG tehnoloģijas. Galvenais mērķis bija samazināt bloķēšanas efekts, kad bitu frekvence ir maza. Praktiskas bildes rāda, ka problēma veiksmīgi atrisināta.

**Psihovizuālo kropļojumu novērtēšana monohroma attēlam un video kompresijai**

(Psycho-Visual Based Distortion Measures for Monochrome Image and Video Compression)

## Abstrākcija

**Sajā apskatā tiks aplūkots psihovizuālais kropļojuma novērtēšanas modelis, kas domāts monohroma attēlam un video. Šis modelis ir līdzīgs cilvēka prāta uztvēršanas principam. Tiek izdarīti subjektīvie testi, kuras mērķis bija saņemt kompresēto video un attēlu rangu rezultātus. Pie tam tiks izmantoti dažādi kropļojumu mērīšanas un vidējas kvadratiskas novērtēšanas algoritmi.**

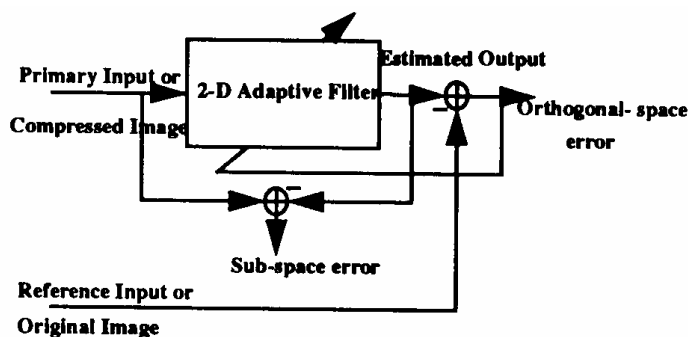
## Ievads

Šodien plaši pielieto dažādi kompresijas algoritmi, lai minimizēt datu skaitu, līdz ar to samazināt prasības atmiņai un parsūtīšanas iespējam. Kompresijas algoritma efektivitāte ir atkarīga no trīm lietām: kompresijas koeficients, izpildes sarežģītība un attēla kvalitāte. Pirmie divi kritēriji ir pietiekami viegli izskaitļot, bet trešo ir ļoti grūti novērtēt, priekš kompresijas algoritma. Lai novērtēt kļūdu, mērījumiem plaši pielieto vienību piksels-kvadrāt kļūdu starp oriģinālo attēlu un saspiesto. Tāds mērīšanas variants sevī iekļauj normētas video kvadrātisko kļūdu, maksimālais absolūtais novirzījums, attiecība signāls-troksnis un maksimāla attiecība signāls-troksnis. Šitas kļūdas izskaitļo neņemot vērā psihofizioloģiskais faktors. Tika taisīti daudz mēģinājumi nomodelēt cilvēka vizuālo uztvēršanas sistēmu kompresēto attēlu kropļojumu mērīšanai, bet tikai daži mēģinājumi bija veiksmīgi. Agrīe darbi šajā jomā tika taisīti izmantojot nelineāro transdukciju stimulācijas sekošanu ar telpisko filtrāciju. Vēlāk tika sataisīts rupjš cilvēka uztvēršanas sistēmas skaitļošanas modelis, kas taisīja operācijas uz kļūdaina signāla, lai novērtēt attēla kvalitāte. Lielākajā daļā šis modeles rezultāts bija ļoti līdzīgs vidējas kvadratiskas kļūdas vērtībai.

Apstiprinājums mūsu modelim atrodas faktā, ka cilvēkam ir daudz svarīgāk nekorelētas kļūdas ar lokalajiem īpašībām, nevis korelētas kļūdas, novērtējot saspiestos attēlus. Priekš straumes video galviene kropļojuma avoti ir telpiska razmitie un kustības objekta malas kontūra zaudēšana.

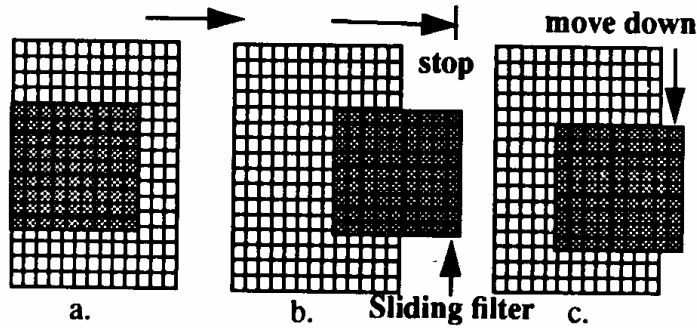
**Monohromas bildes kropļojumu novērtēšana**

**Nodefinēsime kļūdas signāls (e), kurš būs starpība starp oriģinālas bildes pikselim un kompresētas bildes. Sadalīsim kļūdas signāls divās daļās: ortogonālas telpas kļūdas signāls, kas ir daļa no kļūdas signāla, kura korele ar oriģinālas bildes lokalam īpašībām un zemtelpas kļūdas signāls, kas ir daļa no kļūdas signāla, kura nekorele ar oriģinālas bildes lokalam īpašībām. Sadales operāciju taisa ar 2-D adaptīva filtra palīdzību, ka ir uzradīts zīm.2, izmantojot modificēto 2-D LMS algoritmu.**



Zīm.1

**2-D adaptīvais filtrs strāda sekojoši. Piksels, kas pēte, atrodas 9x9 filtru bloka centrā. Viņam ir 4 kaimiņi-no kreisas malas, no labas, no augšas un no lejas. Zīm.2 ir uzradīta filtrācijas operācija ar slidojošo filtru, kas pārvietojas pa labi. Kad filtrācijas logs sasniedz kolonas beigas, viņi pārvietojas viena rinda uz leju un sāk reversīvo kustību. Šitais process turpinās līdz tam, kā mēr pikseli nebūs filtrācijas bloka centrā.**



Zīm. 2

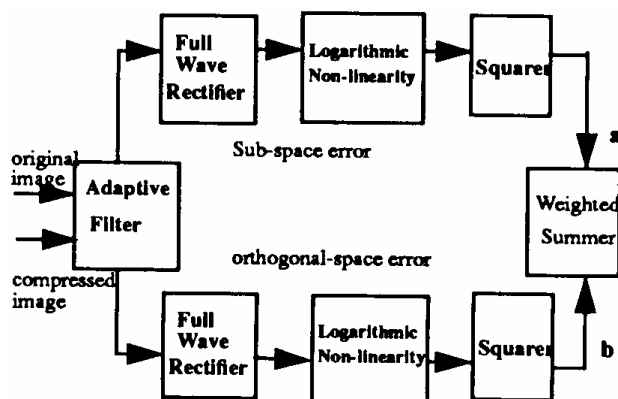
Nakošā solī izmantojot pilno viļņas detektēšanu, saņemam kļudas absolūtu vērtību un ta būs negatīva un talak gatavojam datus nakošajai nelinearai logaritmiskai operācijai. Tas , ka mūsu acim ir logaritmiska gaismu intensitātes uztvēre ir zinams ļoti sen, tas noteik tapec, ka neironu jutība pret gaismu ir proporcionāla gaismas enerģijas logaritmam. Ceturtais solis tas ir judas nelineara pārveidošana. Šī solī, divam kļudu signāla daļam taisa nelineāro pārveidojumu, pec kura jauda pieaug divreiz. Piektāja solī notiek  $D_g$  un  $D_s$  kropļojumu lielumu svaru pārbaudes. Nakoša izteiksme apraksta  $D_{img}$  lielumu:

$$D_{img} = A * x * D_s + b * x * D_o$$

Mēs izpetījam šis modelis dažādiem  $a$  un  $b$  vērtībām, vienai no bildem (Lenna) USC datu bazē. Lai minimizēt kļudu starp subjektīvam kropļojumiem un izmērītam , mēs atrodam ka  $a=0.80$ , bet  $b=0.20$ . Pie šīm vērtībām būs vislabākais vizuālais uztvērē rezultāts. Mēs pārbaudījam rupjo rezultātu , kas ieguvam izmantojot kropļojumu mērīšanas ar  $a$  un  $b$  vērtībām, piecam dažādam bildem datu bazē. Pilnais kropļošans mērīšanas modelis ir uzradīts Zīm.3

### Kropojumu mērīšana video plusmam

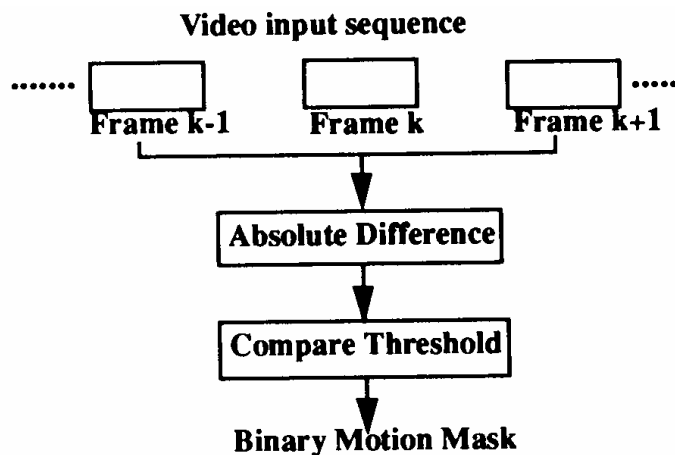
Cilvēka iespēja atšķirt detaļas kustīga bilde ir atkarīga no ta cik daudz ir kustību un cik atri cilvēka acis var sekot mainīgai bildei. Nekustīga daļa no kustīgas video plusmas var but uztverta daudz labāk un ar lielāko detaļas skaitu. Videjais informācijas saturs, laiks video kadru scenai ir daudz mazāk, neka videjais informācijas sturs videja laika kustīgai scenei. Tapec ir velams dalīt kustīgas daļas no video sekvenses.



Zīm. 3

Kustīga kadra segmentācijas algoritms

Šis algoritms izmanto, lai sadalīt kustīgos pikselus no kadra pikselim freimā video sekvencē.  $K$ -ta freima dalīšana, taisa izmantojot  $(k-1)$  un  $(k+1)$  freimus. Pirmā solī atrodama absolūtu starpību starp diviem freimiem. Otrajā solī notiek starpību salīdzināšana starp katru pikseli ar kustības sliekšņa detektēšanu ( $T$ ). Pikseliem, kuram absolūta starpība ir lielāka nekā  $(T)$ , tad šos pikselus uzskata par kustības pikseliem, visus pārējos uzskata par kadra pikseliem. Kustīgiem



pikselim piešķir "1", bet kadra pikselim "0". Tada veida ieguvam bināro kustības masku. Algoritms ir uzradīts Zīm.4

Zīm.4

Parametra  $T$  noteikšana nav tik viegls uzdevums. Ideāla gadījuma  $T$  būs) vērtība, tas ir gadījums kad video ir ieguta ne no kameras un nav trokšņa. Lai noteikt  $T$  vērtības paņemsim 100 video sekvences katrā freimā, kura izmērs ir  $352 \times 240$  p un 8 bitu gaismas intensitātes vārds. Katra sekvencē izdalīsim kustīgas un kadra daļas., ka arī sadalīsim tos divas daļas. Pēc tam pārbaudīsim dažas  $T$  vērtības un ņemsim tos, kas dot minimālo kļūdu.



### Kropļojumu mērīšana kadra pikselīm

Sī mērījumu mērķis ir noņemt kropļojumu efekts kadru daļa video sekvencē. Šeit mēs izmantojam dažādas kropļojumu mērījumus, tadus pašus, ko izmantojam monohromam bildem. Sitas kropļojumus devesim par D(stl.k)

### Kropļojumu mērīšana kustības pikselīm

Visbiežāk video plusma notiek sekojoši kropļojumi: traipes, malas īsuma zaudēšana kustības laukā. Malas kroplējumi rodas komponentes mazas frekvences dēļ. Tas arī varētu būt nepareizas malas pārraides dēļ video plusma izejā, vai kontrasta pazušana.

Tumša traipa efekts rodas freima pazušanas dēļ. Kļūda stap originālo un saspiesto freimu parasti ir diezgan liela un tas ir redzams zem-telpas kropļojumu mērījumos.

**Divi citi artefakti stadīti preiškā kustības plaksnē video plusmā un ieguti izmantojot telpisko Sobēla filtru malu detektēšanai. Mēs izmantojam 8-kaimiņu Sobēla telpisko filtru, tā kā šim filtram ir priekšrocības – gludījums un pazemināta jutība pret troksni. Sobēla filtram ir divas masku daļas – viena horizontāla malām otra vertikālam.**

-1	-2	-1
0	0	0
1	2	1

Horizontalais

-1	0	1
-2	0	2
-1	0	1

Vertikālais

Gradientvektora komponentes x virziena ieguvam no izteiksem

$$G_x = (x_7 + 2x_8 + x_9) - (x_1 + 2x_2 + x_3) \quad (2)$$

Gradient vektors y virziena:

$$G_y = (x_3 + 2x_6 + x_9) - (x_1 + 2x_4 + x_7) \quad (3)$$

, kur  $X_i$  atbilst sekojošai 3x3 bildei apgabalam

$x_1$	$x_2$	$x_3$
$x_4$	$x_5$	$x_6$
$x_7$	$x_8$	$x_9$

Vertības  $G_x$  un  $G_y$  dot vektora modulios  $G$ , lai noteikt malas

$$G = \sqrt{(G_x^2 + G_y^2)}$$

**Filtrācija operācija tiek veikta veselam freimam, līdz tam brīdim kamer visi kustības pikseli neatrastītos 3x3 bildes apgabalam centrā un neatbilstotu  $G$  vērtībam katra piksēļa.  $G$  vērtību salīdzina ar malas sleksni  $E_t$ . Ja  $G$  ir lielāks nekā  $E_t$ , tad šo pikseli var uzskatīt par malas pikseli, citādi tas nav malas pikselis. Katram pikselim piešķir ( $E_i$ )**

$$e_i = \begin{cases} 1, & G \geq E_t \\ 0, & G < E_t \end{cases}$$

Vertība ir atkarīga no tā, vai tas ir malas pikselis vai ne. To apraksta sekojoši vienādojumi:

$$M_k = \left( \sum_i e_i^2 \right)^2$$

**Malas sleksnis  $E_t$  noteica katram freimam individuāli. Katram freimam ir vērtība  $M_k$ , kas ir malas enerģija. Sī malas enerģijas vērtību izskaitļo abam freimam – oriģinālajam un saspiestam.**

Kropļojumu vērtība  $D_{mot,k}$ , freima kustīgas daļas ir absolūtais lielums, kas ieguvam sekojoši:

$$D_{mot,k} = \frac{|M_{inp,k} - M_{out,k}|}{M_{inp,k}}$$

Kopīgais video kropļojumu lielums

**Kopīgo video kropļojumu vērtību video plusmai ieguvam ar relatīvo normalizēto kropļojumu svaršanu, kustības un kadru pikselim. Normālizāciju izskaitļo ar ar video plusmams maksimālo vērtību  $D_{mot,max}$ . Relatīvas kropļojumu kustības un kadru vērtības ieguldījums freimā, kas ieguvam no normalizēta kropļojuma. Video kropļojumi vērtību ieguvam šādi.**

$$D_{vid,k} = s_k \frac{D_{stl,k}}{D_{stl,max}} + (1 - s_k) \frac{D_{mot,k}}{D_{mot,max}}$$

, kur  $s_k$  ir attiecība starp kustības pikselim freimā  $s_k$  un kustības pikselim  $(1 - s_k)$ . Kopīga kropļojumu vērtība  $D_{vid}$ , ir kropļojumu kombinācija no visiem freimiem un noteica šādi:

$$D_{vid} = \left(\frac{1}{M}\right) \sum_k D_{vid,k}$$

, kur ir kopīgais kropļojumu skaits.

### Subjektīvais tests

Tika sataisīts eksperiments, kurā izmantoja 16 cilvēkus. Bija 6 scenas un novērtējums tika realizēts ar 10 ballu ballu skalu palīdzību un ar 10 punktu vājības skalu. Augsta atzīme nozīmē ka kropļojumi gandrīz nav manāmi., zema atzīme – krpļojumi ir gandrīz neciešami. Originālo bildi rāda visu laiku., kamēr katra saspiesta bilde būs atšķirīga. Eksperiments atkārti daudzreizi, pie tam brīva kartība.

Katrai bildes kompsijai izmanto 8 atšķirīgas kompresijas shemas: JPEG kompr., sub-band kompr., vektoru kvantizācija, bloku veidošana, 4 faktoru decimācija, ks balstas uz JPEG un citas. Ir daudz veidu kropļojumu. Dāžiem ir traipes, zvanes un abas kopā. Pēc visu atzīmju savakšanas, kas ieguti no skatītāja, izskaitļo videjo atzīmju vērtību katrai saspiestai bildei. Noapaļošanas shemai pirmo ņems saspiesto bildi, kurai ir visaugstākais novērtējums, bet tai bildei, kurai ir zemākais novērtējums bus noapaļota astota.. Lidzīgs shemas savam apmekinjiem izmanto kropļojumu vērtības un videja kvadrāta kropļojumu vērtības, pie ka saspiestai bildei, kura ieguva viszemāko atzīmi tiks noapaļota pirma un bilde ar augstāko atzīmi – astota.

Līdzīgs eksperiments tika sataisīts saspieštam video sekvencem. Novērtēšanas procesa piedalījās 15 cilvēki, kuriem standīts uzdevums – sadalīt plusmuy 8 daļas. Un novērtēt ar 5 punktu skalu. 5 kvalitātes daļas ir: augsta kvalitāte, kontrast, precizitāte, savietojamība. Originālo video plusmu rada pirmkārt un pēc tam visus pārējus saspieštos brīvā kartībā.

### Rezultāti

**Ir salīdzināti kropļojumi mērījumi, kas balstas uz vidējas – kvadratiskas kļūdas noteikšanas kompresētājam bildem / video. Starp sešiem bildes kadriem, tika savkts rezultāts vienā bilžu datu bāzē.**

Tab.1

Image Number	Subjective Scores	Our distortion measure	Mean square error	Mean abs. error	Max. deviation
1.	3.29	0.54	34.681	4.01	59
2.	1.67	0.77	51.096	6.045	135
3.	2.87	0.61	20.213	2.968	82
4.	4.54	0.36	8.635	2.110	23
5.	0.45	0.91	41.933	4.789	102
6.	3.83	0.47	14.289	2.567	41
7.	2.41	0.69	29.066	3.456	47

Ši tabula rāda vidējus rezultātus, kas ieguti no skatītājiem astoņiem kompresētām bildēm. Tur ir kropļojumi mērījumu rezultāti, vidējas kvadratiskas mērījumi, vidēja absolūta vērtība, maksimāla novirze. Redzamas, ka noapaļotas vērtības ir ieguti izmantojot kropļojumu mērījumi. Šī mērījumu rezultāti protams atšķiras no subjektīvam novērtējumam. Izrādas

,ka 6 bildem, kam izmantoja kropļojumu mērīšanai ir 100% pareizi ar korelāciju 0.95 attiecībā pret subjektīvo novērtējumu, kas izmantoja videjo kvadrātisko metodi.

Līdzīgas darbības tika taisīti 8 video plusmam. Rezultāti ir Tab. 2

Image Number	Subjective Scores	Our distortion measure	Mean square error	Mean abs. error	Max. deviation
1.	3.29	0.54	34.681	4.01	59
2.	1.67	0.77	51.096	6.045	135
3.	2.87	0.61	20.213	2.968	82
4.	4.54	0.36	8.635	2.110	23
5.	0.45	0.91	41.933	4.789	102
6.	3.83	0.47	14.289	2.567	41
7.	2.41	0.69	29.066	3.456	47

Izradas, ka izmantojot kropļojumu mērīšanas, rezultāti bija pareizi uz 96% pie 0.92 korelācijas, bet izmantojot vidējo kvadrātisko metodi rezultātu pareizība bija 57%, pie korelācijas 0,47 attiecība pret subjektīvo vērtēšanu.

### Nobeigums

Tika izstrādāts universāls līdzeklis, lai mērīt kropļojumus monohroma saspieštam bildem. Šī metode balstas uz to, ka lokāla kļūda nekorelē. Izradas, ka šī metode dot ļoti tuvus rezultātus salīdzinot ar cilvēka uztveres sistēmu.

Arī ir piedāvāta kropļojumu mērīšanas metode saspieštam video, kura sadala kompresijas kropļojumus uz kadru daļas efektiem un kustības daļu efektiem. Mēs pārbaudījām, ka šī kropļojumu mērīšanas rezultāti ļoti tuvi atbilst subjektīvai cilvēka uztverei.

## Augstas veiktspējas vektoru kvantēšanas mikroshēma (A High Performance Vector Quantisation Chip)

Demonstrētā mikroshēma domāta reāla laika runas vektoru kvantēšanas pielietojumiem. Mikroshēma tikusi ražota 1.2 mikronu CMOS tehnoloģijā un satur 38 000 tranzistoru. Tas aizņem tikai (4,8×4,6) mm<sup>2</sup> lielu laukumu un var izpildīt vairāk kā 80 miljonus reizināšanas/uzkrāšanas darbības sekundē. Mikroshēma pilnībā demonstrē ieguvumus no parasta specializēta procesora ar ļoti augstas integrācijas pakāpes arhitektūru izmantošanas.

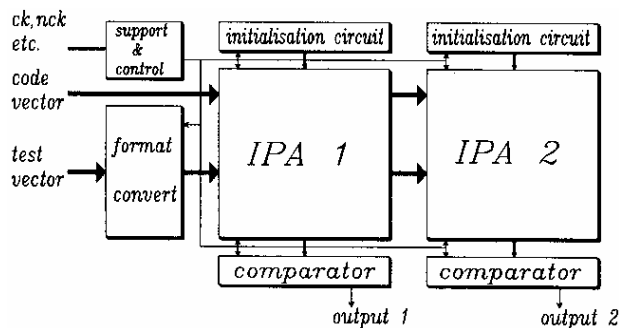
### Ievads

Vektoru Kvantēšana (VK) ir efektīva kodēšanas metode ar kuru datu saspiešana tiek panākta veicot paraugu sakrīšanas procesu. Tomēr nepieciešamā skaitļošanas sarežģītība ir ļoti prasīga, īpaši kad optimālai veiktspējai nepieciešama pilna VK kodu grāmatas meklēšana. Pēdējo gadu laikā tas mudinājis LAIP (ļoti augstas integrācija pakāpes) arhitektūras pētījumus, lai izmantotu VK sistēmas.

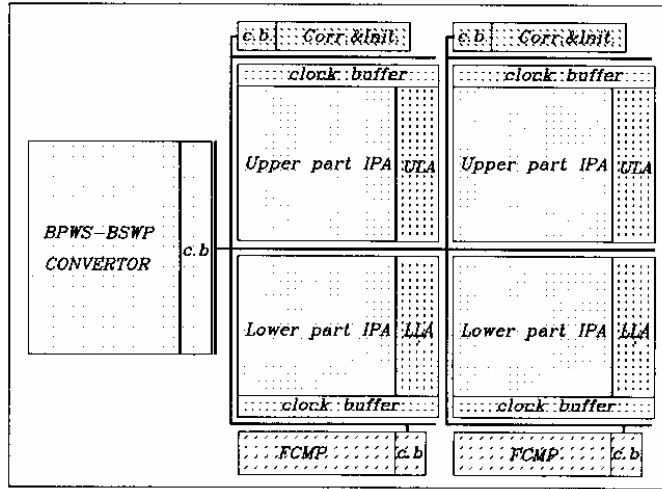
Šī dokumenta mērķis ir aprakstīt uz šādas arhitektūras bāzētu mikroshēmu uzbūvi. Rezultējošā ierīce demonstrē šādas arhitektūras modulāro dabu un tās derīgumu augstas veiktspējas sistēmu izstrādāšanā.

### Arhitektūra

VK mikroshēmu vispārīgā arhitektūra shematiski parādīta 1. att. Tas sastāv no divu vārdu paralēla, bitu virknes IPA ķēdēm un ar tām saistītajām komparatoru ķēdēm. Tas satur arī datu formatētāju kurš ļauj normālu vārdu virknes, bitu paralēlu datu ievadīšanu mikroshēmā. Tās tiek izmantotas kropļojumu mērīšanas aprēķinus un paraugu sakrīšanas meklēšanu, kas detalizēti aprakstīts citur. Saistītā atbalsta un vadības shēma ir minimāla tās pamatos esošās arhitektūras augstās regularitātes dēļ. Rezultējošās mikroshēmas plāns parādīts 2. att. Visi shēmas bloki atbilst 1. att. Diagrammā c un b ir sinhronizācijas buferis un FCMP attiecas uz komparatora shēmu. Augšējā un apakšējā bufera masīvi (ULA un LLA) tiek izmantoti sinhronizācijai.



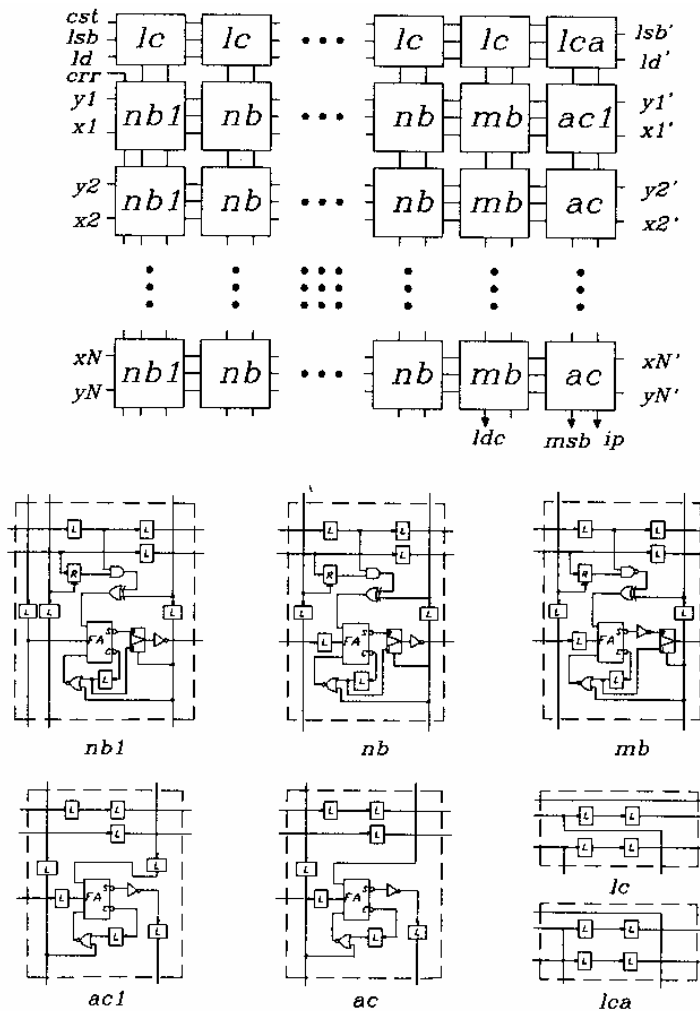
1. att. Mikroshēmas arhitektūra



2. att. Mikroshēmas plāns

Iekšējā Reizinājumu Masīva (IRM) Shēma

Divi iekšējā reizinājumu masīva bloki (3. att.) veido mikroshēmas centrālo daļu. Tie tiek kaskadēti tā ka izvadītie dati no pirmā var tikt izmantoti kā ieejas otrajam.



### 3. att. Iekšējā reizinājuma masīva un elementu ķēdes

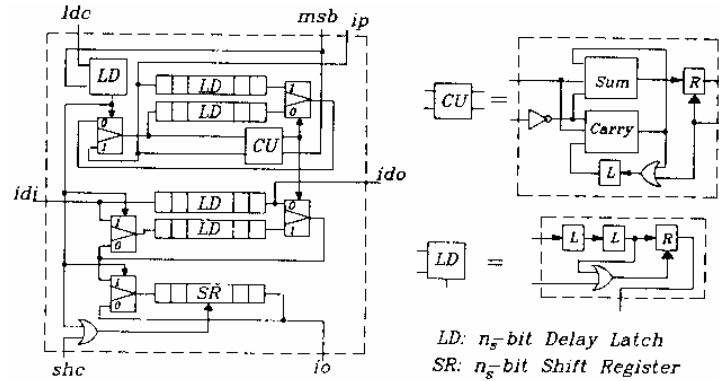
IRM sastāv no N virknes/paralēliem reizinātājiem, kur N ir vektora dimensija. Tas darbojas uz nepārtrauktiem bitu virknes, vārdu paralēliem ieejas vektoriem un skaitļo to iekšējos reizinājumus. Uz lidojošā vārda nogriešanas (noapaļošanas) ķēdes ir ietverti divi IRA. IRA ķēdes rēķina vektoru iekšējo reizinājuma (IR) vērtības formā  $(\sum x \bullet y + z)$ , kur x, y ir divi vektori ar dimensiju N un z ir konstante. Tas ir bāzēts uz novērojumu ka VK ir izplatītā kvadrātiskās kļūdas novērtēšana var tikt aizvietota ar iekšējo reizinājumu rēķināšanu.

#### Komparators

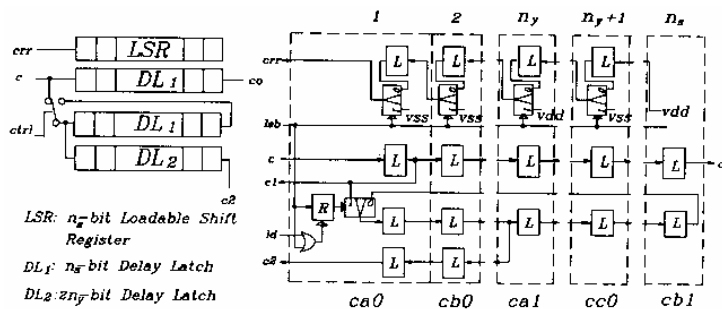
Komparatora ķēde tiek izmantota lai salīdzinātu visas iekšējo reizinājumu (kropļojumu) vērtības, lai noteiktu labāko sakritību. Katrs komparators satur četrus n bitu aiztures buferus, vienu n bitu nobīdes reģistru (n ir sistēmas vārda garums) un ķēdi vērtību salīdzināšanai, apzīmētu CU 4. att. Divas aiztures buferu kopas ir ietvertas, viena atjaunoto maksimālo IR vērtību glabāšanai un patreizējā IR ieejas vērtība no IRA augšā (caur ir), un otra ar tiem saistīto adresu glabāšanai. Nobīdes reģistrs (SR) tiek izmantots lai glabātu galīgo maksimālo IR adreses vērtību un tās izeja (caur io) tiek iespējota ar aktīvu AUGSTU signāla līmeni, kā parādīts.

#### Inicializācijas ķēde

Inicializācijas ķēdei ir divi mērķi. Vispirms, par cik IRA reizinātāji ir balstīti uz Baugh-Wooley papildkoda reizināšanas algoritmu, fiksēts sākotnējais parciālais reizināšanas algoritms, fiksētais sākotnējais reizinājums (pp)  $(2^{ns-1} + 2^{ny-1})$  jāpieskaita lai veidotu fmal reizinājumu, kur fly ir sākotnējais x un y datu garums. Otrkārt, konstante z, ir jāpieskaita parciālajam IR vērtībām lai ģenerētu rezultējošās modificētās IR vērtības formā  $(\sum x \bullet y + z)$ . Inicializācijas shēma parādīta 5. att.



### 4. att. Komparatora shēma

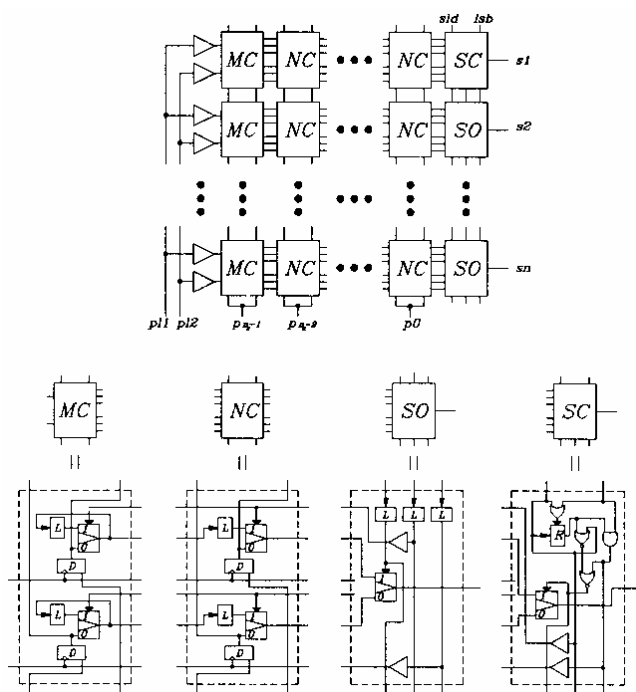


### 5. att. Inicializācijas shēma

Kā parādīts 5. att., sākotnējais pp, apzīmēts err (korekcija), tiek ģenerēts izmantojot n bitu ielādējamu pārbīdes reģistru. Konstantes vērtība (c), kura parasti atbilst y tiek ielādēta n bitu aiztures buferī, ko kontrolē lsb un ld (load) signāli. Tas tiek pareizi laikots, izmantojot papildus buferus un kļūst pieejami caur c3 IRA akumulatoram apakšā. Aizkavētā, nemainītā izeja, co, arī tiek apgādāta lai ļautu konstanšu ielādēšanu sekojošos IRA.

#### Datu Formāta Pārveidotājs

Daudzās ciparu signālu apstrādes (DSP) sistēmās dati tiek organizēti bitu paralēlā, vārdu virknes veidā (BPWS). Tamdēļ ir iekļauts datu formāta pārveidotājs, lai pārveidotu šo bitu virknes, vārdu paralēlā, laikā bīdītā formātā, kas nepieciešams IRA. Tas ir parādīts 6. att. kopā ar atbilstošu elementu loģikas shēmām.

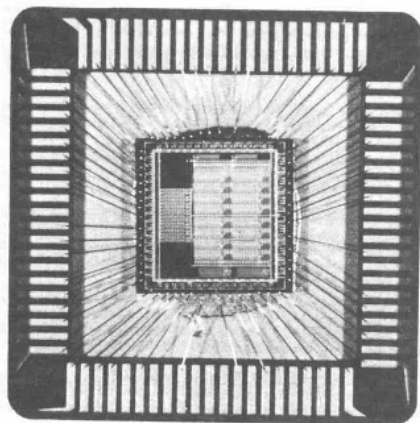


6. att. Datu pārveidošanas shēma

Shēma patiesībā sastāv no diviem identiskiem pārveidotājiem, kas ir acīmredzams no MC un NC elementu loģiskajiem aprakstiem. Tie ļauj bitu paralēlu datu ievadi vienlaicīgi ar bitu virknes datu izvadi. Bitu paralēli dati tiek ievadīti pārveidošanas shēmā caur datu līnijām, kuras apzīmētas ar p apakšā. Tas tiek laikā vadīts ar p11 un p12 signāliem alternatīviem ieejas vektoriem. Bitu virknes, vārdu paralēlas laikā vadītas izejas ir pieejamas pa labi un ir laikā vadītas lai sāktu izmantot Isb signālu. Vienlaicīgi signāls, sid, izvēlas izejas no atbilstošā pārveidotāja.

### VK Mikroshēma

7. att. parādīta VK mikroshēmas fotogrāfija, kas skaidri saskan ar plānu 2. att. Mikroshēmas izmēri ir 4.8x4.6 mm<sup>2</sup> un tā satur aptuveni 38 000 tranzistoru. Tā ražota izmantojot 1,2 mikronu divu līmeņu metāla CMOS tehnoloģiju un atrodas 84 kontaktu PGA formā. Patiesībā tiek izmantoti 58 kontakti.



### 7. att. VK mikroshēmas fotogrāfija

58 kontakti un to f-jas uzskaitītas zemāk.

**Barošana un Zeme** Tam ir kopā 8 kontakti, to skaitā 3  $V_{dd}$  (barošana) un 5  $V_{ss}$  (zeme).

**Datu ievade** Kopā 16 kontakti tiek izmantoti šim mērķim. No tiem 8 tiek izmantoti ieejas kodu vektoriem.

**Datu izvade** 16 kontakti tiek izmantoti datu vektoriem un koda vektoriem. Tie tiek izmantoti, kad divas vai vairākas mikroshēmas tiek kaskadētas, lai palielinātu rēķināšanas jaudu.

**Konstante un adrese** Šim f-jām vajadzīgi 6 kontakti divus no kuriem aizkavētajām izejas vērtībām.

**Sinhronizācija un vadība** Šim tiek izmantoti kopā 10 kontakti.

- divi divlāžu nepārklājošamies sinhronizācijas signāliem, ck1 un ck2.
- viens lsb un lsb0 signāliem (kaskadētām mikroshēmām)
- viens datu vektora izlādēšanas signālam (formāta pārveidošanai) un viens tā aizturētajai izvadei.
- divi 1 datu ielādes signāliem
- divi kontakti adreses izvades iespējošanas signāliem, viens katrai IRA mikroshēmai.

**Minimālās adreses izvadi** Divi izvadi tiek izmantoti kā adreses izvadi no diviem komparatoriem. Tās tiek izvadītas bitu virknē un iespējotas, kad iestādītas AUGSTĀ līmeņi.

**Kropļojumu vērtību izvade** Divi kontakti tiek izmantoti lai izvadītu iekšējo reizinājumu vērtības, bitu virknes veidā, kas izskatītas divos IRA. Tie tiek ģenerēti bitu virknes veidā un izmantoti mikroshēmu testēšanai.

## Veiktspēja

Izstrādātā mikroshēma ir domāta galvenokārt VK skaitļošanai runas pielietojumiem. Tas var strādāt ar frekvenci 40 MHz. Abi IRA var apstrādāt vektoru izmērus līdz 8. Sākotnējais datu garums,  $n_y$ , 8 biti,  $n_s$ , 12 biti. Tāpēc VK mikroshēmas veiktspēja ir sekojoša.

Pieņemsim, ka VK sistēmai ir grāmatas izmērs 1024 ieraksti. Datu temps, ko mikroshēma spējīga apstrādāt ir  $(fN/512n_s)$ , kur  $f$  ir sinhronizācijas frekvence. Ar  $f = 40$  MHz, maksimālais datu apstrādes temps ir apmēram 52K paraugu sekundē. Alternatīvi, ja mēs izmantojam standarta runas diskretizācijas frekvenci 8K sekundē, maksimālais kodu grāmatas izmērs, ko mikroshēma var apstrādāt ir  $(fN/4000n_s)$  ieraksti, tas ir, 6600 ieraksti var tikt apstrādāti reālā laikā. Šie rādītāji ir līdz 500 reižu lielāki nekā izmantojot TMS320 sērijas fiksētā p-ta DSP programmējamās mikroshēmas. Mikroshēmas enerģijas patēriņš pie 40 MHz ir aptuveni 600 mW.

## Secinājumi

Šajā dokumentā mēs esam aprakstījuši parastu VLSI mikroshēmu VK pielietojumiem. Tā ir bāzēta uz augstā pakāpē regulārām, konveijerizētām masīva arhitektūrām. Ar kopējo izmēru tikai  $(4.8 \times 4.6)$  mm<sup>2</sup> tā ir ļoti jaudīga veiktspējas ziņā. Tas savukārt demonstrē saistīto VLSI masīvu arhitektūru augsto veiktspēju.

## Būvējot Jūsu sistēmu (Building Your System) Testēšanai un mērīšanai vai rūpnieciskajai automatizācijai

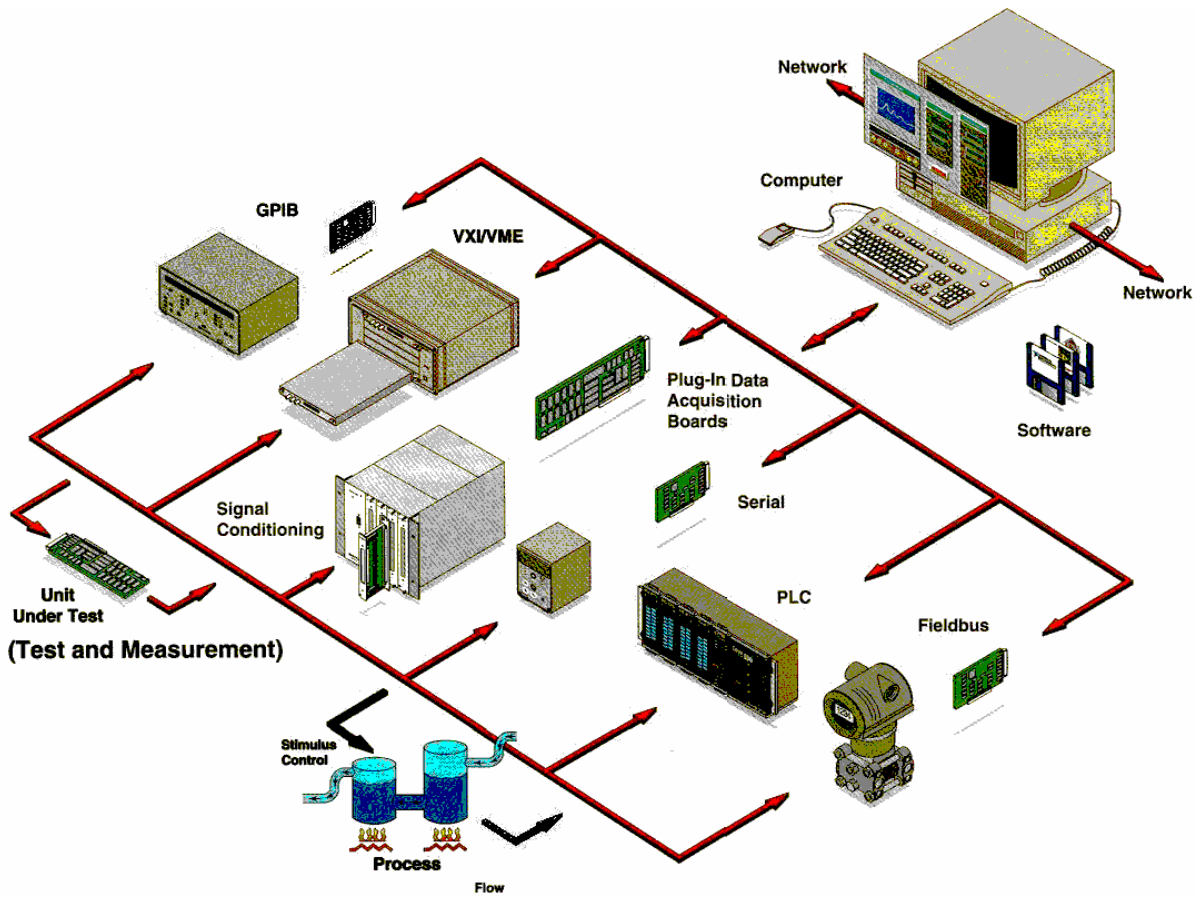
### Pielietojumi

Testēšana un mērīšana

- ATE
- Produkcijas testēšana
- Elektronikas testēšana
- Vibrāciju testēšana
- Akustiskā testēšana
- Dzinēju testēšana
- Pusvadītāju testēšana
- Telekomunikācijas
- Kalibrēšana
- Pētījumu un izstrādes rūpnieciskā automatizācija
- Procesu automatizācija
- Rūpnīcu automatizācija
- SCADA
- MMI
- Datu iegūšana
- Statistisku procesu vadība
- Laboratoriju automatizācija
- Darbgaldu vadība

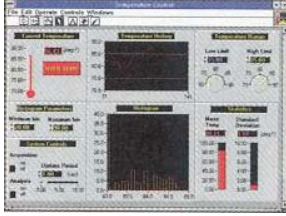
PC ir evolucionējuši par jaudīgām, rentablām skaitļošanas platformām kuras tiek izmantotas gan testēšanas un mērīšanas pielietojumiem un rūpnieciskās automatizācijas pielietojumiem. GPIB ir plaši pazīstami kā vispasaules standarts instrumentiem. Turklāt VXI un VXI plug & play piedāvā jaunas satraucošas iespējas testēšanas sistēmām. ACP tehnoloģijas un signālu saskaņošanas piederumu evolūcija ir palīdzējusi iespraužamajām datu iegūšanas platēm kļūt populāram un elastīgam risinājumam iegūšanas, novērošanas un vadības pielietojumu daudzveidībai. PC ir vadošā izvēle PLC programmēšanai un cilvēka-mašīnas interfeisa (MMI) platformām. Savienojamība ar rūpnieciskajiem tīkliem, vai nu vienkāršo RS-485 vai sarežģītākām lauka kopnes (*fieldbus*) sistēmām, tādām kā FOUNDATION Fieldbus, dod jums pieeju ierīcēm diapazonā no vienkārša I/O līdz gudriem raidītājiem un citiem procesu instrumentiem. Jūs varat izmantot PC visā jūsu uzņēmuma automatizācijas un biznesa vidē. Šodien Jūs varat nostāties vienā rindā ar rūpniecisko standartu datoru platformām un jaukt jūsu sistēmā instrumentu tipus, lai pilnībā izmantotu jums pieejamās izvēles. Šī kataloga programmatūras rīki ļauj katru instrumenta tipu vienlīdz viegli izmantot un integrēt jūsu sistēmā, tā ka jūs varat optimizēt jūsu risinājumu lai tas atbilstu jūsu pielietojuma cenas un veiktspējas prasībām.





(Rūpnieciskā automatizācija)

## Jūsu sistēmas sastāvdaļas



### Programmatūra

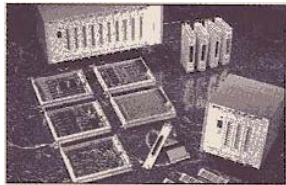
Mūsu programmatūra aptver pielietojumus no pētniecības laboratorijas līdz ATE. Mūsu programmu programmatūras paketes (Lab VIEW un LabWindows/CVI) nodrošina integrētus rīkus iegūšanai, analīzei un vizualizācijai kā arī savienojamību caur tādiem interfeisiem kā DDE un TCP/IP. Instrumentu bibliotēku draiveri sniedz vienkāršu un lietotājam draudzīgu instrumentu programmēšanas un darbošanās interfeisu. Augsta līmeņa analizējošās bibliotēkas ietver f-jas nolikšanas/izkrišanas lēmumu, statistikas, jaudas spektra un citu rēķināšanai. Papildus mūsu programmatūrai mums ir arī GPIB, datu iegūšanas un VXI draiveru programmatūra un utilīti, lielākā daļa no kuriem ir iekļauti kopā ar atbilstošu aparatūru. Ir arī kreisie draiveri PLC un citām rūpnieciskajām ierīcēm kuri darbojas ar mūsu pielikumu programmproduktiem.



### Iespaužamās datu iegūšanas plātes (DAQ)

**DAQ plātes sprauž tieši vajadzīgajā datora paplašinājuma kopnē, tādā kā PCI, PCMCIA, ISA vai Macintosh NuBus. Raidzvēvējus vieno tieši vai caur signālu saskaņošanas moduļiem ar DAQ plati, kura uzver un/vai ģenerē analogus un ciparu signālus. Viena plāte bieži var izpildīt veselu virkni f-ju, ieskaitot AC pārveidošanu, CA pārveidošanu, ciparu I/O un skaitītāja/taimera operācijas. Iespaužamajām DAQ plātēm bieži ir zemāka cena un lielāka elastība kā atsevišķiem instrumentiem.**

NI-DAQ draiveru programmatūra ir ietverta kopā ar visām iespaužamajām DAQ plātēm.



### Signālu saskaņošana

Vienā vai jūsu strādājat ar temperatūru, spiedienu, plūsmu vai 4-20 mA strāvas ķēdēm, mūsu signālu saskaņošanas moduļi nodrošina jūsu DAQ plati ar šiem signāliem. Signālu saskaņošana pielāgo reālās pasaules signālus, lai tie atbilstu jūsu DAQ I/O specifikācijām. Moduļi ir pieejami signālu multipleksēšanai, izolācijai, pastiprināšanai, vāju savienojumu kompensācijai, izlabošanai, filtrēšanai un daudz kam citam. Visi moduļi ir izstrādāti un testēti lai strādātu optimāli ar mūsu DAQ plātēm, tādējādi ietaupot

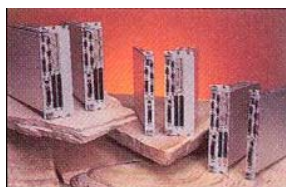
jums laiku un naudu.



### GPIB

GPIB instrumenti ir testēšanas un mērīšanas sistēmu vispasaules standarts. Jūsu datoram vajadzīgs GPIB interfeiss. Gandrīz katram PC ir iespējama izvēle – PCI, PCMCIA, ISA Macintosh NuBus un daudzas citas – un darbstaciju platformai – Sun, HP, IBM, DEC un citi. Jaunākie GPIB interfeisi atbilst IEEE 488.2 standartam un SCPI rūpnieciskajam standartam un strādā ar datu pārsūt. ātrumu lielāku kā 1MB/s. Programmatūras izvēles ir visplašākās, sākot no mūsu pielikumu programmatūras paketēm ar augsta līmeņa instrumentu draiveriem līdz rūpnieciskajā standarta NI-488.2 draiveru programmatūrai

parastai programmēšanai izmantojot operētājsistēmu un valodu pēc jūsu izvēles.



### VXI

Modulāri, uz kartēm bāzēti VXI instrumenti var milzīgi samazināt sistēmas izmēru un palielināt veiktspēju, ar izturīgu iepakojumu lielisks pārvietošanai vai rūpnieciskiem pielietojumiem. Tāpat kā GPIB, VXI vajadzīgs aparatūras interfeiss ar datoru. Iespējams liels daudzums izvēļu. VXI ir ļoti svarīga programmatūra. Programmatūras izvēles ir visplašākās, sākot no mūsu pielikumu programmatūras paketēm ar augsta līmeņa instrumentu draiveriem līdz rūpnieciskajā standarta NI-VXI draiveru f-jām ērtai

programmēšanai.



### Virknes un rūpnieciskie sakari

Daudzas ierīces, kuras izmanto rūpnieciskajos pielietojumos, sakariem izmanto RS-485/422 vai RS-232 virknes interfeisus. Mūsu LabVIEW un LabWindows programmatūras produkti ietver funkciju bibliotēkas komunikācijai ar šīm virknes iekārtām, kuras izmanto vienkāršus ASCII protokolus caur standarta datora virknes portiem. PLC un citām iekārtām, kuras izmanto sarežģītākus rūpniecisko sakaru tīklus, tādas kā *fieldbus*, ir nepieciešami speciāli aparatūras un programmatūras interfeisa draiveri. Šie programmatūras draiveri ir pieejami caur Lab VIEW un LabWindows/CVI, kā arī no valodām, kuras var izsaukt Windows dinamiskās bibliotēkas, piemēram Visual

Basic, Visual C++ un Delphi.

## Jūsu instrumentu programmatūras komponentes un analīzes programmatūra

### Operētājsistēma

Izstrādājot mūsu produktus dažādām OS, mēs dodam jums iespēju izvēlēties rēķināšanas platformu kura vislabāk apmierina jūsu vajadzības. Šogad mēs esam pievienojuši Windows 95 un Windows NT dabisko 32 bitu iespēju lielākajai daļai mūsu pielietojumiem un draiveru programmatūras produktiem. Mēs attīstām un uzlabojam visus mūsu produktus lai izmantotu jaunākās Windows operētājsistēmu kopas, Mac OS, Solaris un HP-UX iespējas.

### Draiveru programmatūra

Mūsu ierīču draiveru programmatūra, NI-DAQ, NI-488.2 un NI-VXI/VISA ir pārnesama starp operētājsistēmām. Programmatūra dod jums iespēju pārvietot programmas, kas izstrādātas ar rūpniecisko standartu valodām, tādām kā C vai Basic un de facto standartu vidēm, tādām kā Lab VIEW un LabWindows/CVI uz datoriem ar dažādām operētājsistēmām. Uz reģistriem bāzētām ierīcēm, tādām kā iespraucamajām DAQ platēm, draiveru programmatūra izslēdz sarežģītības sakarā ar reģistra līmeņa programmēšanu.

### Pielikumprogrammu programmatūra

Mūsu pielikumprogrammu programmprodukti kombinē labākos programmatūras izstrādes rīkus datorindustrijā ar mūsu draiveru programmatūru un labāko instrumentu tehnoloģiju, lai sniegtu programmatūras vides un izstrādes rīkus, kuri ietaupīs jums pielikumu izstrādes laika stundas, ja ne dienas. Pašlaik mēs piedāvājam piecus pielikumproduktus, katrs no kuriem darbojas uz dažādām operētājsistēmām un apmierina daudzas instrumentu sistēmu izstrādātāju vajadzības. Diviem no mūsu pielikumprogrammu produktiem, Lab VIEW un LabWindows/CVI, ir virkne pievienojamu aprīkojumu tādām operācijām kā sakari ar SQL datu bāzēm un statistisku procesu vadība.

Lietotāja tips	Programmatūra
Neprogrammētājs	VirtualBench
Excel	Measure
Visual Basic	ComponentWorks
C	LabWindows/CVI
Graphical	LabVIEW

### Instrumentu draiveri

**Instrumentu draiveri nodrošina rīkus, lai izstrādātu sistēmu bez nepieciešamības zināt kā programmēt jūsu instrumentus. Instrumentu draiveri satur augsta līmeņa f-jas konkrētu instrumentu vadību. Ar daudzautomātu instrumentu bibliotēkām mūsu Lab VIEW un LabWindows/CVI programmatūrai, jūs varat viegli veidot pielikumus ar instrumentiem, kas vislabāk atbilst jūsu vajadzībām. Mūsu instrumentu bibliotēku tehnoloģijas veido VXI plug&play programmatūras struktūras kodolu.**

### Datu analīze ar LabVIEW, LabWindows un Signal Processing komplektu

Iebūvētā datu analīze attiecas uz datu apstrādi un analīzi pēc iegūšanas. Jūs izmantojat analīzes f-jas lai filtrētu signālu lai likvidētu troksni, labojat nepareizus datus bojātas aparatūras dēļ vai kompensējat apkārtējās vides efektus tādus kā temperatūra vai mitrums. Jūs varat iegūt citu derīgu informāciju, izmantojot analīzes f-jas lai pielāgotu datus vajadzīgajam diapazonam, transformēt tos frekvenču apgabalā vai veiktu virkni citu operāciju ar tiem. Nav nepieciešams formātēt jūsu datus pēc tam kad tie iegūti – iebūvētās analīzes f-jas ir izstrādātas lai strādātu vienlaicīgi ar datu iegūšanas f-jām. Attēlošanas grafiki domāti “reālā laika” veikspējai, tā ka operatori var redzēt analīzes f-jas datu iegūšanas laikā.

### Iebūvētās datu analīzes bibliotēkas

- Mērīšana
- Signālu ģenerēšana
- Signālu apstrāde
- Statistiskā analīze
- Atgriezeniskā analīze
- Diapazona mērogošana

### Skaitliskā analīze un modelēšana ar HiQ

Skaitliskā analīze attiecas uz matemātikas klasi – skaitliskās metodes, kuras ir optimizēti algoritmi skaitliskiem aprēķiniem un fizisku sistēmu modelēšanai. Sistēmu modelēšana ietver lineāro algebru, vienādojumu sistēmu risināšanu, diferenciālvienādojumus, vienādojumu sakņu atrašanu, *data fitting*, skaitlisko integrēšanu, signālu apstrādi un daudzas citas operācijas. HiQ ietver bibliotēkas šo un daudzu citu problēmu risināšanai.

# **Skaitliskās analīzes bibliotēkas**

- Lineārā algebra
- Nelineārās sistēmas
- Parastie diferenciālvienādojumi
- Sakņu atrašana
- Optimizācija
- Cita skaitliskā analīze

## Programmatūras pārskats (Software Overview)

National Instruments programmatūras risinājumi

Programmatūrai ir vitāla loma automatizētu datu savākšanas un instrumentu vadības sistēmu izstrādē.

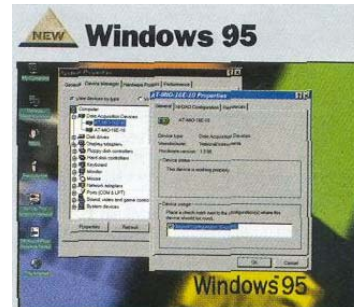
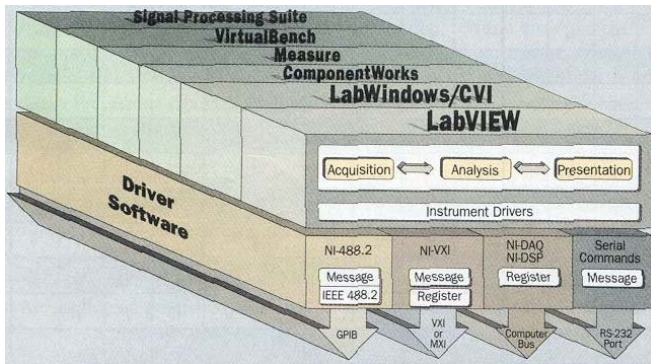
Programmatūra, ko jūs izmantojiet šajās sistēmās ietver plašu funkcionalitātes apgabalu, no ierīču draiveriem specifisku aparatūras interfeisu vadībai līdz pielikumu programmatūras paketēm pilnīgu sistēmu izstrādei.

Instrumentu sistēmu izstrādei izmantotās programmatūras kvalitāte un elastība galā nosaka tās kopējo kvalitāti un derīgumu.

Kā netieši norāda mūsu lozungs, "Programmatūra ir instruments", mēs ne tikai apzināties programmatūras svarīgumu, bet mēs paaugstinām programmatūru kā jūsu instrumentu sistēmas pamatu. Šī kataloga sekcija satur informāciju par visiem National Instruments piedāvātajiem programmproduktiem, ieskaitot plašu datu analīzes un vizualizācijas programmatūras klāstu. Šajā pārskatā mēs aprakstām kā mūsu programmatūras arhitektūra iekļauj katru mūsu produktu. Jūs varat izmantot tos atsevišķi vai kopā, lai būvēt jūsu instrumentu sistēmu.

### Virtuālo instrumentu programmatūra

National Instruments bija virtuālo instrumentu celmlauzis 10 gadus atpakaļ ar LabVIEW ieviešanu. Kopš sākotnējā LabVIEW laidiena 1986. mūsu uzticēšanās instrumentu programmatūrai, uzlabojot un pievienojot jaunus produktus mūsu virtuālo instrumentu komplektam, ir pastāvīgi augusi. Pašlaik mēs piedāvājam vairāk kā 30 programmatūras produktu uz dažādām platformām lai nodrošinātu pārbaudītus virtuālo instrumentu produktivitātes ieguvumus. Papildus LabVIEW, rūpniecības vadošajai grafiskās programmēšanas sistēmai, mums ir 3 produkti vizuālās izstrādes produktu līnijā - LabWindows/CVI, C izstrādes vide virtuālajiem instrumentiem; ComponentWorks, Visual Basic OLE komponentu komplekts un Measure, pievienojumprogramma Excel datu iegūšanai tieši uz izklājlapas. Jauns šogad ir VirtualBench, virtuālo instrumentu kopa izmantošana ar mūsu DAQ aparatūru. VirtualBench ir atsevišķu pielikumu kopa, pilnībā izstrādāta LabVIEW, kura ietver osciloskopu, ciparu multimetru, f-ju ģeneratoru, datu savācēju un dinamisko signālu atbildētāju.



Arhitektūra Dabiskā 32 bitu pielikumu un draiveru programmatūra

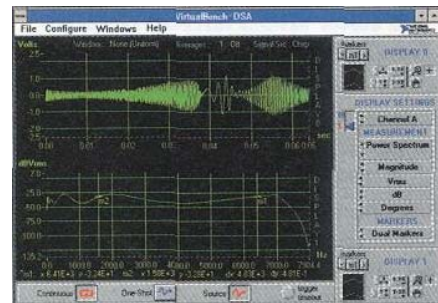
- LabVIEW, LabWindows/CVI, Component Works, Measure, VirtualBench un HiQ pielikumu programmatūra
- NI-DAQ, NI-488.2 un NI-VXIATS draiveru programmatūra

### Plug-and-Play aparatūra

- DAQ, GPIB, VXI un virknes I/O interfeisi

### Datu analīzes un vizualizācijas programmatūra

Mēs piedāvājam arī pilnu datu analīzes un vizualizācijas produktu līniju izpētei, simulācijai, virtuālajiem instrumentiem un rūpnieciskajai automatizācijai. Veidojot mūsu ilglaicīgo uzticību virtuālo instrumentu programmatūrai mēs esam izstrādājuši datu analīzes un vizualizācijas programmatūru 10 gadus. Šogad mēs atkal mūsu datu analīzes un vizualizācijas produktu līnijā ieviešam jaunu programmatūru – HiQ priekš Windows 95 un Signal Processing komplektu. Citi mūsu produkti ir HiQ priekš Macintosh, LabVIEW, LabWindows/CVI un virkni iekļaujamo aprīkojumu LabVIEW un LabWindows/CVI.



## Mūsu programmatūras stratēģija

Mūsu programmatūras privilēģija National Instruments ir piedāvāt pilnu instrumentu un datu analīzes programmatūras līniju, kas aptver plašu funkcionalitātes kopu pielikumiem sākot ar ļoti vienkāršiem līdz ļoti sarežģītiem. Mēs esam iecerējuši, ka mūsu programmatūra atbildīs lielajai lietotāju iemaņu līmeņu un izvēļu atšķirībai. Un, mūsu instrumentu programmatūrai, mēs esam cieši integrējuši mūsu pielikumu programmatūru ar mūsu iekārtu draiveru programmatūru un aparatūru, lai nodrošinātu maksimālo veiktspēju, lietošanas vienkāršību un gludo pārejas ceļu nākotnē.

National Instruments virtuālo instrumentu programmatūras pamats ir mūsu iekārtu draiveru programmatūra - NI-488.2 GPIB aparatūrai.

Ar mūsu iekārtu draiveru programmatūru jūs varat programmēt un vadīt jebkuru no mūsu aparatūras produktiem no tādas programmēšanas valodas kā C vai Basic, vai no pielikuma programmatūras paketes. Katrs no iekārtu draiveriem ir izstrādāts tā, lai maksimizētu programmēšanas elastību un datu caurlaidspēju. Vēl būtiskāk, katram iekārtas draiverim ir kopīgs pielikumu programmēšanas interfeiss (API), tāpēc, lai kāda būtu jūsu skaitļošanas vai operētājsistēmas platforma, pielikumu programmas, kuras jūs radāt ir pārvietojamas.

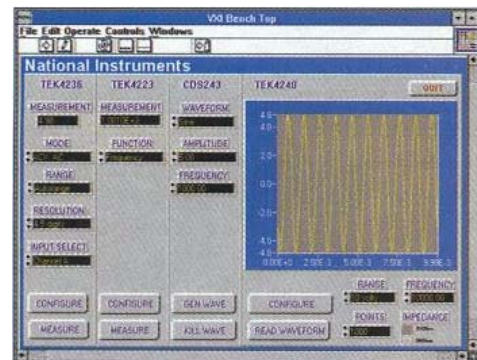
Mūsu programmatūras sistēmas arhitektūras virsotnē atrodas pielikumu programmatūra LabVIEW, LabWindows/CVI, ComponentWorks, Measure, un VirtualBench. Mūsu rūpniecības vadošā pielikumu programmatūra ir būvēta uz mūsu iekārtu draiveru programmatūras, un tāpat kā mūsu draiveru programmatūra, ir pārvietojama starp dažādām operētājsistēmām. LabVIEW un LabWindows/CVI apgādā jūs ar visiem programmatūras ražošanas rīkiem, kas vajadzīgi lai integrētu jūsu instrumentu sistēmu izmantojot darbderīgus, standarta datorus un instrumentācijas aparatūru.

Virtual Instrumentation Software	Platform	Acquisition							Analysis					Presentation				Page	
		GPIB	VXI	MD, DA, DIO	SCXI	Serial	Instrument Drivers	PLC Drivers	HiQ Link	DSP	Statistics	Array Arithmetic	Curve Fitting	GUIs	Graphs/Charts	Hardcopy	File I/O		Intrprocess Communication
LabVIEW – graphical programming	Windows 95/3.1	✓	✓	✓	✓	✓	✓	✓	✓	✓*	✓	✓	✓	✓	✓	✓	✓	✓	11
	Windows NT	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	11
	Macintosh	✓	✓	✓	✓	✓	✓	✓	✓	✓*	✓	✓	✓	✓	✓	✓	✓	✓	11
	HP-UX	✓	✓			✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	11
	Sun	✓	✓	✓		✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	11
LabWindows/CVI – C programming	Windows 95/3.1	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	13
	Windows NT	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	13
	Sun	✓	✓			✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	13
ComponentWorks – Visual Basic	Windows 95	✓		✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	16
Measure – Excel	Windows 95/3.1			✓		✓				✓	✓		✓	✓	✓	✓	✓	✓	16
VirtualBench	Windows 95/3.1			✓	✓				✓	✓			✓	✓	✓	✓	✓	✓	16
Signal Processing Suite	Windows 95/3.1/NT			✓					✓				✓	✓	✓	✓	✓	✓	16
	Macintosh			✓					✓				✓	✓	✓	✓	✓	✓	16

\* Vada arī DSP aparatūra liela ātruma analīzei.

1. tabula Virtuālo instrumentu programmatūras iespējas

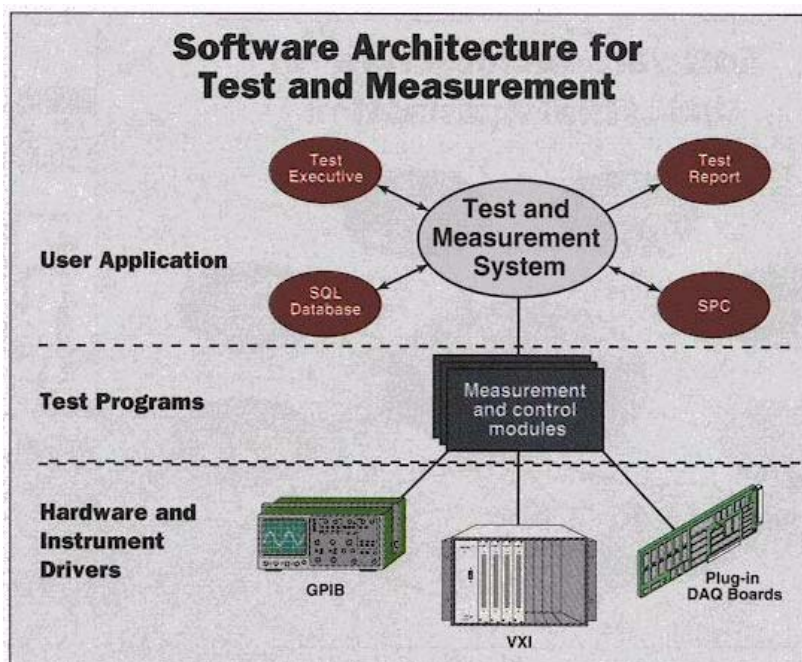
Component Works sniedz Visual Basic programmētājam datu iegūšanas, analīzes un prezentācijas rīkus, kas vajadzīgi instrumentu sistēmu būvēšanai. Measure ļauj vienkārši iegūt tieši uz izklājlapas, izmantojot mūsu DAQ plates, vai jūsu pašu virknes instrumentus. Un, VirtualBench sniedz jums lētu virtuālo instrumentu kopu, kas pārvērš jūsu PC par instrumentācijas ēvelsolu.



## Programmatūras, kas jums ir vislabāk piemērota, izvēlēšanās

Jūsu programmatūras vajadzības ietekmē daudzi faktori, ieskaitot pielikumu vajadzības, datora aparātūra un operētājsistēma, un jūsu instrumentu aparātūra. Jūsu izvēlētajai programmatūrai jābūt pietiekoši daudzveidīgai lai piemērotos dažādām datoru arhitektūrām un dažādiem instrumentiem un datu savākšanas iekārtām. Minimāli, jūsu pielikumu programmatūrai jāizpilda kaut kāda datu iegūšana, analīze un prezentācija. Un, ja jūsu sistēmai ir GPIB, VXI vai RS-232 instrumenti, jums būs vajadzīga arī instrumenta draiveru bibliotēka. Citas sistēmas prasības varētu diktēt ātru izpildīšanu, lielu kanālu skaitu vai reāla laika signālu apstrādi.

Izvēloties programmatūru, jums būtu jāsaprot programmatūras izstrādāšanas filozofija ar jūsu iemaņām un priekšrocībām. Vai programmatūra ir būvēta un atklāta arhitektūras? Vai programmatūra prasa programmēšanas iemaņas? Vai pielikumu programmas, kas izstrādātas ar šo programmatūru ir pārvietojamas starp dažādām skaitļošanas platformām? Atbildes uz šiem un citiem saistītajiem jautājumiem nosaka kuras programmatūras īpatnības ir jums svarīgas.



National Instruments pielikumu programmatūras produktu kopas uzrunā jūsu dažādās vajadzības un priekšrocības. Mūsu grafiskās un vizuālas programmēšanas rīki piedāvā dažādas rīku kopas, kuras jūs varat izmantot gandrīz jebkuram pielikumam. Kombinējot tehnoloģiju no datorindustrijas un instrumentu industrijas, mēs jums piedāvājam daudzas spēcīgas un elastīgas universālus programmproduktus, piemērotus jūsu vajadzībām un priekšrocībām.

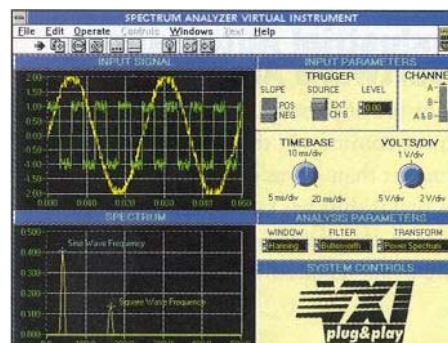
## Testēšanas un mērīšanas programmatūras arhitektūra

Mērīšanas programmatūra sastāv no 3 galvenajiem elementiem – lietotājpielikums, testēšanas un mērīšanas programmām un instrumentu draiveriem. Izmantojot šo modeli, programmatūras izstrādātājs var viegli identificēt nepieciešamos ATE sistēmas testēšanas un mērīšanas izstrādes posmus. Mūsu programmproduktus var izmantot jebkuram vai visiem izstrādes posmam. Viens no galvenajiem mūsu virtuālo instrumentu programmproduktu izmantošanas ieguvumiem ir, ka jūs izmantojat atsevišķu, saturīgu izstrādes rīku visiem 3 izstrādes etapiem. Un, ar jaunākajām LabVIEW un LabWindows/CVI versijām, jūs varat izsaukt LabWindows/CVI instrumentu draiverus no jūsu LabVIEW blokshēmas.

## Instrumentu draiveri

Visu testēšanas un mērīšanas sistēmu kodols ir instrumentu draiveri. Instrumentu draiveri, faktiski, ir vissvarīgākais mūsu virtuālo instrumentu programmatūras arhitektūras elements. Instrumentu draiveri radikāli samazina programmatūras ražošanas izmaksas, jo izstrādātājiem nav jāpavada laiks programmējot viņu instrumentus. Draiverus var arī atkārtoti izmantot plašā sistēmu un konfigurāciju lokā.

LabVIEW un LabWindows/CVI instrumentu bibliotēkās ir vairāk kā 500 instrumentu draiveri instrumentiem no vairāk kā 45 izstrādātājiem. Un, pateicoties instrumentu ražotāju draiveru izstrādāšanai VXI



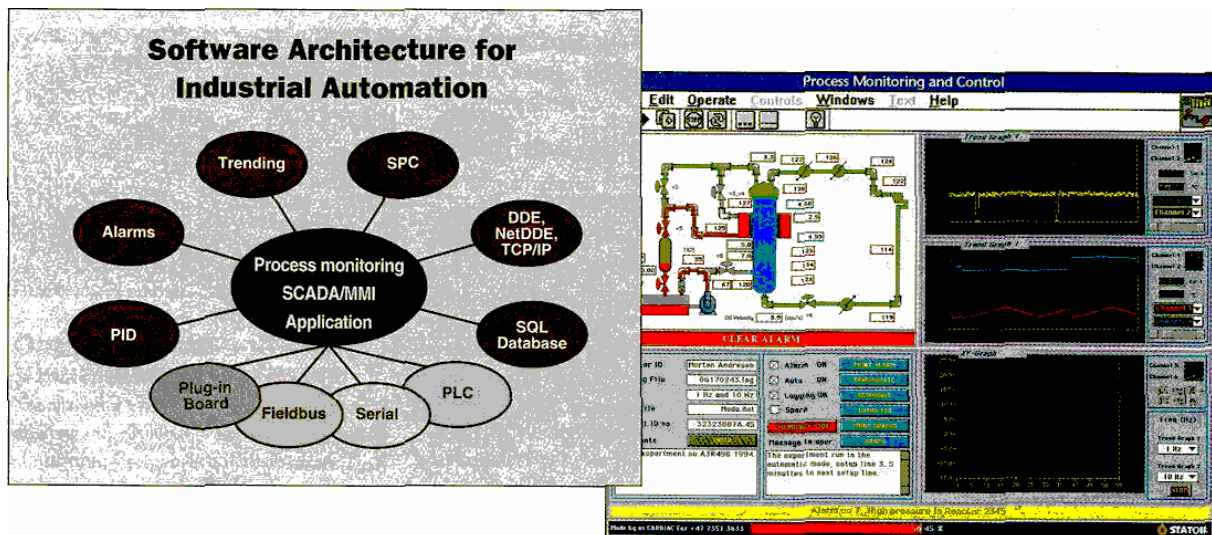
plug&play Systems Alliance kā arī mūsu Instrument Library Developer programmai, bibliotēka nepārtraukti aug ar ātrumu aptuveni 100 jauni draiveri gadā. Katrs draiveris tiek piegādāts izejas koda veidā, tāpēc jums ir iespēja radīt jūsu pašu virtuālos instrumentus. Mūsu virtuālo instrumentu koncepcija sniedz jums iespēju noteikt jūsu pašu instrumentu funkcionalitāti ar programmatūru. Kombinējot dažādu instrumentu funkcionalitāti, jūs varat precīzi radīt jūsu pašu instrumentus.

VXI plug&play Systems Alliance ir apstiprinājusi gan LabVIEW, gan LabWindows/CVI kā standarta izstrādes platformas instrumentu draiveru izstrādei uz GWIN un WIN karkasa.

## Rūpnieciskās automatizācijas programmatūras arhitektūra

PC izmantošana rūpnieciskās ražošanas automatizācijas tirgū ir daudz izplatītāka nekā tikai dažus gadus atpakaļ. Liela izmaiņa ir notikusi, ar kuru slēgtās patentētās automatizācijas sistēmas ir nomainītas ar atklātām programmatūras vidēm, standarta datoriem un standarta mērīšanas un vadības iekārtām. Pateicoties mūsu atvērtajai aparatūras un programmatūras arhitektūrai, mēs esam bijuši spējīgi uzlabot gan mūsu aparatūras, gan programmatūras produktus, lai tie apmierinātu rūpnieciskās automatizācijas tirgus prasības. Mūsu virtuālo instrumentu stratēģija piedāvā arī zināmu iekārtu apmaiņas līmeni, kas parasti neasociējas ar procesu un rūpnīcu automatizāciju.

National Instruments programmatūras modelis rūpnieciskajai automatizācijai sastāv no kontroles un datu iegūšanas (SCADA) pielikuma, kas komunicējas ar vairākiem programmatūras moduļiem, kuri veic specifisku procesu novērošanas un kontroles f-jas. Dažos pielikumos, SCADA programmatūra var f-cionēt tikai kā interfeiss ar sistēmu, šajā gadījumā programmatūru parasti uzskata par cilvēka-mašīnas interfeisu (MMI). Mūsu modelī, LabVIEW, LabWindows/CVI un Visual Basic kalpo par izstrādes vidēm SCADA vai MMI programmatūrai.



## Datu analīze un vizualizācija

National Instruments ir daudz ieguldījis LabVIEW un LabWindows datu analīzes un vizualizācijas rīkos. Šogad, mēs esam pievienojuši mūsu datu analīzes un vizualizācijas līnijai jaunu produktu. HiQ priekš Windows ir interaktīva problēmas risinoša vide matemātikai, zinātnei un inženierijai. HiQ papildina mūsu LabVIEW un LabWindows/CVI programmatūras stratēģiju, pievienojot mūsu zinātnisko un inženieru risinājumu pielikumu modelim divas svarīgas komponentes.

## Automatizācijas rūpnieciskā ievadizvade

Kā daļu no mūsu rūpnieciskās automatizācijas programmatūras stratēģijas, mēs esam iecerējuši, ka mūsu programmatūra saskaņoti darbosies ar faktiski jebkuru I/O iekārtas tipu. Ar vai nu mūsu grafiskajiem vai vizuālajiem izstrādes rīkiem, jūs varat koordinēties ar mērinstrumentiem, PLC, tiešām datu iegūšanas iekārtām un citiem programpielikumiem, kas tieši sazinās ar I/O iekārtām. Elastīgās I/O iespējas ļauj viegli pievienot lētu aparatūru eksistējošām sistēmām.



## **DAQ pārskats (DAQ Overview)**

### Ievads

National Instruments ražo plašu DAQ produktu izvēli, ar kuriem jūs varat izstrādāt virtuālos instrumentus plaša signālu diapazona mērīšanai un vadībai gan testēšanas, gan mērīšanas un rūpnieciskās automatizācijas pielikumiem. Mūsu produkti ietver iespraucamas plātes visām populārajām ISA, EISA, PCMOA, PCI un Macintosh NuBus kopnēm; ārējos paralēlā porta DAQ produktus; un signālu saskaņošanas aksesuārus. Šie aparatūras produkti ir izstrādāti un pārbaudīti ar mūsu pielikumu un draiveru programmatūru priekš Windows 95/NT/3.1, NEC Windows, DOS, DOS/V, NEC DOS, Mac OS un Sun Solaris operētājsistēmām. Lietojiet sekojošās diagrammas un tabulas, lai izvēlētos saviem pielietojumiem labākos DAQ produktus. Dziļākam, datorizētākam DAQ sistēmu konfigurēšanas izklāstam, zvaniet National Instruments un palūdziet DAQ Designer brīvi pieejamu kopiju.

### DAQ Designer

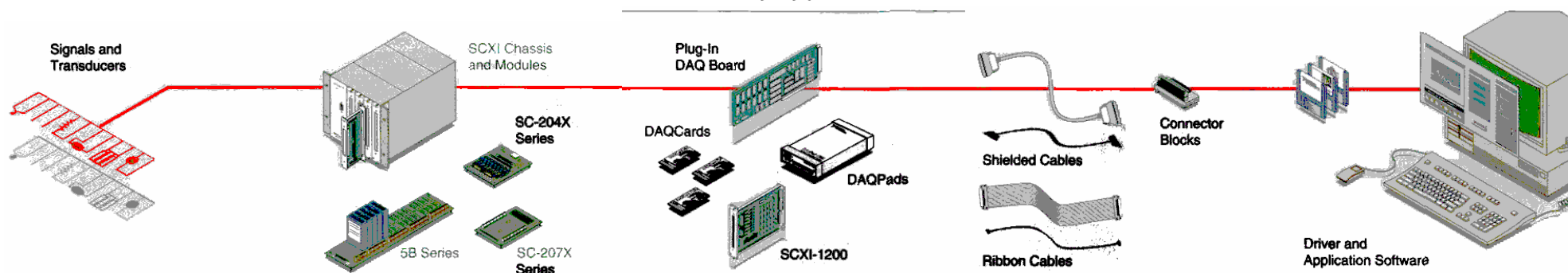
ir uz Windows bāzēta, datorizēta sistēmas konfigurācijas utilīta, kas palīdz izvēlēties no lielā National Instruments DAQ produktu klāsta. Jūs savstarpēji iedarbojaties ar DAQ Designer caur viegli lietojamiem grafiskiem paneliem, lai definētu svarīgas pielikumu prasības un lai indetificētu iespraucamās plātes, programmatūru, signālu saskaņošanas aparatūru, termināļu blokus, savienotājus un citus piederumus, kas vislabāk piemēroti jūsu pielietojumam.

### Produkti, kas domāti datu iegūšanai

Lai pārliecinātos, ka jūsu nolāsiņumi ir precīzi un, ka jūs varat piedabūt savu sistēmu darboties ar vismazākajām pūlēm, National Instruments ir izstrādājusi daudzus programmatūras un aparatūras produktus speciāli datu iegūšanai. Jaunākās tehnoloģiskās inovācijas ir E sērijā, kura nosaka instrumentu klasi, standarta daudzkanālu arhitektūru datu iegūšanai. E sērija kombinē DAQ-PnP, National Instruments ASIC priekš Plug and Play konfigurācijas; DAQ-STC, cits National Instruments ASIC instrumentu klases skaitītājiem un taimeriem; RTSI kopne daudzplašu un daudzf-ju sinhronizācijai; un NI-PGIA augstas precizitātes analogo ieeju mērīšanai, neskatoties uz pastiprinājumu un diskretizācijas frekvenci. E sērija labi darbojas ar mūsu instrumentu klases signālu saskaņošanas produktiem, tā ka jūs vara būvēt izturīgas, elastīgas un precīzas uz PC bāzētas datu pielietošanas sistēmas.

### Plug and Play

Pirms E sērijas visas ISA DAQ plātes bija ar roku jākonfigurē caur divrindu korpusa (DIP) slēdžiem un/vai tiltslēgiem. Pat t. s. "hands-off" plātes prasa papildkomponentes, lai kalibrētu plati un prasa no jums iestādīt bāzes I/O adresi ar roku. Tagad, ar E sēriju, jūs varat izmantot Windows 95 – jaunākās Plug and Play operētājsistēmas – priekšrocības un izmantot programmatūru, slēdžu un tiltslēgu vietā, lai pilnībā konfigurētu DAQ plati, ieskaitot I/O adresi. Varat arī kalibrēt plātes izmantojot programmatūru, potenciometru regulēšanas vietā. Nometams drošinātājs ir izmantots visās E sērijas plātēs. Tādējādi, kad vienreiz E sērijas plāte tiek iesprausta PC, plātēm nekad nav vajadzīgs atkal pieskarties. Plug and Play datu iegūšana ir padarīta iespējama, pateicoties National Instruments DAQ-PnP ASIC.



1. solis – Nosakiet jūsu I/O signālu tipus

Kad izvēlaties komponentes jūsu DAQ sistēmai, jums vispirms jānosaka izmantoto sensoru un I/O signālu tipi. Tipiskie I/O tipi uz PC bāzētām sistēmām iekļauj:

- Analogā ieeja
  - temperatūra
  - spiediens
  - mehāniskais spriegums
  - elektriskais spriegums
  - strāva
- Akustiskie un vibrosignāli
- Analogā izeja
  - spriegums
  - strāva
  - viļņu formas ģenerators
- Ciparu ieeja/izeja
  - TTL-savietojama ieeja/izeja
  - augstsprieguma AC/DC ieeja/izeja
  - paralēlā sakaru iekārta
  - releju vadība
- Laika ieeja/izeja
  - frekvences ieeja/izeja
  - notikumu skaitīšana un taimeris
  - impulsa garuma mērīšana
  - impulsa astes ģenerēšana

2. solis – Izvēlaties signālu saskaņošanas ierīci

Daudzi signālu un sensoru tipi ir vispirms jāaskaņo pirms vienot tos pie datu iegūšanas iekārtas. Jūs varat izmantot signālu saskaņošanas aparāturu lai pastiprinātu, izolētu un filtrētu signālus, un lai nodrošinātu sensoru un signālu multiplexēšanas ierosmi pielietojumiem ar lielāku kanālu skaitu. Signālu saskaņošanas produkti ir sagrupēti 3 vispārējās klasēs.

**Instrumentu klase** — signālu saskaņošanai ar zemu trokšņa līmeni, SCXI nodrošina izturīgu, modulāru signālu saskaņošanas sistēmu ar ļoti plašu I/O iespēju diapazonu.

**Zema cena, mazs kanālu skaits** — pielietojumiem ar kanālu skaitu mazāk kā 16, kur elastība, paplašināmība un izturīgs iepakojums ir mazāk svarīgs, zemas cenas piederumi nodrošina ekonomisku risinājumu.

**Bez signālu saskaņošanas** — ja jūsu signāliem nav vajadzīga saskaņošana, izmantojat I/O savienotāju piederumus, tādus kā universālās I/O savienotāju plates ar skrūvju spailēm, DIN sliežu paliktņu skrūvju spailēm, BNC (*british naval connector*) adapterus un skrūvju spaiļu piederumus ar maketēšanas plates telpu patvaļīgām shēmām.

3. solis – izvēlaties atbilstošu datu iegūšanas I/O ierīci

Tādi kritēriji kā precizitāte, iegūšanas frekvence, kanālu skaits, elastība, uzticamība, paplašināmība, izturība un datora platforma tiek izmantoti lai noteiktu labāko DAQ iekārtu jūsu pielietojumam.

**Instrumentu klase** — visdaudzpusīgākajam aparātūras risinājumam un priekš *instrument-grade* veikspējas, piemēram īss notešanas laiks, garantēta precizitāte, liela diskretizācijas frekvence, daudzplašu sinhronizācija, uzlaboti skaitītāji/timeri vai ekranēti I/O savienotāji, izvēlaties instrumentu klases produktu. Šiem produktiem ir augstas precizitātes mērījumiem nepieciešamā kvalitāte, jauninājumi un veikspēja.

**Zema cena** — pielietojumiem ar mazāk kā 16 kanāliem, kur elastība, paplašināmība un uzlabotas iespējas ir mazāk svarīgas un kur cenai ir primāra nozīme, lēti DAQ produkti var sniegt ļoti ekonomisku risinājumu. Tie ir 2veidoti un testēti pēc tiem pašiem kvalitātes standartiem kā mūsu instrumentu klases DAQ produkti.

**Pārnēsājami** — tādiem pielietojumiem kā *auto-motive*, kosmoss un vides testēšana, ir derīgas pārvietojamas mērīšanas sistēmas. Mēs piedāvājam PC kartes (PCMCIA) un paralēlā porta (EPP un Centronics) produktus ar iespauža..

4. solis – izvēlaties atbilstošus kabeļus jūsu platei un signālu saskaņošanas piederumiem

Lai jūsu DAQ plati savienotu ar signālu saskaņošanas piederumu vai I/O savienotāju bloku. Lielākajai precizitātei, trokšņu izolācijai un izturīgākam savienojumam starp piederumu un instrumentu klases DAQ produktu, mēs rekomendējam ekranētus kabeļus.

Pielietojumiem, kur trokšnis un izturība nav strīda jautājums, bet kur kopējā sistēmas cena ir jāuztur zema, var izmantot lentkabeļus, lai pievienotu signālus DAQ produktam.

5. solis – izvēlaties jūsu programmatūras programmēšanas metodi

Labā aparatūra ir tikai tik laba, cik programmatūra. Lai veiksmīgi paveiktu savus pielietojumus, jums jāizvēlas ne tikai atbilstoša aparatūra bet arī atbilstoša programmatūra. Lai programmētu un vadītu DAQ aparāturu, National Instruments iekļauj NI-DAQ – daudzplatformu, no plates neatkarīgu programmatūras draiveru kopu – kopā ar visiem DAQ produktiem. NI-DAQ ir interfeiss, ko izmanto programmēšanai virknē izstrādes vižu un valodu, ieskaitot LabVIEW, LabWindows/CVI, LabWindows, Measure, ComponentWorks, C, un Visual Basic. NI-DAQ ir domāts Windows 95/NT/3.1, DOS, Mac OS un Sun Solaris operētājsistēmām.

National Instruments piedāvā arī sarežģītas datu analīzes un vizualizācijas pielikumprogrammu paketes, tādās kā LabVIEW, LabWindows/CVI, LabWindows, ComponentWorks, HiQ, Measure un VirtualBench.

## Īss MPEG-7 apskats.

Abstrakti- MPEG-7 ir jauns Starptautiskas standartizācijas organizācijas standarts, kurš atvieglo līdzekļu meklēšanu HTML saturā vairāk nekā šodien pastāvejošie teksta pamata meklētāji. Šis papīrs dod īsu MPEG-7 audio standarta aprakstu, to lietojumprogrammu terminos, kuros tas var atbalstīties, tā struktūra, process, ar kuru viņš tika izstrādāts, viņa speciālie aprakstītāji un aprakstu shēmas.

Indeksa rindas- Audio, meta dati, MPEG, meklēšana un meklēt.

### I. Ievads.

Internets, kā globālais tīkls, var būt izskafīts, kā datu milzīga glabātuve, kuri var būt meklēti, šķiroti un izsiāti lai dabūtu vajadzīgu informāciju. Hiperteksta iezīmēšanas valodas (HTML), Hiperteksta pārsūtīšanas valoda (HTTP), Web- brauzeri un, beidzot, meklēšanas serveru ierašanas deva jebkuram personālam datoram pieeju pie šīs glabātuves.

Tomēr, ne teksta bāzētie dokumenti, tādi kā MPEG-1, MPEG-2 vai MPEG-4 saspīesti audio media faili nav pietiekami atbalstīti šajās valodās, meklēšanas sistēmās, vai meklēšanas serveros. Pirmais, audio saturs nav izteikts teksta formā, darot tekstuālo meklēšanu neiespējamo. Otrais, paskatieties, ka katrs klausās audio saturu: caur mājas stereo sistēmu, bet biežāk caur portatīvu iekārtu, kā radio vai kompaktdisku atskaņotāju. Jebkurā gadījumā- pieeja audio saturam- ar dažu pogu palīdzību un iespējami mazu displeju, bet ne ar portatīva datora klaviatūru un displeju. Kas ir vajadzīgs, ir standartizētie datu līdzekļu uzskati, kuri arī atbalsta kompakta meklēšanas modalitātes rindu.

Sākumā, 1996. gadā, MPEG-7 bija aktīvs daļēji ar iepriekšējo MPEG panākumu multimedija satura efektīvas pārraides veidošanā palīdzību un ar lielu tīklu resursu pieejas auguma palīdzību. Ņemot vērā, ka teksta meklēšanas sistēmas indeksēja Web- lapas Internetā, izmantojot jau zināmas informācijas meklēšanas metodes, neviens no tam metodēm nebija pieejami audio failiem. Parādījās skaidrā vajadzība tīkla izvietotu media- failu indeksēšanā, neatkarība vai viņi ir saspīesti vai PCM. Tomēr, jaunām indeksēšanas metodēm vajag arī kalpot kā fiziskām media arhīvu glabātuvēm, pat ieskaitot mājas mūzikas kolekcijas.

MPEG-7 mēģina apgādāt šo vajadzību. Tas ir standarts līdzekļu aprakstīšanas satura pamatā. Tas ir neatkarīgs no media kodēšanas līdzekļiem (t.i. nav ierobežots ar MPRG- saspīestiem formātiem, pat ar ciparu formātiem) un ir neatkarīgs no fiziskas media atrašanas vietas. Tēlošanas šaurums MPEG-7 formātā ir ne tikai viegls, teksta pamata pieprasījumi (piemēram, atrast artista A visas dziesmas), bet arī sarežģītā satura pamata pieprasījumi (piemērām, atrast visas dziesmas, kuri ir līdzīgi melodijai, kuru es dziedu). Mēs arī pasvītrojām, ka kad šie meta- dati būs pieejami, daudz citu pielikumu nevis tikai meklēšana kļūs pieejami.

### II. Standarta struktūra.

MPEG-7 standartizē meta- datu, kuri satur media saturu, tēlošanu. Pretēji iepriekšējiem MPEG standartiem, MPEG-7 nestandardizē kodēšanas iekārtu vai dekodēšanas procesu, vai dekodētāju vai pieprasījumu vai procesa atbilstību. Kas ir kopējais visiem MPEG standartiem, ir apmaiņas vides standartizācija: normatīvie uzskati un meta datu semantika MPEG-7 gadījumā un šķiru straumes MPEG-1 un MPEG-2 gadījumā.

MPEG-7 audio standarts sastādīts no aprakstu shēmu un aprakstītāju. Aprakstītāji- meta datu ilustrācijas, kuri var būt saistīti ar kaut kādu laika intervālu (iespējami pilnais signāls) audio media signālā vai ar (iespējami periodisks) intervālu kopums media signālā. Cits viedoklis ir, ka deskriptors ir semantiskais atoms- atsevišķais aprakstīšanas modulis atkarīgs no satura īpatnībām. Spetrālā aploksne ir audio deskriptora piemērs.

Aprakstīšanas shēma- deskriptoru kopu un viņu attiecības, kuri adresē specifisku uzdevumu vai lietojumprogrammu, asociācija. Deskriptoru shēma ir Deskriptoru un iespējami citu deskriptoru shēmu koka struktūra.

MPEG-7 audio standarts paredz aprakstīšanas shēmas kā fundamentālas konstrukcijas un kā daļu programmatūras vadītiem instrumentāliem līdzekļiem. Instrumentālu līdzekļu standartizēta kopa nederēs katram pielikumam, bet tomēr, ir deskriptoru aprakstīšanas valoda (DDL) kura ļauj specifiskiem pielikumiem rakstīt jaunus deskriptoru shēmas. Tas ir standartu paplašināšanas mehānisms.

Katram deskriptoram un deskriptoru shēmai, kuri ir definēti audio standartā, ir normatīva definēšana, aprakstīta DDL, līdzīga definēšana kā C++ klasēm un ekstrakcijas aprakstošais un šī komponenta izmantošanas teksts. Ka arī ir informatīvais izejas teksts, ietilpstošais un parauga izvilksanas metodi un parauga pieprasījumu un atbilstību metodei, kuri var strādāt ar C++ asociātiem ar deskriptoriem vai deskriptoru tabulām.

### III. **Lietojumprogrammas.**

Viens no labākiem ceļiem standarta saprašanai ir piemēra lietojumprogrammas.

#### **A. Pieprasījums ar melodiju.**

Apskatīsim straumes audio servisu, kurā ir bāze ar saspiesto MPEG-4 audio vienā (vai vairākos) MPEG-4 media serverī un asociēta datu bāze ar MPEG-7 meta datiem MPEG-7 pieprasījuma serverī. Jo katrs MPEG-4 saspiesta dziesma ir indeksēta, MPEG-7 datubāze satur pilnu dziesmas melodijas priekšstatu un kādu mehānismu meta datu savienošanai, piemēram URL. Datubāze var arī saturēt dziesmas virsrakstu, artistu, dziesmas ilgumu, muzikālo žanru un stilu, un muzikālo "garastāvokli".

Lietotājs spiež pieprasījuma pogu viņa uz bezvadu kabata iekārtas un dzied melodiju. Šablona viļņa formas signāls tiek padots pieprasījuma serverī, kur darbojas process, kurš iegūst melodijas MPEG-7 metadatus. Šie meta dati ir pieprasījuma mērķis, un MPEG-7 serveris meklē viņa MPEG-7 melodiju datubāzē sakrišanas. Paris augšējās sakrītošas melodijas ir padoti atpakaļ pieejas iekārtā, kur displejā atspoguļojas dziesmas nosaukums un artists. Lietotājs var uzreiz nospiegt pogu lai sākt straumes pārraidi no MPEG-7 datubāzes uz viņa bezvadu pieejas iekārtu, kur media tiek dekodēta un atskaņota.

Lietotājs var turpināt sākt pieprasījumu ar vairāk jaudīgu pieprasījumu (piemērām ar pogām un izvēlnes izvēles) lai atskaņotu vairāk dziesmu "pēc artista", vai "ar tādu pašu stilu un žanru" ka šī dziesma, vai vienkārši "ar tādu noskaņojumu" kā šī dziesma.

#### **B. Pieprasījums pēc runāta satura.**

Skatīsim telefonu ziņojumu servisu ar asociētu MPEG-7 meta datiem. Uz katru ierakstītu personas ziņojumu rindā, runas pazīšanas serveris rada runas vēstuli un sūta atbilstošu teksta anotāciju MPEG-7 datubāzē. Datubāze arī glabā zvanīšanas numuru, zvana laiku, zvana garumu.

Lietotājs gaida noteikta zvana saņemšanu. Viņam vajag piezvanīt ziņojumu servisam, kur runas atpazīšanas serviss apstrādās viņa runas pieprasījumu MPEG-7 teksta pieprasījumā. Kur atpazīšanas serviss nevar atpazīt balsi un

pārveidot to tekstā var lietot fonētisku atpazīšanu. Rezidents MPEG-7 pieprasījumu serveris pēc tam meklē sakrītošus pieprasījuma atslēgvārdus ar vārdiem teksta anotācijā un arī MPEG-7 laukos, tēlojošos zvanītāja. Piemērām: “Dodiet man visus ziņojumus no mana priekšnieka par Tallahassae rēķinu.

### C. Atbalsīta lietojuma līmeņa audio rediģēšana.

Iedomājieties, ka audio signāls bija apstrādāts, lai iegūtu meta datus un atvieglotu dialogu lietojumprogrammas. Piemērām, daudz zema līmeņa audio īpatnības, tādi kā jauda, skaļums, spektrāla aploksne var hipotētiski izmantoties lietojuma līmeņa audio redaktorā. Var izmantot šos aprakstītājus, tādus kā skaļums, viļņa formas vietā. Ar kādam zināšanām par standartiem rediģēšanas noteikumiem žanra ietvaros, sarežģītai programmai jālieto spektra īpatnības lai pielīdzināt kanālus un tādā veidā automātiski paplašināt maisījuma skaidrību.

### D. Izvilšana un pieprasījuma paradigma.

*Pirmie divi piemēri izmanto kopējo arhitektūru. Pirmais solis lietojumprogrammas izstrādāšanas ir MPEG-7 meta datu izvilšana no media failu kopas, kā ir parādīts zīm.1. Tas var būt izdarīts vienreizēji media failu lielajai statistikai kopēj (kā belasa pieprasījumā), vai darīts daudzas reizes (kā telefona ziņojuma piemērā).*



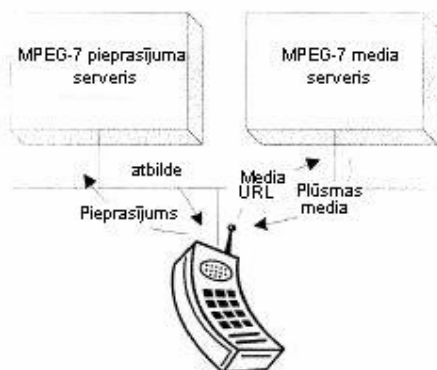
zīm.1. MPEG-7 izvilšanas process.

Nākošais solis- pieprasījuma process, ka ir parādīts zīm.2. Šajā procesā pieprasījums tiek konvertēts MPEG-7 pieprasījuma mērķī, parasti to dara ar to pašu izvilšanas mehānismu, tomēr tas nav parādīts. Pieprasījuma mehānisms pēc tam salīdzina pieprasījuma mērķi ar visu MPEG-7 ilustrācijām datu bāzē caur piemēroto attāluma funkciju un atgriež labāku sakrīšanu.



zīm.2. MPEG-7 pieprasījuma sakrītumu process.

Zīm.3. rada, ka tas var būt apvienots lai izveidotu klientu serveru MPEG-7 lietojumprogrammu, piemērām tādu, kura bija aprakstīta balss pieprasījumā lietojumprogrammā.



zīm.3. MPEG klients- serveris arhitektūra.

#### IV. Eksperimenta pamata process.

Lai garantētu mērenu un organizētu standarta attīstību, vajag izstrādāt diezgan sarežģītu procedūru, lai ievestu piedāvātas tehnoloģijas apzīmēšanu un šīs tehnoloģijas sekmēšanu standartā. Šis process "Pamata eksperiments", jo šis eksperimentālais eksperiments ir MPEG-7 darba attīstīšanas pamats.

Lai iekļūtu jauno aprakstītāju MPEG-7 audio standarta, jāizpilda sekojošus trīs soļus:

1. Piedāvājums;
2. Izsaukums;
3. Integrācija.

Apskatīsim katru atsevišķi:

##### A. Piedāvājums.

Piedāvājuma stadijā, iekšējai pārbaudei un iepriekšējai apskatīšanai jānotiek. Piedāvājumam jāsaturs:

1. Informāciju par to, vai jaunais deskriptors būs papildinājuma standartam vai viņš ir plānots standarta aprakstītāja aizvietošanai vai izmaiņšanai.
2. Tehniskais apraksts, kurš ir diezgan precīzs, lai izglītots šajā sfērā cilvēks varētu dublēt galvenās deskriptora funkcijas.
3. Eksperimentēšanas modelis (XM, vai attīstoša atbalsta programmatūra) struktūras izmaiņšana, kura ir nepieciešama lai īstenot piedāvājumu.
4. Tekstu normatīvajai (var būt informatīvajai) standartu dokumentu daļai.
5. Aprakstītāja kvalitātes pierādījumi (piemērām, potenciālais tirgus, lietotāju prasību pētījumi, piemēri, ka to var izmantot programmatūrā vai pakalpojumos) Tehnoloģijas iespējas ņemsies vērā.

##### B. Izsaukums.

Izsaukuma stadijā notiek metodikas testēšana un apskatīšana no citām pusēm. Piedāvājumu vajag neatkarīgi apskatīt vismaz viena trešās puses persona. Izsaukuma formāta izvēles mērķis ir katras piedāvātas tehnoloģijas trūkumus. Kaut gan ir pieņemts kā piedāvājumi strādās lielākā gadījumu skaitā, izsaukumam jāatrod šos nedarbojošos piedāvājumus, kuros piedāvāta tehnoloģija nedarbojās un izvēlēties piedāvājumus, kuriem ir vairāk izpildīta vai nepieestrādāta. Piedāvātais var būt kā daļa no testa un apskatīšanas.

Standarta kvalitātes skaitļi ir atļauti un atbalstīti. Kur standarta metodes nav pieejamas, "stimula trijnieka" metode ir piedāvāta, kā parasta MPEG-7 utilites atrašanās. Stimula trijnieks- triju satura daļu kopa, apzīmētam kā A, B

un C. Aprakstītāja izklāstītu iespēju ietvaros, A ir perceptuāli ir līdzīgs B vairāk nekā C. Tomēr ja A skaitļošanas ziņā ir vairāk līdzīgs C nekā B, tad šis trijnieks var efektīvi atrast vājumus aprakstītājā. Tāpēc secīgi atradot vājumus konkurējošos piedāvājumus, puses sadarbojas, lai radītu kvalitātes pieradījumu.

### **C. Integrācijas fāze.**

Pēc pamata eksperimenta rezultātu pieņemšanas izejas kodam, kurš dara izvilkšanu un sakrīt ar jauna aprakstītāja aspektiem, jābūt apstiprinātam. Izvilkšanas kods un izvilkšanas izpildītāji pēc tam ir integrēti uz XM. Sakrītošam kodam jābūt integrētam XM tādā veidā lai varētu izpildīt pilnu pieprasījumu.

### **V. Audio aprakstītāju īss apraksts.**

MPEG-7 audio standarts iekļauj sevī sešas galvenās tehnoloģijas, kuras var rupji sadalīt divās klasēs: zemas klases vai vispārēji instrumentālie līdzekļi un instrumentālie līdzekļi, vadāmie ar lietojumprogrammām. Audio aprakstītāja struktūra un klusuma vienlīdzīgs segments, var vērsties jebkuram audio signalam, tāpēc atradas vispārēju instrumentālo līdzekļu grupā. Skaņu efektu aprakstošie instrumenti, instrumentālo tembru aprakstošie instrumenti, runas satura aprakstītāji un melodiju aprakstītāji ierobežo viņu pielietojuma domenu kā līdzekļus, lai atļaut sev varāk aprakstošas jaudas un tāpēc atradas instrumentālo līdzekļu, vadāmo ar lietojumprogrammām grupā.

#### **A. MPEG-7 audio aprakstītāju struktūra.**

Dažas tehnoloģijas apvienojas lai formēt zema līmeņa audio aprakstītāju struktūru. Viens no šiem apvienojumiem- Skalu koks, kurš ļauj aprakstītāju rindu radīt maštabēšanas veidā. Galvenā ideja ir tāda, lai izplatīt tradicionāla laika rindas uzskatu ar spēju izņemt no tā nozīmīgus datus, kurus var pēc tam mainīt pēc parametriem, lai atbilstot jebkuram atmiņas daudzumam. Galvenais atmiņas tips, kurš formē rindu, var būt skalārs, vektors jeb matrica.

Cita datu apstrādes metode lai aprakstīšana augtu labi no vispārēju multimediju aprakstītāju shēmas saukta par "Segmentu". MPEG-7 audio attiecībā, audio segments ir laika intervāls, pie kura tiek pielietoti visi pievienotie aprakstītāji. Segments ir rekursīvā struktūra, ka audio straume var būt hierarhiski dekompreseti. Visi audio aprakstītāji un aprakstīšanas shēmas, ietilpstot visus lietojumprogrammu- pamatotus ierīces un skalu koku, var pievienot dotajam audio segmentam.

Zema līmeņa aprakstītāji kuri ietilps savienojumā, kuru veido audio segments un skalu koks ir ar laika apvalku, spektrālo apvalku, harmoniku, spektrālo centroidu un fundamentālo frekvenci.

#### **B. Klusuma segments.**

Ļoti vienkāršs, bet noderīgs instruments ir MPEG-7 klusuma aprakstītājs. Tas pievieno parastu "klusuma" semantiku audio segmentam. Tajā arī ietilps vienkāršs klusuma līmeņa indikators caur minimālo laika sākumu. To var izmantot audio straumes tālākai segmentācija vai kā mēģienu neapstādot segmentu.

#### **C. Skaņu efektu aprakstītāju instrumenti.**

Skaņu efektu aprakstītāji un aprakstu shēmas ir instrumentu kopa pamata skaņu efektu indeksēšanai un klasifikācijai. Atbalsts automatiskajai skaņu efektu identifikācijai un indeksācijai ir ieslēgta arī kā instrumentālie līdzekļi skaņu klašu taksonomiju noteikšanai un instrumentāliem līdzekļiem skaņas atpazīšanas līdzekļu ontoloģiju noteikšanai. Tādi atpazīšanas līdzekļi var izmantoties lai automatiski indeksēt un segmentēt skaņu celiņus.

Vairāk skaitļošanas pieeja meklēšanai ir pieejama skaņu efektu instrumentu ietvaros. Kaut gan tiem, kuri tikai grib identificēt skaņu efektus, pamata spetrāla dekompozīcija būs kā nevajadzīgs rezultāts. Tomēr, dodot pareizi izgatavotus skaņu efektu parametru kopu, instrumentālie līdzekļi vērā skaņu efekta atpazīšanu skaņu maisījumā. Spetrālais pamats- pēc būtības laika apvalks sakrustots ar spektrālo apvalku, ar dažām no tām tipiski apvienotam bāzēm lai formēt skaņas modeli.

#### **D. Tembura muzikālas iekārtas aprakstīšanas līdzekļi.**

Tembra aprakstītāji tiecas pie instrumentālo skaņu perceptuālu atšķirību aprakstīšanas. Tembrs dotajā brīdī literatūrā noteikts kā perceptuālas atšķirības, kuri veido divas skaņas, kuriem ir tāds pats solis un skaļums skan dažādi. Tembura aprakstīšanas shēma apraksta šīs perceptuālas atšķirības ar samazināto aprakstītāju skaitli. Aprakstītāji skar tādus saprašanas kā skaņas "uzbrukums", "spilgtums" un "bagātība".

Ir četras muzikālo instrumentu skaņu grupas: 1. harmoniskas, atbalstītas, sakarīgas skaņas; 2. neharmoniskas, atbalstītas, sakarīgas skaņas; 3. sišanas, neatbalstītas skaņas; un 4. nesakarīgas, atbalstītas skaņas. Šo četru grupu ietvaros, divas klases labi savienotas ar MPEG-7 standartu un bija galvenā pamata eksperimenta attīstības tēma. Viņi ir harmoniskas, koģerentas, atbalstītas skaņas un neatbalstītas, sišanas skaņas. Citas divas grupas bija uzskatītas ar zemo prioritāti samēra retuma dēļ, bet tomēr var parādīties standartā.

#### **E. Runāta satura aprakstīšanas instrumenti.**

Runāta satura aprakstīšanas shēma pamatota uz idejas, ka runas atpazīšanas sistēmas dotajā laikā nav perfektas. Vairāk nekā būt parastajai runāšanas teksta dekodēšanai, aprakstīšanas shēma sastāv no kombinētiem vārdu un tālruņu krustojuma katram runātājam. Kombinējot režģu, trūkstošo vārdnīcā vārdu problēma ir ļoti atvieglota un meklēšana vēl var turpināties, kad jau ortogonāla izvilksana ir ar kļūdu. Aprakstīšanas shēma var izmantoties divām šaurajām meklēšanas scenāriju klasēm: indeksēšana un audio straumes iegūšana, un pievienotu runai multimediju objektu indeksācija.

#### **F. Melodijas kontūra aprakstīšanas shēma.**

Melodijas kontūra aprakstīšanas shēma- kompaktais melodijas informācijas priekšstats, kurš ļauj efektīvu un spēcīgu melodiju vienlīdzīgumu sakrišanu, piemēram, balss pieprasījumā. Melodijas kontūra aprakstīšanas shēma izmanto piecu soļu kontūru, kurā intervāli ir kvantēti. Melodijas kontūra aprakstīšanas shēma arī rada pamata ritmisku informāciju, saglabājot katra tuvāka pilna bita notēs numurus, informāciju, kura var redzami palielināt pieprasījumu precizitāti.

Ir arī vairāk paplašināta melodijas kontūra aprakstīšanas shēmas versija, kuru izstrādāja līdzstrādnieki, kurš paplašina dažas šauras vietas precizitātē un atteikuma spējas melodijas kontūra aprakstīšanas shēmā.

#### **G. Standarta pārējas daļas.**

Standarta audio daļa bija izstrādāta lai labi strādātu ar MPEG-7 citām daļām, it īpaši ar 5-MDS daļu. Lai izvairīties no citām liekam iespējam standartā ietvaros, audio apakšgrupa uzticas citu grupu darbam lai pabeigtu instrumentālo līdzekļu kopu, kurā ir ieinteresēts audio arhivators. Mēs jau apspriežam DDL izmantošanu kā standartu ceļu standarta paplašināšanā. Mēs arī redzējām, kā audio struktūra izmanto kopēju segmentu no MDS.

Jebkurš aprakstošais saturs var vēlēt pievienot tādas tipiskus aprakstītājus, kā virsraksts, dziedātājs un ierakstīšanas gadu mūzikas fragmentam. Šie aprakstītāji, kuri tagad kalpo bibliotēkās un arhīvos, plašāk atklāti MDS. Tā arī var pamēģināt aprakstīt muzikālo žanru kā hierarhisku ontoloģiju, vai aprakstīt muzikālo instrumentu no vadīto



terminu saraksta. Mehānismi, ar kuriem var veidot tos ontoloģijas un vārdnīcas, atradās MDS kā vadāmais datu tipa termins un klasifikācijas shēmas aprakstīšanas shēma. Izvietojot visas daļas kopā, var pārsūtīt pastāvējušo arhīvu MPEG-7 standarta apvienota arhīvā, un pēc tam paplašināt ar speciāliem saņemtajiem signālu audio aprakstošām shēmām.

## **VI. Nobeigums.**

*MPEG-7 adresē kritisku vajadzību līdzekļu indeksācijā un meklēšanas līdzekļos un dažas piemēra programmas, kurus var izveidot MPEG-7 līdzekļu meta datu pamatā. Mēs devām īsu MPEG-7 standarta veidošanas procesa aprakstu, ka arī aprakstītāju, aprakstu shēmu lomas MPEG-7 audio standartā. Viņi jau atbalsta lielu pielietojuma spektru, tomēr DDL atļauj paplašināt aprakstu shēmu kopu, tādā veidā nodrošinot noderīgumu lietojumprogrammās.*

MPEG-7 standarts nodrošina struktūru audio signālu un audio arhīva aprakstīšanai, izmantojošus ne tikai bibliotēkās, bet arī izmantojot jaunu, signālos bāzētos kritērijus, kuri var būt automātiski izņemti jebkurā laikā. MPEG-7 nodrošina jaunu 21. gadsimta aprakstīšanas tehnoloģiju.

## **Runāta satura reprezentācija MPEG-7.**

*Abstrakti-* Vārdi, runāti audio- vizuālajā dokumenta formā ir acīmredzams un intuitīvs meta datu komponents. Šis komponents ir obligāts, lai garantētu vispusīgo audiovizuāla satura apskatu MPEG-7 standartā. Ar manuālo transkripciju, ļoti dārgu, šie meta dati tipiski būs padoti no automātiskiem runas atpazīšanas sistēmām. Šo atpazīšanas

instrumentu produkcijai raksturīgas kļūdas rada specifiskas grūtības normālai darbībai, ka arī iespējai nevienmērīgu datu bāzu sadarbībai. Mēs aprakstam struktūru, kura ietilps sevī varbūtības apvienoto vardu un tālruņa režģa kopā ar paskaidrojošu meta datu virsrakstu un detaļas kā šī struktūra izvairās vai paaugstina problēmu kvalitāti.

*Indekss-* Meta dati, MPEG-7, runas atpazīšana, runātais saturs, runāta dokumenta mainīšana vai balss anotācija.

## **I. Ievads.**

MPEG-7 ir meta datu standarts multimediju dokumentu satura aprakstīšanai, precīzāk teikt, bet nav ierobežots ar dotā brīdī pastāvējušo MPEG-1,-2 un -4 standartiem. MPEG-7 meta datu standarts apņem plašu aprakstītāju diapazonu. Viena ekstrēmumā ir zema līmeņa aprakstītāji, piemērām, krāsaina histogramma, kurā matemātiskie pārveidojumi veidoti audiovizuālajā dokumentā. Tādi meta dati ar gatavību izņemti programmas nodrošinājumā vai personāla datora aparatūras līdzekļos, bet reti viegli interpretējams cilvēku terminos. Dažos gadījumos, šie meta dati var būt utilizēti tikai kopā ar citiem zema līmeņa meta datiem. Citā ekstrēmumā, MPEG-7 satur augsta līmeņa meta datus, tādus kā personas vārds vai virsraksts. Izņemot retus gadījumus, kad šie var būt izņemti ar palīdzības meta datiem, šo aprakstītāju saturam jābūt izveidotam manuāli. Dabiski augsta līmeņa meta datiem ir priekšrocība cilvēku vieglākā saprašanā. Pozīcija šajā spektrā, okupēta ar aprakstītāju, lielā mērā noteikta ar izvilksanas veidu un cilvēku saprašanu. Meta datu komponenti, kuri atradās spektra vidū rada īpašas problēmas, jo tie nav spējīgi vērsties gan pie matemātiskas precizitātes, gan pie cilvēku saprašanas, lai garantētu viņu pielietojumu

Viens no vidēja līmeņa meta datu piemērs ir runāšanas saturs. Var būt viena no ticamākas meta datu formām, izņemamam no audio vizuālie dokumentiem, runas saturs rada faktiskus pateiktus vārdus. Esot galvenais līdzeklis cilvēku savienošānā, būtisks derīgas izmantošanai informācijas procents var būt glabāts Runas saturā. Audio vizuālajā materiālā, tajā laikā, kad aprakstītāji tādi kā atslēgu struktūras rada vizuālas darbības prezentāciju, satura saprašanu gavēni iet no pateiktiem vārdiem. Ārkārtējos gadījumos, tādus kā balss pats, ir mazs noderīgs saturs, nekā tikai runa.

Esošo audio vizuālo dokumentu anotēšanas malā, runāta satura aprakstītāja pastāvēšana dod iespēju izmantot runāšanas anotāciju gadījumos ja citi interfeisi nav praktiski. Piemērām, daudziem lietotajiem nepatīk anotēt fotogrāfijas vēlāka stadijā interfeisa ierobežošanas dēļ nevar to darīt fotografēšanas laikā. Fotografiju balss anotāciju to darīšanas laikā var izdarīt tikai liela brīvu fotogrāfiju organizācijas ceļā ciparu kamerās.

Runas satura svarīgums, tomēr, atrodas ne tikai uz viņa izplatībā, bet arī uz tā taisnas un dabiskas pielietojšanas viegluma. kaut gan pateiktie vārdi un rakstīti vārdi ir dažādi no algoritmiskas puses, cilvēki parasti pārslēdzas starp diviem. Tādā veidā pārbaudot pateiktu dokumentu ar teksta dokumenta ievadīšanu ir ideāls naturāls audio segmenta veidošanas ceļš un tas sakrīt ar lietotāja prata modeli.

Kaut gan, principā, pateiktu vārdu izmainīšana ir iespējama izmantojot jautājumu- pēc piemēra (QBE) ņemot darbā zema līmeņa audio deskriptorus, bet praktiskā tādās metodes nav spējīgas liela apjoma datubāzu izskaitļošanai. Praktiski runāto meta datu saturu izņems automātisko runas atpazīšanas sistēmas (ASR).

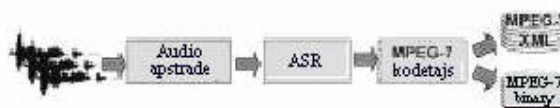
Slikti, ka neskatoties uz virsraksta precizitāti, pieejamu runātam tekstam, reālas ASR ir ļoti tālu no perfektam un viņu lietošana MPEG-7 meta datu veidošanā sastapa daudz problēmu. MPEG-7 ir atšķirība starp normatīvajiem un nenormatīvajiem instrumentiem; normatīvie instrumenti ir uzskatīti par standarta daļu. Izvilksanas instrumenti tiek uzskatīti par nenormatīviem. Piemērām, redzamas finansiālas, kā arī skaitļošanas domas par glabāšanu, diktē kā tas nav praktiski ievietot ASR sistēmas standartā. Tipiska ASR sistēma var saturēt vairākus miljonus parametru kuri adaptēti lietotājam un apkārtējai videi. Tāpēc izvads no diviem ASR sistēmām var atšķirties darot problemātisku nevienmērīgu datu bāzu mainīšanu. Ka arī meta datiem jābūt spēcīgiem pret dekodēšanas kļūdām, garantējot kā anotētie dati palika labojami; lietotājs uzskatīs par nepieļaujamu, ka skaidri pateikta runas anotācija, kuru viņi lemj glabāt datu bāzē, nevar

būt izlabota. Tāpēc ASR pielietošana runas satura meta datu iegūšanā veido problēmas gan sadarbības iespējā, gan kļūdu labošanā.

Tādā veidā mēs varam redzēt kā parasta teksta rinda no ASR, kura attēlo runas saturu būs nepieļaujama un ka runas satura aprakstīšanas shēmai jāapraksta ne tikai labāku ASR izvadu, bet arī apsvērtu hipotēžu kopu un noteiktu kļūdu daudzumu.

## II. Meta datu iegūšanas kļūdas.

Lai saprastu runas satura aprakstītāja projekta ierobežojumus, vispirms noteikti jāsaprot iegūšanas instrumentu dabu un viņu atteices režīmus. Tipisks iegūšanas ceļš ir parādīts zīm.1. ASR sistēmas izvadu var uzskatīt par idealizētas anotācijas varbūtisku korupciju (rokas transkripciju). Precizitāte var variēt no 90% lasīšanas sistēmās izmantojot runātāju adaptētas sistēmas, caur 70% reālas pasaules runai un uz leju līdz 30% sarunvalodas runai. Šajā daļā mēs pasvītrosim pamata ARS sistēmu pazīmes un viņu lomu iegūšanas precizitātē.



zīm1.Kodēšanas ceļš pārveidojot audio datus RunasSatura meta datus.

Lielākā daļa no ASR sistēmām ir pamatoti uz efektīva nepazīstamu runas datu atbilstības statistiskam vārdu attēlojumam algoritma. Unikālas transkripcijas un izrunas savienošana grūtības dēļ, ASR sistēmas tipiski iegūt izmantojot režģi, kurā ietilps vairākas hipotēzes dodot varbūtību tikai pāris frāzēm vai tālrunim katrā reizē. Visticamākais ceļš caur režģiem- parasti ASR sistēmas izvads, dodot ražoto anotāciju. Postprocessori bieži utilizēti N-Best analīzes ražošanai vai vārdu alternatīvai, atšķirībā no lietojumprogrammas.

Kā ar jebkuru statisko paraugu, kurš darbojās pēc metodes, kļūdas un kategorijas nepazīšana notiek. Kļūdu iemesli un daba var būt sadalīta trijās klasēs: audio kvalitāte, runas kvalitāte un trenēšanas datu kvalitāte. Kaut arī katra dažāda kļūdu tipa parādīšana izmainīsies atkarībā no runas satura aprakstītāja izmantošanas, MPEG-7 ir izstrādāts lai izņemt meta datus no audio- vizuālo lietojumprogrammu pilna diapazona, un tāpēc vajag lai tas būtu izturīgs pret visām šiem kļūdām.

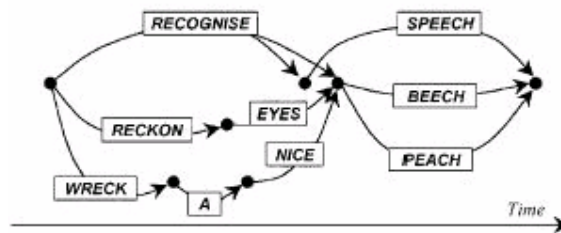
Pirmās kategorijas kļūda, iespējami, ir vieglāka saprašanai. Kad skaņu signālu kropļojumi paaugstinās, piemēram, nenozīmīga trokšņa dēļ, ASR sistēmas iespēja identificēt atšķirīgus vārdu komponentus samazinās. Vārdi, kuri atšķiras fonētiski tikai mazliet, piemēram, vārdi "pele" un "pīle" var būt viegli sajaukti. Kad trokšņu vai korupcijas līmenis paaugstinās, lielākais kļūdu diapazons kļūst pieejams. Dažos apstākļos skaņas sākotnēja apstrāde var samazināt šos efektus. Runas ilgumam var būt papildus kaitīgie efekti precizitātei satura problēmu dēļ. Iepriekšējie eksperimenti lielā mērā saspiežas uz TREC atzīmju rindu, izmantojošus garas anotācijas un lielus jautājuma vārdu skaitļus. Anotācijas garumam samazinoties, dekodēšanas precizitāte arī samazināsies. Tas būs nozīmīgs fotogrāfiju anotācijas gadījumā, kur īsas anotācijas ir iespējamas. Šī nesaprašanas efektu var redzami samazināt ar vairāku hipotēžu apstrādāšanu izņemto meta datu ietvaros, pat ar zudumiem precizitātē.

Otra kļūdas kategorija var ražot ļoti nozīmīga spīkera un satura atkarības variāciju precizitātē. ASR sistēmas parasti izmanto tālrunu vārdnīcas, pārstāvējušos kanonisku vārdu izrunu. Akcenti un dialekti, kuri atšķiras no tā ar kuru trenējās sistēma var veicināt suboptimālo vārdu sakrišanu. Līdzīgi, negramatiska runa, izlaidumus, un svārstīšanas var samazināt atpazīšanas kvalitāti.

Beigu beigās, labošanas liela svarīguma dēļ, mums jāapskata kļūdas, nākošas ārpus ASR trenēšanas laukuma--ne -no -vārdnīcas kļūdas. Tie vārdi, piemēram, cilvēka vārds vai vietas nosaukums, nepazīstamie ASR sistēmai; viens vai vairāki vārdi ir ievietoti viņu vietā veicinot potenciālas kļūdas blakus vārdos. Jo lietvārdi formē lielāku atpazītu vārdu daļu, tādu vārdu neatpazīšana var sekmēt neizlabojamas anotācijas.

Šo kļūdu efekts, ka arī tādu no otras kategorijas, var būt samazināts fonētisku dekodēšanu apvienošanas ceļā. Kur vesels vārds var būt dekodēts nepareizi, tikai daži no pārsūtītajiem dekodētiem tālruniem var būt ar kļūdu. Kaut gan semantiska meklēšana nav pieejama taisni, izteiksmju bāzētie audio labojumi būs pieejami caur anotāciju "līdzīgi skaņo" pieprasījumam sekciju meklēšanu. Vajag atzīmēt kā ASR sistēma lemj, kur vajag nodrošināt vārdus, tālrunus, vai pat abus.

Nobeigumā mēs varam pieņemt, ka atpazīšanas ierīce veidos neideālus meta datus. Lai garantētu labošanu kur labāka ceļa dekodēšana bija ar kļūdu, mēs varam pievienot turpmākas hipotēzes režģu formā. Ka arī mēs varam ietilps fonētisku dekodēšanu lai atbalstīt frāžu- bāzētu labošanu lai garantētu ne- no- vārdnīcas terminu atjaunošanu. Režģu platums un satura vārds- tālruna proporcijas var būt variēts attiecīgi ASR dekodēšanas uzticībai. Daļa no tipiski savienota vārda un režģis ir parādīts zīm.2.



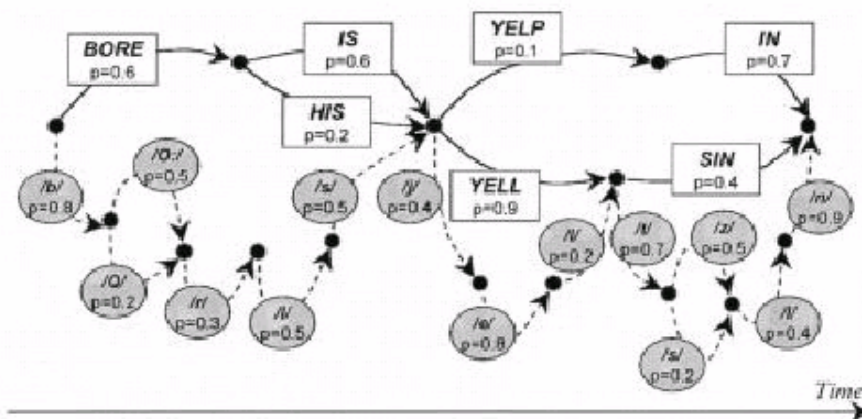
zīm.2. Frāzes "recognize speech" hipotētiska pārveidošana.

### III. Sadarbības spēja.

Mēs demonstrējam, ka fundamentāla saprašana RunasSatura aprakstītāja dizainā ir iespēja atbalstīt (potenciāli) kļūdainus izdabūtus aprakstītājus izmantojot ARS sistēmu uzvedībā lai būtu iespējama normāla izmaiņšana. Mums tagad vajag izskatīt kā mēs varam izmantot ģeteroģēnu RunasSatura datu bāzi, tas ir, datu bāze kura sastāv no runas satura meta datiem izdabūtiem izmantojot vairākas ASR sistēmas. Kā aprakstīts augstāk, ASR sistēmu sarežģītība, izmērs, un maksa, kā arī specializācija caur spīkers/ dialekts adaptāciju, un viņu turpinājuša attīstīšana, nozīmē ka ASR nevar būt normatīva. Diemžēl, dažādu ASR sistēmu izvadī var atšķirties.

Sastopama problēma ir tāda, kā garantēt labošanas kārtību divām vai vairāk anotācijām kad viņu kļūdu statistika atšķiras. Kad viena dekodēšana neder, var kad tas ir nepareizais ceļš caur hipotēžu režģim, vai dekodētājam ir liela kļūdu iespējamība, mums vajag dod mazāk uzticības šiem meta datiem nekā dekodēšanai ar augstāku prioritāti. Tas nav praktiski glabāt informāciju par mutiski līdzīgiem vārdiem. To vietā, trīs informācijas līmeņi ir ierīkoti lai atvieglotu labošanu.

Vairāk svarīga no tiem ir vārdu saistu varbūtības lielumu pastāvēšana (un, to vietā, tālrunis) (sk. zīm.3.). Tie dod dažādu vārdu slepenības pazīmi, sākušo vienā laikā. Šīs varbūtības var būt pietuvināti ar slepenības informāciju, padotu no ASR sistēmas. Nākošais līmenis ir leksikona izmērs, lietota dekodēšanā, kur vārds ir dekodēts no mazā varbūtību loka, vajag dod mazāk uzticības šiem nekā tiem, kuri bija izvēlēti no plaša loka. Beigas palīdzība ir stingri pārskaitīts tips lai inducētu audio kvalitātei, vai tas ir trokšņi, trokšņu runa, vai tīra runa. Kaut gan rupja, šī diskreta vienība ir tāla no ideāla, trokšņa daudzuma noteikšanas un iedarbības uz dekodēšanu problēma nav tīra.

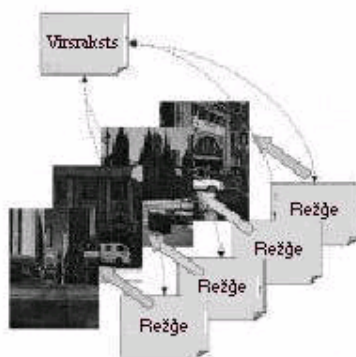


zīm.3. Varbūtiski kombinēta vārdu un tālruņu režģi reprezentējot ne no vārdnīcas vārdus "Boris Yeltsin".

#### IV. Struktūras Īss Apskats.

Tomēr komponenti pieprasīti no kļūdu attieksmes spējas un spējas sadarbībai nodrošina minimālo derīgu izmantošanai struktūru, praktiskām domām jābūt pievienotām.

*Apskatiet, piemēram, cilvēku, kurš fotografē, un, paralēli, anotējot tos fotogrāfijas ar runātiem virsrakstiem. Tas ir prātīgi meta datu saturam, t.i., savienota vārda un tālruņa režģim, lai būti savienoti ar fotogrāfiju. Paskaidrojuma meta datu izvietojums, t.i. leksikonu un tālruņa statistikas, tomēr, nav skaidrs. Gadījumā, kad viens špikeris izmantoja vienu ASR un vienu valodu visām anotācijām, glabāšana virsū ietilpstošās paskaidrošanas meta datus katrā fotogrāfijā nav pareiza. Drīzāk, mums vajag izmantot vienu krātuvī paskaidrošanas meta datiem un visas režģis, kuras to izmanto, jābūt uz to uzradītiem. Fotogrāfijas apraksta scenārijam, mums tādā veidā ir struktūra parādīta zīm.4. Runas saturs tādā veidā sastāv no divām dažādam bet savienotiem aprakstīšanas shēmām: RunasSatura virsraksts un RunasSatura režģis ar savienojamiem starp viņiem.*

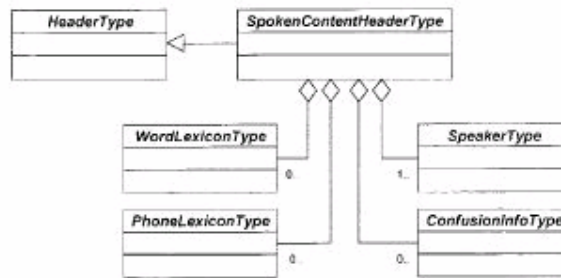


zīm.4. Režģis var izmantoties citu meta datu anotēšanai. Režģis savienojās ar virsrakstu, saturošo paskaidrošanas meta datus.

#### V. RunasSatura Virsraksts.

Kā detalizēt augstāk, kļūdu atteikšanās spēja un sadarbības rezultāts struktūrā divu atsevišķu, bet savienoto komponentu: virsraksts, saturošais paskaidrošanas meta datus, un režģos, saturošās varbūtības režģi, pārstāvējušo vārdus un tālruņus. Šajā iedaļā, mēs paskaidrosim virsraksta elementu detalizētu struktūru un lietderīgumu.

Virsraksta struktūra ir ilustrēta zīm.5.

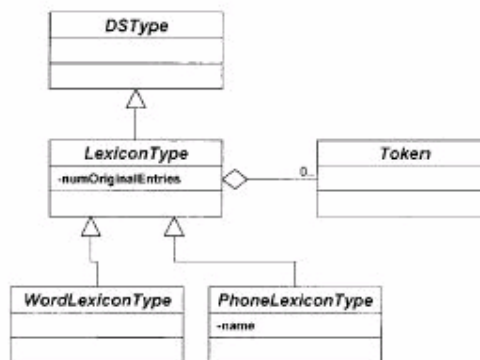


zīm.5. Virsraksts satur statisko informāciju, kuru var pielietot vairāk nekā vienai RunasSatura anotācijai.

Virsraksts satur četras paskaidrošanas meta datu grupas aprakstot kas runā, kurus vārdus viņš runā, kura tālruņa kopa iedarbināta, un fonētiskās runas atpazīšanas sistēmu dekodēšanas kļūdu statistiskas īpašības. Kombinējot šos elementus, mēs varam aprakstīt vairāku runātāju runu kompaktā formā atkārtoti izmantojot veidotus līdzīgus elementus, Piemērām, izmantojot vienu ASR sistēmu vairāku runātāju anotēšanai.

#### A. Vārdu Leksikons.

Vārdu leksikona mērķis ir vienas utilizētas vārdnīcas glabāšana vienīgā vietā no vienas vai vairākām ASR sistēmām. Praktiskā, tas var būt lietderīgi labošanas laikā, jo tas satur informāciju par vārda hipotēzes izmēra vietas, izmantotas dekodēšanas laikā. Lielas vārdnīcās runas atpazīšanas sistēmas ASR leksikons var sastāvēt no vairāk nekā 100 000 vārdiem, tajā laikā normālas runas vienas stundas vārdnīcas izmērs var būt tikai daži tūkstoši vārdi. Atmiņas efektivitātei, leksikons bieži vienkārši pārskaitīs vārdus kuri parādās režģos un inducēs oriģināla leksikona izmēru caur atribūta vērtību. tas ir ilustrēts zīm.6.



zīm.6. Vārdu un tālruņu leksikoni dabūti no kopējā LeksikonaTips.

Problēma no kā sastāv vārds leksikonā ir sarežģīta arī ar to, kā vajag atbalstīt valodu lielu diapazonu. Lielākā daļā no ritumu valodām izmanto romaniešu zīmju kopu, bet bieži paplašināts ar akcentētām zīmēm, kuri izkrīt no normālo ASCII zīmju diapazonu. Austrumu valodas var sastāvēt no pilnīgi atšķirošiem no romaniešu zīmēm. XML ir spējīgs reprezentēt vārdus, izmantojošus unikodu, un mēs neizvietojām citu aizliegumu vārda formātam, kā tas nevar saturēt normālus baltas vietas zīmes.

#### B. Tālruņu Leksikons.

Tālruņa leksikona mērķis ir tālruņa kopas reprezentācija, izmantota ar vienu vai vairākiem runātājiem. Jebkurai specifiskai valodai, ir iespējami daudz tālruņu kopu, un tiešām, ar ASR sistēmu parādīšanu, daudz dažādu tālruņu kopu

izveidoja caur modifikāciju, lai paaugstināt ASR sistēmu precizitāti. Lai garantētu sadarbības spēju, mums vajag pamēģināt aizliegt tos tālrūna kopas, kuras ir gan internacionālie, gan daļas no eksistējošiem standartiem, bez pamata ASR funkcionalitātes aizliegšanas.

Tālrūna leksikona aprakstītāja ietvaros, divas pamata tālrūnu kopas ir atbalstītas: Internacionālais Fonētikas Asociācijas Alfabēts (IPA) un Runas Assesment Metodes Fonētiskais Alfabēts (SAMPA). IPA parstāv pamata skaņas no kuriem jebkura no pasaules valodām var būt izveidota. Tomēr, IPA alfabēts, plaši izmantots ar fonētiķiem, var būt grūts individuālajām valodām. Tādos gadījumos, tālrūnu kopa SAMPA var izmantoties. SAMPA ir ASCII IPA simbolu reprezentācija, un bija sākuma izstrādāta dažām Eiropas valodām. Pēc tam tā bija paplašināta ar dažām papildus grupām, lai apkalpotu vairāk noteiktas prasības. MPEG-7 kontekstā mēs izmantojam "SAMPA" lai uzzinātu pie kuras SAMPA paplašināšanas pieder; tas beigu beigās noteikts ar vārdnīcu aprakstīšanā. Diemžēl, klusums neizskatās tālrūnu un ASR robežās, un neviena standarta nepastāv to reprezentācijai. Lai paskaidrotu klusumu tālrūna dekodēšanā ietvaros, mēs izvēlējamies SAMPROSA piedāvājumu "... " lai parādītu klusumu. Gadījumos, kad SAMPA un IPA tomēr nav derēti, papildus pārskatīšana "unknown" ir izvēlēta; tas ir nestandarta tālrūnu kopu indikācijai un tāpēc ierobežo sadarbības iespējas. Tas, tomēr, ļauj pielietot programmas, kur izmantotie tālrūni nav derīgi, vai zināmas lai būtu tādas kā pieprasījuma sistēmā.

### C. Tālrūna Neskaidrību Statistika.

Tālrūnu limitēts skaitlis ļauj tālrūnu neskaidrību statistikas iekļaušanu. tas dod iespējamās dekodēšanas varbūtības, t.i.,  $P(d|p)$ , kur p ir tālrūnis, kuru izmanto dotajā brīdī un d ir tālrūnis kurs bija dekodēts. Papildus dekodētāja varbūtības, ieslēdztoša vai izslēdztoša tālrūnus, ir arī pievienota kā tālrūna parādīšanas iespēja. Šīs statistikas ļauj pieprasījuma satura neatkarīgu statistisku sakrišanu ar anotācijām.

### D. Spīkers.

Galvenais virsraksta elements ir spīkers kurs savieno kopā leksiku un statistiku. Spīkera elements ir izstrādāts lai reprezentēt vienu personu kura runā, tomēr faktiskā persona var piedomāties, datu struktūra vienkārši kalpo kā sakni asociētiem saitēm. RunasSatura ir izstrādāts lai atbalstītu daudzus spīkerus ar, potenciāli, dažādām vārdnīcām, statistiskām vai pat valodām. Sekojošie atribūti un elementi raksturo spīkerus.

1) *Individuāla identifikācija*: Tas ņem personas vārda un virsraksta formu. Ideāli, tam jābūt unikāliem visiem spīkeriem Virsrakstā.

2) *Saites vārda un/ vai tālrūna leksikai*: Anotācija pārstāvējušā vienu spīkeru var būt reprezentēta vārdos, kā arī fonētiski. Spīkers var lietot maksimāli vienu vārdu leksikonu un vienu tālrūna leksikonu.

3) Norāde neskaidrību statistikai reprezentējot tālrūna neskaidrību iespēju spīkera tālrūna dekodēšanas: Tas nekas nenozīmē kaut gan divi spīkeri var lietot vienādu tālrūna kopu, lietotāju adoptācija nozīmē ka viņu neskaidrību statistika var atšķirties.

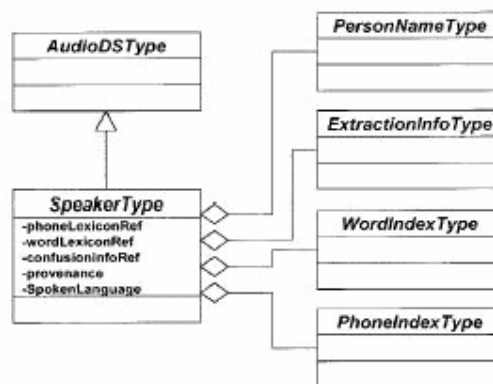
4) *Avots*: Tas ir uzskaitīts tips radot spīkera anotāciju avotu. Atļautas vērtības ir "unknown", "ASR", "manual", "keyword" un "parsing".

5) *Runas valoda*: Tas parstāv valodu un dialektu uz kura runā izmantojot ISO- 3166 un ISO- 639 standartus.

6) *Izdabūšanas informācija detalizējot izmantojamus izdabūšanas instrumentus un parametrus*: Tas vispirms paredzēts izmantošanai ārpus korpusa vai datu bāzes, kā arī šī informācija nevar būt izmantota sadarbības spējas palīdzībai.

7) *Papildus vārda un tālrūna indeksi izmantojot vārds/ tālrūnis n- gram atslēgas kuri ir ietilpsti*: Tomēr principiālā, pietiekošā informācija, to iekļaušana ļauj hierarhiski labošanu tīklos.

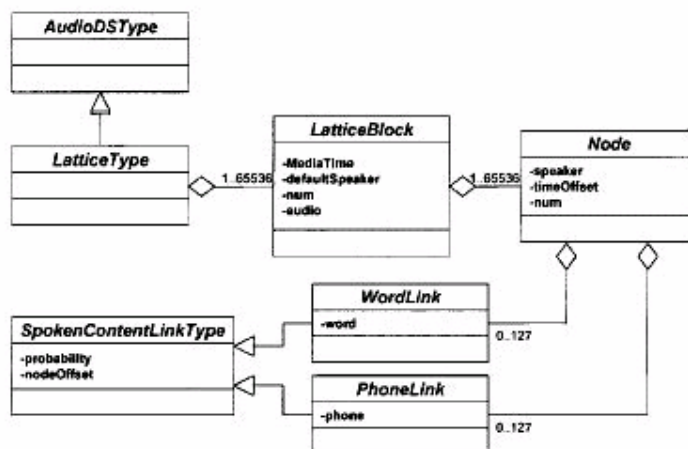
RunasSatura struktūra ir ilustrēta shematiski zīm.7.



zīm.7. Spīkers veido atslēgas elementu savienojot vārdus, tālruņus un indeksus kopā.

## VI. RunasSatura Režģis.

Tajā laikā kad RunasSatura Virsraksts satur paskaidrošanas meta datus, anotācijas saturs izvietojās RunasSatura Režģos, kuru struktūra ir parādīta zīm.8.



zīm.8. Režģis pārstāv vienu anotāciju vai anotācijas segmentu. Tās pārstāv kombinētu vārda un tālruņa statistisku režģi atbalstot viena vai vairāku spīkeru runu.

Režģu mērķis ir RunasSatura reprezentācija savienojamā veida ar lielāko daļu ASR sistēmu izejām, kā arī saturot pietiekošo informāciju kā normālā labošana ir iespējama. Kā aprakstīts augstāk, ideāla pieeja ir izdabūšanas instrumenta rezultāta reprezentācija kombinēta vārda un režģu formā (sk. zīm.3.). Režģis sastāv no saitēm, mezgliem un blokiem.

### A. Vārda un Tālruņa Saites.

Mazāka režģa komponente ir saite. Saite savieno divas mezglus un pārstāv vienu vārdu vai tālruņi. tas satur iespējamību, kā arī mezgla atlīdzība (inducējot mērķa mezglu- mezgla kura saite beigsies).

### B. Režģu mezgli.

Mezģls pārstāv punktu laikā kurā sākas runa. Tas satur laiku, kā arī vienu vai vairāk vārdu vai tālruņa saistu, un papildus spīkera saiti. Mezģls saistīts tikai ar vienu spīkeru. Kas režģis pārsvarā sastāv no mezgliem no viena spīkera, pamata spīkers var būt nozīmēts un spīkera atribūti mezglā ir izņemtas lai paaugstināt atmiņas efektivitāti.



Var redzēt ka kopējā visu vārdu saistu varbūtība vai visu tālruņu saistu varbūtība, izejošā no mezgla, vajag summēt vienam vai mazāk. Summa mazāka neka viens atslogojošu ceļu trūkšanu. Tomēr, vārdu un tālruņu saites ir neatkarīgie aspekti tādas pašas varbūtības vietas un viņu varbūtības nav triviāli kombinēti.

### C. Režģu Bloki.

RunasSatura Režģis var, acīmredzot, pārstāvēt daudz stundas anotāciju. Lai atvieglotu RunasSatura režģa uzprasīšanu platformās ar limitētu atmiņu, režģi sadalīta blokos. Tas ļauj uzticamo, ātrāk neka absolūto, mezglu un laiku indeksēšanu ar atmiņas sekojošo saglabāšanu.

Režģa bloks sastāv no:

1) *Bloka starta laiks*- Tas ir laiks no kura mezglu laiku ir nobīdīti, un ļauj mezglu nobīdīšanas laikiem turēties ar 1/100 no sekundes precizitāti ar mazo atmiņas izmantošanu. Šī precizitāte var būt pieprasīta runas masīvos.

2) *Pamata spīkeri*- Vairākos gadījumos, režģa bloks pārstāvēs vienu spīkeru. Pamata spīkers ļauj izmest spīkera identifikatoru asociētu ar mezglu šajos gadījumos.

3) *Svarīgums*, asociēts ar izdabūto režģi, ir atkarīgs no dominējoša trokšņa stāvokļa. Audio atribūti ir uzskaitīts tips ar atļautām vērtībām: "unknown", "speech", "noise" un "noisySpeech".

RunasSatura režģos tādā veidā sastāv no blokiem, katrs no kuriem sastāv no mezgliem un saitēm.

### VII. Eksperimentālais modelis

Daudz darba jau ir izpildīts runas dokumentu atpazīšanas jomā. RunasSatura struktūrai, aprakstītai augstāk, mēs izskaitļojām to izmantojot parastu dokumentu skaitļošanu

$$S(d; q) = S_w(d; q) + S_s(d; q) + S_p(d; q)$$

kur  $d$  ir runas dokuments MPEG-7 formātā,  $q$  ir teksta pieprasījums un kur

$$S_w(d; q) = \sum_{w_d, w_q} P(w_d) \delta(w_d, w_q)$$

$w_d$  un  $w_q$  ir vārdi dokumentā un pieprasījumā. Morfoloģiskais termins dots pēc

$$S_w(d; q) = \lambda \sum_{m_q = M(w_q), m_d = M(w_d)} P(w_d) \delta(m_q, m_d)$$

kur  $M(x)$  ir vārda kāts pēc stop- vārda izņemšanas un morfoloģiska vārda katošanas. Beigās, tālruņa rēķins ir dots pēc

$$S_p(d; q) = \eta \sum_{T_q, T_d} P(T_d) O(T_q, T_d)$$

$T_x$  ir tālruņa trigramma un  $O(x,y)$  ir tālruņa trigrammas pārklāšanas funkcija. Mūsu eksperimentos mēs izmantojam delta funkciju, tomēr statistiska pārklāšanas funkcijas varēja izmantoties. Maisījuma parametri ir uzstādīti uz  $\lambda = \eta = 0.3$ .

Labošanas algoritms ir speciāli nav mākslas stāvoklis. Kad režģis ir pazīstams iepriekš, piemēram, lielas anotācijas, mēs varam utilizēt tādu metodes kā pieprasījuma paplašināšana lai pamatīgi uzlabot labošanas rezultātus. Tiešām, MPEG-7 SkaņasSatura aprakstītājs saglabā visu informācijas pamatu modernajām sistēmām izmantotām TREC sistēmās, mēs varam gaidīt kā viņu rezultātus varēs pielietot šai datu struktūrai. Mūsu izdabūšanas eksperimentu mērķis ir kombinētu vārdu un režģu svarīguma demonstrēšana izdabūšanai pat grūtajos apstākļos. Tas var būt ilustrēts ar īsu

anotāciju kolekciju, kurus var tipiski atrast audio virsrakstos. Mēs paņēmām datu bāzi no 2907 dokumentiem ar vidējo ilgumu ~8.5 s izmantojot ASR sistēmu ar 20K vārdnīcu un 40% vārdu precizitātei. Izdabūšanas rezultāti ir doti tab.1.

Acīmredzot, ka informācija, saturoša gan vārda, gan tālruņa režģis komponentes ir obligāti un cildinoši. Kur vārdi ir ne no vārdnīcas vai tos nevar atpazīt izdabūšanas kļūdu vai trokšņa dēļ, fonētiska sakrišana var ievērojami paātrināt atpazīšanas procesu. Kur lielas anotācijas un pieprasījumi ir iespējami un ļoti lielas precizitātes sistēmas izmantojās, RunasSatura režģu saturs var būt atpazīta tikai ar dažām hipotēzēm. Tāpēc, MPEG-7 RunasSatura komponente var atbalstīt liela ranga audio kvalitātes un saturus ar papildus varbūtību un multi līmeņu reprezentāciju lielumu pievienošanu.

Tab.1. Atpazīšanas rezultāti īsai runas dokumenta anotācijai.

Number of queries	Number of words in query					
	2		3		4	
	25		13		8	
	Recall	Precision	Recall	Precision	Recall	Precision
Word query	0.41	0.30	0.47	0.23	0.48	0.24
Phonetic query	0.43	0.13	0.52	0.14	0.62	0.18
Word + Phonc query	0.51	0.17	-	-	-	-

## VIII. Kopsavilkums

Šajā papīrā mēs demonstrējam, ka neskatoties uz ASR izejas piemītiem kļūdām, darbojošais RunasSatura aprakstītājs var būt izveidots. Saglabājot saturu multilīmeņu reprezentācijā izmantojot vārdus un tālruņus, mēs varam atbalstīt izdabūšanu, pat ja atpazīšanas instrumentā bija kļūda. Caur statistiskas informācijas aizturēšanu, gan režģis, gan neskaidrību statistikā, raksturojot kļūdas atpazīšanas instrumentos, mēs varam garantēt spēju sadarboties heterogēnu datu bāzēs.

# MPEG-7 skaņas atpazīšanas instrumenti.

Abstrakti- MPEG-7 skaņas atpazīšanas aprakstītāji un aprakstītāju shēmas sastāv no instrumentiem audio medija indeksēšanai izmantojot varbūtības skaņas modeļus. Aprakstītāji sniedz konteinerus kategoriju zīmēm, kā arī datu struktūrām ciparu informācijai par skaņas saturu. Mēs aprakstām normatīvus instrumentus, kā arī informatīvus metodes, aprakstīšanas automātiskajai dabūšanai un skaņu sakrišanai.

Indekss- dimensiju samazināšana, pabeigtais modelis, paraugu sakrišana, skaņas atpazīšana, spetrālas atšķirības, taksonomija.

## I. Ievads.

Audio medija sastāv no plaša ranga skaņu parādu, ieskaitot apkārtējas skaņas, fona skaņas, foley skaņas, dzīvnieku skaņas, runas skaņas, nerunas skanēšanu, un mūziku. Skaņas atpazīšanas instrumenti izstrādāti medija meklēšanai automātiski indeksējot skaņas gabalus, tādus kā filma, ar skaņas satura aprakstu. Nesenu pētījumu parauga atpazīšanā dēļ deva tādu automātisku atpazīšanu iespējamu.

MPEG-7 skaņas atpazīšanas instrumenti nodrošina kopējo interfeisu audio automātiskai indeksēšanai izmantojot trenētas skaņas klases parauga atpazīšanas struktūrā. Aprakstīšana ir sadalīta divos tipos: teksta- bāzēta aprakstīšana pēc kategorijas marķētājiem un daudzuma aprakstīšana izmantojot varbūtības modelis.

Kategoriju aprakstītāji sniedz kvalitatīvu informāciju par skaņas satura dabu. Aprakstītāji šajā formā ir derīgi teksta bāzētiem pieprasījuma lietojumprogrammām, tādiem kā Interneta meklēšanas dienesti, vai jebkuram apstrādes instrumentam, kurš lieto teksta laukumus. Daudzuma aprakstītāji sastāv no kompakta operatīvas informācijas par audio segmentiem, kuri var izmantoties skaņas līdzīgumu ciparu novērtējums. Šie pēdējie aprakstītāji labi der pieprasījuma- pēc- parauga lietojumprogrammām.

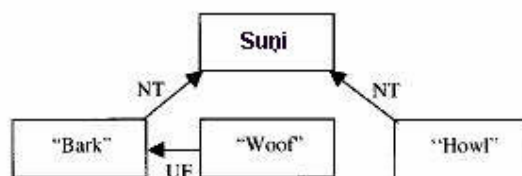
Normatīvie instrumenti bija izstrādāti lai ļautu dažādu veidu izņemšanas izpildīšana ar pamata motivāciju dažādu izņemšanas instrumentu savienojamībai. Ka arī informatīvie instrumenti bija attīstīti lai strādātu kā piemēri lietojumprogrammu īstenošanai izmantojot MPEG-7 aprakstītājus un aprakstīšanas shēmas.

## II. Skaņas atpazīšanas aprakstītāji un aprakstītāju shēmas.

### A. Kvalitātes aprakstītāji.

Starp skaņu segmentu klasifikāciju iemesliem ir vēlšana saprast kaut ko par saturu. Piemēram, kļiedziens filma skaņā var inducēt bīstamību, bet smieklis var nozīmēt komēdiju. Strukturēta indeksēšana ar hierarhiskām kategorijām dod gan absolūto kategoriju koncepciju skāņam, gan norādījumus par kategoriju. SkaņasKategorija un KlasifikācijasShēma aprakstīšanas shēmas noteic kategoriju zīmes un attiecības starp kategorijām.

1) *SkaņasKategorija*: Šī aprakstīšanas shēma ir izmantota skaņu kategoriju apzīmēšanai un tas dabūts no KontrolētaisTermiņš aprakstīšanas shēmas, kura ir noteikta multimediju aprakstīšanas shēmās (MDS) standarta daļā. kā piemērs, suņa riešanas skaņa var būt ievietots kategorijā "Suns" ar "Riet" kā apakškategorija. Starp to "Woof" un "Howl" var būt kategorijas "Suns" apakškategorijas. Pirmajās divas apakškategorijās ir tuvi saistīti, bet trešā ir pavisam cits skaņas notikums.



zīm.1. Skaņu kategoriju parasta taksonomija.

Zīm.1. rada četrus kontrolētus terminus organizētus "Suns" tasonomijā kā cēloņa mezgls. Katram terminam ir vismaz viena attiecību saite citam terminam šajā tasonomijā. Pēc noklusēšanas, saturētais termins uzskatīts par vairāk šauru terminu (ŠT) nekā saturošais termins. Tomēr, šajā piemērā, "Woof" ir noteikts ka vairāk sinonīms, bet mazāk vēlējams, "Bark" terminam. Lai fiksētu šādu struktūru, sekojošas attiecības ir pielietojami kā daļa no KontrolētaisTermiņš aprakstīšanas shēmas.

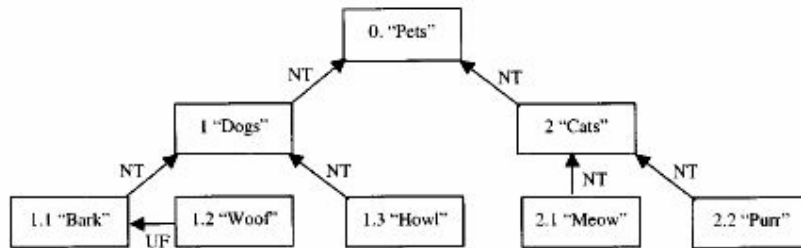
- 1) Plašākais termins (PT)- saistīts termins ir vairāk noteiktais nozīmīgumā nekā saturošais termins.
- 2) Vairāk šaurais termins (ŠT)- saistīt termins vairāk noteiktais nozīmīgumā nekā saturošais termins.
- 3) Izmantošana to vietā (IV)- saistīts termins ir sinonīms saturoša termina, bet saistīts termins vairāk vērtēts nekā saturošais termins.
- 4) Izmantošana kam (IK)- saistīts termins ir sinonīms saturošam termina, bet saturošais termins vairāk vērtēts nekā saistīts termins.
- 5) Saistīts termins (ST)- saistīts termins nav sinonīms, kvazi- sinonīms, šaurāks vai plašāks termins, bet ir saistīts ar saturošo terminu.

Sekojošais XML- shēmas kods kā ievietot SkaņasKategorija aprakstīšanas shēmu "Suns" kategorijā izmantojot MPEG-7 aprakstīšanas noteikšanas valodu (ANV):

```
<SoundCategory term= `` 1" scheme= `` DOGS">
<Label>Dogs<=Label>
<TermRelation term= `` 1.1" scheme= `` DOGS">
<Label>Bark<=Label>
<TermRelation term= `` 1.2"
scheme= `` DOGS" type= `` UF">
<Label>Woof<=Label>
<=TermRelation>
<=TermRelation>
<TermRelation term= `` 1.3" scheme= `` DOGS">
<Label>Howl<=Label>
<=TermRelation>
<=SoundCategory>
```

Termina un shēmas atribūti kopā nodrošina unikālus identifikatorus kuri ir izmantojās saitēm kategorijās un taksonomijās no citām aprakstīšanas shēmām, kā varbūtības modelēs. Zīmes aprakstītājs dod zīmīgu semantisku zīmi katrai kategorijai un TerminuAttiecību aprakstītājs apraksta attiecības starp terminiem tasonomijā.

2) *KlasifikācijasShēma*: kategorijas var būt kombinēti klasifikācijas shēmās lai veidotu bagātāku taksonomiju; piemērām, "Riet" ir "Suns" apakškategorija kura ir kategorijas "Dzīvnieki" apakškategorija; kā arī kategorija "Kaķi". Sekojošais piemērs "Dzīvnieki" parasta klasifikācijas shēma saturoša divas kategorijas: "Suņi" un "Kaķi" (zīm.2.).



zīm.2. Kategoriju kombinēšana lielākā tasonomijā.

Lai īstenotu šo klasifikācijas shēmu pagarinot iepriekšējo shēmu, otrā shēma sauktā "Kaķi" ir ievietota:

```
<SoundCategory term= `` 2" scheme= `` CATS">
<Label>Cats<=Label>
<TermRelation term= `` 2.1" scheme= `` CATS">
<Label>Meow<=Label>
<=TermRelation>
<TermRelation term= `` 2.2" scheme= `` CATS">
<Label>Purr<=Label>
<=TermRelation>
<=SoundCategory>
```

Tagad lai apvienotu šīs kategorijas, KlasifikācijasShēma saukta "Dzīvnieki" ir ievietota lai savienotos ar iepriekšēji noteiktām shēmām.

```
<ClassificationScheme term= `` 0"
scheme= `` PETS">
<Label>Pets<=Label>
<ClassificationSchemeRef scheme= `` DOGS"/>
<ClassificationSchemeRef scheme= `` CATS"/>
<=ClassificationScheme>
```

Tagad klasifikāciju shēma sauktā "Dzīvnieki" sastāv no visa kategoriju "Suņi" un "kaķi" satura ar kategoriju "Dzīvnieki" kā pamatu. Kamēr taksonomijas formēšana var būt pietiekama dažām indeksēšanas lietojumprogrammām,

tas ir tikai pirmais solis automātiskas skaņas klasifikācijas virzienā. Nākošās sekcijas izdala daudzuma aprakstītajos klasificēšanai un indeksēšanai.

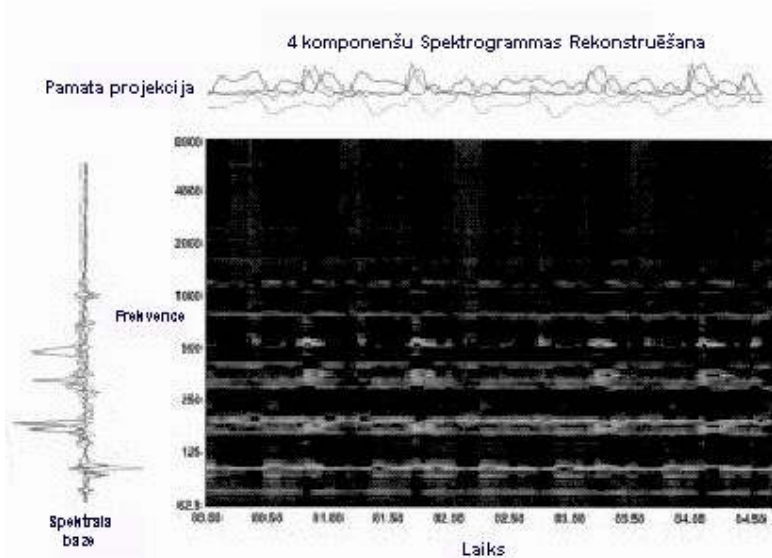
### B. Daudzuma aprakstītāji.

Skaņu atpazīšanas daudzuma aprakstītāji apraksta audio signāla īpašības izmantošanai ar statistiskiem klasifikatoriem. Skaņas atpazīšanas daudzuma aprakstītāji var izmantoties pamata skaņas atpazīšanai ieskaitot skaņas efektus un muzikālus instrumentus. Papildus piedāvātiem aprakstītājiem, jebkurš cits aprakstītājs, noteikts skaņu struktūru ietvaros, var izmantoties klasificēšanai. Tomēr, normatīvo izņemšanas metožu izmantošana ir pieprasītā dažādu atpazīšanas metožu savienojamībai:

1) *AudioSpektraBaze*: skanu klasifikāciju vairāk pielietojamu īpašību vidū ir spektra bāzētie attēlojumi, tādi ka jaudas spektrs. Tomēr, spektra- dabūtie dati galvenokārt nav savietojami ar varbūtības modeļu klasifikatoriem lielas dimensijas dēļ. Katrs spektra slānis ir n- dimensiju vektors, ar n- spektrālo kanālu numurs, kurš parasti sastāv no 64 un 1024 datu kanāliem. Logaritmiskās frekvences spektrs, ka parādīts audio struktūras aprakstītāja *AudioSpektaApvalks*, palīdz samazināt dimensiju līdz apmērām 32 kanāliem, bet varbūtības klasifikatori strādā labāk ar mazāk nekā desmit dimensijām.

**AudioSpektaPamats aprakstītājs ir tvertne pamata funkcijām kuri ir izmantoti spektruma projektēšanai mazāk- dimensiju līmeni, kurš der varbūtības modeļu klasifikatoriem. Pamats izskaitļots katrai skaņu klasei kurš notver statistiski vairāk regulāras pazīmes no skaņas pazīmju daudzuma. Dimensiju samazināšana parādās projektējot spektra vektorus pret matricu no dabūtiem no datiem pamata funkcijām. Pamata funkcijas glabājas matricas kolonnās kurā rindu skaitlis sakrīt ar spektra vektora garumu un kolonnu skaitlis sakrīt ar pamata funkciju numuru.**

Pamata projektēšana ir spektra un bāzes vektoru matricas produkts. Zīm.3. rada spektrogrammu, rekonstruētu no četrām bāzes funkcijām. Vektori pa kreisi kombinēti ar vektoriem virs figūras, izmantojot vektoru ārējo produktu, katra rezultāta matrica summējas lai veidotu beigas rekonstrukciju.



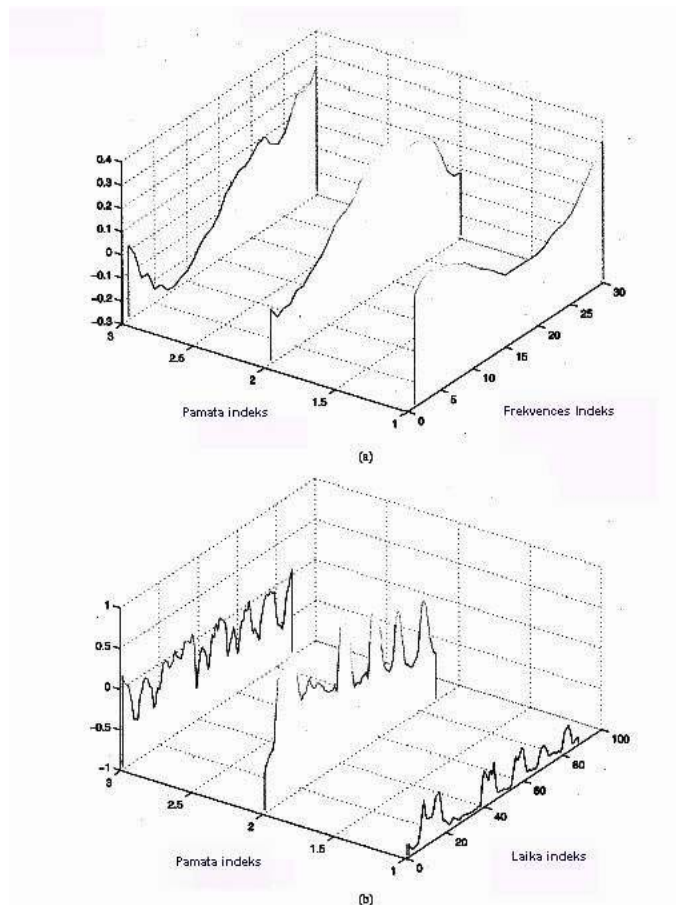
zīm.3. Spektrogrammas rekonstruēšana no pamata funkcijām.

Pamata funkcijas izvēlētas lai maksimizēt informāciju dimensiju mazāka skaitlī nekā oriģinālajos datos. Piemēram, pamata funkcijas atbilst nekorelētiem īpašībām izņemti izmantojot principālo komponentu analīzi (PKA) vai Karhunen- Loeve (KL) transformāciju, vai statistiski neatkarīgu komponentu izņemto ar neatkarīgo komponentu analīzi (NKA). Klasifikācijas mērķiem, pamats dabūts līdz pilnajai klasei; tādā veidā, klasifikācijas vieta sastāv no vairāk statistiski zīmīgiem klases komponentiem. Sekojošais DDL gabals definē pamata projektēšanas matricu kura samazina logaritmisko frekvences spektra rindu ar 31 kanālu līdz piecām dimensijām:

```
<AudioSpectrumBasis loEdge= `` 62.5"
hiEdge= `` 8000" resolution= `` 1/4 octave">
<Matrix dim= `` 31 5">
0.26 0:05 0.01 0:70 0.44
0.34 0.09 0.21 0:42 0:05
0.33 0.15 0.24 0:05 0:39
0.33 0.15 0.24 0:05 0:39
0.27 0.13 0.16 0.24 0:04
0.27 0.13 0.16 0.24 0:04
0.23 0.13 0.09 0.27 0.24
0.20 0.13 0.04 0.22 0.40
0.17 0.11 0.01 0.14 0.37
<!-- etc. -->
<=Matrix>
<=AudioSpectrumBasis>
```

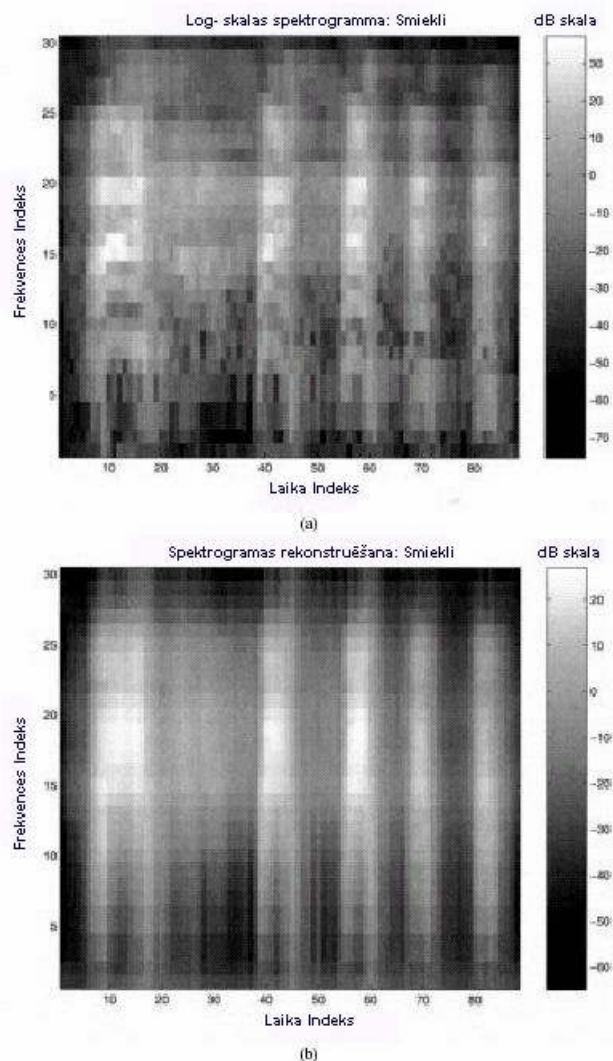
*loEdge*, *hiEdge* un *resolution* atribūti dod zemākas un augšējās pamata funkcijas frekvences robežas un spetrālo kanālu intervālu oktāvu- sakaru apzīmēšanā. MPEG-7 klasifikācijas struktūrā, pamata funkcijas veselai skaņu klasei glabājas ar klases varbūtības modeli.

- 2) *SkaņasAtpazīšanasPazīmes*: Pazīmes lietotas skaņas atpazīšanai savākti viņa aprakstītāju shēmā, kuru var izmantot dažādu lietojumprogrammu kopā. Šie aprakstītāji parādījās lai labi izpildītu dažādu skaņu tipu klasifikāciju. Tomēr, aprakstīšanas shēma arī ļauj klasifikatora alternatīvas izpildīšanas kuri var lietot attiecīgus aprakstītājus ja vajadzīgi to darīt.



zīm.4. (a) Spektrālie pamata funkcijas Smiekla funkcijai; (b) Pamata projekcija smiekla spektrogrammai.

Pamata pazīmes dabūti no AudioSpektraApvalka dabūšanas procesa. Katra spektra slānim, jauda ir izrēķināta un apstrādājas izmantojot AudioJauda aprakstītāju. AudioSpektraPamataProjekcija aprakstītājs ir tvertne dimensiju-samazinātiem pazīmēm kuri ir dabūti projektējot spetrālo apvalku pret bāzes funkciju komplektu, kā bija aprakstīts iepriekš. Zīm.4(a) un (b) rada trīs pamata komponentes un rezultāta pamata projektēšanu spektrogrammai zīm.5.



zīm.5. Smieklu Spektrogrammas un trīs pamata funkciju rekonstruēšana.

Pamata aprakstītāju papildus, neierobežota secība alternatīvo aprakstītāju var izmantoties klasifikatoru definēšanai kuri izmanto skaņu klašu specialas iespējas, tādus kā harmoniskais apvalks un fundamentāla frekvence īpašības, kuras izmantojās muzikālo instrumentu klasificēšanā. Viena ērtība no dimensiju samazināšanas ir tas, ka jebkurš aprakstītājs balstās uz mērogojamiem rindām kuri var pievienoties spektrāliem aprakstītājiem ar to pašu parauga tempu un piemērots pamats var izskaitļoties veselam paplašinātam pazīmju skaitam tādā pašā veidā kā tikai spektra pamatam.

3) *Spektrogrammu summēšana ar bāzi*: alternatīva lietojumprogramma SkaņasAtpazīšanasPazīmes aprakstītāju shēmai ir efektīva spektrogrammu reprezentācija. AudioSpektraPamataProjektēšana un AudioSpektraPamats pazīmes var izmantoties kā ļoti efektīvs uzglabāšanas mehānisms spektrogrammām vizualizācijai un summēšanas nodomiem. Lai rekonstruētu spektrogrammu, mēs izmantojam (2) kurš būvē 2-D spektrogrammu no katras pamata funkcijas krustiska produkta un tas atbilstošu spektrogrammai pamata projekciju. Zīm.5. rada spektrogrammu, parādītu visā garuma ar mazas dimensijas pamata rekonstrukciju. Datu uzglabāšanas pieprasījumi katrai reprezentācijai ir sekojoši: ieraksta spektrogrammai ir 88 laika punkti ar 30 kanāliem, tādā veidā pieprasot 2649 peldošo punktu vērtību. Pamata funkcijas prasa trīs kanālus no 88 laika punktiem un trīs 30- punktu spektrālie vektori tādā veidā prasot 354 peldošo



punktu vērtību lai rekonstruētu spektrogrammu ka parādīts zīmējumā, tas sastāda datu samazināšanu ar faktoru aptuveni 7.5 dotajā gadījumā.

### C. Varbūtības modeļu aprakstīšanas shēmas.

Varbūtības modeļu aprakstīšanas shēmas aprakstīti MPEG-7 MDS daļā. Standartizācijas motivācija vadās ar vesela klasifikatora konstrukcijas un trenēšanas maksu. Lietojumprogrammas, kuras lieto statistiska modeļa satura klasifikācijai iegūs daudz labuma no standarta interfeisa varbūtības modelim. Ar šiem standartizētām shēmām rokās kļūst iespējams dalīties ar trenētiem varbūtības modelim starp lietojumprogrammām pat ja dabūšanas metodes atšķiras. Sekojošās aplūko varbūtības modeļu aprakstīšanas shēmu derīgumu skaņu atpazīšanai.

1) *IerobežotaStāvokļaModele*: skaņas parādība ir dinamiska. Spektrālas vērtības mainās laika, un tā mainīšana kura dod skaņai tai raksturotus nospiedumus atpazīšanai. MPEG-7 skaņas atpazīšanas modeļu skaņas klašu sadalīšana ierobežota skaitļa stāvokļos pamatots uz spetrāliem pazīmēm; individuālas skaņas aprakstītas pēc viņu trajektorijām caur šo stāvokļu telpu. Katrs stāvoklis ir modelēts ar nepārtrauktu varbūtības sadali tādu kā Gausa.

Skaņas klases dinamiskā uzvedība caur stāvokļu telpu ir modelēta ar  $k \times k$  pārejas matricu kura apraksta pārejas varbūtību katram no  $k$  stāvokļiem modelē dotā stāvoklī. Pārejas matricai  $\mathbf{T}$ ,  $i$ -ta rinda un  $j$ -ta kolonna ir pārejas varbūtība stāvoklī  $j$  laikā  $t$  stāvokļa  $i$  laikā  $t-1$ .

Sākuma stāvokļa sadale, kura ir  $k \times 1$  varbūtību vektors, ir arī pieprasīts ierobežota stāvokļa modeļa. Vektora  $k$ -tais elements ir varbūtība būšot stāvoklī  $k$  pirmā apskatīšanās struktūrā.

2) *GausaSadalesTips*: Multidimensionāla Gausa sadale lietojās klašu modelēšanai skaņu klasifikācijā. Gausa sadale ir parametrizēta pa  $1 \times n$  vektoram ar vidējo  $m$  un  $n \times n$  covariance matricu  $\mathbf{K}$ , kur  $n$  ir pazīmju numurs katrā novērošanas vektorā. Varbūtību izskaitļošanas izteiksme nejaušam vektoram  $\mathbf{x}$  dotie Gausa parametri ir

$$f_{\mathbf{x}}(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{n}{2}} |\mathbf{K}|^{\frac{1}{2}}} \exp \left[ -\frac{1}{2} (\mathbf{x} - \mathbf{m})^T \mathbf{K}^{-1} (\mathbf{x} - \mathbf{m}) \right].$$

3) *NepārtrauktaSlēptaisMarkovaModelis*: Nepārtraukta slēptais Markova modelis ir ierobežota stāvokļa modelis ar nepārtraukto varbūtību modeli stāvokļa novērošanas varbūtībai. Sekojoša DDL gabals ir varbūtības modeļa aprakstīšanas shēmas izmantošanas piemērs nepārtraukta slēpta Markova modeļa ar Gausa stāvokļiem attēlošanā; šajā piemērā, peldošo punktu numuri bija noapaļoti līdz diviem decimāliem vietām tikai parādīšanai:

```
<ProbabilityModel
xsi:type="` ` ContinuousMarkovModelType"
numberStates="` ` 7">
<Initial dim="` ` 7">
0.04 0.34 0.12 0.04 0.34 0.12 0.00
<=Initial>
<Transitions dim="` ` 7 7">
0.91 0.02 0.00 0.00 0.05 0.01 0.01
0.01 0.99 0.00 0.00 0.00 0.00 0.00
0.01 0.00 0.92 0.01 0.01 0.06 0.00
0.00 0.00 0.00 0.99 0.01 0.00 0.00
0.02 0.00 0.00 0.00 0.97 0.00 0.00
0.00 0.00 0.01 0.00 0.00 0.98 0.01
```

```

0.02 0.00 0.00 0.00 0.00 0.02 0.96
<=Transitions>
<State><Label>1<=Label></State>
<!--State 1 Observation Distribution -->
<ObservationDistribution
xsi:type= `` GaussianDistributionType">
<Mean dim= `` 6">
5.11 9:28 0:69 0:79 0.38 0.47
<=Mean>
<Covariance dim= `` 6 6">
1.40 0:12 1:53 0:72 0.09 1:26
0:12 0.19 0.02 0:21 0.23 0.17
1:53 0.02 2.44 1.41 0:30 1.69
0:72 0:21 1.41 2.27 0.15 1.05
0.09 0.23 0:30 0:15 0.80 0.29
1:26 0.17 1.69 1.05 0.29 2.24
<=Covariance>
<State><Label>2<=Label><=State>
<!--Remaining states use same structures-->
<nProbabilityModel>

```

VarbūtībasModelis ir deklarēts kā abstrakts tips standartā un viņu vajag ievietot ar xsi:type norādījumu. Šajā piemērā, VarbūtībasModelis ir ievietots kā GausaSadalesTips, kurš dabūts no VarbūtībasModelis klases bāzes.

#### D. Skaņas Atpazīšanas modeļu Aprakstīšanas Shēmas.

Struktūras, aprakstītas augstāk, ir izolētas, viņām pašām nav nekādas lietošanas struktūras. Sekojošie datu tipi kombinē šos aprakstītajos un aprakstītāju shēmas unificētā struktūrā skaņas klasificēšanai un indeksēšanai. Skaņas segmenti var būt indeksēti ar kategorijas zīmi bāzētas uz klasifikatora izejas un, papildus, varbūtības modeļi parametri var izmantoties skaņas indeksēšanai datu bāzē. Indeksēšana pēc modeļa parametriem, tādiem ka stāvokļi, ir nepieciešams pieprasījumam- pēc- piemēra programmām kurās pieprasījuma kategorija ir nezināma vai kad šaurāka kritērija sakrišana nekā kategorijas kompetence ir vajadzīga.

1) *SkaņasAtpazīšanasModelis*: SkaņasAtpazīšanasModelis aprakstītājs noteic skaņas klases varbūtības modeli, tādus kā slēptais Markova modelis vai Gausa maisījuma modelis. Sekojošais piemērs ir daļa no slēpta Markova modeļa "Riešana" skaņas kategorijas. Varbūtības modeļu un asociētas pamata funkcijām skaņu klasei vajag būt definētam tajā pašā veidā kā iepriekšējos piemēros

```

<SoundRecognitionModel id= `` sfx1.1"
SoundCategoryRef= `` Bark">
<ExtractionInformation term= `` Parameters"
scheme= `` ExtractionParameters">
<Label>NumStates=7,
NumBasisComponents=5<=Label>
<=ExtractionInformation>

```

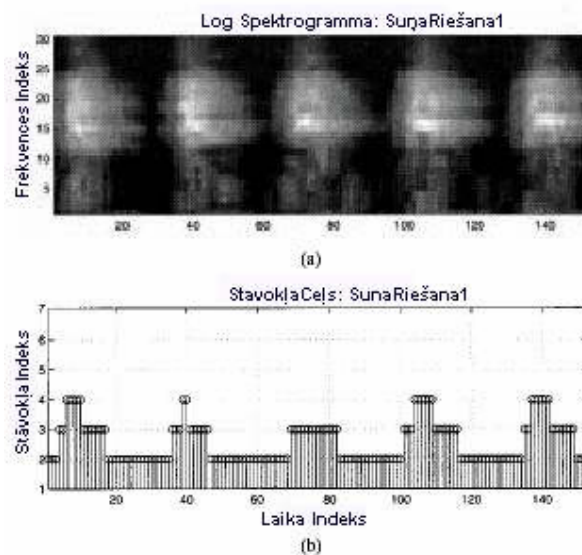
```

<ProbabilityModel
xsi:type="``ContinuousMarkovModelType"
numberStates="``7">
...<!-- see previous example -->

<=ProbabilityModel>
<SpectrumBasis loEdge="``62.5" hiEdge="``8000"
resolution="``1/4 octave">
...<!-- see previous example -->
<=SpectrumBasis>
<=SoundRecognitionModel>

```

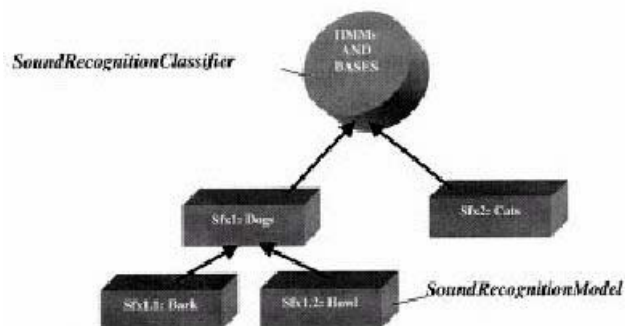
2) *SkaņasModelisStāvokļaTrajektorija*: Šis aprakstītājs griežas pie ierobežota stāvokļa modeļa un apraksta dinamiska skaņas stāvokļa trajektoriju caur modeli. Skaņas ir indeksētas pēc segmentācijas modeļu stāvokļos vai pēc stāvokļa trajektorijas parauga pēc regulāriem intervāliem. Pirmajā gadījumā, katrs audio segments satur norādījumu stāvoklim, un segmenta ilgums rada stāvokļa aktivizācijas ilgumu. Otrajā gadījumā, skaņa ir aprakstīta ar parauga indikatoru sēriju kuri norāda uz modeļu stāvokļiem. Skaņu kategorijas ar gariem stāvokļu ilgumiem ir efektīvi aprakstīti izmantojot viena- segmenta, viena- stāvokļa pieeju. Skaņas ar īsiem stāvokļa ilgumiem ir vairāk efektīvi aprakstīti ar parauga stāvokļu indikatoru sēriju. Zīm.6. rada suņa riešanas spektrogrammu pavadītu ar *SkaņasModelisStāvokļaTrajektorijas* stāvokļu secību riešanas modelim.



zīm.6. (a) Suņa riešanas spektrogramma (b) Stāvokļa ceļš caur slēpto Markova modeļi.

3) *SkaņasAtpazīšanasKlasifikātors*: Šī aprakstīšanas shēma tieši atbalsta skaņas klasifikatoru programmas apgādānot ar vienīgu tvertni visiem vajadzīgiem klasifikatora komponentiem. *SkaņasAtpazīšanasKlasifikātors* apraksta attiecības starp varbūtības modeļu numuru caur klasifikatoru ontoloģijas definēšanu. Piemērām, hierarhiska atpazīšanas ierīce var būt aprakstīta ka klasifikatoru plaša skaņu klase, tāda kā dzīvnieki, cēloņa mezglā un labākas klases, tādas kā suņi:riešana un kaķi:meow, lapu mezglas. Shēma definē plānu starp klasifikatoru ontoloģiju un skaņas kategoriju taksonomiju izmantojot Graph DS struktūru, tādā veidā dodot iespēju hierarhisku skaņas modeļu izmantošanu kategorijas izņemšanas aprakstītajos dotajai taksonomijai (sk. Zīm.7.)

Ar šo shēmu, mēs beidzam Aprakstīšanas shēmu apspriešanu un pārslēdzam uzmanību izvilksanas instrumentiem un programmām.



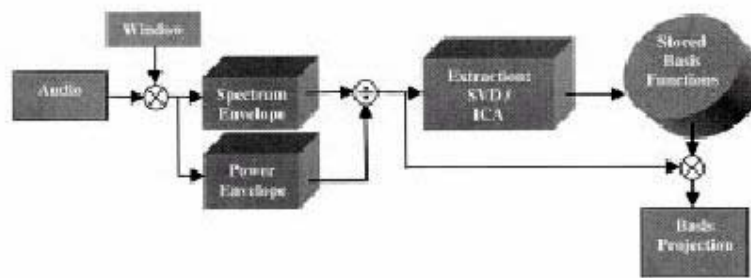
zīm.7. Skaņas atpazīšanas modeļu kombinēšana klasifikatoru ontoloģijā.

### III. Skaņas- Atpazīšanas Klasifikatora Veidošana.

Modeļu veidošanas process īstenots MPEG-7 eksperimentālajā modelī (XM) aprakstīts sekojošās nodaļās. Izstrādātais autonomais instruments struktūras izdabūšanas izpildīšanai no direktorijas ar WAV formāta audio failiem un slēpta Markova modeļa trenēšana ar šīm pazīmēm. Skaņu piemēru direktorija ir pieprasīta katrai skaņu klasei un hierarhiskā direktoriju struktūra definē ontoloģiju, kura atbilst vēlējama taksonomijai. Viens slēptais Markova modelis ir trenēts katrai direktorijai ontoloģijā atsevišķi.

1) *HMM Modeļa trenēšana*: Liela daļa no sasniegumiem klasifikatora projektēšanā ir tērēta savākot un gatavojot trenēšanas datus. Skaņu diapazonam vajag atspoguļot skaņu kategorijas darbības diapazonu. Piemērām, suņa riešana var ietilpst individuālas riešanas, kopējas riešanas secībā, vai daudz suņu var riet vienlaikus. Modeļa izdabūšanas algoritms adaptēsies datu diapazonam, un tādā veidā šaurākais piemēru diapazons veidos vairāk specificētu klasifikatoru.

2) *Audio Pazīmes Izdabūšana*: Sistēmas diagramma zīm.8. rada izdabūšanas shēmu AudioSpektraPamatam un SkaņasAtpazīšanasPazīmēm. Audio viļņa forma ir logota struktūrās un īsa laika logaritmiskais laikā spektrs ir izdabūts. Ja spektra izdabūšana izmantoja jaudas spektru, kā standartizēts AudioSpektraApvalka, pēc tam šajā stadijā spektrs ir konvertēts no jaudas spektra decibelu skalas amplitūdas spektrā kurš labāk der klasifikācijai. Katrs spektra slānis ir sadalīts ar to jaudu, un rezultāta normalizēts spektra apvalks padots pamata izdabūšanas procesā.



zīm.8. Zema līmeņa audio īpašību izdabūšana skaņas atpazīšanas klasifikatoram.

Spektra pamata funkcijas ir izskaitļoti no normalizēta spektra apvalka izmantojot pamata dekompresijas algoritmu tādu ka vienskaitļa vērtības sadalīšana (VVS), kura īstenībā izskaitļo PCA, vai neatkarīgu komponentu analīzi (NKA). Labi zināms NKA algoritms, tāds kā *INFOMAX*, *JADE* vai *FASTICA*, der pamata izdabūšanai. Spaktrogramas matrica ir dekompresēta pamata komponentu lineāra kombinācijā no kuriem mazs skaitlis saglabāts kuri atbilst vajadzīgai dimensijai izejas pazīmēm; parasti starp trešo un desmito ir pietiekami izdabūšanai.

Specifiskā pamata funkciju izvēle ir nenormatīva bet tā ir gaidīta, maksimālas savienojamības iemesliem starp programmām, ka pamats būs izvēlēts lai būtu kolonnas ar vienības L2- norm un ka viņi maksimizē informāciju  $k$  dimensijās ar attiecību citām iespējamām pamata funkcijām. Pamata funkcijas var būt ortogonāli kā dots ICA izdabūšanā. Pamata projekcija un rekonstrukcija ir aprakstīta ar sekojošiem analīzes sintēzes vienādojumiem:

$$\mathbf{Y} = \mathbf{XV} \quad (1)$$

$$\mathbf{X} = \mathbf{YV}^+ \quad (2)$$

kur

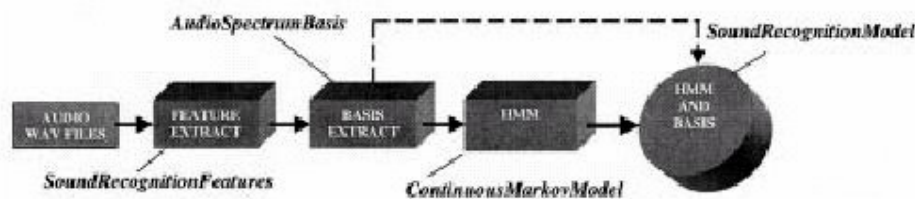
$\mathbf{Y}$  izdabūta  $m \times k$  novērošanas matrica;

$\mathbf{X}$   $m \times n$  spektra datu matrica ar spektrāliem vektoriem organizētiem rindu plašumā;

$\mathbf{V}$   $n \times k$  matrica no bāzes funkcijām organizēta kolonnās.

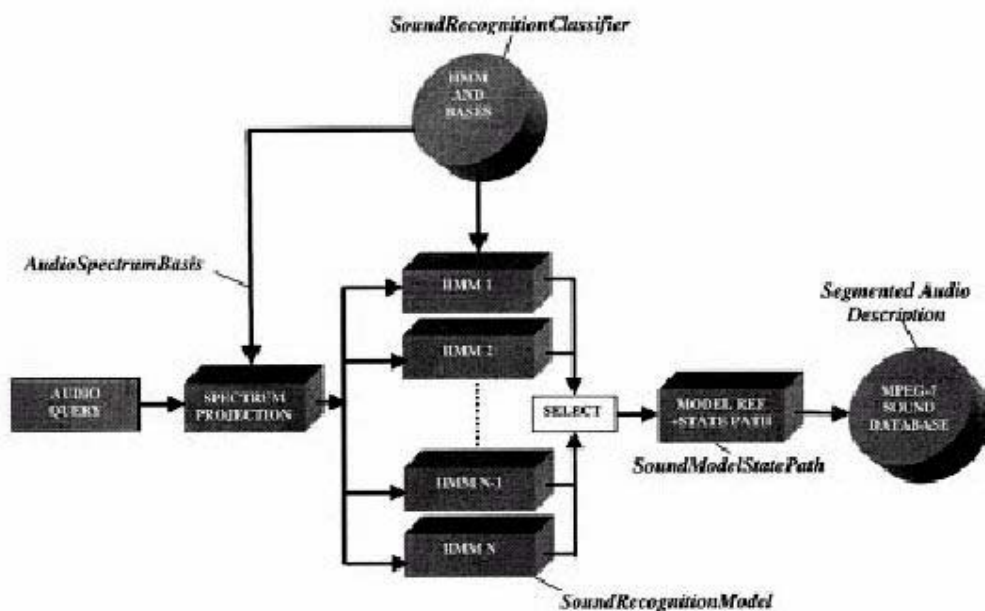
Vienādojums (1) atbilst pazīmes izdabūšanai un (2) atbilst spektra pārveidošanai, kur  $\mathbf{V}^+$  norāda viltoto apgriezto  $\mathbf{V}$  neortogonālam gadījumam.

3) *Modeļa Iegūšana*: Nekontrolētais klasterizācijas algoritms izmantojās lai sadalītu  $n$ - dimensiju pazīmju telpu, apdzīvota ar samazinātas dimensijas novērošanas vektoriem,  $k$  stāvokļos. Algoritms noteic stāvokļu optimālo skaitli dotiem datiem ar pārejas matricas svītrosanu dodot sākuma minēšanas  $k$ , parādītie eksperimenti kuri ir starp pieciem un desmitiem stāvokļiem ir pietiekami labai klasifikatoru darbībai. Trenēšana izmanto labi zināma Baum- Welch algoritma variantu kurš ir paplašināts ar entropijas prioru un gaidīšanas maksimizēšanas (GM) algoritma determinēšanas atkvēlināšanas izpildīšanu. Vienu reizi trenēti, katrs skaņas atpazīšanas HMM saglabājas diskā, kopā ar to pamata funkciju, DDL formātā izmantojot SkaņasAtpazīšanasModeļa aprakstīšanas shēmu (sk. zīm.9.)



zīm.9. Slēpta Markova modeļa un pamata funkciju izdabūšana un to glabāšana DLL

Kad skaņu modeļi bija trenēti, atbilstoši kategorijas pilnai taksonomijai, HMM ir savākti kopā lielākā SkaņasAtpazīšanasKlasifikatora datu struktūrā, tādā veidā ģenerējot modeļu ontoloģiju, kā ir parādīts zīm.7.



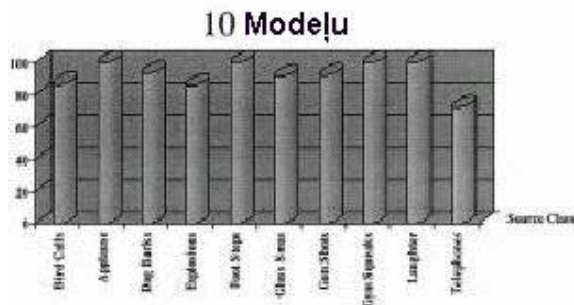
zīm.10. Skaņas indeksu izdabūšana izmantojot skaņas atpazīšanas klasifikatoru.

#### IV. Skaņas Aprakstīšana

Ontoloģija izmantojas jaunu skaņu indeksēšanai ar kvalitatīviem un kvantitatīviem aprakstītājiem. Zīm. 10. ilustrē automātisko izdabūšanas sistēmu skaņas indeksēšanai datu bāzē izmantojot pirms trenētu klasifikatoru saglabātu kā DDL fails. Skaņas ir nolasītas no mediju formāta avota, tāda kā WAV fails, un projektēti uz katram modelim atsevišķu klasifikācijas telpā. Pēc tam viņi iet caur Viterbi dekodētāju kurš dod ka labāk derošo modeli, tā arī stāvoļa ceļu caur modeli skaņas kursa laikā. Katra skaņa pēc tam indeksēti pēc viņu kategorijām, modeļa norādījuma, un modeļa stāvokļa ceļa, un aprakstītāji uzrakstīti datu bāzē DDL formātā. Indeksēta datu bāze ir gatava šajā stadijā pieprasījuma un meklēšanas programmām izmantojot jebkuru no saglabātiem aprakstītājiem.

Zīm.11. rada klasifikācijas veikšanas procesu desmit skaņas klasēm izmantojot izdabūšanas shēmu, aprakstītu augstāk. Veikšana bija mērogota ar zemes precizitāti izmantojot avota skaņas zīmi kā parādīts profesionālajā skaņas

efektu bibliotēkā. Rezultāti parādīti jaunus testa datus, nav izmantoti klasifikatoru trenēšanā, un tādēļ demonstrē klasifikatora vispārīguma potenciālās iespējas.



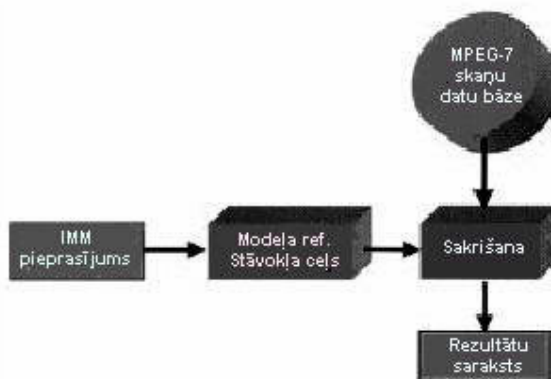
zīm.11. Skaņas efektu klasifikatora darbošanas piemērs ar 10 skaņas klasēm.

## V. Piemēra Meklēšanas Programmas.

Sekojošas sekcijas dod piemērus kā var izmantot aprakstītāju shēmas meklēšanai izmantojot abas DDL- bāzētus pieprasījumus un mediju avota formāta pieprasījumus.

### A. Pieprasījumi pēc piemēra I.

Pirmajā meklēšanas piemērī, skaņas pieprasījums pasniegts sistēmai izmantojot SkaņasModeļaStāvokļaCeļš aprakstītāju shēmu DDL formātā. Sistēma lasa pieprasījumu un apdzīvo iekšējās datu struktūras ar aprakstīšanas informāciju. Šī aprakstīšana salīdzinās ar aprakstītajiem ņemtiem no skaņu datu bāzes glabātiem diskā, sakrišanas algoritms iedarbojās, un atlasīto rezultātu saraksts atgriežas atpakaļ (sk. zīm.12.).

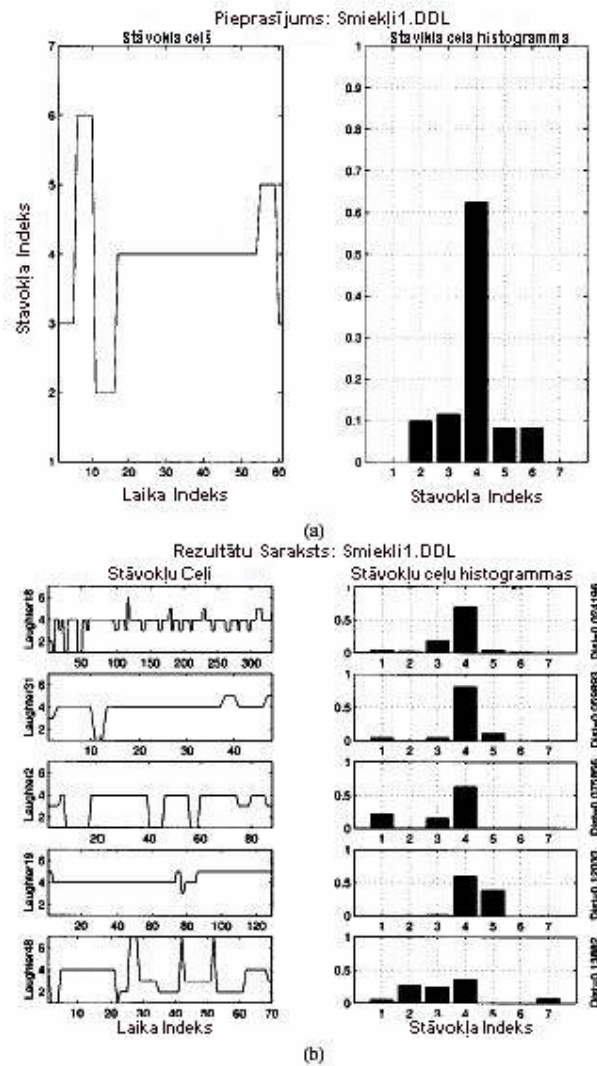


zīm.12. Piemēra meklēšanas programma izmantojot DDL formātu.

Stāvokļa ceļa sakrišanas algoritms ir nenormatīvs; tomēr, parastais piemēra sakrišanas algoritms ir kļūdu starp stāvokļu ceļu histogrammām kvadrāta summa. Šis sakrišanas algoritms pieprasa maz izskaitļošanu un var būt izskaitļots tieši no glabāta stāvokļa ceļa aprakstītāja.

Stāvokļa ceļa histogrammas ir izskaitļotas kā summārais laika garums, kuru tērē skaņa katrā stāvoklī dalīta ar summāro skaņas ilgumu, tādā veidā dodot diskrētu varbūtības blīvuma funkciju ar stāvokļa indeksu kā nejaušu lielumu. Kļūdu kvadrātu summa (KKS) starp pieprasījuma skaņas histogrammu un katras skaņas datu bāzē ir izskaitļota un izmantojās kā metriskā distance. Nulles distance nozīmē identisku sakrišanu un paaugstinātas nulles distances ir vairāk nelīdzīgie sakritumi. Šī metriskā distance izmantojās skaņu klasificēšanai datu bāzē līdzīguma secībā, pēc tam vajadzīgais sakrišanu numurs ir izdots. Zīm.13(a) rada stāvokļa ceļu un stāvokļa ceļa histogrammu Smiešanas skaņas pieprasījumam un zīm. 13(b) rada stāvokļa ceļus un histogrammas piecām labākām sakrišanām; visas sakrišanas ir no vienas klases tapāt kā pieprasījums, kurš ilustrē veiksmīgu klasifikācijas sistēmas izpildīšanu.

Lai turēt ontoloģijas struktūru, skaņas ekvivalentos vai ierobežoto kategoriju robežās, ka noteikts ar taksonomiju, ir atgriezti kā sakritumi. Tādā veidā “Suņi” kategorija atgriezīs skaņas piederošus visām kategorijām saistītiem ar “Suņi” tasonomijā.



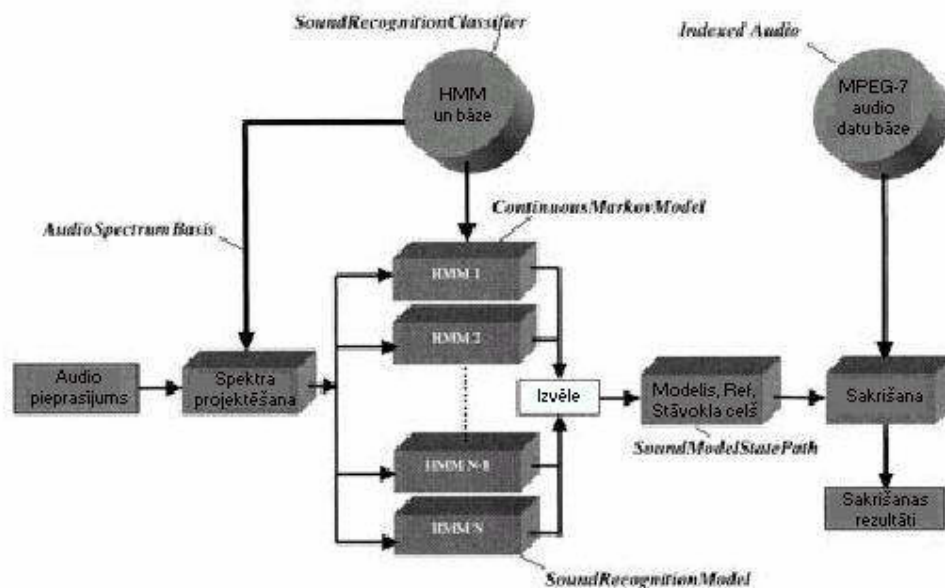
zīm.13. (a) Smiekla skaņas pieprasījums radot stāvokļa ceļa reprezentāciju un stāvokļa ceļa histogrammu; (b)

Pieci labākie sakritumi



## B. Pieprasījums pēc piemēra II

Sistēma parādīta zīm.14. Īsteno vairāk sarežģīto pieprasījuma pēc piemēra programmas versiju izmantojot audio pieprasījumu DDL aprakstītāja bāzēta pieprasījuma vietā. Šajā gadījumā, audio pazīmju izdabūšanas procesam jābūt izdarītam, proti spektrogrammu un apvalku izskaitļošana seko līdzās projekcijai pretēji glabāta pamata funkciju komplektu katram modelim klasifikatora. Rezultāta pazīmes ar samazinātām dimensijām padoti Viterby dekodētājā dotajām klasifikatoram un HMM ar maksimālas iespējas rēķinu dotajam pazīmēm ir izvēlēts. Viterby dekodētājs īstēnībā funkcijas ka modeļa sakrišanas algoritms klasifikatoru shēmai. Modeļa raksturojums un stāvokļu ceļš ir pierakstīts un sakrītošie rezultāti ar pirms izskaitļoto datu bāzi, ka pirmajā piemērā.



zīm.14. Pieprasījuma pēc piemēra programma ar pieprasījumu mediju formā.

## VI. Nobeigums

Fināla piemērs ilustrē ka ir nepieciešami glābāt pamata funkcijas un HMM parametrus attiecīgi kategorijai, modeļa raksturojumiem un stāvokļa ceļa aprakstītajiem skaņu efektu datu bāzei. SkaņšAtpazīšanasKlasifikatora shēmas funkcijas galvenais fails kurš pieprasīts lai izdabūšanas process veidotu aprakstītājus, kuri ir savienojami ar aprakstītajiem, kuri glabājas asociētā datu bāzē. Galvenie dati dod iespēju klasifikācijai un indeksēšanai savienojamiem ar dažādiem izdabūšanas īstenošanām un tādēļ neprasa standartizāciju specifiskiem klasifikatora tiem.

Aprakstīšanas shēmas un izdabūšanas metodoloģijas aprakstītas šajā papīrā savienojamu struktūru skaņu indeksēšanai un pieprasījumam no plaša ranga dažādu klašu. Dažas MDS bija aprakstītas papildinājumā audio specifiskiem aprakstītajiem. Kategoriju nosaukuma un klasifikatoru modeļu noteikšanas metodes bija aprakstītas ka arī piemēra programmas ar specifiskam aprakstītajam sakrišanas algoritma detaļas.

Instrumenti, aprakstīti šajā papīrā rada stingru karkasu pamata skaņu klasifikācijai, kura var adaptēties plaša ranga programmām, kuriem ir audio komponente.

## MPEG-7 Multimediju Aprakstīšanas Shēmas.

*Abstrakti-* multimediju MPEG-7 aprakstīšanas shēmas ir meta datu struktūras kas apraksta audio- vizuālo saturu(AV). Aprakstīšanas shēmas(AS) nodrošina apraksta standartu pēc XML kas attiecās uz AV aprakstu un tā satura vadību lai atvieglotu meklējumu, indeksāciju filtrāciju un pieeju. DS tiek aprakstītas MPEG-7 aprakstīšanas shēmu valodā, kura ir veidota uz XML valodas shēmas un ir aprakstītas(ilustrētas) dokumentos vai straumēs. Rezultāta apraksts var būt teksta formā (tas nozīmē cilvēka lasīšanai XML valodā redakcijai, meklēšanai un filtrācijai) vai saspiegtā binārā formā (glabāšanai vai pārraidei). Šajā dokumentā, mēs sniedzam un aprakstam MPEG-7 MDS aprakstām to darbību un izmantošanu multimedijā.

*Indekss –* Pāršķiršana, satura labošana, ciparu bibliotēkas, indeksācija, meta dati, MPEG-7, multimedija datu bāzes, multimediju Aprakstīšanas shēmas(MDS), labošana, meklēšana, XML.

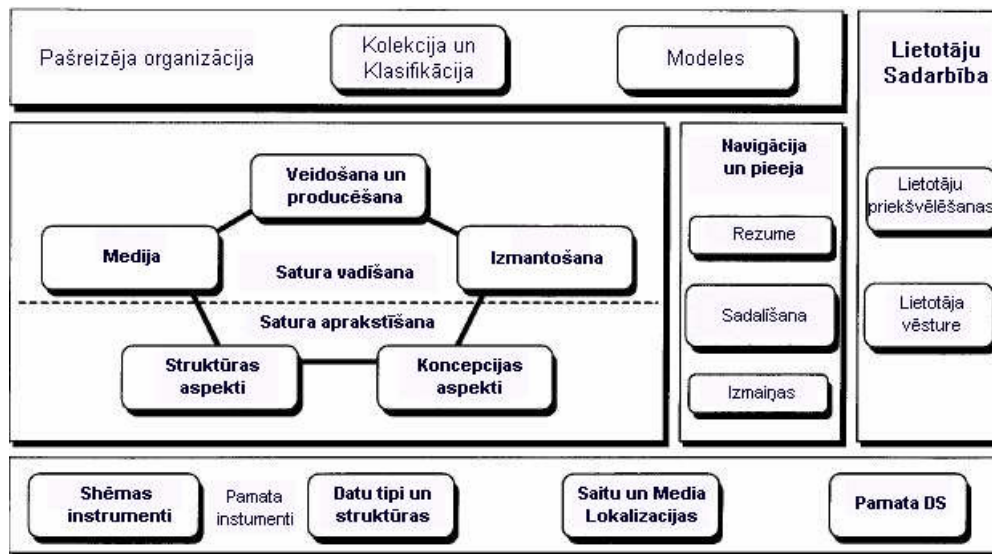
### I. Ievads

MPEG-7 standarta mērķis ir nodrošināt iekšējo meklēšanu, indeksāciju, filtrēšanu un pieeju pie audio- vizuālo saturu(AV) kad ir nodrošināta iekšējā darbība starp mehānismiem un programmām kas nodarbojas ar AV saturu aprakstu. MPEG-7 apraksta AV saturu specifiku, kā arī informāciju par AV saturu vadību. MPEG-7 apraksts var būt divās formas: 1) XML teksta forma kas var būt izmantota redakcijai, meklēšanai un filtrēšanai, un 2) bināra forma kas ir izmantojama glabāšanai, pārraidei un straumju piegādei. Standarts nosaka četrus normatīvo elementu veidus: Aprakstītāji, Aprakstīšanas shēmas(DS), Aprakstīšanas shēmu valoda(ASV) un kodēšanas shēmas.

MPEG-7 Aprakstītāji ir viss pirms veidoti vai aprakstītie zema līmeņa audio un vizuālās īpašības – krāsa, tekstura, kustība, audio enerģija u.t.t., kā arī AV satura sastāvdaļas – vieta, laiks kvalitāte, u.t.t. ir gaidāms ka daudzi zemā līmeņa Aprakstītāji būs automātiski izvilkti ar programmas palīdzību. No otras puses, MPEG-7 DS ir veidotas viss pirms lai aprakstītu AV augsta akas līmeņus, tādus kā reģionus, segmentus, objektus, notikumus, un citi nomainīgi meta dati kura attiecās uz radīšanu, produkciju, izmantošanu u.t.t. DS izveido vairāk kompleksu aprakstu kad integrēts kopā ar daudzu veidu Aprakstiem un DS kā arī caur sakariem starp aprakstāmām sastāvdaļām. MPEG-7 standartā, MDS ir kategorijā kas pieder multimedijam, audio un vizuālai sfērai. MDS apraksta saturu kas ietver audio vai vizuālās DS attiecās specialī īpašībām piemērotām audio vai vizuālai sfērai attiecīgi. Dažos gadījumos automātiskie instrumenti var būt izmantoti ilustrēt piemērus DS, bet daudzos gadījumos piemēru ilustrēšanai DS ir nepieciešama cilvēka palīdzība DS instrumentu izvilkšanai vai paskaidrošana.

MPEG-7 DDL tā ir valoda kas nosaka DS un Aprakstītāju sintaksisku. DDL pieļauj ka MPEG-7 standarta DS un Apraksti var būt izmantoti specifiskas gadījumos DDL bāzējas uz W3C XML Valodas Shēmas. MPEG-7 kādu shēmas ražo bināro ražošanas formu kas ir kompakta, viegli pārraidāma, un nepieļauj kļūdu transmisijas laikā.

Šis dokumenta mērķis ir parādīt MPEG-7 MDS. Zīm.1. uzrāda MDS organizāciju dažādās funkcionālās sfērās: pamatelementi, saturs vadība, saturs apraksts, navigācija un pieeja, saturs organizācija un lietotāja darbība. MPEG-7 DS skeneris ir izskatāms ka aprakstīšanas instrumentu bibliotēka, un praktiski programma izvēlās attiecīgu nodaļu attiecīgajā DS. Šis dokuments apskata katru no dažādam MDS funkcijām. Dotajā laikā, detalizētāka informācija var būt atrasta MPEG-7 Eksperimentālajā modelē un MPEG-7 komitejas melnraksts(CD) dokumentos kuru izstrādāja MPEG MDS grupa. Papildinājumā daudzi no attiecīgām koncepcijām multimedija sfērās ir aprakstīti MPEG-7 konceptuālajā modelī, kas pievienots kā papildinājums MPEG-7 Prasību dokumentiem. Dokuments ir sekojoši organizēts. Sekcija2 apraksta MDS Pamat Elementus, Sekcija3 apraksta DS kā saturs vadību, Sekcija 4 apraksta DS saturs aprakstu. Sekcija 5 apraksta DS navigāciju un pieeju, Sekcija 6 apraksta organizācijas saturu. Un viss beidzot, Sekcija 7 apraksta DS lietotājam.



zīm.1. MPEG-7 MDS pārskats.

## II. Pamatelementi

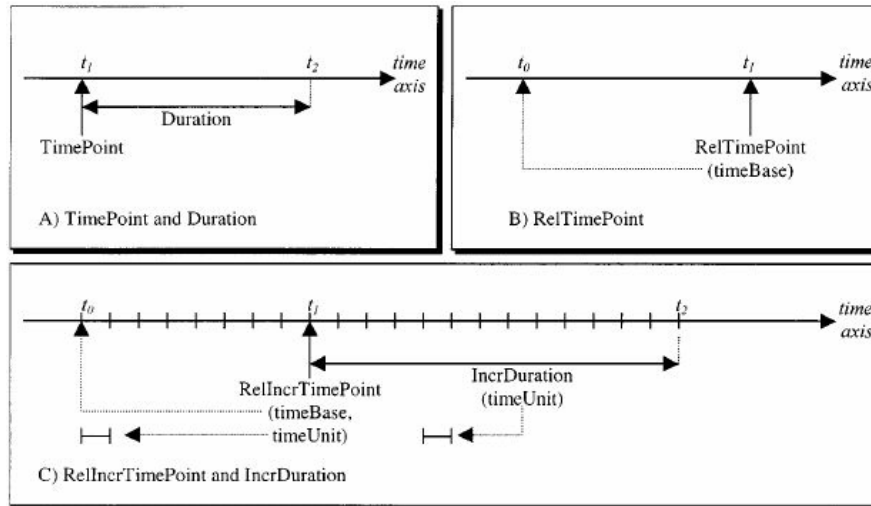
MPEG-7 nodrošina veselu rindu Shēmas Instrumentu kas palīdz formācijā, pakešu sastādīšana un MPEG-7 aprakstu anotācijas. MPEG-7 apraksts sākās ar pamatelementu kas nosaka vai apraksts ir pilnīgs var daļējs. Pilns apraksts sniedz atsevišķo AV Saturs aprakstu programmai. No otras puses, aprakstīšanas bloks saturs tikai daļēju vai papildinformāciju kas papildina esošo aprakstu. Pilnīgā apraksta gadījumā, MPEG-7 augstākā līmeņa elements seko pamatelementam. Augstākā līmeņa nosaka aprakstu kas ir saistīts ar specifisko apraksta uzdevumu, piemēram apraksts par atsevišķā AV saturs veidu, attēls, video, audio vai multimedija, vai atsevišķa funkcija var attiecās uz saturs vadību – radīšana, izmantošana, rezumēšana u.t.t. austākā līmeņa veidi savāc kopā nepieciešamos instrumentus lai veiktu specifisko apraksta uzdevumu. Kad tiek izmantots apraksta bloks, pēc pamatelementa var nākt MPEG-7DS brīvai piemērs vai Aprakstītājs. Atšķirība no pilnā apraksta kas panākti ietver “Semantiski pilnie” MPEG-7 aprakstu, aprakstīšanas bloks var sūfīt daļēju aprakstu ka tas ir pieprasīts programmā – vietas apraksts, kontūru faktūras apraksts u.t.t. DS pakete apraksta lietotāja noteikto MPEG-7DS un DS organizāciju pakete, kas atļauj MPEG-7 organizētai

instrumentu izvēlei saistīties ar meklējuma mehānismu vai lietotāju. Papildus, AprakstīšanasMetadati DS apraksta metadatus par aprakstu, t.i. radīšanas laiks, instrumentu izvēle, versija, konfidencialitāti u.t.t.

MDS specifikācijā tiek izmantota vesela rinda pamat elementu kas būvē MPEG-7DS bāzi.

Pamatdatu veidu nosaka paplašinātas datu veidus un matemātiskās struktūras, tādus ka vektori un matricas, kas ir nepieciešami DS lai aprakstītu AV saturu. Pamatelementi iekļauj arī konstrukcijas savienošanas mēdiju failiem, nosaka apraksta punkta atrašanās vietu, apraksta laiku, vietas, personas, indivīdus, grupas, organizācijas un citu teksta anotāciju. Mēs īsumā ieskatīsim MPEG-7 pieejas lai aprakstītu laiku un teksta anotācijas.

Informācija: DS laika aprakstīšanai pamatojas uz ISO 8601 standartiem, kas arī bija adaptēts XML Shēmas valodai. Laika DS un Mēdija Laika DS apraksta laika informāciju reālajā pasaulē un attiecīgi mēdija sakaros. Abas seko tai pašai stratēģijai aprakstītai Zīm.2. Zīm2(a) ilustrē vienkāršāko ceļu kā aprakstīt laika mirkli un laika intervālu. Laika mirklis var būt aprakstīts leksiski izmantojot Laika Punktu. Intervāls var būt aprakstīts ar tā sākuma punktu (izmantojot Laika Punktu) un Ilgumu. Alternatīvi kā var aprakstīt laika mirkli ir uzrādīts Zīm.2(b). Tas ir pamatots uz Attiecīga Laika Punktu. Mirklis ir aprakstīts laika atziņojumā attiecīgi uz pieprasījumu, satur Laika Bāze. Jāņem vērā ka Relatīvā Laika Punkta mērķis ir atpazīt laika mirkli un intervālu kā Ilgumu Zīm.2(a). Zīm.2(c) ilustrē laika specifikāciju izmantojot noteiktu intervālu kuru sauc par Laika Bloku un skaita intervālu daudzumu. Šī specifikācija ir īpaši derīga periodiskiem vai piemēru laika signāliem. Tādēļ ka stratēģija sastāv no Laika Bloka skaita laika mirkļa specifikācijai ir jābūt relatīvai Laika Bāzei (vai laika pamata sākumam). Zīm.2.(c) ir parādīts Attiecīgais Papildlaika punkts atskaitot 8 Laika blokus (sākuma forma). Intervāls var būt arī noteikts ar Laika Blokus skaitu. Zīm2(c) Papild ilgums ir izmantots lai saskaitītu 13 Laika Blokus lai noteiktu intervālu.



zīm.2. Laika DS apskats

Tekstuālā Anotācija: Teksta anotācija ir svarīga DS sastāvdaļa. MPEG-7 nodrošina daudzus dažādus pamata konstrukciju priekš tekstuālās anotācijas. Viss brīvākā teksta anotācijas uzbūve ir datu veida kas izmantojas brīvi tekstā. Brīvais teksts atļauj patvaļīgas teksta sfīgas uzbūvi, kas dažreiz iekļauj informāciju par teksta valodu. MPEG-7 ir arī instruments priekš daudz struktūras teksta anotācijai kad iekļauj specifiskas sfēras kuras atbild uz jautājumiem: Kas? Kāds objekts? Kāda darbība? Kur? Kad? Kāpēc? Un Kā? Saradītās tekstuālās anotācijas var būt noteiktas aprakstot noteikto sintaksisko atkarību starp gramatiskiem elementiem kas tiek sastādīts teikums (t.i. attiecība starp darbības vardu un lietvārdu u.t.t.) šis tekstuālās anotācijas veids ir īpaši piemērots anotācijai kas tiek automātiski apstrādāta. Un pēdējais, MPEG-7 nodrošina klasifikācijas shēmu un kontrolējamo termiņu bloku kas izveido vārdnīcu speciālai

programmai vai sfērai. Kontrolējamie termini tiek izmantoti aprakstos kas atpazīst ievadītos datus klasifikācijas shēmās. Kad kontrolējamie termiņi ir aprakstīti ar klasifikācijas shēmu palīdzību ir dota priekšroka standartizēt vārdnīcas dažādām programmām un sfērām. Kad tas ir iespējams ka vārdnīcas multimediju programmās tiks palielinātas ar laiku.

### III. Satura vadība

MPEG-7 nodrošina DS priekš AV satura vadības. Šie instrumenti apraksta sekojošo informāciju: 1) radīšana un ražošana; 2) mēdiņu kodēšana, uzglabāšana un failu formatēšana; 3) satura izmantošana. Vairāk paskaidrojumu par MPEG-7 instrumentiem satura vadīšanai ir sekojoši.

- Radoša Informācija apraksta AV satura un cietu attiecīgo materiālu radīšanu un klasifikāciju. Radošā Informācija nosaka nosaukumu (kas pats par sevi var būt tekstuāls vai cita AV satura daļa) tekstuālā anotācija, informācija tāda ka radītāji, radīšanai atrašanas vieta un datumi. Klasifikācija informācija apraksta kā AV materiāls ir klasificēts kategorijās. Žanrs, tēma, mērķis, valoda u.t.t. tas nodrošina arī pastatu un norādošo informāciju tādi ka vecuma klasifikācija, norādījumi vecākiem, subjektīvais pārskats. Beidzot, attiecīgā materiāla informācija apraksta vai pastāv citi AV materiāli kas attiecas uz saturu kas tika aprakstīts.

- Mēdiņa Informācija apraksta uzglabāšanu formātu, kompresiju un AV satura kodēšanu. Mēdiņa Informācijas DS nosaka galveno mēdiņu kas ir pamata avots no kura rodas dažādi AV satura mirkļi. AV satura mirkļi attiecas uz mēdiņa profilu, kuri ir master versijas iegādātas, iespējami, no citas kodēšanas, vai glabāšanas vai pārraides formātiem. Katrs mēdiņu profils ir individuāli aprakstīts iekodēšanas parametros, mēdiņa uzglabāšanas informācijā un savā atrašanas vietā.

- Lietošanas Informācija apraksta pielietošanas AV satura informāciju, tādu ka izmantošanas tiesības, izmantošanas pieraksts, un finansiālo informāciju. MPEG-7 aprakstā informāciju par tiesībām nav īpaši iekļauta; tā vietā ir pieeja datiem par īpašnieku tiesībām un cita informācija kas attiecas uz tiesību vadību un aizsardzību. Tiesību DS ir uzrādīti unikālā identifikācijas formā kas tiek vadīta ar ārpus organizāciju. Galvenā stratēģija ir nodrošināt pieeju īpašnieka esošajām tiesībām bez tiešām pārrunām un informācijas pa tiešo. Lietotāja ieraksta DS un pieejamības DS nodrošina informāciju kas attiecas uz satura izmantošanu: pārraides, pasūtījumu piegādei, CD pārdošanā, u.t.t. Beidzot, finansiālai DS piegādē informāciju par produkcijas cenu un ienākumu, par statusa izmantošanu. Izmantojama informācijā ir raksturīga dinamiska un var būt mainīta savā AV satura eksistēšanas laikā.

### IV. Satura Apraksts

MPEG-7 nodrošina DS struktūras un AV satura semantikas aprakstu. Struktūras instrumenti apraksta AV saturu, tā video segmentus, rāmjus, kustošos un nekustošos posmus un audio segmentus. Semantiskie instrumenti apraksta objektus, notikumus un reālas pasaules uzskatus, kuri ir ierakstīti AV saturā.

#### A. Strukturāla Aspekta Satura Apraksts

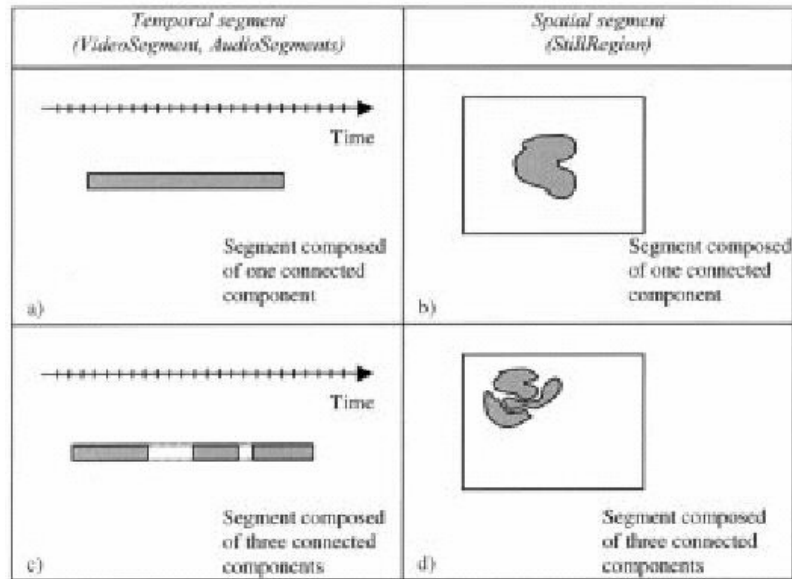
Segmentu DS apraksta AV satura sadalīšanu telpā, laikā, telpā- laikā. Segmentu DS var parakstīt AV satura atkārtotu vai hierarhisko dekompozīciju segmentos kas veido segmentu koku. Segmentu Attiecību DS apraksta papildus telpas- laika attiecības starp segmentiem. Segmentu DS veido dažādu specializēto segmentu veidu abstrakto bāzes veidu: audio segments, video segments, AV segments, kustošie posmi un nekustošie posmi. Kā rezultātā, segmentam var būt telpu un/vai laika īpašības. Piemērām Audio Segmentu DS var aprakstīt laika audio segmentu kas attiecas uz audio sekvenses laika periodu. Video Segmentu DS var aprakstīt video sekvenses rāmju bloku. Audio vizuālai

Segmentu DS var aprakstīt audio un vizuālās informācijās kombināciju, tādu kā video sinhronizētu ar audio. Nekustāmo Posmu DS var aprakstīt attēla telpas segmentu un posmu vai video rāmjus. Beidzot, Kustošā Posma DS var aprakstīt video sekvences telpas- laika segmentu vai kustošo posmu.

Ir arī specializēto segmentu rinda priekš AV satura specifiskiem veidiem. Piemēram, Mazaikas DS ir Nekustoša Posma specialais veids. Tas apraksta mozaiku vai video segmentu panorāmu, būvētu liekot rindā un deformējot Video Segmenta rāmjus izmantotos priekš katra kopēju attiecīgu sistēmu. Video Teksta DS aprakta video satura posmu kas attiecas uz tekstu vai nosaukumu. Tur liek iekļautu dubultsalikto tekstu, kā arī tekstu kas parādās darbībā. Tintes Segmentu DZ apraksta elektroniskās tintes dokumenta segmentu, kas ir radīts “pildspalvas” sistēmā vai arī elektroniskā “baltā ekrānā”.

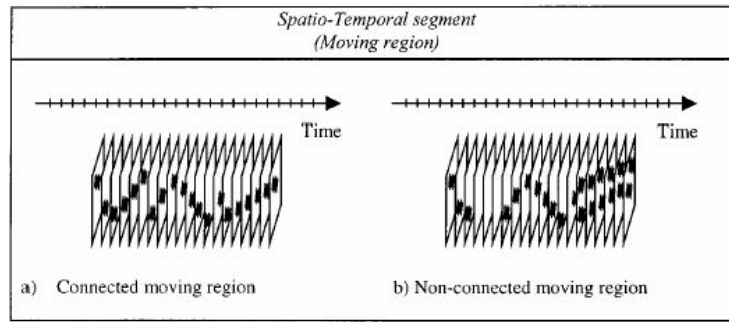
Tādēļ ka Segmentu DS ir abstrakts viņš nevar būt aprakstīts ar piemēriem. Toties, Segmentu DS satur kopēju ar dažādiem segmentu veidiem. Starp vispārējām segmenta īpašībām ir informācija kas attiecas un radošo pusi , izmantošanu, mēdiju atrašanas vietu un teksta anotāciju. Segmentu DS var būt izmantots lai aprakstītu Segmentus, kuri nav noteikti savienoti, bet salikti no dažādiem nesavienotiem komponentiem. Savienotās daļas šeit attiecas uz abām telpas un laika sfērām. Laika segments (video segments, audio segments un audio- vizuālais segments) ir savienoti uz laiku, ja tā ir nepārtraukta video rāmju vai audio piemēru secība. Telpas segments (nekustošs posms) ir daļēji savienots, ja tā ir savienoto attēlu grupa. Telpas un Laika Segments (Kustošs Posms) ir telpiski vai laika savienots, ja laika segments (kur tas ir aprakstīts ar piemēriem) ir uz laiku pievienots un katrs piemēru posms rāmjos ir savienots telpā (Tas nav 3-D klasiskais savienojums).

*Zīm.3. ilustrē dažus laika kropļojuma vai telpu segmentu piemērus un viņu sakaru nodrošināšanu. Zīm.3. (a) un (b) ilustrē laika kropļojumu un telpa segmentu, sastādītu tikai no vienas savienotas komponentes. Zīm.3. (c) un (d) ilustrē laika un telpas segmentu sastādītu no trīs savienotiem komponentēm. Zīm.4. rada savienoto un nesavienoto kustošo reģionu piemērus. Pēdējā gadījumā, segments nav savienots jo tas nav ilustrēts visos freimos un, bez tam, tas iekļauj vairākus telpiski savienotus komponentes dažos freimos.*



*zīm.3. Segmentu piemēri. (a), (b) Segmenti, sastādīti no vienas savienotās komponentes. (c), (d) Segmenti, sastādīti no trim savienotām komponentēm.*

Pievērsiet uzmanību, ka visos gadījumos, Aprakstītāji un AS pievienoti segmentam ir globāli savienotu komponentu, būvējošo segmentu, savienojumam. Šajā līmenī, nav iespējami aprakstīt individuāli segmenta savienotus komponentes. Ja savienotus komponentus vajag aprakstīt individuāli, pēc tam segmentam jābūt saspīestam vairākos apakš komponentēs atbilstoši viņu individuāli savienotiem komponentēm.

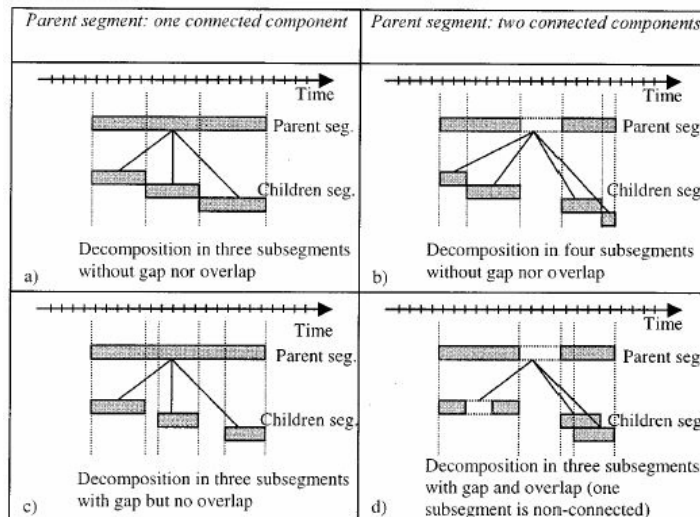


zīm.4. (a) savienotu un (b) nesavienota kustīga reģiona piemērs.

Segmenta DS ir rekursīvs, t.i., viņu var sadalīt apakš segmentos, un tādā veidā var formēt hierarhiju (koku). Rezultāta segmenta koks izmantojās lai aprakstītu mediju avotu, laika un/vai telpas AV satura struktūru. Piemēram, video programma var būt laicīgi segmentēta vairākos scēnu, kadru, un mikro segmentu līmeņos; satura tabula tāpēc var būt ģenerēta pamatojoties uz šo struktūru. Līdzīgas stratēģijas var būt izmantotas telpu un telpu- laicīgu segmentu.

Segments var arī būt sadalīts vairākos mediju avotos, tādos kā audio celiņos vai uzskatos no vairākām kamerām. Hierarhiskā sadalīšana ir lietderīga efektīviem meklēšanas stratēģijām. Tas arī ļauj aprakstīšanai būt par mērogojamu: segments var būt aprakstīti ar viņu pašu Aprakstītāju un AS shēmām, bet tie var arī būt aprakstīti ar Aprakstītāju un AS vienību, kura ir saistīta ar viņas apakš segmentiem. Pievērsiet uzmanību tam, ka segments var būt sadalīts dažādu tipu apakš segmentos, piemēram, video segments var būt sadalīts kustīgos reģionos, kuri ir paši sadalīti tajos pašos reģionos.

Kad tas ir izdarīts telpu- laika vietā, sadalīšana ir aprakstīta ar apakš daļiņas tipa definējošo atribūtu kopu: laika, telpas un telpa- laika. Turklāt, telpu un laiku apakš divīzijās var atstāt atstarpes un daļējas sakrišanas starp apakš segmentiem. Daži sadalīšanas piemēri ir aprakstīti laika segmentiem zīm.5. Zīm.5. (a) un (b) apraksta divus sadales piemērus bez atstarpēm un daļējas sakrišanas. Abos gadījumos bērnu vienība atbilst tieši vacāku laika paplašinājumam, pat ja vecākais nav savienots (sk. piemēru zīm.5. (b)). Zīm.5. (c) rada sadales piemēru ar atstarpēm, bet bez daļējām sakrišanām. Beigās, zīm.5. (d) ilustrē vairāk sarežģīto lietu, kad vecākais ir sastādīts no divām savienotām komponentēm un tas sadale rada trīs bērnus: pirmais ir sastādīts no divām savienotām komponentēm, tajā laikā kad divi atlikušie bērni sastādīti no vienas savienotas komponentes. Sadalīšana ļauj atstarpi un pārklāšanu. Pievērsiet uzmanību, ka jebkurā gadījumā, sadale nosaka, ka telpu- laika vieta, noteikta ar segmentiem, bērnu segmenti ir iekļauti telpu- laika vietā definētā ar viņu izejas segmentu.



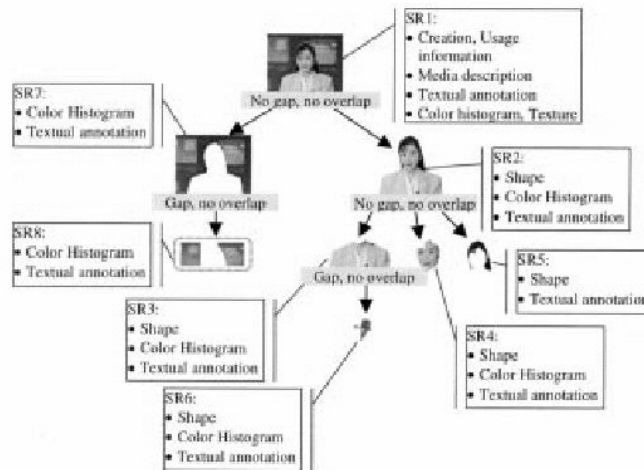
zīm.5. Segmenta dekompozīcijas piemērs. (a), (b) segmenta dekompozīcija bez spraugas un noseģšanas (c), (d) segmenta dekompozīcija ar noseģšanu vai spraugu.

Kā aprakstīts augstāk, segments var būt aprakstīts ar veidošanas informāciju, izmantošanas informāciju, mēdiju informāciju un tekstuālo informāciju. Tomēr, specifiskas īpašības atkarīgas no segmenta tipa ir arī atļauti. Šīs specifiskas īpašības ir parādītas tab.1. Lielākā daļa no Aprakstītāju, atbilstošu šiem īpašībām, var būt automātiski izdabūti no oriģinālā satura. Šiem nolūkiem daudz instrumentu bija aprakstīts literatūrā. Sadalīšanas ilustrēšana iekļauta DS segmentā var būt parādīta kā hierarhiskās segmentācijas problēma kur elementāriem objektiem jābūt definētiem un strukturētiem iekļaušanas attiecībām koka ietvaros.

Tab.1. Segmenta Aprakstīšanas Specialās pazīmes.

Feature	Video Segment	Still Region	Moving Region	Audio Segment
Time	X	.	X	X
Shape	.	X	X	.
Color	X	X	X	.
Texture	.	X	.	.
Motion	X	.	X	.
Camera motion	X	.	.	.
Audio features	.	.	X	X

Attēla aprakstīšanas piemērs ir parādīts zīm.6. Oriģinālais attēls ir aprakstīts kā Still Region SR1, kurš ir aprakstīts ar veidošanu (virsraksts, veidotājs), izmantošanas informāciju (kopēšanas tiesības) un medija informāciju (faila formāts), kā arī ar tekstuālo informāciju (attēla satura summēšana), krāsu histogramma un faktūra aprakstītāju. Šī sākuma reģions var būt tālāk sadalīts individuālajos reģionos. Katram dalīšanas solim mēs norādām ja Atstarpes un daļējas sakrišanas ir pieļauti. Segmentu koks ir sastādīts no astoņiem kadru reģioniem. Katram reģionam zīm.6. rada īpašības tipu, kura ir ilustrēta. Pievērsiet uzmanību, ka tas nav obligāti atkārtot koka hierarhijā veidošanu, izmantošanas informāciju un mediju informāciju, jo bērnu segmenti saglāba vecāku vērtības.

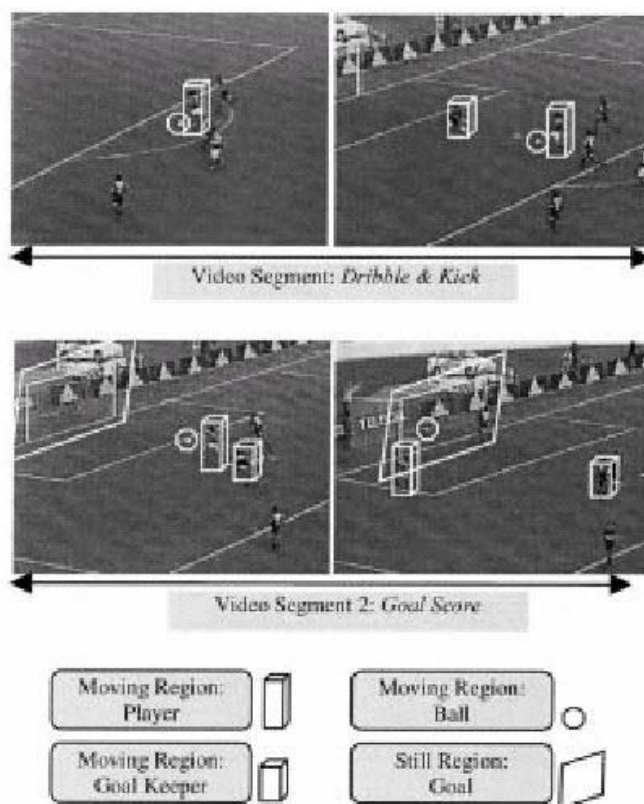


zīm.6. Attēla aprakstīšanas ar kadru reģioniem piemērs.

Satura struktūras aprakstīšana neierobežota paļaušanas kokam. Tomēr, hierarhiska struktūras tādas kā koki ir adekvāti efektīvai pieejai, izdabūšanai un mērogojamai aprakstīšanai, viņi nozīmē ierobežojumus, kurus var to izdarīt neatbilstošiem dotajai programmai. Šajos gadījumos vajag izmantot SegmentuAttiecību DS. Grafa struktūra ir ļoti viegli

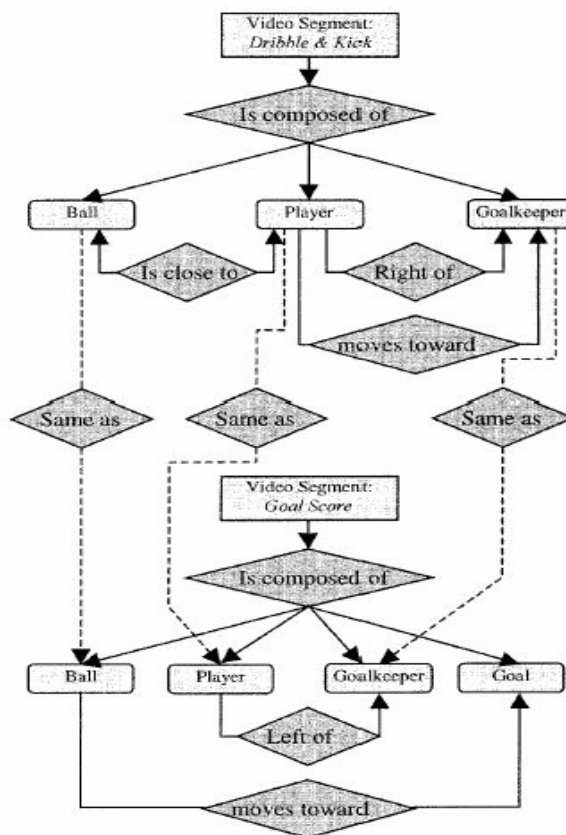


aprakstīta ar mezglu kopu, katrs attiecībā segmentam, un apmales kopu, katrs attiecībā pret divu mezglu attiecībām. Lai ilustrētu grafu lietderīgumu, apskatīsim piemēru, parādītu zīm.7.



zīm.7. Video segmentu un reģionu piemērs Segmentu Attiecību Grafam zīm.8.

Šis piemērs rada kadrus no futbola mača. Divi video segmenti, viens Still Region un trīs Kustošie Reģioni ir izskatīti. Iespējams grafs aprakstošais satura struktūru ir parādīts zīm.8. Video segments Dribble& Kick iesaista bumbu, vārtsargu un spēlētāju. Bumba palika tuvu spēlētājam kurš kustās uz vārtsargu. Spēlētājs uzstājas vārtsargam pa labi. Vārtu rēķina video segments iesaista tos pašus kustošus reģionus plus kadra reģionu sauktu pār vārtiem. Šajā secības daļā, spēlētājs ir pa kreisi no vārtsarga un bumba kustas uz vārtiem. Šis ļoti vienkāršs piemērs ilustrē šīs attēlojuma veida elastīgumu. Pievērsiet uzmanību, ka ir šī aprakstīšana galvenokārt strukturāla, jo attiecības detalizētas grafa malās ir tīri fiziskas un mezgli attēlo segmentus. Vienīga skaidri izteikta semantiska informācija ir pieejama no tekstuālas anotācijas (kur atslēga vārdi tādi kā bumba, spēlētājs var būt specificēti).



zīm.8. Segmentu Attiecību Grafa piemērs.

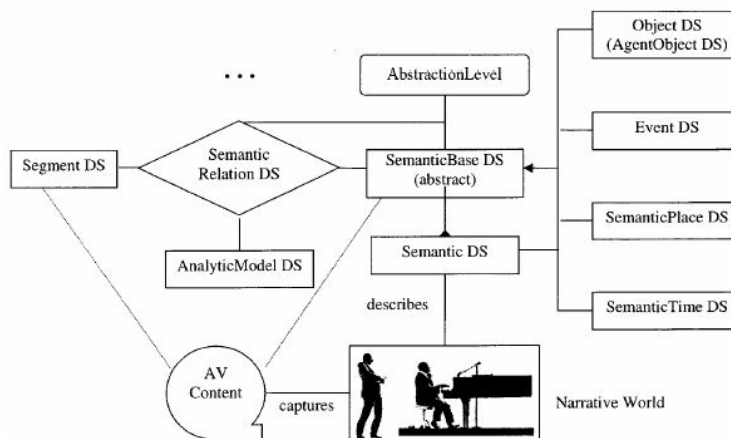
### B. Satura Konceptuālo Aspektu Aprakstīšana.

Dažām programmām, uzskats, aprakstītais iepriekšējā sekcijā, nav piemērots jo tas pasvītro satura strukturālos aspektus. Programmā, kurā nav struktūras reālas pielietošanas, bet lietotājs galvenokārt ieinteresēts satura semantikā, alternatīva pieeja ir apgādāta ar Semantikas DS. Šajā pieejā, uzsvārs nav segmentos, bet uz notikumiem, objektiem, vietām, laiku stāstījuma vārdos un abstrakcijā.

Stāstījuma vārdi attiecās saturam semantiskajai aprakstīšanai- tas ir “realitāte”, kur aprakstīšanai ir jēga. Priekšstats pārklāj pasauli attēlotu AV satura īpašās instancēs, ka arī vairāk abstraktas aprakstīšanās attēlojošās iespējamu pasauli aprakstītu iespējamajos mediju notikumos. Aprakstīšana vai iekļaut vairākas stāstījuma pasaules aprakstot vairākās AV satura instancēs.

Kā parādīts zīm.9., Semantikas Pamata DS apraksta stāstījuma pasaules un semantiskas būtības stāstījuma pasaulē. Turklāt, specializētu DS numurs ir iegūts no vispārēja Semantikas Pamata DS, kurš apraksta specifiskus semantikas būtību tipus, tādus kā stāstījuma pasaules, objekti, aģentu objekti, notikumi, vietas un laiki. Semantikas DS apraksta stāstījuma pasaules kuras ir aprakstītas vai saistītas ar AV saturu. Tas arī var būt izmantotas AV satura šablonu aprakstīšanai. Praktiski Semantikas DS ir paredzēts stāstījuma pasaules aprakstīšanas iekapsulēšanai. Objektu DS apraksta saskatāmus vai abstraktus objektus. Saskatāms objekts ir būtība kas pastāv, t.i., ir laika un telpas pakāpes, aprakstīšanas pasaulē. Abstraktais objekts ir abstrakcijas pielietošanas saskatāmam objektam rezultāts. Būtībā tas ģenerē objekta šablonu. Aģents Objekts DS izstiepjas no Objekta DS. Tas apraksta personu, organizāciju, cilvēku grupu vai personificētu objektu. Notikumu DS apraksta saskatāmu vai abstraktu notikumu. Saskatāms notikums ir dinamiskās attiecības iesaistot vienu vai vairākus objektus, sastopamus laika un telpas stāstījuma pasaules reģionā. Abstraktais

notikums ir abstrakcijas pielietošanas saskatāmajam objektam rezultāts. Šeit arī tas ģenerē notikuma šablonu. Beidzot, Semantikas Vieta un Semantikas Laiks DS apraksta, attiecīgi, vietu un laiku stāstījuma pasaulē.



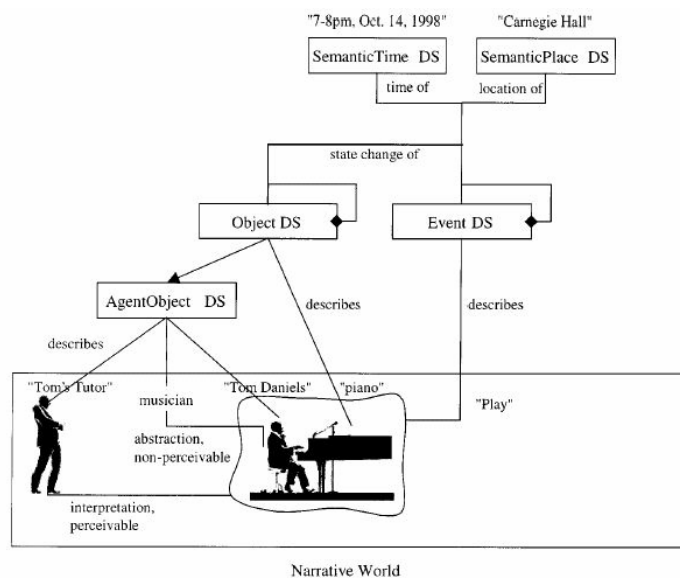
zīm.9. Konceptuālo aspektu aprakstīšanas instrumenti

Kā gadījumā ar Segmenta DS, aprakstīšanas konceptuālais aspekts var būt organizēts kokā vai grafā. Grafa struktūra ir definēta ar mezglu komplektu, attēlojot semantiskus uzskatus, un malu komplektu specificējot attiecības starp mezgliem. Malas ir aprakstītas Semantikas Attiecību DS.

Blakus individuālo instanču semantiskai aprakstīšanai AV saturā, MPEG-7 Semantiskais DS arī ļauj abstrakciju aprakstīšanu. Abstrakcija attiecās pie apraksta ņemšanas no specifiskās AV satura instances procesa un vispārinot tos vairāku AV satura instanču komplektam vai specifiskiem aprakstītājiem. Divu tipu abstrakcijas, sauktas mediju abstrakcija un standarta abstrakcija ir izskatīti. Mediju abstrakcija ir aprakstīšana, kurai jābūt atdalītai no AV satura specifiskām instancēm, un var aprakstīt visas AV satura instances, kuri ir pietiekoši līdzīgi.

Standarta abstrakcija ir mediju abstrakciju vispārinājums lai aprakstītu semantisku būtību vai aprakstītāju vispārējo klasi. Vispār, standarta abstrakcija ir dabūta ar specifisku objektu, notikumu vai citu semantisku būtību aizvietošanu pēc klasēm. Piemēram, ja “Toms spēlē klavierēs” ir aizvietots ar “Vīrietis spēlē klavierēs”, aprakstīšana tagad ir standarta abstrakcija. Standarta abstrakcijas var būt arī rekursīvas, tas ir var definēt vairāku abstrakciju abstrakciju. Tipiski, standarta abstrakcija ir paredzēta jaunai izmantošanai, vai izmantoties kā norādījums aprakstīšanā.

Parasts aprakstīšanas konceptuālo aspektu piemērs ir ilustrēts zīm.10. Stāstījuma pasaules iekļauj Tom Danielu spēlējot klavierēs un viņu skolotāju. Notikums ir aprakstīts ar semantisku laika aprakstīšanu: “7- 8 pēcpusdienā 14. oktobrī 1998. gadā” un semantisku vietu: “Karnēģija halle”. Aprakstīšana iekļauj vienu notikumu: spēlēšanu un četrus objektus: klavieres, Toms, viņa skolotājs un muzikantu abstraktais uzskats. Pēdējie trīs objekti pieder pie Aģents klases.



zīm.10. Konceptuālo aspektu aprakstīšanas piemērs.

## V. Navigācija un Pieeja.

MPEG-7 sekmē navigāciju un pieeju AV saturam ar kopsavilkumu, skatu, nodalījumu un variāciju aprakstīšanu. DS kopsavilkums apraksta semantiski nozīmīgus kopsavilkumus un rezumē AV saturu lai ieslēgt efektīvu pārlūkošanu un navigāciju. Telpas un Frekvences Skats DS apraksta AV signāla strukturālo skatu domēnu telpās un frekvencēs lai ļautu multi lēmumu pieeju un progresīvo atgūšanu. Variāciju DS apraksta attiecības starp dažādiem AV programmu variācijām lai ieslēgtu adaptīvu izvēlēšanu zem dažādiem termināliem, pārsūtīšanas un lietotāju izvēles apstākļiem. Šie instrumenti ir aprakstīti detaļās sekojoši:

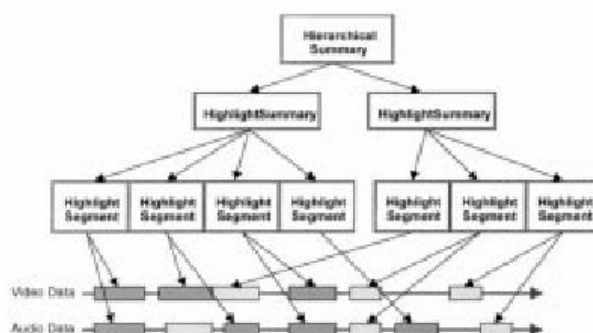
### A. Kopsavilkumi

Kopsavilkuma DS apraksta dažādus kompakus AV satura kopsavilkumus kas sekmē AV satura atklāšanu, pārlūkošanu un navigāciju. Kopsavilkuma aprakstītāji ļauj AV saturam būt vadītam kā hierarhiskajā, tā arī sekojošajā veidā. Hierarhiskie kopsavilkumi organizē saturu secīga detaļu līmenī. Sekojošie kopsavilkumi sastāda tēlu secību, iespējami sinhronizētu ar audio, lai aprakstītu slaidu izrādi vai AV izšķiršanu.

1) Apkopošanas DS: MPEG-7 pārskati ļauj ātru un efektīvu pārlūkošanu un navigāciju ievērojamas informācijas atšķiršanu no AV satura. Apkopošanas DS satur saites AV saturam, segmentu vai struktūras līmenī. Dodot MPEG-7 apkopošanas aprakstīšanu, termināla ierīce, tāda kā ciparu televīzijas kaste, piekļūst AV materiālam sastādot pārskatu un sniedzot rezultātu mijiedarbībai ar lietotāju. Apkopošanas DS var aprakstīt daudzkārtīgus pārskatus vienādam AV saturam, lai sniegtu dažādus detaļu līmeņus vai izceltu specifiskas īpašības, objektus, notikumus vai semantikas. Pievienojot saites AV saturam pārskatos, ir iespējams ģenerēt un saglabāt daudz pārskatus bez AV satura multi versiju glabāšanas.

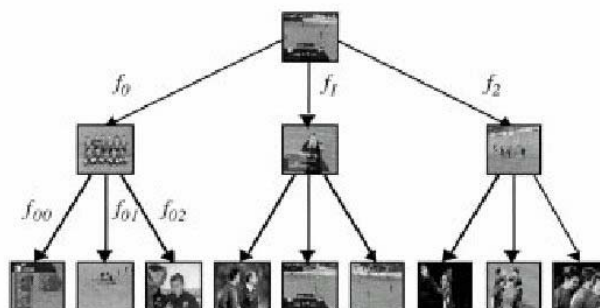
2) Hierarhiska Pārskata DS: Hierarhiska pārskata DS apraksta pārskatu organizāciju daudzkārtīgos līmeņos lai aprakstītu dažādus līmeņus pārejošo detaļu līmeni. Hierarhiska Pārskata DS konstruēts visapkārt pārejošo AV satura segmentu vispārēja priekšstatam, aprakstītam ar DS Izcelšanas Segmentu. Katrs Izcelšanas Segments satur AV

saturu lokatorus, kuri sniedz pieeju asociētiem atslēgām- videoklipiem, atslēgām- audio klipiem, atslēgām- struktūrām un atslēgām- skaņām. Katrs var saturēt tekstuālo anotāciju kura apraksta atslēgas- tematus. Šie AV segmenti grupēti pārskatos, vai izcelšanās izmantojot Izcelšanas Pārskata DS. Piemērām, zīm.11., hierarhiskais pārskats satur divus pārskatus, kur pirmais pārskats sastāv no četriem izcelšanas segmentiem un otrais pārskats sastāv no trim izcelšanas segmentiem. Pārskatiem jāatbild divām dažādām tēmām un jāsniedz alternatīvus skatu oriģinālajam AV saturam. Izcelšanas Pārskata DS ir rekursīvs pēc dabas, ļaujot pārskatam saturēt citus pārskatus. Šī iespēja var izmantoties vairāku hierarhisku pārskatu veidošanai, t.i., lai aprakstītu saturu no dažādām pusēm. Papildus, vairāki pārskati var būt grupēti kopā izmantojot Hierarhiska Pārskata DS.



zīm.11. Hierarhiskās Struktūras DS, saturoša divas pārskaites, ilustrācija.

Zīm.12. rada futbola video hierarhiska pārskata piemēru. Hierarhiska Pārskata aprakstītājs dod trīs detaļu līmeņus. Šajā piemērā futbola spēles video pārskatīts vienā pirmsākuma struktūrā. Nākošais hierarhijas līmenis sniedz trīs struktūras kuras pārskata dažādus video segmentus. Beidzot, zemākais līmenis sniedz attiecīgas struktūras, attēlojot vairākās detaļās scēnas attēlotas segmentos.



zīm.12. Futbola spēles video hierarhiska pārskata piemērs nodrošinot multi līmeņu atslēgu struktūras hierarhiju.

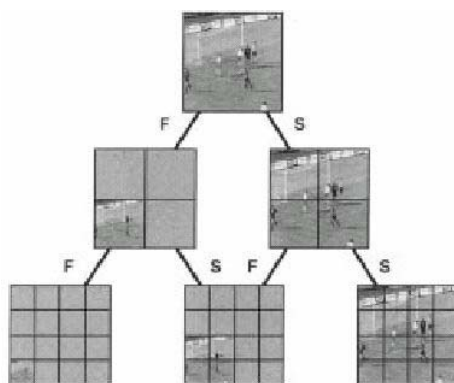
3) Sekojošā Pārskata DS: Sekojošā Pārskata DS apraksta pārskatu, sastāvošo no attēlu vai video kadru secības, kura ir iespējami savienotā ar audio. Sekojošais Pārskats var arī saturēt audio klipu secību. AV saturs kurš veido Sekojošo Pārskatu var glabāties atsevišķi no oriģinālā AV satura lai pieļautu ātru vadīšanu un pieeju. Alternatīvi, Sekojošais Pārskats var tieši savienot oriģinālo AV saturu lai samazinātu glabāšanas izmērus.

## B. Sadalīšanas un Dekompozīcijas

Pārskata DS apraksta audio vai vizuāla signāla strukturālo pārskatu, sadalīšanu vai dekompozīciju telpā, laikā un frekvencē. Vispār, signālu skati atbilst zemas rezolūcijas skatiem, vai telpas, vai laika segmentiem, vai frekvences apakš grupām. *Telpas* un *Frekvences Skata* DS apraksta skatu atbilstoši sadalīšanas telpā vai frekvencē. Papildus, Dekompozīcijas DS apraksta koka- vai grafa- bāzētas audio vai vizuāla signāla, vai skatu organizācijas dekompozīcijas. Koka- vai grafu- bāzētos dekompozīcijās mezgls atbilst skatam un pāreja atbilst analīzes un sintēzes signāla apstrādei starp savienotiem skatiem:

- 1) Skata DS: Skata DS apraksta telpas vai frekvences audio vai vizuāla signāla skatu. Telpas Skata DS apraksta telpas audio vai vizuāla signāla skatu, piemēram, attēla telpas segmentu. Frekvences Skata DS apraksta audio vai vizuāla signāla skatu noteiktas frekvenču grupas ietvaros, piemēram, audio signāla vilnīto apakš grupu. Telpas Frekvences Skata DS apraksta vairāku dimensiju audio vai vizuāla signāla skatu vienlaicīgi telpā un frekvencē, piemēram, attēla telpas segmenta vilnīto apakš grupu. Rezolūcijas Skata DS apraksta zemas rezolūcijas audio vai vizuāla signāla skatu, tādu kā sīktēla attēla skats. Konceptuāli, rezolūcijas skats ir frekvences skata īpašais gadījums kurš atbilst signāla zemas frekvences apakš grupai. Telpas Rezolūcijas Skata DS apraksta skatu vienlaicīgi audio vai vizuāla signāla telpā un rezolūcijā, piemēram, attēla telpas segmenta zemas rezolūcijas skatu.
- 2) Skata Dekompozīcijas: Skata Dekompozīciju DS apraksta telpas un frekvences audio vai vizuālo signālu skatu dekompozīciju vai organizāciju. Skatu Komplekts DS apraksta skatu komplektu, kuram var būt komplektības un pārmērības dažādas īpašības. Piemēram, vilnītas apakš grupas audio signālu komplekts formē pilnu un pārmērīgo skatu komplektu. Telpas Koka DS apraksta telpas- koka audio vai vizuāla signāla dekompozīciju, piemēram, telpas četrstūra koka attēla dekompozīciju. Frekvences Koka DS apraksta frekvences koka audio vai vizuālo signālu dekompozīciju, piemēram, vilnīto paketes koka attēla dekompozīcija. Telpas Frekvences Grafa DS apraksta audio vai vizuāla signāla dekompozīciju vienlaicīgi telpā un frekvencē kuros skati ir organizēti izmantojot telpas un frekvences grafus. Video Skata Grafs DS apraksta specifisku video signālu dekompozīcijas tipu kā telpas, tā arī laika frekvences kuri atbilst 3-D apakš grupas dekompozīcijai. Beigās, Multi Rezolūcijas Piramīdas DS apraksta audio vai vizuālo signālu multi rezolūciju skatu hierarhiju, tādu kā piramīdas attēls.

Zīm.13. rada Telpas un Frekvences Grafa attēla dekompozīcijas piemēru. Telpas un Frekvences Grafa struktūra satur mezglus kuri atbilst citiem attēla telpu un frekvenču skatiem. Skati atbilst 2-D attēla signāla sadalīšanai telpā (telpas segmenti), frekvencē (vilnītas apakš grupas) un telpas un frekvences (telpas segmentu vilnītas apakš grupas). Telpas un frekvences grafs satur arī pārejas kuras atbilst analīzes un sintēzes atkarībām skatos. Piemēram, zīm.13, katra "S" pāreja nozīmē telpas dekompozīciju, kamēr katra "F" pāreja nozīmē frekvences vai apakš grupas dekompozīciju.

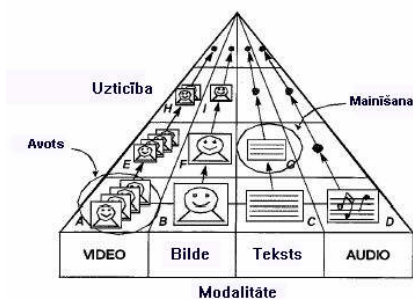


zīm.13. Telpas un Frekvences Grafs apraksta audio vai vizuāla signāla dekompozīciju

### C. Satura Variācijas.

Variāciju DS apraksta AV satura variācijas, tādas kā saspiešanas vai mazas rezolūcijas versijas, pārskati, dažādas valodas un dažādas modalitātes, tādas kā audio, video, attēli, teksts u.t.t. Viena no Variāciju DS mērķa funkcijām ir pieļaut serverim vai proxy izvēlēties AV satura vairāk piemērotu variāciju piegādei saskaņā ar terminālu iekārtu, tīkla apstākļiem vai lietotāja izvēles funkcionālām iespējām. Variāciju DS apraksta dažādas alternatīvas variācijas. Variācijas var attiekties pa jaunu autorizētiem AV saturam, vai atbilst AV saturam dabūtam no cita avota. Variācijas precizitātes vērtība dod variācijas kvalitāti salīdzinot ar oriģinālu. Variācijas tipa atribūts norāda variācijas tipu, tādu kā pārskats, abstraktu, ekstraktu, modalitātes tulkošana, valodas tulkošana, krāsas reducēšana, telpas reducēšana, likmes reducēšana, saspiešana u.t.t.

Zīm.14. ilustrē AV programmu variāciju komplektu. Piemērs rada video programmas avotu kreisā stūrā (A) un astoņas programmu variācijas. Variācijām ir dažādas modalitātes: divas variācijas ir video programmas (E,H), trīs ir attēli (B,F,I), divi ir teksti (C,G), un viena ir audio (D). Katrai no variāciju programmām ir dažāda precizitātes vērtība, kura norāda variāciju programmu precizitātes ar uzticību avota programmai.



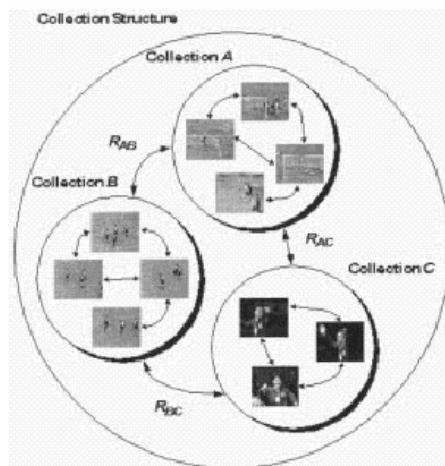
zīm.14. AV programmu avota variācijas ilustrācija.

## VI. Satura Organizācija

Savākšanās DS apraksta AV satura kolekcijas, aprakstītāja instances, koncepcijas, vai maisīto saturu. Kolekcijas var būt izmantotas uzdevumiem, tādiem kā dziesmu albuma, objektu grupas vai aprakstītāju krāsas pazīmju klastera aprakstīšana. Modeļa DS apraksta AV satura vai kolekcijas parametru modeļus. Modeļi var būt izteikti statistikas vai varbūtības ietvaros asociētas ar AV satura kolekcijas atribūtiem, vai var būt izteikti caur AV satura klašu piemēriem vai eksemplāriem.

### A. Kolekcija

Kolekcijas DS apraksta kolekciju saistītu ar AV saturu. Kolekcijas DS iekļauj instrumentus AV materiāla kolekciju, AV satura aprakstītāju kolekcijas, semantisku koncepciju kolekcijas, satura, aprakstīšanu un koncepciju maisījuma kolekciju aprakstīšanai. Zīm.15. rada vienu kolekciju organizēšanu kolekcijas struktūras ietvaros. Šajā piemērā, katra kolekcija sastāv no attēlu komplekta ar līdzīgām īpašībām, piemēram, katrs attēlotais līdzīgais notikums futbola spēlē. Katras kolekcijas iekšā, attiecības starp attēliem var būt aprakstīti, tādi kā attēlu līdzīguma pakāpe klasteri. Pāri kolekcijai, attiecīgas attiecības var būt aprakstītas, tādas kā līdzīguma pakāpe kolekcijā.



zīm.15. Kolekcijas Struktūras DS

### B. Modeļi

Modeļu DS apraksta parametru AV satura, aprakstītāju vai kolekciju modeļus. Varbūtības Modeļa DS apraksta dažādas statistiskas funkcijas un varbūtības struktūras, kuras var izmantot lai aprakstītu AV satura paraugus un Aprakstītāju klases izmantojot statistisko aproksimāciju. Analītiskā modeļa DS apraksta AV satura piemēru vai Aprakstītāju klasteru kolekciju, kura ir izmantota lai sagādātu modeļi sevišķajai semantiskajai klasei. Piemērām, mākslas attēlu kolekcija apzīmēta ar etiķeti nozīmē, ka gleznas ir Impressionizma perioda formu piemēri. Analītiskā Modeļa DS arī neobligāti apraksta uzticības, pie kurām semantiskas etiķetes ir piešķirti. Klasifikatora DS apraksta dažādus klasifikatoru tipus, kuri izmantojās lai piešķirtu semantiskas etiķetes AV saturam vai kolekcijām.

## VII. Lietotāju Mijiedarbība.



Lietotāju Mijiedarbības DS apraksta lietotāju attiecības izvēles AV satura patērēšanā, kā arī izmantošanas vēsturi. MPEG-7 AV satura aprakstītāji var sakrist ar izvēles aprakstītāju lai atlasīt un personificēt AV saturu vairāk efektīvu un iedarbīgu pieeju, prezentāciju un izmantošanu. Lietotāja Izvēles DS apraksta satura un pārlūkošanas modeļu dažādu tipu priekšrokas, ieskaitot satura atkarību laika un vietas ietvaros. Lietotāju Izvēles DS apraksta arī dažādu izvēļu relatīvu svarīgumu svaru, privātas izvēļu īpašības. Lietotāja Vēstures DS apraksta notikumu vēsturi, kurus vadīja multimediju sistēmu lietotājs. Lietošanas vēstures aprakstītāji var būt apmainīti starp lietotājiem, vidu aģentiem, satura sniedzējiem un iekārtām, un var būt izmantoti lai izskaitļot lietotāja izvēles AV saturā.

#### VIII. Nobeigums

Šajā papīrā mēs prezentējam MDS pārskatu, attīstītu ar MPEG dabu MPEG-7 standarta. MDS ir meta datu struktūras AV satura aprakstīšanai un anotācijai un XML aprakstīšanas ceļa sniegšanu svarīgās koncepcijās attiecīgo AV saturam. Mērķis ir iespēja meklēt, indeksēt, filtrēt un gūt pieeju ar daudzu iekārtu ieslēgšanu kuri darbojas ar AV satura aprakstīšanu.

Mēs prezentējam MPEG-7 MDS organizāciju dažādos pamata elementu, shēmu instrumentu, satura aprakstīšanas, satura vadīšanas, satura organizācijas, vadīšanas un pieejas vietās. MPEG-7 MDS attīstīšanas procesā, mēģinājumi bija vērsti harmonizēt šanai ar citu multimediju meta datu standartiem un eksistējošiem standartiem. Bez standartizētas MPEG-7 MDS kopas, MPEG-7 DDL arī ļauj MPEG-7 standarta paplašinājumam pievienot specifiskus AV satura domeinus un programmas.

## MPEG-7 Sistēmas: Pārskats

*Abstrakti-* Šis papīrs dod ISO/IEC 15939 (MPEG-7 Sistēmas) pārskatu. Sākumā tas rada MPEG-7 Sistēmu darbības mērķus. MPEG-1 un MPEG-2 standartos, "Sistēmas" attiecas pie pilnas arhitektūras, multipleksēšana, un sinhronizācijas. MPEG-4, papildus šiem īpašībām, daļa no sistēmām apņēma dialogu scēnu aprakstīšanu, satura aprakstīšanu un programmēšanas iespēju. MPEG-7 nes jaunus izsaukumus Sistēmu ekspertīzei, tādai kā valodas

aprustīšanas reprezentācijai, bināro aprakstīšanas reprezentācijai un aprakstīšanu nešanu gan atsevišķu, gan ar audio-vizuāliem saturiem. Šis papīrs pēc tam rada MPEG-7 sistēmu specifiku aprakstīšanu, sākot ar pamata arhitektūru līdz individuālo MPEG-7 sistēmu instrumentu aprakstīšanas. Beigās, nobeigums apraksta standartizācijas mēģinājumu statusu, kā arī nākotnes specifiku paplašinājumus.

*Indekss-* API, arhitektūra, audio- vizuāls, bināro formāts, satura aprakstīšana, MPEG-7 sistēmas, multipleksēšana, programmēšana, specifiku, sinhronizācija, instrumenti.

## **I. Ievads.**

"Sistēmu" koncepcija MPEG dramatiski attīstījās no MPEG-1 un MPEG-2 standartu attīstīšanas. Pagātnē, "Sistēmas" attiecas pie pilnas arhitektūras, multipleksēšana, un sinhronizācijas. MPEG-4, papildus šiem īpašībām, daļa no sistēmām apņēma dialogu scēnu aprakstīšanu, satura aprakstīšanu un programmēšanas iespēju.

MPEG-7 nes jaunus izsaukumus Sistēmu ekspertīzei, tādai kā valodas aprakstīšanas reprezentācijai, bināro aprakstīšanas reprezentācijai un aprakstīšanu nešanu gan atsevišķu, gan ar audio- vizuāliem saturiem. Jaunu audio-vizuāla satura aprakstīšanas iespēju kombinācija, piedāvāta no MPEG-7 sistēmām, un efektīvie aprakstīšanas instrumenti, doti no vizuālas, audio un MDS standarta daļām, būs jauna veida domāšanas par audio- vizuālo informāciju pamatu.

Tiešām, pirms MPEG-7, audio vizuālie dati bija galveni par neskaidru bitu sēriju. Tikai šo bitu dekodēšana dos kādu informāciju par to, kas tā ir par informāciju un kas lietotājam ar to jādara. Dekodēšanas process prasa, pamatā, kompleksu un atmiņas prasošus operācijas un prasa lielu caurlaides spēju tīklos.

Ar MPEG-7 aprakstītāju un aprakstīšanu shēmu izmantošanu, MPEG-7 nodrošina iespēju saņemt informāciju par audio- vizuāliem datiem bez šo datu faktiskas dekodēšanas prasības. MPEG-7 sistēmu specifiku pabeidz ainu, saistot MPEG-7 aprakstīšanu ar audio- vizuālo saturu un nodrošinot efektīvu bināro aprakstīšanas un aprakstīšanas shēmu datu reprezentāciju labākajās MPEG tradīcijās.

Šis papīrs dod MPEG-7 sistēmu īsu apskatu. Tas ir strukturēts ap MPEG-7 sistēmu mērķiem un arhitektūras sekojoši:

- Mērķi: Šī sekcija apraksta MPEG-7 sistēmu specifiku attīstīšanas motivāciju un paskaidrošanu. Kā ar visiem MPEG aktivitātēm, MPEG-7 sistēmas vadās ar prasījumu kopu, t.i. mērķu kopu, kurus vajag izpildīt ar specifiku rezultātiem no apakšgrupu darba vai aktivitātēm. Šis papīrs sniedz specifisku uzmanību ceļam kā MPEG-7 sistēmu prasības ir dabūti no pamata koncepcijas MPEG-7 Multimediju Satura Aprakstīšanas Interfeiss.

- Arhitektūra un Instrumenti: Šī sekcija apraksta kopējo MPEG-7 struktūru, zināmu kā "MPEG-7 Sistēmas Arhitektūra". Pilnais MPEG-7 sesijas apskats izdala dažādas fāzes, kuras lietotājs sekos lietojot MPEG-7 aprakstītājus. MPEG-7 ir "instrumentu komplekta" standarts, sniedzot instrumentu kopu, kuru grupas der dažādām programmām. Šī sekcija sniedz arī MPEG-7 sistēmas instrumentu funkcionālo aprakstīšanu.

Šī MPEG-7 sistēmu apskatīšanas nobeigumā, šis papīrs sniedz informāciju par standartizācijas mēģinājumu statusu, kā arī īsu dažu sistēmu tehnoloģiju aprakstīšanu, kuri tagad ir zem MPEG Standartizācijas Komitejas apskatīšanas nākošiem MPEG-7 standarta versijām.

## **II. Mērķi.**

### **A. Prasījumi.**

Lai saprastu MPEG-7 aktivitātes iracionalitāti, labs starts ir viens no fundamentāliem MPEG-7 dokumentiem- MPEG-7 prasījumi. Šis dokuments sniedz plašu mērķu sarakstu, kurus vajag izpildīt ar MPEG-7 specifikācijām. Tas arī definē terminoloģiju, kura ir izmantota tālāk šajā papīrā.

MPEG-7 sistēmas pieprasījumi var būt kategorizēti divās grupās: tradicionālas MPEG-7 sistēmas pieprasījumi, kuri aprakstīti Sekcijā II-B un specifiskie MPEG-7 sistēmas pieprasījumi detalizēti Sekcijā II-C.

### **B. Tradicionālie MPEG sistēmu pieprasījumi.**

Atslēga pieprasījumi sistēmu specifikāciju MPEG-1, MPEG-2 un MPEG-4 attīstīšanai bija kodēta audio, video un lietotāja- noteiktus privāto datu sniegšanas iespēja, un laika izvēles mehānisma ieslēgšana, lai atvieglotu sinhrono dekodēšanu un šo datu reprezentācija klienta pusē. Šie prasījumi arī ir daļa no MPEG-7 sistēmu fundamentāliem prasījumiem un aprakstīti zemāk:

1) Sniegšana: Multimediju aprakstītājiem jābūt sniegtiem izmantojot pārsūtīšanas un glabāšanas protokolu dažādumu. Daži no šiem pārsūtīšanas protokoliem iekļauj straumi, piemēram, tieša laika aprakstītāju un satura pārsūtīšana. Šajos gadījumos, multimediju aprakstītājus vajag pārsūtīt pa daļām, lai aprakstītāju pārsūtīšana klientam sakristu ar limitētiem tīkla un terminālu iespējam. Sniegšana nozīmē, arī, multipleksēšanas instrumentu definēšana, lai ieslēgt daudzkārtējus MPEG-7 aprakstītājus vienīga datu straumē kopa ar asociētu saturu.

2) Sinhronizācija: Tipiski, dažādi audio- vizuālas prezentācijas komponenti ir tuvu saistīti laikā. Dažām programmām, aprakstīšanas informācijai jābūt prezentētai lietotajam precīzajos laikos, kopā ar saturu. MPEG-7 reprezentācijai jāļauj laika definīcijas precīzu noteikšanu lai dati, saņemti straumes veidā, varētu būt apstrādāti un parādīti pareizajos laika momentos un būt sinhronizētiem viens ar otro ar laiku.

3) Straumes Vadīšana: Beidzot, audio vizuālas informācijas pilna vadīšana ieskaitot MPEG-7 aprakstītājus nozīmē dažu instrumentu nepieciešamību lai programmas izmantot saturu. Šie papildus mehānismi, tādi kā viennozīmīga datu atrašanas vieta, datu tipa identifikācija, neatkarības starp datu elementiem aprakstīšana, aprakstītāju asociācija ar saturu, un pieeja intelektuālam īpašumam asociētam ar datiem.

### **C. MPEG-7 specifisku sistēmu prasījumi.**

Papildus šiem prasījumiem, MPEG-7 nes noteiktas prasības lai būtu risināti sistēmas līmenī: valodas aprakstīšanas shēmu reprezentācijai un bināro un dinamisku aprakstītāju reprezentācija. Pirmā prasība ir risināta ar MPEG-7 Aprakstīšanas Definīcijas Valodu (MPEG-7 DDL). Prasījumi dinamiskiem un bināro aprakstītājiem ir aprakstīti tālāk.

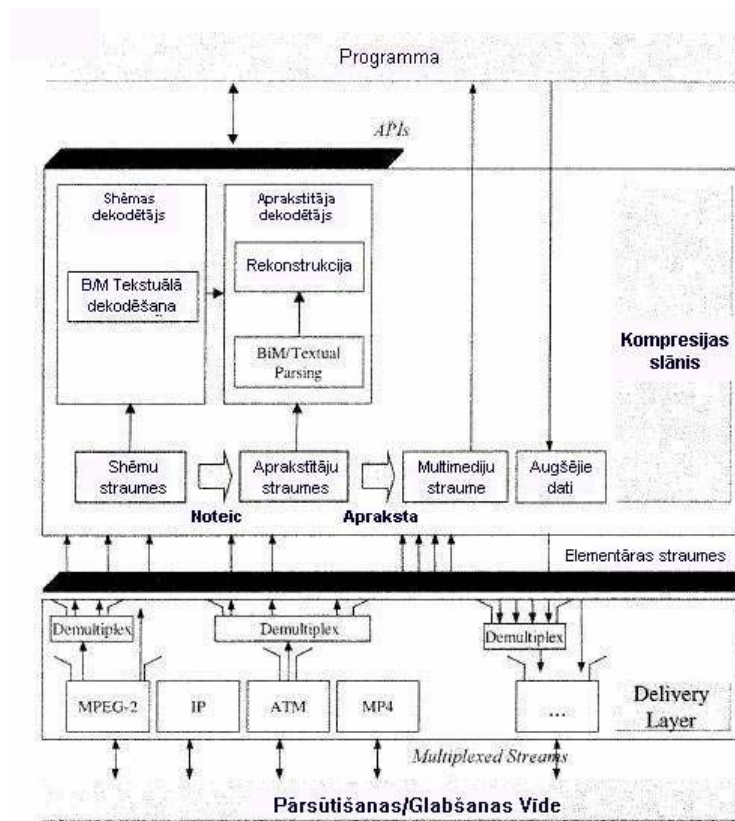
Ir gaidīts kā dažos ekspluatācijas MPEG-7 vīdēs, sniegšana (tīkla vai glabāšanas) resursi būs nepietiekami. Tāpēc datiem jābūt saspīestiem un pārsūtīti augošā ceļā. Tāpēc galvenie prasījumi BiM ir kompakta un straumes MPEG-7 aprakstītāju reprezentācija. BiM ir ekvivalents testa (XML) aprakstīšanai definētai ar MPEG-7 DDL un ir divvirzienu kartogrāfija starp BiM un teksta DDL bāzētiem reprezentācijām. Kā arī ir gaidīts ka dažas programmas izmantos MPEG-7 BiM- kodētu saturu tieši, bez obligāta starposma teksta satura rekonstruēšanas soļa.

Bināro formāts tāpēc ļauj ātru MPEG-7 straumes analīzes. Tas arī izstrādāts ceļā, kurš ļauj BiM analizatoram pārbaudīt sintaksiskas pareizību, piemēram, labu formu un likumību attiecībā MPEG-7 bitstream normatīviem aspektiem.

## **III. Arhitektūra un Instrumenti.**

### **A. MPEG-7 Terminālu Arhitektūra.**

Informācijas reprezentācija, specificēta MPEG-7 standartā nodrošina līdzekļus lai aprakstītu multimediju saturu. Objekts kurš izmanto tādu reprezentāciju ir parasti saucas par "terminālu". Šis termināls var atbilst autonomajai programmai vai būt par programmu sistēmas daļu. MPEG-7 termināla pilna arhitektūra ir attēlota zīm.1.



zīm.1. MPEG-7 sistēmu struktūra.

Pārsūtīšanas/ glabāšanas vide ir figūras apakšā. Šī vide attiecas pie zema līmeņa pārsūtīšanas infrastruktūrā. Šie slāņi pārsūta multipleksētas straumes Pārsūtīšanas slānim. MPEG-7 aprakstītāju transportēšana var parādīties vairākās pārsūtīšanas sistēmās. Šie ir, piemēram, MPEG-2 transporta straumes, Interneta Protokols (IP), vai MPEG-4 (MP4) faili vai straumes. Pārsūtīšanas slānis aptver mehānismus, ļaujot MPEG-7 aprakstīšanas sinhronizāciju, kadrēšanu un multipleksēšanu. MPEG-7 aprakstīšana var būt pārsūtīta neatkarīgi vai ar aprakstītu saturu. Pēc demultipleksēšanas soļa, pārsūtīšanas slāņa izeja ir elementāru straumju kopa. Šīs elementāras straumes nodrošina informācijas gabalus par MPEG-7 aprakstīšanu, par shēmu izmantotu aprakstītāja definēšanai, pat par pašu multimediju saturu. MPEG-7 ir tikai standartizēšanas straumes aprakstīšanai un shēmām, kuri ir MPEG-7 aprakstītāju un aprakstīšanas shēmu XML-Shēmas specifikācijas.

Informācijas pārsūtīšana ne vienmēr iekļauj tūru downstream scenāriju. MPEG-7 termināla arhitektūra ļauj datus pārsūtītiem atpakaļ no termināla uz nosūtītāju vai serveru. Šīs datu tips saucās par upstream datiem un tipiski iekļauj pieprasījumus formulētus ar beigas lietotāju vai atsevišķu informācijas daļu pieprasījumu.

Pārsūtīšanas slānis arī sniedz MPEG-7 elementāras straumes saspīstajam slānim. MPEG-7 elementāras straumes sastāv secīgi un individuāli pieejamās daļās sauktās par pieejas vienībām. Pieejas vienība ir mazākais datu objekts pie kura laika informācija var būt pielietota. MPEG-7 elementāras straumes satur dažāda rakstura informāciju.

1) *Shēmas Straumes*: Šī informācija apraksta MPEG-7 aprakstītāja struktūru. Dažām programmām, tā var zināt kā programmai ir zināms, kāda shēma pielietojās aprakstītāja ģenerēšanai. Šajā gadījumā, shēmai nevajag būt pārsūtītai terminālā.

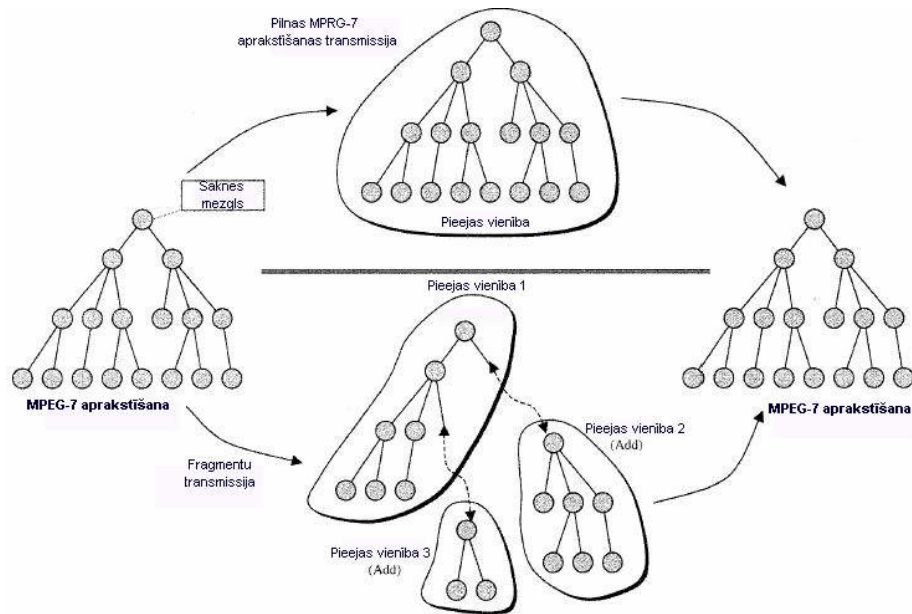
2) *Aprakstīšanas Straumes*: Šī informācija var būt gan pilna multimedija satura MPEG-7 aprakstīšana, gan šī aprakstīšanas fragments. Fragmenti ir izmantoti situācijās kur veselas aprakstīšanas kā viena gabala pārsūtīšana nav iespējama. Šajā gadījumā MPEG-7 aprakstīšanai jābūt rekonstruētai kombinējot vairākus fragmentus. MPEG-7 aprakstīšana var būt reprezentēta gan tekstuālajā formātā (XML), gan bināro formātā (BiM), vai divu formātu maisījums, atšķirībā no lietojamās programmas. MPEG-7 noteic unikālo kartogrāfiju starp bināro un teksta formātiem. Divpusēja loss- less kartogrāfija starp teksta reprezentācijai un bināro reprezentācijai ir iespējama. Tomēr, to nevajag vienmēr lietot: dažas programmas varbūt negribēs pārsūtīt visu informāciju saturēto teksta reprezentācijā un var izvēlēties lossy pārsūtīšanu kura ir vairāk efektīva.

Kompresijas līmenī, Pieejas Vienību straume (kā tekstuālo, tā bināro) ir analizēti, un MPEG-7 aprakstīšana rekonstruējas. MPEG-7 nepilnvaroti teksta reprezentācijas rekonstruēšanu kā starpposma solis dekodēšanas procesā. MPEG-7 bināro straume var būt analizēti ar BiM analizatoru, transformēta teksta formātā un pēc tam apstrādāta teksta formātā tālākajai rekonstruēšanai. Alternatīvajā scenārijā, bināro straume var būt analizēti ar BiM analizatoru un pēc tam izmainīta vajadzīgajā formātā turpmākajai apstrādei.

Vienu reizi rekonstruēta, programma pēc tam ir gatava ekspluatēt MPEG-7 aprakstītājus, iespējami ar multimediju elementāriem straumēm. Sekojošā sekcija sniedz vairāk informācijas par trīs galvenajiem MPEG-7 sistēmas instrumentiem: MPEG-7 pieejas vienība, BiM un MPEG-7 aprakstītāju sniegšana.

## B. Pieejas Vienības

MPEG-7 pieejas vienības strukturēti kā komandas iekapsulē MPEG-7 aprakstītāju vai shēmu. Komandas sniedz MPEG-7 aprakstītāju dināmiskus aspektus: viņi ļauj aprakstītājam būt pārsūtītiem kā viens gabals vai būt fragmentētam vairākos gabalos. Šī īpašība ir ilustrēta zīm.2. MPEG-7 aprakstīšana vai shēma tekstuālā (XML) formātā ir fiziski koka struktūra. Augšējā zīm.2. daļa ilustrē kā koks var būt slēpta vienā pieejas vienībā kura ir pēc tam pārsūtīta terminālā. Otrais scenārijs ilustrēts zīm.2. zemākā daļā. Šajā gadījumā MPEG-7 aprakstīšana ir fragmentēta trijos gabalos, kuri ir slēpti dažādos pieejas vienībās. Finālā aprakstīšana ir rekonstruēta ar pieejas vienību 2 un 3 "pievienošanu" pie attiecīga pieejas vienības 1 satura koka mezgla. Izņemot "pievienot" funkcionālas iespējas, komandas arī ļauj pamata operācijas ar MPEG-7 aprakstīšanu tādu kā aprakstītāju vērtības atjaunošana, aprakstītāja daļas iznīcināšana vai jaunas shēmas pievienošana. Saspiešanas slāņa rekonstrukcijas stadija atjauno aprakstīšanas informāciju un asociētas shēmas informāciju pielietojot šīs komandas. Kopumā, pieejas vienības definē: 1) komandu tips pielietošanai terminālā (pievienot, iznīcināt, atjaunot un citi); 2) MPEG-7 aprakstīšana vai shēma izmantojama modifikācijai un 3) Koka mezgla atrašanas vieta kur šīs modifikācijas ir vajadzīgas.



zīm.2. MPEG-7 aprakstīšanas un pieejas vienības.

### C. Binārais Formāts.

Ir divi galvenie iemesli pakešu- pamatotu bināro formāta izmantošanai MPEG-7 aprakstītājiem. Pirmais, teksta formāts pieprasa vairāk glabāšanas vai pārsūtīšanas resursu nekā vajadzīgi no teorētiska viedokļa. Tāpēc, teksta formāta efektīva saspiešana ir pielietota mainot to bināro formātā. Otrais, teksta formāts nav ļoti derīgs straumes programmām. Šī tipa programmām pieprasīts liela lokanība uzticības līmenis elementu pārsūtīšanas kārtībai. Ka arī viegla nejausai pieejai jābūt sniegtai bez analīzes veselu bitu straumi. Tas pieprasa pakešu bāzētu struktūru, kurai virsraksti pamatā parādīti bināro formātā.

MPEG-7 bitu straumi ir stādīts no tā sauktiem BiM fragmentiem. Viņu struktūra ir ilustrēta zīm.3. BiM fragments ir sastādīts no trijām daļām un pārsūta tāda paša tipa informāciju kā tekstuāla pieejas vienība.



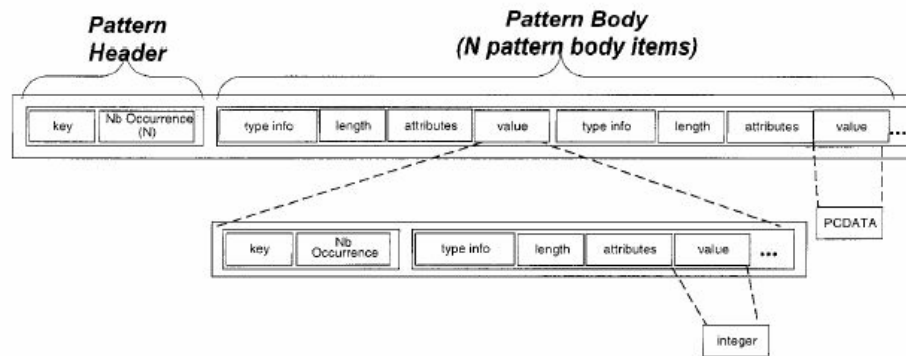
zīm.3. BiM fragmenta struktūra.

Pirmā daļa "Navigācijas komanda un ceļš" kura signalizē terminālam kur vajag pielietot MPEG-7 aprakstītāju atjaunošanu. Adresēšana noteikta ar navigācijas ceļu, kura noteic koka mezgla adresi kuram dekodētājs saņēma informāciju saturētu BiM fragmentā. Dažādie aprakstīšanas elementu adresēšanas iespējas ir iespējamas. Piemērām, viens var izmantot relatīvu adresēšanu ar uzticību mezglam, kur tagad atrodas termināls vai absolūto adresēšanu ar uzticību aprakstīšanas saknes mezglam. BiM fragmenta otrā daļa ir "Satura manipulācijas komanda". Tas stāsta dekodētājam kādu satura manipulācijas veidu viņam vajag izpildīt jau rekonstruētas aprakstīšanas un informācijas daļas fragmentā kombinēšanā. Kā gadījumā ar pieejas vienībām, iespējamas komandas ir "pievienot", "atjaunot" un "izdzēst".

Daļā no MPEG-7 aprakstīšanas ir saturēta trešajā BiM fragmenta komponentā. Tas galvenais bināro formāts ir sastādīts no viena globāla virsraksta un virknes un ligzdas paraugu kopas. Paraugi ir sastādīti no diviem galvenajiem elementiem, kā ir parādīts zīm.4.:

1) Parauga virsraksts, kurš ir sastādīts no "atslēga", specificējoša sekojošo datu elementu formātu un "parādīšanas numura", specificējot šo datu elementu numuru;

2) Parauga ķermenis, kurš ir  $n$  virknes vienādu gabalu kopa, kur  $n$  ir parādīšanas numurs, parādīts parauga virsrakstā.



zīm.4. Apakš koka bināras reprezentācijas formāts.

Katra ķermeņa vienība ir sastādīta no "tipa informācija" kodēšanas XML shēmas elementa tips, "garums" aprakstot kodēšanas garumu bitos lai nodrošināt ātru pieeju straumē, "atribūti" un "vērtība" kodējot aprakstīšanas datus. Vērtībā paša var būt par paraugu.

Kad termināls zina shēmu atbilstošu aprakstīšanai, dažas komponentes parauga ir papildus: piemērām, "atslēga" nav kodēta ja XML shēmas elements vai atribūts ir piespiedu. "Parādīšanas numurs" nav kodēts ja tas ir fiksēts shēmā. Atribūtu dati nav kodēti ja tipam nevar būt atribūti (piemērām XML shēma simpleType). garums ir kodēts tikai ja ātras pieejas īpašība ir pieprasīta, u.t.t.

#### D. MPEG-7 Aprakstītāju Sniegšana.

MPEG-7 aprakstītāju sniegšana specifiskajās sistēmās ir ārpus MPEG-7 standarta iespēju loka. Eksistējošie pārsūtīšanas instrumenti var būt izmantoti šiem mērķiem. MPEG izstrādā specifikācijas MPEG-7 datu transportēšanai MPEG-2 sistēmās, kā arī ar MPEG-4 saturu. MPEG-7 apraksta transportēšana citās sistēmās (piemērām analogos pārsūtīšanas sistēmās) var būt līdzīgi izstrādāta dažādās organizācijās.

MPEG-7 datu transportēšana MPEG-2 sistēmās būs izdarīta atbilstībā ar jaunu izmaiņu MPEG-2 sistēmu specifikācijā. Šī izmaiņa sniedz dažādas meta datu nešanas iespējas: sinhronais transports sniegts ar meta datu nešanu Programmas Elementārajās Straumēs (PES) paketēs. Asinhronais transports bez karuseļa tipa sniegšanas ir nodrošināta ar meta datu sekcijām, tajā laikā kad DSM- CC instrumenti var sniegt karuseļa sniegšanas mehānismus, ar vai bez struktūras.

MPEG-7 datu pārsūtīšana MPEG-4 saturā ir darīta ar MPEG-7 datu apskatīšanu kā MPEG-4 elementāru straumju specifisku šķiru. Elementāra straumes identifikācija MPEG-7 datiem ir jau sniegta MPEG-4 specifikācijā un konfigurācijas informācija šādai straumei jābūt pievienotai MPEG-4 on-going izmaiņai. No šīs izmaiņas dabūs MPEG-7 datu transportu populāros pārsūtīšanas mehānismos jau esošos MPEG-4, tādi kā IP tīkli un faili izmantojoši MP4 failu formātu.

## IV. Nobeigums

Augstāk aprakstītie instrumenti satur lielāku daļu no MPEG-7 sistēmu funkcionalitātes un ļauj nepārvaramu MPEG-7 multimediju programmu attīstību. Viņi ir sniegti ar esošo MPEG-7 standartu, kurš būs pār internacionālo standartu 2001. gada septembrī.

Viņi var būt apkopotī sekojoši:

- 1) Multimediju aprakstītāju pārsūtīšana, sinhronizācija un vadīšana izmantojot pārsūtīšanas un glabāšanas protokolu dažādību;
- 2) Dinamisku multimediju aprakstītāju reprezentācija ar aprakstītāju un shēmu struktūras iekapsulēšanas komandās palīdzību;
- 3) Tekstuāla un bināro aprakstītāju reprezentācija.

Tehnoloģijas, apskatītas standartizācijai MPEG-7 nav vienā gatavības līmenī. Tādēļ, kā MPEG-4, MPEG-7 organizēja viņu specifikāciju vairākās fāzēs, sauktās par versijām. Jaunas versijas pabeidz esošas standartizācijas aprīkojumu ar jauniem instrumentiem un jaunām funkcijām. Viņi neaizvieto pagājušās versijas instrumentus. Šis papīrs aprakstīja MPEG-7 sistēmas 1. versijas aprīkojumu.