NOKIA
Connecting People

FlexCode

Speech and audio coding with flexible geometric structures?

Adriana Vasilache
Flexcode Seminar
Jan 10, 2008

Flexible geometric structures

- Flexible
  - Definition: adaptable or variable
- Geometric structures
  - Lattices

Lattice definition

- Algebraically, an *n*-dimensional lattice Λ is a set of real vectors whose coordinates are integers in a given basis $\{\mathrm{b}_i \in \mathrm{R}^n\}_{i=\overline{1,n}}$.

$$\Lambda = \left\{ \mathrm{v} \in \mathrm{R}^n \mid \mathrm{v} = \sum_{i=1}^{n} \alpha_i \mathrm{b}_i, \alpha_i \in Z \right\}$$

- Geometrically, a lattice is an infinite regular array of points which uniformly fills the *n*-dimensional space.

Lattice advantages

- Parameters: scale and support region.
- There are efficient algorithms for finding nearest neighbors, indexing, and lossless coding lattice quantizers, which makes them attractive in some applications such as audio coding.
- Good theoretical approximations for performance – special case of high rate theory
- Approximately optimal if used with entropy coding: constant quantizer point density functions over finite volume regions.

Lattice terminology

- Lattice truncation

$$\overline{\Lambda} = \{ \mathrm{x} \in \Lambda \mid N(\mathrm{x}) \le K \}$$

- Lattice shell

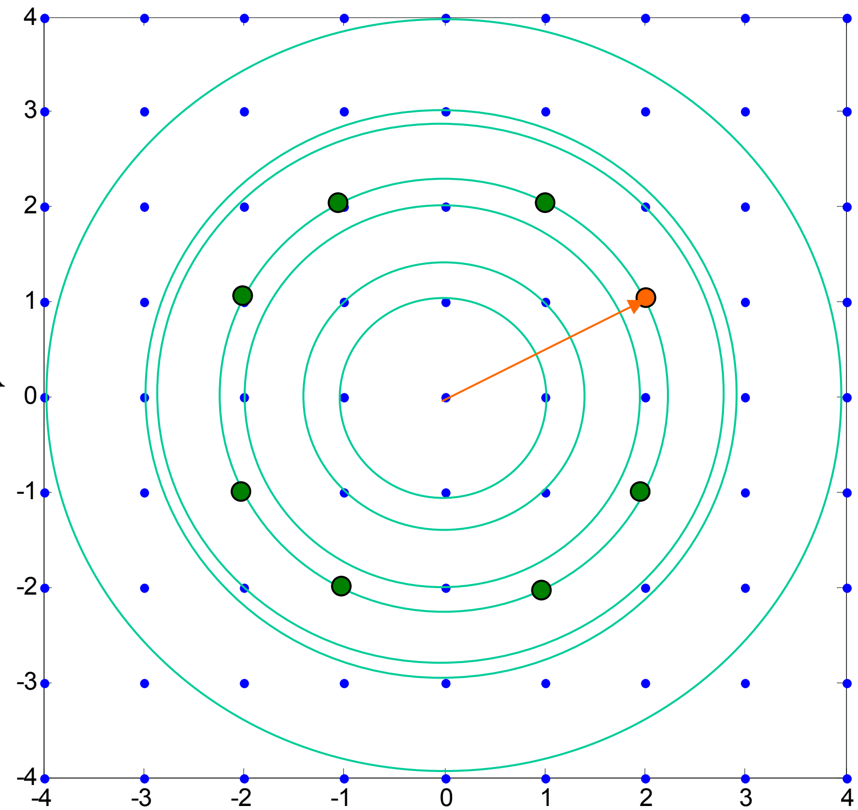
$$\overline{S_K} = \{ \mathrm{x} \in \Lambda \mid N(\mathrm{x}) = K \}$$

- Lattice leader

$$v = (v_m, \ldots, v_m, \ldots, v_i, \ldots v_i, \ldots v_1, \ldots, v_1) \in \Lambda$$

$$0 < v_1 < \ldots < v_i < \ldots < v_m$$

- Leader class



(2 1)
(-2 1)
(2 -1)
(-2 -1)
(1 2)
(-1 2)
(1 -2)
(-1 -2)

FlexCode

Presentation topics

NOKIA Connecting People

- Speech and audio coding with geometric structures
- Toward flexibility
  - Lattice codevectors indexing
    - Indexing on leaders (GG truncation)
    - Indexing on shells (spherical, pyramidal, rectangular truncation)
    - Index factorization
  - Entropy coding
    - Entropy coding of lattice codevectors using codevectors partitioning
    - Canonical Huffman encoding of lattice codevectors

Speech and audio coding

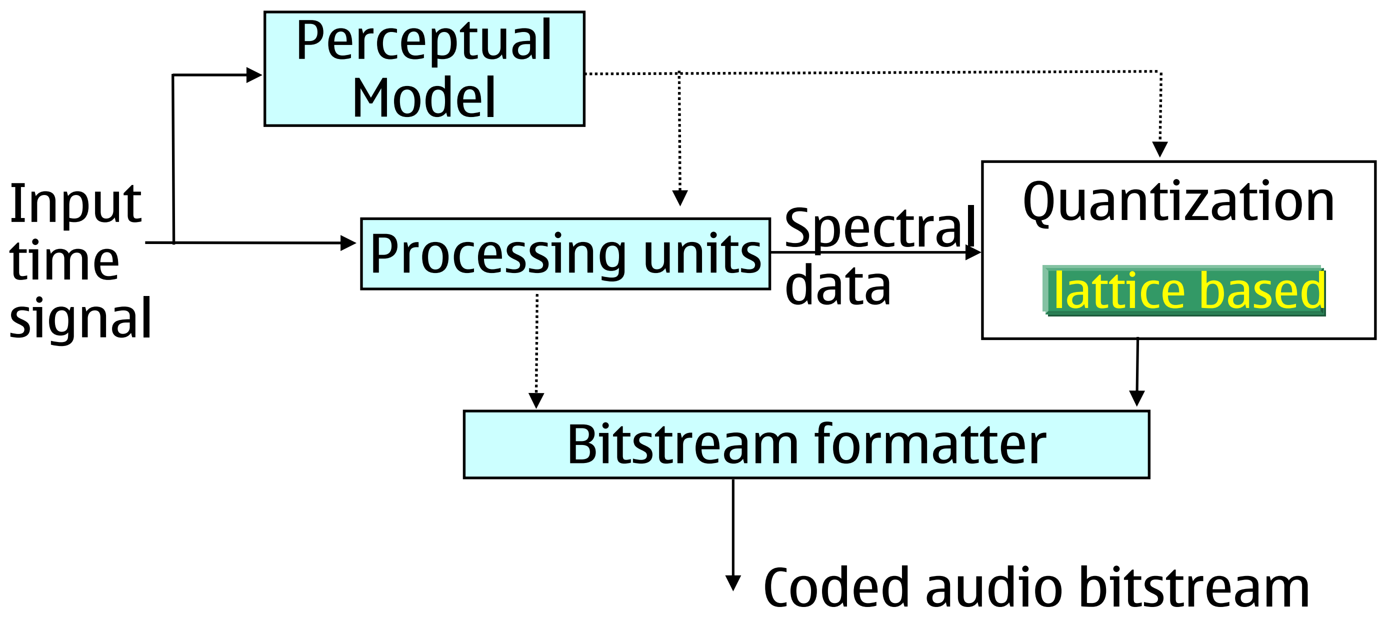
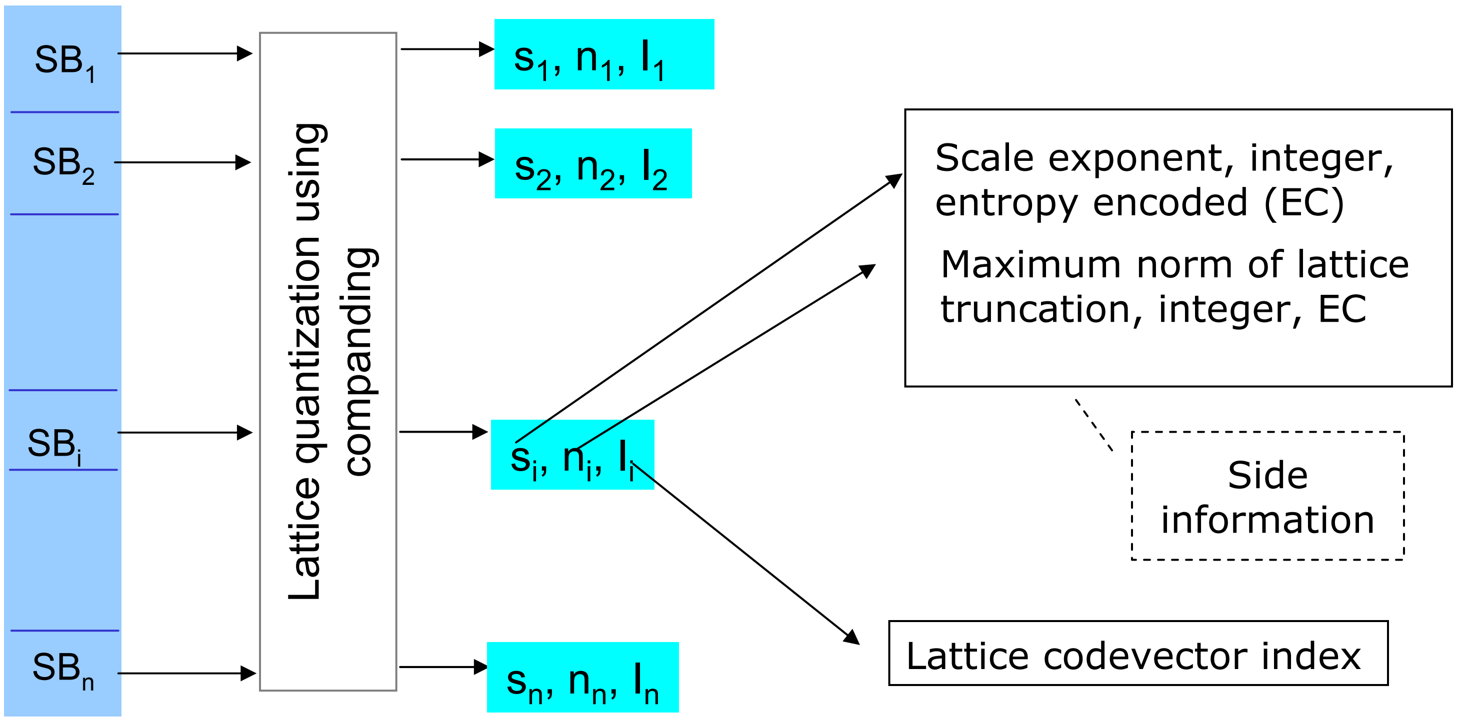


Figure 1. Block diagram of the general audio encoder

Spectral quantization



Flexibility?

- Inflexible indexing function
- Inflexible bit-allocation

... towards flexibility

Indexing functions

- Indexing on leaders (GG truncation)
  - Generalized indexing
- Indexing on shells (spherical, pyramidal, rectangular truncation)
  - Index factorization

Indexing on leaders (IL)

- The lattice is defined as a union of leader classes.
- The lattice points are enumerated within the leader class they belong to.
- Each leader class has an offset index.
- Enumeration of the unsigned vectors followed by the sign enumeration.

IL – lexicographic algorithms

- The vector $(x_1,\ldots,x_n)$ precedes lexicographically $(y_1,\ldots,y_n)$ if $\exists j$ such that $x_j < y_j$ and $\forall i < j, x_j = y_j$.
- The lexicographical enumeration can be based on:

$$\binom{n}{n_1 \ldots n_m} = \binom{n-1}{n_1 - 1 \ldots n_m} + \ldots + \binom{n-1}{n_1 \ldots n_m - 1}$$

v_i appears n_i times in the leader vector.

Binomial indexing

- Binomial indexing: it counts in how many ways *n1* values *v1* can be put on n positions, then in how many ways *n2* values can be put on *n-n1* positions and so on.

$$\binom{n}{n_1 \dots n_m} = \binom{n}{n_1}\binom{n-n_1}{n_2}\dots\binom{n-\sum_{i=1}^{m-1} n_i}{n_m}$$

FlexCode

IL - example

NOKIA Connecting People

Leader vector (3 1 0 0)

Index	Binomial				Lexicographic			
000000	3	1	0	0	0	0	1	3
000001	3	-1	0	0	0	0	1	-3
000010	-3	1	0	0	0	0	-1	3
000011	-3	-1	0	0	0	0	-1	-3
000100	3	0	1	0	0	0	3	1
001000	3	0	0	1	0	1	0	3
001100	1	3	0	0	0	1	3	0
010000	0	3	1	0	0	3	0	1
010100	0	3	0	1	0	3	1	0
011000	1	0	3	0	1	0	0	3
011100	0	1	3	0	1	0	3	0
100000	0	0	3	1	1	3	0	0
100100	1	0	0	3	3	0	0	1
101000	0	1	0	3	3	0	1	0
101100	0	0	1	3	3	1	0	0

Binomial indexing - example

Index	Binomial 1				Binomial 2			
0	2	1	1	0	2	1	1	0
1	1	2	1	0	2	1	0	1
2	1	1	2	0	2	0	1	1
3	1	1	0	2	1	2	1	0
4	2	1	0	1	1	2	0	1
5	1	2	0	1	0	2	1	1
6	1	0	2	1	1	1	2	0
7	1	0	1	2	1	0	2	1
8	2	0	1	1	0	1	2	1
9	0	2	1	1	1	1	0	2
10	0	1	2	1	1	0	1	2
11	0	1	1	2	0	1	1	2

Precedence sequence of vectors obtained from the permutation of (2 1 1 0) using the two variants of the binomial indexing.

IL – generalized indexing

- Both lexicographical and binomial indexing rely on an assumed order relation between the vector component values.
- Assuming different order relations between the leader vector component values generates different indexing functions.

Generalized lexicographic indexing

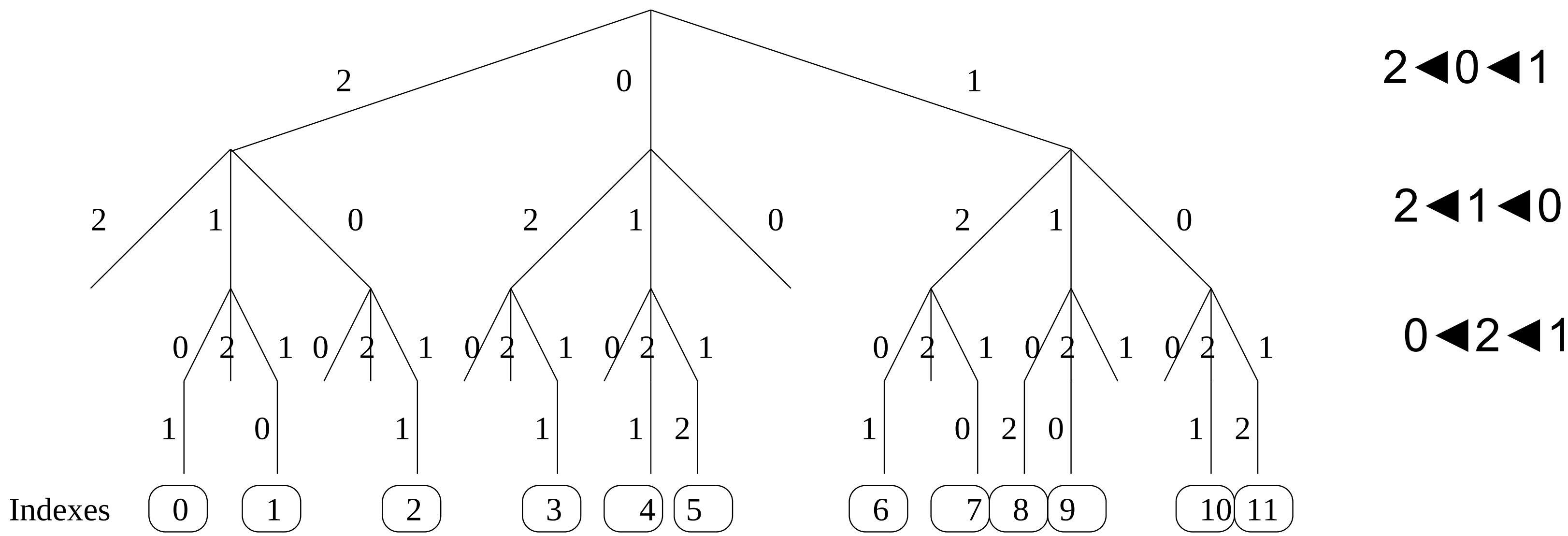


Figure 1. Generalized lexicographical indexing of the 12 codevectors belonging to the unsigned leader class (2 1 1 0). The framed index at a leaf is associated to the codevector having as entries the labels read from the root to the leaf.

E.g. the index 5 is associated to the codevector (0 1 1 2). Only leaves at the maximum depth correspond to codevectors.

FlexCode IL – generalized binomial indexing

NOKIA Connecting People

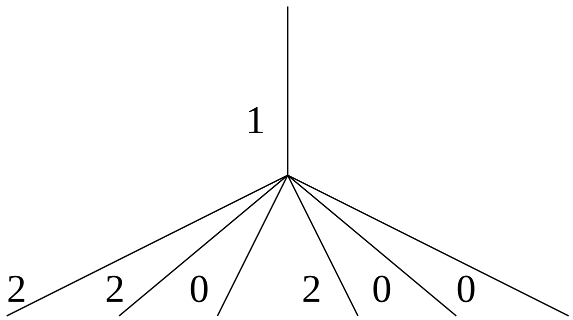


Figure 2. Example of graph of the generalized binomial indexing – second variant - for the vector (2 1 1 0).

Index				
0	1	1	2	0
1	1	1	0	2
2	1	2	1	0
3	1	0	1	2
4	1	0	2	1
5	1	2	0	1
6	2	1	1	0
7	0	1	1	2
8	0	1	2	1
9	2	1	0	1
10	0	2	1	1
11	0	2	1	1

Example of binomial indexing -second variant- of the vectors obtained from the permutation of (2 1 1 0).

IL- indexing function optimization

- The indexing function can be optimized with respect to a given criterion as a function of the quasi order relation at different levels in the indexing tree/graph.
- The optimization
  - Greedy algorithm (generalized lexicographical and binomial)
  - Iterative algorithm (generalized lexicographical)

IL - applications

- The parameterization of the indexing function allows its optimization with respect to a given criterion
- E.g.: channel distortion
  - Close to practical channel distortion lower bound for leader classes
  - Practical improvements of 10% for lattice truncations; limited performance by the leader class separation

FlexCode

Indexing on shells (IS)

NOKIA Connecting People

- The lattice is defined as a union of shells (under the given norm).
- The lattice points are enumerated within the shells they belong to.
- Product codes type of indexes.
- Each shell has an offset index.

IS – product code indexing

- Truncations (spherical, pyramidal, rectangular) of Zn lattices and cosets of Zn
- E.g. for a rectangular shell of Zn
  - $\max\{|x_i|\}=K$, $(x_1,\ldots,x_n)$ lattice point
  - $\exists\, 0 < j \le n$ such that $x_j = K$
  - Product code
    - Number of significant components
    - Number of maximum valued components
    - Position of maximum valued components
    - Values of significant non-maximum components
    - Position of significant non maximum values
    - Sign of significant components

IS – product code indexing –ctnd.

FlexCode — NOKIA Connecting People

Example Indexing of a 3D vector having maximum norm equal to 2.

n=3, K=2

		Vectors	Indexes
S=3	2 2 2 M=3	±2 ±2 ±2	0,…, 7,
	2 2 1 M=2	±2 ±2 ±1	8,…,15,
		±2 ±1 ±2	16,…,23,
		±1 ±2 ±2	24,…,31,
	2 1 1 M=1	±2 ±1 ±1	32,…,39,
		±1 ±2 ±1	40,…,47,
		±1 ±2 ±2	48,…,55,
S=2	2 2 0 M=2	±2 ±2 0	56,…,59,
		±2 0 ±2	60,…,63,
		0 ±2 ±2	64,…,67,
	2 1 0 M=1	±2 ±1 0	68,…,71,
		…	…
S=1	2 0 0 M=1		

FlexCode

IS - applications

NOKIA Connecting People

- Index factorization – beneficial for channel error resilience
  - Demo
- Index factorization
  - Enables scalable indexing because only parts of the index can be decoded to an approximation of the initial codevector.

Entropy coding of lattice codevectors

- EC of lattice codevectors using partitioning on
  - shells
  - leader classes
- EC using canonical Huffman coding

FlexCode EC using lattice codevectors partitioning

NOKIA Connecting People

- The index of shell/leader class is entropy coded.
- The index of the codevector within the shell/leader class is sent on $\lceil \log_2 C_i \rceil$ bits, where *Ci* is the cardinality of the leader class/shell.

EC using canonical Huffman coding

- Canonical Huffman
  - To increasing order of symbols probability it corresponds decreasing code lengths;
  - For the same code length the symbols have consecutive binary numbers as code words;
  - Only a table of length equal to the number of different code lengths in the code is needed.
- The symmetry of the source allows the assumption of constant probability on the shells.
- The symmetry of the source allows the assumption of constant probability with a leader class.
- The shells/leader classes can be ordered such that the codevector probability are decreasing (this is equivalent to setting the offset values for each shell/leader class).

FlexCode

EC of lattice codevectors

NOKIA Connecting People

	EC on shells			EC on leader classes		
lead	L_{HR}	L_{idxR}	L_R	L_{HL}	L_{idxL}	L_L
3	1.00	7.98	8.98	1.23	7.31	**8.54**
8	1.08	12.59	13.67	2.02	11.10	**13.20**
9	1.51	13.00	14.51	2.41	11.42	**13.83**
12	1.52	14.81	16.33	2.83	12.94	**15.77**
13	1.43	15.19	16.62	2.92	13.25	**16.17**
20	1.54	15.06	16.60	3.23	13.16	**16.39**
21	1.74	15.73	17.47	3.44	13.60	**17.04**

Table 1. Code length distribution between the leader class index and the position of codevectors within a class for the entropy coding with partitioning on norms (R) and on leaders (L) for Laplacian data.

lead.	H	CHC L_H	ECR L_R	ECL L_L	Enum. coding
3	7.59	**7.65**	8.98	8.54	8
8	11.53	**11.85**	13.67	13.20	13
9	13.38	**13.41**	14.51	13.83	15
12	15.07	**15.10**	16.33	15.77	16
20	15.56	**15.59**	16.60	16.39	17
21	16.18	**16.20**	17.47	17.04	18

Table 2. Comparison of canonical Huffman coding (CHC) with entropy coding with partitioning of lattice codevectors on norms (ECR) and on leaders (ECL) for different sizes of a pyramidal LVQ on Laplacian data. The entropy H is also given for every considered case.

FlexCode

EC on lattice codevectors - cntd

NOKIA Connecting People

	EC on shells			EC on leader classes		
lead.	L_{HR}	L_{idxR}	L_R	L_{HL}	L_{idxL}	L_L
3	1.01	7.94	8.95	1.03	7.88	**8.92**
4	1.10	11.58	12.68	1.12	11.53	**12.65**
7	1.31	13.37	14.68	1.69	12.96	**14.65**
8	1.76	14.43	**16.19**	1.90	14.31	16.21

Table 3. Code length distribution between the class index and the position of codevectors within a class for the entropy coding with partitioning on norms (R) and on leaders (L) for Gaussian data.

lead.	H	CHC L_H	ECR L_R	ECL L_L	Enum. coding
3	7.57	**7.64**	8.95	8.92	8
4	11.77	**11.82**	12.68	12.65	12
7	14.20	**14.27**	14.68	14.65	15
8	15.66	**15.72**	16.19	16.21	16

Table 4. Comparison of canonical Huffman coding (CHC) with entropy coding with partitioning of lattice codevectors on norms (ECR) and on leaders (ECL) for different sizes of a spherical LVQ on Gaussian data. For the two latter methods the best achievable average code-length is considered. The entropy H is also given for every considered case.

EC on lattice cv. – LSF quantization

- Narrow band speech coding
- 10 dimensional LSF vectors
- For a SD of 0.98dB

– Zero order entropy	19.08bits
– CHC length	19.11bits
– EC on norms	19.75bits
– EC on leaders	19.54bits
– Enum. encoding	20.00bits

FlexCode

NOKIA Connecting People

Speech and audio coding with flexible geometric structures!