Efficient Storing of Multidimensional Histograms Using Advanced Compression Techniques

V. Matoušek *, 1, 3, M. Morháč 1, 2, J. Kliman 1, 2,
I. Turzo 1, 3, L. Krupa 1, 2 and M. Jandel 1, 2

1 Institute of Physics, Slovak Academy of Sciences, Bratislava, Slovakia
2 Flerov Laboratory of Nuclear Reactions, JINR, Dubna, Russia
3 Laboratory of High Energies, JINR, Dubna, Russia

* E-mail: matousek@savba.sk

ACAT - 2002 June 24 – 28, 2002
DATA STORING

Data types

- list of events
- histograms (spectra) 1, 2, 3, 4-dimensional

Event formats

Supported event formats for off-line analysis

- Acq format
  
  Event variable 1
  Event variable 2
  .
  .
  Event variable N
  End of event (-1)

- Gammasphere format

  It is more complicated and is available on WEB

Practical problems

- enormous amount of information stored as list of events on Exabyte tapes

- to read one tape (4 GB) it takes about 2 hours

- in one experiment 20, 30 or more tapes of events are usually collected

- only to read data from one experiment it takes sometimes several days

- to handle data and to create different slices it is necessary that the spectra be available to the experimenter in interactive way

- therefore it is desirable to work with histograms – spectra, preferably with multidimensional spectra and then to slice these spectra at specified energies to obtain one dimensional slices
- today’s resolution of ADCs used in GAMMASPHERE experiments is 16k
- let us assume that one channel in a multidimensional spectrum is stored in 2 bytes. Then the storage requirements for multidimensional spectra are as follows:

\[
\begin{align*}
2\text{-param.}- & \quad 2 \times 2^{14} \times 2^{14} = 2^{29} \text{B} = 512 \text{MB} \\
3\text{-param.}- & \quad 2 \times 2^{3 \times 14} = 2^{43} \text{B} = 8 \text{TB} \\
4\text{-param.}- & \quad 2 \times 2^{4 \times 14} = 2^{57} \text{B} = 128 \text{PB} (\text{peta}) \\
5\text{-param.}- & \quad 2 \times 2^{5 \times 14} = 2^{71} \text{B}
\end{align*}
\]

- to fit the capacity of computer memory it is unavoidable to decrease the size of multidimensional array, i.e. to compress spectra in a way

**Compression**

- the aim is to decrease the size of data to the size of memory while preserving as much information as possible.
  
a. **binning** – channels are summed together – decreases resolution - loss of information
  
b. **natural compression** – utilizes special properties of data. In the case of multidimensional $\gamma$-ray spectra from GAMMASPHERE we can utilize the property of natural symmetry of the data. It holds

2-parameter spectra
\[
E(\gamma_1, \gamma_2) = E(\gamma_2, \gamma_1)
\]
- it decreases space by factor of 2

3-parameter spectra
\[
E(\gamma_1, \gamma_2, \gamma_3) = E(\gamma_1, \gamma_3, \gamma_2) = E(\gamma_2, \gamma_3, \gamma_1) = E(\gamma_2, \gamma_1, \gamma_3) = E(\gamma_3, \gamma_1, \gamma_2) = E(\gamma_3, \gamma_2, \gamma_1)
\]
- decreases space by factor of 6

k-parameter spectra
- in general by utilizing the property of symmetry the k-dimensional space can be decreased by factor of k!
• memory requirements for symmetrical histograms

\[
N = 2^{14}
\]

\[
2D - (N^2 + N)/2 \approx 256\text{MB}
\]

\[
3D - (N^3 + 3N^2 + 2N)/6 \approx 1.25\text{TB}
\]

\[
4D - (N^4 + 6N^3 + 11N^2 + 6N)/24 \approx 5\text{PB}
\]

\[
5D - (N^5 + 10N^4 + 35N^3 + 50N^2 + 24N)/120 \approx 16\text{EB}
\]

c. compression using adaptive transforms – off-line block data method

• adaptive Walsh, Fourier, Cosine transforms

• principle consists in modification of the transform kernel to reference vectors so that the distortion is as small as possible

• for reference vectors we have used integral spectra

Examples

d. on-line compression method using Walsh adaptive transform

• we have to store 2, 3, 4- dimensional histograms in memory

• original uncompressed size of spectrum very frequently exceeds the memory capacity available

• therefore it is impossible to store a multidimensional histogram in its original form, then transform it as block data and subsequently store only a part of transformed spectrum
For $N = 4$ and bit-reversed output vector the Fourier transform matrices are

$$
F_C = \frac{1}{2} \begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & -1 & 1 & -1 \\
1 & W & -1 & -W \\
1 & -W & -1 & W
\end{bmatrix};
$$

$$
F_C^{-1} = \frac{1}{2} \begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & -1 & -W & W \\
1 & 1 & -1 & -1 \\
1 & -1 & W & -W
\end{bmatrix}.
$$

where $W = e^{-j\frac{2\pi}{N}}$.

Let us generalize this transform and replace the multiplication coefficients 1 in the signal flow graph of the FFT by $a, b, c, d$

Signal flow graphs of the adaptive direct (a) and inverse (b) fast Fourier transforms.
The direct and inverse adaptive Fourier transform matrices for $N = 4$ are

$$\mathbf{F}_A = \begin{bmatrix} a_0 a_2 & a_1 c_2 & c_0 a_2 & c_1 c_2 \\ a_0 b_2 & -a_1 d_2 & c_0 b_2 & -c_1 d_2 \\ b_0 a_3 & b_1 c_3 w & -d_0 a_3 & -d_1 c_3 w \\ b_0 b_3 & -b_1 d_3 w & -d_0 b_3 & d_1 d_3 w \end{bmatrix} ,$$

$$\mathbf{F}_A^{-1} = \begin{bmatrix} a_0 a_2 & b_2 a_0 & a_3 b_0 & b_3 b_0 \\ a_1 c_2 & -d_2 a_1 & -c_3 b_1 w & d_3 b_1 w \\ a_2 c_0 & b_2 c_0 & -a_3 d_0 & -b_3 d_0 \\ c_2 c_1 & -d_2 c_1 & -c_3 d_1 w & -d_3 d_1 w \end{bmatrix} .$$

The matrices $\mathbf{F}_A$ and $\mathbf{F}_A^{-1}$ must be orthonormal and therefore it must hold that

$$\mathbf{F}_A \cdot \mathbf{F}_A^{-1} = \mathbf{E} ,$$

where $\mathbf{E}$ denotes the unit matrix.

The conditions are satisfied if and only if

$$a_i b_i - c_i d_i = 0$$

$$a_i^2 + c_i^2 = 1$$

$$b_i^2 + d_i^2 = 1 ,$$

and thus

$$b_i = \pm c_i$$

$$a_i = \mp d_i.$$
For each butterfly of the signal flow graph, it must hold that the input energy must be equal to the output energy. One of the ways how to obtain the best compression ratio is to concentrate the energy from the input into the first node in the output so that

\[ 1x_i = \sqrt{x_i^2 + x_j^2}, \]
\[ 1x_j = 0. \]

Since the values \( 1x_2, 1x_3 \) are zeros the coefficients of the second butterfly in the second iteration step cannot be computed using this algorithm. Therefore we shall suppose that

\[ a_3 = b_3 = \frac{1}{\sqrt{2}}. \]

The calculation of coefficients is needed only in the butterflies not containing the complex coefficient \( w \). When considering the above equations with the positive sign then in general for one butterfly of the adaptive transform one can write

\[ ax_i + bx_j = y_i, \]
\[ bx_i - ax_j = y_j, \]

The solution for coefficients \( a_0, b_0, c_0, d_0 \) of the first butterfly in the first iteration step is

\[ a_0 = \frac{x_0}{\sqrt{x_0^2 + x_2^2}} \quad b_0 = \frac{x_2}{\sqrt{x_0^2 + x_2^2}}, \]
\[ d_0 = a_0 \quad c_0 = b_0. \]
Original two-dimensional signal (coincident nuclear spectrum)
Two-dimensional nuclear spectrum after the compression using classic Walsh-Hadamard transform with $CR = 64$
Two-dimensional nuclear spectrum after the compression using classic cosine transform with $CR = 64$
Two-dimensional nuclear spectrum after the compression using adaptive Walsh-Hadamard transform with $CR = 64$
Two-dimensional nuclear spectrum after the compression using adaptive cosine transform with $CR = 64$
Coefficients of the two-dimensional nuclear spectrum in the transform domain after transformation using classic Walsh-Hadamard transform.
Coefficients of the two-dimensional nuclear spectrum in the transform domain after transformation using adaptive Walsh-Hadamard transform
Since the energy in the transform domain of two-parameter (also multiparameter) spectra is concentrated around the null point of the coordinate system we shall define the energy packing efficiency (EPE). For two-dimensional case it is defined as a part of energy contained in the first $M \times M$ from $N \times N$ transformed coefficients.

$$EPE(M) = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{M-1} y_{i,j}^2}{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} y_{i,j}^2} \cdot 100 \,[\%].$$
EPE functions for the following transforms:
1 - classic Walsh-Hadamard transform,
2 - classic cosine transform,
3 - adaptive Walsh-Hadamard transform,
4 - adaptive cosine transform
Example of the signal flow graph for on-line compression
e. randomizing transformation mode compression

- the size of multidimensional arrays for higher dimensions (4,5) is very large

- the number of different descriptors, i.e. the values read out from ADCs which actually occur during an experiment is much smaller

- therefore the multidimensional space must be almost empty. Very large part of locations of multidimensional arrays are zeros or statistical fluctuations only.

- to preserve correspondence between original space and compressed array in this method of compression the descriptors (addresses in multidimensional arrays, e.g. values read out from ADCs) and counts are stored for each channel separately

- the descriptors are passed through a transformation

- it would be ideal if any distribution of descriptors would produce the uniform distribution throughout the memory

- however in practice there exists a possibility of more descriptors being transformed to the same address

- because of that in our algorithm this address is used to determine position where to start searching

- a list of d successive locations, where d is depth of searching are checked

- if a location within this depth is empty descriptor is written to this location and counts is set to 1

- if in a location the descriptor coincides with already recorded descriptor the counts in this location are incremented. Otherwise the event is ignored.

- we employ modular arithmetic. Let a is address in original space, then \( b = a^{-1} \pmod{M} \), where M is prime, gives pseudorandom distribution

- for 3D compression we have chosen M=601.
DATA STORING

Data types

- list of events
- histograms (spectra) 1, 2, 3, 4-dimensional

Event formats

Supported event formats for off-line analysis

- Acq format
  
  Event variable 1
  Event variable 2
  .
  .
  Event variable N
  End of event (-1)

- Gammasphere format
  
  It is more complicated and is available on WEB

Practical problems

- enormous amount of information stored as list of events on Exabyte tapes
- to read one tape (4 GB) it takes about 2 hours
- in one experiment 20, 30 or more tapes of events are usually collected
- only to read data from one experiment it takes sometimes several days
- to handle data and to create different slices it is necessary that the spectra be available to the experimenter in interactive way
- therefore it is desirable to work with histograms – spectra, preferably with multidimensional spectra and then to slice these spectra at specified energies to obtain one dimensional slices
Data organization in $\gamma$ - $\gamma$ coincidence spectra compression schemes from Gammasphere data
Example of 1-parameter slice from 3-parameter spectrum (solid line) and slice (dashed line) compressed using non-constant binning (Radware software package, $CR \approx 34000$)
Example of 1-parameter slice from 3-parameter spectrum (solid line) and slice (dashed line) compressed using our adaptive WHT transform ($CR \approx 750000$)
Example of 1-parameter slice from 4-parameter spectrum (solid line) and slice (dashed line) compressed using our adaptive WHT transform ($CR \approx 4.5 \times 10^9$)
Example of 1-parameter slice from 4-parameter spectrum (solid line) and slice (dashed line) compressed using address randomizing transform ($CR \approx 750 \times 10^8$)
List of some relevant publications:


