EFFICIENT ONE AND TWO DIMENSIONAL GOLD DECONVOLUTION AND ITS APPLICATION TO γ-RAY SPECTRA DECOMPOSITION

MIROSLAV MORHÁČ
Institute of Physics, Slovak Academy of Sciences, Dúbravska cesta 9
Bratislava, 842 28, Slovak republic

and

JÁN KLIMAN, VLADISLAV MATOUŠEK, MARTIN VESELSKÝ, IVAN TURZO
Institute of Physics, Slovak Academy of Sciences,

The use of deconvolution operation makes it possible to improve the resolution in γ-ray spectra. The paper describes a non-oscillating method of Gold deconvolution. The method is generalized for multidimensional spectra. A new optimized algorithm of one and two dimensional Gold deconvolution aimed to reduce number of computer operations has been derived in the paper.

1. Introduction

In past years deconvolution algorithms have found many applications in various domains of experimental science. They have been applied also to longstanding problem of determination of positions and intensities of γ rays multiplets [1]. From numerical point of view the deconvolution belongs to one of the most critical problems. It is so called ill-posed problem, which means that many different functions solve convolution equation within error bounds of the experiment [2]. The estimates of solution are extremely sensitive to errors in the measured data. It implies that a suitable method of regularization must be employed [3].

Methods of regularization can be classified from different aspects:

- quality of deconvoluted signal (smoothness, positive solution, oscillations etc.)
- computational complexity
- convergence speed.

In [4] to solve the problems of deconvolution of experimental spectra of positron annihilation the Van Cittert and Gold methods [10] have been employed. These methods can be successfully applied also for the decomposition of γ ray multiplets. In particular the Gold method gives very good results. Its basic property is that the solution is always non-negative. When processing spectra this is very important property as calculated spectrum represents histogram where negative values are senseless.

On the other hand from computational point of view the Gold deconvolution is very time consuming operation. It requires a great number of numerical operations as its convergence speed is rather slow. The convergence properties are studied in [4]. This problem is becoming relevant for big sizes of spectra and for two and multidimensional spectra where the number of operations grows exponentially with the sizes. Therefore the use of the method for two dimensional spectra, also of common sizes, is on small computer practically impossible. The implementation of the method requires optimization from the point of view of time (redundant operations may be omitted) and of memory storage (the data during computation may be stored in memory more efficiently).

Hence the aim of the paper is to derive optimal efficient algorithm of non-oscillating one and two dimensional Gold deconvolution and to apply it to one and two dimensional γ ray spectra with the purpose to improve resolution and thus to decompose multiplets.

2. Multidimensional Gold deconvolution

The relation between the input value of a linear time-invariant system and its output value can be described by convolution integral equation. Subsequently for one, two, $\ldots, n$-dimensional continuous linear system one can write

$$y_1(t_1) = \int_{-\infty}^{+\infty} h_1(t_1-\tau)x_1(\tau)\,d\tau \tag{1}$$

$$y_2(t_1,t_2) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} h_2(t_1-\tau_1,\, t_2-\tau_2)x_2(\tau_1,\tau_2)\,d\tau_1 d\tau_2 \tag{2}$$

$$\vdots$$

$$y_n(t_1,t_2,\ldots,t_n) =$$

$$= \underbrace{\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\cdots\int_{-\infty}^{+\infty}}_{\mathrm{n-fold}} h_n(t_1-\tau_1,t_2-\tau_2,\ldots,t_n-\tau_n)x_n(\tau_1,\tau_2,\ldots\tau_n)d\tau_1 d\tau_2\ldots d\tau_n \tag{3}$$

where h_i, $i \in <1, n>$ is impulse response function of i-dimensional convolution system, y_i and x_i are output and input i-dimensional signals, respectively. Analogously for discrete signals the Eqs. (1)-(3) will become

$$\begin{aligned}
y_1(i) &= \sum_{k=0}^{N_1-1} h_1(i-k)x_1(k); && (4)\\
&\qquad i = 0, 1, \ldots, 2N_1 - 2 \\
y_2(i_1, i_2) &= \sum_{k_1=0}^{N_1-1}\sum_{k_2=0}^{N_2-1} h_2(i_1-k_1, i_2-k_2)x_2(k_1,k_2); && (5)\\
&\qquad i_1 \in <0, 2N_1-2>, i_2 \in <0, 2N_2-2> \\
&\vdots \\
y_n(i_1, i_2, \ldots, i_n) &= \sum_{k_1=0}^{N_1-1}\sum_{k_2=0}^{N_2-1}\cdots\sum_{k_n=0}^{N_n-1} h_n(i_1-k_1, \ldots, i_n-k_n)x_n(k_1, \ldots, k_n); \\
&\qquad i_j \in <0, 2N_j-2>, j \in <1, n>. && (6)
\end{aligned}$$

We shall assume the knowledge of the impulse response function (resolution instrumental function) and the measured output values y. Based on this, in deconvolution procedure, we are looking for the solution of the corresponding system of linear equation Eq. (6).

The impulse response function of j-dimensional system is supposed to have finite length in all dimensions. It implies that $h_j(i_1, i_2, \ldots, i_j) = 0$ for $i_k < 0$ and for $i_k \geq N_k, k \in <1, j>$. Then for one dimensional system Eq. (4) can be written in matrix form

$$\begin{bmatrix} y_1(0) \\ y_1(1) \\ y_1(2) \\ \vdots \\ y_1(2N_1-2) \end{bmatrix} = \begin{bmatrix} h_1(0) & 0 & 0 & \cdots \\ h_1(1) & h_1(0) & 0 & \cdots \\ h_1(2) & h_1(1) & h_1(0) & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ h_1(N_1-1) & h_1(N_1-2) & h_1(N_1-3) & \cdots \\ 0 & h_1(N_1-1) & h_1(N_1-2) & \cdots \\ 0 & 0 & h_1(N_1-1) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ \vdots \\ x_1(N_1-1) \end{bmatrix}. \tag{7}$$

For two dimensional convolution system a similar procedure can be used. Let us assume that we denote the shifted i-th column of the two dimensional response as submatrix

$$h_2^{(1)}(i_2) = \begin{bmatrix} h_2(0,i_2) & 0 & 0 & \cdots \\ h_2(1,i_2) & h_2(0,i_2) & 0 & \cdots \\ h_2(2,i_2) & h_2(1,i_2) & h_2(0,i_2) & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ h_2(N_1-1,i_2) & h_2(N_1-2,i_2) & h_2(N_1-3,i_2) & \cdots \\ 0 & h_2(N_1-1,i_2) & h_2(N_1-2,i_2) & \cdots \\ 0 & 0 & h_2(N_1-1,i_2) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{8}$$

and the i-th columns of the input and output matrices, respectively, as vectors

$$\mathbf{x}_2^{(1)}(i_2) = [x_2(0,i_2), x_2(1,i_2), \ldots, x_2(N_1-1,i_2)]^T \tag{9}$$

$$\mathbf{y}_2^{(1)}(i_2) = [y_2(0,i_2), y_2(1,i_2), \ldots, y_2(N_1-1,i_2)]^T \tag{10}$$

where $i \in < 0, N_2-1 >$. Then we have

$$\begin{bmatrix} \mathbf{y}_2^{(1)}(0) \\ \mathbf{y}_2^{(1)}(1) \\ \vdots \\ \mathbf{y}_2^{(1)}(2N_2-2) \end{bmatrix} = \\ = \begin{bmatrix} h_2^{(1)}(0) & 0 & 0 & \cdots \\ h_2^{(1)}(1) & h_2^{(1)}(0) & 0 & \cdots \\ h_2^{(1)}(2) & h_2^{(1)}(1) & h_2^{(1)}(0) & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ h_2^{(1)}(N_2-1) & h_2^{(1)}(N_2-2) & h_2^{(1)}(N_2-3) & \cdots \\ 0 & h_2^{(1)}(N_2-1) & h_2^{(1)}(N_2-2) & \cdots \\ 0 & 0 & h_2^{(1)}(N_2-1) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \mathbf{x}_2^{(1)}(0) \\ \mathbf{x}_2^{(1)}(1) \\ \vdots \\ \mathbf{x}_2^{(1)}(N_2-1) \end{bmatrix}. \tag{11}$$

The system matrix in Eq. (11) consists of shifted submatrices $h_2^{(1)}(i)$ given by Eq. (8).

We can continue in an analogous way for 3-rd, 4-th dimension up to n-dimensional convolution system. Let us denote the submatrix of j-th order of the n-dimensional response

$$h_n^{(j)}(i_{j+1},...,i_n) =$$

$$= \begin{bmatrix} h_k^{(j-1)}(0,i_{j+1},...,i_n) & 0 & 0 & \cdots \\ h_k^{(j-1)}(1,i_{j+1},...,i_n) & h_k^{(j-1)}(0,i_{j+1},...,i_n) & 0 & \cdots \\ h_k^{(j-1)}(2,i_{j+1},...,i_n) & h_k^{(j-1)}(1,i_{j+1},...,i_n) & h_k^{(j-1)}(0,i_{j+1},...,i_n) & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ h_k^{(j-1)}(N_j-1,i_{j+1},...,i_n) & h_k^{(j-1)}(N_j-2,i_{j+1},...,i_n) & h_k^{(j-1)}(N_j-3,i_{j+1},...,i_n) & \cdots \\ 0 & h_k^{(j-1)}(N_j-1,i_{j+1},...,i_n) & h_k^{(j-1)}(N_j-2,i_{j+1},...,i_n) & \cdots \\ 0 & 0 & h_k^{(j-1)}(N_j-1,i_{j+1},...,i_n) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{12}$$

and the submatrices of the j-th order of the input and the output signals, respectively, as

$$\begin{aligned} x_n^{(j)}(i_{j+1},\ldots,i_n) &= [x^{(j-1)}(0,i_{j+1},\ldots,i_n), x^{(j-1)}(1,i_{j+1},\ldots,i_n),\ldots \\ &\qquad \ldots, x^{(j-1)}(N_j-1,i_{j+1},\ldots,i_n),]^T \end{aligned} \tag{13}$$

$$\begin{aligned} y_n^{(j)}(i_{j+1},\ldots,i_n) &= [y^{(j-1)}(0,i_{j+1},\ldots,i_n), y^{(j-1)}(1,i_{j+1},\ldots,i_n),\ldots \\ &\qquad \ldots, y^{(j-1)}(N_j-1,i_{j+1},\ldots,i_n),]^T. \end{aligned} \tag{14}$$

One can observe that the submatrix of j-th order in Eq. (12) consists of shifted submatrices of $(j-1)$-st order. Taking into account Eqs. (13), (14) finally one can write

$$\begin{bmatrix} \mathbf{y}_n^{(n-1)}(0) \\ \mathbf{y}_n^{(n-1)}(1) \\ \vdots \\ \mathbf{y}_n^{(n-1)}(2N_n-2) \end{bmatrix} =$$

$$= \begin{bmatrix} h_n^{(n-1)}(0) & 0 & 0 & \cdots \\ h_n^{(n-1)}(1) & h_n^{(n-1)}(0) & 0 & \cdots \\ h_n^{(n-1)}(2) & h_n^{(n-1)}(1) & h_n^{(n-1)}(0) & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ h_n^{(n-1)}(N_n-1) & h_n^{(n-1)}(N_n-2) & h_n^{(n-1)}(N_n-3) & \cdots \\ 0 & h_n^{(n-1)}(N_n-1) & h_n^{(n-1)}(N_n-2) & \cdots \\ 0 & 0 & h_n^{(n-1)}(N_n-1) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \mathbf{x}_n^{(n-1)}(0) \\ \mathbf{x}_n^{(n-1)}(1) \\ \vdots \\ \mathbf{x}_n^{(n-1)}(N_n-1) \end{bmatrix} \tag{15}$$

or

$$\mathbf{y}_n^{(n)} = h_n^{(n)} \cdot \mathbf{x}_n^{(n)} \quad . \tag{16}$$

The dimension of the matrix $h_n^{(n)}$ equals

$$\prod_{i=1}^{n} (2N_i - 1) \cdot \prod_{j=1}^{n} N_j . \tag{17}$$

One can conclude that using above given procedure the convolution system of any dimension can be converted to the product of matrix with vector given by Eq. (15). However from Eq. (17) it is observable that the size of the matrix for the multidimensional convolution system would be enormous.

The Gold algorithm for the deconvolution of one dimensional spectra (non-optimized) is described in [4]. As the multidimensional convolution system can be expressed in the form of linear equations according to Eq. (16) the algorithm of multidimensional Gold deconvolution is analogous. Therefore we present only its final form. For details we refer to [4]. We calculate

$$C = {h_n^{(n)}}^T \cdot h_n^{(n)} \cdot {h_n^{(n)}}^T \cdot h_n^{(n)} \tag{18}$$

and

$$\mathbf{y}' = {h_n^{(n)}}^T \cdot h_n^{(n)} \cdot {h_n^{(n)}}^T \cdot \mathbf{y}_n^{(n)}, \tag{19}$$

where the matrix $h_n^{(n)}$ and the vector $\mathbf{y}_n^{(n)}$ are defined by Eqs. (15), (16). Then the Gold algorithm of multidimensional deconvolution is

$$^{k+1}\mathbf{x}_n^{(n)}(i) = \frac{\mathbf{y}'(i)}{\sum_{m=0}^{M-1} C(i,m) \cdot ^k x_n^{(n)}(m)} \cdot {}^k x_n^{(n)}(i), \tag{20}$$

where k represents number of iterations, $i \in < 0, M-1 >, M = \prod_{j=n}^{n} N_j$. One can easily imagine that for number of iterations $\approx$1000 the realization of the algorithm (20) with respect to Eqs. (15), (16), (18), (19) requires enormous number of operations. From the computational point of view the algorithm is becoming non-realizable in reasonable time even for small two dimensional convolution systems. On these grounds the optimization of the Gold deconvolution algorithm is unavoidable.

3. Optimization of the Gold deconvolution algorithm

Let us start with one dimensional deconvolution. In the previous chapter we supposed the length of the impulse response to be equal to N, which is also the length of the sought vector $\mathbf{x}$. However in γ ray spectroscopy the number of channels of one peak (impulse response) with non-zero counts is of order of 10 or less. Outside this interval the impulse response counts vanish to zero. Therefore the length of impulse response is much more less than the length of both input and output vectors (spectra). Let us denote the length of the impulse response as L. Then, if N is the length of the input vector $\mathbf{x}$ the length of the output vector $\mathbf{y}$ is $N + L - 1$.

The algorithm of one dimensional deconvolution resides in calculation of the matrix C (Eq. (18)), vector $\mathbf{y}'$ (Eq. (19)), before starting iterations, and in successive correction of the vector $\mathbf{x}$ according to Eq. (20). Each correction is done by the multiplication of the matrix C with the particular solution ${}^k\mathbf{x}$. Exactly this multiplication is critical. In one dimensional deconvolution N^2 multiplications must be carried out. However in n-dimensional deconvolution the number of multiplications increases to M^2, where $M = \prod_{j=1}^{n} N_j$.

Let us illustrate the situation for one dimensional Gold deconvolution for $L = 3$ and $N = 7$. According to Eq. (7) we obtain

$$\begin{bmatrix} y_1(0) \\ y_1(1) \\ y_1(2) \\ y_1(3) \\ y_1(4) \\ y_1(5) \\ y_1(6) \\ y_1(7) \\ y_1(8) \end{bmatrix} = \begin{bmatrix} h_1(0) & 0 & 0 & 0 & 0 & 0 & 0 \\ h_1(1) & h_1(0) & 0 & 0 & 0 & 0 & 0 \\ h_1(2) & h_1(1) & h_1(0) & 0 & 0 & 0 & 0 \\ 0 & h_1(2) & h_1(1) & h_1(0) & 0 & 0 & 0 \\ 0 & 0 & h_1(2) & h_1(1) & h_1(0) & 0 & 0 \\ 0 & 0 & 0 & h_1(2) & h_1(1) & h_1(0) & 0 \\ 0 & 0 & 0 & 0 & h_1(2) & h_1(1) & h_1(0) \\ 0 & 0 & 0 & 0 & 0 & h_1(2) & h_1(1) \\ 0 & 0 & 0 & 0 & 0 & 0 & h_1(2) \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \\ x_1(4) \\ x_1(5) \\ x_1(6) \end{bmatrix} \tag{21}$$

or

$$\mathbf{y} = H \cdot \mathbf{x} \ . \tag{22}$$

To calculate the matrix C and the vector $\mathbf{y}'$ one needs the matrix

$$B = H^T H. \tag{23}$$

In our example we have

$$B = \begin{bmatrix} b_0 & b_1 & b_2 & 0 & 0 & 0 & 0 \\ b_1 & b_0 & b_1 & b_2 & 0 & 0 & 0 \\ b_2 & b_1 & b_0 & b_1 & b_2 & 0 & 0 \\ 0 & b_2 & b_1 & b_0 & b_1 & b_2 & 0 \\ 0 & 0 & b_2 & b_1 & b_0 & b_1 & b_2 \\ 0 & 0 & 0 & b_2 & b_1 & b_0 & b_1 \\ 0 & 0 & 0 & 0 & b_2 & b_1 & b_0 \end{bmatrix} \tag{24}$$

where

$$\begin{aligned} b_0 &= {h_1}^2(0) + {h_1}^2(1) + {h_1}^2(2) \\ b_1 &= h_1(0)h_1(1) + h_1(1)h_1(2) \\ b_2 &= h_1(0)h_1(2) \quad . \end{aligned} \tag{25}$$

One can observe that the matrix B is 5-diagonal symmetrical matrix. In general case it is $(2L-1)$-diagonal symmetrical matrix. However to store its elements the vector $\mathbf{b}$ of the length L (in our example 3) is completely sufficient. The element i

of the vector **b** is

$$b_i = \sum_{j=o}^{L-1-i} h_1(j) \cdot h_1(i+j); \qquad i \in< 0, L-1 > \quad . \tag{26}$$

Regarding to Eqs. (18), (23) the matrix C is

$$C = B^T \cdot B = B \cdot B = \begin{bmatrix} d_{00} & d_{01} & c_2 & c_3 & c_4 & 0 & 0 \\ d_{01} & d_{11} & c_1 & c_2 & c_3 & c_4 & 0 \\ c_2 & c_1 & c_0 & c_1 & c_2 & c_3 & c_4 \\ c_3 & c_2 & c_1 & c_0 & c_1 & c_2 & c_3 \\ c_4 & c_3 & c_2 & c_1 & c_0 & c_1 & c_2 \\ 0 & c_4 & c_3 & c_2 & c_1 & d_{11} & d_{01} \\ 0 & 0 & c_4 & c_3 & c_2 & d_{01} & d_{00} \end{bmatrix} \tag{27}$$

where

$$\begin{aligned} c_0 &= {b_0}^2 + 2{b_1}^2 + 2{b_2}^2 \\ c_1 &= 2b_0 b_1 + 2b_1 b_2 \\ c_2 &= 2b_0 b_2 + {b_1}^2 \\ c_3 &= 2b_1 b_2 \\ c_2 &= {b_2}^2 \\ d_{00} &= c_0 - {b_1}^2 - {b_2}^2 \\ d_{01} &= c_1 - b_1 b_2 \\ d_{11} &= c_0 - {b_2}^2 \quad . \end{aligned} \tag{28}$$

It can be seen that the matrix C is 9-diagonal (in general case $(4L-3)$-diagonal) symmetrical matrix. Its elements can be stored in vector **c** of the length $2L-1$ (in our example 5) and in symmetrical submatrix D of the size $(L-1) \cdot (L-1)$. These elements are

$$c_i = \sum_{j=-L+1}^{L-1-i} b_{|j|} \cdot b_{|i+j|}; \qquad i \in< 0, 2L-2 > \tag{29}$$

and

$$d_{ij} = c_{j-i} - \sum_{k=j+1}^{L-1} b_k \cdot b_{k-j+i}; \qquad j \in< 0, L-2 >, i \in< j, L-2 > . \tag{30}$$

For completeness' sake we introduce relations to calculate vector $\mathbf{y}'$. According to Eq.(19) and with respect to Eq.(22), (23) we have

$$\mathbf{y}' = H^T H H^T \mathbf{y} = B \cdot H^T \mathbf{y} = B \cdot \mathbf{p} \tag{31}$$

where the vector **p** is

$$p_i = \sum_{j=0}^{L-1} h_1(j) \cdot y_1(i+j); \qquad i \in< 0, N-1 > . \tag{32}$$

Then the vector $\mathbf{y}'$ is

$$\mathbf{y}' = \sum_{j=-\min(i,L-1)}^{\min(N-1-i,L-1)} b_{|j|} \cdot p_{i+j}; \qquad i \in < 0, N-1 > \quad . \tag{33}$$

To store the vector $\mathbf{y}'$ we need $N-1$ elements.

The Eqs. (26), (29), (30), (32), (33) represent optimal algorithm to compute vectors $\mathbf{b}, \mathbf{c}, \mathbf{p}, \mathbf{y}'$ and submatrix d. These computations can be carried out before starting the deconvolution iterative algorithm itself. From the point of view of number of numerical operations (multiplications and additions), and thus the execution time, the multiplication of the matrix C with the vector ${}^k\mathbf{x}_1$ (see denominator in Eq. (20)) is the most critical. This operation must be carried out in each iteration step.

However from the above given example it is worth noticing that the matrix C contains zero elements. By removing redundant multiplications with zeros in the matrix C, the operation $C \cdot {}^k\mathbf{x}_1$ can be substantially sped up. The optimal algorithm of the operation

$$\mathbf{z} = C \cdot {}^k\mathbf{x}_1 \tag{34}$$

where k is iteration step, can be expressed

$$z_i = \sum_{j=-\min(i,2L-2)}^{\min(N-1-i,2L-2)} e \cdot {}^k\mathbf{x}_1(i+j); \qquad i \in < 0, N-1 > \tag{35}$$

where

$$e = \begin{cases} d_{i,i+j} & \text{if} \quad i < L-1 \quad \text{and} \quad i+j < L-1 \\ d_{N-1-i-j,N-1-i} & \text{if} \quad i > N-L \quad \text{and} \quad i+j > N-L \\ c_{|j|} & \text{else.} \end{cases} \tag{36}$$

The Table 1. presents the number of needed multiplications for the optimized and non-optimized algorithms to carry out one dimensional Gold deconvolution. Obviously for $L \approx 10$ and $N > 1000$ the optimized algorithm substantially speeds up the calculation of Gold deconvolution.

Table 1.

Let us go ahead in these considerations for two dimensional Gold deconvolution. Again let us illustrate it using a small example for $L_1 = 3, L_2 = 3, N_1 = 7, N_2 = 7$. Then expressing $\mathbf{x}, \mathbf{y}$ using Eqs. (9), (10), (11), analogously to Eq. (22), one can write the matrix H

$$H = \begin{bmatrix} H_0 & 0 & 0 & 0 & 0 & 0 & 0 \\ H_1 & H_0 & 0 & 0 & 0 & 0 & 0 \\ H_2 & H_1 & H_0 & 0 & 0 & 0 & 0 \\ 0 & H_2 & H_1 & H_0 & 0 & 0 & 0 \\ 0 & 0 & H_2 & H_1 & H_0 & 0 & 0 \\ 0 & 0 & 0 & H_2 & H_1 & H_0 & 0 \\ 0 & 0 & 0 & 0 & H_2 & H_1 & H_0 \\ 0 & 0 & 0 & 0 & 0 & H_2 & H_1 \\ 0 & 0 & 0 & 0 & 0 & 0 & H_2 \end{bmatrix} \tag{37}$$

where

$$H_i = \begin{bmatrix} h_2(0,i) & 0 & 0 & 0 & 0 & 0 & 0 \\ h_2(1,i) & h_2(0,i) & 0 & 0 & 0 & 0 & 0 \\ h_2(2,i) & h_2(1,i) & h_2(0,i) & 0 & 0 & 0 & 0 \\ 0 & h_2(2,i) & h_2(1,i) & h_2(0,i) & 0 & 0 & 0 \\ 0 & 0 & h_2(2,i) & h_2(1,i) & h_2(0,i) & 0 & 0 \\ 0 & 0 & 0 & h_2(2,i) & h_2(1,i) & h_2(0,i) & 0 \\ 0 & 0 & 0 & 0 & h_2(2,i) & h_2(1,i) & h_2(0,i) \\ 0 & 0 & 0 & 0 & 0 & h_2(2,i) & h_2(1,i) \\ 0 & 0 & 0 & 0 & 0 & 0 & h_2(2,i) \end{bmatrix}; \quad i = 0,1,2 \quad . \tag{38}$$

This time the matrix B is

$$B = H^T \cdot H = \begin{bmatrix} B_0 & B_1 & B_2 & 0 & 0 & 0 & 0 \\ {B_1}^T & B_0 & B_1 & B_2 & 0 & 0 & 0 \\ {B_2}^T & {B_1}^T & B_0 & B_1 & B_2 & 0 & 0 \\ 0 & {B_2}^T & {B_1}^T & B_0 & B_1 & B_2 & 0 \\ 0 & 0 & {B_2}^T & {B_1}^T & B_0 & B_1 & B_2 \\ 0 & 0 & 0 & {B_2}^T & {B_1}^T & B_0 & B_1 \\ 0 & 0 & 0 & 0 & {B_2}^T & {B_1}^T & B_0 \end{bmatrix}, \tag{39}$$

where

$$\begin{aligned} B_0 &= {H_0}^T H_0 + {H_1}^T H_1 + {H_2}^T H_2 \\ B_1 &= {H_1}^T H_0 + {H_2}^T H_1 \\ B_2 &= {H_2}^T H_0 \end{aligned} \tag{40}$$

and 0 represent submatrices of the size 7x7. The matrix B has the size 49x49. The submatrices B_0, B_1, B_2 are 5-diagonal (in this case non-symmetrical) matrices

$$B_i = \begin{bmatrix} b_i(0) & b_i(1) & b_i(2) & 0 & 0 & 0 & 0 \\ b_i(-1) & b_i(0) & b_i(1) & b_i(2) & 0 & 0 & 0 \\ b_i(-2) & b_i(-1) & b_i(0) & b_i(1) & b_i(2) & 0 & 0 \\ 0 & b_i(-2) & b_i(-1) & b_i(0) & b_i(1) & b_i(2) & 0 \\ 0 & 0 & b_i(-2) & b_i(-1) & b_i(0) & b_i(1) & b_i(2) \\ 0 & 0 & 0 & b_i(-2) & b_i(-1) & b_i(0) & b_i(1) \\ 0 & 0 & 0 & 0 & b_i(-2) & b_i(-1) & b_i(0) \end{bmatrix}; \quad i = 0,1,2\,. \tag{41}$$

The storage of submatrices B_i can be accomplished in 3 vectors with 5 elements (in general case in a matrix with the size $L_2 \cdot (2L_1 - 1)$). The j-th element of i-th vector **b** is

$$b_i(j) \quad = \quad \sum_{k=0}^{L_2-1-i} \; \sum_{l=\max(0,-j)}^{\min(L_1-1-j,L_1-1)} h_2(l,k)\, h_2(l+j,k+i); \tag{42}$$

where $i \in < 0, L_2 - 1 >, j \in < -L_1 + 1, L_1 - 1 >$.

Now we shall draw an analogy with one dimensional deconvolution. We define the matrix C

$$C = B^T \cdot B = B \cdot B = \begin{bmatrix} D_{00} & D_{01} & C_2 & C_3 & C_4 & 0 & 0 \\ {D_{01}}^T & D_{11} & C_1 & C_2 & C_3 & C_4 & 0 \\ {C_2}^T & {C_1}^T & C_0 & C_1 & C_2 & C_3 & C_4 \\ {C_3}^T & {C_2}^T & {C_1}^T & C_0 & C_1 & C_2 & C_3 \\ {C_4}^T & {C_3}^T & {C_2}^T & {C_1}^T & C_0 & C_1 & C_2 \\ 0 & {C_4}^T & {C_3}^T & {C_2}^T & {C_1}^T & D'_{11} & D'_{01} \\ 0 & 0 & {C_4}^T & {C_3}^T & {C_2}^T & {D'_{01}}^T & D'_{00} \end{bmatrix}. \tag{43}$$

The submatrices C, D of the size 7x7 are

$$\begin{aligned} C_0 &= {B_2}^T B_2 + {B_1}^T B_1 + B_0 B_0 + B_1 {B_1}^T + B_2 {B_2}^T \\ C_1 &= {B_1}^T B_2 + B_0 B_1 + B_1 B_0 + B_2 {B_1}^T \\ C_2 &= B_0 B_2 + B_1 B_1 + B_2 B_0 \\ C_3 &= B_1 B_2 + B_2 B_1 \\ C_4 &= B_2 B_2 \\ D_{00} &= B_0 B_0 + B_1 {B_1}^T + B_2 {B_2}^T = C_0 - {B_1}^T B_1 - {B_2}^T B_2 \\ D_{01} &= B_0 B_1 + B_1 B_0 + B_2 {B_1}^T = C_1 - {B_1}^T B_2 \\ D_{11} &= {B_1}^T B_1 + B_0 B_0 + B_1 {B_1}^T + B_2 {B_2}^T = C_0 - {B_2}^T B_2 \end{aligned} \tag{44}$$

and

$$D'_{i,j}(l,k) = D_{i,j}(N_1 - 1 - k,\, N_1 - 1 - l) \tag{45}$$

where $k, l \in < 0, N_1 - 2 >$ and $i, j \in < 0, L_2 - 2 >$. In this case the C is 9-diagonal matrix consisting of submatrices $C_i, D_{jk}, 0$. Each of the submatrices C_i, D_{jk} is 9-diagonal matrix. In general case the C is $(4L_2 - 3)$-diagonal matrix of submatrices from which each is $(4L_1 - 3)$-diagonal matrix, i.e.,

$$C_i = \begin{bmatrix} cd_i(0,0) & cd_i(0,1) & c_i(2) & c_i(3) & c_i(4) & 0 & 0 \\ cd_i(1,0) & cd_i(1,1) & c_i(1) & c_i(2) & c_i(3) & c_i(4) & 0 \\ c_i(-2) & c_i(-1) & c_i(0) & c_i(1) & c_i(2) & c_i(3) & c_i(4) \\ c_i(-3) & c_i(-2) & c_i(-1) & c_i(0) & c_i(1) & c_i(2) & c_i(3) \\ c_i(-4) & c_i(-3) & c_i(-2) & c_i(-1) & c_i(0) & c_i(1) & c_i(2) \\ 0 & c_i(-4) & c_i(-3) & c_i(-2) & c_i(-1) & cd_i(1,1) & cd_i(0,1) \\ 0 & 0 & c_i(-4) & c_i(-3) & c_i(-2) & cd_i(1,0) & cd_i(0,0) \end{bmatrix};$$

$$i \in < 0, 2L_1 - 2 > \tag{46}$$

and

$$D_{ij} = \begin{bmatrix} dd_{ij}(0,0) & dd_{ij}(0,1) & d_{ij}(2) & d_{ij}(3) & d_{ij}(4) & 0 & 0 \\ dd_{ij}(1,0) & dd_{ij}(1,1) & d_{ij}(1) & d_{ij}(2) & d_{ij}(3) & d_{ij}(4) & 0 \\ d_{ij}(-2) & d_{ij}(-1) & d_{ij}(0) & d_{ij}(1) & d_{ij}(2) & d_{ij}(3) & d_{ij}(4) \\ d_{ij}(-3) & d_{ij}(-2) & d_{ij}(-1) & d_{ij}(0) & d_{ij}(1) & d_{ij}(2) & d_{ij}(3) \\ d_{ij}(-4) & d_{ij}(-3) & d_{ij}(-2) & d_{ij}(-1) & d_{ij}(0) & d_{ij}(1) & d_{ij}(2) \\ 0 & d_{ij}(-4) & d_{ij}(-3) & d_{ij}(-2) & d_{ij}(-1) & de_{ij}(0,0) & de_{ij}(0,1) \\ 0 & 0 & d_{ij}(-4) & d_{ij}(-3) & d_{ij}(-2) & de_{ij}(1,0) & de_{ij}(1,1) \end{bmatrix}$$

$$i,j \in < 0, 2L_2 - 2 > . \tag{47}$$

Further let us define the function $\mathrm{sgn}(x) = \frac{x}{|x|}$ for $x \neq 0$ and $\mathrm{sgn}(0) = 1$. After thorough analysis we arrive to the formulas expressing the elements of the matrix C. The elements of the vector $\mathbf{c}_{i_2}$ of the submatrix C_{i_2} are

$$\begin{aligned} c_{i_2}(i_1) &= \sum_{j_2=-L_2+1}^{L_2-1-i_2} \sum_{j_1=\max(-L_1+1,-L_1+1+i_1)}^{\min(L_1-1,L_1-1+i_1)} b_{|j_2|}\big((j_1-i_1)\mathrm{sgn}(j_2)\big) \cdot \\ &\quad \cdot b_{|i_2+j_2|}\big(j_1 \mathrm{sgn}(i_2+j_2)\big), \end{aligned} \tag{48}$$

where $i_1 \in < -2L_1+2, 2L_1-2 >, i_2 \in < 0, 2L_2-2 >$. Similarly the elements of the submatrix cd_{i_2} of the submatrix C_{i_2} are

$$\begin{aligned} cd_{i_2}(i,j) &= c_{i_2}(j-i) - \\ &\quad - \sum_{j_2=-L_2+1}^{L_2-1-i_2} \sum_{j_1=\max(-L_1+1,-L_1+1+j-i)}^{-i-1} b_{|j_2|}\big((j_1-j+i)\mathrm{sgn}(j_2)\big) \cdot \\ &\quad \cdot b_{|i_2+j_2|}\big(j_1 \mathrm{sgn}(i_2+j_2)\big), \end{aligned} \tag{49}$$

where $i,j \in < 0, L_1-2 >, i_2 \in < 0, 2L_2-2 >$. The elements of the vector $\mathbf{d}_{ij}$ of the submatrix D_{ij} are

$$\begin{aligned} d_{ij}(i_1) &= c_{j-i}(i_1) - \\ &\quad - \sum_{j_2=-L_2+1}^{-j-1} \sum_{j_1=\max(-L_1+1,-L_1+1+i_1)}^{\min(L_1-1,L_1-1+i_1)} b_{|j_2|}\big((j_1-i_1)\mathrm{sgn}(j_2)\big) \cdot \\ &\quad \cdot b_{|j-i+j_2|}\big(j_1 \mathrm{sgn}(j-i+j_2)\big), \end{aligned} \tag{50}$$

where $j \in < 0, L_2-2 >, i \in < j, L_2-2 >, i_1 \in < -2L_1+2, 2L_1-2 >$. Finally for the elements of the submatrix D_{i_1,i_2} we obtain

$$\begin{aligned} dd_{i_1,i_2}(i,j) &= d_{i_1,i_2}(j-i) - \\ &\quad - \sum_{j_2=-i_2}^{L_2-1-i_2+i_1} \sum_{j_1=\max(-L_1+1,-L_1+1+j-i)}^{-i-1} b_{|j_2|}\big((j_1-j+i)\mathrm{sgn}(j_2)\big) \cdot \\ &\quad \cdot b_{|j_2+i_2-i_1|}\big(j_1 \mathrm{sgn}(j_2+i_2-i_1)\big), \end{aligned} \tag{51}$$

$$\begin{aligned} de_{i_1,i_2}(L_1-2-j, L_1-2-i) &= d_{i_1,i_2}(j-i) - \\ &- \sum_{j_2=-i_2}^{L_2-1-i_2+i_1} \sum_{j_1=j+1}^{\min(L_1-1,L_1-1+j-i)} b_{|j_2|}\big((j_1-j+i)\mathrm{sgn}(j_2)\big) \cdot \\ &\cdot b_{|j_2+i_2-i_1|}\big(j_1\mathrm{sgn}(j_2+i_2-i_1)\big), \end{aligned} \quad (52)$$

where $i_2 \in <0, L_2-2>, i_1 \in <i_2, L_2-2>, i,j \in <0, L_1-2>$. We have determined all elements of the matrix C.

Analogously to one dimensional deconvolution we calculate the vector $\mathbf{y}'$ (Eqs. (31), (32), (33)). Here we assume that the vectors $\mathbf{h}_2^{(2)}$ and $\mathbf{y}_2^{(2)}$ were created through the use of procedure given in the chapter 2 (Eqs. (8)-(16)). Then the vector $\mathbf{y}'$ is

$$p_{i_1+N_1\cdot i_2} = \sum_{j_2=0}^{L_2-1} \sum_{j_1=0}^{L_1-1} h_2^{(2)}(j_1+N_1 j_2)\, y_2^{(2)}\big(i_1+j_1+N_2(i_2+j_2)\big), \quad (53)$$

$$\begin{aligned} &y'(i_1+N_1\cdot i_2) = \\ &= \sum_{j_2=-\min(i_2,L_2-1)}^{\min(N_2-1-i_2,L_2-1)} \sum_{j_1=-\min(i_1,L_1-1)}^{\min(N_1-1-i_1,L_1-1)} b_{|j_2|}\big(j_1\mathrm{sgn}(j_2)\big)\, p_{i_1+j_1+N_1(i_2+j_2)}, \end{aligned}$$

$$(54)$$

where $i_1 \in <0, N_1-1>, i_2 \in <0, N_2-1>$. The Eqs. (42), (48)-(54) represent optimal algorithms to compute the vectors $\mathbf{b}$ and the matrix C.

Again, from the point of view of time, the multiplication of the square matrix C of the size $N_2N_1 \cdot N_2N_1$ with the vector ${}^k\mathbf{x}_2^{(2)}$ is the most critical. From the Eqs. (43), (46), (47) it is apparent that on the one hand the matrix C contains zero submatrices and on the other hand each of the submatrices C_i, D_{ij} contains zero elements. The multiplications with zeros are redundant and therefore can be omitted. Consequently the optimal algorithm of multiplication matrix C with the vector of particular solution ${}^k\mathbf{x}_2^{(2)}$

$$\mathbf{z} = C \cdot {}^k\mathbf{x}_2^{(2)}, \quad (55)$$

can be expressed

$$\begin{aligned} &\mathbf{z}_{i_2\cdot L_1+i_1} = \\ &= \sum_{j_2=-\min(i_2,2L_2-2)}^{\min(N_2-1-i_2,2L_2-2)} \sum_{j_1=-\min(i_1,2L_1-2)}^{\min(N_1-1-i_1,2L_1-2)} e \cdot {}^k x_2^{(1)}\big((i_2+j_2)L_1+i_1+j_1\big) \end{aligned}$$

$$(56)$$

where k is iteration step, $i_1 \in <0, L_1-1>, i_2 \in <0, L_2-1>$. The value e can be determined employing the following algorithm if $\quad i_2 < L_2 - 1 \quad$ and $\quad i_2 + j_2 < L_2 - 1$

$$\begin{cases} \text{if} \quad i_1 < L_1 - 1 \quad \text{and} \quad i_1 + j_1 < L_1 - 1 \\ \qquad\qquad e = dd_{i_2, i_2 + j_2}(i_1, i_1 + j_1) \\ \text{if} \quad i_1 > N_1 - L_1 \quad \text{and} \quad i_1 + j_1 > N_1 - L_1 \\ \qquad\qquad e = de_{i_2, i_2 + j_2}(i_1 - N_1 + L_1 - 1, i_1 + j_1 - N_1 + L_1 - 1) \\ \text{else} \\ \qquad\qquad e = d_{i_2, i_2 + j}(j_1) \end{cases}$$

if $\quad i_2 > N_2 - L_2 \quad$ and $\quad i_2 + j_2 > N_2 - L_2$

$$\begin{cases} \text{if} \quad i_1 < L_1 - 1 \quad \text{and} \quad i_1 + j_1 < L_1 - 1 \\ \qquad\qquad e = de_{N_2 - 1 - i_2, N_2 - 1 - i_2 - j_2}(L_1 - 2 - i_1 - j_1, L_1 - 2 - i_1) \\ \text{if} \quad i_1 > N_1 - L_1 \quad \text{and} \quad i_1 + j_1 > N_1 - L_1 \\ \qquad\qquad e = dd_{N_2 - 1 - i_2, N_2 - 1 - i_2 - j_2}(N_1 - 1 - i_1 - j_1, N_1 - 1 - i_1) \\ \text{else} \\ \qquad\qquad e = d_{N_2 - 1 - i_2, N_2 - 1 - i_2 - j}(j_1) \end{cases}$$

if $\quad j_2 < 0$

$$\begin{cases} \text{if} \quad i_1 < L_1 - 1 \quad \text{and} \quad i_1 + j_1 < L_1 - 1 \\ \qquad\qquad e = cd_{j_2}(i_1 + j_1, i_1) \\ \text{if} \quad i_1 > N_1 - L_1 \quad \text{and} \quad i_1 + j_1 > N_1 - L_1 \\ \qquad\qquad e = cd_{j_2}(N_1 - 1 - i_1, N_1 - 1 - i_1 - j_1) \\ \text{else} \\ \qquad\qquad e = c_{j_2}(-j_1) \end{cases}$$

else

$$\begin{cases} \text{if} \quad i_1 < L_1 - 1 \quad \text{and} \quad i_1 + j_1 < L_1 - 1 \\ \qquad\qquad e = cd_{j_2}(i_1, i_1 + j_1) \\ \text{if} \quad i_1 > N_1 - L_1 \quad \text{and} \quad i_1 + j_1 > N_1 - L_1 \\ \qquad\qquad e = cd_{j_2}(N_1 - 1 - i_1 - j_1, N_1 - 1 - i_1) \\ \text{else} \\ \qquad\qquad e = c_{j_2}(j_1) \quad . \end{cases} \tag{57}$$

With regard to Eqs. (43), (46), (47) the matrix C can be stored in the form of vectors $\mathbf{c}, \mathbf{d}$ and submatrices cd, dd, de. The memory requirements are given in Table 2.

Table 2.

The total memory size needed to store the matrix C in optimized Gold deconvolution algorithm is

$$N_0 = {L_1}^2(2{L_2}^2 - 2L_2 + 1) + 2L_1(2L_2 - 1) - {L_2}^2 - 2L_2 + 1 \tag{58}$$

whereas in non-optimized algorithm it is

$$N_N = {N_1}^2 \cdot {N_2}^2 \,. \tag{59}$$

For the purpose of comparison let us present a small example. Provided we have $L_1 = L_2 = 10$ and $N_1 = N_2 = 128$ then $N_N = 256$ Mwords and $N_O = 17561$ words. If we suppose to store each word in 4 bytes (real in Fortran, float in C) then in non-optimized algorithm we need 1GB of memory as compared to approximately 70KB in optimized algorithm. Clearly without optimization the algorithm of two dimensional Gold deconvolution would not be realizable, also on the grounds of memory requirements, on majority of accessible computers.

Again, in analogy with one dimensional deconvolution we have analyzed the number of needed multiplications (Table 3) to calculate appropriate vectors and matrices for both optimized and non-optimized algorithms.

Table 3.

As mentioned above the multiplication of the matrix C with the vector of particular solution ${}^k\mathbf{x}_2^{(2)}$, according to Eqs. (55)-(57), is very time consuming and thus critical. In our example ($L_1 = L_2 = 10, N_1 = N_2 = 128$) the execution of operation (55) in one iteration step in the optimized algorithm would require 19.10^6 multiplications whereas in non-optimized algorithm 268.10^6 multiplications.

4. Experimental results

The algorithms of one and two dimensional deconvolution described so far have been used to process γ-ray spectra [5]. An example of the original one dimensional spectrum from GAMMASPHERE experiment on ^{252}Cf(s.f.) is shown in Fig. 1.

Fig. 1

When we solve the over-determined system of linear equations as a cyclic one we obtain the result given in Fig. 2. It illustrates extreme sensitivity of deconvolution operation to the statistical fluctuations presented in the spectrum. From mathematical point of view it represents exact solution. However from practical point of view it is worthless.

Fig. 2

Applying Van Cittert deconvolution [6], [7] to the spectrum in Fig. 1 (1000 iterations) one gets the result presented in Fig. 3. The method decomposes multiplets in the spectrum. Oscillations are undoubtedly smaller than in the previous case. Nevertheless such a result can be hardly accepted. The spectrum contains unrealistic negative values.

Fig. 3

In the next example we have processed the spectrum from Fig. 1 employing Fourier deconvolution [8], [9]. The result, after Fourier deconvolution, in Fig. 4

shows that the method decomposes, to some extent, multiplets. The oscillations are smaller than in Van Cittert deconvolution. Still the spectrum contains unrealistic negative values.

Fig. 4

Fig. 5 presents the spectrum after Gold deconvolution (1000 iterations) [4], [10]. The method unfolds multiplets. It does not oscillate. The resulting spectrum does not contain any negative value. The Gold deconvolution preserves peak positions and peak areas. In all 5 Figs. the scales in both dimensions (channels and counts) were the same. The properties of Van Cittert and Gold deconvolution were studied in detail in [4].

Fig. 5

Fig. 6 shows a detail (channels 200-342) of the original spectrum (Fig. 1) and the spectrum after Gold deconvolution (Fig. 5). It can be seen that in some places in deconvoluted spectrum one can discover peaks whose existence in original spectrum is indistinct (see 4-fold multiplet at the beginning of the spectrum).

Fig. 6

In Fig. 7 we give an example illustrating decomposition efficiency of the deconvolution procedure on gated coincidence spectrum from GAMMASPHERE experiment on ^{252}Cf(s.f.). The gate on 192.2keV $2^+ - 0^+$ transition in ^{104}Mo was used. The prompt γ-ray spectra associated with Ba isotopes are shown before (thick line) and after the deconvolution (thin line). Peaks marked by mass numbers of Ba isotopes are shown together with 131.9keV and 162.8keV transitions in ^{140}Ba – ^{140}La decay.

Fig. 7

The optimization of the algorithm of two dimensional Gold deconvolution permitted its implementation also for small PC based computers. Fig. 8 shows a part (64x64 channels) of a two dimensional γ-ray spectrum from multidetector system. The result after Gold deconvolution (200 iterations) is presented in different display modes [11] in Figs. 9,10. Apparently the method unfolds spatial doublets and multiplets (see dominant doublet in the background of Figs. 9, 10).

Figs. 8,9,10

In the next example we have processed a part (180x180 channels) of a two dimensional γ-ray spectrum from GAMMASPHERE (Fig. 11). The resulting spectrum after Gold deconvolution (1000 iterations) is shown in Fig. 12.

Figs. 11,12

Figs. 13, 14 show details (channels 80-180 in both dimensions) of spectra before and after Gold deconvolution.

Figs. 13,14

5. Conclusions

The paper presents non-oscillating deconvolution method applied in γ-ray spectroscopy. The use of deconvolution is a simple way to eliminate blurring effects of the electronic instabilities, i.e., the influence of detector response function. From all deconvolution methods tested the Gold deconvolution proved to work as the best one.

In the paper multidimensional Gold deconvolution method is derived. However the direct extension of the computational algorithm of one dimensional non-optimized Gold deconvolution to two, three and multidimensional systems leads to exponential growth of numerical operations. Given a rather slow convergence speed of iterative Gold method one needs considerable number of iteration steps (> 100). Therefore in practical cases the method without optimization is not directly applicable in multidimensional deconvolution.

Hence it was unavoidable to optimize number of numerical operations and to omit all of them that are redundant (e.g. multiplication with matrix elements, which are a priori known to be zeros). The proposed optimization algorithm benefits from the fact that the detector response function has only limited, relatively small, number of channels with non-zero counts. The optimized algorithms of both one and two dimensional Gold deconvolutions are derived in the chapter 3. In principle the method can be generalized to three or more dimensions.

One should come round to realizing that the optimization of the algorithm allows its realization in practice on small PC computers. The carrying out of one iteration step using optimized algorithm in the example presented in Figs. 11, 12 lasted 30 seconds (using Pentium 166MHz). Theoretically without optimization the whole calculation (1000 iterations) would last several hundreds hours.

Perhaps these execution times may seem too long. However one should be aware that in the example given we have linear system with 180x180=32400 unknowns. Thus the Gold deconvolution represents solution of the linear system of equations with the matrix of the size 32400x32400. Without optimization it would not be realizable in reasonable time. Moreover the optimized algorithms make it possible to store efficiently data in the memory. The speed and memory requirements of both optimized and non-optimized algorithms were analyzed in detail in the chapter 3.

The described Gold deconvolution method enables to improve resolution in one and two dimensional γ-ray spectra. This allows to perform decomposition of multiplets and often to discover hidden peaks.

Both one and two dimensional deconvolutions are implemented in "Integrated Multiparameter Nuclear Data Analysis Package" [11]. The deconvolution methods were applied to spectra after background subtraction. The efficient methods of background determination are described in [11], [12].

Acknowledgements

The work is supported in part by the Grant Agency of Slovak Republic through contract GAV 2/1155/96. The authors would like to thanks the GAMMASPHERE collaboration, to J. H. Hamilton, J. O. Rasmussen, G. M. Ter-Akopian and A. V. Daniel, for help in data preperation and valuable discussions.

References

1. J. M. Los Arcos, Nucl. Instr. and Meth. A369 (1996) 634.
2. J. Weese et al, Nucl. Instr. and Meth. A378(1996) 275.
3. C. W. Groetch, The Theory of Tikhonov Regularization for Fredholm Equations of the First Kind (Pitman, Boston, 1984).
4. P. Bandzuch, M. Morhac, J. Kristiak, Nucl. Instr. and Meth., A348 (1997) 506
5. G. M. Ter-Akopian, J. H. Hamilton, J. Kliman, V. Polhorsky, M. Morhac et al., Phys. Rev. Lett 11 (1994) 1477.
6. P. H. Van Cittert, Zum Einfluss der Spaltbreite auf die Intensitatverteilung in Spektralinien II, Z. Phys. 69 (1933) 298.
7. C. Xu, I. Assaoui, S. Jacquery, Algebraic analysis of the Van Cittert iterative method of deconvolution with a general relaxation factor, J. Opt. Soc. Am. A 11 (1994).
8. M. Cernansky, J. Appl. Cryst. 16 (1983) 103.
9. J. Colmenero, A. Arbe, A. Alegria, Phys. Rev. Lett. 71 (1993) 2603.
10. R. Gold, An iterative unfolding method for matrices, in: Mathematics and Computer Research and Development Rep. ANL -6984 (Argonne National Laboratory, Argone, III., 1964).
11. M. Morhac, J. Kliman, V. Matousek, I. Turzo, Integrated multiparameter nuclear data analysis package, Nucl. Instr. and Meth., In press.
12. M. Morhac, J. Kliman, V. Matousek, M. Veselsky, I. Turzo, Background estimation methods for multidimensional nuclear coincidence spectra, Nucl. Instr. and Meth., submitted for publication.

Figure captions:

1. Example of one dimensional γ-ray spectrum of ^{104}Mo and accompanying Ba isotopes at 192keV gate.
2. Exact cyclic deconvolution of the spectrum from Fig. 1.
3. Van Cittert deconvolution of the spectrum from Fig. 1.
4. Fourier deconvolution of the spectrum from Fig. 1.
5. Gold deconvolution of the spectrum from Fig. 1.
6. Detail of the original spectrum (dashed line) and the spectrum after Gold deconvolution (solid line).
7. Typical γ-ray spectrum obtained from GAMMASPHERE experiment on ^{252}Cf(s.f.) for gate on 192keV transition in ^{104}Mo. The prompt γ-ray spectra associated with Ba isotopes are shown before (thick line) and after the deconvolution (thin line). Peaks are marked by mass numbers of Ba isotopes.
8. Original two dimensional γ-ray spectrum (64x64 channels).
9. Spectrum from Fig. 8 after Gold deconvolution shown in grid display mode.
10. Spectrum from Fig. 8 after Gold deconvolution shown in contours display mode.
11. Original two dimensional γ-ray spectrum (180x180 channels).
12. Spectrum from Fig. 11 after Gold deconvolution shown in contour display mode.
13. Detail of the spectrum from Fig. 11 (channels 80-180 in both dimensions).
14. Detail of deconvoluted spectrum from Fig. 12 (channels 80-180 in both dimensions).

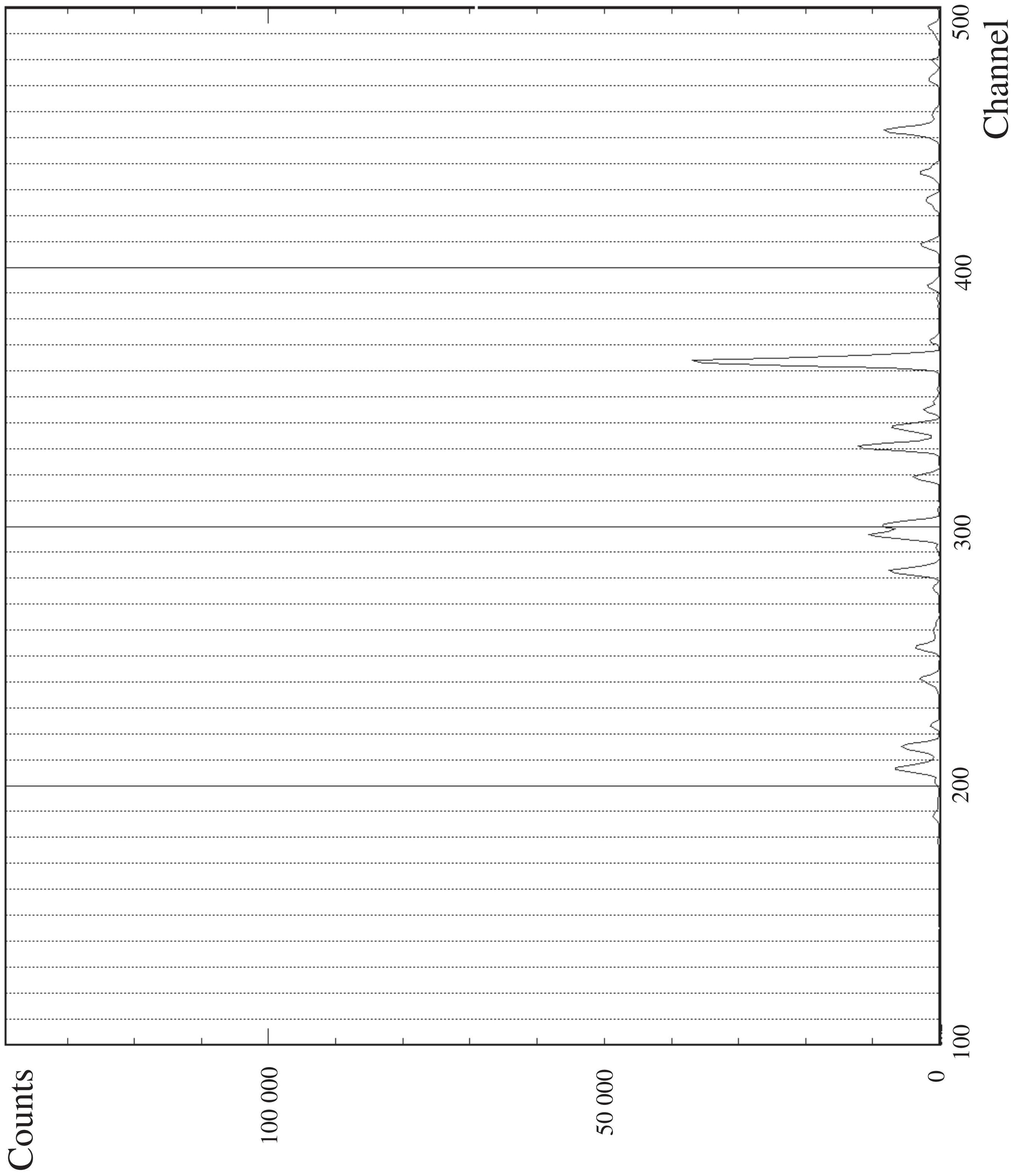

Fig.1 Example of one dimensional γ-ray spectrum of ^{104}Mo and accompanying Ba isotopes at 192keV gate.

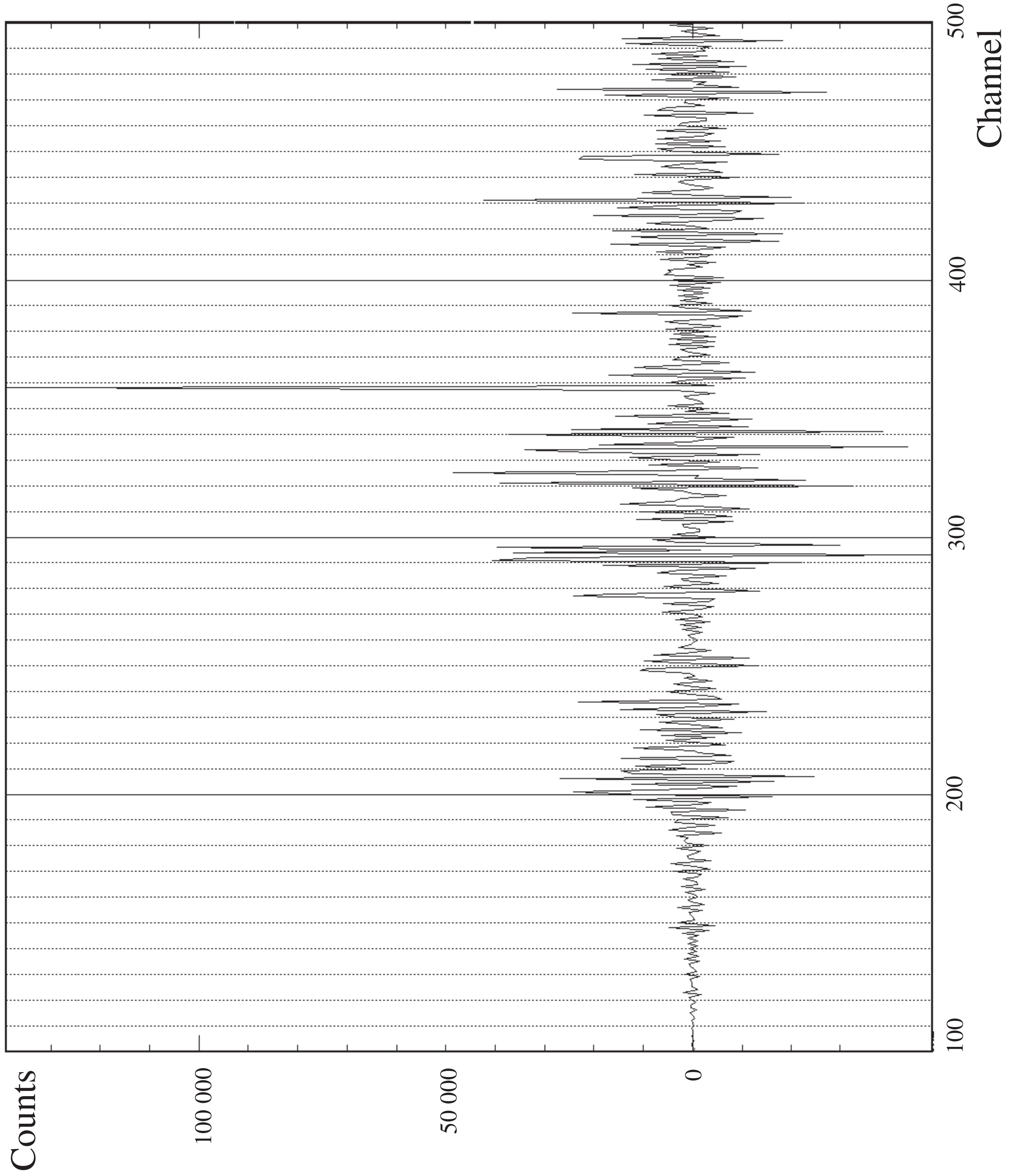

Fig.2 Exact cyclic deconvolution of the spectrum from Fig. 1.

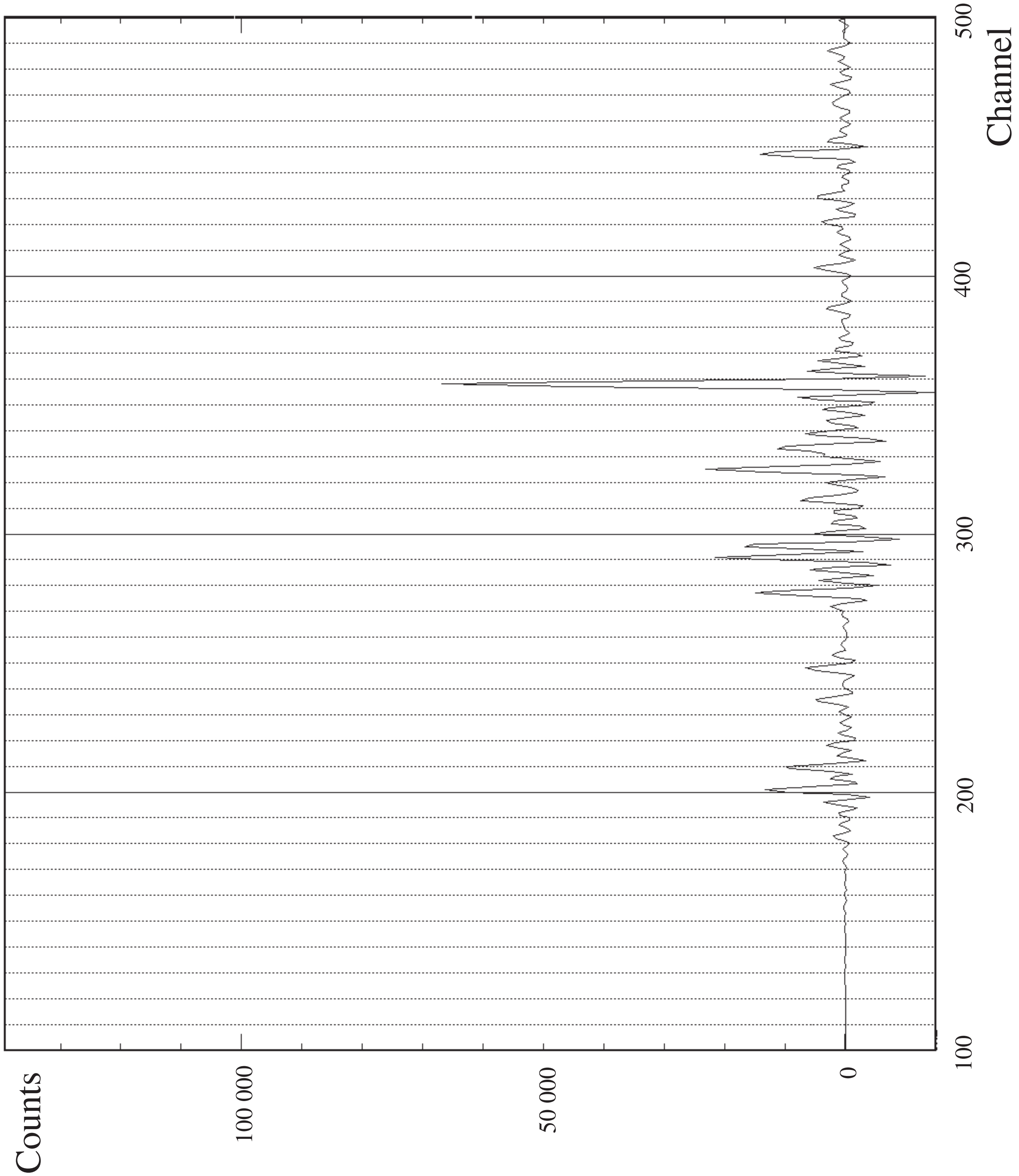

Fig.3 Van Cittert deconvolution of the spectrum from Fig. 1.

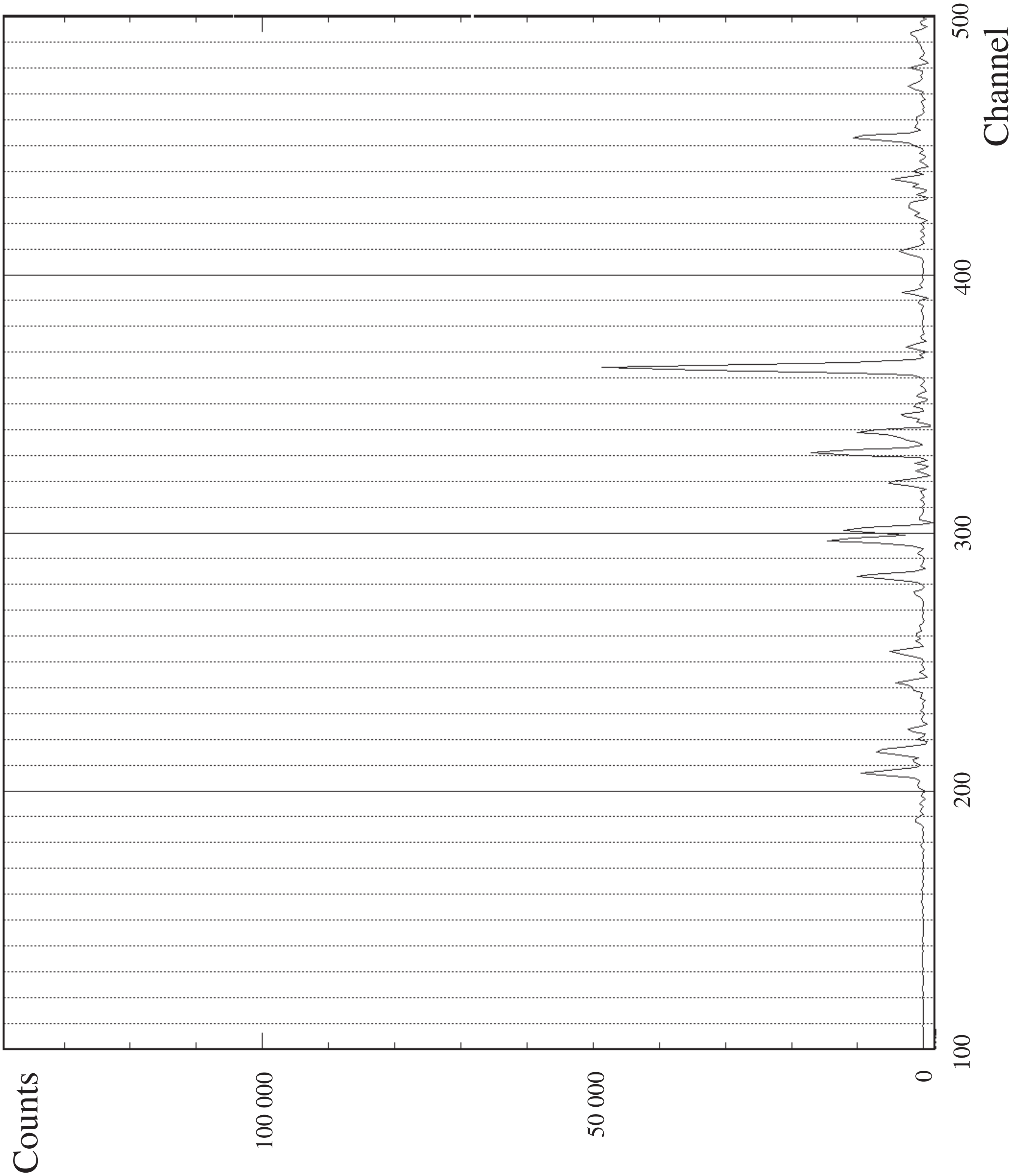

Fig.4 Fourier deconvolution of the spectrum from Fig. 1.

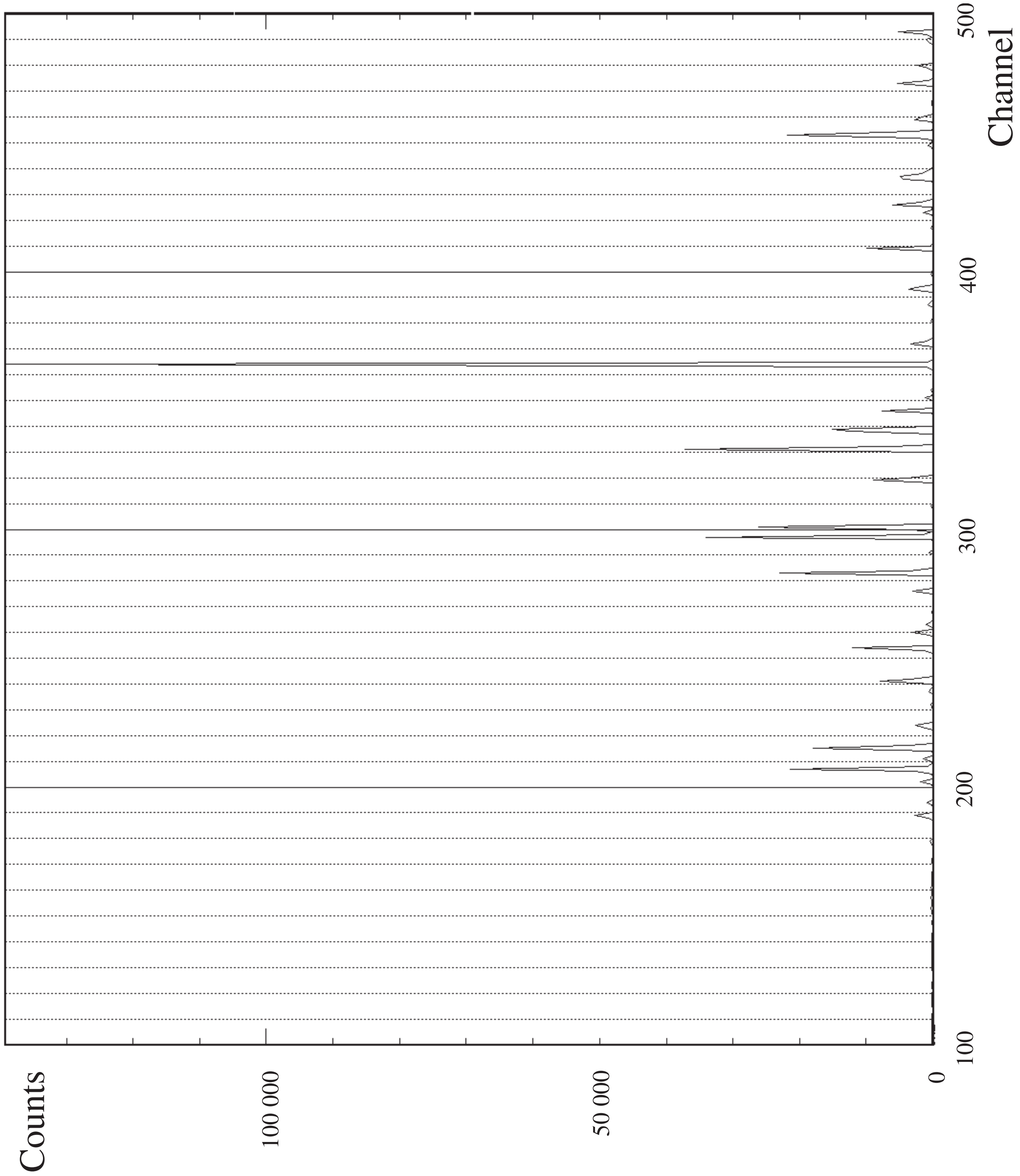

Fig.5 Gold deconvolution of the spectrum from Fig. 1.

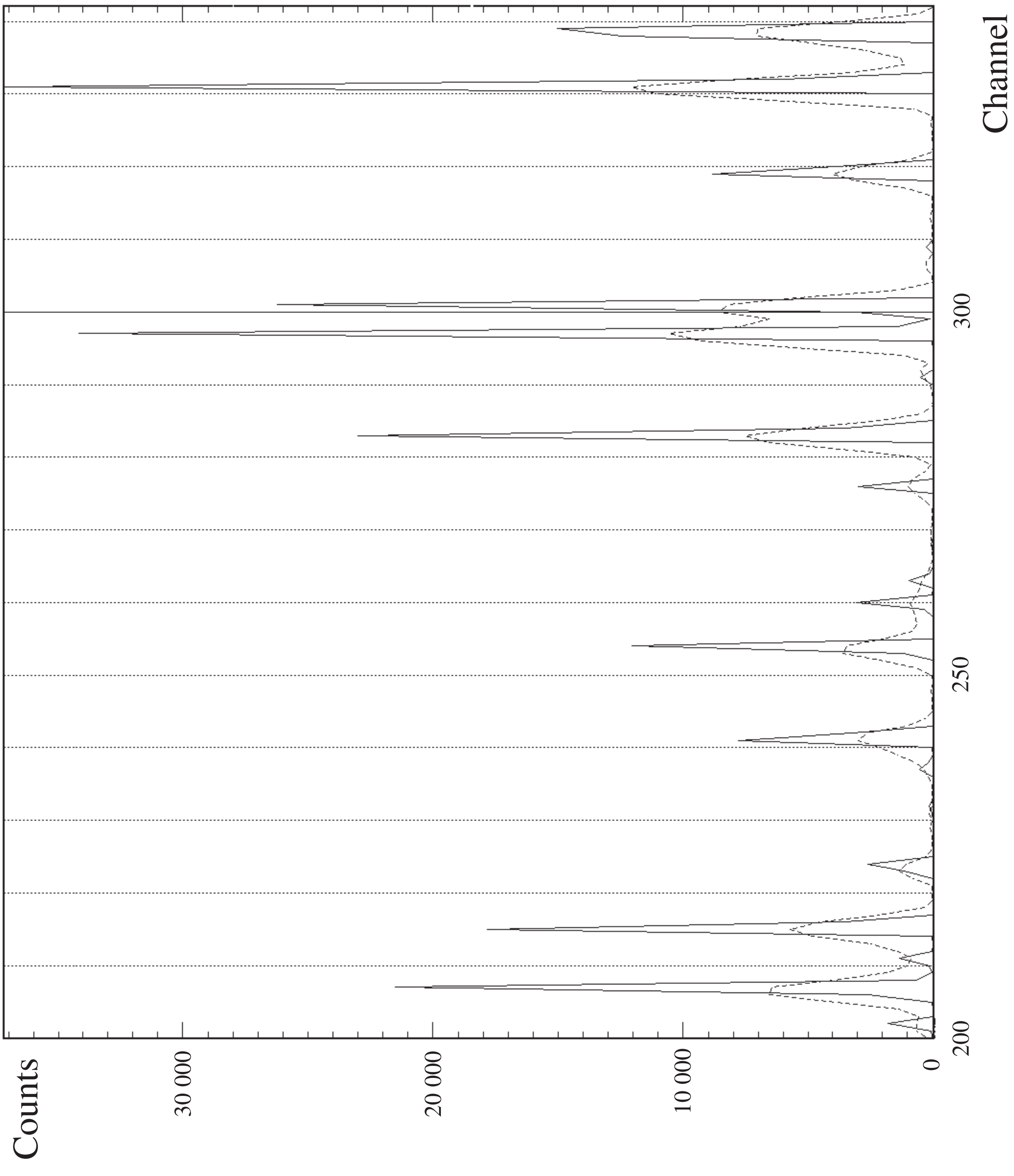

Fig.6 Detail of the original spectrum (dashed line) and the spectrum after Gold deconvolution (solid line).

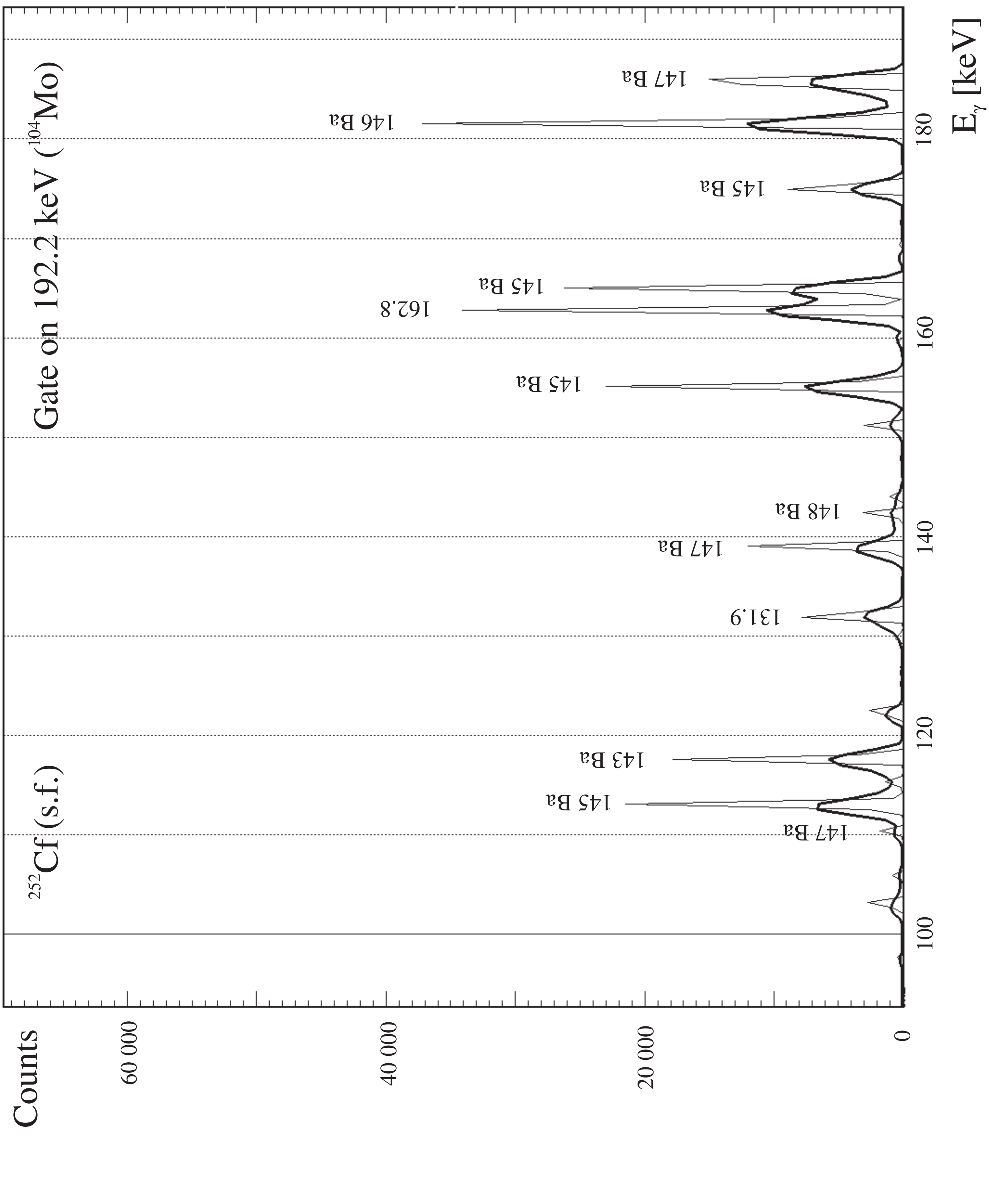

Fig.7 Typical γ-ray spectrum obtained from GAMMASPHERE experiment on ^{252}Cf(s.f.) for gate on 192keV transition in ^{104}Mo. The prompt γ-ray spectra associated with Ba isotopes are shown before (thick line) and after the deconvolution (thin line). Peaks are marked by mass numbers of Ba isotopes.

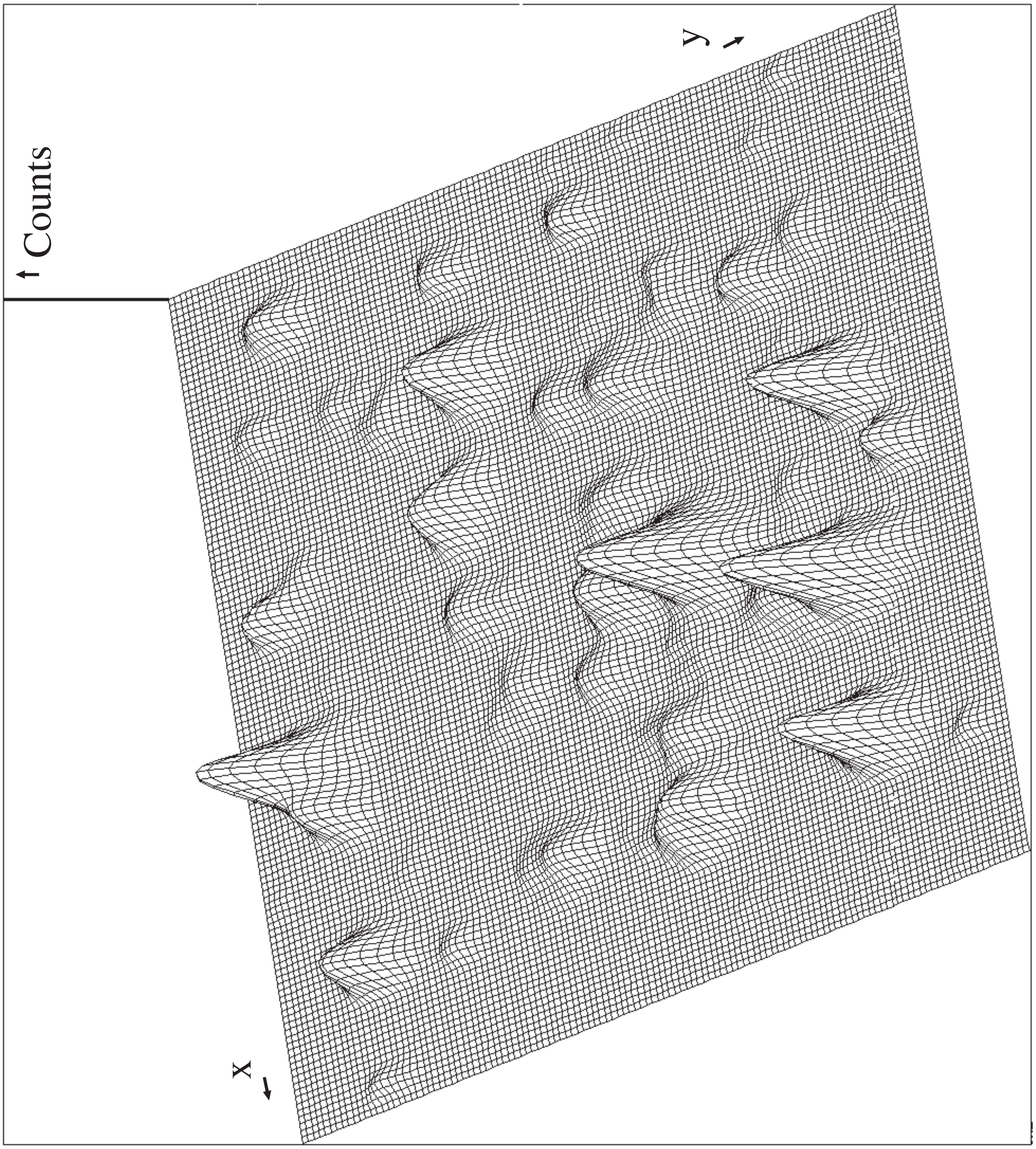

Fig.8 Original two dimensional γ-ray spectrum (64x64 channels).

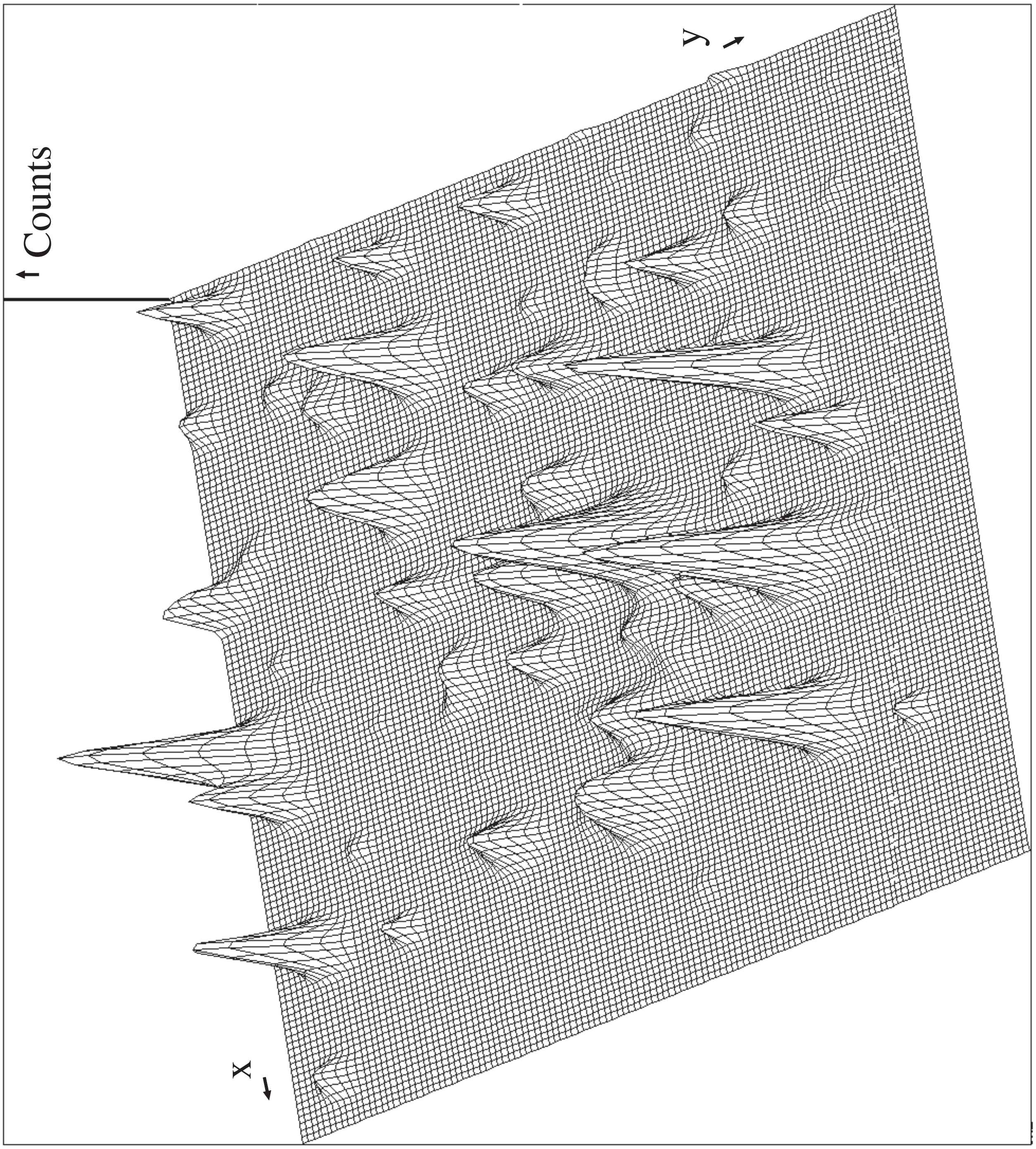

Fig.9 Spectrum from Fig. 8 after Gold deconvolution shown in grid display mode.

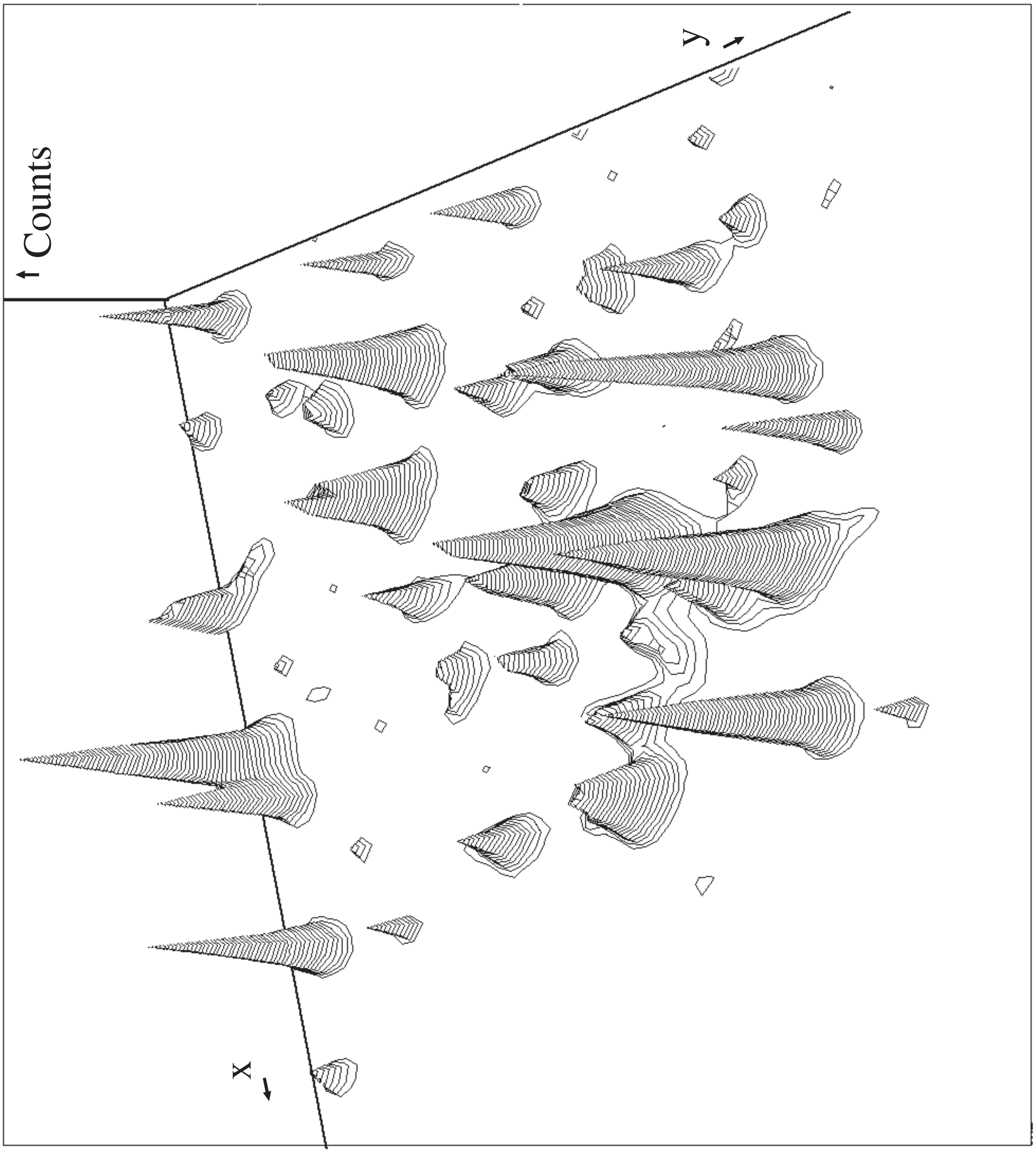

Fig.10 Spectrum from Fig.8 after Gold deconvolution shown in contours display mode.

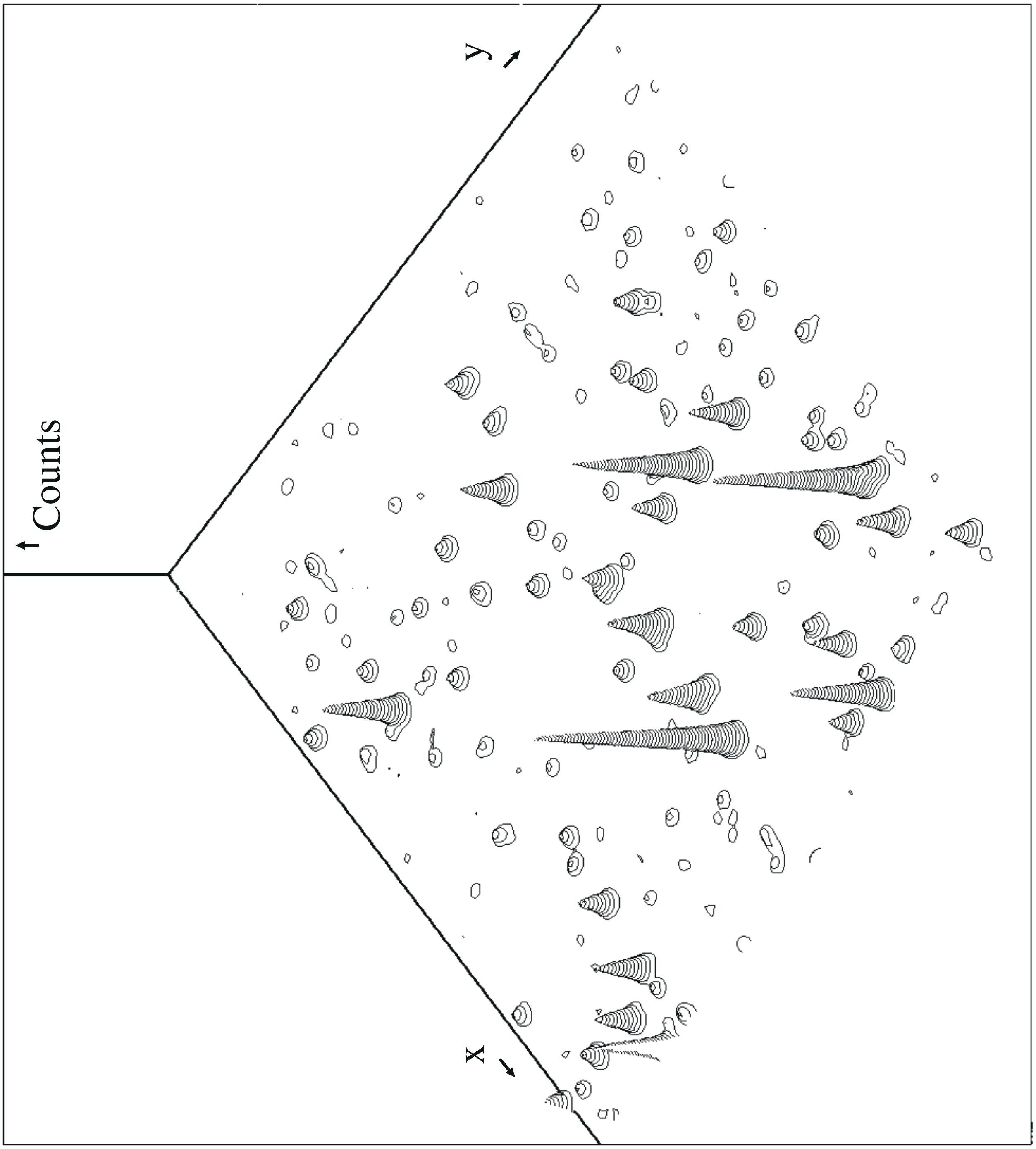

Fig.11 Original two dimensional γ-ray spectrum (180x180 channels).

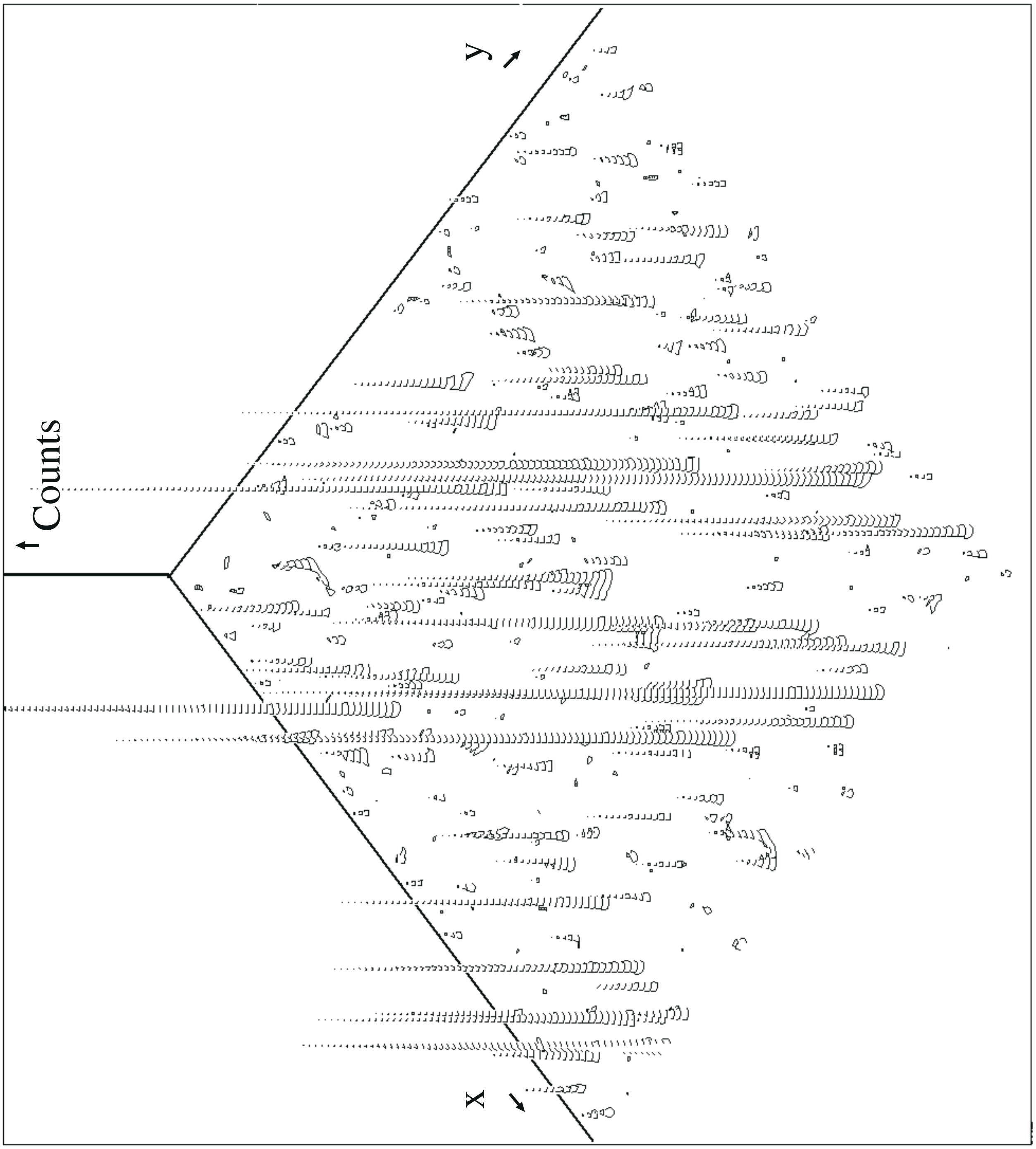

Fig.12 Spectrum from Fig. 11 after Gold deconvolution shown in contour display mode.

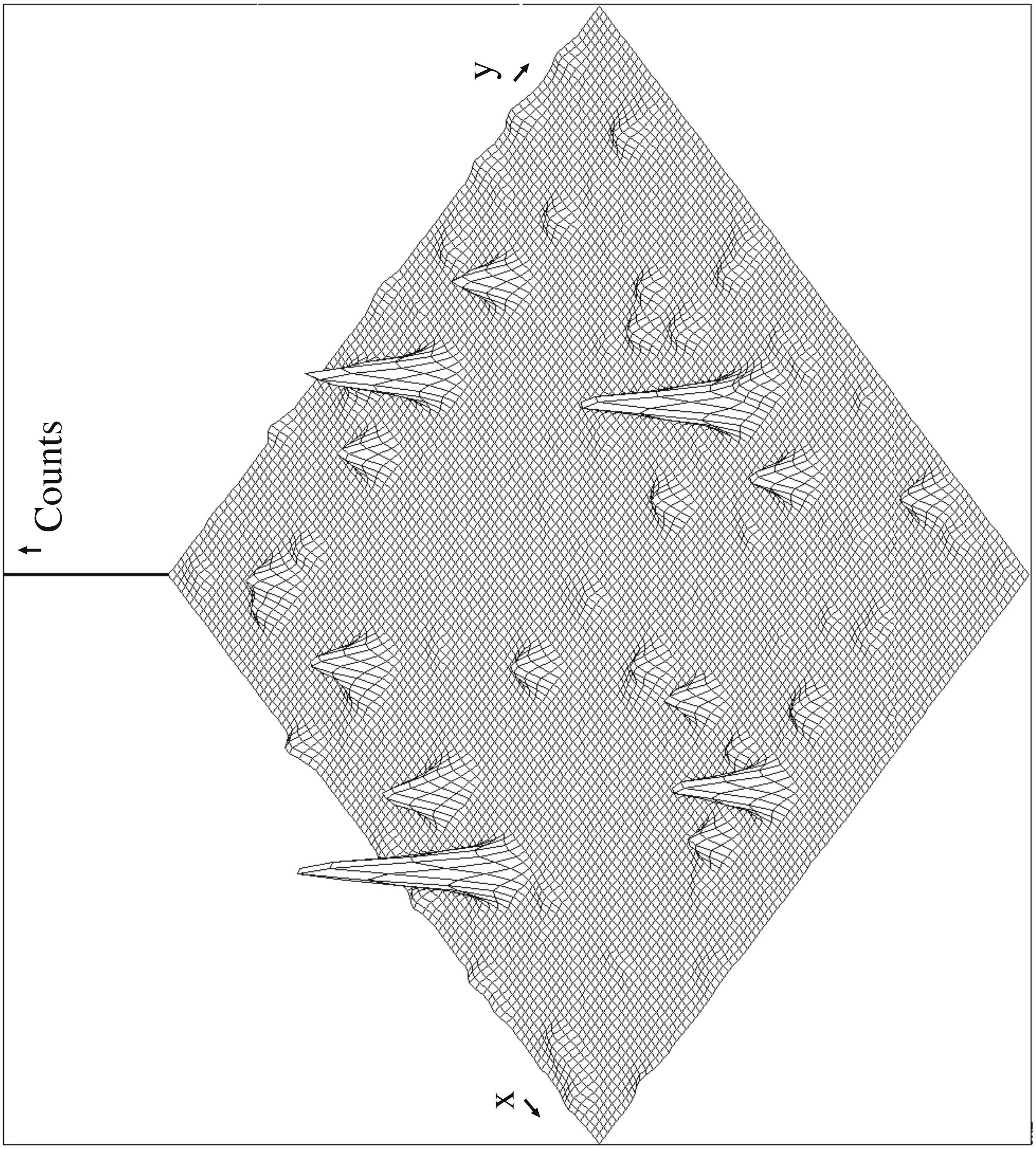

Fig.13 Detail of the spectrum from Fig. 11 (channels 80-180 in both dimensions).

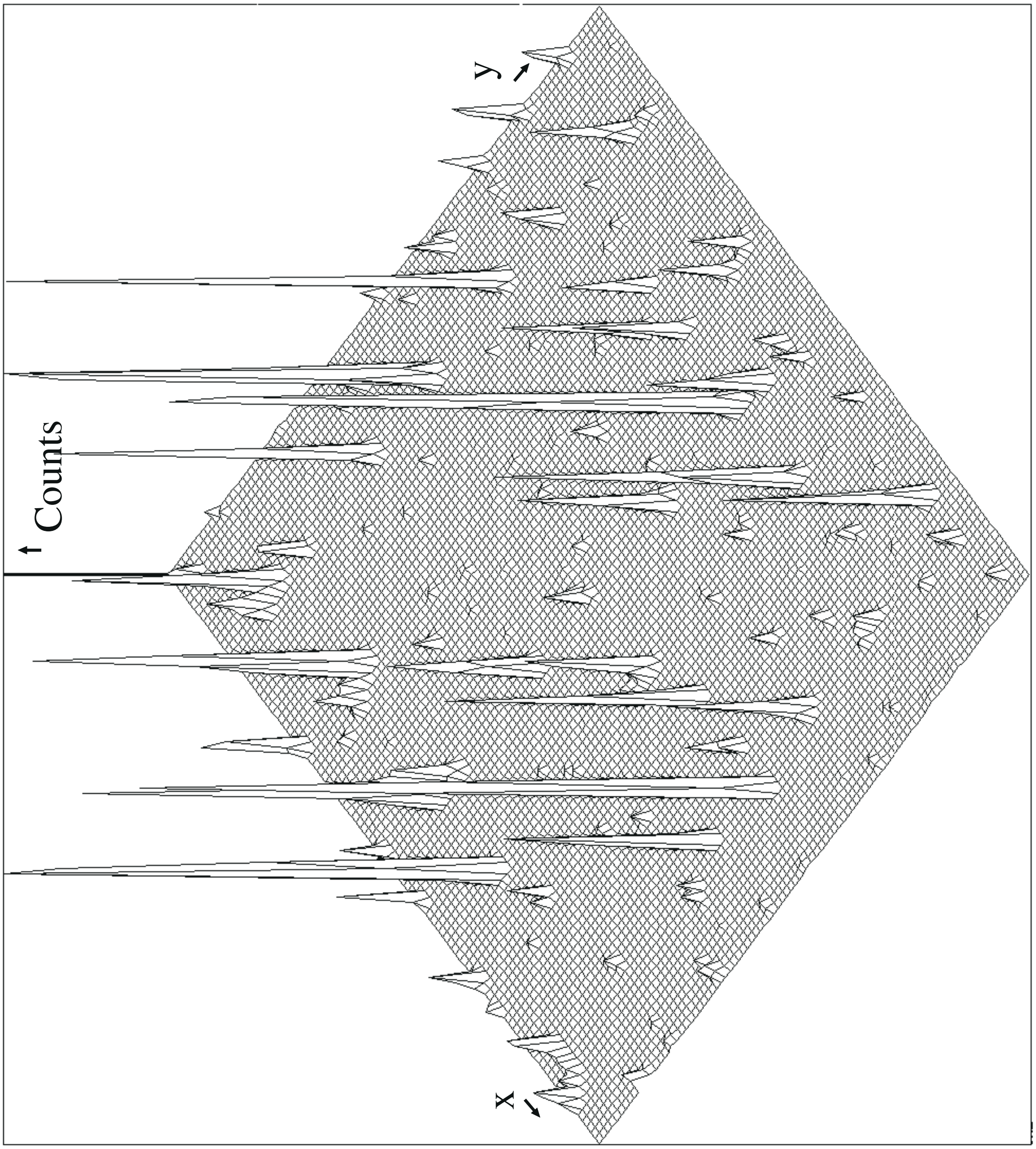

Fig.14 Detail of deconvoluted spectrum from Fig. 12 (channels 80-180 in both dimensions).

Table captions:

1. Number of multiplications for optimized and non-optimized one dimensional Gold deconvolution.
2. Memory requirements to store matrix C in optimized Gold algorithm.
3. Number of multiplications for optimized and non-optimized two dimensional Gold deconvolution.

Table 1: Number of multiplications for optimized and non-optimized one dimensional Gold deconvolution

	Optimized	Non-optimized
vector $\mathbf{b}$	$\frac{(L^2+L)}{2}$	$N^2(N+L-1)$
matrix C	$\frac{(2L^3+9L^2-6)}{6}$	N^3
vector $\mathbf{p}$	$N \cdot L$	$N(N+L-1)$
vector $\mathbf{y}'$	$N(2L-1)-L^2+L$	N^2
vector $C \cdot {}^k\mathbf{x}$	$N(4L-3)-4L^2+6L-2$	N^2

Table 2: Memory requirements to store matrix C in optimized Gold algorithm

	Total number	Size
vector $\mathbf{c}$	$2L_2 - 1$	$4L_1 - 3)$
vector $\mathbf{d}$	$(L_2 - 1) \cdot (L_2 - 1)$	$4L_1 - 3$
matrix cd	$2L_2 - 1$	$(L_1 - 1) \cdot (L_1 - 1)$
matrix dd	$(L_2 - 1) \cdot (L_2 - 1)$	$(L_1 - 1) \cdot (L_1 - 1)$
matrix de	$(L_2 - 1) \cdot (L_2 - 1)$	$(L_1 - 1) \cdot (L_1 - 1)$

Table 3: Number of multiplications for optimized and non-optimized two dimensional Gold deconvolution

	Optimized	Non-optimized
vector $\mathbf{b}$	${L_1}^2 \frac{({L_2}^2+L_2)}{2}$	${N_1}^2(N_1+L_1-1)\cdot$ $\cdot {N_2}^2(N_2+L_2-1)$
matrix C	$\frac{L_2}{18}\left[(2{L_1}^3-3{L_1}^2+L_1)\cdot\right.$ $\cdot(5{L_2}^2-3L_2+1)+3(2L_1-1)^2\cdot$ $\left.\cdot({L_2}^2+12L_2-7)\ \right]$	${N_1}^3 \cdot {N_2}^3$
vector $\mathbf{p}$	$N_1 N_2 L_1 L_2$	$N_1(N_1+L_1-1)\cdot$ $N_2(N_2+L_2-1)\cdot$
vector $\mathbf{y}'$	$[N_1(2L_1-1)-{L_1}^2+L_1]\cdot$ $\cdot[N_2(2L_2-1)-{L_2}^2+L_2]$	${N_1}^2 \cdot {N_2}^2$
vector $C \cdot {}^k\mathbf{x}$	$[N_1(4L_1-3)-4{L_1}^2+6L_1-2]\cdot$ $\cdot[N_2(4L_2-3)-4{L_2}^2+6L_2-2]$	${N_1}^2 \cdot {N_2}^2$