Journal of Computing and Information Technology - CIT 7, 1999, 2, 153–163

An Efficient Implementation of the Iterative ML-EM Image Reconstruction Algorithm for PET on a Pentium PC Platform

George Kontaxakis[1], Ludwig G. Strauss[1] and George S. Tzanakos[2]

[1] Division of Oncological Diagnostics and Therapy, Medical PET Group - Biological Imaging, German Cancer Research Center (DKFZ), Heidelberg, Germany
[2] Physics Department, Division of Nuclear and Particle Physics, University of Athens, Greece and Biomedical Engineering Department, Rutgers University, New Brunswick, NJ, USA

The EM (Expectation-Maximization) algorithm is becoming more and more popular as a solution to the image reconstruction problem in Positron Emission Tomography (PET). However, as an iterative method, it shows high computational cost in terms of the time required to complete the reconstruction procedure and the computer memory needed (main memory and disk space) for the storage of the weight coefficients (probability matrix). These were the main problems, which impeded the practical application of this promising method in the modern PET units. In principle, the conventional filtered backprojection algorithms are still in use, although the EM algorithm and the other maximum likelihood estimation (MLE) techniques of the same kind have been known and extensively studied during the past 15 years. An efficient implementation of the ML-EM algorithm is presented here, for a low-cost PC Pentium platform running Windows NT, which can be applied to any PET system configuration, without major modifications. For the first time an iterative reconstruction algorithm for emission tomography is brought down to the PC level. A detailed description of the implementation of the algorithm is given here. Emphasis is given on the calculation of the transition (probability) matrix and its efficient implementation using sparse matrix techniques. A practically feasible implementation of the EM algorithm is the final result of this work, with optimal performance on the common PC systems available today and producing tomographic reconstruction in clinically meaningful times. The implementation of the various methods proposed to further improve the results obtained by the EM algorithm, such as acceleration methods (i.e., ordered-subsets EM), and other Bayesian, maximum entropy, etc., reconstruction techniques can be also developed and performed following the same principles described here.

Keywords: Positron Emission Tomography, Image Reconstruction, Maximum Likelihood Estimation, Expectation Maximization (EM) Algorithm, Sparse Matrix, PC Pentium.

Introduction

The use of PCs is becoming more and more popular among physicians. The modern PC Pentium platforms, with the recently widespread multitasking operating systems and numerous software packages for medical applications, combine excellent computational capabilities at a low cost. In nuclear medicine, the use of computers is a necessary and indispensable tool for the processing and further analysis of large sets of data collected by the different kinds of cameras. It is shown here that even the processing and the iterative reconstruction of PET data, a task that was up to now the exclusive privilege of higher-cost workstations, can be efficiently performed with a Pentium PC, right over the desk of the physician. Implementation of the PET image reconstruction algorithm presented here is part of a larger on-going project undertaken by the Medical PET Group at the German Cancer Research Center, within the framework of the development of a complete PET data processing tool, which will include sinogram correction, iterative image reconstruction (IIR) and image quantification modules. To the knowledge of the authors, this is the first reported implementation of an IIR algorithm for a high-resolution multi-ring commercial PET tomograph on a PC

Pentium platform running a standard operating system such as Windows NT.

During the last decade, mathematical models for image reconstruction in PET have been developed, which take into account the Poisson nature of positron emissions and the statistical characteristics of the generation of the annihilation events inside the source. In mathematical statistics there is a general iterative method known as the *Expectation - Maximization* (EM) *Algorithm*, presented by Dempster et al., (Dempster et al, 1977) in its full generality. The EM algorithm is applied in Emission Tomography (Shepp and Vardi, 1982; Lange and Carson, 1984) as an iterative technique for computing maximum likelihood estimates of the activity density parameters. In this approach to the problem of image reconstruction, the measured data are considered to be samples from a set of random variables, whose probability density functions are related to the object distribution according to a mathematical model of the data acquisition process. Using the mathematical model it is possible to calculate the probability that any initial distribution density in the object under study could have produced the observed data. In the set of all possible images, that represent a potential object distribution, the image having the highest such probability is the *maximum likelihood estimate* of the original object.

On a typical PET tomograph there is an array of discrete detector elements placed around the detector ring which contains the γ-ray source, and the pair of photons produced in an annihilation event are detected in coincidence by a pair of detector elements that define a cylindrical volume, or *detector tube* (Vardi et al, 1985). The set of data collected in a PET scan is represented by the vector $\mathbf{y}$, where $y(j)$ is the total number of coincidences counted in the j^{th} detector tube and J is the total number of detector tubes. The measured coincidence events also include scattered and accidental coincidences and, in addition, not all the events produced inside the source are detected, because of tissue attenuation or photon traveling paths that do not intersect the detector ring and thus pass undetected.

The problem of image reconstruction in PET is to estimate the true unobserved counts $\mathbf{x_{true}}$ inside the source using the vector of the measured data $\mathbf{y}$. The EM algorithm provides an iterative formula to solve the above problem, by maximizing the probability to observe the given counts in the coincidence detector tubes if the true activity distribution in the source is $\mathbf{x}$, under a Poisson probability model for the positron emissions (Shepp and Vardi, 1982). This probability is expressed by the likelihood function:

$$L(\mathbf{x}) = P(\mathbf{y}|\mathbf{x}) = \prod_{j=1}^{J} e^{-\tilde{y}(j)} \frac{\tilde{y}(j)^{y(j)}}{y(j)!} \qquad (1)$$

where $\tilde{y}(j)$ is the backprojection of the estimated image vector $\mathbf{x}$ to the space of the measured data:

$$\tilde{y}(j) = \sum_{i=1}^{I} x(i)a(i,j) \qquad (2)$$

and $a(i,j)$ describes the probability for an annihilation event that occurred in the area of the source covered by the i^{th} pixel to be detected by the j^{th} tube. Application of MLE techniques to Eq. (1), leads to the EM update of the i^{th} pixel at the $(k+1)^{\text{th}}$ iteration according to the following multiplicative pixel-updating scheme:

$$x^{(k+1)}(i) = x^{(k)}(i) \cdot C^{(k)}(i) \qquad (3)$$

where $C^{(k)}$ is the vector of the correction factors applied to the image, defined by the ratio of the measured projections and the backprojection of the current estimate of the image vector at the k^{th} iteration:

$$C^{(k)}(i) = \sum_{j=1}^{J} a(i,j) \cdot \frac{y(j)}{\tilde{y}^{(k)}(j)}. \qquad (4)$$

The EM algorithm has been found to produce better results, i.e., better contrast, S/N ratio and resolution (Shepp et al, 1984; Llacer et al, 1993), than other well established reconstruction techniques, such as filtered backprojection. Its properties of non-negativity and self-normalization (Shepp and Vardi, 1982), are considered the major advantages of this methodology (Lewitt and Muehllehner, 1986), as well as the practical ability of the ML-EM techniques to incorporate in the probability matrix $\mathbf{A}$ several physical factors (Lange and Carson, 1984), such as attenuation, scatter and accidental coincidence corrections, time-of-flight, positron range and angulation information, etc. As an iterative technique, however, it requires a lot of computational effort and shows a relatively slow convergence rate. As the use of the ML-EM image reconstruction algorithms is becoming more and more popular in clinical practice (Doll et al, 1995), an efficient implementation of the EM algorithm on a conventional, low-cost PC Pentium platform is presented here, which is able to produce images in a clinically meaningful time and without the need of dedicated parallel vector machines or other hardware requirements, which, however, have been already presented as solutions to the problem of the high computational complexity of the algorithm.

Parallel and Hardware Implementations

Methods to overcome the problem of the high computational demand of the EM algorithm have been proposed, using parallel machines or specialized hardware design. Therefore, images can be produced with the use of specially designed hardware that exploits the structure of this algorithm by means of a high degree of parallelism and pipelining (Klotz and Snyder, 1988). A dedicated VLSI architecture has also been proposed (Jones, et al, 1988) towards this direction. It is true that the EM reconstruction algorithm, as expressed by Eqs. (2)-(4), can be easily parallelized, since the update of each pixel in one iteration does not depend on the updated values of the other elements of the image vector.

In a parallel implementation of the EM algorithm, data parallelism schemes can be used and are of three types (Chen et al, 1991): The *partition-by-box* scheme, in which one processing unit (PE) is assigned all the computation associated with a box (pixel or voxel) from the image vector in both forward and backward projection steps. In the *partition-by-tube* scheme one PE is assigned all tasks and data associated with one tube from the data vector in both steps, and the third scheme is the *partition-by-tube-and-box*, where the partition-by-tube is used for the forward projection and the partition-by-box is used for the backprojection. Ideas on parallel implementation of the EM reconstruction technique have been proposed as early as 1985 (Llacer and Meng, 1985) and preliminary results on the application of the EM algorithm on parallel machines, especially in attempts to perform 3-D image reconstruction in PET, have been published since the late '80s (Miller et al, 1988; Hebert and Leahy, 1989; Herman et al, 1990).

The main disadvantage of these implementations is that parallel programming for dedicated architectures is highly platform-dependent. Recently, the idea of distributed processing on a cluster of workstations seems to become more and more popular. Sharing the workload between 8 and 16 CM-5 processors and (in a separate experiment) the same number of SPARC5 Workstations (Olesen et al, 1994), as well as among 8 SPARCstation 10/40 class CPUs (Fricke, 1996) has been reported to significantly reduce the reconstruction time for dynamic studies. A parallel implementation of the algorithm on a heterogeneous workstation cluster has been recently presented (Zaers et al, 1997), where the computational load is distributed to the workstations available within the clinical environment.

Instead of occupying the CPUs of all the computing systems available and developing interfaces for the distribution of the computational load to the different machines or developing software for highly specialized parallel architectures, a much simpler implementation of the EM algorithm is presented here. The proposed computing system is a standard low-cost Pentium PC, running Windows NT and with adequate RAM, which will be dedicated for the iterative reconstruction of PET data. The ability of the Windows NT to address more than one Pentium processor makes this idea even more appealing, since a multiprocessor Pentium platform will allow a certain degree of parallelism for the EM reconstruction, without any significant increase in cost or major modification of the basic implementation. The reconstruction module on the PC server will accept Java scripts with the reconstruction parameters and will automatically initiate the reconstruction of the data, under the implementation scheme presented in the following sections (Kontaxakis et al., 1998). The completed module will perform all corrections to the raw data (attenuation, scatter, normalization, etc., corrections) on the PC and will also include routines for the quantification and parametrization of the reconstructed images.

The Probability Matrix

For the study described here a model of the CTI ECAT EXACT HR+ tomograph is created and its basic configuration is used to calculate the values of the probability matrix. This tomograph delivers images in 63 planes (32 direct and 31 cross-planes) and has an axial field-of-view of 15.5 cm. It is constructed using 4 rings of 72 8 × 8 BGO detector blocks. Each of its 32 rings consists of 576 individual detector crystals, each of dimensions $4.39 \times 4.05 \times 30\ \mathrm{mm}^3$ and images a transaxial field-of-view with diameter of 58.3 cm. Each plane delivers 82944 lines-of-response (LORs) or detector tubes. A standard angular compression method (mashing) of 2 reduces this number by half.

According to Shepp and Vardi (Shepp and Vardi, 1982), the choice in the calculation method for the matrix $a(i,j)$ is not critical and does not affect significantly the final result, in contrast to the algebraic-reconstruction-type iterative reconstruction techniques, where the choice and the calculation of the transition rule is of critical importance. However, some discussion on the above argument has been raised (Llacer, et al, 1986) during the first practical implementations of the EM algorithm in PET image reconstruction and the experience of the authors also showed that an accurate choice of the probability matrix is of significant importance to the final reconstructed image quality.

The calculation of the probability matrix $a(i,j)$ is based on the general observation that the matrix depends on the geometry of the tomograph. In this implementation there is no use of approximate formulas such as the averaging of the angles of view (Vardi et al, 1985; Chen et al, 1991). Instead, a Monte Carlo method for the calculation of the values $a(i,j)$ has been used. Similar techniques in the calculation of the probability matrix have been used by Veklerov and colleagues (Veklerov et al, 1988) who found that, in general, Monte Carlo calculation of the matrix $a(i,j)$ is more accurate than techniques based on the angle of view or other methods. According to the procedure followed here, in the area of the box i a sufficient number of lines is generated uniformly in the geometry of the tomograph and the number of γ-ray pairs (represented by these lines) in each detector tube are recorded. Then, the probability $a(i,j)$ is given by the expression:

$$a(i,j) = \frac{N_j}{N_{tot}} \tag{5}$$

where N_{tot} is the total number of lines produced in the area of the box i and N_j is the number of these lines that intersect the detector tube j. The accuracy of this method depends on the total number of annihilation events (lines) generated in each box. The accuracy of the reconstructed image quality versus N_{tot} has been carefully studied, since it has been observed that as N_{tot} increases, the number of non-zero elements in the probability matrix increases rapidly, and therefore the quality of the produced images is not the best achievable at that stage. After a

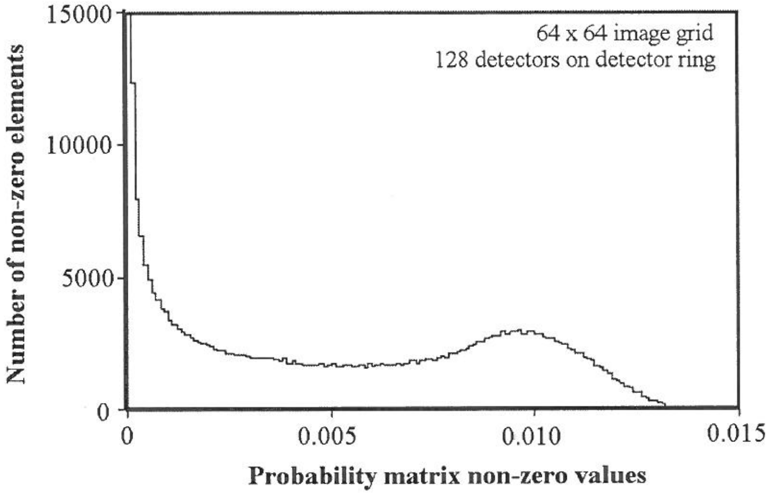

Fig. 1. Distribution of the non-zero values in the probability matrix, for a model of a head tomograph with 128 detector crystals, tomograph diameter of 30 cm, patient port of 20 cm and a 64 × 64 image grid.

certain value for the N_{tot}, which depends on the system configuration, the number of non-zero values reaches a plateau phase and stops increasing with N_{tot}. After that point, the produced matrix $a(i,j)$ has no visible effect on the reconstructed image quality.

The dimensions of $a(i,j)$ are $I \times J$ (*number of pixels in the image grid* $\times$ *number of detector tubes*). For an image grid of 128×128 pixels including the whole patient port, $I = 16384$, and for a mashing factor of 2, $J = 41472$. The matrix $a(i,j)$ has therefore a total of $I \times J \approx 6.8 \times 10^8$ elements. However, it is clear that from these elements, the non-zero ones are these for which the box i is intersected by the detector tube j, and $a(i,j) = 0$ if box i does not intersect tube j, therefore $a(i,j)$ is a sparse matrix. For the tomograph configuration used, the number of non-zero elements was found to be about 8.15×10^6 (more than 98% of the total number of elements of the matrix $a(i,j)$ are zero). Fig. 1 shows a typical distribution of the values in the probability matrix, for a model of a one-ring tomograph, with 128 detector crystals on the ring and an image grid of 64×64 pixels. It is interesting to observe that there is a considerable number of elements in the probability matrix with very small values, which correspond to elements $a(i,j)$ where the i^{th} box has only a small intersection area with the j^{th} detector tube.

This Monte Carlo calculation of $a(i,j)$ takes into consideration the variable width of detector tubes as one moves away form the center of the image, but assumes perfect detectors otherwise. Exclusive use of the geometrical characteristics of the system in the calculation of the probability matrix $a(i,j)$ could possibly lead to a significant reduction in the number of the non-zero elements of the matrix that need to be stored, if one considers the symmetry of a detector ring and an image grid centered in it. Kaufman (Kaufman, 1987) showed that there could be a 4- or even an 8-fold symmetry in such a system and a specific value is the same for 4 matrix elements, associated with different pixels and detector tubes and related with simple geometrical rules dictated by the symmetry of the tomograph layout: for example, rotating the system about the central point of the ring by 90°, 180° and 270°, one will find the same probabilities, but associated with different boxes and detector tubes.

The above could be proven very useful, especially for very large configurations and 3- D systems, where a parallel implementation of the EM algorithm is to be developed. However, in this implementation the symmetry of the tomograph model was selected not to be used, in order to present a reconstruction method that could be easily expanded and applied to systems without circular symmetry. For a system with an 8-fold symmetry, the number of detectors in one ring has to be a multiple of 8, the image grid should be a square with an even number of pixels on each side and there should be no wobbling or other motion that would break the system's symmetry. In addition, the image plane has to be centered in the detector ring and any attempt to move the field-of-view (FOV) off-center would require not only a new calculation of the probability matrix, but also a totally new implementation of the EM reconstruction algorithm. When other information is to be included in the probability matrix, such as scatter and attenuation correction information, the concept of symmetry does not apply. The use of the (possible) system's symmetries is therefore an excellent method to save space and memory, however, such an implementation is limited not only to that particular system and data processing protocol used, but it also limits the FOV to the center of the tomograph. In addition, the unfolding of the matrix would also increase the computational effort with the necessary decoding calculations.

Implementation of the EM Algorithm

The size and the sparsity of the probability matrix $a(i,j)$ are the basic features that require most of the attention and effort for an efficient implementation of the algorithm. During one iteration each of the matrix elements is used twice:

- The forward projection of the reconstructed image vector $\mathbf{x}$ to the data space requires a pass through the probability matrix along each column j.
- The calculation of the updating coefficients C in each iteration requires a pass through the probability matrix along each row i.

An efficient implementation of the algorithm would therefore require a fast scanning through

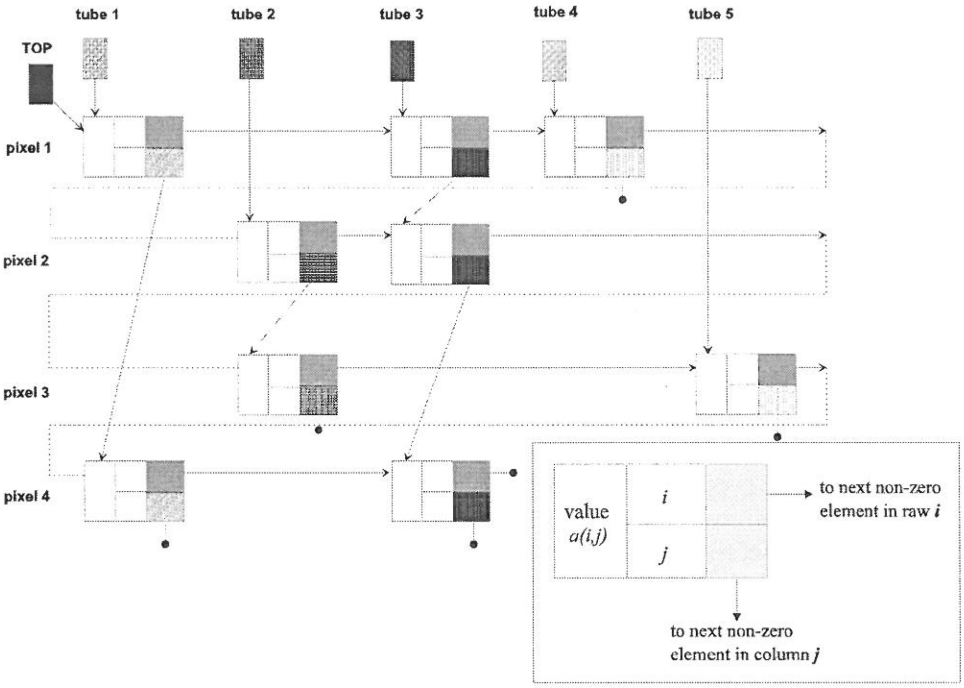

Fig. 2. Diagram of the sparse-matrix representation (two-dimensional linked list) of the probability matrix $a(i,j)$. A floating point record is reserved for the value of each non-zero element and two integer records contain its i (pixel number) and j (detector tube number) coordinates. Two pointers show to the next non-zero element in the i and j directions, permitting fast scanning of the matrix in both directions. The *TOP* pointer is always located at the first element of the matrix and an array of pointers is located at the top of each column; each element is linked to the following one in each column and row; the row pointer connects the last element in each row with the first element of the next row; the pointer of the last element in each column shows to NUL.

$a(i,j)$ row-wise and column-wise in each iteration and sparse matrix techniques have to be used. A two-dimensional linked list structure was recently proposed (Kontaxakis, 1996) for the representation of the matrix non-zero elements: a floating point record is reserved for the value of each element $a(i,j)$ and two integer records contain its i (pixel number) and j (tube number) coordinates. Two pointers show to the next non-zero element in the i and j direction, permitting a fast scanning of the matrix in both directions.

Fig. 2 shows a schematic example of the sparse matrix structure used for loading $a(i,j)$. This diagram corresponds to an oversimplified model of a PET system with 5 lines-of- response and 4 pixels in the image matrix. Each element is linked with the next non-zero one for the same row (pixel) and column (tube or LOR). The link that connects all elements in the structure is the row link (at the end of each row the next element in the list is the first one in the next row). In order to carry out fast the backprojection step of Eq. (2), a matrix of links is placed to show at the top of each column. These links are called during this step and another pass of the whole structure suffices to perform the forward projection step. With this technique, each element is visited exactly twice (as needed) during each iteration, without having to search within the matrix structure for the elements needed at each step.

This is a simple but efficient approach to solve the problem of the long computation time required for this iterative algorithm. The computation of the matrix elements is done once and before the reconstruction procedure, as described in the previous sections. No additional computation is performed on the matrix elements during the EM reconstruction. Encoding schemes (such as using the same integer to store both coordinates of an element) could have been used to reduce the amount of memory required, however the additional computation required for decoding purposes, each time a matrix element is used, would significantly increase the computational complexity of the method (Kontaxakis, 1996).

It is clear that this implementation trades off memory for speed, an idea recently also proposed by Miller and Wallis (Miller and Wallis, 1992), who did not, however, take full advantage of the sparsity of the probability matrix. Today, large quantities of memory can be purchased at low cost and are easily upgraded in conventional computing systems, whereas speed deals with hardware considerations and the architectural limits of the computer used, which cannot be improved unless an expensive investment on equipment is undertaken.

Since all the elements of the matrix $a(i, j)$ are visited during the calculation of the updating coefficients C according to the sequence defined by the pointer to the next non-zero element in row i, one can eliminate the use of this pointer if each element $a(i, j)$ is placed in a static one-dimensional array of the size equal to the number of the non-zero elements. Since this implementation requires the calculation beforehand of the non-zero $a(i, j)$ elements, their number is known and therefore a static array can be defined. The pointer to the next non-zero element in each column however is still necessary to efficiently perform the forward projections of Eq. (2). For the implementation of the EM algorithm for the ECAT EXACT HR+ tomograph, with a standard acquisition protocol employing an angular compression (mashing) of 2, there are 41472 LORs and therefore a 16-bit (unsigned short) integer can be used for the indexing of the detector tubes in each slice. The same holds for the pixel indexing, up to an image grid of size 256 × 256. The values of the $a(i, j)$ elements of the probability matrix are bounded between 0 and 1.0, therefore these values can also be encoded in a 2-byte integer format by the use of an adequate multiplicative coefficient, without significant loss in accuracy. The pointer can only be represented with a 32-bit integer. The total size of the structure is now 10 bytes per non-zero element, however for a Windows NT implementation, this practically means 12 bytes per element. Fig. 3 shows a diagram of the static array implementation proposed here.

```
typedef struct probability_matrix {
  unsigned short i,j,value;
  int            next_j;
}
```

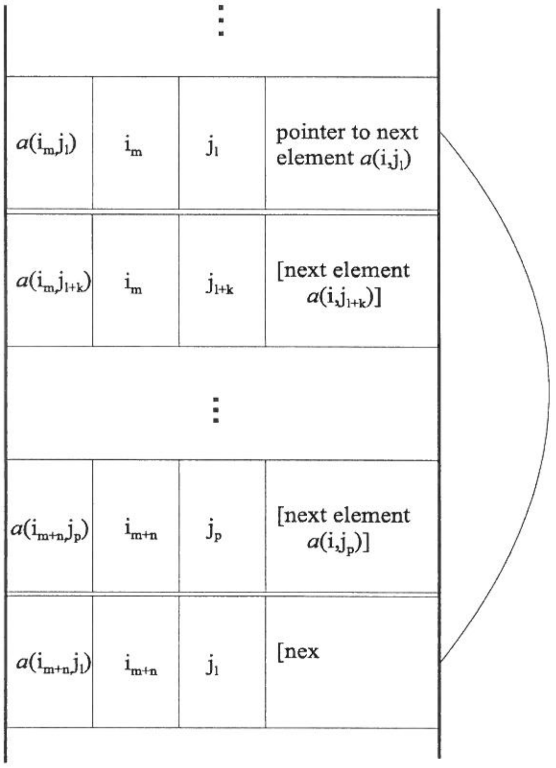

Fig. 3. Diagram of the static array implementation for the sparse matrix representation of the probability matrix proposed here. The value of each non-zero element and its coordinates in the $a(i, j)$ matrix are stored in 16-bit integers. The need of the row pointer of the two-dimensional linked list in Fig. 2 is eliminated and a 32-bit integer stores the memory location of the next element for each detector tube.

Image grid (whole FOV)	Size of probability matrix (in MB)	Time required for one EM iteration (in sec) on a Pentium PC* (200 MHz, 256 MB)	Time required for one EM iteration (in sec) on a SPARCserver 1000 system
64× 64	30	2.6	8.8
96× 96	60	5.4	14.0
128× 128	90	9.1	22.9
192× 192	170	16.6	43.0
256× 256	220	21.0	65.8

Table 1. This table shows the size (in MB) of the probability matrix represented in memory according to the implementation described in Fig. 3 and the computation time (average over 100 iterations) required for one EM iteration for different sizes of the image grid. The size of the sparse matrix depends on the accuracy with which it has been calculated with the Monte Carlo procedure described.
* using one Pentium processor

Table 1 shows the execution (user and system) time required for one EM iteration and for different sizes of the image grid for the PET tomograph used. The results were obtained from the implementation of the algorithm on a PC Pentium Pro (200 MHz, 256 MB RAM) platform. At the same table the memory required for implementing the EM algorithm for each configuration is also reported. The memory requirements reported here correspond to the minimum memory necessary for the setup of the static-array implementation. The same implementation was tested for comparison on the RCI (Research Computing Initiative) three- SPARC-server 1000 cluster at Rutgers University, running Solaris 2.5 and with adequate memory in order to avoid memory paging. The corresponding execution times per iteration are also given in Table 1. For the comparison, one should take into account that RCI is a Unix multi-user system, however the experiments were performed during night-time off-peak hours. The EM algorithm runs approximately three times faster on a dedicated Pentium Pro platform than on this Unix system.

To achieve fast EM reconstruction sufficient memory must be available in the computing system used. It is interesting to compare the results presented in Table 1 with the ones recently presented by Rajan and colleagues (Rajan et al, 1995), where a simpler PET tomograph model was used (128 detectors on the detector ring) and the EM algorithm was implemented on an IBM 6000 RISC workstation. For an EM reconstruction on a 64 × 64 image grid the authors in (Rajan et al, 1995) report an execution time of 2.49 seconds for one iteration, which is comparable to the time reported in Table 1. However, for a reconstruction on a 128 × 128 grid (which corresponds to a 4-fold increase in the image vector dimension) they report a 10-fold increase in the execution time (21.48 sec), whereas in this implementation only 9.1 seconds are needed. However, such an observation should not be taken as a direct comparison on the performance of both implementations, since no sufficient information of the exact configuration used in the experiments described in (Rajan et al, 1995) is available. The size of the field-of-view and the method for the calculation of the probability matrix are of crucial importance on the number of non-zero elements and they directly affect the execution time. In addition, no information on the exact memory requirements and management of that implementation has been disclosed.

It is important to remark that such an implementation can be universally applied to any PET system, independently of the tomograph design, the image grid and the location of the field-of-view, and can be applied either in 2D or a 3D reconstruction of PET data. Each element of the (one-dimensional) image vector $\mathbf{x}$ is numbered according to a certain protocol. The same procedure should be followed for the data vector $\mathbf{y}$. Therefore, it makes no difference for the reconstruction procedure if the data acquisition was performed for a 2D or a 3D reconstruction, and the inherent properties of the EM algorithm for such an implementation will remain the same for any configuration and any system design. A fully portable implementation of the EM algorithm for any PET system is presented here. The steps to be taken for adapting this algorithm

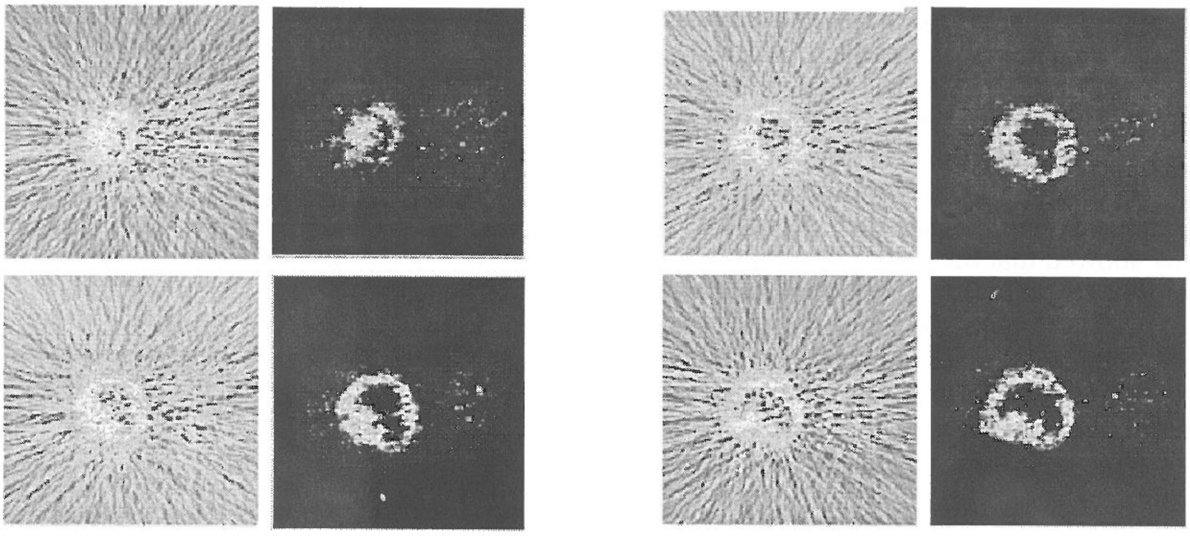

Fig. 4. Four slices (128 × 128 matrix) of a soft-tissue necrotic sarcoma of the right leg of a patient. Left: reconstructed images with filtered-backprojection (ramp filter with 0.5 cutoff frequency). Right: reconstructed images with the implementation of the EM algorithm presented here and the stopping criterion presented in (Kontaxakis and Tzanakos, 1996).

to a particular PET tomograph and acquisition protocol are the following:

1. Create a numbering protocol for the elements of the image grid and the data vector and sort the projection data according to it.
2. Calculate the probability matrix according to the system's geometrical characteristics and the numbering protocol used.

For this implementation, the sinogram data should be pre-corrected for random coincidences, attenuation, scatter, detector efficiencies and isotope decay. Arc correction is not necessary, as the Monte Carlo calculation of the probability matrix takes into account the variable width of the detector tubes for each view. Negative entries in the sinogram vector that occur after corrections are set to zero.

Fig. 4 shows four slices of a necrotic soft-tissue sarcoma of the right leg of a patient. The images are reconstructed using the filtered-backprojection algorithm provided by the manufacturer of the tomograph and the implementation of the EM algorithm presented here, using the stopping criterion described in (Kontaxakis and Tzanakos, 1996). Due to the low-count statistics in the sinogram data, there are strong reconstruction artifacts in the filtered-backprojection images. The iteratively reconstructed images on the other hand provide a much clearer picture on the activity distribution and even the left leg is there much better delineated. Such efficient implementation of the iterative EM algorithm is therefore expected to provide images of higher diagnostic value by reducing the artifacts that often lead to false-positive results (Strauss, 1996).

Conclusions

The class of maximum likelihood reconstruction algorithms has been briefly discussed, with emphasis on the EM (Expectation Maximization) algorithm. This method has several attractive features: the non-negativity of reconstructed images, preservation of the total number of counts in the image and convergence of the reconstructed images to a unique maximum likelihood estimate. It also allows direct incorporation to the reconstruction model of many physical factors that are fundamental in emission tomography, namely the Poisson nature of the positron annihilations, otherwise neglected. Moreover, the EM algorithm tends to reduce the statistical noise artifact over filtered backprojection algorithms.

A major disadvantage in the practical implementation of the EM algorithm in image reconstruction in PET, is the fact that it is a very computationally demanding method, requiring times to produce an image, on a conventional serial computer, that up to now have been impractical for the routine clinical practice. The computational needs of the EM algorithm in execution time and main memory have been extensively studied. For the first time a detailed implementation of the EM algorithm on a low-cost PC Pentium platform has been presented. This study showed that the use of low-cost memory can provide fast execution and produce results in clinically meaningful times, thus alleviating the argument of the computational burden of the method, which prevented the extensive use of this new and promising reconstruction algorithm in today's modern PET systems.

References

[1] C–T. CHEN, X. OUYANG, W.H. WONG, X. HU, V.E. JOHNSON, C. ORDONEZ AND C.E. METZ, Sensor fusion in image reconstruction, *IEEE Trans. Nucl. Sci.*, **38**(2), 687–692, April 1991.

[2] A.P. DEMPSTER, N.M. LAIRD AND D.B. RUBIN, Maximum likelihood from incomplete data via the EM algorithm, *J. Roy. Stat. Soc.*, Ser. B, **39**, 1–38, 1977.

[3] J. DOLL, H.J. OSTERTAG, M.E. BELLEMANN, P. SCHMIDLIN, W.K. KÜBLER, L.G. STRAUSS AND W.J. LORENZ, Effects of distorted PET projection data on the reconstructed image using different reconstruction algorithms, *Radioactive Isotopes in Clinical Medicine and Research – Advances in Pharmacological Sciences*, Birkhäuser Verlag Basel, pp. 85–90, 1995.

[4] H. FRICKE, Iterative image reconstruction of dynamic PET studies, Acceleration through data compression and distributed processing, 1996 ECAT Technical Users' Meeting, Stockholm, April 1996.

[5] T. HEBERT AND R. LEAHY, A generalized EM algorithm for 3–D Bayesian reconstruction from Poisson data using Gibbs Priors, *IEEE Trans. Med. Imag.*, **8**(2), 194–202, June 1989.

[6] G. HERMAN, D. ODHNER, X. TOENNIES AND S. ZENIOS, A parallelized algorithm for image reconstruction from noisy projections, *Proc. Workshop Large Scale Optimization*, T. Coleman and Y. Li, Eds., Philadelphia, SIAM, 1990.

[7] W.F. JONES, L.G. BRYARS AND M.E. CASEY, Positron emission tomographic images and expectation maximization, A VLSI architecture for multiple iterations per second, *IEEE Trans. Nucl. Sci.*, **35**, 620–624, 1988.

[8] L. KAUFMAN, Implementing and accelerating the EM algorithm for Positron Emission Tomography, *IEEE Trans. Med. Imag.*, **6**(1), 37–51, March 1987.

[9] H. A. KLOTZ, JR., AND D. L. SNYDER, A hardware architecture using finite – field arithmetic for computing maximum likelihood estimates in Emission Tomography, *IEEE Trans. Med. Imag.*, **7**(4), Dec. 1988.

[10] G. KONTAXAKIS, Maximum likelihood image reconstruction in Positron Emission Tomography, Convergence characteristics and stopping rules, *PhD Dissertation*, Rutgers, The State University of New Jersey, New Brunswick, NJ, USA, January 1996.

[11] G. KONTAXAKIS AND G.S. TZANAKOS, A stopping criterion for the EM-MLE image reconstruction for PET, *Proc. of the SPIE Medical Imaging 1996 Symposium*, 2710, 133–144, Newport Beach, CA, USA, Feb. 1996.

[12] G. KONTAXAKIS, L.G. STRAUSS AND G. VAN KAICK, Optimized implementation and performance evaluation of iterative image reconstruction algorithms for PET on distributed Pentium systems and a Web-based interface, *J. Nucl. Med.*, S39, 8P, May 1998.

[13] K. LANGE AND R. CARSON, EM reconstruction algorithms for Emission and Transmission Tomography, *J. Comp. Ass. Tomogr.*, **8**, 302–316, 1984.

[14] R.M. LEWITT AND G. MUEHLLEHNER, Accelerated iterative reconstruction for Positron Emission Tomography based on the EM Algorithm for maximum likelihood estimation, *IEEE Trans. Med. Imag.*, **5**(1), 16–22, March 1986.

[15] J. LLACER, S. ANDREAE, E. VEKLEROV AND E. J. HOFFMAN, Towards a practical implementation of the MLE algorithm for Positron Emission Tomography, *IEEE Trans. Nucl. Sci.*, **33**(1), 468–477, Feb. 1986.

[16] J. LLACER, E. VEKLEROV, L.R. BAXTER, S.T. GRAFTON, L.K. GRIFFETH, R.A. HAWKINS, C.K. HOH, J.C. MAZZIOTTA, E.J. HOFFMAN AND C.E. METZ, Results of a clinical receiver operating characteristics study comparing filtered backprojection and maximum likelihood estimator images in FDG PET studies, *J. Nucl. Med.*, **34**(7), 1198–1203, July 1993.

[17] J. LLACER AND J.D. MENG, Matrix-based image reconstruction methods for tomography, *IEEE Trans. Nucl. Sci.*, **32**(1), 855–864, Febr. 1985.

[18] M.I. MILLER, B. ROYSAM AND A.W. MCCARTHY, Bayesian image reconstruction for Emission Tomography, implementation of the EM algorithm and Good's roughness prior on massively parallel processors, *Intl. Conf. on Acoustics, Speech and Signal Processing*, March 1988.

[19] T.R. MILLER AND J.W. WALLIS, Fast maximum-likelihood reconstruction, *J. Nucl. Med.*, **33**(9), 1710–1711, Sept. 1992.

[20] S.P. OLESEN, J. GREGOR AND M.G. THOMASON, Parallel PET reconstruction by EM iteration with reduced processor communications, Technical Report CS–94–256, Univ. of Tennessee, Knoxville, TN, USA, Dec. 1994.

[21] K. RAJAN, L.M. PATNAIK AND J. RAMAKRISHNA, Linear array implementation of the EM algorithm for PET image reconstruction, *IEEE Trans. Nucl. Sci.*, **42**(4), 1439–1444, Aug. 1995.

[22] L.A. SHEPP AND Y. VARDI, Maximum likelihood reconstruction in Positron Emission Tomography, *IEEE Trans. Med. Imag.*, **1**, 113–122, 1982.

[23] L.A. SHEPP, Y. VARDI, J.B. RA, S.K. HILAL, Z.H. CHO, Maximum likelihood PET with real data, *IEEE Trans. Nucl. Sci.*, **31**(2), 910–913, April 1984.

[24] L.G. STRAUSS, Fluorine–18 deoxyglucose and false-positive results, a major problem in the diagnostics of oncological patients, *Eur. J. Nucl. Med.*, (23)10, 1409–1415, Oct. 1996.

[25] Y. VARDI, L.A. SHEPP AND L. KAUFMAN, A statistical model for Positron Emission Tomography, *J. Amer. Stat. Assoc.*, **80**(389), 8–20, March 1985.

[26] E. VEKLEROV, J. LLACER, E. J. HOFFMAN, MLE reconstruction of a brain phantom using a Monte Carlo probability matrix and a statistical stopping rule, *IEEE Trans. Nucl. Sci.*, **35**(1), 603–607, Feb. 1988.

[27] J. ZAERS, J. DOLL, P. SCHMIDLIN, G. BRIX AND W.J. LORENZ, [Parallel implementation of iterative reconstruction methods for PET on a cluster of heterogeneous workstations], (abstract in German), Nuklearmedizin, 36, A17, April 1997.

Received: July, 1997
Revised: June, 1998
Accepted: November, 1998

Contact address:
George Kontaxakis, PhD
German Cancer Research Center (DKFZ)
Division of Oncological Diagnostics & Therapy
Medical PET Group – Biological Imaging
Im Neuenheimer Feld 280
D–69120 Heidelberg
Germany
phone: +49 6221 422500
fax: +49 6221 422476
e-mail: g.kontaxakis@dkfz-heidelberg.de
www: http://www.dkfz-heidelberg.de/pet/home.htm

GEORGE KONTAXAKIS was born in Athens, Greece, in 1967. He studied Electrical Engineering at the National Technical University of Athens (1985-90), graduated from the University of Nancy, France (1990) under an Erasmus Fellowship and received the MS (1992) and PhD (1996) degrees in Biomedical Engineering from Rutgers, the State University of New Jersey, USA. In 1996 he was with the University Clinic of Frankfurt, Germany, as postdoctoral fellow in the Department of Nuclear Medicine. He is currently a guest scientist fellow at the German Cancer Research Center in Heidelberg. His main interest is medical imaging and especially image reconstruction algorithms, data processing and design tools for Positron Emission Tomography.

LUDWIG G. STRAUSS was born in Germany in 1949. After studying medicine and mathematics he received his M.D. 1978, the board qualifications for nuclear medicine in 1982 and for radiology in 1986 respectively. The position of an Assistant Professor was achieved 1986 and that of a Full Professor of radiology and nuclear medicine in 1992. Since 1987 he is an Associate Director of the Department of Oncological Diagnostics and Therapy at the German Cancer Research Center in Heidelberg, Germany and is heading the Medical PET Group - Biological Imaging. Image processing is one of his major research aspects, focusing on digital radiology and functional nuclear medicine procedures, mainly Positron Emission Tomography (PET). His scientific work includes 76 publications and more than 350 contributions to proceedings and books.

GEORGE S. TZANAKOS is an Associate Professor at the Division of Nuclear and Particle Physics, Physics Department, University of Athens, Greece and Visiting Professor at the Biomedical Engineering Department, Rutgers, The State University of New Jersey.