

EMICO3DR.DOC

(last update Feb 10, 1997)

CONTENTS

- A. INTRODUCTION
- B. PROGRAM INPUT
- C. PROGRAM EXECUTION
- D. EXAMPLE EMICO3DR BATCH JOB
- E. PROGRAM NOTES
- F. PROGRAM NOTES FOR SYSTEM MANAGER
- G. ADDITIONAL NOTES
- H. FLOW CHART FOR EMICO3DR PROGRAM

A. INTRODUCTION

EMICO3DR is the main program in the icosahedral image reconstruction process that is used to produce a 3D map from a set of particle images. The program first extracts the data for normal equations from each particle view, then combines the normal matrices and solves them to give big G's, which are temporarily or permanently stored in a disk file (DEFAULT = BG.DAT), then computes little g's from the big G's, and finally computes a Fourier Bessel transformation of the little g's to produce the three-dimensional density map of an icosahedral particle in the "standard" 2-fold setting (see [TSB.DOC]EMICO_SYS.DOC).

EMICO3DR replaces either of two previously used program combinations: 1) EMICOMATBG and EMICOLGFB or 2) EMICOMAT, EMICOBG, EMICOLG, and EMICOFB. EMICO3DR is normally run after EMPFT or EMICOGRA.

Aside from the obvious advantages of consolidating input/output options and making it easier to set up batch command procedures, EMICO3DR offers several advantages compared with the previous programs, especially the "famous four" (MAT, BG, LG, FB). The EMICOMAT.DAT and NIEWS *.MAT files previously produced by EMICOMAT are no longer generated, thereby reducing requirements for large disk storage. The only absolute requirement for disk space is that required to store an intermediate file containing the big G data (DEFAULT=BG.DAT), and potential scratch space for temporary 3D map data (if core memory is used up), and the space needed to store the final 3D map. In *THEORY*, EMICO3DR has no restrictions as to number of particles combined or the resolution limit. Of course, the memory capacity of the computer and disk

space available WILL pose ultimate limits if you attempt to step outside these bounds.

B. PROGRAM INPUT

1. MODE, IPRINT, BG_OUT, LG_OUT, FB_OUT (5I)
2. INPUT IMAGE FILENAME (A)
3. ICO_IDIM, BG_NANNULI, FFT_STEPSIZE, ICO_DIAM, ICO_SPREAD
(2I,3F)
(ICO_IDIM must be 128, 256, 512, or 1024)
4. THIS RECORD IGNORED
5. LG_NANNULI, RADIUS, STEP_SIZE (2I,F)
6. 3D MAP OUTPUT FILENAME (A; DEFAULT = ICOS2F.MAP)
7. HEADER FOR 3D MAP (18A4)
8. ID, THETA, PHI, OMEGA, FFT_ORIGX, FFT_ORIGY, SCALEF (I,6F)

1. MODE, IPRINT, BG_OUT, LG_OUT, FB_OUT (5I)

Input parameters control the operation of EMICO3DR as follows:

MODE = 0 ENTIRE PROGRAM IS RUN
 = 1 STOP AFTER CALCULATING BG'S (STORED IN BG.DAT)
 = 2 START WITH EXISTING BG.DAT DATA AND COMPUTE 3DR

IPRINT = 0 GIVES MINIMAL OUTPUT TO TERMINAL OR *.LOG FILE
 = 1 GIVES EXTRA OUTPUT
 = 2 GENERATES LOTS OF OUTPUT!

BG_OUT = 1 BG DATA SAVED IN BINARY FILE 'BG.DAT'

LG_OUT = 1 LG DATA SAVED IN BINARY FILE 'LG.DAT'

FB_OUT = 0 MEANS OUTPUT WHOLE 3D MAP IN STANDARD 2-FOLD VIEW
 = 5 MEANS OUTPUT TOP HALF IN 5-FOLD VIEW

When FB_OUT = 5, only the top(near) half of the 3D density MAP in the five-fold view is computed and stored in a disk file. Choosing this option may be very useful for producing preliminary maps where you just need a quick look at the MAP to see if it is correct. Invoking this option saves a tremendous amount of program I/O, since, for very large MAPs, much of the elapsed time is spent on I/O rather than CPU.

Future enhancements to EMICO3DR may include options to write out even smaller portions of the 3D map (e.g. just an octant) which will generate greater disk storage savings and perhaps enhancements in other routines.

2. INPUT IMAGE FILENAME (A)

Input to EMICOMATBG is in the form of IMAGE data stored as a group of individual images stored in a single file (PURDUE BYTE-PACKED FORMAT). Read BABE3:[TSB.DOC]EMPROGS.DOC for a description of the IMAGE file format.

NOTE: Information in this record is ignored if MODE.EQ.2.

3. ICO_IDIM,BG_NANNULI,FFT_STEPSIZE,ICO_DIAM,ICO_SPREAD (2I,3F)

ICO_IDIM defines the size of the FFT calculated for each image. ICO_IDIM must = 128, 256, 512, or 1024 and it MUST be greater than the largest image dimension (all particle images typically have identical NCOL,NROW dimensions).

BG_NANNULI, the number of annuli in the Fourier transform included in the calculations, defines the highest possible resolution of the final 3D MAP that can be computed. If BG_NANNULI is set too large, then errors are likely to be generated especially for data stes with very few particles. The upper limit to BG_NANNULI depends on the size of the FFT. A warning is given when an attempt is made to exceed this limit.

FFT_STEPSIZE determines how finely or coarsely the FFT is sampled. It is given by the equation:

$$\text{FFT_STEPSIZE} = \frac{(\text{ICO_IDIM} * \text{SCALEF})}{(\text{ICO_DIAM} * \text{ICO_NSAMP})} \quad (\text{transform X-steps})$$

ICO_NSAMP is a floating point number, specified by the user, that sets the sampling interval in Fourier space. It equals the number of annuli per interval of 1/ICO_DIAM, where ICO_DIAM is the diameter of the 'particle' (original image box dimension) in pixels. ICO_NSAMP = 1.0 for single or 2.0 for double sampling of the FFT data. In most instances, ICO_NSAMP is chosen to be = 2.0 for double sampling of the Fourier transform. Smaller values (coarser sampling) like 1.0 should normally only be used during preliminary processing as a means to speed up calculations (e.g. to get a quick and dirty look at a 3D map). Double sampling of the Fourier transforms (ICO_NSAMP=2.0) is the usual setting when the final 3D reconstruction calculations are performed (see pp.

321-323 of Crowther, DeRosier and Klug (1970) Proc. Roy. Soc. Lond. A. Vol. 317 for further discussion).

ICO_DIAM is the diameter (pixels) of the input particle images.

ICO_SPREAD sets the condition on the mean inverse eigenvalue. This is usually set equal to 1.0 (The DEFAULT). Set to a lower value for a less stringent test.

NOTE: Information in this record is ignored if MODE.EQ.2.

4. FILE CONTAINING BG DATA (A; DEFAULT = BG.DAT)

Enter the name of the permanent file in which the big G data are stored (MODE=2).

NOTE: Information in this record is ignored if MODE.LE.1.

5. LG_NANNULI, RADIUS, STEP_SIZE (2I,F)

LG_NANNULI is the cut-off radius (in annuli) in transform space. The maximum value, MAX_NANNULI (a value passed from the MAT/BG portion of the program or from the header record of the BG.DAT file) is the DEFAULT, but LG_NANNULI may be reduced to compute the 3D reconstruction at lower resolution.

RADIUS is the number of radial steps in real space, i.e. the outer radius of the reconstruction in pixels (no limit). A DEFAULT is calculated on the basis of the value of ICO_DIAM (input record 3) according to the relation: $[(ICO_DIAM - 1) / 2]$. ICO_DIAM is the original boxed particle diameter in densitometer pixels. For example, if ICO_DIAM = 99 pixels, then the DEFAULT value for RADIUS would be 49 pixels.

STEP_SIZE is the size of each radial step in pixels (DEFAULT = 1.0) Thus, it is the size of pixels in the reconstruction RELATIVE to the pixel size in the original scanned image.

EMICO3DR traps for improper values of LG_NANNULI, RADIUS, and STEP_SIZE. For testing or preliminary reconstructions, RADIUS should be reduced from the DEFAULT value and STEP_SIZE increased (so that the product of these two parameters stays constant) in order to speed up program execution. For computing a "final" reconstruction, the MAP has the best appearance when it is calculated with a fine step size, i.e. with RADIUS twice or more than the DEFAULT setting and STEP_SIZE half or less of the DEFAULT setting. Under these conditions however, the program takes considerably longer to finish and may require enormous amounts of

scratch and 'permanent' disk space. Typical, final reconstructions used to generate nice surface-shaded renderings are computed with RADIUS = ICO_DIAM and STEP_SIZE = 0.5.

NOTE 1: When RADIUS is set larger than the DEFAULT and STEP_SIZE is not reduced in proportion, meaningless density points outside the boxed boundary will be generated in the 3D MAP.

NOTE 2: Information in this record is ignored if MODE.EQ.1.

6. 3D MAP OUTPUT FILENAME (A; DEFAULT = ICOS2F.MAP)

Specify the name of the file to store the 3D MAP.

NOTE: Information in this record is ignored if MODE.EQ.1.

7. HEADER FOR 3D MAP (18A4)

The header is used to identify the 3D MAP data. The DEFAULT header text is "ICOS2F.MAP: 3D Reconstruction of icosahedral particle in standard 2-fold".

NOTE: Information in this record is ignored if MODE.EQ.1.

8. ID,THETA,PHI,OMEGA,FFT_ORIGX,FFT_ORIGY,SCALEF (I,6F)

ID is the image number--the storage position of the IMAGE data for byte-packed format data.

THETA,PHI,OMEGA are the current values of the three orientation angles (in degrees) for each particle image (obtained from previous runs of EMPFT or EMICOGRAD or EMICOFV).

FFT_ORIGX,FFT_ORIGY are the pixel coordinates of the particle center (the point 0.0,0.0 corresponding to the lower left corner of the boxed particle image). These values are estimated or refined by EMPFT (or with older programs like EMCORORG, EMICOORG, and EMICOORG2).

SCALEF is a radial scale factor (EMPFT or EMICOCMP can be used to determine this). With images of frozen-hydrated particles boxed from a single micrograph, SCALEF = 1.0 (DEFAULT) is normally assumed for all particles.

Examples of different acceptable ways to enter IMAGE data parameters are given in [TSB.DOC]EMICOGRAD.DOC.

NOTE: Information in this record is ignored if MODE.EQ.2.

C. PROGRAM EXECUTION

Normal operation of EMICO3DR leads to a 3D that MAP contains the entire icosahedral particle viewed down the 2-fold axis in the standard orientation: the three mutually perpendicular two-fold particle axes are aligned with a Cartesian (XYZ) MAP coordinate system (NCOL columns in the X direction; NROW rows in the Y direction; NSEC sections in the Z direction).

The maximum pixel resolution obtained in the three-dimensional reconstruction is estimated in following way:

Assume the particle images were originally boxed in 99 square pixel arrays, with each pixel corresponding to a 0.5 nm sampling interval (e.g. from a 50,000X micrograph digitized at 25 micron intervals). When 128 by 128 FFT's of such images are computed, and SCALEF = 1.0, and the transform is double-sampled (ICO_NSAMP = 2.0), then $FFT_STEPSIZE = (128 \cdot 1.0) / (99.0 \cdot 2.0) = 0.646$ TPU. The resolution obtained then depends on how large BG_NANNULI is set. The maximum radius, IRAD, at which data in the transform are used = $BG_NANNULI \cdot FFT_STEPSIZE$.

The following table illustrates the relationships between several of these parameters for the above example:

BG_NANNULI	IRAD	RESOLUTION (pixels)	RESOLUTION (nm)
16	11	11.64	5.82
32	21	6.10	3.05
48	32	4.00	2.00
64	42	3.05	1.52
80	52	2.46	1.23

WARNING: it is senseless to set BG_NANNULI large if the data is NOT reliable to the corresponding resolution. Since the typical rule of thumb is to digitize the data at a pixel resolution that is 3 or 4 times the expected resolution, then IRAD should NORMALLY not exceed $ICO_IDIM/3$ or $ICO_IDIM/4$. In the example above, if the IMAGE RESOLUTION is 2.0 nm and it is digitized at 0.5 nm intervals, IRAD should not exceed $128/4 = 32$.

NOTE: This program is generally run in BATCH mode. The following is an example batch job command file (BABE3:[TSB.FOR]EMICO3DR.BCH).

D. EXAMPLE EMICO3DR BATCH JOB

```
!           COMMAND FILE FOR RUNNING EMICO3DR FOR SEVERAL PARTICLES
!
!           THE INPUT ARE AS FOLLOWS:
!
!           1. MODE, IPRINT, BG_OUT, LG_OUT, FB_OUT (5I)
!              MODE      = 0 RUN ENTIRE PROGRAM
!                       = 1 STOP AFTER CALCULATING BG'S (STORED IN BG.DAT)
!                       = 2 START WITH EXISTING BG.DAT DATA AND COMPUTE 3DR
!              IPRINT    = 1 MEANS EXTRA OUTPUT TO TERMINAL OR *.LOG FILE
!                       = 2 MEANS EVEN MORE OUTPUT TO TERMINAL OR *.LOG
!                           FILE
!              BG_OUT    = 1 MEANS BG.DAT SAVED (OTHERWISE NOT SAVED)
!              LG_OUT    = 1 MEANS LG.DAT SAVED (OTHERWISE NOT SAVED)
!              FB_OUT    = 0 MEANS OUTPUT WHOLE 3D MAP IN STANDARD 2-FOLD
!                           VIEW
!                       = 5 MEANS OUTPUT TOP HALF IN 5-FOLD VIEW
!
!           IF MODE.EQ.0 THEN (RUNS ENTIRE PROGRAM)
!
!           2. INPUT IMAGE FILENAME
!           3. ICO_IDIM, NANNULI, FFT_STEPSIZE, ICO_DIAM, ICO_SPREAD
!              (2I,3F)
!              (ICO_IDIM must be 128, 256, 512, or 1024)
!           4. THIS RECORD IGNORED
!           5. NANNULI,RADIUS,STEP_SIZE (2I,F)
!           6. 3D MAP OUTPUT FILENAME (A; DEFAULT = ICOS2F.MAP)
!           7. HEADER FOR 3D MAP (18A4)
!           8. ID,THETA,PHI,OMEGA,FFT_ORIGX,FFT_ORIGY,SCALEF (I,6F)
!
!           IF MODE.EQ.1 THEN (STOPS AFTER CALCULATING BG DATA)
!
!           2. INPUT IMAGE FILENAME
!           3. ICO_IDIM, NANNULI, FFT_STEPSIZE, ICO_DIAM, ICO_SPREAD
!              (2I,3F)
!              (ICO_IDIM must be 128, 256, 512, or 1024)
!           4. THIS RECORD IGNORED
!           5. THIS RECORD IGNORED
!           6. THIS RECORD IGNORED
!           7. THIS RECORD IGNORED
!           8. ID,THETA,PHI,OMEGA,FFT_ORIGX,FFT_ORIGY,SCALEF (I,6F)
!
!           IF MODE.EQ.2 THEN (STARTS WITH EXISTING BG DATA FILE)
!
!           2. THIS RECORD IGNORED
```

```

!      3. THIS RECORD IGNORED
!      4. FILE CONTAINING BG DATA (DEFAULT = BG.DAT)
!      5. NANNULI,RADIUS,STEP_SIZE (2I,F)
!      6. 3D MAP OUTPUT FILENAME (A; DEFAULT = ICOS2F.MAP)
!      7. HEADER FOR 3D MAP (18A4)
!      8. THIS AND SUBSEQUENT RECORDS IGNORED
!
$ DEFINE SYS$EMSCRATCH "JUSTEM2:[SCRATCH.TSB]"
$ SET DEFAULT JUSTEM2:[SCRATCH.TSB]
$ RUN SYS$EXELIB:EMICO3DR.EXE
0, 0, 0, 0, 0
BABE3:[TSB.TEST]1019-box.PCK
256, 30, 0.757, 169.0, 1.0
BABE3:[TSB.TEST]BG.DAT
30, 84, 1.0
reo.map
15 particle reo map computed with EMICO3DR
10, 85.000, 20.076, 169.805, 85.658, 84.264, 1.002
13, 79.000, 9.168, 53.789, 90.083, 80.719, 1.002
15, 88.000, -28.017, 330.117, 88.976, 83.627, 1.002
20, 80.000, 9.139, 108.984, 87.322, 81.299, 1.001
35, 87.000, -15.021, 87.188, 83.139, 79.838, 1.002
38, 82.000, -16.157, 216.914, 86.053, 81.466, 1.000
45, 88.000, 25.015, 211.641, 88.203, 79.416, 1.002
47, 89.000, -28.004, 169.805, 86.051, 81.753, 0.998
58, 87.000, -25.034, 346.992, 85.577, 82.448, 0.998
62, 88.000, 26.016, 203.555, 87.421, 81.606, 0.998
64, 81.000, 12.150, 107.227, 86.847, 79.774, 0.998
65, 84.000, -19.105, 133.594, 83.676, 78.427, 0.998
66, 75.000, 3.106, 201.094, 86.338, 79.218, 1.000
67, 81.000, 14.175, 18.281, 85.537, 76.735, 0.998
68, 79.000, -11.206, 20.391, 85.976, 81.223, 1.000
$ WAIT
$ EXIT

```

E. PROGRAM NOTES

Several parameters are listed at the terminal (or in the *.LOG file):

LG_NANNULI = Cut-off radius (in ANNULI) in transform space.
 Usually equals MAX_NANNULI (set in MAT/BG part of program), but may be made smaller to reduce resolution.

FFT_STEPSIZE = width of each transform annulus, in transform pixel units (TPU), given by:

$$\frac{(ICO_IDIM * RSCALE)}{-----} TPU$$

$$(ICO_DIAM * ICO_NSAMP)$$

- MAX_BESSEL_ORDER = The highest Bessel order used in the computation of the layerplane data.
- STEP_SIZE = Size of each radial step (i.e size of each pixel in the reconstruction relative to the original pixel size in the digitized particle image).
- ICO_DIAM = Original particle diameter (in densitometer pixels). This also equals ICO_NSAMP times the reciprocal of annuli spacing in EMICOMATBG (ICO_NSAMP/FFT_STEPSIZE).
- RADIUS = Number of radial steps (pixels) in real-space for the reconstruction. The particle diameter = (2*RADIUS + 1).
- RESLUTION = Reconstruction resolution (in units of pixels in the original digitized image).
- NCOL = Dimension of 3D MAP (# of pixels/row). Also equal to NROW (# of pixels/column) and NSEC (# sections in the MAP).

EMICO3DR first computes one half of the icosahedral particle, viewed in a five-fold orientation (5-fold axis coincident with the Z axis and one, perpendicular 2-fold axis coincident with the Y direction), from the little g's. The little g's are only computed for Bessel orders that are multiples of 5. The density map is calculated a section at a time in unique $2\pi/5$ (= 72 degree) sectors. Each sector is subdivided in cylindrical coordinates (RHO(r,phi)) since the little g's are computed in polar form. The polar section is interpolated to produce a Cartesian section (MAP(x,y)).

After all sections are computed (the number of them = RADIUS+1), the half-particle 3D MAP is rotated so that the old Y direction (2-fold) becomes the new Z direction and a 3D MAP of the entire particle is temporarily stored (in core memory if available or on scratch disk). The MAP in this orientation has two other mutually perpendicular 2-fold axes in the XY plane of the central section, but not coincident with the X and Y directions. A final transformation rotates the particle into the "standard" orientation so the three mutually perpendicular 2-fold axes

coincide with the X, Y, and Z directions (ROWS,COLUMNS, SECTIONS) of the 3D MAP.

The program produces a 3D MAP with strict 522 symmetry. Symmetry about the 3-fold axes is not enforced in the current version of the program, but may be enforced by use of EMICOSYM (to give 532 symmetry).

F. PROGRAM NOTES FOR SYSTEM MANAGER

The program performs many calculations in core memory if it is available, thus cutting down some of the I/O overhead and increasing execution speed. If memory is unavailable, computations are performed with the use of scratch space. In order to run this program efficiently, it may be necessary to ask your friendly VAX/VMS system manager to set up your account with the following job parameter limits:

Working Set (pagelets) /Limit=2000 /Quota=64000 /Extent=300000
Adjustment enabled Authorized Quota=64000 Authorized Extent=300000

Working Set (8Kb pages) /Limit=125 /Quota=4000 /Extent=18750
Authorized Quota=4000 Authorized Extent=18750

CPU limit:	Infinite	Direct I/O limit:	150
Buffered I/O byte count quota:	99616	Buffered I/O limit:	150
Timer queue entry quota:	10	Open file quota:	511
Paging file quota:	397088	Subprocess quota:	10
Default page fault cluster:	64	AST quota:	498
Enqueue quota:	2000	Shared file limit:	0
Max detached processes:	0	Max active jobs:	0

These settings may not be ideal, but they do seem to work fine for running the routine on the Purdue ALPHAVAX 4000-300 with 512 Mbytes core memory.

G. ADDITIONAL NOTES

1. Use EMICOROT to obtain 3D MAPS of the icosahedral particle in any equatorial setting (THETA=90.0 degrees).
2. Use option "X" of EMMAP to project the 3D MAP in any view orientation or EMICOPRJ to project the 3D MAP in any equatorial view (THETA=90.0 degrees).

H. FLOW CHART FOR EMICO3DR PROGRAM

```

*****
*      MAIN      *
* (EMICO3DR.FOR) *
*****
*
* - PIRADDEG*
*          | - STRING_UPPER* | - STRING_UPPER*
* - ICO_3DR_INFO - | - IMG_OPEN ----- | -
IMG_PACK_BIMGCOM_CLEAR*
*          | - FILE_CHECK*
* - ICO_3DR_GETMEM1 - LIB$GET_VM*
*
* - ICO_MATBG_GETPARAMS*
*
*          | - IMG_PACK_FIND - IMG_PACK_NRECS*
*          | - IMG_FILL - IMG$FILL(IMG$PACK_FILL)
* - ICO_3DR_FFTS - | - IMG_FFT - [see below]
*          | - FFT_TO_ATBT - PIRADDEG*
*          | - ICO_ATBT_STORE*
*
* - ICO_3DR_GETMEM2 - LIB$GET_VM*
*
*          | - ICO_EM4IMR*
*          | - ICO_EMMEET*
*          | - ICO_EM4RMI*
* - ICO_3DR_MATBG - | - ICO_ZEROZ*
ICO_LOCATE*
*          | - ICO_INTRP*
ICO_TRIDI*
*          | - ICO_NMAT* | - ICO_HOW ----- | -
ICO_EIGVAL*
*          | - ICO_BGSOLVE - | - ICO_BGPROD* | -
ICO_EIGVEC*

```

```

*                               | - ICO_BGOUT*                               | -
ICO_RELOCT*
*
*- ICO_3DR_FREEMEM1 - LIB$FREE_VM*
*
*- ICO_3DR_RECS*
*                               I--BSL0-*
*- ICO_3DR_NEWBES -----I--BSL1-*
*
*- ICO_3DR_LG-*
*
*- ICO_3DR_MRC5F - LIB$MOVC3-*
*
*- ICO_3DR_MRC2F - LIB$MOVC3-*
*
*- ICO_3DR_XYZ2F - LIB$MOVC3-*
*
*- TSBEND

*
*- IMG__FFT - | - IMG_FFT_FILL - FFT_CLEAR*
*           | - FFT_2D - FOURT - L6TOL9*

```