

EMMAPZOOM.DOC

(last update by TSB, November 3, 1998)

CONTENTS

A. INTRODUCTION

B. PROGRAM INPUT

C. FLOW CHART FOR EMMAPZOOM PROGRAM

A. INTRODUCTION

EMMAPZOOM is used to magnify or demagnify a 3D MAP or a portion of a 3D MAP. This program is a highly condensed version of two programs, EMMAP3DT and EMSF3DBT, which compute forward and reverse Fourier transformations, respectively. That is, the EMMAP3DT portion of EMMAPZOOM computes structure factors from a 3D density MAP (Fourier analysis) and the EMSF3DBT portion of EMMAPZOOM computes a 3D density MAP from the structure factors (Fourier synthesis). The obvious advantage of EMMAPZOOM is that it accomplishes the job of two programs and does so without a need to generate an intermediate set of structure factor data. Also, EMMAPZOOM is designed to be much easier to run as there are many fewer input parameters compared to those required by EMMAP3DT and EMSF3DBT. The 'cost' if any is that EMMAPZOOM is less flexible. However, EMMAP3DT or EMSF3DBT are still available.

Generally EMMAPZOOM will be useful for magnifying small regions of a 3D MAP (*i.e.* create data with very high pixel resolution for producing nice shaded-surface renderings in RobEM). The features of EMMAPZOOM will be incorporated into RobEM before too long, which will give users the opportunity to compute and visualize high pixel resolution MAPs at interactive or near interactive speeds. EMMAPZOOM will remain available for those that don't have access to RobEM or interactive graphic device.

EMMAPZOOM accepts as input a PIF format 3D density MAP and outputs a PIF format 3D density MAP, generally of different dimensions.

EMMAPZOOM is fairly straightforward and can probably be run interactively with most size maps (probably 201^3 or smaller). The majority of time is spent on I/O (reading in and writing out 3D MAP data). The time spent running EMMAPZOOM should certainly be shorter in general than running the two programs, EMMAP3DT and EMSF3DBT.

B. PROGRAM INPUT

1. 3D MAP input filename (A)
2. 3D MAP output filename (A)
3. MAP output file header (80A1)
4. PIXSIZ_OLD, PIXSIZ_NEW, RES_HI (3F)
5. XMIN, XMAX, YMIN, YMAX, ZMIN, ZMAX (6F)

Detailed descriptions of program input:

1. 3D MAP input filename (A FORMAT)

Enter the name for the file that contains the input 3D MAP. The program expects as input a map with Z-sections, in which X varies most rapidly (rows) and Y most slowly (columns).

2. 3D MAP output filename (A FORMAT)

Enter the name for the output file that will contains the new 3D MAP. The program will output a map with Z-sections, in which X varies most rapidly (rows) and Y most slowly (columns).

3. MAP output file header (80A1 FORMAT)

Enter any header information (up to 80 characters) you like here. This information is stored in the header record of the PIF MAP data file.

4. PIXSIZ_OLD, PIXSIZ_NEW, RES_HI (3F)

DEFAULTS: 1.0, 1.0, 2.00001

PIXSIZ_OLD specifies the dimension of each pixel in the input 3D MAP and PIXSIZ_NEW specifies the dimension of each pixel in the output 3D MAP. PIXSIZ_OLD and PIXSIZ_NEW may be defined in any units you choose (Å, nanometers, pixels, yards, etc.) but you **MUST BE CONSISTENT** and use the **same** units to define PIXSIZ_OLD, PIXSIZ_NEW and RES_HI. RES_HI defines the upper resolution limits of the 3D MAP data. RES_HI must be **larger** than $2.0 \times \text{PIXSIZ_OLD}$ or the Nyquist limit (two-pixel resolution) would be violated (a definite no no). The DEFAULTS for PIXSIZ_OLD, PIXSIZ_NEW, and RES_HI are 1.0, 1.0, and $2.0001 \times \text{PIXSIZ_OLD}$. This value for RES_HI should be treated as an **absolute lower limit**, which, if used, is likely to be an unrealistic value for 3D MAP data which normally have pixels that are at least two or three times smaller in dimension than the resolution limit of the data.

C. FLOW CHART FOR EMMAPZOOM PROGRAM

```

*
*          | - PIRADDEG !
*          | - PIF_OPEN !
*- MAP3DT_INFO - | - PIF_READ_GH - differentEndian !
*          | - PIF_READ_DH - | - differentEndian !
*          |                   | - convertBackFloat !
*          | - CHK_PRIME !
*          | - PIF_INIT_HEAD !
*          | - REAL_TO_REC !
*
*- MAP3DT_GETMEM1 - MALLOC !
*
*- PIF_READ_TMAPI2 - | - PIF_READ_MAP_SHORT_IMAGE - | - differentEndian !
*
*          |                   | - SRFP !           | - R2CFTK !
*          |                   | - MDFTKD ----- | - R3CFTK !
*- MAP3DT_PASS1 - | - REALFT - CMLPFT - | - DIPRP !       | - R4CFTK !
*          |                   |                   | - R5CFTK !
*          | - CMLPFT - (see above)           | - R8CFTK !
*          |                   | - RPCFTK !
*
*- FREE !
*
*- PIF_CLOSE !
*
*- MAP3DT_GETMEM2 - MALLOC !
*
*- MAP3DT_STORE !
*
*          |                   | - R2CFTK !
*          |                   | - R3CFTK !
*- MAP3DT_PASS2 - CMLPFT - | - MDFTKD ----- | - R4CFTK !
*          |                   | - DIPRP !       | - R5CFTK !
*          |                   |                   | - R8CFTK !
*          |                   | - RPCFTK !
*
*- MAP3DT_SFOUT - | - PIRADDEG !
*          | - PIF_OPEN !
*          | - PIF_WRITE_GH - differentEndian !
*          | - PIF_WRITE_DH - | - differentEndian !
*          |                   | - convertBackFloat !
*          | - PIF_WRITE_SF - | - differentEndian !
*          |                   | - convertBackFloat !
*          | - PIF_CLOSE !
*          | - FREE !
*
*- MAP3DT_LOADDATA !
*

```