CCP4 NEWSLETTER ON PROTEIN CRYSTALLOGRAPHY

An informal Newsletter associated with the BBSRC Collaborative Computational ProjectNo. 4 on Protein Crystallography.

Number 47 **Winter 2007-2008**

Contents

CCP4 Activities and news

Peter Briggs, Martyn Winn, Francois Remacle

CCP4, CSE Department, STFC Daresbury Laboratory, Warrington WA4 4AD

ACA 2007, Salt Lake City Utah

The 2007 Annual Meeting of the American Crystallographic Association took place in Salt Lake City, Utah, from the 21st to 26th July, and once again CCP4 was present with a stand in the commercial exhibition - this year manned by Charles Ballard and Peter Briggs, with some assistance from Paul Emsley (developer of Coot).

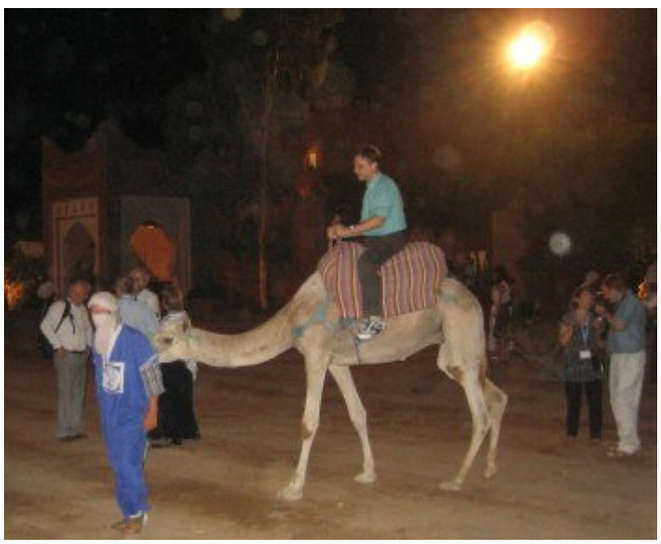
Any last requests ...? From left to right: Paul Emsley, Peter Briggs and Charles Ballard

As on previous occasions, we met a mixture of new and familiar faces whom we were happy to help with various questions about the CCP4 software. The two CCP4-related graphics programs CCP4mg and Coot continued to be of interest to visitors, as well as the automated molecular replacement system "MrBUMP".

We'd like to thank all the people who came along and talked to us during the conference and provided us with feedback on the suite - we hope that it was equally interesting and useful for you - and the ACA organisers for their help.

ECM 2007, Marrakech

This year's European Crystallographic Meeting was held in the exotic location of Marrakech. CCP4/Daresbury was represented by Martyn Winn and Norman Stein. Martyn presented aspects of CCP4 at the pre-conference Advanced Training Workshop, and gave a talk on the molecular replacement pipeline MrBUMP. That talk was in a session on "Advances in crystallographic phasing and refinement" which also included Kevin Cowtan on the latest developments in Buccaneer for automated model building at lower resolutions, and Raj Pannu on the latest in the Leiden suite of programs, in particular Afro for FA calculation. Norman presented a poster on the next generation Truncate program which is scheduled for 6.1. Martyn also chaired a lively session on "Structure validation and quality control" which covered X-ray data quality and complementary experimental techniques as well as more traditional areas of validation.



Martyn, somehow riding his "coach" to dinner

The conference was rounded off by a spectacular Congress

Gala Dinner under canvas. As the picture shows, the coach to the dinner was somewhat unusual.

BSR 2007, Manchester

The 9th International Conference on Biology and Synchrotron Radiation (BSR) was held from 13th to 17th August in Manchester, close to Daresbury, and was attended by a number of CCP4 staff and collaborators who presented various posters, including:

- "MrBUMP - An automated framework for doing Molecular Replacement in Macromolecular structure solution" - Ronan Keegan, Martyn Winn, Wendy Yang, & Peter Briggs
- "CCP4 Diffraction Image library" - Francois Remacle & Graeme Winter
- "dbCCP4i: Tracking data in PX Structure Determination Software Pipelines" - Peter Briggs, Wanjuan Yang & Ronan Keegan)

Ronan Keegan and Peter Briggs in the poster session at BSR 2007

Staff Changes: Comings and Goings

Wanjuan Yang

At the end of August the Daresbury CCP4 group said farewell to **Wanjuan Yang** (known to friends and colleagues as Wendy). Wendy had been part of the Daresbury group since December 2004, and had been working on CCP4's contribution to the BIOXHIT project, specifically the development of the "dbCCP4i" database backend and related tools for CCP4i (see the articles in newsletter 45 and newsletter 46).

After over two and a half years with us, she has now moved on to take up a new post working with genomics databases at the Sanger Centre in Hinxton, Cambridge. Wendy has made a significant contribution to the project during her time with CCP4, and we shall certainly miss working with her in future. We wish her every success in her new job and in her post-CCP4 life.

Maeri Howard

On 14th September Maeri Howard (the CCP4 administrator) gave birth to a baby boy. Maeri is now on maternity leave until May 2008, and in the meantime her various roles - including handling of commercial licensing and organisation of the CCP4 Study Weekend - are being filled by very capable colleagues within the CSE Department at Daresbury Laboratory.

We wish both parents and baby well, and look forward to seeing Maeri back again next May - when she will return to the lab, no doubt for a break from motherly duties!

2008 CCP4 Activities

- **2008 CCP4 Study Weekend** : *Low Resolution Structure Determination and Validation*
  This took place in leeds from 3rd January to 5th January with 440 people attending.
- **CCP4 Workshop in India**:
  This is currently (18th -> 22nd February) taking place in Bangalore.
- **CCP4 school:** *From data processing to structure refinement and beyond*
  This will take place in the APS, Argonne Laboratory in Chicago, Illinois, from 23rd May to 28th May.
  http://www.ccp4.ac.uk/courses/APS2008/
- **ACA 2008** *(Knoxville, Tennessee)*
  This will take place from 31st May to 5th June. CCP4 staff will be present with their stand.
- **IUCr 2008** *(Osaka, Japan)*
  This will take place from 23rd to 31st August. CCP4 staff will be present with their stand as well.
- **CCP4 Workshop in Japan**:
  This will take place in Tokyo the week after the IUcr.

4th Conference on Structural Analysis of Supramolecular Assemblies by Hybrid Methods

On behalf of the program committee, we are excited to announce that the 4th International Conference on Structural Analysis of Supramolecular Assemblies by Hybrid Methods will be held from March 12-16, 2008 in Lake Tahoe, California at Granlibakken Conference Center. Following the success of the three prior meetings, this conference will be the fourth in a series that brings together specialists from diverse branches of structural biology, while drawing primarily from the fields of X-ray crystallography, electron cryomicroscopy and computational biology. The agenda will also extend into biophysical methods, proteomics and cell biology.

The meeting will be divided into seven sessions and the topics include:

- Hybrid Approaches to Macromolecular Filaments
- Hybrid Approaches to Membrane Complexes
- Hybrid Approaches to Dynamic Assemblies
- Computational Approaches to Hybrid Analysis
- Hybrid Approaches to Macromolecular Machines
- Hybrid Approaches to Cellular Organization
- Cellular Proteomics

The submission of abstracts is encouraged: **deadline December 14, 2007.** In addition to a poster session, additional platform presentations will be selected from the submissions. About a third of the talks in the final program will have been selected in this way. For additional information and instructions, please feel free to view our website at: http://www.hybridmethods2008.com/ or if you have any questions, you can contact us directly at: hybridmethods2008@burnham.org

We look forward to seeing you in March at this exciting event.

People behind the conference:

Chair: Dorit Hanein

Co-Chair: Wes Sundquist

Organizing committee: Alasdair C. Steven, Benjamin Geiger, Clare M. Waterman-Storer, Robert M. Stroud, and Wolfgang Baumeister

Contact:
Dawn Fischkelta
Hybrid Methods 2008 Conference Secretariat
10901 North Torrey Pines Road
La Jolla, California 92037

News of CCP4 Release 6.1

Peter Briggs, Martyn Winn, Charles Ballard, Francois Remacle, Norman Stein and Ronan Keegan

CCP4, CSE Department, STFC Daresbury Laboratory, Warrington WA4 4AD
Email: ccp4@dl.ac.uk

---

1 Introduction

The current release of the CCP4 software suite is version 6.0.2, which was made available in December 2006 and which has previously been described in some detail in an article in newsletter 44. This article outlines the new and updated programs and features that are currently scheduled for inclusion in the next major release of CCP4, version 6.1, due out later in 2008.

2 Major Changes in CCP4 6.1

2.1 New programs

A number of significant new programs will be included in CCP4 6.1:

- **AFRO**: a program for calculating E_A values (the normalised heavy atom contribution) from SAD, MAD and SIRAS data.
- **BUCCANEER**: statistical model building program that can be used to trace protein structures in electron density maps by identifying connected alpha-carbon positions using a likelihood-based density target.
- **CTRUNCATE**: a program for converting intensities to structure factors and checking data quality. CTRUNCATE is intended to supersede the existing TRUNCATE program.
- **CRUNCH2**: program for determining the substructure of the anomalous scatterers or heavy atoms.
- **MrBUMP**: an automated pipeline for molecular replacement developed by Ronan Keegan and Martyn Winn, which includes search model retrieval and search model preparation.
- **PISA**: a standalone version of Eugene Krissinel's PISA (Protein Interfaces, Surfaces and Assemblies) program that is a useful for examining various characteristics of protein packing and other interactions. Previously the PISA functionality was only available as a web-based service from the EBI.
- **POINTLESS**: program for Laue and Patterson group determination from unmerged reflection data, as well as a number of subsidary functions including: reflection format conversion, checking and reindexing of reflection data against a reference set, and apply a pre-selected reindexing matrix.
- **RAPPER**: program for generating protein conformers by discrete sampling of likely conformers within a given set of restraints. It can be applied to a number of problems, for example: *ab initio* loop building, comparative modelling and C• -trace modelling. It can also be used to build and refine conformers using X-ray crystallographic data.

The RAMPAGE module of RAPPER also provides an analysis tool that generates Ramchrandran plots which are more consistent with current models than those in PROCHECK.

- **SEQUINS**: SEQUence INSertion detection program that performs sequence validation by comparing model side chains against electron density. It can be used after molecular replacement or to validate structures in the PDB.
- **XIA2**: an expert system for performing automated reduction and analysis of X-ray diffraction image data with minimal user input. It is aimed at two principal sets of users: novice users with little knowledge or experience of data processing, and expert users who wish to process higher volumes of data.

There are also new utility programs scheduled for inclusion:

- **IDIFFDISP**: a standalone viewer for raw diffraction images which is intended as a replacement for the old IPDISP program.
- **MTZ2CIF**: a utility that generates mmCIF reflection files suitable for deposition, and is intended to replace OUTPUT CIF option of MTZ2VARIOUS.
- **R500**: utility for correcting REMARK 500 lines in PDB files before submission to a deposition site.
- **SEQWT**: a program that estimates protein molecular weight from the sequence
- **SYMCONV**: a utility for interrogating the CCP4 symmetry libraries in order to look up information about spacegroups and symmetry operations in various formats.
- **BAUBLES**: a utility for re-rendering CCP4 log files into HTML markup. Baubles will be integrated into CCP4i.

It should be noted that many of these programs have been available for some time already, either via the CCP4 Prerelease Pages at http://www.ccp4.ac.uk/prerelease/ or via their own project-specific pages. However, their inclusion in CCP4 6.1 means that users will no longer have to download and install them separately.

2.2 Updated Programs

In addition to the new programs outlined in the previous section there are various updates to many of the existing programs in the suite. These include:

- **PHASER** 2.1.2 (this version covers MR, SAD and combined MR and SAD)
- **REFMAC** 5.4.0067
- **SCALA** 3.3.1
- **SFCHECK** 7.02.6
- **MOLREP** 10.1.7
- **MOSFLM** 7.0.2 (along with the new iMOSFLM interface)
- **OASIS** updated to OASIS-2006
- **PDB_EXTRACT** 3.0
- **CRANK** 1.20

Please note that these version numbers are correct at the time of writing but may be superseded by newer versions before CCP4 6.1 is finally released. The latest versions of COOT and CCP4MG will also be made available for download with CCP4 6.1.

2.3 Changes to Graphical User Interface CCP4i

The improvements and changes to CCP4i are described extensively elsewhere in this newsletter. They include new task interfaces for the new programs plus many new core features and enhancements.

2.4 New Libraries

The release of CCP4 6.1 will include the new **DiffractionImage** library, which provides a set of C++ functions (plus wrappers in different languages including Tcl and Python) for handling diffraction image data from a variety of sources.

DiffractionImage also comes with a set of utility programs:

- **diffdump**: displays all the "standard" information from a specific diffraction image file
- **printpeaks**: prints a list of peaks found on an image
- **automask**: automatically generate a backstop mask from a an image
- **diff2jpeg**: generate a JPEG from an image file

2.5 Deprecated and Withdrawn Programs

As part of the CCP4 6.1 release a number of programs have been designated as "deprecated", in anticipation of removing them completely from future releases. These programs have either been superseded by superior programs, or have fallen out of mainstream use. They are:

- BEAST (replaced by Phaser)
- ROTAPREP (replaced by Combat)
- ARP_WATERS (the latest version of ARP/wARP should be used instead)
- XLOGGRAPH (replaced by loggraph)
- IPDISP (replaced by idiffdisp)
- BPLOT, POLYPOSE, RSEARCH, RESTRAIN, XDLMAPMAN and XDLDATAMAN

The source code of deprecated programs will still be included for download, however they will not be built as part of the standard installation process and will not be included in the binary distributions that CCP4 provides.

2.6 Other changes and new features

There are various miscellaneous changes and features scheduled for CCP4 6.1:

- **Dependency on Fortran 90**
  The latest versions of REFMAC (5.3+) and MOLREP (10+) now require a Fortran 90 compiler in order to build. This should only affect users who build the suite from source code.
  The CCP4 configure will check for Fortran 90 support and will automatically disable the REFMAC and MOLREP builds if it is not available.

- **Update Alert Mechanism**
  It is intended to implement an "update alert mechanism" as part of CCP4 6.1, which will automatically notify users when bug fixes or updated program versions become

available on the CCP4 server. Although the details are still being worked out, the intention is to help users keep their CCP4 distributions more up-to-date in between major releases of the suite.

- **InstallShield installation wizard for Linux systems**
  For CCP4 6.1 an InstallShield-based installer (similar to that already used for Microsoft Windows platforms) will be available for installing pre-compiled executables onto Linux systems.

3 Current Status and Availability

At the time of writing (January 2008) there are currently still a few components and features which have not yet been fully integrated into the suite, however initial test versions have been made and are undergoing testing with a selected group of developers and users.

It is planned that a small number of additional test releases will be made over the next couple of months ahead of the full public release. If you are interested in trying out a trial version of 6.1 then please contact ccp4@dl.ac.uk.

CCP4 6.1 is scheduled for a full public release sometime in spring 2008. Watch for announcements on the CCP4bb mailing list and other crystallographic lists, or check the CCP4 home page for news.

Finally, this article reflects our current state of knowledge about the expected content of CCP4 6.1, however it is possible that there may be some changes before the public release. It is recommended therefore that you check the list of significant changes that is distributed with the suite and linked from the main documentation index.

4 Acknowledgements

The CCP4 project is a collaborative effort and continues to thrive through generous contributions of time, energy and software from members of the UK and international PX communities. Unfortunately time and space do not permit the acknowledgement here of all these valuable contributions, however a list of acknowledgements is included in the current release, and acknowledgements for the specific developments described in this article are given below:

- AFRO is developed by Navraj Pannu; CRUNCH2 is developed by Jan Pieter Abrahams (Leiden University)
- BUCCANEER and SEQUINS are developed by Kevin Cowtan (University of York)
- CTRUNCATE is developed by Norman Stein (CCP4, Daresbury Laboratory)
- MrBump is developed by Ronan Keegan and Martyn Winn (CCP4, Daresbury Laboratory)
- PISA is developed by Eugene Krissinel (MSD-EBI)
- SCALA and POINTLESS are developed by Phil Evans (MRC-LMB Cambridge).
- RAPPER is developed by Nicholas Furnham, Paul de Bakker, Mark DePristo, Reshma Shetty, Swanand Gore and Tom Blundell (University of Cambridge)
- XIA2 is developed by Graeme Winter (CSE, Daresbury Laboratory)

- IDIFFDISP is developed by Francois Remacle (CCP4, Daresbury Laboratory). The DiffractionImage library is developed by Francois Remacle in collaboration with Graeme Winter.
- MTZ2CIF and SYMCONV are developed by Peter Briggs. BAUBLES has been developed by Peter Briggs in collaboration with Kevin Cowtan and Phil Evans.
- R500 is developed by Kim Henrick (MSD-EBI)
- SEQWT is developed by Eleanor Dodson (University of York)
- PHASER is developed by Randy Read's group (Wellcome Trust, Cambridge).
- REFMAC5 is developed by Garib Murshudov's group (University of York)
- SFCHECK and MOLREP are developed by Alexei Vagin (University of York)
- OASIS is developed by Tao Zhang *et al* (Beijing National Laboratory for Condensed Matter Physics) and Quan Hao (Cornell University)
- PDB_EXTRACT is developed by Huanwang Yang (RCSB PDB)
- CRANK is developed by Navraj Pannu (Leiden University)
- CCP4mg is developed by Liz Potterton and Stuart McNicholas (York University)
- COOT is developed by Paul Emsley (Oxford University)
- MOSFLM is developed by Harry Powell and Andrew Leslie; iMOSFLM is currently developed by Luke Kontogiannis (MRC-LMB Cambridge)
- CCP4i is maintained and developed by the CCP4 group at Daresbury

The CCP4 suite is maintained, developed and released by the CCP4 group in the Computational Science and Engineering Department (CSED) at STFC Daresbury Laboratory, and comprises Charles Ballard, Peter Briggs, Maeri Howard, Ronan Keegan, Francois Remacle, Norman Stein and Martyn Winn.

The CCP4 project is supported by the BBSRC, by income from commercial distribution of the software, and by STFC Daresbury Laboratory. CCP4 would also like to thank the many people past and present who support the project, both with their time and with their contributions to the software suite itself - without which the project would not be able to exist.

New Developments in CCP4i: December 2007

Peter Briggs

CCP4, CSE Department, STFC Daresbury Laboratory, Warrington WA4 4AD
Email: p.j.briggs@dl.ac.uk

1 Introduction

CCP4i [Potterton03] is the CCP4 graphical user interface. This article is intended to give an overview of some of the developments in CCP4i which will be available in the next release of the CCP4 software suite (version 6.1). The article is divided into sections outlining the new task interfaces, new utilities and the other major and minor new features scheduled for the next version of CCP4i.

Details of features in the current version of CCP4i can be found in a previous article in newsletter 42.

2 New Task Interfaces

There are a number of new task interfaces scheduled for the next release of CCP4i. Many of these correspond to the introduction of major new packages in the main suite, while others provide access to programs in the suite that have previously not had a graphical interface. A number of these interfaces are shown in figure 1.

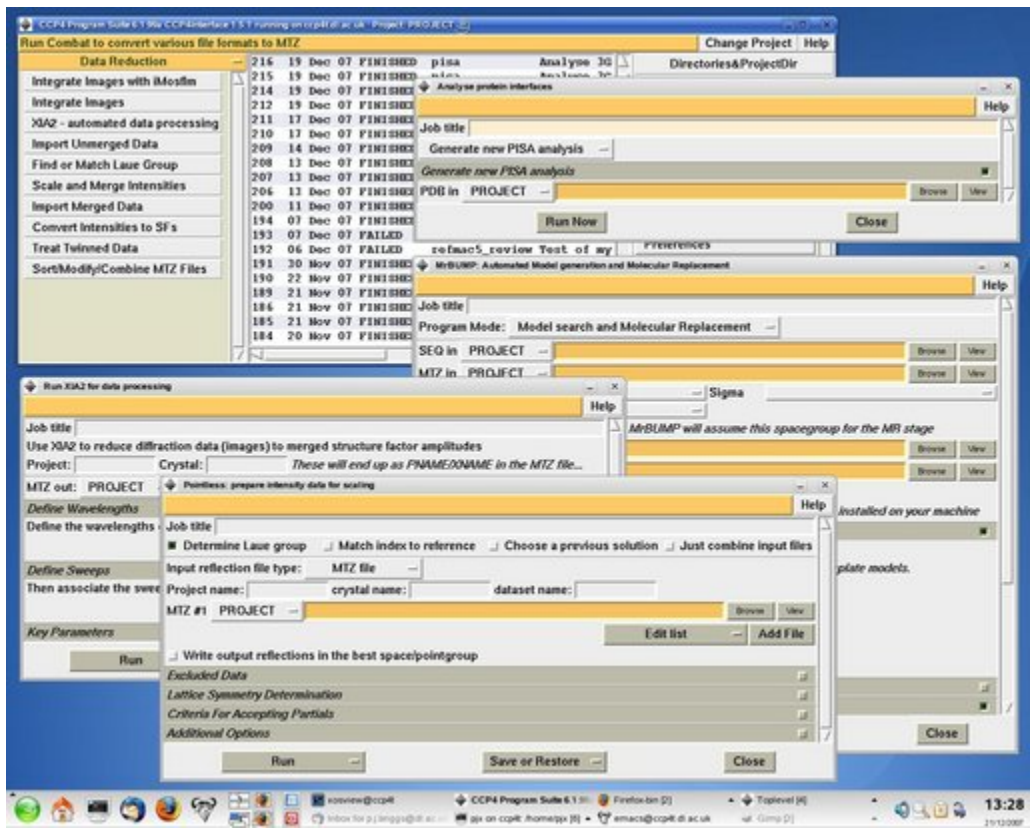


Figure 1: *CCP4i surrounded by some of the new task interfaces (XIA2, Pointless, MrBUMP and PISA)*

- **MrBUMP** [Keegan07] is a package for automated molecular replacement. It has been available as a separate download for some time, however with the next release be incorporated officially into the CCP4 suite and its interface will automatically be available from CCP4i without any additional installation steps.

  MrBUMP was also the first program to officially make use of the updated database system and viewer, which will also be incorporated into this release CCP4i (see sections 3.2 and 4.4).

  More information about MrBUMP can be found at http://www.ccp4.ac.uk/MrBUMP

- **BUCCANEER** [Cowtan07] is a new model building program developed by Kevin Cowtan. It can be used to trace protein structures in electron density maps by identifying connected alpha-carbon positions using a likelihood-based density target. Two interfaces are provided to the program: the "autobuild/refine" procedure cycles BUCCANEER with REFMAC5, while "fast build only" procedure runs BUCCANEER alone. Both interfaces are provided in the "Model Building" module.

  Additionally Kevin has also provided the **SEQUINS** sequence validation tool, which also has an interface in the "Model Building" module.

- **RAPPER** [Furnham07] is a program for generating protein conformers by discrete sampling of likely conformers within a given set of restraints. It can be applied to a number of problems, for example: *ab initio* loop building, comparative modelling and C**.**-trace modelling. It can also be used to build and refine conformers using X-ray crystallographic data.

  As part of the incorporation of RAPPER into CCP4, the RAPPER developers have provided an interface to the program which will be accessible from the "Model Building" module of CCP4i. In addition, the the analysis tool **RAMPAGE** (a "module" of RAPPER that generates Ramchrandran plots using more recent data than those in PROCHECK) will also be accessible from the "Validate model and/or data" task.

- The **PISA** program (Protein Interfaces Surfaces and Assemblies) [Krissinel07] developed by Eugene Krissinel has been available for some time as a webservice from the MSD-EBI (see http://www.ebi.ac.uk/msd-srv/prot_int/pistart.html). It is a useful tool for examining various characteristics of protein packing and other interactions.

  CCP4 6.0.2 provided a simple "launcher" task in CCP4i that gave quick access to the PISA website but otherwise provided no additional functionality. However CCP4 6.1 will include a command line version of PISA that can be run on a local machine, and CCP4i will also offer an interface to this local version.

- **Pointless** [Evans06] has been available in a pre-release form for some time along with a set of three CCP4i task interfaces to the program that were developed to allow access to particular functions of the program - specif ically Laue group determination, checking the indexing of reflection data, and checking the centre of symmetry.

  Following developments with Pointless to extend its functionality (for example, the ability to accept multiple input files in different formats) Phil Evans has developed an entirely new interface to Pointless which replaces the original trio, and this is the version of the interface that will be offered in the next release of CCP4i.

- **XIA2** [Winter08] is an expert system developed by Graeme Winter for performing automated reduction and analysis of X-ray diffraction image data with minimal user input. It is aimed at two principle sets of users: novice users with little knowledge or experience of data processing, and expert users who wish to process higher volumes of data.

  XIA2 will be included in the CCP4 6.1 release and will be accessible from a simple CCP4i interface that should facilitate running the program and provide a complement to the the interactive data processing afforded through iMosflm.

  More information about XIA2 can be found at http://www.ccp4.ac.uk/xia

Other new tasks for existing programs

A number of new tasks have been added to provide access to existing programs. These include interfaces for the CHOOCH and DYNDOM programs (which previously haven't had interfaces at all) and for the new experimental phasing functionality in Phaser 2.1.1.

3 New Utilities

Two new utilities are planned for inclusion in the next version of CCP4i.

3.1 idiffdisp: diffraction image viewer

idiffdisp is a standalone viewer for raw diffraction images (figure 2). It has been developed by Francois Remacle and which is intended as a replacement for the old IPDISP program. idiffdisp is more fully-featured than IPDISP (for example it includes options to "play" a set of images like a movie, which useful for rapid visual inspection to find anomalous images) and also runs on Windows as well as Linux and UNIX systems.

idiffdisp has been described in a previous newsletter article:
http://www.ccp4.ac.uk/newsletters/newsletter46/articles/DiffractionImage.html.

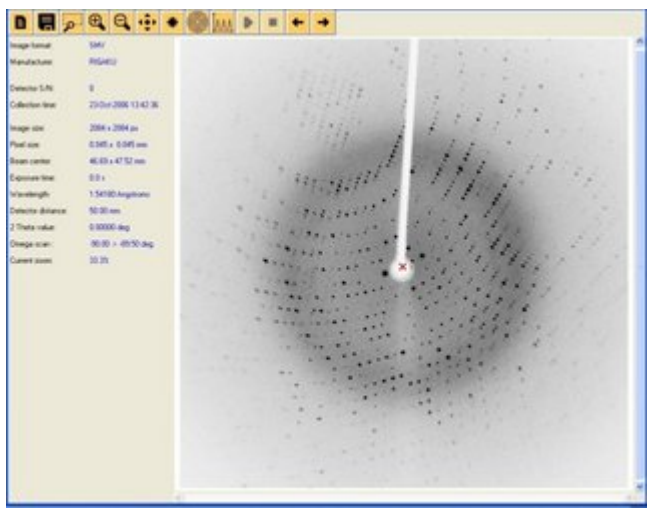


Figure 2: *Screenshot showing the idiffdisp viewer displaying an X-ray diffraction image*

3.2 dbviewer: a visualiser for CCP4i project history

dbviewer provides graphical views of jobs within a CCP4i project, giving a perspective on the project history and the flow of data (see figure 3). It is a component of the database handler system (see section 4.2 below) and may already be familiar to people who have used more recent versions of MrBUMP. Within the next version of CCP4i the viewer will provide views of any of the user's projects as a complement to the existing joblist view in the main CCP4i window.

The viewer has previously been described in a recent newsletter article: http://www.ccp4.ac.uk/newsletters/newsletter46/articles/project-tracking.html.

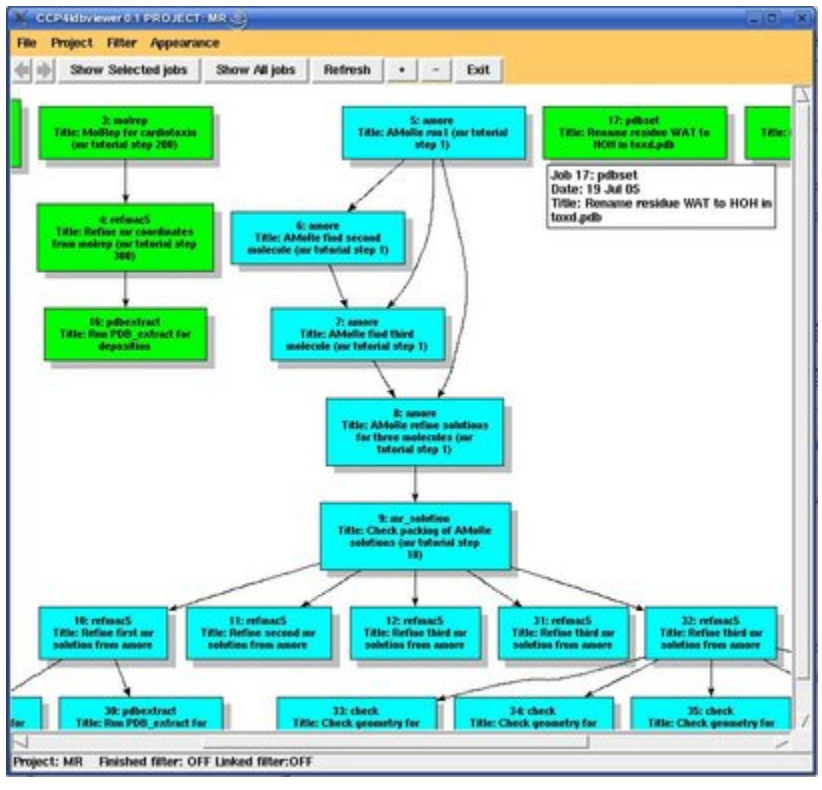


Figure 3: *The dbviewer showing a view of the jobs in a CCP4i project. Each box represents a job, with the arrows linking jobs where the output of one is used as the input for another.*

4 Major New Features

A number of new features have been implemented in the next version of CCP4i.

4.1 Better integration with Coot and CCP4mg ("plugins")

Although increasingly powerful tools have become available over the last few years, users of CCP4i have increasingly suffered from the lack of integration between CCP4i and many of these tools - most significantly, when wishing to access model building tools via Coot [Emsley04] and visualisation facilities via CCP4mg [Potterton02].

To begin to address this, in the next release of CCP4i the "View Files from Job" menu will include new options from certain tasks (most notably Refmac5 [Murshudov96]) to view the results in Coot or CCP4mg (provided that the programs themselves are installed and available). An example of how these shortcuts appear in the menu is shown in figure 4.

Figure 4: *"View Files from Job" menu offering options to view the results of a Refmac5 run in Coot and CCP4mg directly.*

Selecting these options will launch the chosen application with the files preloaded - for example, from Refmac5 Coot or CCP4mg will be started with both the appropriate MTZ and coordinate files already selected.

Although still fairly limited, the aim of this integration is to make it much easier to move from CCP4i into the appropriate tool. Feedback or suggestions on other tools or tasks where this mechanism could be added would be gratefully received.

4.2 Integration with iMOSFLM

iMOSFLM is the new interface for the MOSFLM data processing and integration program [Leslie92], which improves vastly over the old X-windows based interface. The CCP4 6.1 will include iMOSFLM and CCP4i will provide a button to launch it.

The development of iMOSFLM will ultimately make the current MOSFLM-in-batch CCP4i task interface obsolete. However the older interface will be retained for now and should still be available in the next CCP4i alongside the iMOSFLM option.

More information and downloads of iMosflm can be found at http://www.mrc-lmb.cam.ac.uk/harry/imosflm/. The program is also described in an earlier newsletter article.

4.3 Improved presentation of tasklists and modules

In previous (and current) versions of CCP4i, the only forms of organisation for the tasklists (the sets of tasks available in the menus on the left hand side of the main CCP4i window) have been a division into separate modules, and the ordering of tasks within those modules.

Traditionally, tasks appropriate for a particular part of the struture determination are grouped together in a module, for example "data reduction" or "molecular replacement". But an increasingly frequent criticism of this presentation is that it can difficult for novice (and sometimes for experts) to see which tasks are related, what order they could be used in, and which tasks are alternatives to each other. In an attempt to mitigate this criticism, the next version of CCP4i will see the incorporation of an extra level of organisation within modules, which allows tasks to also be grouped into "subfolders".

Some examples of the new arrangements are shown in figure 5. Although the changes to allow subfolders have been made to the CCP4i code, some work still remains to be done on the actual reorganisation itself and feedback from users would be very valuable.

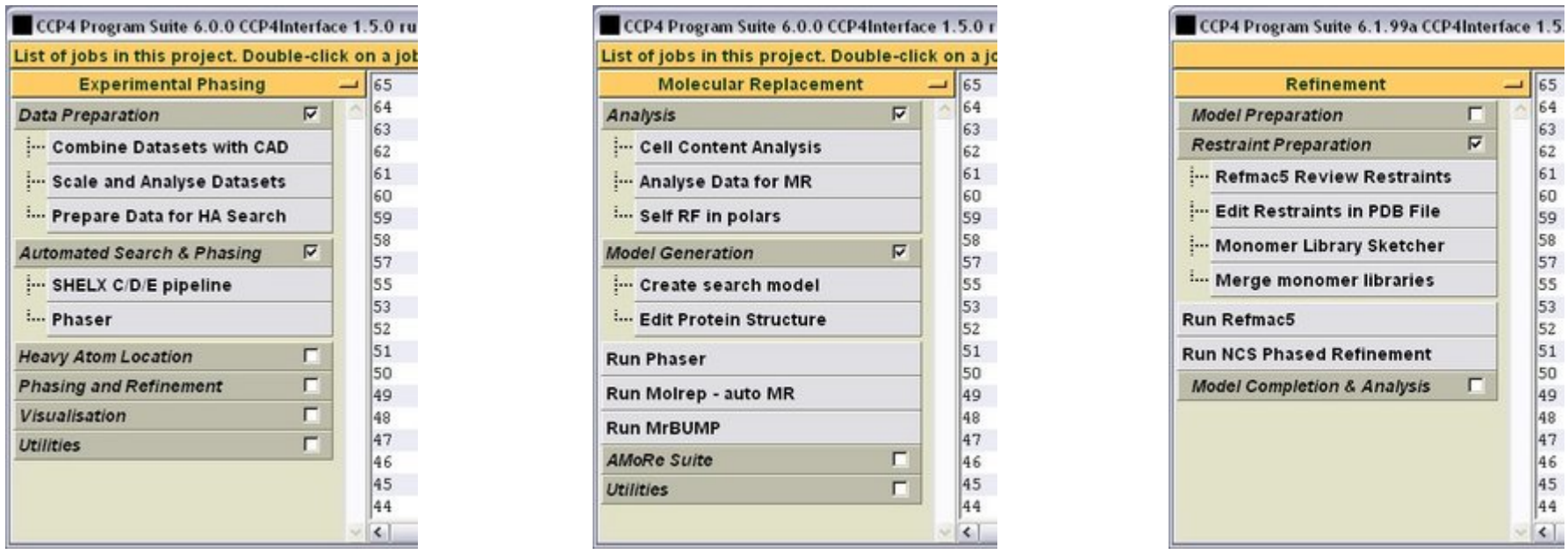


Figure 5: *examples of the suggested reorganised tasklists for (from left to right respectively) "Experimental Phasing", "Molecular Replacement" and "Refinement" modules, with open and closed subfolders.*

4.4 Integration with database handler system "dbccp4i"

Since 2004 there has been an ongoing project to develop the job database in CCP4i under the auspices of the European Union BIOXHIT project. The project includes a "database handler" program called **dbccp4i**, that manages the job database and provides access to the jobs by other programs. The dbviewer visualisation tool (discussed earlier in section 3.2) is also part of this system.

While an earlier version of dbccp4i has successfully been incorporated into the more recent versions of MrBUMP, as of the next release of the CCP4 suite, the database handler will be incorporated directly into CCP4i. This work is currently on-going.

Most of the changes should happen "under the hood" and initially there should be few perceptible differences in usage. However in the longer term the incorporation of dbccp4i should allow other programs such as Coot, CCP4mg and iMOSFLM to communicate directly with the CCP4i project database and help provide a more integrated CCP4 environment.

5 Other new features

In addition to the major new features outlined in the previous section there are a number of minor new features that are intended to improve the general ease of use of CCP4i.

5.1 Task importer

The task importer is a function within CCP4i that allows the reinstallation of "3rd party" interfaces from an old installation of CCP4 into a newer installation. For example: if you have installed the ARP/wARP CCP4i interface into your CCP4 6.0.2 installation, then this function makes it possible to automatically copy that interface to a new CCP4 6.1 installation.

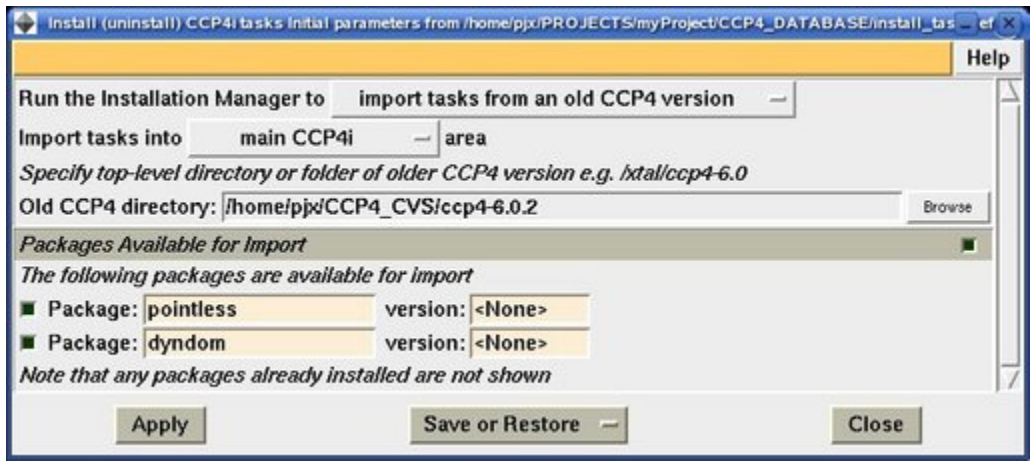


Figure 6: *example of the task importer interface showing the packages already installed in a different CCP4 installation that can be copied to the current installation.*

An example of the interface window is shown in figure 6. The function will be accessed from the **"System Administration->Install/uninstall tasks"** menu option. It can also be activated from the command line using:

```
ccp4i -import_packages /home/pjx/CCP4_CVS/ccp4-6.0.2
```

Note that only the task interfaces will be imported using this function - any associated programs will (if necessary) need to be moved manually to your new installation.

5.2 Improvements to interactions with the Job Database List

In an attempt to improve usability of the job database list (in the centre of CCP4i's main window), a number of improvements have been implemented.

- A right mouse click on the job list brings up a "context menu" with appropriate options depending on the selection of jobs in the window - examples are shown in figures 7 and 8, for cases where there is one or several jobs selected in the list. The options that are presented in these menus are essentially the same as are visible on the right-hand side of the main window. However some users may find this a more convenient shortcut to those tasks.
- A left mouse double-click on a job in the job list brings up the logfile for that job while a double-click while holding down the shift key is equivalent to selecting "rerun" for the job.

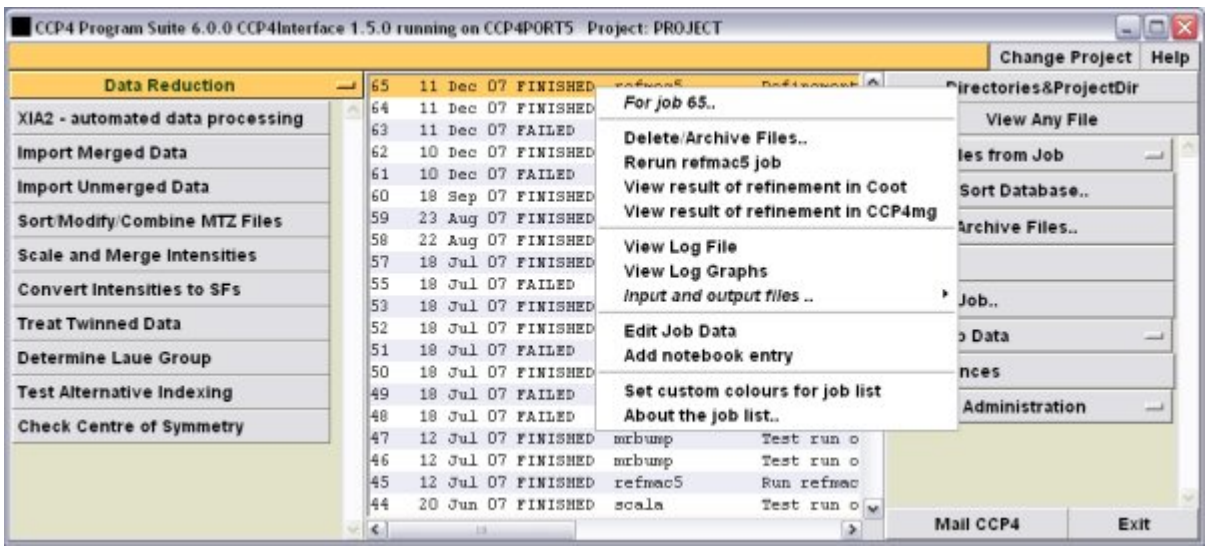


Figure 7: *job list menu for a single selected job, offering various options including viewing files, launching plugins and rerunning the job.*

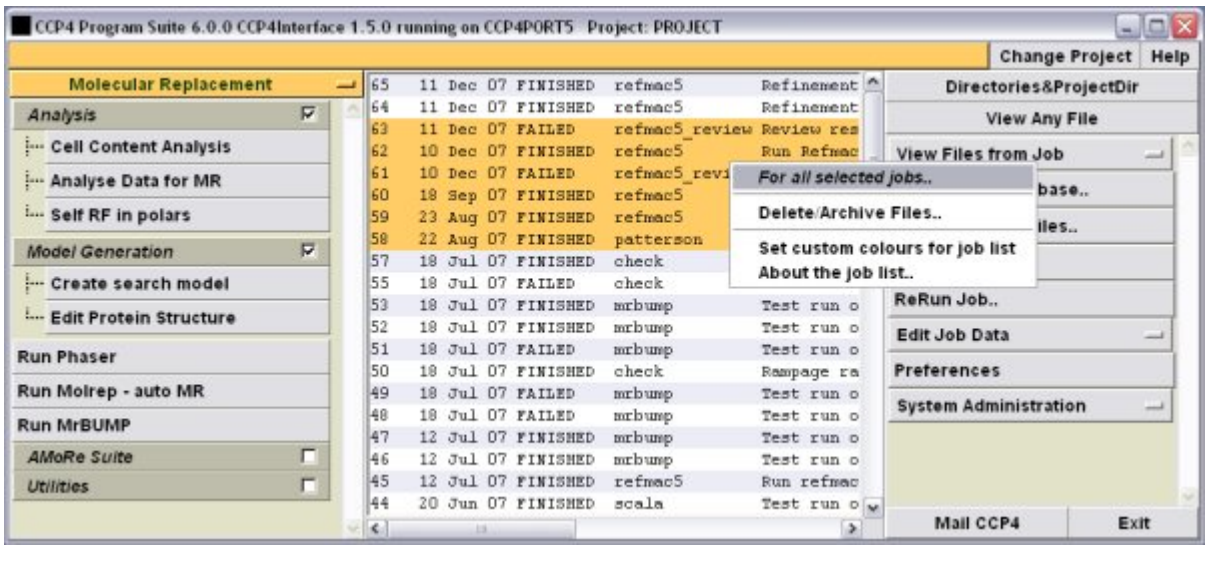


Figure 8: *job list menu for a several selected jobs.*

5.3 Loggraph support for Scala xmgr-formatted ROGUES, CORRELPLOT and NORMPLOT files

The Scala program (accessed in CCP4i through the "Scale and Merge" task in the "Data Reduction" module) generates a number of different output files which contain useful analyses of the data. Examples of such files include the NORMPLOT (normal probability distribution from the merge stage), CORRELPLOT (a scatter plot of pairs of anomalous differences from random half-datasets) and ROGUES (outliers on the detector).

Inspection of these plots is useful as a way of assessing the quality of the scaling and merging steps. However as the files are formatted for display using the XMGR or XMGRace programs, this presents a problem for users on systems where these programs are not available (particularly Microsoft Windows). However modifications to Loggraph mean that it can now also be used to display these files as a last resort if XMGR is not found on the system. Some examples of Loggraph's display can be seen in figure 9.

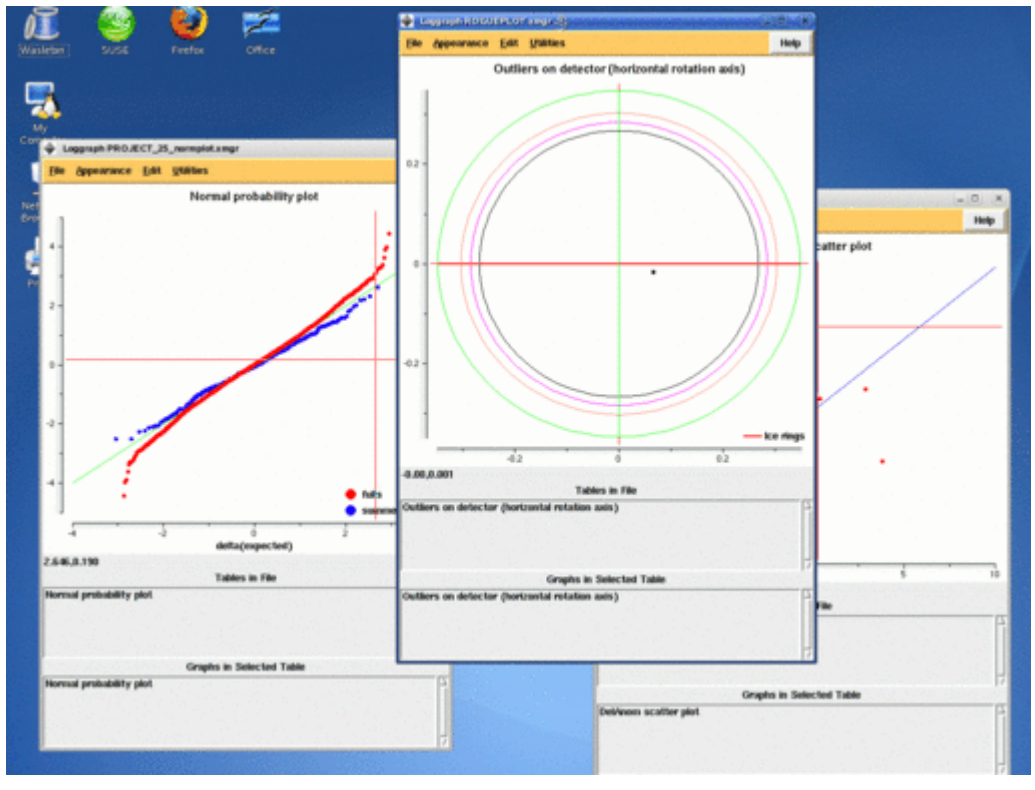


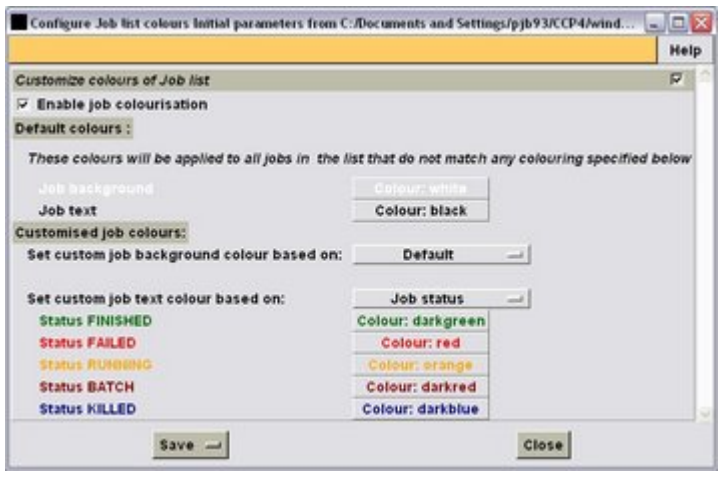
Figure 9: *examples of loggraph displaying the XMGR-formatted NORMPLOT, CORRELPLOT and ROGUES output from Scala*

5.4 Colourisation of the job list

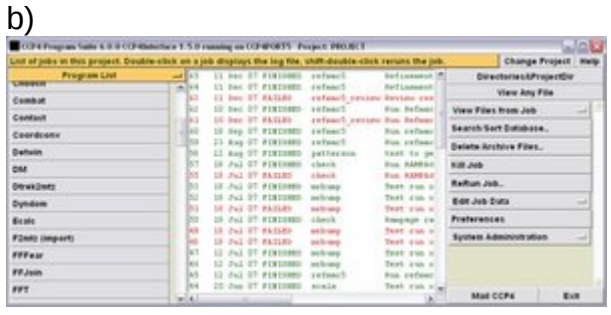
This is actually an old feature that has been available since CCP4 6.0, which provides the ability to customise the colours of jobs displayed in the main window according to user-defined criteria (for example, based on the job status). Some examples are shown in figure 10 b) and c). The purpose of providing the colourisation is to allow users to define their own colour scheme which helps with comprehending the status of the jobs in their projects.

In the current version of CCP4i this functionality is accessed via the "Configure Interface" window under the "System Administration" menu, so it's not very obvious. In the next version of CCP4i however the colourisation functionality will be more easily accessible via direct links to a dedicated interface from the "System Administration" menu and from a right-mouse click on the job database.

a)



b)



c)

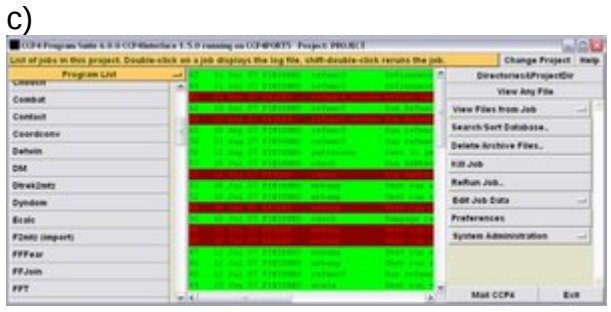


Figure 10: *The interface for applying custom colours to the job database list displayed in the main CCP4i window (figure a) and two examples of custom colour schemes applied to the job list (b and c).*

5.5 Incorporation of "baubles"

The `baubles` program is a reformatter for logfiles from CCP4i and CCP4 programs, which generates an "annotated" HTML version of standard logfiles that includes "inline" graphs using the JLogGraph applet (see figure 11). It will be incorporated into CCP4i to give the option of viewing the logfile in this format.

See the article elsewhere in this newsletter for more information about baubles and integration with CCP4i.

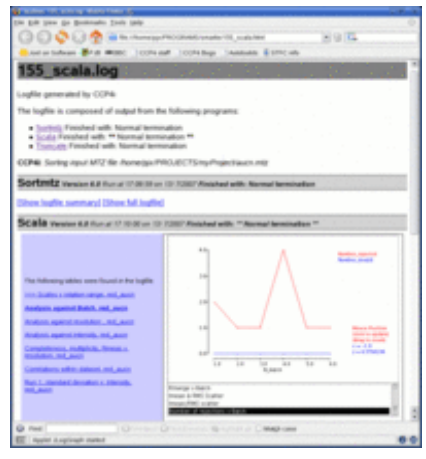


Figure 11: *Annotated logfile produced by* `baubles` *with inline loggraphs and viewed in a web browser.*

6 Current Status and Availability

Although the majority of the changes and updates described in this article are already implemented, they will unfortunately not be generally available until the release of CCP4 6.1. In the meantime if you are particularly interested in trying out a developmental version of CCP4i then please contact me for information on how to get hold of the updated code.

7 Acknowledgements

MrBUMP and its interface has been developed by Ronan Keegan and Martyn Winn. The interface to XIA2 was developed by Graeme Winter and Peter Briggs. Martyn Winn developed the interface to the local version of the PISA program. The original pre-1.2.11 interfaces to Pointless were developed by Peter Briggs; the latest version has been developed by Phil Evans with input from Peter Briggs. The interfaces to CHOOCH and DYNDOM have been developed by Francois Remacle, and the Phaser interfaces are developed and maintained by the Phaser developers Randy Read and Airlie McCoy. The RAPPER interface has been developed by Nick Furnham. The BUCCANEER and SEQUINS interfaces have been developed by Kevin Cowtan.

idiffdisp has been developed by Francois Remacle. Francois also integrated iMOSFLM into CCP4i and implemented the joblist-colourisation functionality. iMosflm has been developed by Geoff Battye, Harry Powell and Luke Kontogiannis.

The dbviewer and dbccp4i programs were developed by Wanjuan Yang in collaboration with Peter Briggs as part of CCP4's contribution to the BIOXHIT project, which is funded by the European Commission within its FP6 Programme under the thematic area "Life sciences, genomics and biotechnology for health", contract number LHSG-CT-2003-503420. Additional funds are provided from CCLRC Daresbury Laboratory via the CCP4 project.

Various other modifications and changes were made by Peter Briggs with useful input from CCP4 staff and collaborators. The remodelled modules and tasklists were developed from original suggestions by Charlie Bond.

CCP4i was originally developed by Liz Potterton, and is now maintained and developed by the Daresbury CCP4 staff (Peter Briggs, Martyn Winn, Charles Ballard, Francois Remacle, Norman Stein and Ronan Keegan) who contributed other fixes and developments. Please send questions, requests and bug reports to us at ccp4@dl.ac.uk.

The figures in this article were prepared using the Gimp and ImageMagick programs.

8 References

- [Cowtan06] K. Cowtan, CCP4 Newsletter 44 (Summer 2006), "The 'Buccaneer' protein model building software" http://www.ccp4.ac.uk/newsletters/newsletter44/articles/buccaneer.html
- [Emsley04] P. Emsley and K. Cowtan (2004) *Acta Cryst.* **D60**, 2126-2132 "Coot: Model-Building Tools for Molecular Graphics"
- [Evans06] P. Evans (2006) *Acta Cryst.* **D62**, 72-82 "Scaling and assessment of data quality"
- [Furnham07] Nicholas Furnham and Tom L. Blundell, CCP4 Newsletter 45 (Winter 2006/7), "RAPPER: Real Space Automated Conformer Generation" http://www.ccp4.ac.uk/newsletters/newsletter45/articles/CCP4_Newsletter_Jan_2007_FINAL.htm
- [Krissinel07] E. Krissinel and K. Henrick (2007). "Inference of macromolecular assemblies from crystalline state". *J. Mol. Biol.* **372**, 774-797.
- [Keegan07] R.M.Keegan and M.D.Winn, *Acta Cryst.* **D63**, 447-457 (2007) "Automated search-model discovery and preparation for structure solution by molecular replacement"
- [Leslie92] A.G.W. Leslie, (1992), Joint CCP4 + ESF-EAMCB Newsletter on Protein Crystallography, No. 26.
- [Murshudov96] G. Murshudov, A.Vagin and E.Dodson, (1996) in the *Refinement of Protein structures Proceedings of Daresbury Study Weekend.* "Application of Maximum Likelihood Refinement"
- [Potterton02] E. Potterton, S. McNicholas, E. Krissinel, K. Cowtan and M. Noble *Acta Cryst.* (2002). **D58**, 1955-1957 "The CCP4 molecular-graphics project"
- [Potterton03] E. Potterton, P. Briggs, M. Turkenburg and E. Dodson, *Acta Cryst.* **D59** (2003) 1131-1137 "A graphical user interface to the CCP4 program suite"
- [Winter08] Winter et al (2008) in preparation

CCP4mg:A New GUI

Liz Potterton and Stuart McNicholas

YSBL, University of York, York YO10 5YW

Email: ccp4mg@ysbl.york.ac.uk

Introduction

Following the release of CCP4mg version 1.1 we will be working on converting the CCP4mg GUI from using the TCL/Tk toolkit to the QT toolkit. This move will not only give a more modern look-and-feel to the program but will also give significant improvements to program performance.

QT provides a native-like interface on all platforms and is used many by major software developments such as KDE and GoogleEarth. It is developed commercially by a Norwegian company, Trolltech, but is also available with a GPL license. The QT toolkit has many off-the-shelf widgets such as file browser, colour browser and text/html display which will enable us to update CCP4mg and implement new functionality rapidly.

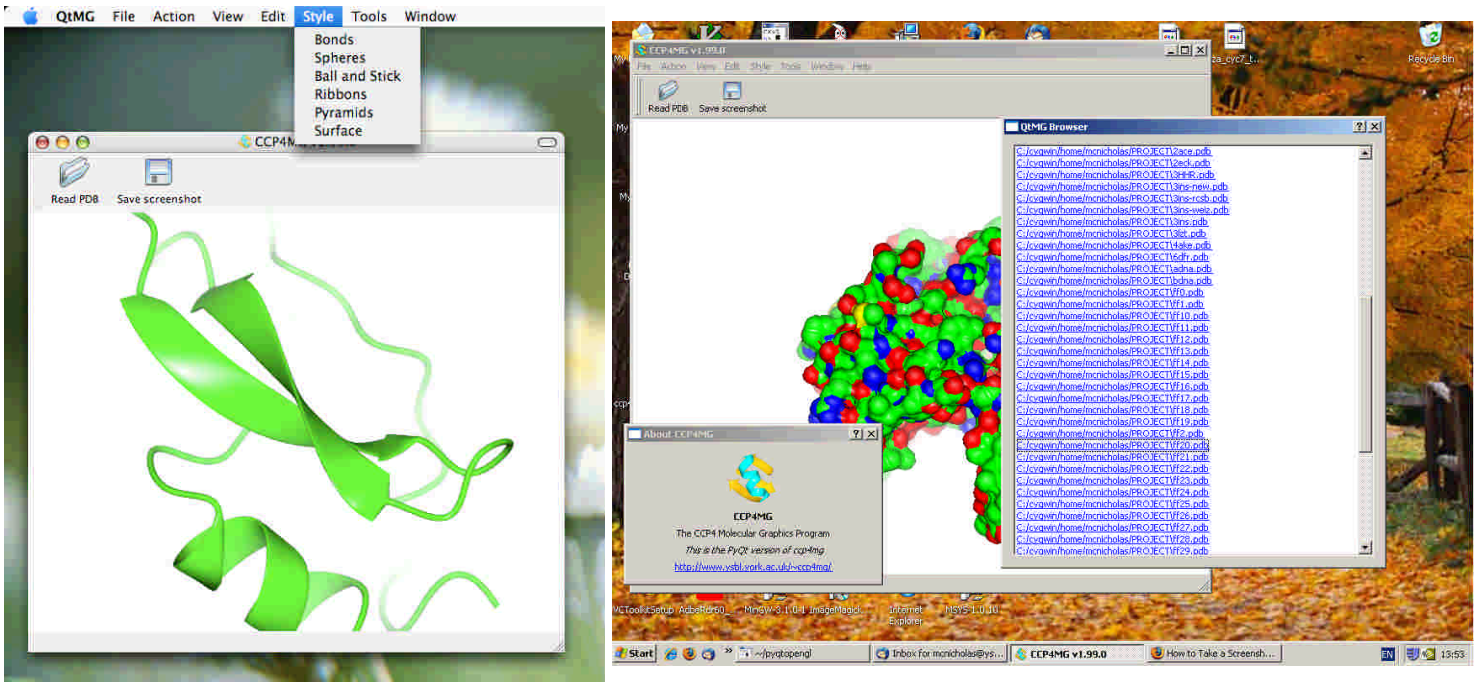


Figure 1. A prototype CCP4mg running on MAC OSX and Windows operating systems.

Improving Performance

A vital feature of QT is an openGL widget which will support the main graphics window of CCP4mg. In our initial prototype porting the existing molecular graphics to run in the QT openGL widget proved to be straightforward and resulted in a massive improvement in graphics performance over the existing program. This is due to a simplification in the program architecture, reducing the number of threads in the program, and also eliminating the need for the GLUT package which handled input to the openGL window. We expect this improvement to be retained as we extend the functionality of the prototype.

Moving to QT

The original GUI was written in TCL/Tk, the same as CCP4i since at the start of the CCP4mg project there was no clear alternative which was supported on all platforms and was without licensing issues. But it was clear that TCL/Tk was being overtaken by more modern toolkits and that a change of toolkit would be desirable in the future. Therefore the GUI was implemented via an *abstract GUI definition* layer which separates the GUI toolkit from the rest of program and which should enable a relatively rapid toolkit conversion. We intend to keep a similar program layout so moving to the new GUI should not disorient users. Some aspects of the GUI will be updated though: the file browser and colour selection will certainly be revised.

There is now a CCP4mg bulletin board hosted at JISCMail where you can discuss these proposals and anything else CCP4mg related. Please sign up and give us your feedback.

Liz Potterton & Stuart McNicholas, November 15th 2007

Further improvements to AREAIMOL code

Ian J. Tickle, Astex Therapeutics Ltd.

The CCP4 program AREAIMOL, originally written by Peter Brick (Imperial), for computing the solvent-accessible surface area from co-ordinates in PDB format (see Briggs, 2000 for more information), is frequently used to perform buried surface area calculations when a complex is formed. This requires taking the difference between two large numbers (*i.e.* the sum of the solvent-accessible surface areas of the separate molecules minus the surface area of the complex). The buried surface area may be only 5-10% of the accessible area so a small relative error in the accessible area can be multiplied 10-20 times. Hence it's important to make the area calculation as accurate as possible. AREAIMOL uses a surface point counting algorithm but it is critical that the distribution of points is as uniform as possible, *i.e.* the local surface point density is the same everywhere (note that this is not the same problem as randomly distributing points with a uniform probability distribution over the surface of a sphere, since then the local surface point density will have a statistical variation). Except for special values of the number of points on the surface of a sphere (corresponding to the numbers of vertices for the Platonic solids: 4, 6, 8, 12 & 20) an exact solution is not possible, and anyway for sufficient accuracy it's normally necessary to have 1500-2000 points per atom, so an approximate solution is needed. It was noticed that the algorithm used didn't perform very well in the polar regions of the sphere, particularly at low point densities, thus contributing to inaccuracies in the area calculation.

I performed some tests with two C atoms separated along the **z** axis by the sum of their VDW radii (3.6 Å), varying the surface point density input to AREAIMOL and got the solvent-accessible surface area results with the distributed version shown in the Figure (labelled OLD). As can be seen the computed area varies significantly; an exact calculation using geometry, labelled CALC, gives 201.06 $Å^2$ (= $4\pi(1.8+1.4)(1.8+1.4+3.6/2)$ where the VDW radius is 1.8 and the probe radius is 1.4). The distributed version of AREAIMOL uses a default surface point density of 1 $Å^{-2}$ which is clearly much too low for accurate calculations.

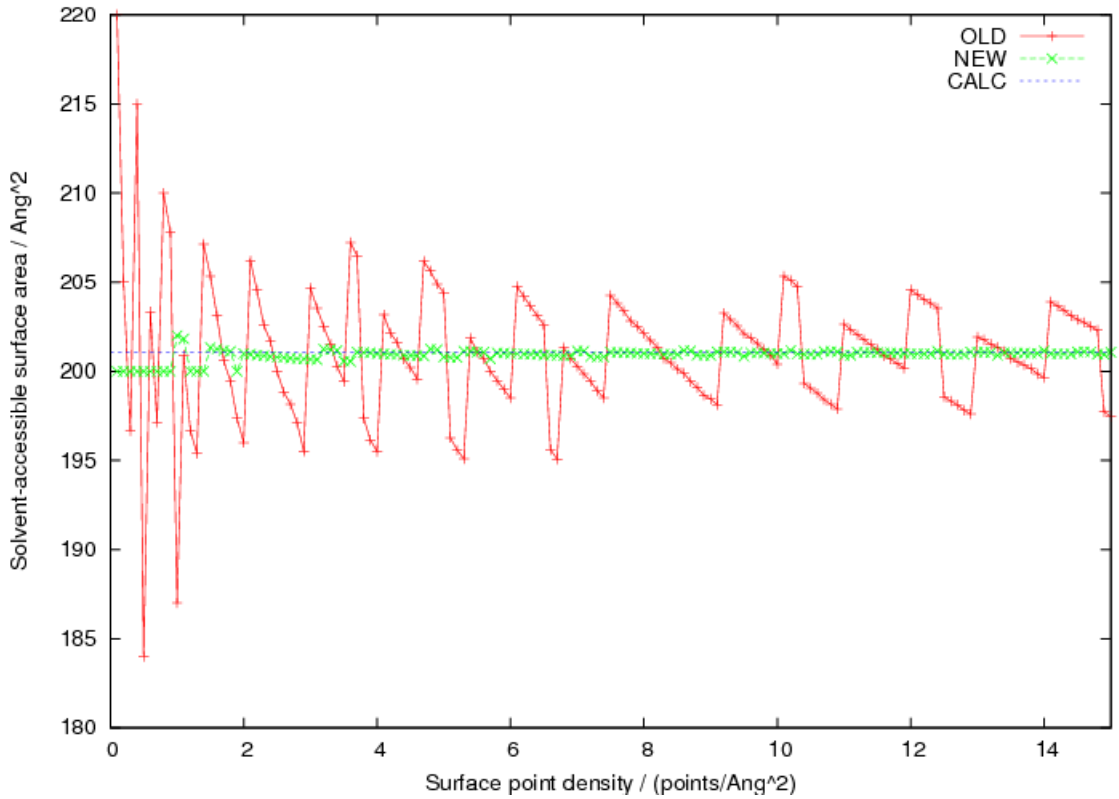
For reference here's the exact PDB file that was used for the surface area calculation for the 2-atom molecule:

```
CRYST1   57.023   64.684  188.747  90.00  90.00  90.00 P 21 21 21
ATOM      4  C   LEU A  22       0       0      -1.8    1     0
ATOM      4  C   LEU B  22       0       0       1.8    1     0
```

The errors arise because of a non-uniform distribution of points over the sphere: the problem is that the algorithm used distributes the points in circles of constant latitude but takes no account of the relative positions of points in adjacent circles. I looked up better algorithms and found a spiral distribution algorithm (Saff & Kuijlaars, 1997) which distributes a specified number of points in a spiral pattern over the surface of the sphere; it's a deceptively simple algorithm, the Fortran code (about half as many lines as the original point-generating code!) is:

```
DO I=1,N
  CTHETA=2*REAL(I-1)/(N-1)-1
  STHETA=SQRT(1-CTHETA**2)
  IF (I.EQ.1 .OR. I.EQ.N) THEN
    PHI=0
  ELSE
    PHI=PHI+3.6/(STHETA*SQRT(REAL(N)))
  ENDIF
  X(1,I)=R*STHETA*COS(PHI)
  X(2,I)=R*STHETA*SIN(PHI)
  X(3,I)=R*CTHETA
ENDDO
```

In the code N is the number of points desired (surface area of sphere x surface point density to nearest integer), R is the radius of the sphere (atomic van der Waals radius + probe radius) and X is the output array of Cartesian point co-ordinates (where (R, θ, φ) are the corresponding spherical polar co-ordinates).



With this simple modification I get the results shown (labelled NEW), which are clearly much more accurate and more stable over a wide range of surface point density values. To be fair, the arrangement of atoms used in the test, with the interatomic vector along the **z** axis, is the worst possible case as then non-uniformity in the distribution at the polar regions where the atoms are in contact has the biggest effect. In real structures where the interatomic vectors are essentially in random directions, the average error is undoubtedly lower. With the new version a surface point density of 15 points/Å^2 seems to provide a reasonable trade-off between accuracy and computation time, so the default value has been reset to this.

The updated version of AREAIMOL will be in the next release of CCP4.

References

Briggs, P.J. (2000). CCP4 Newsletter No. 38, CCLRC, Daresbury, http://www.ccp4.ac.uk/newsletters/newsletter38/03_surfarea.html

Saff, E.B. & Kuijlaars, A.B.J. (1997). The Mathematical Intelligencer, **19**, 5-11, http://www.math.vanderbilt.edu/~esaff/texts/161.pdf

Baubles: Making the World a Better Place for Logfile Viewing

Peter Briggs[*] *and Kevin Cowtan*[**]

[*]*CCP4, CSE Department, STFC Daresbury Laboratory, Warrington WA4 4AD*
Email: p.j.briggs@dl.ac.uk

[**]*Structural Biology Laboratory, University of York, Heslington, York YO10 5YW*
Email: cowtan at ysbl.york.ac.uk

Introduction

It is a perennial problem with running many CCP4 programs, that in order to understand the results of a program run it is often necessary to look at the log file from that program and this can sometimes be quite challenging. Traditional methods such as viewing the "raw" log file using the UNIX/Linux `more` command in a terminal window (see figure 1a) have given way to tools such as the file viewer and loggraph programs in CCP4i (figure 1b). However the log file can still be difficult to navigate, particularly if it consists of the output from many programs run in sequence - as is often the case for CCP4i tasks.

Figure 1a) *Log file viewed using* `more` *in a Linux terminal window.*

Figure 1b) *Log file displayed using CCP4i viewer, with graphs shown in separate loggraph window.*

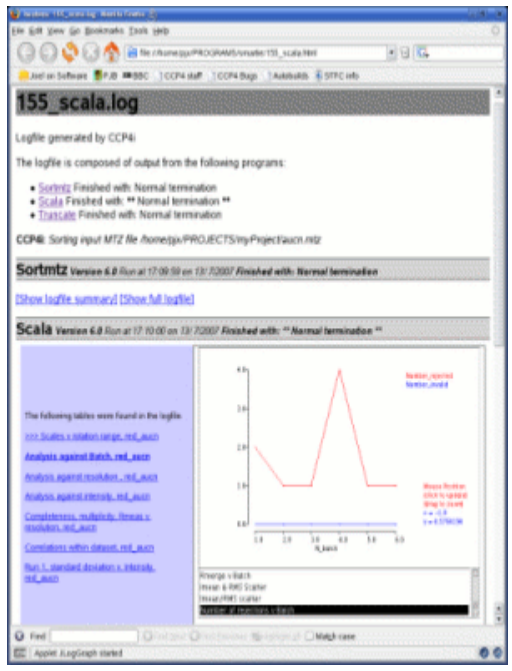


Figure 1c) *Log file reformatted by* `baubles` *and displayed in web browser with inline graphs in JLogGraph.*

Baubles is an experimental program that attempts to make log file navigation and intepretation easier, by reformatting them into HTML so they can be viewed in a web browser (figure 1c).

It aims to work with the existing log file structure (for example, the standard program banners, warning and termination messages, and summary tags) so that it can work with the existing CCP4 programs without requiring substantial modifications. It uses a (substantially improved) version of the JLogGraph Java applet to show log graphs inline, and uses JavaScript to add interactivity (for example, the ability to toggle between "summary" and "full" logfile displays) using JavaScript.

Baubles is written in Python and uses the `smartie` module to do the work of interpreting log files (see the article in newsletter 46: http://www.ccp4.ac.uk/newsletters/newsletter46/articles/smartie.html). It runs on Linux and Windows systems and the generated output should be viewable using both Firefox and Internet Explorer.

Some examples of logfiles reformatted using baubles

The best way to get a feel for what baubles does is to look at a few examples:

- **Scala:** 57_scala.html (see the original log file)
- **Buccaneer with Refmac5:** 393_buccaneer_ref.html (see the original log file)
- **Refmac5:** 156_refmac5.html (see the original log file)
- **Truncate:** 28_truncate.html (see the original log file)

The Scala and Buccaneer examples each include a "results" section, which is transformed by baubles and placed in a prominent position at the start of each program summary. Similarly the Truncate example includes some warning messages that have been extracted and displayed prominently by baubles.

Some other examples can also be found at http://www.ccp4.ac.uk/peter/logfiles/.

Limitations

Baubles is still very much an experiment and there are a number of limitations, for example:

- Some programs don't (for whatever reason) produce particularly useful summaries in their log files, so the reformatted version output from baubles isn't always that much more useful. You can't make a silk purse from a sow's ear.
- Some programs aren't readily recognised by baubles because they don't output "standard" CCP4 program banners or other features that baubles can interpret.
- The reformatted log file from baubles doesn't always help with understanding the outcomes of a task, particularly when many programs are run together in a single script.

Trying it out

You can give baubles a try now - a version of baubles is available for download from:

- ftp://ftp.ccp4.ac.uk/pjx/ccp4/smartie/baubles-0.0.5.tar.gz

This file includes an installer script that will update CCP4 6.0.2 to make baubles available from CCP4i (sorry it only installs into Linux and UNIX-type systems at the moment - see the README file for instructions on how to manually install the update for Windows).

Once baubles has been installed, any new jobs that are run in CCP4i will also run baubles at the end to generate an "annotated" version of the log file which can then be accessed via a "View Annotated Log in Web Browser" option in the "View Files from Job" menu (see figure 2).

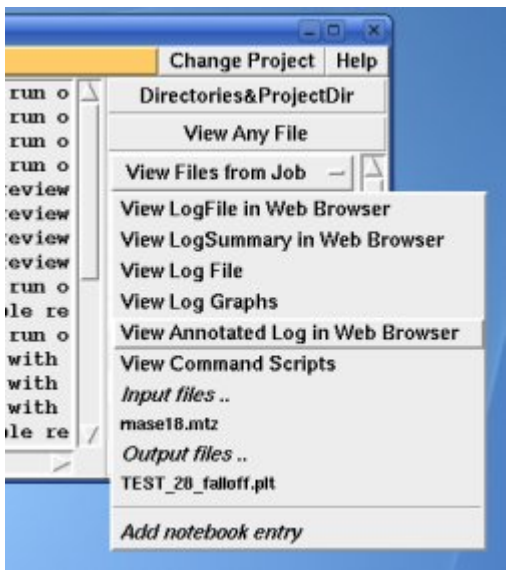


Figure 2) *The "View Annotated Log..." option for accessing HTML versions of log files generated by baubles.*

If you're a program developer

You can make your program output more "baubles-compatible" by following a few guidelines:

- Use the standard CCP4 banners to flag the start (and end, if you're adventurous) of the output from your program - cut and paste from an existing program (but be careful not to add extra #'s!).
- Use the $TABLE markup for tables of data that should be displayed as graphs.
- Write summary tags <!-SUMMARY_BEGIN--> and <!--SUMMARY_END--> into your program output to enclose any sections that should be particularly interesting to users, for example the key results at the end of the program run.
- Use the $TEXT markup for warnings or informational messages. For example, warning messages can be flagged using:
-
-
```
$TEXT:Warning: $$ Stuff here is ignored $$
 Something very important that the user really ought to know about!
$$
```

  while a "result" section can be denoted by:

```
$TEXT:Result: $$ Stuff here is ignored $$
      Put details of the major results here.
      $$
```

  (Note that there should only be a single "result" section per program log). In both cases, the messages will be extracted and displayed prominently within the baubles-annotated log file. In the case of the "results" section, additional "magic" formatting is applied in baubles to make the information more readable (for example by putting tabulated data into a HTML table).

  Examples of warning messages can be seen in the Truncate example (28_truncate.html) and of the results section in the Scala and Buccaneer/Refmac5 examples (57_scala.html and 393_buccaneer_ref.html).

Information on the $TABLE and $TEXT markup formats can be found in the CCP4 documentation, for example at http://www.ccp4.ac.uk/dist/html/loggraphformat.html. In addition, the article on smartie in the previous newsletter has some information about the kind of log file features that baubles is able to recognise. You can also email us and we'll try to help make your program output more baubles-compliant and (hopefully) more user-friendly.

If you're a program user

Give baubles a try on your favourite programs, and send feedback to us about how we might improve it. Also send feedback to program developers suggesting which bits of program log file you would like to see in a summary.

The Future of Baubles

Baubles is still quite experimental, however we intend to incorporate it in some form in the next release of CCP4. Beyond that we hope to look at ways of improving baubles reformatting to make it more useful operating on "standard" logfiles, and at the same time to put pressure on program developers to make improvements to the "raw" log files output from their programs. Beyond that we would also like to investigate ways to link to other relevant resources and possibly include "commentaries" on the program output.

Credits

Baubles was originally written by Peter Briggs and is based on the "smartie" Python module. It has since undergone several iterations of development by both Peter Briggs and Kevin Cowtan. Kevin also produced the original changes to CCP4i to integrate baubles into it, along with the installer script. Both have subsequently been adapted by Peter.

The JLogGraph Java applet was written by Kevin Cowtan who has also made several recent substantial improvements.

Phil Evans has provided some invaluable feedback on both baubles and JLogGraph, for which PJB and KC are both very grateful.

MOLECULAR REPLACEMENT AS A *PHASING* METHOD: MULTIPLE TRANSLATION FUNCTION

Andreas Bühler[1] , Vladimir Y. Lunin[2] and Alexandre Urzhumtsev[1]

[1]*Physics Department, Nancy-University, 54506 Vandoeuvre-les-Nancy, France*
[2]*Institute of Mathematical Problems of Biology, Russian Academy of Sciences, 142290 Pushchino, Moscow Region, Russia*
[1]*Departement of Structural Biology, IGBMC, CNRS-INSERM-ULP, 1 rue L.Fries, 67404 Illkirch, France*

Synopsis

Simultaneous use of multiple translation functions may solve the phase problem even when individual translation functions fail to identify the solution by the conventional molecular replacement procedure.

Keywords

Molecular replacement, translation function, persistent solution, cluster analysis, phasing

Abstract

Molecular replacement is a very attractive tool for structure determination because it uses a single experimental data set and also because it gives immediately a starting model, or at least its major part. The method is based on the idea that the search model is sufficiently similar to the molecule under study so that positioned correctly it reproduces the best possible experimental structure factor magnitudes. Therefore, the optimal position of the model can be recognised by this feature. Unfortunately, this hypothesis fails and the method does not work when the model is too incomplete or too different from the structure under study. Recently, several new approaches have been suggested, either varying automatically the model with a hope to find a better one or taking into account this imperfection of the current model. Alternatively, the whole strategy of the search of a single extremum (usually, the global one) of a single target can be revised. Previously, advantages of a simultaneous analysis of several rotation functions for solution of difficult molecular replacement cases have been demonstrated. This article suggests the next step in this procedure. Knowing approximate orientations of an imperfect atomic model (or several models), one may obtain a starting image of the electron density when the conventional translation search fails to recognise a single optimal position of the search model.

1. Introduction

Crystallographic literature always considers the molecular replacement method (Rossmann & Arnold, 2001, as a review for the method) as a method to solve the phase problem. In fact, one of features of this technique is that it gives not only the phase set but also an approximate atomic model which simplifies further structure determination. This requires the identification of the *unique* position of the search model. The main hypothesis of the molecular replacement is that at this *optimal* position the model structure factor magnitudes correspond to the experimental values as good as possible. The phases of the structure factors calculated from such an optimally positioned model may be used to calculate Fourier syntheses, to correct and complete the model accordingly and to refine it after all. The search for the best fit may be done either by simultaneous rotation and translation the model(s) (for example, Kissinger *et al*., 1999; Glycos & Kokkinidis, 2000) or by execution these two steps one after another to gain computing time as it is implemented in most of molecular replacement packages.

When the search model differs significantly from the structure under study, there is no reason to believe that for its optimal position the best match of computed and observed structure factors is achieved. On the contrary, the presence of such *irremovable* model errors (Lunin *et al*., 2002) may lead to an incorrect model position. Traditionally, in such a situation crystallographers vary the model or/and the target (for example, resolution of the data set, integration radius etc.). A great step in molecular replacement was done by introducing maximum-likelihood targets (Read, 2001; Storoni, 2004) that can statistically take into account the model imperfections, namely incompleteness and mean discrepancy. A new and more appropriate model can be obtained automatically by model modification (for example, Suhre & Sanejouand, 2004) or model building (Keegan & Winn, 2007).

Alternatively, one may change completely the search strategy. We still suppose that the search model, although being imperfect, is somehow relevant to the structure. It contains some structural information and in its optimal position it reproduces crystal structure factors reasonably well but not necessarily with the best magnitude correspondence. This means that a molecular replacement target has a peak at this position even when this peak is not the highest one. Of course, the target may have a number of spurious peaks. When we update the model or modify the target, the peak for the correct position usually stays whereas spurious peaks may appear or disappear. Therefore, we do not look anymore for the *global extrema* of targets knowing that they may correspond to a wrong position. Instead, we are looking for the *most persistent* peak that, following this logic, stands for the solution of the molecular replacement problem (**Fig. 1**). Making several searches, as expected, increases the information available and simultaneous analysis of multiple searches may allow to identify the solution non visible in a single search.

Study of multiple rotation functions (Urzhumtsev & Urzhumtseva, 2002) confirmed that the correct model orientation in difficult cases can be identified using this principle. The current article extends this principle to the next step of molecular replacement, the translation search, leading to the final solution. The presented technique is based on the simultaneous use of results of multiple searches. In what follows for simplicity we consider the molecular replacement search with a single body.

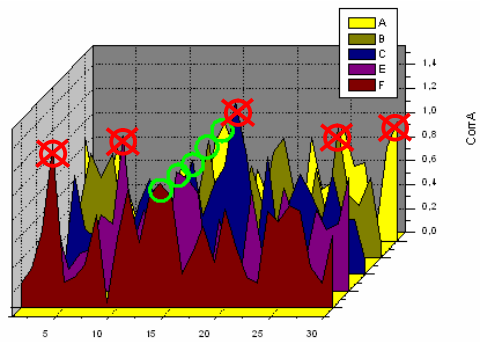


Fig. 1. Schematic presentation of the search for a persistent signal. The answer corresponds to the peak presented in all search functions (green circles) and not to any of the highest peaks in individual functions (red circles).

2. Multiple translation function

2.1. Comparison of search results

To search for a persistent signal, one needs to define the closeness of two peaks of the target. A conventional measure as the root-mean-square (*rms*) is not a good choice for translation searches. First, when the search model is significantly different from the structure under study, the *exact* orientation and position have no strict meaning. In slightly different orientations obtained after rotation analysis, a search model may have a good superposition with the answer by one or another part. For this reason, with slightly different rotation peaks, the translation search may give different positions with a large *rms* discrepancy between them. Second, it is even more difficult to compare the position of *different* models used in the same molecular replacement project. Third, different positions of imperfect models may correspond to equally good solutions equivalent in terms of electron density. For example, a b-sheet composed of *n* more or less equal strands may be roughly well superimposed in different ways with a larger (*n*+1 strands) or shorter (*n*-1 strands) b-sheet in the crystal. Similarly, a helical model may be reasonably well superimposed with the structure being simultaneously rotated and translated accordingly to the pseudo-helical symmetry.

In fact, molecular replacement targets do not fit the model to the structure atom by atom but compare the magnitudes of the Fourier coefficients of the density generated from the model with the corresponding experimental values. An example is the magnitude correlation

$$CorrA(\mathbf{t}) = \frac{\sum_{\mathbf{s}} F_{obs}(\mathbf{s}) F_{calc}(\mathbf{s},\mathbf{t})}{\left(\sum_{\mathbf{s}} F_{obs}^{2}(\mathbf{s})\right)^{\frac{1}{2}} \left(\sum_{\mathbf{s}} F_{calc}^{2}(\mathbf{s},\mathbf{t})\right)^{\frac{1}{2}}}$$

where **t** is the translation vector of the of the model initially positioned at the origin. An early substitution of the atomic model by the Fourier coefficients of its density is one of the major advantages of molecular replacement packaged starting from *AMoRe* (Navaza, 1994).

If we agree that the search model(s) is (are) significantly different from the structure under study, we may decide to use only a part of the information obtained from molecular replacement, namely the phases of structure factors. Like in

experimental phasing methods, these phases will be used in model building by the electron density maps interpretation. We do not search any more for the atomic positions while of course the search models may be always used as a guide.

Comparison of two peaks of the translation search, done with different search models each in its own orientation, may be done by comparison of the *phases* calculated from the corresponding models. An example is the weighted phase correlation (Lunin & Woolfson, 1993)

$$CorrP(\mathbf{t}_1,\mathbf{t}_2)=\frac{\sum_{\mathbf{s}} F_{obs}^2(\mathbf{s})\cos\left(\mathrm{j}_{calc}(\mathbf{s},\mathbf{t}_1)-\mathrm{j}_{calc}(\mathbf{s},\mathbf{t}_2)\right)}{\sum_{\mathbf{s}} F_{obs}^2(\mathbf{s})}$$

It correspond s to the correlation of two Fourier syntheses both calculated with the experimental structure factor magnitudes but with two different phase sets, $\{\mathrm{j}_{calc}(\mathbf{s},\mathbf{t}_1)\}$ and $\{\mathrm{j}_{calc}(\mathbf{s},\mathbf{t}_2)\}$, respectively. In fact, to cope with the problem of a different possible choice of the origin of the unit cell (Lunin & Lunina, 1996) a modified expression should be used where **u** stands for all possible choices of the origin for the given space group

$$CorrP(\mathbf{t}_1,\mathbf{t}_2)=\max_{\mathbf{u}}\frac{\sum_{\mathbf{s}} F_{obs}^2(\mathbf{s})\cos\left(\mathrm{j}_{calc}(\mathbf{s},\mathbf{t}_1)-\mathrm{j}_{calc}(\mathbf{s},\mathbf{t}_2+\mathbf{u})\right)}{\sum_{\mathbf{s}} F_{obs}^2(\mathbf{s})}$$

2.2. Two strategies of multiple searches

Eventually, several possible strategies may be investigated. In a simplest one, the search models are positioned randomly (both orientation and translation) in the unit cell. For each model structure factors are calculated and the phase sets are selected if the magnitude correlation is high enough. As an option, only the model orientation is chosen randomly (or generated systematically in a grid) followed by a fast translation search from which the peaks are selected. Both variants of this strategy are similar to FAM searches (Lunin *et al*., 1995, 1998) but use much more information about the structure. After having selected a (large) number of phase sets, their mutual closeness is studied by cluster analysis procedures (for example, see Lunin *et al*., 1990, 1995). Then the average phase values and corresponding figures of merit for the largest cluster are computed and used to calculate the Fourier syntheses. This approach may require a lot of computer time to accumulate enough of good combinations of model orientation and position.

In the second strategy the rotation search precedes the translation search. First, multiple rotation function analysis suggests a cluster of model orientations (several models in several similar orientations). This requires a preliminary orientation alignment of the search models. Then, the translation searches are done only with the models from this orientation cluster. Here an obvious gain in CPU time is compensated by a risk of choos ing a wrong orientation cluster. Eventually, one may try to invert this procedure and to solve first the translation problem and only then to search for the model orientation.

In the current project, we had no possibility to check various strateg ies and some results of only consecutive rotation-translation searches are shown below.

3. Numerical tests with multiple translation functions

3.1. Experimental data

Corn Hageman factor inhibitor (CHFI) previously has been shown as one of the difficult molecular replacement cases (Chen *et al*., 2000). The protein crystallises in space group $P4_22_12$ with the unit cell parameters $a = b = 57.12$ Å, c = 80.24 Å. Twenty NMR models (Strobl *et al*., 1995, PDB code 1bip) were used for the molecular replacement search. They are reasonably similar to the final model (**Fig. 2**) obtained by Behnke *et al*. (1998, PDB code 1bea). However, the difference was enough to make the molecular replacement solution very difficult with conventional tools.

For the final model, structure factors have been calculated and their phases were used to compare with the phases obtained from the search models.

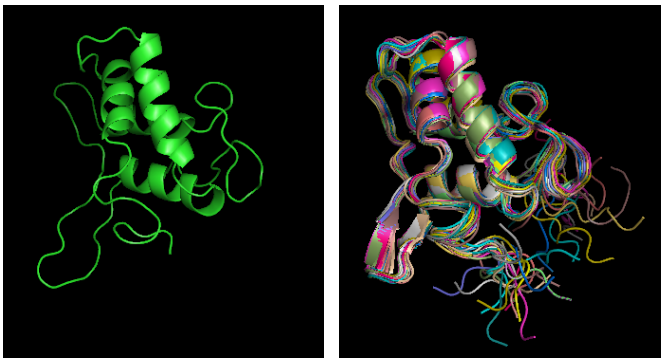


Fig. 2. Ribbon view of the CHFI final structure (left) and superimposed twenty NMR models (right) used for molecular replacement searches

3.2. Rotation searches and multiple rotation function

Previously, Urzhumtsev & Urzhumtseva (2002) used this data set for studies with multiple rotation functions. None of *AMoRe* individual rotation function, calculated with each of twenty NMR models at various resolutions, gave a signal allowing to interpret it as the peak for the correct model orientation. Then the peaks for twenty rotation functions calculated at the resolution 4-10 Å were studied together. Provided that some angle-distance cut-off κ is chosen, the total set of peaks may be split into a number of clusters. Inside a cluster, the models oriented accordingly to their peaks, may be superimposed to each other by a rotation of less than κ degrees. On the contrary, this is impossible for models from different clusters (Urzhumtseva & Urzhumtsev, 2002).

The authors expected that the peak for the correct orientation exists for most of rotation functions and therefore should belong to the largest cluster of orientations. Indeed, it was the case when the cut-off varied in quite large reasonable limits (**Fig. 3**).

For the set of rotation functions calculated with the diffraction data at 5-10 Å resolution, the correct orientation also belonged to the largest cluster when the cut-off level was 6.5° and became slightly smaller when the cut-off changed (not shown here).

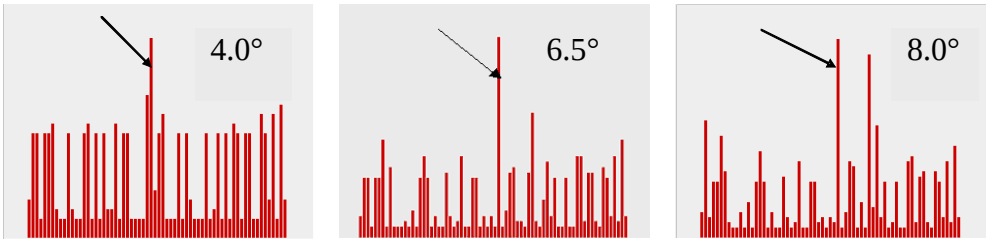


Fig. 3. Size of clusters of close model orientations corresponding to highest peaks of twenty rotation functions calculated with NMR models of CHFI at the resolution 4-10 Å. The three images shown correspond to a different choice of the angular cut-off level. For all three cases the correct model orientation belongs to the largest cluster indicated by arrow.

3.3. Conventional translation searches

First translation tests were done with the exact model taken in its approximately correct orientation; an artificial rotation error of only 2° was introduced. At a conventional resolution of 5-15 Å, the translation function calculated with *AMoRe* had a peak with the magnitude correlation 0.85 versus 0.76-0.80 for a large group of next peaks. This characteristic unambiguously identifies the peak as the solution. The phases calculated from the model positioned accordingly have the weighted phase correlation of 0.84 with the phases from the correct solution, which is much larger than the correlation for a phase set calculated from the model positioned in any other peak. **Fig. 4** shows the results of this search in the form of a two-dimensional distribution of points. For the model positioned accordingly to each peak of the translation function its structure factors are calculated. The correlation of their magnitudes with the experimental magnitudes and the correlation of their phases with the phases from the exact solution are used as coordinates of the point in the diagram. The correct solution is clearly distinguished by its horizontal coordinate (the criterion to choose the solution) and gives much better phases (a very large gap along the vertical axis).

The same calculations were repeated for the exact model but with a significant error in its orientation, namely 10°. Now the translation function, calculated as the correlation of magnitudes, has a number of peaks of a similar height without a significant gap between the highest one and the next one. More important, this highest peak does not correspond to the position with the highest correlation of phases. The latter is hidden in the bulk of other peaks and cannot be distinguished by *CorrA*. Moreover, even when we can identify this ‘best’ peak, the corresponding phases are quite wrong with a very low correlation *CorrP*, only about 0.20.

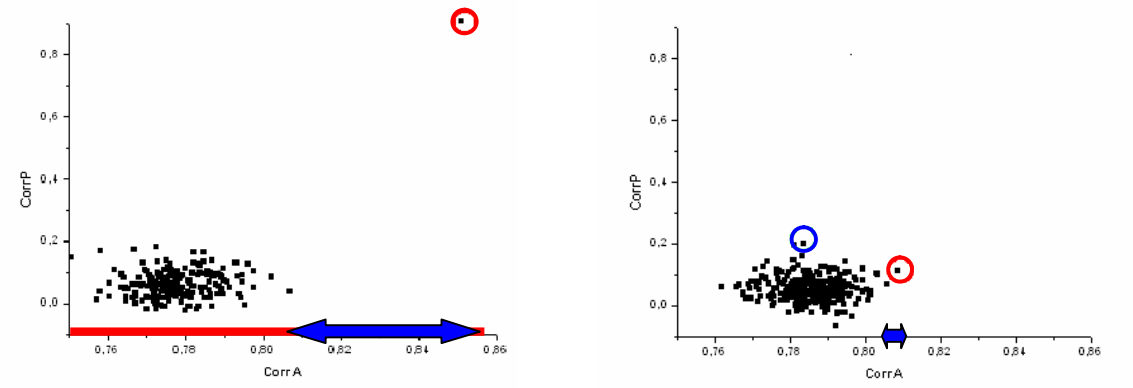


Fig. 4. Results of the translation function search with the exact model for CHFI at the resolution 5-15 Å. Each point corresponds to a model positioned accordingly; coordinates of the point are the correlations of the magnitudes (horizontal axis) and phases (vertical axis) of the calculated structure factors with the experimental magnitudes and the phases

calculated from the exact solution. Left image shows the results for the model in a relatively correct orientation (orientation error 2°). Right image shows the results for the model with a significant error in its orientation (10°). Red circle indicates the model with the higher correlation of the magnitudes. In the right image it is different from the model with the best phase values (blue circle) that cannot be identified by its magnitude correlation among other models.

3.4. Translation functions for multiple models

Accordingly to the main idea, we took all twenty NMR models together and calculated for each of them the standard *AMoRe* translation function (Navaza & Vernoslova, 1995) at the resolution of 5-15 Å. As previously for the study case with the known answer, magnitude and phase correlations have been calculated for each position of the model and the results of each search were presented in the form of the two-dimensional distribution (*CorrA*, *CorrP*). For each test, peaks from all translation functions were taken together, about 300 highest of them were selected and analysed. The three tests presented below differ in accuracy of model orientations.

In the first test, all NMR models were taken in the optimal orientation, obtained from their best superposition with the refined structure. Even in this idealised case the highest peak (magnitude correlation near 0.775) indicates a wrong solution with the phase correlation near 0.2 (**Fig. 5, left**). However, the second solution, in the decreasing order of *CorrA*, has a quite good phase correlation close to 0.60. The best solution, in terms of phases, has a magnitude correlation *CorrA* = 0.76 and is 'hidden' among tens of other solutions. In general, this behaviour completely reminds the behaviour of FAM models (Lunin *et al*., 1995).

Then two other scenarios were tried, based on orientations of the search models selected from the peaks of twenty rotation functions. In the second test, the models were taken in the orientations from the best cluster obtained by the multiple rotation function at the resolution 5-10 Å with the cut-off level 6.5°. In the third test they were taken in the orientations determined from the largest cluster of the multiple rotation function calculated at the resolution 4-10 Å when the cut-off level is equal to 4.0°. For both tests the highest peak in all rotation functions had *CorrA* = 0.77, was practically indistinguishable in height from the next peaks and obviously corresponded to a wrong model position (**Fig. 5**, **centre and right**). The corresponding phase correlation is 0.1 and 0.3, respectively. For less precise set of orientations (second test), the highest phase correlation was near 0.50; for more precise orientations obtained after rotation searches at 4-10 Å (third test) it was slightly better, 0.55 (**Fig. 5, right**). However, in both tests the best model (in terms of phase correlation) is hidden among tens of other peaks with the *CorrA* near 0.76 and therefore cannot be identified without knowing the answer. (One may imagine that all these points are projected on the horizontal axis and we need to identify the correct solution knowing only this projection).

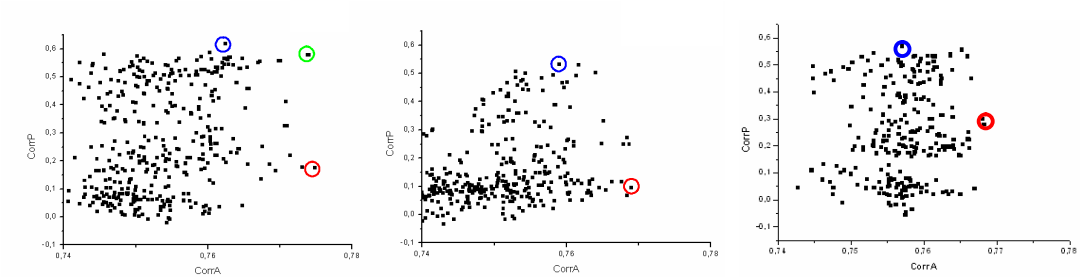


Fig. 5. Results of the translation search for CHFI at the resolution 5-15 Å with twenty NMR models. Each point corresponds to a peak of the translation function. For the model positioned accordingly to each peak a set of structure factors was calculated. Correlation *CorrA* of their magnitudes with the experimental data and the correlation *CorrP* of their phases with those calculated from the exact model are the point coordinates (horizontal and vertical axes, respectively). Red circle indicates the highest peak, the blue one indicates the peak resulting in the best phases. (Left) Search models are in their optimal orientations. (Centre) Orientations chosen from the peaks of 20 rotations functions; 5-10 Å resolution, 6.5° cluster cut-off. (Right) Orientations chosen from the peaks of 20 rotations functions; 4-10 Å resolution, 4.0° cluster cut-off.

3.5. Phases from cluster analysis

In order to identify the signal persistent in the ensemble of the translation functions, the obtained phase sets have been analysed by a clustering procedure (see for example Lunin *et al*., 1990, 1995) for each of the three tests. **Fig. 6** shows corresponding cluster trees. Phase sets are shown by points at the horizontal axis. The points are joined (corresponding phase sets are merged) at the height proportional to the 'distance' between the phase sets. Each of the trees is clearly split in a number of clusters indicated by letters. The larger is cluster, the more models are contained. The lower the cluster in the tree, the more compact it is, and the closer are the model phases composing this cluster. According to the initial hypothesis, the cluster for the best phase set, corresponding to the group of models giving the persistent signal, should be the low and large one. Indeed, in all three cluster trees such a cluster can be identified (selected in white) and it corresponds actually to the best phase set (**Table 1**). The mean figure of merit is the measure of the dispersion of the phase values in the cluster around their average. We remind that the number of phase sets in each cluster and its mean figure of merit are obtained directly from calculated phase sets and that they can be used to select the cluster. On the contrary, the phase correlation *CorrP* is available only for known structures.

For the exactly oriented models the cluster for the correct solution (cluster B) is very different from other clusters by its size and compactness. Interestingly, the mean phase correlation is larger than for any individual phase set belonging to it. The mean figure of merit above 0.5 suggests usefulness of these phases (Lunin & Woolfson, 1993).

If the choice between a few clusters is not clear, a larger cluster including all of them can be taken. **Table 1** shows that in this case the mean phase correlation does not decrease drastically however this reduces the mean figure of merit calculated for each reflection. Finally, all selected phase sets can be averaged together. **Table 1** shows that even in this case the mean phase values are good enough. In other words, the phases from wrong positions are distributed more or less uniformly in comparison with the phases from the correct cluster. Nevertheless, the overall figure of merit becomes unreasonably low and does not assure the quality of selected phases.

For the models oriented approximately, the cluster tree (**Fig. 6, centre**) still distinguishes one cluster that is more compact then others (also cluster B). Its vertex is higher than it is in the previous test showing a larger dispersion of the models inside it. Even if this cluster is larger than cluster C, the corresponding figure of merit is higher than that for C and the choice could be done for this cluster. In case of doubt, B and C can be considered together which again does not decrease phase correlation but does decrease the mean figure of merit.

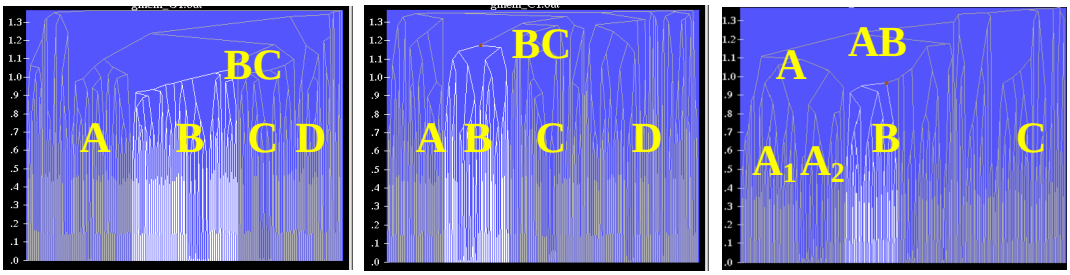


Fig. 6. Cluster trees for the phase sets obtained after multiple translation function analysis with the twenty NMR models of CHFI. (Left) Models in the optimal orientations obtained from their superposition with the final model. (Centre) Models in approximate orientations obtained from the multiple rotation function analysis at 5-10 Å and the cluster selected with the cut-off 6.5°. (Right) Models in more precise orientations obtained from the multiple rotation function analysis at 4-10 Å and the cluster selected with the cut-off 4.0°. Letters indicate individual clusters of their groups referred in **Table 1** and in text.

Table 1. Characteristics of the principal clusters of the cluster trees shown in **Fig. 6**.

test	Cluster	A	B	C	D	AB	BC	all
1 : ideal orientation	$N_{variants}$	61	**106**	57	26	-	163	312
	CorrP	0.33	**0.67**	0.38	0.15	-	0.64	0.60
	<fom>	0.53	**0.57**	0.53	0.54	-	0.49	0.31
2 : imprecise orientation	$N_{variants}$	56	**65**	50	61	-	120	308
	CorrP	0.08	**0.55**	0.30	0.13	-	0.52	0.51
	<fom>	0.32	**0.49**	0.47	0.35	-	0.35	0.17
3 : good orientation	$N_{variants}$	74	**61**	33	-	179	-	273
	CorrP	0.33	**0.62**	0.05	-	0.33	-	0.52
	<fom>	0.56	**0.61**	0.61	-	0.56	-	0.29

In the intermediate case of imprecise model orientations but with a smaller error (**Fig. 6, right**) there is a cluster low in the tree and therefore compact in phase space (again, cluster B). It is lower than the cluster B in the previous case indicating that its phase sets are closer to each other. Different to the first test with the optimally oriented models where the choice is unambiguous, the cluster A may be considered as an alternative candidate with a slightly larger number of sets. However, in fact the cluster A is composed of two subclusters of a smaller size, A_1 and A_2, each of them smaller than B. The mean figure of merit for B is larger than that for A, suggesting that B is the correct choice. As a final check the maps calculated with these two phase sets are analysed. They are quite different in quality. The map calculated at 5 Å resolution with the phases from the cluster B (**Fig. 7, left**) shows continuous density where a-helices are clearly seen. In the equivalent map with the phases from the cluster A the density is seen as a set of isolated blobs and can be hardly interpretable (**Fig. 7, centre**).

To complete the study, one more test with the multiple translation function has been done with the models in the orientations from the alternative, wrong cluster obtained for the rotation functions at resolution of 5-10 Å. The corresponding cluster tree for the multiple translation function shows a number of clusters, more or less of the same size and compactness, none of them are large and compact enough to fulfil the presented conditions (**Fig. 7, right**). Such a tree, since it does not show a persistent signal, may serve as an indicator of a probably wrong choice of the set of model orientations.

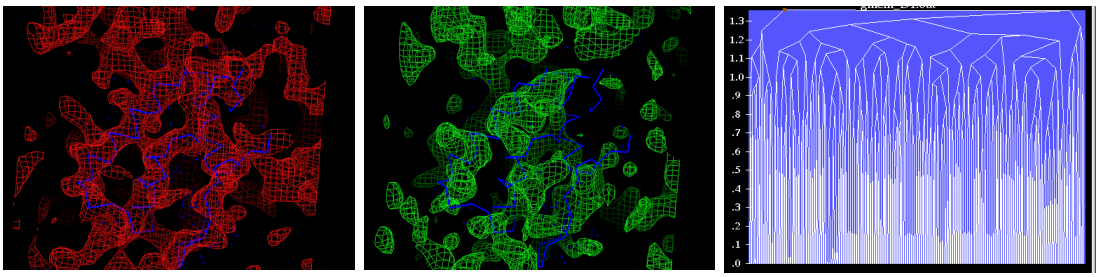


Fig. 7. (Left) 5 Å-resolution Fourier map calculated with experimental structure factor magnitudes and phases from the cluster B (**Fig.6, right**). A main chain of the correct model is superimposed. (Centre) A similar image, the map is calculated with the phases from the cluster A. (Right) Cluster tree for the multiple translation search with the models from a wrong orientation cluster.

4. Discussion

When the search models in molecular replacement studies are insufficiently similar to the structure under study, the input information is weak. As a consequence one may fail to find strong information in the answer, namely a single position for one of these models. In such difficult cases we suggest to search for an answer in a weaker form, only as phases of structure factors and corresponding Fourier syntheses. These maps will be used then to build a model as for any other phasing methods.

The current tests show that a simultaneous use of several sources of information, namely several translation functions, may amplify the information and find the answer in the situation when none of individual translation functions does. While in practice the persistence of signal is an important feature to check the answer, it is not used directly as a target. The new approach is based on the persistence and does not result in a single model positioned in the unit cell. This changes completely the traditional way of a search for the answer as a single set of model parameters (rotation and translation) corresponding to the global maximum (minimum) of a single target.

The first tests with the multiple translation function reported here illustrate the potential of this approach and open a number of questions to be addressed in the future.

First, the best strategy should be established. The question is whether to do a simultaneous search for the model orientation and translation or to separate them. The current procedure can be applied in the same way in both cases, but with different drawbacks. Determining in advance an approximate model orientation saves CPU-time but increases the risk of missing the solution due to a wrong set of model orientations, whereas the simultaneous variation of orientations and translations is

much safer but considerably time-consuming. Probably the preference for one or another strategy depends on the practical situation.

Second, the question is how one may extract maximal information from such searches. When the searches are done at a given resolution, the phases can be calculated from the models and then clustered at a different resolution, eventually at a higher one. This can increase the amount of information in the maps.

One more question is the performance of this approach when the crystal structure is composed of several domains and especially when these domains are somehow similar (or when the structure has an internal symmetry). Here the translation function 'in half of trials' may put the search model at the place of one domain, and 'in the second half of trials' at the place of another one. Probably operating at the level of phases and their similarity may simplify the solution for such crystals.

To make this approach a routine procedure, special software should be developed both for clustering analysis and for merging it with the molecular replacement packages.

Acknowledgement

AB participated in this project in the frame of his *M2* stage of the 'cursus intégré' of the universities of Nancy (France), Saarbrücken (Germany) and Luxembourg. VYL and AU were supported by grant RFBR 05-01-22002_CNRS. The authors thank Markus Sander for his participation at the initial stage of this project, Natalia Lunina for an access to the programs from the *FAM* phasing suite and Pavel Afonine for critical reading of the manuscript. *PyMOL* (DeLano, 2002) was used to display and analyse the maps.

References

Behnke, C.A., Yee, V.C., Le Trong, I., Pedersen, L.C., Stenkamp, R.E., Kim, S.-S., Reeck, G.R. & Teller, D.C. (1998). *Biochemistry*, **37**, 15277-15288.

Chen, Y.W., Dodson, E.J. & Kleywegt, G.J. (2000). *Structure*, **8**, R214-R220.

DeLano, W.L. (2002). *The PyMOL Molecular Graphics System,* DeLano Scientific, San Carlos, CA, USA. *http://www.pymol.org*

Keegan, R.M. & Winn, M.D. (2007). *Acta Cryst*., **D63**, 447-457.

Kissinger, C.R., Gehlhaar, D.K. & Fogel, G.B. (1999). *Acta Cryst*., **D55**, 484-491.

Glykos, N.M. & Kokkinidis, M.. (2000). *Acta Cryst*., **D56**, 169-174.

Lunin, V.Yu., Urzhumtsev, A.G. & Skovoroda, T.P. (1990). *Acta Cryst*. **A46**, 540-544.

Lunin, V.Y. & Wolfson, M.M. (1993). *Acta Cryst*. **D49**, 530-533.

Lunin, V.Yu., Lunina, N.L., Petrova, T.E., Vernoslova, E.A., Urzhumtsev, A.G. & Podjarny, A.D. (1995). *Acta Cryst*. **D51**, 896-903.

Lunin, V.Y. & Lunina, N.L. (1996). *Acta Cryst*. **A52**, 365-368.

Lunin, V.Y., Lunina, N.L., Petrova, T.E., Urzhumtsev, A.G. & Podjarny, A.D. (1998). *Acta Cryst*. **D53**, 726-734.

Lunin, V.Y., Afonine, P.V. & Urzhumtsev, A. (2002). *Acta Cryst*. **A58**, 270-282.

Navaza, J. (1994). *Acta Cryst*., **A50**, 157-163

Navaza, J. & Vernoslova, E.A. (1995). *Acta Cryst*., **A51**, 445-449.

Read, R. (2001). *Acta Cryst*., **D57**, 1373-1382.

Rossmann, M.G. & Arnold, E. (2001). *International Tables for Crystallography, Vol. F.*, Kluwer, Dordrecht-Boston-London, pp.263-292.
Strobl, S., Muhlhahn, P., Bernstein, R., Wiltscheck, R., Maskos, K., Wenderlich, M., Huber, R., Glockshuber, R. & Holak, T.A. (1995). *Biochemistry*, **34**, 8281-8293.
Storoni, L.C., McCoy, A.J. & Read, R.J. (2004). *Acta Cryst.*, **D60**, 432-438.
Suhre, K. & Sanejouand, Y.-H. (2004). *Acta Cryst.*, **D60**, 796-799.
Urzhumtsev, A. & Urzhumtseva, L. (2002). *Acta Cryst.,* D**58**, 2066-2075.
Urzhumtseva, L. & Urzhumtsev, A. (2002). *J. Appl. Cryst.*, **35**, 644-647.

Why the moments of E take the values they do

Norman Stein

CCP4, Daresbury Laboratory
Warrington WA4 4AD, United Kingdom

The values of the moments of the normalised structure amplitude E depend on whether or not the data is twinned and can therefore be used as a test for the presence of merohedral twinning. For example the CCP4 Truncate program [1] lists two sets of theoretical values for these moments, one for untwinned data and one for perfectly twinned data. Partially twinned data give rise to values for the moments of E which lie somewhere between these two extreme cases. This article explains how one can calculate these theoretical values. None of the results presented here are new, but it appears to be hard to find their derivation in the protein crystallography literature. Thus the calculations presented here can perhaps best be viewed as an extension of the last section of the 'Basic Maths for Protein Crystallographers' tutorial included in the CCP4 distribution.

The normalised structure factor amplitude E is defined in terms of the reduced intensity I by $E = \sqrt{I/\Sigma}$, where Σ is the mean intensity in a thin spherical resolution shell containing the reflection under consideration. (The reduced intensity is the measured intensity divided by the symmetry factor ε.) The intensities obey the Wilson distribution [1,2], which for acentric, untwinned reflections has probability density function

$$\begin{aligned} p_I(x) &= (1/\Sigma)\exp(-x/\Sigma) \qquad (x \geq 0) \\ p_I(x) &= 0 \qquad (x < 0) \end{aligned} \tag{1}$$

In other words, the probability that the intensity I lies in the range $[x, x+dx]$ is $p_I(x)dx$. Using angle brackets to denote an average over a resolution shell, the n'th moment of E is

$$\langle E^n \rangle = \left\langle I^{n/2} \right\rangle / \Sigma^{n/2} = \int_0^\infty (x/\Sigma)^{n/2} \exp(-x/\Sigma)\, dx \tag{2}$$

Making the change of variable $y = x/\Sigma$ gives

$$\langle E^n \rangle = \int_0^\infty y^{n/2} \exp(-y)\, dy = \Gamma(n/2+1) \tag{3}$$

Using the following properties of the Gamma function [3]

$$\begin{aligned} \Gamma(m+1) &= m! \qquad (m \text{ integer}) \\ \Gamma(z+1) &= z\Gamma(z) \\ \Gamma(1/2) &= \sqrt{\pi} \end{aligned} \tag{4}$$

it is straightforward to calculate the values shown in Table 1 for untwinned, acentric data. In order to do the same for perfectly twinned, acentric data, it is convenient to introduce the characteristic function

$$\tilde{p}_I(\theta) = \langle \exp(i\theta I) \rangle = \int_{-\infty}^{\infty} \exp(i\theta x)\, p_I(x)\, dx \tag{5}$$

which is just the Fourier transform of the probability density function. For the untwinned, acentric Wilson distribution

$$\tilde{p}_I(\theta) = \frac{1}{1 - i\theta\Sigma} \tag{6}$$

For perfectly twinned data, the observed intensities are given by $I = (J + K)/2$, where J and K represent the individual contributions of two reflections whose Miller indices are related by the twinning operator [4]. In order to obtain the characteristic function for such data, we make use of two results from probability theory. First, if I has probability density $p(x)$ then $I/2$ has probability density $2p\,(2x)$. Applying this to (1) gives

$$\begin{aligned} p_{J/2}(x) &= (2/\Sigma) \exp(-2x/\Sigma) \qquad (x \geq 0) \\ p_{J/2}(x) &= 0 \qquad (x < 0) \end{aligned} \tag{7}$$

with characteristic function

$$\tilde{p}_{J/2}(\theta) = \frac{1}{1 - i\theta\Sigma/2} \tag{8}$$

Secondly, if J are K are two independent random variables, then

$$\tilde{p}_{J+K}(\theta) = \langle \exp(i\theta(J + K)) \rangle = \langle \exp(i\theta J) \rangle \langle \exp(i\theta K) \rangle = \tilde{p}_J(\theta)\, \tilde{p}_K(\theta) \tag{9}$$

Applying this to (8) gives the characteristic function for perfectly twinned, acentric data

$$\tilde{p}_I(\theta) = \frac{1}{(1 - i\theta\Sigma/2)^2} \tag{10}$$

A useful property of the characteristic function is that the moments of the intensity distribution can be expressed in terms of the derivatives of the characteristic function evaluated at $\theta = 0$

$$\langle I^n \rangle = \tilde{p}_I^{(n)}(0)/i^n \tag{11}$$

This gives a quick way of calculating the even moments of E. For example, to find $\langle E^4 \rangle$, we have

$$\tilde{p}_I''(\theta) = \frac{3\Sigma^2}{2(1 - i\theta\Sigma/2)^4} \tag{12}$$

so that

$$\left\langle E^4 \right\rangle = \left\langle I^2 \right\rangle / \Sigma^2 = -\tilde{p}_I''(0)/\Sigma^2 = 1.5 \tag{13}$$

In order to calculate odd moments of E, we need to invert the Fourier Transform using contour integration or by consulting tables of Fourier transforms. The result is

$$\begin{aligned} p(x) &= (4x/\Sigma^2)\exp(-2x/\Sigma) \qquad (x \geq 0) \\ p(x) &= 0 \qquad (x < 0) \end{aligned} \tag{14}$$

The n'th moment of E is therefore

$$\left\langle I^{\,n/2} \right\rangle / \Sigma^{n/2} = 4 \int_0^\infty x^{n/2+1} \exp(-2x/\Sigma)\, dx/\Sigma^{n/2+2} \tag{15}$$

The substitution $y = 2x/\Sigma$ then gives

$$\langle E^n \rangle = \int_0^\infty y^{n/2+1} \exp(-y)\, dy/2^{n/2} = \Gamma(n/2+2)/2^{n/2} \tag{16}$$

For centric reflections, the Wilson distribution is

$$\begin{aligned} p_I(x) &= (1/\sqrt{2\pi\Sigma\, x})\exp(-x/2\Sigma) \qquad (x \geq 0) \\ p_I(x) &= 0 \qquad (x < 0) \end{aligned} \tag{17}$$

with characteristic function

$$\tilde{p}_I(\theta) = \frac{1}{\sqrt{1 - 2\Sigma i\theta}} \tag{18}$$

The moments are given by

$$\langle E^n \rangle = \left(1/\sqrt{2\pi\Sigma^{n+1}}\right) \int_0^\infty x^{(n-1)/2} \exp(-x/2\Sigma)\, dx \tag{19}$$

The substitution $y = x/2\Sigma$ yields

$$\langle E^n \rangle = \left(2^{\,n/2}/\sqrt{\pi}\right) \int_0^\infty y^{(n-1)/2} \exp(-y)\, dy = \frac{2^{\,n/2}}{\sqrt{\pi}}\, \Gamma\left(\frac{n+1}{2}\right) \tag{20}$$

Applying the same arguments as for the acentric case to (18), the characteristic function for perfect twinning is

$$\tilde{p}_I(\theta) = \frac{1}{1 - \Sigma i\theta} \tag{21}$$

Note that this is exactly the same as for the untwinned acentric case. This means that the moments for the perfectly twinned centric case are the same as for the untwinned centric case.

	$\langle E \rangle$	$\langle E^3 \rangle$	$\langle E^4 \rangle$	$\langle E^5 \rangle$	$\langle E^6 \rangle$	$\langle E^7 \rangle$	$\langle E^8 \rangle$
acentric untwinned	0.886	1.329	2	3.323	6	11.632	24
acentric perfect twinning	0.940	1.175	1.5	2.056	3	4.626	7.5
centric untwinned	0.798	1.596	3	6.383	15	38.30	105
centric perfect twinning	0.886	1.329	2	0.000	6	11.632	24

Table 1. Numerical values for the first eight moments of E. ($\langle E^2 \rangle = 1$ by definition.)

References

1. S. French & K. S. Wilson *Acta Cryst.* A**34**, 517–525 (1978).
2. A. J. C. Wilson *Acta Cryst.* **2**, 318–321 (1949).
3. M. S. Abramovitz & I. A. Stegun *Handbook of Mathematical Functions* (Dover, New York, 1965).
4. T. O. Yeates *Acta Cryst.* **44**, 142–144 (1988).