ARP/wARP 6.1 Manual

This manual covers the fundamentals of the ARP/wARP software suite and most of its areas of use. If you cannot find the answer(s) here, visit the ARP/wARP web page **http://www.arp-warp.org**.

Table of contents

Chapter 1. General information

Introduction

ARP/wARP is a package for automated protein model building and structure refinement. It is based on a unified approach to the structure solution process by combining electron density interpretation using the concept of the hybrid model, pattern recognition in an electron density map and maximum likelihood model parameter refinement. The ARP/wARP suite is under continuous development. The present release, version 6.1, can be used in the following ways:

1. Automated tracing of the density map and model building (GUI module ***ARP/wARP***, script module ***auto_warp.sh***). This includes construction of both main chain and side chain polypeptide fragments for the cases of MR solutions or for MAD and M(S)IR(AS) phases.
2. Free atoms density modification (GUI module ***ARP/wARP***).
3. Building of the solvent structure (GUI module ***ARP/wARP***).
4. Fitting side chains into an existing model (GUI module ***ARP/wARP ReBuild***).
5. Automated building of bound ligands (GUI module ***ARP/wARP LigandBuild***, script module ***auto_ligand.sh***).
4. Automated building of helical fragments (GUI module ***ARP/wARP HelixBuild***).

An overview of the ARP/wARP can be found in:

- Lamzin, V.S., Perrakis, A. & Wilson, K.S. (2001) The ARP/wARP suite for automated construction and refinement of protein models. In *International Tables for Crystallography. Volume F: Crystallography of biological macromolecules* (Rossmann, M.G. & Arnold, E. eds.), Dordrecht, Kluwer Academic Publishers, The Netherlands, pp. 720-722.

Applications are presented in:

- Perrakis, A., Morris, R. and Lamzin, V.S. (1999). Automated protein model building combined with iterative structure refinement. *Nature Struct. Biol.* **6**, 458-463.
- Perrakis, A., Harkiolaki, M., Wilson, K.S. and Lamzin, V.S. (2001) ARP/wARP and molecular replacement. *Acta Cryst*. D57, 1445-1450.
- Perrakis, A., Sixma, T.K., Wilson, K.S. and Lamzin, V.S. (1997) wARP: improvement and extension of crystallographic phases by weighted averaging of multiple refined dummy atomic models. *Acta Cryst*. **D53**, 448-455.
- Lamzin, V.S. and Wilson, K.S. (1993) Automated refinement of protein models. *Acta Cryst*. **D49**, 129-149.
- Zwart, P.H. and Lamzin, V.S. (2004) Modelling bound ligands in protein crystal structures. *Acta Cryst*. D, in press.

Algorithmic details are given in:

- Lamzin, V.S. & Wilson, K.S. (1997) Automated refinement for protein crystallography. In *Methods in Enzymology* (Carter, C.W. & Sweet, R.M. eds.) **277**, 269-305.
- Morris, R.J., Perrakis, A. & Lamzin, V.S. (2002) ARP/wARP's model-building algorithms. I. The main chain. *Acta Crystallogr*. **D58**, 968-975.

For other publications please refer to the ARP/wARP web page or the references listed hereafter.

Major changes in Version 6.1

• The standard refinement engine is now REFMAC5 from the CCP4 software suite. CCP4 5.0 is the recommended version to use with ARP/wARP 6.1.

• Improved and faster chain tracing with side chain docking applicable to NCS cases. The tracing should generally be applicable to X-ray data extending to 2.6 Å resolution or higher. Success at lower resolution is strongly dependent on the quality of initial phases. In some cases one may be lucky enough to obtain partial tracing in the resolution range from 2.6 to 2.9 Å.

• A beta version of the automated ligand building into difference map at resolution of 2.5 Å or higher. A plain PDB file with the ligand coordinates is required on input without additional specification of conformation and torsional flexibility of the ligand.

• A beta version of the tracing of helical fragments at resolution down to 3.5 Å.

• A possibility for remote submission of a chain tracing task to a 16-processor Linux cluster at EMBL Hamburg, freely available to both academic users and industrial groups that hold a valid license.

• In addition to a full graphical user interface based on the CCP4i GUI, now containing new modules, ARP/wARP 6.1 also offers one-line job submission for the tasks of autotracing protein chain fragments and construction of bound ligands.

Latest News and Bug Report

For the latest news and announcements please visit the ARP/wARP page. Users are kindly requested to report any bugs or suggested features to the authors.

Distribution

The ARP/wARP package is freely available to academic users provided that they agree to the ARP/wARP license conditions and the applications of ARP/wARP are properly cited.

Industrial users are requested to obtain a commercial license via the ARP/wARP web page.

Please consult the ARP/wARP web page for the most relevant citation that describes your application of ARP/wARP. For convenience, an ARP/wARP job launched via the GUI or one-line scripts will suggest a suitable reference for the chosen protocol - please inspect "view log files".

Chapter 2. Installing ARP/wARP

The installation of ARP/wARP is straightforward, please follow the procedure detailed below:

1. Install (or make sure you have installed) the CCP4 suite (the CCP4 distribution can be fetched from the CCP4 web page **http://www.ccp4.ac.uk/main.php**). CCP4 5.0 is the recommended version to use with ARP/wARP 6.1.

2. Download the full ARP/wARP package arp_warp_6.1.tar.gz from the ARP/wARP web page and save it in a location of your choice. Next, type:

```
% gunzip arp_warp_6.1.tar.gz
```

```
% tar xvf arp_warp_6.1.tar
```

The distribution will unpack under the directory called *arp_warp_6.1* and will contain all the required files and subdirectories. The supported operating systems include Irix/Irix64, OSF1 version 5 (OSF1 version 4 is supported for all tasks except local execution of the autotracing task), Linux-i686 and OSX. Examples include the diffraction data and sequence for Leishmanolysin (courtesy of Peter Metcalf). *ARP_wARP_CCP4I.tar.gz* includes everything necessary to run ARP/wARP from the CCP4i interface; *install.sh* is an installation script to help you set the appropriate environmental variables. *README* will walk you through the installation process.

3. Run the install.sh script by simply typing

```
% ./install.sh
```

The install.sh script will create a *bin* subdirectory where the correct executables will be copied from src in system-specific directories, and will output two lines that must be added to your *.cshrc* file to provide the proper environment for ARP/wARP.

You can have multiple operating system installation in the same directory tree. The setup lines will then recognise the machine type upon login and set up the correct binaries. To create these you have to run the `./install.sh` script once per each operating system you plan to use.

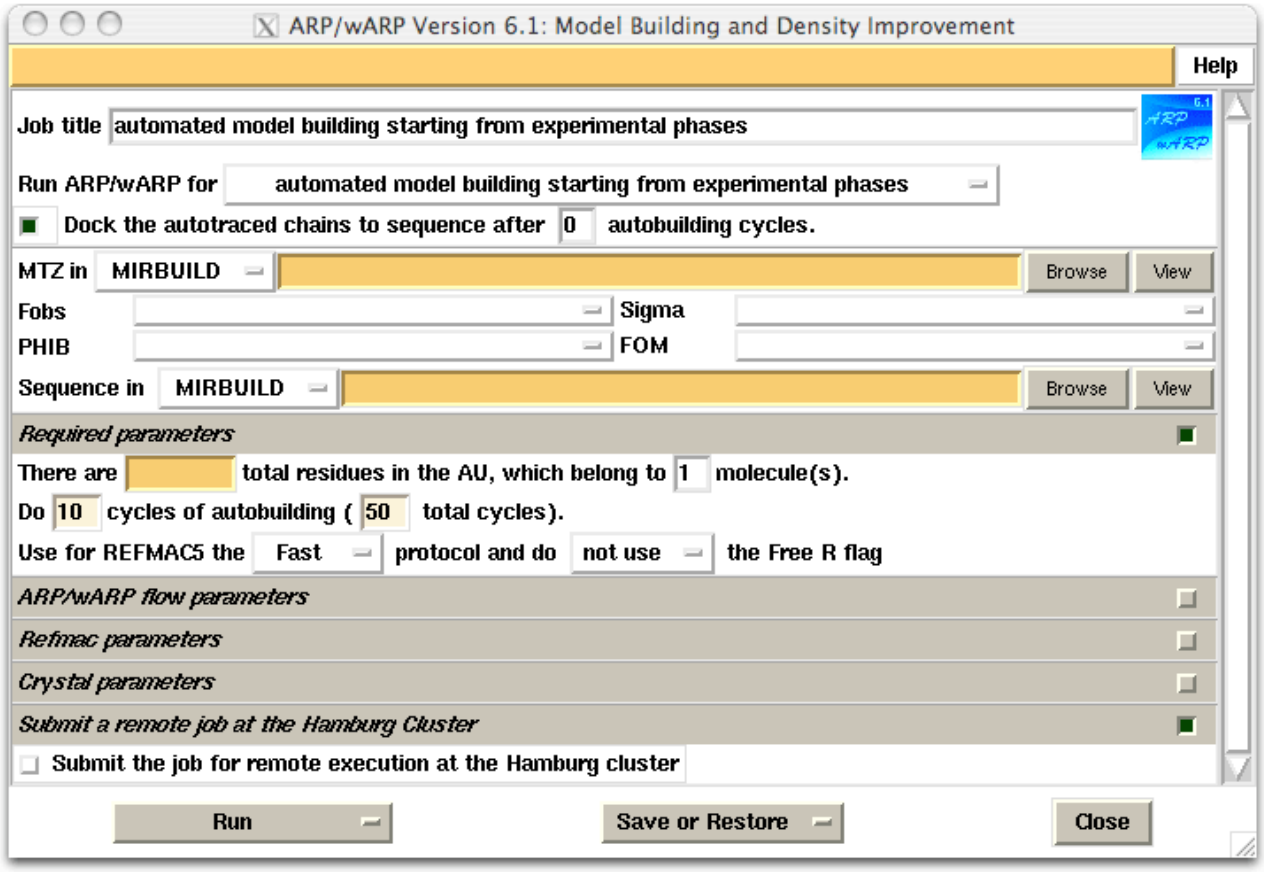
4. Start CCP4i by typing `ccp4i` and setup your project if you use this CCP4i for the first time in your user account. Within the CCP4i go to "System administration" and uninstall any earlier version of ARP/wARP (if there is any). Restart CCP4i. Go to "System administration" and install *arp_warp_6.1* by navigating to the file *arp_warp_6.1/ARP_wARP_CCP4I.tar.gz*. The interface should ungzip, untar and install arp_warp_6.1 automatically. At this point, the installation is complete. Restart CCP4i and click on the desired ARP/wARP module.

We recommend that the installation of the CCP4 GUI be done by the person who installed the CCP4 package, so that all users have an up-to-date interface and the correct permissions are set.

Unless you are already an experienced ARP/wARP user, you should try to get started with the test files provided under examples. We are working on several tutorials to help beginners use and understand the software. If things still do not work as expected please consult your system manager first. If problems remain, please contact us at the ARP/wARP bulletin board (available from the ARP/wARP web page)

Chapter 3. Using ARP/wARP

The main ARP/wARP module



This is the main module of ARP/wARP, which provides execution of the following tasks:

(a) automated model building starting from experimental phases
(b) automated model building starting from existing model
(c) improvement of maps by atoms update and refinement
(d) build solvent atoms

Applications (a) and (b) (so called *warpNtrace* protocol) start with input experimental / density modified phases or available (preliminary refined or partially autotraced) model and are aimed to deliver an essentially complete model and obviously an improved map. The software that is driven by these applications has considerably advanced from the previous 6.0 version so that the task now converges faster, may be applicable to lower resolution of the X-ray data and may tolerate poorer starting phases. As a rule of thumb, the resolution of the data should be 2.6 Å or higher.

warpNtrace protocol utilises the idea of the *hybrid* model in which *protein* and *free* atoms can co-exist. *warpNtrace* keeps whatever was recognised as protein (in a form of polypeptide fragments) and the rest as free atoms and refines this *hybrid* model during a 'big' cycle, consisting of several (typically 5) ARP/REFMAC refinement/update cycles. At the end of each ‘big’ cycle the map is

interpreted anew and this is expected to provide a better interpretation (more residues in less fragments). This whole procedure is iterated (typically 10 times).

The output of *warpNtrace* is a set of refined polypeptides fragments which, in a case of success, comprise 80 to 98 % of the whole structure. If sequence is available, the traced fragments will be docked and side chains will be built while iterative refinement. After the last building cycle the fragments will be arrange to form a globular structure (or, for a case of NCS, several NCS-related structures). The remainder of the structure (*cis*-prolines, poorly ordered loops and terminal residues for each fragment) will have to be constructed by the user. Since the output model is refined, it is fairly accurate – its accuracy being comparable to the accuracy of the finally refined structure. Mistracing (incorrect tracing of polypeptide fragments) is not impossible but should not normally exceed 1 % of the whole structure (this is very much subject to the resolution and quality of the data, quality of starting phases and the level of convergence of the *warpNtrace* task).

Application (c) has not changed since the previous 6.0 release. It can be used if *warpNtrace* was unsuccessful and may provide improvement in density map. The map is first interpreted as a pseudo protein model, consisted of non-connected free atoms (similar to the map interpretation in application (a)). This model is then refined and updated with iterative cycles of ARP/REFMAC. However, no autotracing (interpretation of the map in terms of polypeptide fragments as in *warpNtrace*) is carried out.

Application (d) for building a solvent structure into a model where the protein part is complete has also not changed since the previous 6.0 release. Within this task restrained reciprocal space refinement is carried out with REFMAC while ARP/wARP is performing automatic adjustment of the solvent structure. Resolution of the data should be 2.5 Å or higher. The output is the protein model with the solvent molecules transformed with symmetry operations to lie around the protein.

Below is the application (a) is described in detail, input to applications (b), (c) and (d) is very similar and should be obvious.

- Launch **ARP/wARP** window within the CCP4i GUI.

- Provide required input:
  - ***Run ARP/wARP for*** Choose applications (a) to (d) as described above.
  - ***Dock the autotraced chains to sequence*** The default is to dock the fragments starting from building cycle 0. The cycle number can be changed, although this should not be advantageous. Should the sequence not be available, the docking can be disabled by clicking on the check box on the left.
  - ***MTZ in*** X-ray data in the MTZ format containing structure factor amplitudes, their standard deviations, phases and figures of merit. If pre-weighted structure factor amplitudes (e.g. from SHARP) are to be used to construct initial map, please check the corresponding box in *ARP/wARP flow parameters* (see below).
  - ***Fobs Sigma PHIB FOM*** If the MTZ column labels for structure factor amplitudes, their standard deviations, phases and figures of merit have obvious names, they will be recognised automatically. Otherwise please use the scrolling button, navigate to *List All Labels* and chose appropriate ones.
  - ***Sequence file in*** Provide the sequence file in the following format (pir):
    The first line should start with ">"
    The second line should be blank
    The sequence (1 letter code) starts from the third line. The spaces hereafter are ignored.
  - ***Total residues in the AU / number of molecules*** For monomers provide the total number of residues in the asymmetric unit, the number of molecules is obviously 1. In a case of NCS, please also provide the total (!) number of residues in the asymmetric unit

and the number of NCS related molecules (e.g. if you have 2 molecules in the AU with 200 residues each, enter 400 for the number of residues). If you have a heteromer, e.g. 3α/3β structure, the NCS order is 3 but please make sure that the sequence file contains both sequences separated by about 20 alanines:
SEQUENCE_OF_α_SUBUNIT_AAAAAAAAAAAAAAAAAAAA_SEQUENCE_OF_β_SUBUNIT

- ***Cycles of autobuilding / total cycles*** The default is 10 'big' building cycles separated with 5 ARP/REFMAC cycles (thus making 50 cycles in total). In cases of good starting phases the autobuilding may converge faster, in cases of poorer phases more cycles may be required. You can always submit *warpNtrace* for further cycles using the output of the previous tracing (application *automated model building starting from existing model*).
- ***Protocol for REFMAC5 / Rfree*** The fast and slow protocols differ in the number of internal Refmac cycles and the dumping factors. The type of the protocol will be set automatically judging from the resolution of the X-ray data. Usually there is no need to change it. For warpNtrace task the default is to not use Rfree, since the number of traced residues serves as excellent indicator of the success of the job. You can turn the use of Rfree on but the authors have seen marginal cases (low resolution and hence low observation-to-parameter ratio) when this adversely affected the tracing.

- Now you are ready to start the job: Click on **Run** and choose **Run now**

There is a number of additional parameters that you normally should not worry about. Brief description is given below

- **ARP/wARP flow parameters**:
  - ***Pre-weighted Fobs for initial map calculation*** (e.g. from SHARP). Checking this box will result in a pool-down menu asking for *FBEST* label.
  - ***Number of ARP/REFMAC refinement cycles between autobuilding*** The default is 5 cycles. In cases of poor convergence you can try to increase this number to 10.
  - ***Skip the autobuilding for the first cycles*** Checking this box will disable the autotracing for the provided number of cycles. This was sometimes advantageous with previous version 6.0 when the initial phases were poor. The default is to start autotracing from cycle 0.
  - ***Randomisation of atomic positions*** This was sometimes advantageous with previous version 6.0 when the initial bias was high. The default is not to randomise.
  - ***Truncate excessive shifts*** This is a leftover from earlier version, ignore this parameter.
  - ***Removal of protein atoms of traced model*** During the ARP/REFMAC cycles in between the tracing, the hybrid model is updated. If you would like to keep track on what part of traced fragments has been removed during the update, then check the box. This option is provided primarily for developers only.
  - ***Iterate the tracing*** Each main chain tracing is carried out in several iterations. The module will decide on its own how many iterations is needed. The default maximum number is 5 and it is NOT recommended to change this value.
  - ***Density thresholds for atom removal and addition*** These parameters are defined automatically on the basis of the resolution of X-ray data. In cases of poor convergence, particularly when the number of both added and removed atoms is considerably less than the number "requested" (as can be seen from the log file), the threshold for atoms removal can be slightly increased. This option is provided primarily for developers only.

- ***Increase in the number of atoms to be added and removed as compared to the automatically set values*** The default is 1 (no increase) and it is not recommended to change this parameters. This option is provided primarily for developers only.

- **Refmac parameters**:
  - ***Cycles of refinement in each Refmac run*** Refmac is invoked to refine the hybrid model before the density maps are computed. The default is 1 cycle for the *fast* protocol and 3 cycles for the *slow* protocol, see above. There is usually no need to change these parameters.
  - ***Damp shifts*** The defaults of 0.9 for the *fast* protocol and 0.4 for the *slow* protocol. There is usually no need to change these parameters.
  - ***Matrix weight for Xray / Geometry*** The default is *automatic* weighting. This proved to work well and, probably, there is no need to change this parameter.
  - ***Scaling model*** The default is to use *bulk* solvent correction for scaling low angle part of the X-ray data. You can turn this off (chose *simple* solvent correct) if your low angle data are missing (e.g. your data have about 8 Å low resolution cutoff) or they suffer from missing overloaded reflections.
  - ***Scaling B factor*** The default is to use *anisotropic* B factor for scaling the X-ray data. You can turn this off (chose *isotropic* scaling B factor) if your data are systematically incomplete (e.g. a cone is missing in reciprocal space).
  - ***Data with free R label*** This parameter appears if the free R flag is chosen for refinement of the protein part of the model. Here you can provide a column label for the free R flag.
  - ***Use of free R reflections*** This parameter appears if the free R flag is chosen for refinement of the protein part of the model. The scaling and calculation of σ_A coefficients by Refmac map can be computed on the bases of the free reflections (this is the default) or using all reflections.
  - ***Solvent mask correction*** This is different form the low resolution bulk solvent correction. The default is not to use solvent mask correction with Refmac.
  - ***TLS refinement*** The default is not to do a TLS refinement of a hybrid model.

- **Crystal parameters**:
  - ***Space group*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Cell*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Wilson B factor*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Solvent content*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed. However, you may want to check this number whether it conforms to your expectations.
  - ***Resolution*** By default all reflections present in the MTZ file will be used. You can check the box and then narrow the range if you are aware of certain deficiencies of your data.

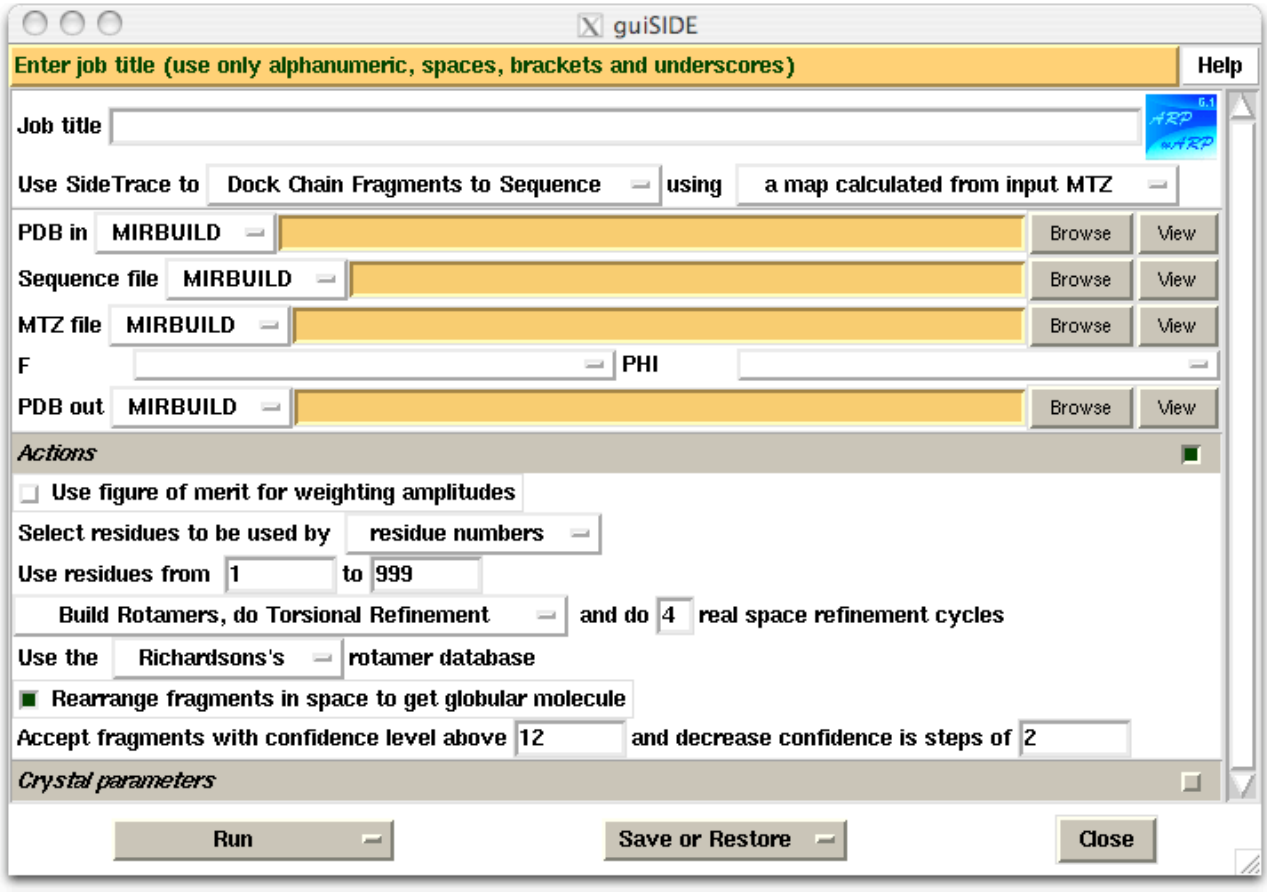
- **Submit a remote job at the Hamburg Cluster**:
  - Checking in this button will activate remote submission. This is described below in a separate chapter of this document.

- **OUTPUT files, short Log File**:
  - ***Had to go as low as XXX sigma to complete atoms search*** The initial free-atoms model is built into the starting density map. The density threshold is successively reduced. A typical value that you can see in the log file is between 0.3 and 0.6 sigma. A lower value may be an indication of too-much flattened map or an overestimation of the number of residues in the asymmetric unit. If you suspect the latter, please check the derived solvent content in the GUI window.
  - ***Building cycle zero*** Normally one should expect a considerably part of the structure to be built already at the building cycle zero. If this is not the case, observe the situation for a few building cycles. If, however, there is essentially nothing autotraced for 10 building cycles, please inspect whether the initial phases are sufficiently good.
  - ***Rounds within building cycle*** As was mentioned above, each cycle of the the main chain tracing is carried out in several rounds. Normally each successive round should result in more residues and in fewer fragments. The maximum length of the traced fragment is also printed for information.
  - ***Chains, residues and connectivity index*** The output from the best tracing round is further processed. Terminal residues are removed and the fragments of 5 peptides or shorter are converted back to free atoms. The leftover is printed and used to provide restraints for subsequent ARP/REFMAC cycles. The value of the connectivity index should increase if the tracing is successful. Its value below 0.6 is not very promising. A value around 0.8 indicates a good progress. A value above 0.95 indicates an essentially complete tracing.
  - ***Residues docked into sequence*** If the sequence was provided, the autotraced fragments are docked into it and the side chains are built and refined in real space. The results of this are printed out.
  - ***R factor from Refmac*** R factor typically oscillates. It goes up after each tracing cycle (because the model is entirely rebuilt) and then decreases during the ARP/REFMAC cycles. Overall, it should reach a value typical for a restrained refinement.
  - ***Sequence coverage*** This is defined as the ratio between the number of docked residues and the total number of traced residues. If sequence is provided, this is printed in the log file. A value higher than 0.8 is deemed as good convergence. All free (dummy) atoms are removed from the file and the task moves into a few cycles of restrained refinement with solvent search. If, however, a value of sequence coverage is lower than 0.8, the free atoms are left in the file. You can inspect the density maps, start changing the model on the graphics or, alternatively, submit another *warpNtrace* task using the output of this job.

  - ***CPU requirements*** Execution of the autotracing task is time consuming. Using a standard protocol of 10 building cycles interspaced with 5 arp/refmac cycles, one should expect a job for a structure of 200 residues to be completed within 1 hour (subject to the power of the computer you are using).

  - ***Job termination***
    The statement *Task completed successfully* indicates that the job is finished with no error. An error statement
    QUITTING ... PROGRAM TO BLAME: name_of_the_programme
    indicated that one of the modules of the task has terminated with an error message. You will also be referred to the specific log file.

The Re-Build Module



This module has not changed since Version 6.0 (former name “guiside”). It provides a construction of side chains into an existing model. This could be used for e.g. side chain “mutations” and fitting them into the density. The built side chains are reasonably accurate and the whole procedure is fast. However, the module uses the old *side_dock* algorithm and does not support NCS (unlike the new *snow* program involved for side chain docking within autotracing). This module will be superseded in subsequent releases.

- Launch **ARP/wARP Rebuild** window within the CCP4i GUI.

- Provide required input:
  - ***Use Side Trace to*** Two options are to dock side chains to sequence or to only rebuild side chains. Both follow with the real space fit.
  - ***using*** Either an MTZ or a map file can be supplied.
  - ***PDB in*** Provide the PDB file with the initial model.
  - ***Sequence file in*** Provide the sequence file in the following format (pir):
    The first line should start with “>”
    The second line should be blank
    The sequence (1 letter code) starts from the third line. The spaces hereafter are ignored.
  - ***MTZ in*** X-ray data in the MTZ format containing structure factor amplitudes and their standard deviations.
  - ***Fobs PHI*** This bar appears only if the input is MTZ (see *using* above). If the MTZ column labels for structure factor amplitudes and phases have obvious names, they will

be recognized automatically. Otherwise please use the scrolling button, navigate to *List All Labels* and chose appropriate ones.

- ***PDB out*** Provide the output PDB file name.
- ***Use figure of merit for weighting amplitudes*** Checking out this box will result in the FOM bar appearing above within the label selection of the MTZ file.
- ***Select residues to be used by*** The two options are: by the residue numbers or by the chain names. According to the selection, the next item will either be ***Use residues from to*** or ***Use chains***. Please provide required information there.
- ***Use the rotamer database*** The two options are: the Richardsons's data base and the one from ARP/wARP. The use of Richardsons's database is recommended and is a default.
- ***Rearrange fragments in space to get globular molecule*** This is a default (unclick the box to disable the option). If clicked, the fragments will be arranged using symmetry operators to form a globular molecule. This option does not support NCS.
- ***Accept fragments with confidence level above NUMBER1 and decrease confidence in steps of NUMBER2*** The defaults are 12 and 2 respectively. Decreasing the first parameter to a value lower than 12 may result in an incorrect sequence docking.

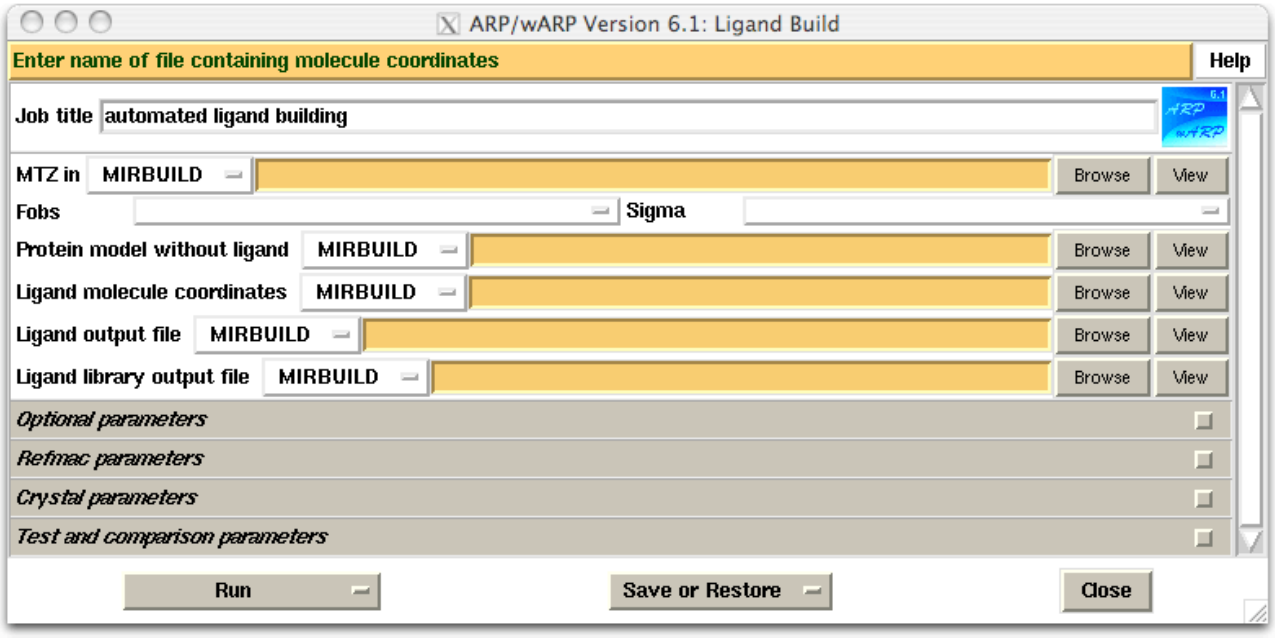
- Now you are ready to start the job: Click on **Run** and choose **Run now**

There is a number of additional parameters that you normally should not worry about. Brief description is given below

- **Crystal parameters**:
  - ***Space group*** This is derived automatically from the MTZ file, is displayed for information only and should not be changed.
  - ***Cell*** This is derived automatically from the MTZ file, is displayed for information only and should not be changed.
  - ***ARP/wARP asymmetric unit*** This is derived automatically from the space group, is displayed for information only and should not be changed.

- **OUTPUT files, Log File**:
  - The output is fairly excessive. Look for "Summary of results". Just above that the number of side chains that are docked into sequence is given. This follows by a sequence alignment.
  - ***CPU requirements*** Execution of the *Re-Build* task is fairly quick. One should expect a structure of 1,000 residues to be processed within 1 minute.
  - ***Job termination*** The statement *Task completed successfully* indicates that the job is finished with no error. An error statement
    QUITTING ... PROGRAM TO BLAME: name_of_the_programme
    indicated that one of the modules of the task has terminated with an error message. Please refer to the file *XXX_guiside.log*

Automated Ligand Building



The ligand building procedure within ARP/wARP Version 6.1 is based of the use of distance matrices (Zwart, P.H. & Lamzin, V.S. (2004) Modelling bound ligands in protein crystal structures. Acta Cryst. Section D, in the press). A difference electron density map, that supposedly contains the ligand, is parameterised by an orthogonal grid. The algorithm automatically defines the threshold in the difference density map and selects the largest cluster. The volume of the cluster is matched to the volume of the ligand looked for. The interpretation of this grid set is performed by "swapping" the ligand atom names assigned to the grid points. The stereochemical information and van der Waals repulsions in combination with the electron density allows one to obtain a suitable estimate of the position, orientation and conformation of the ligand. For moderate size ligands (10 to 50 atoms) a model is typically built with an r.m.s.d. of 0.2 - 0.6 Å from the final model.

The accuracy of building is dependent on the resolution of the X-ray data and the overall atomic displacement parameter of the ligand. The constructed models are close enough to their correct positions for REFMAC5 to straightforwardly refine the protein-ligand complex. The procedure takes from minutes to about half an hour and can be iterated to locate additional ligands, if any are present. The authors consider this module as a beta-test version and the feedback from the users would be much appreciated.

- Launch **ARP/wARP Ligandbuild** window within the CCP4i GUI.

- Provide required input:
  - ***MTZ in*** X-ray data in the MTZ format containing structure factor amplitudes and their standard deviations.
  - ***Fobs Sigma*** If the MTZ column labels for structure factor amplitudes and their standard deviations have obvious names, they will be recognized automatically. Otherwise please use the scrolling button, navigate to *List All Labels* and chose appropriate ones.

- ***Protein model without ligand*** Provide the PDB file with coordinates of the protein only. If the file contains solvent atoms, free atoms or fragments of other ligands, please make sure that their location is not overlapping with the supposed location of the ligand.
- ***Ligand molecule coordinates*** Stereochemical information about the ligand to be built is read in a form of a PDB file. This file should contain the ligand molecule only. The molecule can be in any conformation, however the interatomic distances, bonding angles and the chirality (if present) should correspond to the target stereochemistry of the ligand to be built. Please also check that there is atom-bonded connectivity throughout the whole target ligand molecule (i.e. you do not accidentally have several unconnected clusters of atoms).

- Now you are ready to start the job: Click on **Run** and choose **Run now**

There is a number of additional parameters that you normally should not worry about. Brief description is given below

- **Optional parameters**:
  - ***Refmac5*** By default the "fast" protocol is used (1 cycle of refinement with damping shifts 0.9). If your PDB file needs considerable pre-refinement with Refmac before the difference electron density map can be computed, you can chose the slow protocol (3 cycles of refinement with damping shifts 0.4).
  - ***Free R Flag*** The default is not to use "R-free" for ligand building. You can chose to use R-free, this will cause additional optional parameters to appear within the section "Refmac parameters".
  - ***Grid spacing for ligand finding map*** The default is 0.5 Å. This value may be changed automatically by the software if it becomes short of memory for very large protein structures. There is no need and it is *not recommended* to change this parameter.
  - ***Grid sparse*** The default is 1.3 Å. This value will further be adjusted by the software if needed. Generally, there should be no need to change this parameter. However if the ligand building is unsuccessful, and you like to set this value and to disable its automatic adjustment, enter the negative number, e.g. –1.3 (do not set this value to higher than 1.3 and lower than 1.0 !).
  - ***Excess volume*** The default is 1.3 times the expected volume of the ligand. There should be no need to change this parameter. However, if the ligand building is unsuccessful, you can try setting it to 1.2 or 1.4.

- **Refmac parameters**:
  - ***Cycles of refinement in each Refmac run*** Refmac is invoked to refine your protein part of the structure before the difference density map is computed. The default is 1 cycle for the "fast" protocol" and 3 cycles for the slow protocol, see above. Since the protein part of the model is expected to be well refined, there is no need for a large number of cycles.
  - ***Damp shifts*** The defaults of 0.9 for the "fast" protocol" and 0.4 for the slow protocol. There is no need to change these parameters.
  - ***Matrix weight for Xray / Geometry*** The default is *automatic* weighting. Since the aim of the ligand building module is not to produce a well-geometrised protein structure, there is no need to change this parameter.
  - ***Scaling model*** The default is to use *bulk* solvent correction for scaling low angle part of the X-ray data. You can turn this off (chose *simple* solvent correct) if your low angle

data are missing (e.g. your data have about 8 Å low resolution cutoff) or they suffer from missing overloaded reflections.

- ***Scaling B factor*** The default is to use *anisotropic* B factor for scaling the X-ray data. You can turn this off (chose *isotropic* scaling B factor) if your data are systematically incomplete (e.g. a cone is missing in reciprocal space).
- ***Data with free R label*** This parameter appears if the free R flag is chosen for refinement of the protein part of the model. Here you can provide a column label for the free R flag.
- ***Use of free R reflections*** This parameter appears if the free R flag is chosen for refinement of the protein part of the model. The scaling and calculation of σ_A coefficients by Refmac map can be computed on the bases of the free reflections (this is the default) or using all reflections.
- ***Solvent mask correction*** This is different form the low resolution bulk solvent correction. The default is not to use solvent mask correction with Refmac.
- ***Input a user-defined library file*** If you already have a Refmac-style cif library for your ligand, you can input it here. Otherwise, Refmac will use its own library if it knows the ligand. If it does not, the job will stop with an error message, see below how to get around it.

- **Crystal parameters**:
  - ***Space group*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Cell*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Wilson B factor*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Solvent content*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed. However, you may want to check this number whether it conforms to your expectations.
  - ***Resolution*** By default all reflections present in the MTZ file will be used. You can check the box and then narrow the range if you are aware of certain deficiencies of your data.

- **Test and comparison parameters**:
  - ***Compare with an already fitted ligand*** If you have the final model of the ligand in the correct orientation and would like to check the installation and the performance of the software, you can check this box. You will then have to provide a PDB file that will be used for comparison.

- **OUTPUT files, short Log File**:
  - ***Ligand library*** If the ligand description is already present in the Refmac ligand library, it will be used and the appropriate message will be printed. If the description is missing (or differs), it will be generated with Refmac but not used for ligand construction. If your protein already has this type of ligand in, and it is not known to Refmac, the job will stop with an error message. Please provide Refmac with the ligand library description that it has generated and re-run the job. Please note that this automatically generated library description should be inspected if you intend to use it for final restraint refinement of your protein-ligand complex.
  - ***Refinement with refmac*** The R factor (and R free if requested) are printed after refinement of the protein part only with Refmac. Check that the value of the R factor is

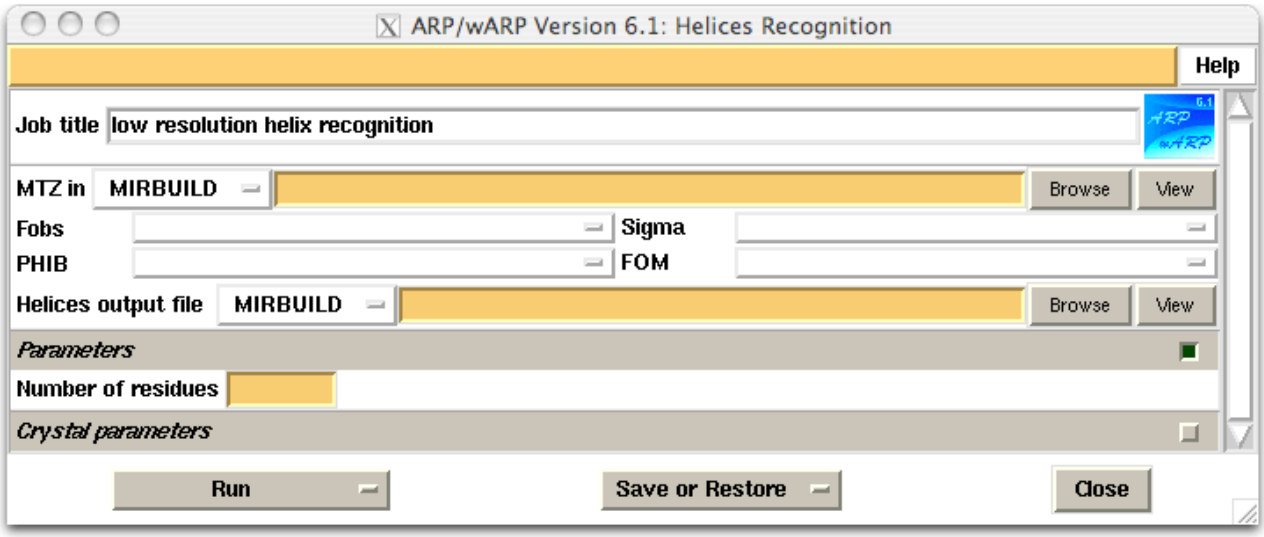
reasonable. A value of higher than about 30 % may indicate that something is wrong and that the computed difference map may be too noisy for location of the ligand.

- ***The ligandbuild program*** The mapping of the difference density synthesis parameterised with grid points onto the ligand atoms is the main routine of this ARP/wARP module. In an unfortunate case it may stop with an error message *run out of options*. This can indicate that (a) either the difference density map does not contain the ligand as the most pronounced feature and it was therefore trying to build the ligand into the noise – please inspect the map (!) or (b) that the difference density was incorrectly parameterised with the grid points. In the case of the latter you can try to re-run the job with a slightly reduced value of *excess volume* and/or *grid sparsing*, see above.
- ***Ligand characteristics and CPU requirements*** The main characteristics of the ligand molecule looked for are printed. Check that the number of atoms in the ligand model is correct. A typical value of the density threshold is between 1.5 and 2.5 sigmas above the mean. The number of grid points to be used for construction of the ligand model is defined automatically and is typically about 1.5 to 3.0 times higher than the number of ligand atoms. This number defines the CPU requirements for the ligand construction. The table below serves as a rough guide on the expected CPU (subject to your machine architecture):

Number of atoms in the ligand	CPU
Less than 15	About a minute
20	A few minutes
30	About 5 minutes
40	More than 15 minutes

- ***Real space fit*** The top 100 constructed ligand models are geometrised and fit into the difference density map. The best solution is output. If the *test and comparison* option is selected, the r.m.s.d to the reference PDB file (XYZREF) is printed. Typically this value is around 0.2 - 0.6 Å. The accuracy of building is dependent on the resolution of the X-ray data and the overall atomic displacement parameter of the ligand. There will be a warning given if the stereochemistry of the constructed ligand is poor. Also a warning will be given if the constructed ligand molecule has severe sterical clashes. This may be a sign of an incorrect ligand building. You may want to inspect the ligand and the density and, if there is a clear part of the ligand that is disordered, try to remove it from the ligand target PDB file and to re-run the job.
- ***Job termination*** The statement *Task completed successfully* indicates that the job is finished with no error. An error statement
  QUITTING ... PROGRAM TO BLAME: name_of_the_programme
  indicated that one of the modules of the task has terminated with an error message. Please refer to the file *XXX_warp_ligand_details.log*

Automated Construction of Helical Fragments



The procedure for building helical fragments within ARP/wARP Version 6.1 is based on the use of distance matrices and discriminant analysis (Kirillova, O.V. & Lamzin, V.S. (2004) Application of discriminant analysis for recognition of helical structural motifs in electron density maps. Acta Cryst. Section D, in the press). An electron density map is first analysed and a suitable threshold is selected. The next module uses stereochemical information about the helix geometry and produces a set of overlapping helical fragments. These fragments are then geometrised and the chain direction is chosen on the basis of their fit to the density. Finally the fragments are refined in real space and overlapping ones are removed. The terminal CA atoms are removed as well and the fragments are arranged so that they comprise a compact globule.

The accuracy of helix building is dependent on many parameters. It should be able to build helices at resolution as low as 4.5 Å. However, it may not result in complete helical structure and it may also contain parts that are mis-interpreted. The authors consider this module as a beta-test version and the feedback from the users would be much appreciated. The procedure is relatively fast and takes a few minutes.

The helix recognition module can be used if the resolution of data is lower than about 2.6 Å or if straight warpNtrace protocol has not been successful.

- Launch **ARP/wARP Helixbuild** window within the CCP4i GUI.

- Provide required input:
  - ***MTZ in*** X-ray data in the MTZ format containing structure factor amplitudes and their standard deviations
  - ***Fobs Sigma Phib FOM*** If the MTZ column labels for structure factor amplitudes, their standard deviations, phases and figures of merit have obvious names, they will be recognized automatically. Otherwise please use the scrolling button, navigate to *List All Labels* and chose appropriate ones.
  - ***Helices output file*** Provide the PDB file name where the constructed helical fragments will be output.
  - ***Parameters (Number of residues)*** Please provide the expected number of residues in the asymmetric unit.

- Now you are ready to start the job: Click on **Run** and choose **Run now**

There is a number of additional parameters that you normally should not worry about. Brief description is given below

- **Crystal parameters**:
  - ***Space group*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Cell*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Wilson B factor*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed.
  - ***Solvent content*** This is derived automatically from the MTZ file, is displayed for information only and cannot be changed. However, you may want to check this number whether it conforms to your expectations.
  - ***Resolution*** By default all reflections present in the MTZ file will be used. You can check the box and then narrow the range if you are aware of certain deficiencies of your data.

- **OUTPUT files, short Log File**:
  - ***Density thresholds*** The lowest density threshold is typically around 1.0 sigma. It may be lower for lower resolution or poorer data. The second number should be higher than the first one.
  - ***Helix recognition*** The helical fragments are successively filtered out, see above. Therefore, the important are the numbers at the end of the short log file, indicating the number of residues and the number of fragments into which these residues are arranged.
  - ***Further extension of the model*** If your structure is alpha-helical, you may try to feed the output of the helix recognition module into warpNtrace. However, subject to the resolution of the data, this may not provide enough seed for subsequent automatic tracing of the full chain.
  - ***CPU requirements*** Execution of the *HelixBuild* task is relatively fast. One should expect a structure of 200 residues to be processed within 1 to 2 minutes.
  - ***Job termination*** The statement *Task completed successfully* indicates that the job is finished with no error. An error statement
    QUITTING ... PROGRAM TO BLAME: name_of_the_programme
    indicated that one of the modules of the task has terminated with an error message. Please refer to the file *XXX_warp_helices_details.log*

One-line shell script for auto-tracing

The script file ***auto_warp.sh*** in the ***$warpbin*** directory allows to run the automated model building as a single-line command without the use of the GUI.

The use of ***auto_warp.sh*** is fairly simple as it involves only the call to the script itself and the specification of files related to the task together with keywords. If invoked without arguments the script will print help information.

Required keywords are the following: *workdir* (followed by the absolute path to the working directory), *datafile* (followed by the mtz-file name with the absolute path) and *nresidues* (followed by the number of residues).

Optional keywords include: *seqin* (followed by a sequence-file name with the absolute path), *modelin* (followed by a starting pdb-file with the absolute path), *fp* (followed by the fp label), *sigfp* (followed by the sigfp label), *phibest* (followed by the best phi label), *fom* (followed by the figure of merit label), *bigcycles* (followed by the number of building cycles) and *arpcycles* (followed by the number of arp refinement cycles between the building blocks).

Example call with the test psp data (assumed to be started from workdir where test data should reside):

```
$warpbin/auto_warp.sh workdir $PWD datafile $PWD/psp.mtz \
nresidues 475 seqin $PWD/psp.pir
```

The script will then create a directory in the *workdir* whose name will be printed and where a parameter file (similar to the case of a GUI-driven job) will be created. Finally, the job will invoke the same script (***arp_warp.sh***) as when run through the GUI. The whole job will run on the background.

In the directory created by ***auto_warp.sh*** in the *workdir*, the usual log files and additional output files as well as the building results can be found.

One-line shell script for ligand building

The script file ***auto_ligand.sh*** in the ***$warpbin*** directory allows to run the ligand building as a single-line command without the use of the GUI.

The use of ***auto_ligand.sh*** is fairly simple as it involves only the call to the script itself and the specification of files related to the task together with keywords. If invoked without arguments the script will print help information.

Required keywords are the following: *workdir* (followed by the absolute path to the working directory), *datafile* (followed by the mtz-file name with the absolute path), *protein* (followed by the pdb-file name of the protein model without the ligand with the absolute path) and *ligand* (followed by the pdb-file containing the ligand description with the absolute path)

Optional keywords include: *fp* (followed by the fp label), *sigfp* (followed by the sigfp label) and *validpdb* (followed by a pdb-file with the (hand- or pre-)fitted ligand with the absolute path)

Example call with test data (assumed to be started from workdir where test data should reside):

```
$warpbin/auto_ligand.sh workdir $PWD datafile $PWD/1cbs.mtz \
protein $PWD/1CBS_noligand.pdb ligand $PWD/RETINOICACID.pdb
```

The script will then create a directory in the *workdir* whose name will be printed and where a parameter file (similar to the case of a GUI-driven job) will be created. Finally, the job will invoke the same script (***warp_ligand.sh***) as when run through the GUI. The whole job will run on the background.

In the directory created by ***auto_ligand.sh*** in the *workdir*, the usual log files and additional output files as well as the building results can be found.

Remote submission of the auto-tracing task

This option offers you the following possibilities:

a) Your task will run using external computational facilities, which CPU performance may be superior to your local installation

b) You may be assured that the most recent working executables will be used should you have a problem with your local installation

c) Should the task crash, an automatic notification will be forwarded to the ARP/wARP developers who can then promptly help you

d) You can share the results of the completed task with other software developers

Clicking on the button with "Submit the job for remote execution at the Hamburg cluster" within the main ARP/wARP GUI panel allows one to execute an autotracing task remotely. The panel will expands and ask for an email address to be provided. Then choose from one of the options from the drop down menu to indicate how you would like your data to be handled. The options are:

a) *the data must be kept confidential and deleted after the job has finished*
b) *the data can be made available to ARP/wARP developers*
c) *the data can be archived and made available to SPINE and BIOXHIT partners*
d) *the data can be archived and made available to any software developer that requests them*

Needless to say, that the users will make an important contribution towards future software development if they decide to share their data and results of the autotracing job. Option (b) will only allow the data share to the ARP/wARP development team. Option (c) will extend the data share to the partners of the EC FW 5 and FW 6 integrated projects, SPINE and BIOXHIT. Option (d) will further extend the share to any software developer world-wide.

Once the job has submitted for remote execution (but not yet launched !), the GUI window will indicate that the job has finished. Please inspect the log file from the drop down menu option "View files from job" for further instructions. An email will be sent to you at the email address that you entered in the GUI window. Please follow the instructions in the email (http link, login and password) to actually launch the job at the Hamburg cluster. You can then monitor the log file through your browser window. As soon as the job is finished, you will be provided with an automatic link to the results that you can then download.

Keep in mind that once the job is finished, your data will be kept for only a week. Make sure that you download your data within that time.

The remote job submission relies on the ***curl*** software installed at your site. Availability of ***curl*** is checked while installing ARP/wARP and a warning (and http link) are given if ***curl*** is not available.

Chapter 4. Additional Remarks

Quality of the X-ray Data

The X-ray data should be complete as possible, especially in the low resolution range (5 Å and lower). Ideally the X-ray data should have no low resolution cutoff. If the low resolution strong data are systematically incomplete (e.g. missing or overloaded reflections), the density map, even in the case of a good model, may be discontinuous and inconsistent with the model. Because ARP/wARP involves updating on the basis of density maps, such discontinuity can lead partially to slow convergence or even non-interpretable maps.

ARP/wARP automatically checks the fit of the your data to the expected Wilson plot and will report if necessary. If suggested to cut the data from the high resolution side - follow the suggestion. Ensure that the highest resolution is at least 2.6 Å. If suggested to cut the data from the low resolution side - do so but do not cut to a resolution below 8 or 10 Å. If suggested to ignore all data or there are still other complaints after the cut - go and recollect/reprocess your data. You may, of course, choose to ignore the ARP/wARP warning messages and submit the refinement job but then don't expect much.

A common misconception is that you have to have experimental *phases* to 2.6 Å resolution. Though it would be advantageous, it is not at all a requirement. You have to have *native data* to high resolution, do a quick phase extension to around 2.6-3.0 Å by solvent flattening, and then go on with ARP/wARP. All the above references to the highest resolution refer to the case with 50 % solvent content.

Limitations

The ARP/wARP procedure requires the use of some of the CCP4 programs and particularly REFMAC5. Therefore, these must be installed before ARP/wARP can be launched.

ARP/wARP itself is limited to:

1. The CCP4 conventions should be set up before running ARP/wARP.
2. Density maps and reflection MTZ files in CCP4 format.
3. Coordinate files in standard PDB format.
4. Only acentric space groups (typical for protein crystals) and *P*1 bar are supported.

Chapter 5. Author-Abuse Information and Acknowledgements

The authors to abuse of...

The Hamburg team (European Molecular Biology Laboratory (EMBL) Hamburg Outstation, c/o DESY, Notkestrasse 85, 22603 Hamburg, Germany):

Victor S Lamzin (tel +49-40-89902-121, fax +49-40-89902-149, email: victor@embl-hamburg.de)
Gerrit G. Langer
Parthasarathy Venkat
Francisco Fernandez
Tilo Strutz
Guillaume Evrard

The Amsterdam team (Molecular Carcinogenesis Programme, Netherlands Cancer Institute, Plesmanlaan 121, 1066 CX Amsterdam, The Netherlands):

Anastassis Perrakis (tel. +31-20-512-1951, fax +31-20-512-1954, email: a.perrakis@nki.nl)
Serge Cohen
Marouane Ben Jelloul
Krista Joosten

Former members

Richard J. Morris
Petrus H. Zwart
Olga V. Kirillova
Matheos Kakaris

The authors are especially grateful to:

• Keith S Wilson (York, UK) one of the originators of the software.

• Zbyszek Dauter (Brookhaven, USA) for significant contributions at earlier stages the software development.

• The CCP4 developers, particularly to Garib Murshudov, Liz Potterton and Eleanor Dodson (York, UK) and Peter Briggs (Daresbury, UK)

• Many of our collaborators and active users: Jozef Sevcik (Bratislava, SLO), Phil Evans (Cambridge, UK), Titia Sixma (Amsterdam, The Netherlands), Erik van Asselt (Groningen, The Netherlands), Clemens Vonrhein (Freiburg, Germany), Santosh Panjikar and Andrea Schmidt (Hamburg, Germany) and Rob Meijers (Harvard, USA) and many, many others.

• We would also like to take this opportunity to thank the support of ARP/wARP: The NIH and the EU commission for research and infrastructure grants; the EMBL and the NKI, for hosting the research groups; our industrial users, for generating a license income which strengthens our ability to keep to our commitment for free distribution to the academic community.