

# **DOCK 5.2 User Manual**

Irwin D. Kuntz  
Demetri T. Moustakas  
Paula Therese Lang

Copyright University of California 2005

Last updated March 2005

## **Introduction to DOCK 5.2**

We are proud to announce the release of DOCK version 5.2. More information about this and future release can be found online at:

<http://dock.compbio.ucsf.edu/>

We will be periodically adding more information to the manual. Please check back for more details.

## **Installing DOCK**

*NOTE: DOCK comes with platform specific compiled binaries. You should not need to compile the code or accessories unless you have made changes to the source code, or are planning to run DOCK on a platform for which we do not distribute binaries. If you do need to compile the files, there are makefiles included in the appropriate folders.*

1. Save file for appropriate operating system to hard drive.
2. Uncompress the archive into a folder called dock5/ in a directory of your choice.
  - a. For windows systems, a Zip file is provided
  - b. For \*nix systems, a gzipped archive is provided
3. All DOCK 5 binaries are installed in dock5/bin/

The dock5 directory contains the following subdirectories:

- bin/
- demo/
- docs/
- mpich/
- parameters/
- src/
- utilities/
  - accessories/
  - grid/
  - GBSA\_Grids/

## **Compiling DOCK (if required)**

If the binaries provided work on your system, and you have no desire to recompile the program, feel free to skip to the rest of this section. Otherwise we'll assume you have either a good spirit of adventure, or the need to compile DOCK 5 on a system other than the ones listed above. In the event the latter is

the case, please feel free to contact the DOCK team regarding compilation problems/successes on different platforms.

DOCK 5 is built upon one external library. The MPICH library is provided freely by Argonne National Labs (<http://www-unix.mcs.anl.gov/mpi/mpich/>). The MPI library must be built in order to compile DOCK 5, however it only needs to be installed and running on the system if the MPI features are to be used. Otherwise, if DOCK will be run as a single processor program, MPICH can be installed locally into the dock5/ directory.

*Building MPICH: (on SGI platforms)*

```
From the dock5 directory:  
cd mpich/  
./configure --with-arch=IRIXN32  
make
```

*Building MPICH: (on Linux platforms)*

```
From the dock5 directory:  
cd mpich/  
./configure  
make
```

*Building DOCK 5: (all platforms)*

```
From the dock5 directory:  
cd src/  
make clean  
make dock  
make install
```

*Building DOCK utilities (all platforms):*

```
From the dock5 directory:  
cd utilities/(desired utility file)  
make clean  
make all  
make install
```

NOTE 1: The install command will move the binary into the bin/ sub-directory.

NOTE 2: The default for compiling DOCK 5 is with no MPI. If you want to compile with the MPI function, you must set the "USE\_MPI" flag in "utils.h" to true before building.

## **Running DOCK**

DOCK 5 can be run from any of the standard unix shells (csh, tcsh, bash, etc...). On windows, the DOCK 5 binary can be run from a standard DOS command line

prompt. The DOCK accessory programs, however, will require that the Cygwin environment ([www.cygwin.com](http://www.cygwin.com)) be installed and used as the shell.

DOCK 5 reads a parameter file containing field/value pairs. The program is run as follows:

```
dock5/bin/dock5 -i parameter.in [-v1] [-v2] [-o outputfile.txt]
```

If the parameter file exists, any parameter values found will be read, and any required but not found will be queried to the user via stdin/stdout. An important note regarding MPI use is that the stdin/stdout interfaces are disabled across MPI, therefore the parameter file must be complete in order to work properly. It is advisable to test the parameter file on a single processor job prior to launching an MPI job. If an MPI job is launched with missing parameters, the job will terminate.

DOCK 5 outputs the job parameters to the screen at the start of the job, and prints summary information for each molecule processed. Additional summary information will be included in future releases. The `-v1` flag prints a histogram of sphere matching information. The `-v2` flag prints details about the breakdown of the GB/SA terms.

### **DOCK 5 Parameters**

The DOCK 5 parameter parser requires that the values entered for a parameter exactly match one of the legal values if any legal values are specified. For example:

```
param_a      [5] ():  
param_b      [5] (0 5 10):
```

param\_a can be assigned any value, however param\_b can only be assigned 0, 5, or 10. If no value is entered, both will default to a value of 5. Below are listed all DOCK 5 parameters, their default values, legal values, and a brief description of each. The parameters are listed in order of function. Also, for questions requiring a yes/no answer, please use the full word (yes or no) as opposed to y or n.

#### *Ligand I/O Parameters*

Parameter Name	Default	Values	Description
ligand_atom_file	database.mol2		The ligand input filename
ligand_outfile_prefix	output		The prefix that all output files will use
limit_max_ligands	no	yes, no	The maximum # of ligands that will be read in from a library
write_orientations	no	yes, no	Flag to write orientations

write_conformations	no	yes, no	Flag to write conformations
initial_skip	0		The # of molecules to skip over at the beginning of a library
calculate_rmsd	no	yes, no	Flag to perform an RMSD calculation between the final molecule pose and its initial structure.
use_rmsd_reference_mol	no	yes, no	Specify reference structure for RMSD calculation (default is starting structure)
rmsd_reference_filename	ligand_rmsd.mol2		File containing RMSD reference structure
rank_ligands	no	yes, no	Flag to enable a ligand top-score list. These ligands will be written to outfile_ranked.mol2, and outfile_scored.mol2 will be empty by default
max_ranked_ligands	500		The # of ligands to be stored in the top score list
scored_mol_output_override	no	yes, no	This flag causes all ligands to be written to outfile_scored.mol2, even when rank_ligands is true
num_scored_poses_written	1		The # of scored poses for each ligand printed to output_scored.mol2

### *Orient Ligand Parameters*

<b>Parameter Name</b>	<b>Default</b>	<b>Values</b>	<b>Description</b>
orient_ligand	yes	yes, no	Flag to orient ligand to spheres
automated_matching	yes	yes, no	Flag to perform automated matching instead of manual matching
distance_tolerance	0.25		The distance tolerance applied to each edge in a clique
distance_minimum	2.0		The minimum size for an edge in a clique
nodes_minimum	3		The minimum # of nodes in a clique
nodes_maximum	10		The maximum # of nodes in a clique
receptor_site_file	receptor.sph		The file containing the receptor spheres
max_orientations	500		The maximum # of orientations that will be cycled through
critical_points	no	yes, no	Flag to use critical point sphere labeling to target orientations to particular spheres
chemical_matching	no	yes, no	Flag to use chemical "coloring" of spheres to match chemical

			labels on ligand atoms
chem._match_tbl	chem.._match.tbl		File defining the legal chemical type matches/pairings
use_ligand_spheres	no	yes, no	Flag to enable a sphere file representing ligand heavy atoms to be used to orient the ligand. Typically used for macromolecular docking
ligand_sphere_file	ligand.sph		Ligand spheres

### *Flexible Ligand Parameters*

<b>Parameter Name</b>	<b>Default Value</b>	<b>Legal Values</b>	<b>Description</b>
flexible_ligand	yes	yes, no	Flag to perform ligand conformational searching
ag_conf_search	yes	yes, no	Flag to use the anchor & grow algorithm to search ligand conformations
min_anchor_size	50		The minimum # of heavy atoms for an anchor segment
num_anchor_poses	50		The maximum number of anchor orientations promoted to the conformational search
number_confs_per_cycle	50		The maximum number of conformations carried forward in the anchor & grow search
use_internal_energy	yes	yes, no	Flag to add an internal energy term to the score during the conformational search
internal_energy_att_exp	6		VDW attractive exponent
internal_energy_rep_exp	12		VDW repulsive exponent
internal_energy_dielectric	4.0		Dielectric used for electrostatic calculation
use_clash_overlap	no	yes no	Flag to check for overlapping atom volumes during anchor and grow
clash_overlap	0.5		Percent of overlap allowed before a clash is declared

### *Ligand Scoring Parameters*

<b>Parameter Name</b>	<b>Default Value</b>	<b>Legal Values</b>	<b>Description</b>
bump_filter	yes	yes, no	Flag to perform bump filtering
bump_grid_prefix	grid		The prefix to the grid file(s) containing the desired bump grid
max_bumps	2		The maximum allowed # of bumps for a molecule to pass

			the filter
score_molecules	yes	yes, no	Enables scoring of molecules
energy_score_primary	yes	yes, no	Flag to perform energy scoring as the primary scoring function
energy_score_secondary	yes	yes, no	Flag to perform energy scoring as the secondary scoring function
vdw_scale	1		Scalar multiplier of the vdw energy component
es_scale	1		Scalar multiplier of the electrostatic energy component
nrg_grid_prefix	grid		The prefix to the grid files containing the desired nrg grid
continuous_score_primary	no	yes, no	Flag to perform continuous non-grid based scoring
continuous_score_secondary	no	yes, no	Flag to perform continuous non-grid based scoring
cont_nrg_score_rec_filename	Receptor.mol2		File that contains receptor coordinates
cont_nrg_score_att_exp	6		VDW L-J potential attractive exponent
cont_nrg_score_rep_exp	12		VDW L-J potential repulsive exponent
cont_nrg_score_dielectric	4.0		Dielectric constant for electrostatic term
cont_nrg_score_vdw_scale	1		Scalar multiplier of vdw energy component
cont_nrg_score_es_scale	1		Scalar multiplier of electrostatic energy component
contact_score_primary	no	yes, no	Flag to perform contact scoring as the primary scoring function
contact_score_secondary	no	yes, no	Flag to perform contact scoring as the secondary scoring function
contact_cutoff_distance	4.5		The distance threshold defining a contact
contact_clash_overlap	0.75		Contact definition for use with intramolecular scoring
contact_clash_penalty	50		The penalty for each contact overlap made
cnt_grid_prefix	grid		The prefix to the grid files containing the desired cnt grid
gbsa_score_primary	no	yes, no	Toggles whether or not to use GB/SA scoring as the primary scoring function
gbsa_score_secondary	no	yes, no	Toggles whether or not to use GB/SA scoring as the secondary scoring function
gb_grid_prefix	gb_grid		The path to the pairwise GB grids
sa_grid_prefix	sa_grid		The path to the SA grids
screen_file	screen.in		GB parameter file for electrostatic screening. Its

			located in the parameters dir by default
solvent_dielectric	78.300003		The value for the solvent dielectric
vdw_grid_prefix	grid		The path to the dock4 nrg grids, used for the vdw portion of the GB/SA calculation

### *Score Optimization Parameters*

<b>Parameter Name</b>	<b>Default</b>	<b>Values</b>	<b>Description</b>
minimize_ligand	yes	yes, no	Flag to perform score optimization
minimize_anchor	yes	yes, no	Flag to perform rigid optimization of the anchor
minimize_flexible_growth	yes	yes, no	Flag to perform complete optimization during conformational search
minimize_final_pose	yes	yes, no	Flag to perform minimization of the final ligand pose
use_advanced_simplex_parameters	no	yes, no	Flag to use a simplified set of common minimization parameters for each of the minimization steps listed above

### *Basic Simplex Minimizer Parameters*

<b>Parameter Name</b>	<b>Default</b>	<b>Values</b>	<b>Description</b>
simplex_max_cycles	1		Maximum # of minimization cycles
simplex_score_converge	0.1		Exit cycle at when energy converges at cutoff
simplex_cycle_converge	1.0		Exit minimization when cycles converge at cutoff
simplex_trans_step	1.0		Initial translation step size
simplex_rot_step	0.1		Initial rotation step size
simplex_tors_step	10.0		Initial torsion angle step size
simplex_anchor_min_max_iterations	50		Maximum # of iterations per cycle per anchor
simplex_flex_min_max_iterations	10		Maximum # of iterations per cycle per growth step
simplex_final_min_max_iterations	100		Maximum # of iterations per cycle for entire molecule
simplex_random_seed	0		Seed for random number generator

### *Advanced Simplex Minimizer Parameters*

<b>Parameter Name</b>	<b>Default</b>	<b>Values</b>	<b>Description</b>
-----------------------	----------------	---------------	--------------------

simplex_anchor_min_max_iterations	50		Maximum # of minimization cycles
simplex_anchor_min_max_cycles	1		Maximum # of minimization cycles
simplex_anchor_min_score_converge	0.1		Exit cycle at when energy converges at cutoff
simplex_anchor_min_cycle_converge	1.0		Exit minimization when cycles converge at cutoff
simplex_anchor_min_trans_step	1.0		Initial translation step size
simplex_anchor_min_rot_step	0.1		Initial rotation step size
simplex_anchor_min_tors_step	10.0		Initial torsion angle step size
<i>NOTE: Repeat for minimization of each layer (simplex_flex_xxx) and final minimization (simplex_final_xxx)</i>			
simplex_random_seed	0		Seed for random number generator

### *Atom & Bond Typing Parameters*

<b>Parameter Name</b>	<b>Default</b>	<b>Legal</b>	<b>Description</b>
atom_model	all	all, united	Choice of all atom or united atom models
vdw_defn_file	vdw.defn		File containing vdw parameters for atom types
flex_defn_file	flex.defn		File containing bond definition parameters
flex_drive_file	flex_drive.tbl		File containing conformational search parameters
chem._defn_file	chem.defn		File containing chemical label (pharmacophore) definitions