Chapter 12

Modeling Protein–Protein Complexes Using the HADDOCK Webserver "Modeling Protein Complexes with HADDOCK"

Gydo C.P. van Zundert and Alexandre M.J.J. Bonvin

Abstract

Protein–protein interactions lie at the heart of most cellular processes. Determining their high-resolution structures by experimental methods is a nontrivial task, which is why complementary computational approaches have been developed over the years. To gain structural and dynamical insight on an atomic scale in these interactions, computational modeling must often be complemented by low-resolution experimental information. For this purpose, we developed the user-friendly HADDOCK webserver, the interface to our biomolecular docking program, which can make use of a variety of low-resolution data to drive the docking process. In this chapter, we explain the use of the HADDOCK webserver based on the real-life Lys48-linked di-ubiquitin case, which led to the 2BGF PDB model. We demonstrate the use of chemical shift perturbation data in combination with residual dipolar couplings and further highlight a few other cases where our software was successfully used. The HADDOCK webserver is available to the science community for free at haddock.science.uu.nl/services/HADDOCK.

Key words Docking, Protein–protein interactions, Biomolecular complexes, NMR, Ubiquitinitation

1 Introduction

Protein–protein interactions are at the basis of cellular function. In order to understand and manipulate the cell and its processes, insight into biomolecular interactions at an atomic scale is required. Major genomic and proteomic initiatives are working toward this goal. However, while the size of the human proteome is predicted to be in the order of 20,000, the interactome, the network of all interacting proteins, is estimated to be more around 650,000 [1], with additional levels of complexity linked to the dynamics of the assemblies and the localization and time of expression of their components in the cell. To make things worse, crystallizing protein complexes has proven to be substantially more difficult than single chains; studying them by NMR is a nontrivial undertaking. To close the gap between the number of interactions and structural knowledge about them, computational approaches complementary to

Daisuke Kihara (ed.), *Protein Structure Prediction*, Methods in Molecular Biology, vol. 1137,
DOI 10.1007/978-1-4939-0366-5_12,

experimental methods have been devised. One of these is protein–protein docking which aims to predict the structure of a complex starting from atomic models of the unbound subunits.

Within the plethora of docking software available, one can distinguish between several classes. Most docking programs try to sample thoroughly the interaction space by generating conformations using computational methods such as correlation techniques (often FFT-based) or geometrical hashing, to name only a few. Their scoring functions are primarily based on shape and electrostatic complementarity [2]. Our docking software HADDOCK (*H*igh *A*mbiguity-*D*riven *DOCK*ing) [3, 4] takes a somewhat unique approach by being mainly data-driven, limiting the sampling to the regions of the interaction space defined by the data.

HADDOCK is capable of using ambiguous experimental data to drive the docking process, such as, NMR chemical shift perturbation (CSP) and mutagenesis data. It does this by transforming the data into ambiguous interaction restraints (AIRs) that define a large network of ambiguous distances between residues expected to be involved in the binding mode without imposing any specific orientation on the components. Since the original publication [3] HADDOCK has been extended to handle other NMR sources of information such as residual dipolar couplings (RDCs) [5] diffusion relaxation [6], and pseudo-contact shifts [7]. Other low-resolution data such as small angle X-ray scattering (SAXS) [8] and cross-link data from mass-spectrometry can also be used for scoring and/or generating models.

In addition to using ambiguous and low-resolution information, HADDOCK is also well known for its way of handling flexibility of the subunits and its final refinement in explicit solvent, which can either be water or, to represent a hydrophobic environment, DMSO [9]. Currently HADDOCK is the most cited software in its category [2] and more importantly, one of the best performing docking software based on the CAPRI (Critical Assessment of PRediction of Interactions) competition, a community-wide experiment that allows comparison of the success of docking software by doing blind tests [10, 11].

HADDOCK has been used to generate quite a number of models deposited in the PDB (~100 to date). One of those structures is the Lys48-linked di-ubiquitin (Ub2) complex (PDB: 2BGF [5]). As is well known, ubiquitin plays a major role in the ubiquitin proteasomal pathway, which is the main mechanism of protein degradation in eukaryotic cells. It is also involved in many regulatory pathways. The minimal required signal for flagging proteins to be degraded by the proteasome is Lys48-linked tetra-ubiquitin. To gain insight into the conformation of poly-ubiquitin in solution, Lys48-linked Ub2 has been investigated by NMR, which resulted in CSP, RDC, and ^{15}N-relaxation data [12].

In this chapter, we describe the steps to perform the docking of Lys48-linked Ub2 using HADDOCK, based on CSP and RDC data, mimicking the docking process that led to the 2BGF model. In the Materials section, we provide information about the HADDOCK web server and also describe the data files needed to run this protocol. In the Methods section, we first shortly describe the HADDOCK docking protocol, followed by a step-by-step tutorial describing how we tackled the Lys48-linked Ub2 case using the HADDOCK web server. In the Notes section, we provide more in-depth information about specific aspects of the docking process. We end this chapter with a Case Studies section illustrating some biologically relevant cases where HADDOCK was successfully used.

2 Materials

2.1 Software Requirements

The first requirement to run this protocol is of course access to the HADDOCK software. This can be either through a local copy running on a desktop computer or cluster or more conveniently using the HADDOCK web server, which will be used in this protocol. The HADDOCK software consists of a collection of Python and CNS (Crytallography and NMR System) [13, 14] scripts with additional tools written in various languages. CNS is used as the computational engine that performs the computationally intensive part such as the energy calculations, minimizations and molecular dynamics refinement stages, while the Python routines are used for controlling the data flow, scoring and performing various pre- and post-processing tasks.

To facilitate the use of the software, we have developed the user-friendly HADDOCK web server [15] accessible at haddock.science.uu.nl/services/HADDOCK. Besides eliminating possible dependencies, it also comes with additional error checking and other automatic procedures, which makes it more robust. A special version of the web server making use of European Grid Initiative (EGI, www.egi.eu) resources is also available via the WeNMR web site (www.wenmr.eu) [16]. To use our web server one first needs to register for a user-account. The user-accounts come in various flavors, each giving a different amount of control over the docking and its associated parameters:

- The *Easy interface*, which is usually sufficient and the most straightforward to use, allows the user to upload PDB structures and specify the active and passive residues that define the interface of each molecule.
- The *Expert interface* gives some more control over the docking. In addition to the features available to the Easy interface, it allows the user to manually define the histidine protonation states and to specify which residues should be treated as semi- or fully flexible (both steps are performed automatically at the Easy level).

Furthermore, it permits the upload of user-defined distance, dihedral and hydrogen-bond restraints files and fine-tuning of various restraining and sampling parameters, e.g., the number of structures generated at the various stages.

- Finally, the *Guru interface* allows the user to tweak every parameter as if one was running a local version of HADDOCK. This is also the interface that gives access to additional restraints such as RDCs and relaxation anisotropy data. Symmetry can also be imposed at this level. In addition, it gives full control over almost all parameters including the various force constants and scoring weights for the docking.

Next to these three main interfaces, the server provides four additional interfaces:

- A *Prediction interface*, similar in its input requirements to the Easy interface, but with settings tuned for the use of bioinformatics predictions for docking [17].
- A *Refinement interface,* which only performs the final refinement in explicit solvent for a binary complex (the provided structures should thus already be in proper orientation).
- A *Multibody interface,* which allows the simultaneous docking of up to six different molecules.
- A *File upload interface*, which allows a one click upload of a parameter file previously saved from the web server (useful to repeat a docking with slight changes in parameter settings for example), and a *tool to generate ambiguous distance restraints*, especially useful for multicomponent (>2) systems.

For this tutorial the user should have registered for access to the HADDOCK web server and requested guru access in order to be able to use the Guru interface.

2.2 Data Requirements

The main data requirements to perform a docking run are atomic structures of each of the subunits of the complex in PDB format. These should preferably be structures determined by X-ray crystallography or NMR spectroscopy, but homology models may also be used [18]. For this particular protocol, we use as starting structures the NMR-determined ubiquitin structure 1D3Z, which corresponds to an ensemble of 10 solution structures, and 1AAR, which is a crystal structure with two ubiquitin chains (thus in total an ensemble of 12 structures). The experimental data to drive the docking consist of distance restraints derived from NMR CSP data and orientational restraints derived from RDC data [12]. All necessary files can be found in the corresponding Extra Material at extras.springer.com from where you can download an archive containing the mentioned PDB files, already prepared for docking, the CSP and RDC data files and the HADDOCK/CNS restraint files derived from these data files.

3 Methods

In this section, we first shortly describe the docking protocol that HADDOCK uses so that the user gets a better insight into what happens during the docking process. This helps in understanding the parameters. In the second part, we discuss how to perform the Ub2 docking using CSP and RDC data using the web server. It is assumed that the user has downloaded the tutorial folder from the Extra Materials at extras.springer.com

3.1 The HADDOCK Docking Protocol

Docking in HADDOCK is performed in three consecutive stages:

1. *Rigid body docking* (it0): the subunits are placed randomly in space with an approximate spacing of 25Å between them and subjected to a rigid body energy minimization to form the complex.
2. *Semiflexible refinement* (it1): the top scoring models (default 200 out of 1,000) in the it0-stage are refined using a simulated annealing in torsion-angle space procedure during which the interface is treated as flexible (first side chains only, then both side-chains and backbone).
3. *Flexible refinement in explicit solvent* (itw): in this final stage, the models from it1 are subjected to a gentle restrained molecular dynamics simulation in an explicit solvent shell (either water or DMSO as membrane mimic).

For further details refer to refs. [3, 4].

3.2 Docking Lys48-Linked Ub2 Using the HADDOCK Web Server

In this section, we describe the process of setting up a docking run using the Guru interface (http://haddock.science.uu.nl/services/HADDOCK/haddockserver-guru.html) of the web server. In order to make sense of the docking parameters, some knowledge about the Lys48-linked Ub2 complex is useful. It consists of two ubiquitin subunits that are linked together by a Gly76-Lys48 isopeptide bond. The subunit with the linked Gly76 is called the Distal Domain (ubiD) and the subunit with the linked Lys48 the Proximal Domain (ubiP).

1. Open an Internet browser and go to haddock.science.uu.nl/services/HADDOCK. Choose the Guru interface. This opens up the docking input screen as displayed in Fig. 1. Sections can be expanded or folded by clicking with the left-mouse-button on the double arrows on the right of each section name.
2. First give a name to your docking run. No spaces or special characters other than "-" or "_" are allowed! We named the run di-ubiquitin_CSP_RDC.

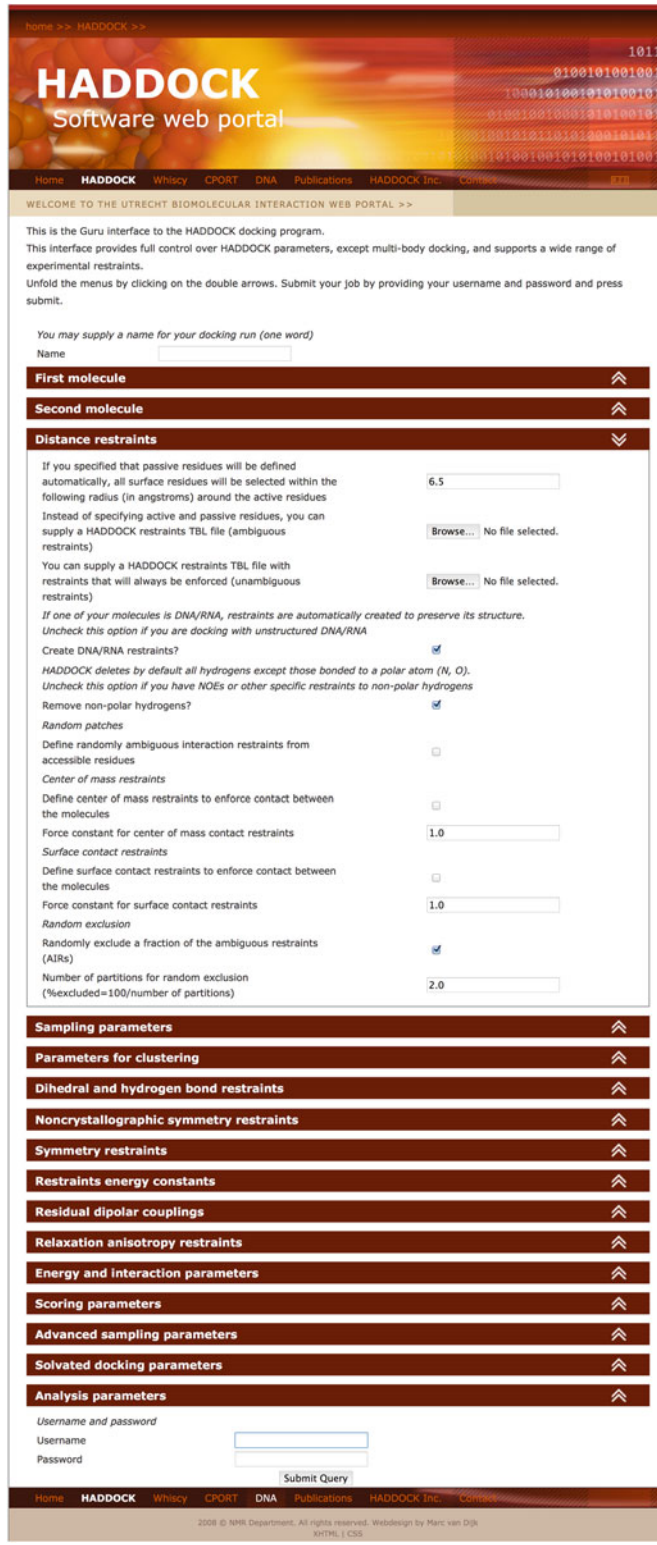

Fig. 1 Overview of the HADDOCK webserver Guru interface (accessible from http://haddock.science.uu.nl/services/HADDOCK). Each section can be expanded by clicking on the *double arrows* on the *right* of the various sections in *red*. In the current view, the distance restraints section is expanded, revealing the forms used to upload the user-defined distance restraints file and various control parameters related to this class of restraints (color figure online)

Table 1
Data used during the docking

Distal domain (ubiD)	
Active residues	8, 9, 46, 47, 48, 49, 51, 68, 72, 73
Passive residues	6, 10, 11, 12, 39, 52, 53, 54, 71, 74, 75, 76
RDCs	46 NH RDCs
Fully flexible segments	72–76
Proximal domain (ubiP)	
Active residues	8, 9, 47, 48, 51, 68, 70, 72, 73, 74, 76
Passive residues	6, 10, 11, 12, 39, 46, 49, 52, 53, 54, 71, 75
RDCs	46 NH RDCs
Fully flexible segments	48; 72–76
Intervector projection angle restraints (VEAN)	***Number of restraints***
Intermolecular	981
Intramolecular	972
Isopeptide bond (Gly76–Lys48)	*Unambiguous restraint distance (Å)*
O–NZ	2.25 ± 0.05
C–NZ	1.35 ± 0.05
C–CE	2.45 ± 0.05
CA–NZ	2.45 ± 0.05

3. Secondly we have to define the PDB file of the first molecule, ubiD, to be docked. Expand the section *First molecule*. At the entry *Where is the structure provided?* click on the dropdown menu next to it and select *I am submitting it*. Set *Which chain of the structure must be used?* to *A* (*see* **Note 1**). Next to *PDB structure to submit* press the *Browse…* button and move to the location where the tutorial data were unpacked. Go to the *pdbs/*directory and select the *1AAR_1D3Z_ensemble.pdb* file (*see* **Note 2**).
4. Specify the interface by defining active and passive residues (*see* **Note 3**). The residues that are considered active from the CSP data are listed in Table 1. Fill in the numbers of the active residues in the textbox next to *Active residues*. Since CSP data typically does not show all residues participating in the binding we also want to define passive residues. Fill in the residue numbers in the textbox next to *Define passive residues* as given in Table 1 (*see* **Note 4**).
5. Specify the *Segment ID to use during the docking* for the first molecule as A (*see* **Note 5**).
6. HADDOCK distinguishes between semi- and fully flexible segments (*see* **Note 6**). The semiflexible segments will be determined automatically during this docking run, but the fully flexible segments need to be defined manually. In this particular case, we want to give more freedom to residues involved in the isopeptide bond and also to the unstructured

C-terminus. Expand the *Fully flexible segments* subsection. Below *Segment 1* set *First number* to *72* and *Last number* to 76. This will make residues 72–76, to which the second ubiquitin is attached, fully flexible.

7. The C-terminus of ubiD should be uncharged, since Gly76 is covalently bound to ubiP. So uncheck *The C-terminus of your protein is negatively charged* box.
8. Expand the *Second molecule* section. Since we are dealing with a peptide-linked homodimer, the *Structure definition* of the second molecule, ubiP, will be the same as ubiD. Set these parameters identical to the values as in the *First molecule* section. However, another set of active and passive residues were derived for this chain, as displayed in Table 1. Again, fill in the active and passive residue numbers in their respective boxes. Set *Segment ID to use during the docking* to *B*. For this chain we define two segments to be fully flexible, namely residue 48, the lysine, and the unstructured C-terminal tail. Set *First number* and *Last number* for *Segment 1*–48 and for *Segment 2*–72 and 76, respectively.
9. In addition to the AIRs that HADDOCK generates using the active and passive residues, additional distance restraint files can be uploaded. In this particular case, we want to include the Gly76-Lys48 isopeptide bond as an unambiguous restraint. These restraints are predefined in the file *restraints/ubiD-ubiP_pepbond.tbl* and shown in Table 1 (*see* **Note 7**). To upload this file, unfold the *Distance restraints* section and click on the *Browse…* button next to *You can supply a HADDOCK restraints…*(*unambiguous restraints)*. Point to the file located on disk. Uncheck the *Randomly exclude a fraction of the ambiguous restraints (AIRs)* box (*see* **Note 8**).
10. Expand the section *Sampling parameters*. Increase the *Number of structures for rigid body docking* to 1,440 (we have 12 starting models for each ubiquitin, giving 144 combinations, each sampled ten times, which amounts to 1,440 models) (*see* **Note 9**). The other default sampling parameters do not need to be changed. So after the rigid body docking stage the top 200 scoring structures will be refined.
11. Go to the *Restraints energy constants* section and in the *Energy constants for unambiguous restraints* subsection change the entries *hot, cool1, cool2*, and *cool3* to 0.1, 1, 5, and 5, respectively.
12. Unfold the *Residual dipolar couplings* section. There are three RDC restraint files in the *restraints/*directory: two with Intervector Projection Angle (VEAN) restraints and the other with direct SANI restraints (*see* **Note 10**). Expand the *Residual dipolar couplings 1* subsection. Set *RDC type* to VEAN.

Expand the *SANI energy constants* subsection and set *First iteration* to 0 and *Last iteration* to 1. The file containing the intermolecular VEAN restraints is named *ubiDP_vean_inter.tbl* (*see* **Note 11**). Upload the file by clicking on the *Browse…* button and selecting it.

13. Expand the *Residual dipolar coupling 2* subsection to upload the second set of RDC restraints, which are intramolecular VEAN restraints. Set *RDC type* to *VEAN* again. In the *SANI energy constants* subsection change *First iteration* and *Last iteration* both to 1. Again, upload the restraints by clicking on *Browse…* and select the r*estraints/ubiDP_vean_intra.tbl* file.
14. The third set of RDC restraints contains the previous two combined but as a SANI restraint. For this, set the *RDC type* to SANI and the *R* and *D* value to 0.057 and −11.49, respectively. The SANI restraints will only be used in the final refinement in explicit solvent (itw). Change the *First iteration* entry to 2 in the *SANI energy constants* subsection. The SANI restraint file is named *ubiDP_sani.tbl*, so define the *RDC file* accordingly.
15. We are now ready to send the docking run to the HADDOCK server. Fill in your Username and Password at the bottom of the screen and press the *Submit Query* button. This sends the information to the server and adds the docking run to the queue once properly validated. You should be redirected to a new page that allows you to download a HADDOCK parameter file containing all parameters and data for your docking run (it is recommended to save it—this file can be uploaded again to the File upload interface). The extra material contains an example of such a file. The page also gives a link to the page that shows the current status of the docking run and where the final results will appear. A confirmation message will be sent to the email address provided at registration.
16. After the docking run has completed, typically after a few hours depending on the server load, an email is sent informing you where to find the results (this is the same link as provided by the server page upon successful submission). By following the link in the email you will be redirected to the HADDOCK web server results page, as shown in Fig. 2.
17. The result page first displays the name of your docking run and its status. It provides you a link where you can download the complete docking run as a gzipped tar file for further manual analysis. Also the docking parameter file containing all of your input data and parameter settings can be downloaded.
18. After that a summary is given of the docking run, giving information about the number of clusters created and how many water-refined models do cluster in these. In this case you should see four clusters containing almost all water-refined models (*see* **Note 12**).

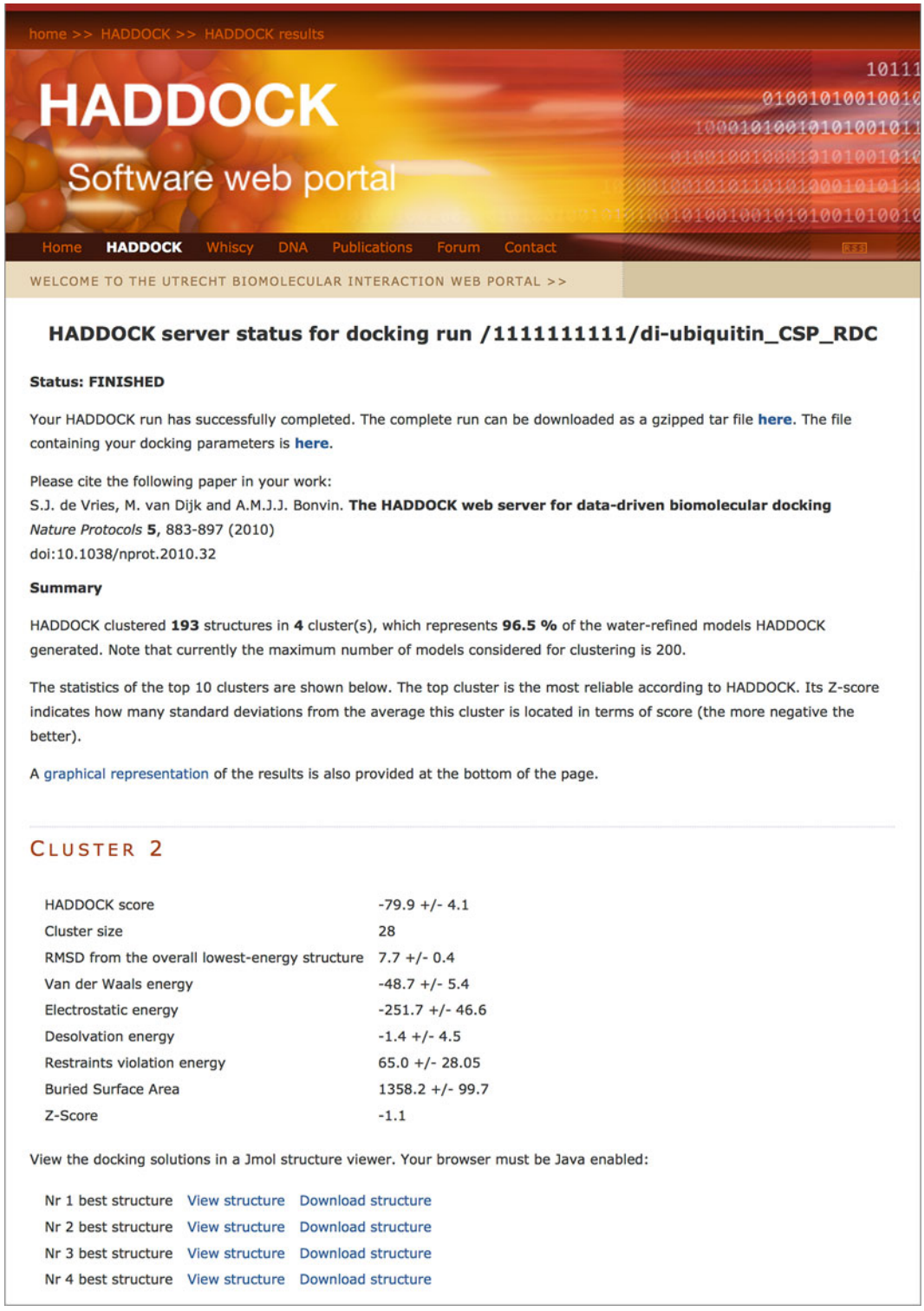

Fig. 2 Example view of a result page of the HADDOCK webserver. This view of the *top* part of the window shows the name of the docking run, its status, and gives information about the number of clusters found. Moreover, it provides detailed information on a per-cluster basis, with the values of the HADDOCK score and its various components indicated. In this view only the top scoring cluster is displayed

19. This is followed by a more rigorous analysis of each cluster. Only the ten best scoring clusters are shown maximally, so in this docking run all clusters are reported. The clusters are named sequentially based on their size, i.e., the largest cluster is named *Cluster 1*. However, the server returns the clusters in the order of their ranking based on the average HADDOCK score of the top four members of each cluster. The best scoring cluster, i.e., the cluster with the smallest HADDOCK score, is the first in the list. For this run *Cluster 2* appears at the top.
20. Each cluster section reports the various average scores with standard deviations based on the top four scoring structures of each cluster, as shown in Fig. 2. First the HADDOCK score (*see* **Note 13**) and the cluster size are given, then the RMSD value of the top four members of the cluster with respect to the overall lowest-energy structure. This is followed by the values of the individual energy terms used in the HADDOCK score, such as the van der Waals, electrostatic, desolvation and restraints violation energies, and ends with the buried surface area (BSA) in ångstrom (*see* **Note 14**) and the cluster *Z*-score (*see* **Note 15**). In addition, links are provided to the PDB files of the four best scoring structures in the cluster, which can be viewed online with Jmol or downloaded for further analysis.
21. After the individual cluster analysis, the results are displayed graphically in the *Results analysis* section, as displayed in Fig. 3. When you click on a plot a larger version appears in the browser. In each plot a dot represents a model and the color of the dot indicates the cluster to which it belongs. The cluster averages with standard deviations are displayed as colored triangles with associated error bars, based again on the top four scoring structures in the cluster. The first three plots show the HADDOCK score versus the interface-ligand-RMSD (i-l-RMSD), the i-RMSD, and the l-RMSD, respectively (*see* **Note 16**). The next plot displays the HADDOCK score versus the fraction of common contacts (FCC) (*see* **Note 17**). The last three plots show the van der Waals, electrostatics, and AIRs energy versus i-RMSD.
22. The web page ends with some supplementary information, which explains the abbreviations used and notifies you that the HADDOCK results will be deleted after a week. So make sure to download the docking run.
23. Congratulations, you performed and analyzed your first docking run using CSP and RDC data! To see whether everything worked out nicely you can compare the resulting structures to the published 2BGF PDB file, which can be downloaded from the PDB website (http://www.pdbe.org).

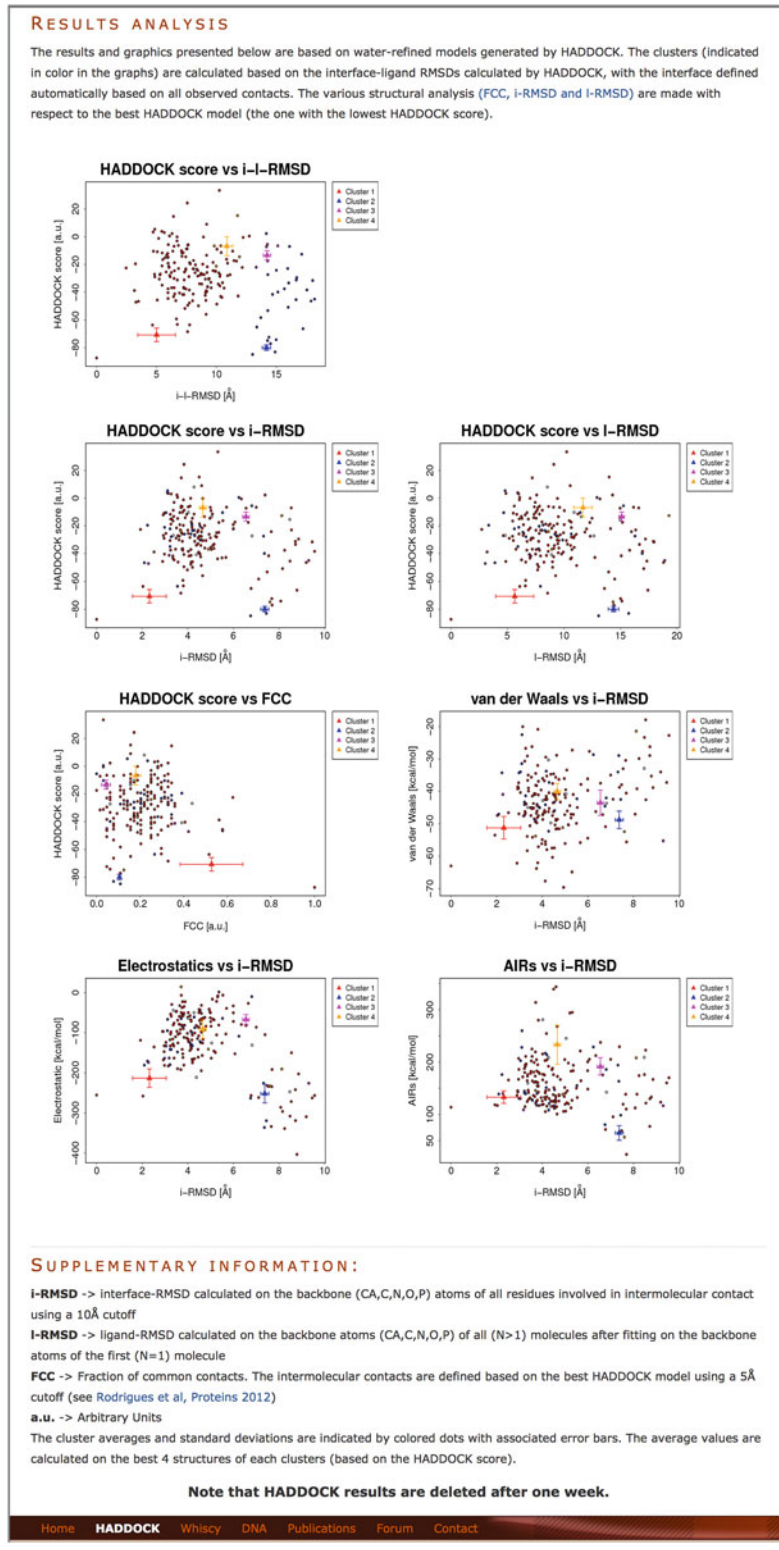

Fig. 3 Example view of a result page of the HADDOCK webserver. This view is of the *bottom* part of the window, which shows a graphical analysis of the results, displaying various scores and energy terms as well as cluster averages versus various RMSD values calculated with respect to the best scoring solution. The various clusters are color-coded. By clicking on a specific plot, an enlarged version is displayed for better viewing

4 Case Studies

The HADDOCK software has been used to solve quite a number of biologically relevant questions as illustrated by the high number of citations and the resulting models deposited in the PDB. One of these cases involved plectasin, a fungal defensin, and the bacterial cell-wall precursor Lipid II [22]. Defensins are host defense peptides that are part of the innate immune system and have antibiotic activity. Usually their activity is explained by their amphipathic structure, which binds and subsequently disrupts the microbial cytoplasmic membranes. Surprisingly, it was discovered that plectasin targets the bacterial cell-wall precursor Lipid II. HADDOCK was used, including CSP data, to unravel the primary binding mode of plectasin to Lipid II. The resulting model revealed that the interaction involved, via hydrogen bonding, the pyrophosphate moiety of Lipid II and several amide protons of plectasin. The resulting docking model, in combination with other experimental data, strongly supports a model in which plectasin gains affinity and specificity by binding to the solvent-exposed part of Lipid II, while its hydrophobic part interacts with the membrane (which was also revealed by NMR CSP data). Such studies can provide important insights for the development of new classes of antibiotics that are highly required considering the increase of resistant bacterial strains.

Other application examples of HADDOCK can be found in the latest CAPRI rounds [18, 23], where models of interactions have to be predicted in a blind manner. These are then compared to the actual crystal structure of the complex based on well-defined criteria. These criteria are i-RMSD, l-RMSD (as explained in **Note 16**), and the fraction of native contacts, which is the percentage of common residue contacts found in the binding mode of the predicted model with the crystal structure [11]. The models are ranked as either incorrect, acceptable, medium, or high quality. In the most recent CAPRI evaluation [23, 24], out of ten complexes that the HADDOCK group predicted, nine were of at least acceptable quality. This is a remarkable performance especially considering that several of the targets required first prediction of the structure of one of the components [18]. The modeling challenges were diverse and consisted of protein–protein complexes, dimers as well as multimers, a protein–polysaccharide complex, the prediction of the hydration structure at a protein–protein interface and even involved engineered interactions in designed complexes.

In conclusion, HADDOCK has gained a unique place among both the docking and experimental communities by being data-driven and having the abilities to handle flexibility and incorporate explicit water during the modeling process. Its user-friendly web interface makes it easily accessible to the science community. This is reflected by its large user group and the diverse scientific endeavors where it has been used to answer and provide insight into biology questions.

5 Notes

1. It is important to check the chainID values in the PDB file that you are uploading. If the chainID column is empty, simply set *Which chain of the structure must be used?* to *Any*. In this case, the ubiquitin models consist of only one chain denoted A.
2. The HADDOCK web server can deal with ensembles in PDB files as long as each model has the same number of residues and atoms. The PDB file *1AAR_1D3Z_ensemble.pdb* contains the 10 original NMR models from the 1D3Z entry and 2 from the 1AAR entry, making a total of 12 models. The models in the PDB file are separated by *MODEL/ENDMDL* statements. The two 1AAR chains were renumbered and processed to match the number of atoms and residues with 1D3Z and to make it compliant with the web server.
3. Active and passive residues are handled differently within the HADDOCK protocol as follows. HADDOCK generates distance restraints between the active residues of the first molecule and active and passive residues of the second molecule and vice versa. This means that, for each chain, active residues "feel" all active and passive residues of all other chains (unless specific chain selections are made using the *generate AIR* web server tool (http://haddock.science.uu.nl/services/GenTBL)). Contrarily, passive residues only "feel" active residues of other chains.
4. The active and passive residues were determined using 1H and 15N CSP data as follows. An active residue has a combined 1H and 15N CSP above average (0.033 ppm) and its backbone or side chain a relative solvent-accessible surface area of higher than 50 %. The solvent-accessible neighboring residues were defined as passive. Instead of manually selecting active residues, these can be automatically defined using the SAMPLEX software, which we developed (*see* ref. 19).
5. Make sure that the *Segment ID to use during the docking* is the same as was used during the creation of the restraint files. We gave ubiD the segment ID *A* and ubiP the segment ID *B* when creating the restraints.
6. The HADDOCK software distinguishes between semi- and fully flexible residues. Semiflexible residues become flexible during the last two stages of it1: First only their side-chain dihedral angles are allowed to vary, and then in the final simulated annealing stage both side chain and backbone are treated as flexible. Fully flexible residues are treated as flexible (both backbone and side-chain dihedral angles) from the start of the flexible refinement stage (it1), i.e., also during the high temperature searches.

7. The Gly76-Lys48 isopeptide bond is incorporated in this docking run as unambiguous distance restraints instead of treating it as a covalent bond (*see* Table 1). This allows the separation of the two ubiquitin chains at the beginning of the docking for a better sampling of conformations.
8. When *Randomly exclude a fraction of the ambiguous restraints (AIRs)* is checked (which is the case by default), a given percentage of restraints (default 50 %) is randomly discarded for each docking trial. In this way, "bad" data will be removed from time to time, ideally leading to better solutions. Thus, it provides a way to deal with false-positive predictions. When using bioinformatics predictions, this percentage can be as large as 87.5 % (default value on the Prediction interface server).
9. The entry *Number of structures for rigid body docking* defines the number of structures that are written to disk after the rigid body energy minimization. However, the parameter *Number of trials for rigid body energy minimization* gives the number of internal trials for the rigid body docking procedure. In addition, if the *Sample 180° rotated solutions during rigid body EM* box is checked then 180° rotated solutions with respect to the normal to the interface are automatically sampled. So, effectively, each model written to disk is the result of 10 docking trials (5 trials × 2 rotated solutions). For the di-ubiquitin case, the total number of models that are sampled amounts to 1,440 × 5 × 2 for a total of 14,400, or 100 docking poses per combination of starting structures (12 × 12).
10. The RDC restraints are incorporated into HADDOCK in two different ways. The first option is as direct SANI restraints where the two molecules are orientationally restrained with respect to an externally defined tensor. However, the RDCs can also be interpreted as Intervector Projection Angle Restraints [20] or VEAN restraints in CNS. This defines orientational restraints directly between two residues, eliminating the need of the cumbersome external tensor formalism, which has also been shown to facilitate the sampling in the initial structure calculation stages [20]. The use of VEAN restraints during the rigid body and first refinement stage and SANI restraints in the last refinement stage in water has been shown to be slightly superior in comparison to using only SANI or VEAN restraints [5].
11. Since the VEAN restraints represent intervector projection angles between two residues, the restraints can be divided into intermolecular and intramolecular restraints. During the rigid body energy minimization stage the intramolecular restraints serve no purpose since each molecule is kept rigid and so cannot change its conformation. Because of this, the intermolecular

restraints are used during the rigid body and first refinement stage, while the intramolecular restraints are only used in the first refinement stage.

12. Usually not all generated models do cluster, since a cluster should consist of minimally four structures maximally separated by an interface-ligand-RMSD (*see* **Note 16**) cut-off distance of 7.5 Å (by default). The minimal cluster size or cut-off value can be changed in the *Parameters for clustering* section at the Guru interface.
13. The HADDOCK score is a heuristic empirical function, which is a linear combination of several physical and empirical energy terms and a BSA term in the first stages. The HADDOCK score puts different weights on its components during each docking stage.
14. The BSA is calculated as follows. First the solvent-accessible surface area is calculated of each of the separated subunits and the modeled complex. The resulting BSA is the difference between the sum of the individual surface areas and the modeled complex.
15. The *Z*-score indicates how many standard deviations from the average a cluster is located in terms of its HADDOCK score. So the more negative the better.
16. All reported RMSDs are calculated with respect to the lowest scoring model (the best model according to the HADDOCK score). The i-l-RMSD, which is used for clustering if RMSD clustering is defined, is calculated on the interface backbone atoms of all chains except the first one after fitting on the backbone atom of the interface of the first molecule. The i-RMSD is calculated by fitting on the backbone atoms of all the residues involved in intermolecular contacts within a cutoff of 10 Å. The l-RMSD is obtained by first fitting on the backbone atoms of the first molecule and then calculating the RMSD on the backbone atoms of the remaining chains.
17. The FCC is the fraction of residue–residue contacts that two structures have in common. It can be used to cluster structures, and is faster, biologically more relevant, and has advantages in the case of symmetrical multimers in comparison to RMSD clustering [21].

Acknowledgments

Financial support from the Dutch Foundation for Scientific Research (NWO) (ECHO grant no. 711.011.009 and VICI grant no. 700.56.442) and the European Union (FP7 e-Infrastructure grant WeNMR no. 261572) is acknowledged.

References

1. Stumpf MPH, Thorne T, de Silva E et al (2008) Estimating the size of the human interactome. Proc Natl Acad Sci USA 105: 6959–6964
2. Moreira IS, Fernandes PA, Ramos MJ (2010) Protein-protein docking dealing with the unknown. J Comput Chem 31:317–342
3. Dominguez C, Boelens R, Bonvin AMJJ (2003) HADDOCK: a protein-protein docking approach based on biochemical or biophysical information. J Am Chem Soc 125: 1731–1737
4. de Vries SJ, van Dijk ADJ, Krzeminski M et al (2007) HADDOCK versus HADDOCK: new features and performance of HADDOCK2.0 on the CAPRI targets. Proteins 69:726–733
5. van Dijk ADJ, Fushman D, Bonvin AMJJ (2005) Various strategies of using residual dipolar couplings in NMR-driven protein docking: application to Lys48-linked di-ubiquitin and validation against 15N-relaxation data. Proteins 60:367–381
6. van Dijk ADJ, Kaptein R, Boelens R et al (2006) Combining NMR relaxation with chemical shift perturbation data to drive protein-protein docking. J Biomol NMR 34: 237–244
7. Schmitz C, Bonvin AMJJ (2011) Protein-protein HADDocking using exclusively pseudocontact shifts. J Biomol NMR 50:263–266
8. Karaca E, Bonvin AMJJ (2013) On the usefulness of ion-mobility mass spectrometry and SAXS data in scoring docking decoys. Acta Crystallogr D Biol Crystallogr 69:683–694
9. van Dijk ADJ, Bonvin AMJJ (2006) Solvated docking: introducing water into the modelling of biomolecular complexes. Bioinformatics 22: 2340–2347
10. Janin J (2005) Assessing predictions of protein-protein interaction: the CAPRI experiment. Protein Sci 14:278–283
11. Lensink MF, Wodak SJ (2013) Docking, scoring and affinity prediction in CAPRI. Proteins 81:2082–2095
12. Varadan R, Walker O, Pickart C et al (2002) Structural properties of polyubiquitin chains in solution. J Mol Biol 324:637–647
13. Brünger AT, Adams PD, Clore GM et al (1998) Crystallography & NMR system: a new software suite for macromolecular structure determination. Acta Crystallogr D Biol Crystallogr 54:905–921
14. Brünger AT (2007) Version 1.2 of the crystallography and NMR system. Nat Protoc 2:2728–2733
15. de Vries SJ, van Dijk M, Bonvin AMJJ (2010) The HADDOCK web server for data-driven biomolecular docking. Nat Protoc 5: 883–897
16. Wassenaar T, van Dijk ADJ, van Dijk M et al (2012) WeNMR: structural biology on the grid. J Grid Comp 10:743–767
17. de Vries SJ, Bonvin AMJJ (2011) CPORT: a consensus interface predictor and its performance in prediction-driven docking with HADDOCK. PLoS One 6:e17695
18. Rodrigues JPGLM, Melquiond ASJ, Karaca E et al (2013) Defining the limits of homology modelling in information-driven protein docking. Proteins 81:2119–2128
19. Krzeminski M, Loth K, Boelens R et al (2010) SAMPLEX: automatic mapping of perturbed and unperturbed regions of proteins and complexes. BMC Bioinformatics 11:51
20. Meiler J, Blomberg N, Nilges M et al (2000) A new approach for applying residual dipolar couplings as restraints in structure elucidation. J Biomol NMR 16:245–252
21. Rodrigues JPGLM, Trellet M, Schmitz C et al (2012) Clustering biomolecular complexes by residue contacts similarity. Proteins 80: 1810–1817
22. Schneider T, Kruse T, Wimmer R et al (2010) Plectasin, a fungal defensin, targets the bacterial cell wall precursor Lipid II. Science 328:1168–1172
23. Janin J (2013) The targets of CAPRI rounds 20-27. Proteins 81:2075–2081
24. Lensink MF, Méndez R, Wodak SJ (2007) Docking and scoring protein complexes: CAPRI 3rd edition. Proteins 69:704–718