Integrative structure modeling with the Integrative Modeling Platform

Benjamin Webb,[1] Shruthi Viswanath,[1] Massimiliano Bonomi,[2] Riccardo Pellarin,[3] Charles H. Greenberg,[1] Daniel Saltzberg,[1] and Andrej Sali[1]*

[1]California Institute for Quantitative Biosciences, University of California, San Francisco, California 94158
[2]Department of Chemistry, University of Cambridge, Cambridge, United Kingdom
[3]Structural Bioinformatics Unit, Institut Pasteur, CNRS UMR 3528, Paris, France

Received 3 July 2017; Accepted 25 September 2017
DOI: 10.1002/pro.3311
Published online 28 September 2017 proteinscience.org

Abstract: Building models of a biological system that are consistent with the myriad data available is one of the key challenges in biology. Modeling the structure and dynamics of macromolecular assemblies, for example, can give insights into how biological systems work, evolved, might be controlled, and even designed. Integrative structure modeling casts the building of structural models as a computational optimization problem, for which information about the assembly is encoded into a scoring function that evaluates candidate models. Here, we describe our open source software suite for integrative structure modeling, *Integrative Modeling Platform* (https://integrativemodeling.org), and demonstrate its use.

Keywords: integrative modeling; hybrid modeling; computational optimization; structural biology

Introduction

To understand the function of a macromolecular assembly, we must know the structure of its components and the interactions between them.[1–4] However, direct experimental determination of such a structure is generally rather difficult, as no experimental method is universally applicable. For example, crystals suitable for X-ray crystallography cannot always be produced, especially for large assemblies of multiple components.[5] Cryo-electron microscopy (cryo-EM), on the other hand, can be used to study large assemblies, but it is generally limited to worse than atomic resolution.[6–8] Finally, molecular biology, biochemistry, and proteomics techniques, such as yeast two-hybrid,[9] affinity purification,[10] and mass spectrometry,[11] yield information about the interactions between proteins, but not the positions of these proteins within the assembly or the structures of the proteins themselves.

One approach to solve this problem is integrative modeling,[12] which is an approach for characterizing the structures of large macromolecular assemblies that relies on multiple types of input

Importance/Impact: Building models of a biological system that are consistent with the myriad data available is one of the key challenges in biology. Integrative structure modeling casts the building of structural models as a computational optimization problem, for which information about the assembly is encoded into a scoring function that evaluates candidate models. Here, we describe our open source software suite for integrative structure modeling, *Integrative Modeling Platform* (https://integrativemodeling.org), and demonstrate its use.

Grant sponsor: National Institutes of Health; Grant numbers: R01 GM083960, P01 1AI091575l, P01 GM105537, and P01 GM71790.

*Correspondence to: Andrej Sali, UCSF MC 2552, Mission Bay, Byers Hall, 1700 4th Street, Suite 503B, San Francisco, CA 94143. E-mail: sali@salilab.org

information, including from varied experiments, physical theories and statistical analysis. Therefore, it maximizes the accuracy, precision, completeness, and efficiency of structure determination. Moreover, it can often produce a structure for systems that are refractive to traditional structure determination methods, such as X-ray crystallography, EM, and nuclear magnetic resonance (NMR) spectroscopy. Numerous structures have already been solved using this technique. For example, the structure of the 26S proteasome was determined from an EM map of the whole assembly, proteomics data about its subunit composition, and comparative protein structure models of the component proteins.[13] The structure of the bacterial type II pilus was assembled from sparse NMR data and X-ray crystallographic structures of constituent proteins.[14] The structure of chromatin around the alpha-globin gene was assembled from so-called 5C data (chromosome conformation capture carbon copy).[15] A two-state model of the signaling mechanism of the PhoQ sensor histidine kinase was constructed using disulfide cross-linking data.[16] The overall architecture of the yeast nuclear pore complex (NPC)[17,18] was also determined by integrative modeling, integrating information from multiple sources, including stoichiometry from protein quantification, protein proximities from subcomplex purification, protein positions from immuno-EM, sedimentation analysis that sheds light on protein and subcomplex shapes, and the overall NPC shape from EM, resulting in an ensemble of medium-resolution models. The models were summarized by a three-dimensional (3D) probability map, resembling an EM map and localizing the 456 constituent proteins with an average precision of approximately 5 nm. This map has revealed fundamental new insights into the function of the NPC as a gatekeeper controlling the entry into and exit from the nucleus of macromolecules, and also shed light on its evolution.[17,19–21] Furthermore, this medium-resolution model has informed further modeling of the NPC components at higher resolution, including the Nup84 heptamer,[10,22] Pom152,[23] and the Nup82 subcomplex.[24]

The integrative structure determination procedure used here[18,25,26] is schematically shown in Figure 1. It proceeds through four stages. The first step is to collect all information that describes the system of interest, including data from wet lab experiments, structural propensities such as atomic statistical potentials,[27,28] and physical laws such as molecular mechanics force fields.[29] Second, a suitable representation for the system is chosen depending on the quantity and resolution of the available information. Different parts of a model may be represented at different resolutions, and a given part of the model may be represented at several different resolutions simultaneously. The available information is then translated into a set of spatial restraints on the components of the system. For example, in the case of characterizing the molecular architecture of the NPC,[17,18] atomic structures of the protein subunits were not available, but the approximate size and shape of each protein was known, so each protein was represented as a "string" of connected spheres whose volumes were consistent with its size and shape. A simple distance between two proteins can be restrained by a harmonic function of the distance, while the fit of a model into a 3D EM density map can be scored by means of the cross-correlation between the model and experimental densities. Next, the spatial restraints are combined into a single scoring function that ranks alternative models based on their agreement with input information. Third, the alternative models are sampled using a variety of techniques, such as conjugate gradients, molecular dynamics, Monte Carlo (MC),[30] and divide-and-conquer message passing methods.[31] This sampling generally generates not a single structure, but an ensemble of models that are as consistent with the input information as possible. There may be many models that score well if the data are incomplete, or none if the data are inconsistent with each other due to errors or unconsidered multiplicity of states of the assembly. Finally, input information and output structures need to be analyzed to estimate structure precision and accuracy, detect inconsistent and missing information, and to suggest more informative future experiments. Assessment begins with structural clustering of the modeled structures produced by sampling, followed by assessment of the thoroughness of structural sampling, estimating structure precision based on variability in the ensemble of good-scoring structures, quantification of the structure fit to the input information, structure assessment by cross-validation, and structure assessment by data not used to compute it.

Integrative modeling can iterate through these stages until a satisfactory model is built. Many iterations of the cycle may be required, given the need to gather more data as well as to resolve errors and inconsistent data.

We have developed the Integrative Modeling Platform (IMP) software (https://integrativemodeling.org/)[17,18,26,32–34] to implement this integrative modeling procedure. Integrative modeling problems vary in size and scope. Thus, IMP offers a great deal of flexibility and several abstraction levels as part of a multitiered platform (Fig. 2). At the lowest level, IMP is designed as a set of "building blocks," providing components and tools to allow method developers to convert data from new experimental methods into spatial restraints, to implement sampling and analysis techniques, and to implement an integrative modeling procedure from scratch, using the C++ and Python programming languages. IMP is

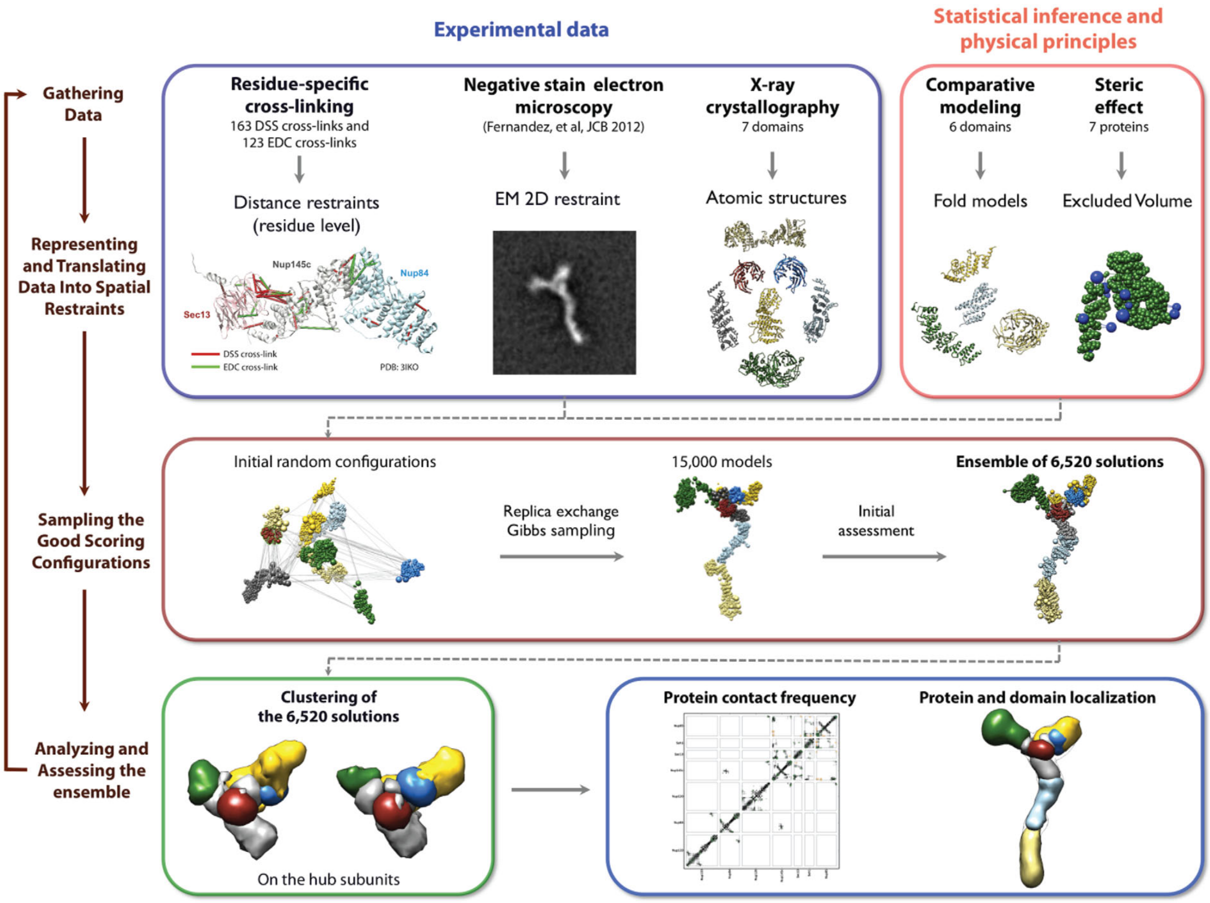

Experimental data
Statistical inference and physical principles
Gathering Data
Representing and Translating Data Into Spatial Restraints
Sampling the Good Scoring Configurations
Analyzing and Assessing the ensemble
Residue-specific cross-linking
163 DSS cross-links and 123 EDC cross-links
Distance restraints (residue level)
Nup145c
Nup84
Sec13
DSS cross-link
EDC cross-link
PDB: 3IKO
Negative stain electron microscopy
(Fernandez, et al, JCB 2012)
EM 2D restraint
X-ray crystallography
7 domains
Atomic structures
Comparative modeling
6 domains
Fold models
Steric effect
7 proteins
Excluded Volume
Initial random configurations
Replica exchange Gibbs sampling
15,000 models
Initial assessment
Ensemble of 6,520 solutions
Clustering of the 6,520 solutions
On the hub subunits
Protein contact frequency
Protein and domain localization

Figure 1. The four stages of integrative structure modeling (see text). These are illustrated by the determination of the Nup84 subcomplex of the yeast NPC.[10]

freely available as open source software under the terms of the GNU Lesser General Public License. To allow a community of developers to easily add sources of information, sampling schemes and analysis methods, IMP is structured as a collection of self-contained modules that can be developed and distributed independently. In IMP, models are encoded as collections of particles, each representing a piece of the system. Depending on the data available, particles can be used to create atomic, coarse-grained, and/or hierarchical representations. It is straightforward to represent a protein at any resolution, from fully flexible atomic models (one particle per atom), to rigid bodies, to coarse-grained models consisting of only one or a few particles for the whole protein. Different parts of the model can be represented differently, as dictated by the available information. Each particle has associated attributes, such as coordinates, radius, residue information, and mass.

Candidate models are evaluated by a scoring function composed of terms called spatial restraints, each of which measures how well a model agrees with the information from which the restraint was derived. A restraint encodes what is known about structures in general or what is known about this particular structure. Thus, a candidate model that scores well is consistent with all used information. The precision and accuracy of the resulting model increases with the amount and quality of information, that is, encoded in the restraints. IMP's growing set of restraints supports small angle X-ray (SAXS) profiles,[35] proteomics data,[36] EM images and density maps,[32,37] most of the NMR spectroscopy-derived restraints,[14] the CHARMM force-field,[29] restraints implied by an alignment with related structures,[38] cross-linking,[39] hydrogen-deuterium exchange,[40] chromosome conformation capture,[15] Förster resonance energy transfer (FRET),[41] and a variety of statistical potentials.[28]

For most applications, the full flexibility of defining a system from the bottom up as sets of particles is unnecessary. IMP provides a higher-level interface called Python Modeling Interface (PMI) that allows for a top-down representation of the system, using biological names for protein subunits. It provides simple mechanisms to set up higher order structure, such as multiple copies of subunits or symmetry-related subsets of the system, at multiple resolutions. It also allows easy setup of the myriad advanced restraints available in IMP. Finally, it provides ready-built protocols and other utilities, for example to generate publication-ready plots. Using

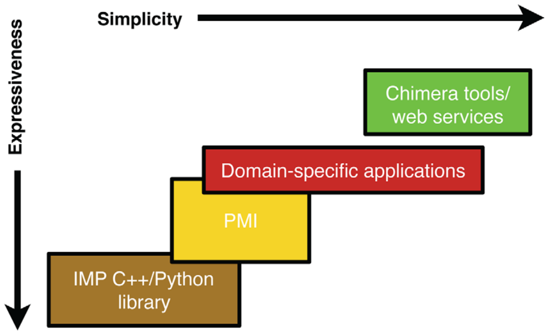


Figure 2. Multitiered organization of the IMP software. At the lowest level, IMP provides a C++/Python "tool box" of interchangeable components to build arbitrary integrative modeling protocols. To solve common modeling problems, it also provides a higher-level programming interface called PMI, and command-line tools, web services, and linkages with the Chimera visualization software.

PMI, the entire modeling protocol can be described with a set of Python scripts, which are typically deposited, together with the input data and output models, in a publicly available repository, such as GitHub and the nascent Protein Data Bank (PDB) archive of integrative structures (https://pdb-dev.wwpdb.org/);[42] for examples, see Refs. 22,24,39, and 43–48.

Finally, at the highest abstraction levels, for users with limited programming experience, IMP provides less flexible but more user-friendly applications to handle specific tasks, such as fitting of proteins into a density map of their assembly,[31] scoring protein-ligand interactions,[49] combining multiple SAXS profiles,[50] comparing a structure with the corresponding SAXS profile,[51–53] or enriching pairwise docking using SAXS data,[52] and can be used through web interfaces, from Chimera,[54] or from the command line.

IMP has been used to produce a wide variety of models; for example, a eukaryotic ribosome,[55] a ryanodine receptor channel,[56] the yeast Mediator complex,[46] the Hsp90 chaperonin,[57] a yeast exosome in multiple states,[45] the actin-scruin complex,[58] deoxyribose nucleic acid (DNA) transcription factor II H (TFIIH),[47] chromatin,[15,59] and the NPC and its subcomplexes.[10,17,22–24] More information about IMP can be found at the IMP website, https://integrativemodeling.org/. The website provides a technical introduction, a tutorial, and a variety of examples to help users get started.

Demonstration

In this section, we will demonstrate the use of the IMP software with two applications—first, determining the structure of a protein complex using IMP's high-level PMI module; and second, building and running a custom integrative modeling protocol using the low-level library. Monospaced text is used below for computer file and folder/directory names, command lines, file contents, and variable and class names.

Software installation

The IMP software is available for Linux, Mac, and Windows at https://integrativemodeling.org/download.html. The examples below also use a number of Python packages (numpy, scipy, scikit-learn, and matplotlib). The easiest way to install IMP and these Python packages is to first install Anaconda Python (https://www.continuum.io) and then run from a command line

```
conda config --add channels salilab
```

```
conda install imp numpy scipy scikit-learn matplotlib
```

Modeling of RNA polymerase II

We will first illustrate the use of the PMI interface by determining the localization of two subunits of the ribonucleic acid (RNA) polymerase II complex, a eukaryotic complex of 12 subunits (Rpb1 to Rpb12) that catalyzes DNA transcription to synthesize mRNA strands. The yeast RNA Pol II dissociates into a 10-subunit core and a stalk-like Rpb4/Rpb7 heterodimer protrusion. We will utilize chemical cross-linking coupled with mass spectrometry,[60,61] negative-stain EM, and X-ray crystallography data to reconstruct this stalk, hypothesizing that we know already the structure of the core. The example can be easily generalized to any other set of subunits.[22,24,39,43–48]

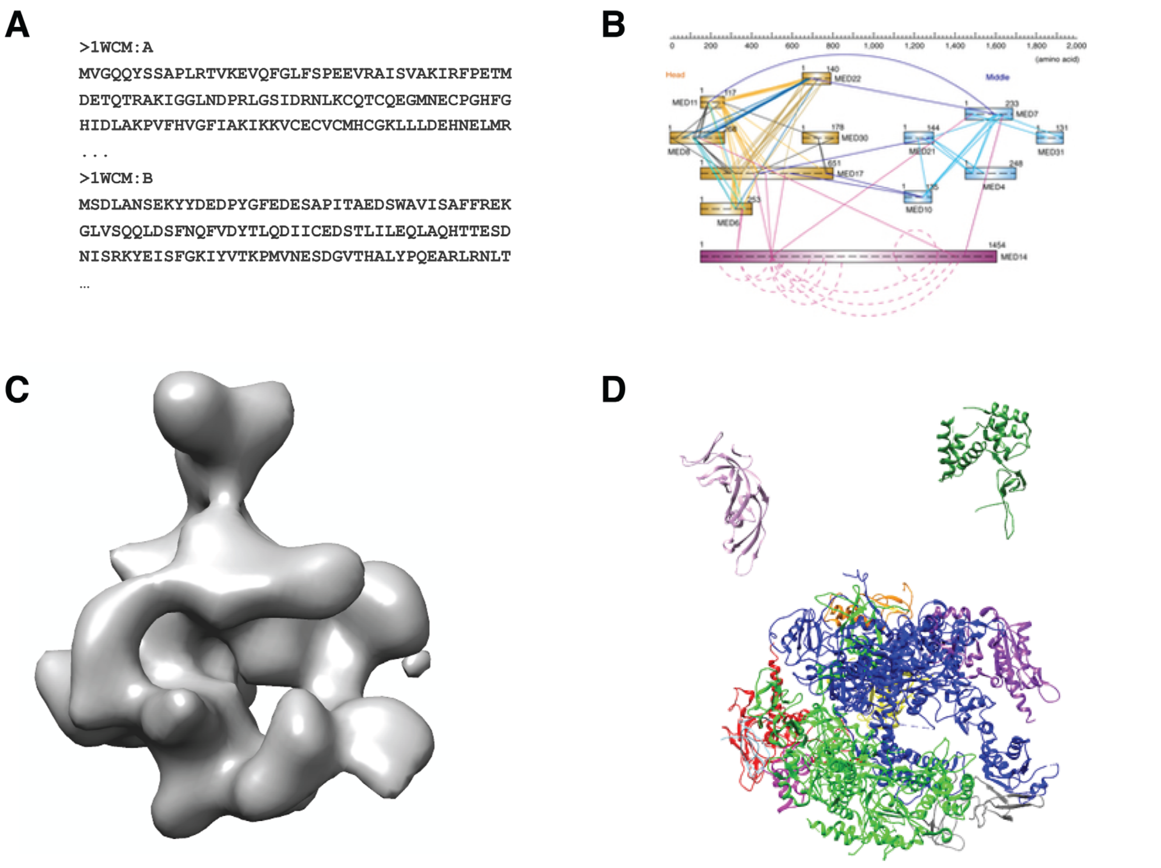
All files for this example are available in a GitHub repository at https://github.com/salilab/imp_tutorial/, which should be downloaded to the same computer where IMP is installed.

PMI is controlled by a Python script. As discussed above (Fig. 1), modeling proceeds in four stages. The first three stages (collecting the experimental data; deciding on a representation for the system and converting the data into spatial restraints; and sampling conformations) are all handled by a single Python script. This script can be run by first changing into the `rnapolii/modeling` directory of the repository and then running the script with Python:

```
python modeling.py
```

Stage 1—Gathering of data. In this stage, we gather all data that we wish to utilize in structural modeling. In theory, any method that provides structural information can be used. The `rnapolii/data` folder contains all of the data included in this example (Fig. 3).

Sequence information. Each residue included in modeling must be specified in a FASTA (https://www.ncbi.nlm.nih.gov/pubmed/2983426) text file, `1WCM.fasta.txt`. This file contains 12 amino acid



```
>1WCM:A
MVGQQYSSAPLRTVKEVQFGLFSPEEVRAISVAKIRFPETM
DETQTRAKIGGLNDPRLGSIDRNLKCQTCQEGMNECPGHFG
HIDLAKPVFHVGFIAKIKKVCECVCMHCGKLLLDEHNELMR
...
>1WCM:B
MSDLANSEKYYDEDPYGFEDESAPITAEDSWAVISAFFREK
GLVSQQLDSFNQFVDYTLQDIICEDSTLILEQLAQHTTESD
NISRKYEISFGKIYVTKPMVNESDGVTHALYPQEARLRNLT
…
```

Figure 3. All data used in the integrative determination of the RNA Pol II structure. A: Primary sequences of all subunits in the FASTA format. B: Chemical cross-linking data, which yields a list of proximate residue pairs. C: A 3D negative-stain EM density map of the entire complex. D: X-ray crystal structures of each of the subunits.

sequences, corresponding to the 12 subunits in the complex.

Chemical cross-links. Two data sets are used here,[62,63] stored in the files `polii_xlinks.csv` and `polii_juri.csv`. Each file contains multiple comma-separated columns; four of these columns specify the protein and residue number for each of the two linker residues. The length of the cross-linker reagent, 21Å, is not stated in this file; it will be specified later in the Python script.

Electron density maps. The EM map of the entire complex is obtained from the EM Data Bank[64,65] and stored in file `emd_1883.map.mrc`. IMP uses Gaussian mixture models (GMMs) to greatly speed up scoring, by approximating the electron density of both individual protein subunits and experimental EM maps as a sum of 3D Gaussians.[66] A GMM has been created for this experimental map, and is stored in `emd_1883.map.mrc.gmm.50.mrc`.

3D structure. High-resolution coordinates for all 12 chains of the complex are found in the PDB,[67,68] stored in file `1WCM.pdb`.

Stage 2—Representation of subunits and translation of the data into spatial restraints. We represent the complex with spherical beads of varying sizes, which coarsen domains of the complex using several resolution scales simultaneously (at the same time, subunits are also represented with 3D Gaussians to fit them against the EM map). The restraints will be applied to individual resolution scales as appropriate. Beads and Gaussians of a given domain are arranged into either a rigid body or a flexible string, based on the crystallographic structures. In a rigid body, all the beads and Gaussians of a given domain have their relative distances constrained during configurational sampling, while in a flexible string they are restrained by the sequence connectivity. This representation is controlled by means of a topology file, `rnapolii/data/topology.txt`, which is read in by the Python script.

After defining the representation of the model, we build the restraints by which the individual structural models will be scored based on the input data.

Excluded volume. The excluded volume restraint (called `ExcludedVolumeSphere` in the Python script) prevents subunits from occupying the same space. For speed, this restraint is applied to the low-resolution representation of the system (20 residues represented by each bead).

Cross-links. A cross-linking restraint[39] (called ISDCrossLinkMS in the script) is implemented as a distance restraint between two residues. The two residues are each defined by the protein name and the residue number. Because the cross-linking information is per-residue, this restraint is applied to the high-resolution representation (one residue per bead). The script also specifies the length of the cross-linking reagent (21 Å). Two such restraints are created in the script, one for each cross-link dataset.

Electron microscopy. The GaussianEMRestraint uses a density overlap function to compare model with data, using the previously created GMMs.

Stage 3—Sampling. We are now ready to sample configurations, guided by the scoring function. We use a replica exchange MC scheme[69] (using the ReplicaExchange0 class in the Python script). Each replica is subjected to 200,000 MC[30] sampling steps; at each step, the system is perturbed by randomly rotating and translating each rigid body, and randomly translating each flexible bead.

The script generates an output directory containing the following:

- pdbs: a directory containing the 100 best-scoring models from the run, in the PDB format.
- rmfs: a single RMF file (Rich Molecular Format, https://integrativemodeling.org/rmf/) containing all the frames. RMF is a file format specially designed to store coarse-grained, multiresolution, and multistate models, such as those generated by IMP. It is a compact binary format and (as in this case) can also be used to store multiple models or trajectories.
- Statistics from the sampling, contained in a "statfile," stat.*.out. This file includes the values of each restraint and MC acceptance criteria.

Stage 4—Analysis. In the analysis stage, we cluster (group by similarity) the sampled models to determine high-probability configurations. Comparing clusters may indicate that there are multiple acceptable configurations given the data. Clustering is done using another Python script called clustering.py, found in the rnapolii/analysis directory. This script performs *k*-means clustering,[70] followed by a basic cluster analysis, including creating localization densities for each subunit. This script can be executed in the same way as before, by changing into the rnapolii/analysis directory and then running:

```
python clustering.py
```

The script generates a new directory containing information on the determined clusters, and a subdirectory for each cluster (Fig. 4). Within the cluster folder are PDB and RMF files containing members of each cluster, localization densities (as .mrc files), and a stat file. All RMF, PDB, and MRC files are viewable in Chimera.[54]

A file dist_matrix.pdf is created containing two plots [Fig. 4(B,C)]. The first plot (panel B) is the distance matrix of the models after being grouped into clusters. The matrix generally shows the requested number of clusters with much lower within-cluster than between-cluster distance; otherwise, too many clusters may have been chosen.

Localization densities can give a qualitative idea of the precision of a cluster. In Figure 4(A), we show results from one cluster: the native structure without Rpb4/7 (as spheres), the target density map (in mesh), and the localization densities for Rpb4 and Rpb7 as 3D surfaces (Rpb4 in yellow, Rpb7 in gray). In this case, the localizations are quite narrow and close to the native solution.

Cluster precision. The precision_rmsf.py script can be used to determine the within- and between-cluster differences. To run, use:

```
python precision_rmsf.py
```

It will generate precision.*.*.out files containing precision information in text format, while in each cluster directory it generates PDF files showing the within-cluster residue root-mean-square fluctuation [Fig. 4(D)].

Sampling exhaustiveness. As discussed above, integrative structure modeling involves defining the representation of the model system, a scoring function based on input information, and a sampling scheme that generates the models for ranking by the scoring function. Any of these three aspects of integrative modeling can limit the accuracy of the output models. These three limitations are discussed in detail in Ref. 71. Rather than repeat this discussion, we focus here on outlining our recent advance on assessing the sampling in particular, namely the assessment of model precision at which sampling is likely exhaustive.

Because enumeration of all structures at a desired precision is generally not feasible (too many degrees of freedom that need to be sampled on too fine a grid), we generally use stochastic sampling methods. Therefore, we cannot guarantee sampling exhaustiveness by construction. No characterization of a model at a certain level of precision is valid if the sampling is not exhaustive at that precision.[71,72] Sampling exhaustiveness is not achieved in this demonstration, which is deliberately undersampling to make it faster, but in production modeling, testing for sampling exhaustiveness is the first step of the analysis and validation stage of our four-stage integrative modeling process (Fig. 1) and an objective

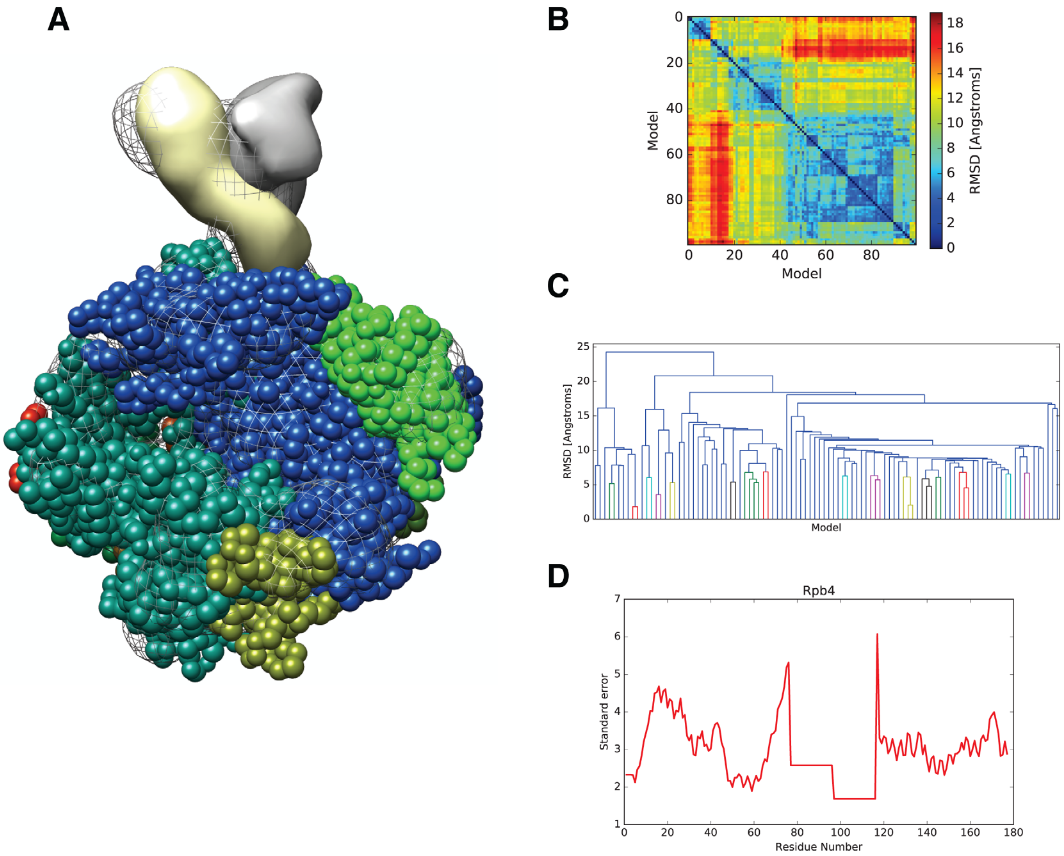


Figure 4. Outputs from modeling of the RNA Pol II stalk. A: One of the two clusters of models. A single model from the cluster of the complex core is shown (as one-residue beads) together with the EM density map (mesh) and the localization densities (probabilities of finding a subunit at each position in space, over the entire cluster) of Rpb4 and Rpb7 (solid surfaces, contoured to enclose each protein's volume). B: Heat map of the distance matrix of all best-scoring models in the two clusters. C: Model-model distances displayed as a dendrogram. D: Root-mean-square fluctuation of each residue in Rpb4 over all structures in one cluster.

and automated protocol for this task has been recently described.[72] As a proxy for sampling exhaustiveness, we evaluate whether or not two independently and stochastically generated sets of models are sufficiently similar. Such samples can be obtained, for example, from two independent simulations using random starting models or different random number generator seeds. The protocol includes testing: (1) convergence of the model score; (2) whether model scores for the two samples were drawn from the same parent distribution; (3) whether each structural cluster includes models from each sample proportionally to its size; and (4) whether there is sufficient structural similarity between the two model samples in each cluster. The evaluation also provides the sampling precision, defined as the smallest clustering threshold that satisfies the third, most stringent test. The protocol is validated with the aid of enumerated good-scoring models for five illustrative cases of binary protein complexes.[72]

The protocol is general in nature and can be applied to the stochastic sampling of any set of models, not only structural models. In addition, the tests can be used to stop stochastic sampling as soon as exhaustiveness at desired precision is reached, thereby improving sampling efficiency; they may also help in selecting a model representation, that is, sufficiently detailed to be informative, yet also sufficiently coarse for sampling to be exhaustive. The protocol is not applicable to nonstochastic sampling methods or expensive sampling methods that cannot generate a large enough sample of independent models. Further, passing the tests is a necessary, but not sufficient condition for exhaustive sampling; a positive outcome of the test may be misleading if, for example, the landscape contains only a narrow, and thus difficult to find, pathway to the pronounced minimum corresponding to the native state. Nevertheless, based on five illustrative binary protein complexes, we argue that convergence of stochastic sampling at some precision often also indicates sampling exhaustiveness at that precision. This protocol will be incorporated into future versions of the IMP software.

Modeling of the yeast spindle pole body core

Background. Accurate duplication, assembly, and regulation of the centrosome is crucial for proper partitioning of chromosomes during cell division.[73–76] The spindle pole body (SPB) of *Saccharomyces cerevisiae* is a simple model of the centrosome.[77] All of its components have been identified and localized within its structure by immuno-EM,[78] yet little is known about its molecular architecture.

We recently[79] determined the molecular architecture of part of the SPB core comprised of the Central Plaque (CP) and Intermediate Layer 2 (IL2) by Bayesian integrative structure modeling,[80] based on data from *in vivo* FRET, SAXS, X-ray crystallography, yeast two-hybrid, EM, and genetic experiments (Fig. 5). The corresponding integrative modeling is outlined below. Files constituting the input data, scripts, and outputs are available in a GitHub repository at https://github.com/integrativemodeling/spb, together with a more detailed description of the protocol.

Software. In this example, the low-level IMP C++ library was used directly (Fig. 2). Restraints, sampling procedures, and modeling protocols specific to the SPB system were implemented in C++ in a custom IMP module called "spb," which is included in IMP version 2.8.0 and later. The tools supporting the four stages of the modeling protocol (Fig. 1) are standalone C++ programs (included with IMP).

Configuration files. All SPB executables take as input a configuration file that specifies their parameters. These files are in the GitHub repository in the `config_files` directory. Each file contains parameters common to all SPB executables (*global parameters*), which are documented in the file itself.

Stage 1—Gathering of data. As before, the first step is to collect all data that describe the system. These data can be found in files in the `inputs` directory.

- Data from EM[81,82] identified the SPB core proteins, which include Spc42, Cmd1, Spc29, Spc110-C, and Cnm67-C.
- 41 $FRET_R$ data points obtained by *in vivo* FRET spectroscopy[83] informed the spatial proximities of the termini of the SPB and the coiled-coil of Spc110 (files `shared_inputs/fret_2014.dat` and `shared_inputs/fret_new_exp.dat`).
- A low-resolution cryo-EM density map of overexpressed Spc42[84] informed the P3 symmetry of Spc42, and other model parameters such as the range of the primitive unit cell size and the maximum coiled-coil tilt angle for Spc42 (file `analysis/SPB_2d_padded.tiff`).
- Data from yeast two-hybrid experiments[82,85] and a Cnm67 binding site for Spc42[86] provided information about interacting domains.
- SAXS profiles of the Spc110/Cmd1 subcomplex and Spc29 informed their shapes and the distance between the N- and C-termini for Spc29.
- The crystal structure of Cnm67-C[86] (in files `shared_inputs/3OA7_A.pdb` and `shared_inputs/3OA7_B.pdb`), homology model of domain-swapped Spc110/Cmd1 subcomplex (files `shared_inputs/4DS7_*_swapped.pdb`), and homology models of coiled-coil domains of Spc42 (files `shared_inputs/CC_78_A.pdb` and `shared_inputs/CC_78_B.pdb`), and Spc110 (files `shared_inputs/CC_120_A.pdb` and `shared_inputs/CC_120_B.pdb`) provided high-resolution structural information on these components.
- Metallothionein-tagged cryo-electron tomography helped identify the possible stoichiometry of Spc29.

Some other sources of data were excluded from the modeling, but used to validate the final model. These include the presence of the IL2-CP gap and hexagonal lattice spacing of Spc42 from EM;[84] genetic analysis of Spc110; molecular weight estimation of Spc110/Cmd1 dimer and Spc29 by SAXS; and yeast two-hybrid data.

Stage 2—Representation of subunits and translation of the data into spatial restraints. Cnm67-C, Spc110/Cmd1 subcomplex, and coiled-coil domains of Spc42 and Spc110 were represented as rigid bodies at a resolution of 10 residues per bead, while green fluorescent proteins (GFPs) were represented as rigid bodies with 50 residues per bead. For domains whose structure is unknown but expected to be extended, such as Spc29 and Spc42-C, only the termini were represented by 10-residue beads. Other domains without known structure were represented by approximately 60 residues per bead.

The restraints on the SPB model included the Bayesian FRET restraint,[41] cryo-EM density restraint, restraints on the unit cell size, CP thickness and tilt angle of coiled-coil domains, planar restraints for Spc42 termini, restraints based on SAXS shape and yeast two-hybrid data, Cnm67-Spc42 binding site restraint, restraint on the length of Spc29, sequence connectivity, and excluded volume restraints.

Stage 3—Sampling. To increase the efficiency of sampling, the model was sampled with symmetry constraints.[79] Each hexagonal supercell was modeled as a triple of rhomboid primitive unit cells. The central supercell and the surrounding six hexagonal supercells were modeled in effect as an infinite lattice. A Gibbs sampler based on MC enhanced by Parallel Tempering (PT)[87] was used to sample model coordinates as well as other parameters from the

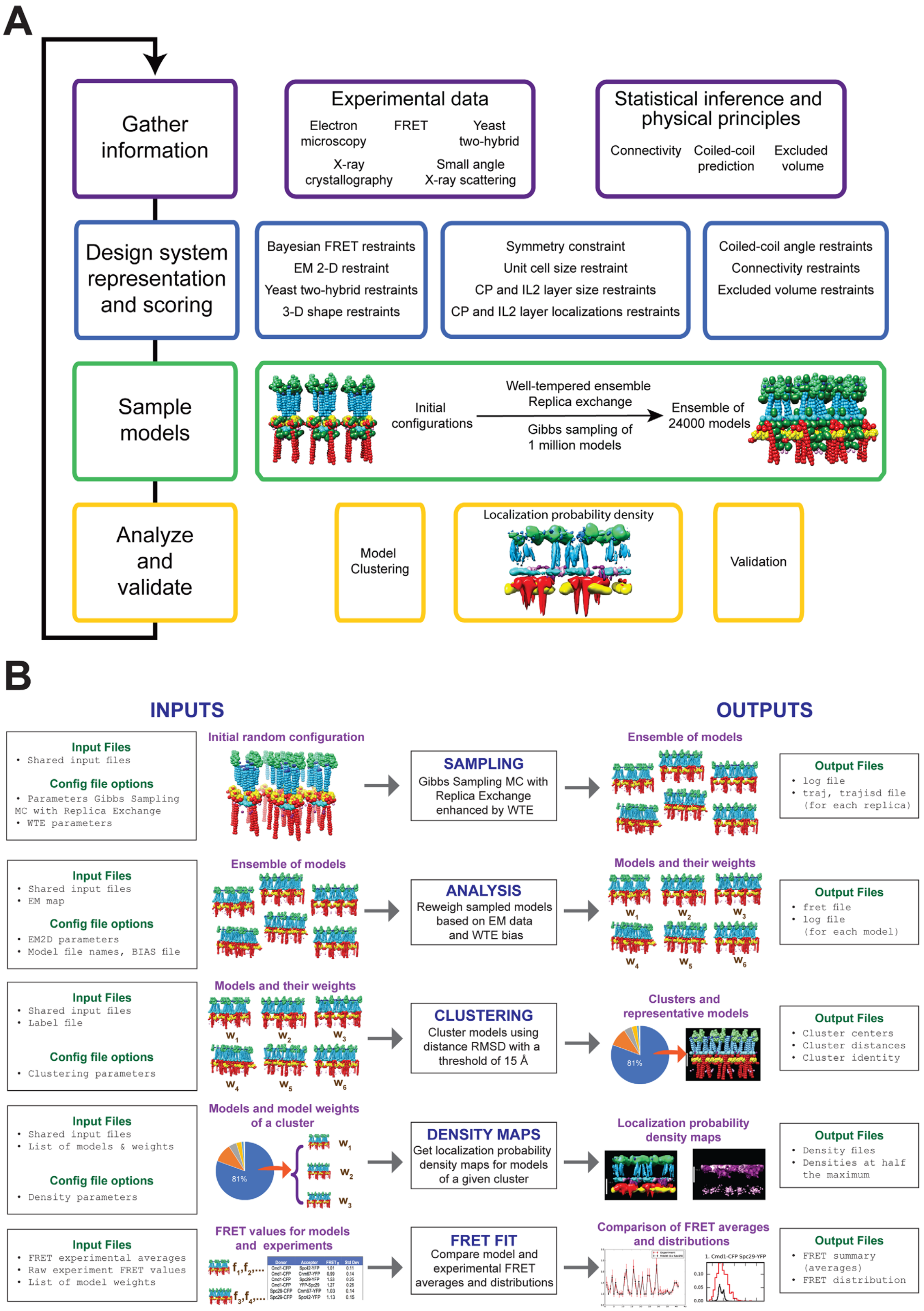

A
Gather information
Experimental data
Electron microscopy
FRET
Yeast two-hybrid
X-ray crystallography
Small angle X-ray scattering
Statistical inference and physical principles
Connectivity
Coiled-coil prediction
Excluded volume
Design system representation and scoring
Bayesian FRET restraints
EM 2-D restraint
Yeast two-hybrid restraints
3-D shape restraints
Symmetry constraint
Unit cell size restraint
CP and IL2 layer size restraints
CP and IL2 layer localizations restraints
Coiled-coil angle restraints
Connectivity restraints
Excluded volume restraints
Sample models
Initial configurations
Well-tempered ensemble
Replica exchange
Gibbs sampling of 1 million models
Ensemble of 24000 models
Analyze and validate
Model Clustering
Localization probability density
Validation
B
INPUTS
OUTPUTS
Input Files
• Shared input files
Config file options
• Parameters Gibbs Sampling MC with Replica Exchange
• WTE parameters
Initial random configuration
SAMPLING
Gibbs Sampling MC with Replica Exchange enhanced by WTE
Ensemble of models
Output Files
• log file
• traj, trajisd file (for each replica)
Input Files
• Shared input files
• EM map
Config file options
• EM2D parameters
• Model file names, BIAS file
Ensemble of models
ANALYSIS
Reweigh sampled models based on EM data and WTE bias
Models and their weights
w_1 w_2 w_3 w_4 w_5 w_6
Output Files
• fret file
• log file (for each model)
Input Files
• Shared input files
• Label file
Config file options
• Clustering parameters
Models and their weights
w_1 w_2 w_3 w_4 w_5 w_6
CLUSTERING
Cluster models using distance RMSD with a threshold of 15 Å
Clusters and representative models
81%
Output Files
• Cluster centers
• Cluster distances
• Cluster identity
Input Files
• Shared input files
• List of models & weights
Config file options
• Density parameters
Models and model weights of a cluster
81%
w_1 w_2 w_3
DENSITY MAPS
Get localization probability density maps for models of a given cluster
Localization probability density maps
Output Files
• Density files
• Densities at half the maximum
Input Files
• FRET experimental averages
• Raw experiment FRET values
• List of model weights
FRET values for models and experiments
$f_1, f_2, \ldots$
$f_3, f_4, \ldots$
Donor
Acceptor
Std Dev
FRET FIT
Compare model and experimental FRET averages and distributions
Comparison of FRET averages and distributions
1. Cmd1-CFP Spc29-YFP
Output Files
• FRET summary (averages)
• FRET distribution

Figure 5. Bayesian integrative structure modeling of the SPB core. A: Structure determination proceeds through four stages: (1) gathering of data, (2) representation of subunits and translation of the data into spatial restraints, (3) configurational sampling to produce an ensemble of structures that satisfies the restraints, and (4) analysis and validation of the ensemble structures. The process is iterative, until an acceptable model is obtained. B: The input data is encoded into spatial restraints and used to generate structural models. Starting from initial random configurations, a MC Gibbs sampler with PT in the WTE is used to generate an ensemble of models (first row). The models are then analyzed and additional weights are calculated to incorporate EM data not used in the sampling and to remove the effect of the WTE bias (second row). Models are subsequently grouped into clusters of similar models (third row). For each cluster, localization densities of individual SPB components are generated (fourth row). Finally, the agreement between experimental FRET data and the structural models is computed (fifth row).

posterior distribution, such as the unit cell size, CP thickness, and FRET parameters. To further enhance sampling, PT was carried out in the well-tempered ensemble (WTE).[88]

This sampling is performed by the `spb` program in IMP. Parameters related to the MC sampler and PT algorithm are specified in the configuration file under `Parameters Gibbs sampling—Monte Carlo + Replica Exchange`. Parameters for the WTE algorithm are specified under the `WTE` section. The parameters were tuned for the SPB modeling and should be adapted when modeling a different system.

Once the configuration file (e.g., `config_files/production/sample/config.ini`) and input files (from `inputs/shared_inputs`) are placed in a working directory, the sampling program `spb` is executed in parallel in that directory using the following command:

```
mpirun -np 8 spb_sample
```

For each replica (eight in this case), the SPB sampling program produces the following output files, that are used in further modeling steps:

- `log`: contains all the basic information for each model produced in the sampling stage;
- `traj`: this trajectory file contains all the structural models produced during sampling, as an RMF file;
- `trajisd`: this trajectory file contains for each structural model stored in `traj` the corresponding value of all the Bayesian parameters used in the modeling.

Stage 4—Analysis, clustering, and fit with data. In the published study,[79] assessment of the SPB models began with a test of the thoroughness of structural sampling. This test included structural clustering of the models. The model precision was estimated as the distance cutoff for defining a cluster (15 Å), and the variability in the ensemble of structures was visualized using localization probability density maps. Next, models were assessed by quantification of the structure fit to the input information. Models were also assessed by cross-validation: each time 95% of the FRET data was randomly selected and modeling was performed to see if the models recovered were similar to the original model and they fit the unused FRET data. Finally, the structures were assessed by data not used to compute them.

Analysis. The program `spb_analysis` analyzes the models produced during the sampling stage. Most importantly, `spb_analysis` quantifies the agreement with the EM data (not used as a restraint in the generation of models) and calculates a reweighting term to update the probability of the model upon incorporation of the EM data. Furthermore, `spb_analysis` calculates the unbiasing weight to correct for the presence of the metadynamics (WTE) bias potential during the sampling stage. The models to analyze are loaded from multiple RMF files, each named `frame.rmf` and containing a single model among all those sampled. These models can be extracted from the RMF generated by program `spb` using the scripts available in the GitHub repository in the `scripts/analysis` directory.

See the `Parameters specific to analysis` section in the config files for parameters used by `spb_analysis`. Once the configuration file is prepared, and all input files are in place, the analysis program `spb_analysis` is executed on all the individual RMF frames, as follows, for each extracted frame (see also `scripts/analysis/job_analysis.sh`):

```
spb_analysis
```

The outputs of `spb_analysis` are, for each frame:

- a `log` file, which contains the total score from all the restraints used during the modeling stage, the rescoring weight to account for the WTE bias potential and the EM map, the values of all the Bayesian parameters, and the cross-correlation with the EM map
- a `fret` file, with the value of the forward model for each FRET_R measurement
- the frame itself, in the `RMF` directory

Clustering. The program `spb_cluster` performs structural clustering of the ensemble of models, taking into account the model unbiasing weight obtained in the previous analysis step, using a modified GROMOS clustering algorithm.[89] The extra input for clustering (apart from shared input files) is `label.dat`, a file containing the names of beads/domains to include in the distance root-mean-square deviation calculation.

See the `Parameters specific to clustering` section in the config files for adjustable parameters.

Before running `spb_cluster`, one needs a file `weight.dat` that lists model weights (see also `scripts/cluster/job_cluster.sh`). The clustering executable is then run as follows:

```
spb_cluster
```

The outputs from `spb_cluster` include three files:

- `cluster_center` contains the list of clusters, with the cluster population, cluster center, cluster diameter, and mean distance between models in the cluster;

- `cluster_distance` contains pairwise distances between cluster centers;
- `cluster_traj_score_weight` contains the cluster identity for each model. Each line contains the model number, its cluster, model score, model weight, and unit cell size in the model.

One can also obtain a representative model (top-scoring model) for each cluster using the `scripts/cluster/get_top_scoring_model.sh` script.

Density maps. The program `spb_density_perbead` calculates the localization probability density maps. Apart from the shared input files, we need one other extra file for running `spb_density_perbead`, called `lista_frames.dat`, which contains a list of the model RMFs (and the corresponding Bayesian parameter RMFs) of the cluster we are interested in. It is automatically generated using the script in `scripts/density_perbead/job_density_perbead.sh`. See the `Parameters specific to density map` section in the config files for adjustable parameters.

The executable can then be run as (more cores will make this run faster):

```
mpirun -np 64 spb_density_perbead
```

The outputs are files `*.dx` corresponding to the densities of different proteins and domains. Also produced is a file `HM.dat`, that provides the value of the densities at half the maximum. The script `scripts/chimera/create_chimera_command_file_densities.sh` can be used to display the densities in Chimera.

Fit with FRET data. The models in a cluster are then assessed by their fit to FRET data in two ways. First, the average FRET_R value from the models is compared against the average FRET_R value from experiment, assessing whether the model FRET_R value fits the experimental FRET_R value within the experimental error. Second, the distribution of FRET_R values from the models is compared against the distribution of raw experimental values.

The inputs for calculating FRET fit are files `fret_exp.dat`, the file containing FRET averages and standard deviations from experiment, and `rawdata_all_date.csv`, containing the raw FRET values from experiment. These two files are in `inputs/fretfit`. Additionally, the file `cluster_traj_score_weight.dat` from the earlier clustering step is used.

The scripts are run as below, assuming `CLUSTER_NUMBER` is the number of the cluster we are interested in, `ANALYSIS` is the directory containing the output of `spb_analysis`, and `SUFFIX` is the name of the output file.

```
python plot_FRETR_summary.py CLUSTER_NUMBER
cluster_traj_score_weight.dat ANALYSIS
fret_exp.dat SUFFIX
python plot_FRETR_distribution.py CLUSTER_NUMBER
cluster_traj_score_weight.dat
ANALYSIS fret_exp.dat
rawdata_all_date.csv SUFFIX
```

The summary and distribution scripts produce PDF output files.

Conclusion

Integrative structure determination is a powerful approach for obtaining 3D structures of systems that are refractive to single methods alone. The IMP software provides a range of tools to build protocols for integrative modeling, together with some ready-made protocols for common integrative modeling problems. A wide variety of biological systems have already been characterized with the aid of IMP (https://integrativemodeling.org/systems/). The computational methods used in these applications are similar to those demonstrated here.

Publication of macromolecular structures includes deposition of coordinates and X-ray scattering factors,[67] NMR restraints,[90] and EM particle images.[64] However, the complete modeling protocol for integrative modeling is still rarely available in a usable form,[91–93] making reproduction and use of the published results laborious or even impossible. To address this issue and others, the wwPDB established a Hybrid/Integrative Methods Task Force.[94] Following the Task Force recommendations, the mmCIF file format used to archive PDB structures has been extended to support the deposition of integrative models, together with information on the input data used and the modeling protocol. A prototype archive, PDB-Development (PDB-Dev, https://pdb-dev.wwpdb.org/)[42] currently contains three example mmCIF integrative models generated by IMP.[22,45,46] A wide variety of researchers stand to benefit from this archive. For example, experimental labs will be able to use a deposited model to plan experiments by simulating potential benefits gained from new data. Computational groups will more easily experiment with new scoring, sampling, and analysis methods, without having to reimplement the existing methods from scratch. Finally, the authors themselves will maximize the impact of their work, increasing the odds that their results are incorporated into future modeling.

Acknowledgments

We are grateful to all members of our research group. This tutorial is based partly on our prior reviews.[26,34]

Conflict of interest

The authors declare no competing financial interests.

References

1. Schmeing TM, Ramakrishnan V (2009) What recent ribosome structures have revealed about the mechanism of translation. Nature 461:1234–1242.
2. Mitra K, Frank J (2006) Ribosome dynamics: insights from atomic structure modeling into cryo-electron microscopy maps. Annu Rev Biophys Biomol Struct 35: 299–317.
3. Robinson C, Sali A, Baumeister W (2007) The molecular sociology of the cell. Nature 450:973–982.
4. Sali A, Glaeser R, Earnest T, Baumeister W (2003) From words to literature in structural proteomics. Nature 422:216–225.
5. Blundell T, Johnson L (1976) Protein Crystallography. New York: Academic Press.
6. Stahlberg H, Walz T (2008) Molecular electron microscopy: state of the art and current challenges. ACS Chem Biol 3:268–281.
7. Chiu W, Baker ML, Jiang W, Dougherty M, Schmid MF (2005) Electron cryomicroscopy of biological machines at subnanometer resolution. Structure 13: 363–372.
8. Lucic V, Leis A, Baumeister W (2008) Cryo-electron tomography of cells: connecting structure and function. Histochem Cell Biol 130:185–196.
9. Parrish JR, Gulyas KD, Finley RL Jr. (2006) Yeast two-hybrid contributions to interactome mapping. Curr Opin Biotechnol 17:387–393.
10. Fernandez-Martinez J, Phillips J, Sekedat M, Diaz-Avalos R, Velazquez-Muriel J, Franke J, Williams R, Stokes D, Chait B, Sali A, Rout M (2012) Structure-function map for a heptameric component of the nuclear pore complex. J Cell Biol 196:419–434.
11. Gingras AC, Gstaiger M, Raught B, Aebersold R (2007) Analysis of protein complexes using mass spectrometry. Nat Rev Mol Cell Biol 8:645–654.
12. Ward A, Sali A, Wilson I (2013) Integrative structural biology. Science 339:913–915.
13. Lasker K, Forster F, Bohn S, Walzthoeni T, Villa E, Unverdorben P, Beck F, Aebersold R, Sali A, Baumeister W (2012) Molecular architecture of the 26S proteasome holocomplex determined by an integrative approach. Proc Natl Acad Sci USA 109:1380–1387.
14. Simon B, Madl T, Mackereth CD, Nilges M, Sattler M (2010) An efficient protocol for NMR-spectroscopy-based structure determination of protein complexes in solution. Angewandte Chem 49:1967–1970.
15. Bau D, Sanyal A, Lajoie BR, Capriotti E, Byron M, Lawrence JB, Dekker J, Marti-Renom MA (2011) The three-dimensional folding of the alpha-globin gene domain reveals formation of chromatin globules. Nat Struct Mol Biol 18:107–114.
16. Molnar K, Bonomi M, Pellarin R, Clinthorne G, Gonzalez G, Goldberg S, Goulian M, Sali A, DeGrado W (2014) Cys-scanning disulfide crosslinking and Bayesian modeling probe the transmembrane signaling mechanism of the histidine kinase, PhoQ. Structure 22:1239–1251.
17. Alber F, Dokudovskaya S, Veenhoff L, Zhang W, Kipper J, Devos D, Suprapto A, Karni-Schmidt O, Williams R, Chait B, Sali A, Rout M (2007) The molecular architecture of the nuclear pore complex. Nature 450:695–701.
18. Alber F, Dokudovskaya S, Veenhoff L, Zhang W, Kipper J, Devos D, Suprapto A, Karni-Schmidt O, Williams R, Chait B, Rout M, Sali A (2007) Determining the architectures of macromolecular assemblies. Nature 450:683–694.
19. Wente SR, Rout MP (2010) The nuclear pore complex and nuclear transport. Cold Spring Harb Perspect Biol 2:a000562.
20. Devos D, Dokudovskaya S, Williams R, Alber F, Eswar N, Chait BT, Rout MP, Sali A (2006) Simple fold composition and modular architecture of the nuclear pore complex. Proc Natl Acad Sci USA 103:2172–2177.
21. DeGrasse JA, DuBois KN, Devos D, Siegel TN, Sali A, Field MC, Rout MP, Chait BT (2009) The establishment of nuclear pore complex architecture occurred early in evolution. Mol Cell Proteom 8:2119–2130.
22. Shi Y, Fernandez-Martinez J, Tjioe E, Pellarin R, Kim SJ, Williams R, Schneidman D, Sali A, Rout M, Chait B (2014) Structural characterization by cross-linking reveals the detailed architecture of a coatomer-related heptameric module from the nuclear pore complex. Mol Cell Proteom 13:2927–2943.
23. Upla P, Kim SJ, Sampathkumar P, Dutta K, Cahill SM, Chemmama IE, Williams R, Bonanno JB, Rice WJ, Stokes DL, Cowburn D, Almo SC, Sali A, Rout MP, Fernandez-Martinez J (2017) Molecular architecture of the major membrane ring component of the nuclear pore complex. Structure 25:434–445.
24. Fernandez-Martinez J, Kim SJ, Shi Y, Upla P, Pellarin R, Gagnon M, Chemmama IE, Wang J, Nudelman I, Zhang W, Williams R, Rice WJ, Stokes DL, Zenklusen D, Chait BT, Sali A, Rout MP (2016) Structure and function of the nuclear pore complex cytoplasmic mRNA export platform. Cell 167:1215–1228.
25. Alber F, Chait BT, Rout MP, Sali A, Integrative structure determination of protein assemblies by satisfaction of spatial restraints. In: Panchenko A, Przytycka T, Eds. (2008) Protein-protein interactions and networks: identification, characterization and prediction. London, UK: Springer-Verlag, pp 99–114.
26. Russel D, Lasker K, Webb B, Velazquez-Muriel J, Tjioe E, Schneidman-Duhovny D, Peterson B, Sali A (2012) Putting the pieces together: integrative structure determination of macromolecular assemblies. PLoS Biol 10:e1001244.
27. Sippl MJ (1990) Calculation of conformational ensembles from potentials of mean force. An approach to the knowledge-based prediction of local structures in globular proteins. J Mol Biol 213:859–883.
28. Shen MY, Sali A (2006) Statistical potential for assessment and prediction of protein structures. Protein Sci 15:2507–2524.
29. Brooks BR, Brooks CL, 3rd, Mackerell AD, Jr., Nilsson L, Petrella RJ, Roux B, Won Y, Archontis G, Bartels C, Boresch S, Caflisch A, Caves L, Cui Q, Dinner AR, Feig M, Fischer S, Gao J, Hodoscek M, Im W, Kuczera K, Lazaridis T, Ma J, Ovchinnikov V, Paci E, Pastor RW, Post CB, Pu JZ, Schaefer M, Tidor B, Venable RM, Woodcock HL, Wu X, Yang W, York DM, Karplus M (2009) CHARMM: the biomolecular simulation program. J Comput Chem 30:1545–1614.
30. Metropolis N, Ulam S (1949) The Monte Carlo method. J Am Statist Assoc 44:335–341.
31. Lasker K, Topf M, Sali A, Wolfson H (2009) Inferential optimization for simultaneous fitting of multiple components into a cryoEM map of their assembly. J Mol Biol 388:180–194.
32. Lasker K, Phillips JL, Russel D, Velazquez-Muriel J, Schneidman-Duhovny D, Webb B, Schlessinger A, Sali

A (2010) Integrative structure modeling of macromolecular assemblies from proteomics data. Mol Cell Proteom 9:1689–1702.

33. Russel D, Lasker K, Phillips J, Schneidman-Duhovny D, Velazquez-Muriel J, Sali A (2009) The structural dynamics of macromolecular processes. Curr Opin Cell Biol 21:97–108.
34. Webb B, Lasker K, Velazquez-Muriel J, Schneidman-Duhovny D, Pellarin R, Bonomi M, Greenberg C, Raveh B, Tjioe E, Russel D, Sali A, Modeling of proteins and their assemblies with the integrative modeling platform. In: Chen Y, Ed. (2014) Methods in molecular biology. London, UK: Humana Press, pp 277–295.
35. Schneidman-Duhovny D, Hammel M, Sali A (2011) Macromolecular docking restrained by a small angle X-ray scattering profile. J Struct Biol 73:461–471.
36. Alber F, Forster F, Korkin D, Topf M, Sali A (2008) Integrating diverse data for structure determination of macromolecular assemblies. Annu Rev Biochem 77: 443–477.
37. Lasker K, Sali A, Wolfson HJ (2010) Determining macromolecular assembly structures by molecular docking and fitting into an electron density map. Proteins 78: 3205–3211.
38. Sali A, Blundell TL (1993) Comparative protein modelling by satisfaction of spatial restraints. J Mol Biol 234:779–815.
39. Erzberger J, Stengel F, Pellarin R, Zhang S, Schaefer T, Aylett C, Cimermancic P, Boehringer D, Sali A, Aebersold R, Ban N (2014) Molecular architecture of the 40S•eIF1•eIF3 translation initiation complex. Cell 158:1125–1135.
40. Saltzberg DJ, Broughton HB, Pellarin R, Chalmers MJ, Espada A, Dodge JA, Pascal BD, Griffin PR, Humblet C, Sali A, (2016) A residue resolved Bayesian approach to quantitative interpretation of hydrogen deuterium exchange from mass spectrometry: application to characterizing protein-ligand interactions. J Phys Chem B 121:3493–3501.
41. Bonomi M, Muller EG, Pellarin R, Kim SJ, Russel D, Ramsden R, Sundin BA, Davis TA, Sali A (2014) Determining protein complex structures based on a Bayesian model of in vivo FRET data. Mol Cell Proteom 13: 2812–2823.
42. Burley SK, Kurisu G, Markley JL, Nakamura H, Velankar S, Berman HM, Sali A, Schwede T, Trewhella J (2017) PDB-Dev: a prototype system for depositing integrative/hybrid structural models. Structure 25: 1317–1318.
43. Wang X, Cimermancic P, Yu C, Sakata E, Guo X, Greenberg C, Schweitzer A, Huszagh AS, Yang Y, Novitsky EJ, Leitner A, Nanni P, Kahraman A, Dixon J, Rychnovsky SD, Aebersold R, Baumeister W, Sali A, Huang L (2017) Molecular details underlying dynamic structures and regulation of the human 26S proteasome. Mol Cell Proteom 16:840–854.
44. Chen ZA, Pellarin R, Fischer L, Sali A, Nilges M, Barlow PN, Rappsilber J (2016) Structure of complement C3(H2O) revealed by quantitative cross-linking/ mass spectrometry and modelling. Mol Cell Proteom 15:2730–2743.
45. Shi Y, Pellarin R, Fridy P, Fernandez-Martinez J, Thompson M, Li Y, Wang QJ, Sali A, Rout M, Chait B (2015) A strategy for dissecting the architectures of native macromolecular assemblies. Nat Methods 12: 1135–1118.
46. Robinson P, Trnka M, Pellarin R, Greenberg C, Bushnell D, Davis R, Burlingame A, Sali A, Kornberg R (2015) Molecular architecture of the yeast Mediator complex. eLife 4:e08719.
47. Luo J, Cimermancic P, Viswanath S, Ebmeier C, Kim B, Dehecq M, Raman V, Greenberg C, Pellarin R, Sali A, Taatjes D, Hahn S, Ranish J (2015) Architecture of the human and yeast general transcription and DNA repair factor TFIIH. Mol Cell 59:794–806.
48. Algret R, Fernandez-Martinez J, Shi Y, Kim SJ, Pellarin R, Cimermancic P, Cochet E, Sali A, Chait B, Rout M, Dokudovskaya S (2014) Molecular architecture and function of the SEA complex—a modulator of the TORC1 pathway. Mol Cell Proteom 13:2855–2870.
49. Fan H, Schneidman D, Irwin JJ, Dong G, Shoichet B, Sali A (2011) Statistical potential for modeling and ranking protein-ligand interactions. J Chem Inf Model 51:3078–3092.
50. Spill Y, Kim SJ, Schneidman-Duhovny D, Russel D, Webb B, Sali A, Nilges M (2014) SAXS Merge: an automated statistical method to merge SAXS profiles using Gaussian processes. J Synchrotron Radiat 21:203–208.
51. Schneidman-Duhovny D, Hammel M, Tainer J, Sali A (2013) Accurate SAXS profile computation and its assessment by contrast variation experiments. Biophys J 105:962–974.
52. Schneidman D, Hammel M, Tainer J, Sali A (2016) FoXS, FoXSDock, and MultiFoXS: Single-state and multi-state structural modeling of proteins and their complexes based on SAXS profiles. Nucleic Acids Res 44:W424–W429.
53. Schneidman-Duhovny D, Hammel M, Sali A (2010) FoXS: a web server for rapid computation and fitting of SAXS profiles. Nucleic Acids Res 38:541–544.
54. Pettersen EF, Goddard TD, Huang CC, Couch GS, Greenblatt DM, Meng EC, Ferrin TE (2004) UCSF Chimera—a visualization system for exploratory research and analysis. J Comput Chem 25:1605–1612.
55. Taylor DJ, Devkota B, Huang AD, Topf M, Eswar N, Sali A, Harvey SC, Frank J (2009) Comprehensive molecular structure of the eukaryotic ribosome. Structure 17:1591–1604.
56. Serysheva I, Ludtke S, Baker M, Cong Y, Topf M, Eramian D, Sali A, Hamilton S, Chiu W (2008) Subnanometer-resolution electron cryomicroscopy-based domain models for the cytoplasmic region of skeletal muscle RyR channel. Proc Natl Acad Sci USA 105: 9610–9615.
57. Krukenberg K, Forster F, Rice L, Sali A, Agard D (2008) Multiple conformations of E. coli Hsp90 in solution: insights into the conformational dynamics of Hsp90. Structure 16:755–765.
58. Cong Y, Topf M, Sali A, Matsudaira P, Dougherty M, Chiu W, Schmid M (2008) Crystallographic conformers of actin in a biologically active bundle of filaments. J Mol Biol 375:331–336.
59. Tjong H, Li W, Kalhor R, Dai C, Hao S, Gong K, Zhou Y, Li H, Zhou XJ, Le Gros MA, Larabell CA, Chen L, Alber F (2016) Population-based 3D genome structure analysis reveals driving forces in spatial genome organization. Proc Natl Acad Sci USA 113:E1663–E1672.
60. Cohen SL, Chait BT (2001) Mass spectrometry as a tool for protein crystallography. Annu Rev Biophys Biomol Struct 30:67–85.
61. Young MM, Tang N, Hempel JC, Oshiro CM, Taylor EW, Kuntz ID, Gibson BW, Dollinger G (2000) High throughput protein fold identification by using experimental constraints derived from intramolecular crosslinks and mass spectrometry. Proc Natl Acad Sci USA 97:5802–5806.

62. Trnka MJ, Baker PR, Robinson PJ, Burlingame AL, Chalkley RJ (2014) Matching cross-linked peptide spectra: only as good as the worse identification. Mol Cell Proteom 13:420–434.
63. Chen ZA, Jawhari A, Fischer L, Buchen C, Tahir S, Kamenski T, Rasmussen M, Lariviere L, Bukowski-Wills JC, Nilges M, Cramer P, Rappsilber J (2010) Architecture of the RNA polymerase II-TFIIF complex revealed by cross-linking and mass spectrometry. EMBO J 29:717–726.
64. Lawson CL, Baker ML, Best C, Bi C, Dougherty M, Feng P, van Ginkel G, Devkota B, Lagerstedt I, Ludtke SJ, Newman RH, Oldfield TJ, Rees I, Sahni G, Sala R, Velankar S, Warren J, Westbrook JD, Henrick K, Kleywegt GJ, Berman HM, Chiu W (2011) EMDataBank.org: unified data resource for CryoEM. Nucleic Acids Res 39:D456–D464.
65. Czeko E, Seizl M, Augsberger C, Mielke T, Cramer P (2011) Iwr1 directs RNA polymerase II nuclear import. Mol Cell 42:261–266.
66. Kawabata T (2008) Multiple subunit fitting into a low-resolution density map of a macromolecular complex using a gaussian mixture model. Biophys J 95:4643–4658.
67. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. Nucleic Acids Res 28:235–242.
68. Armache KJ, Mitterweger S, Meinhart A, Cramer P (2005) Structures of complete RNA polymerase II and its subcomplex, Rpb4/7. J Biol Chem 280:7131–7134.
69. Swendsen RH, Wang JS (1986) Replica Monte Carlo simulation of spin glasses. Phys Rev Lett 57:2607–2609.
70. MacQueen J, editor (1967). Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Vol. 1: Statistics. Berkeley, CA: University of California Press.
71. Schneidman-Duhovny D, Pellarin R, Sali A (2014) Uncertainty in integrative structural modeling. Curr Opin Struct Biol 28:96–104.
72. Viswanath S, Chemmama I, Cimermancic P, Sali A (2017) Assessing exhaustiveness of stochastic sampling for integrative modeling of macromolecular structures. Biophys J, Submitted.
73. Hassold T, Hunt P (2001) To err (meiotically) is human: the genesis of human aneuploidy. Nat Rev Genet 2: 280–291.
74. Conduit PT, Wainman A, Raff JW (2015) Centrosome function and assembly in animal cells. Nat Rev Mol Cell Biol 16:611–624.
75. Luders J, Stearns T (2007) Microtubule-organizing centres: a re-evaluation. Nat Rev Mol Cell Biol 8:161–167.
76. Petry S, Vale RD (2015) Microtubule nucleation at the centrosome and beyond. Nat Cell Biol 17:1089–1093.
77. Jaspersen SL, Winey M (2004) The budding yeast spindle pole body: structure, duplication, and function. Ann Rev Cell Develop Biol 20:1–28.
78. Burns S, Avena JS, Unruh JR, Yu Z, Smith SE, Slaughter BD, Winey M, Jaspersen SL (2015) Structured illumination with particle averaging reveals novel roles for yeast centrosome components during duplication. Elife 4:e08586.
79. Viswanath S, Bonomi M, Kim SJ, Klenchin VA, Taylor KC, Yabut KC, Umbreit NT, Van Epps HA, Meehl J, Jones MH, Russel D, Velazquez-Muriel JA, Winey M, Rayment I, Davis TN, Sali A, Muller EG (2017) The molecular architecture of the yeast spindle pole body core determined by Bayesian integrative modeling. Mol Biol Cell E17-06–0397.
80. Rieping W, Habeck M, Nilges M (2005) Inferential structure determination. Science 309:303–306.
81. Rout MP, Kilmartin JV (1990) Components of the yeast spindle and spindle pole body. J Cell Biol 111:1913–1927.
82. Adams IR, Kilmartin JV (1999) Localization of core spindle pole body (SPB) components during SPB duplication in Saccharomyces cerevisiae. J Cell Biol 145: 809–823.
83. Muller EG, Snydsman BE, Novik I, Hailey DW, Gestaut DR, Niemann CA, O'Toole ET, Giddings TH, Jr., Sundin BA, Davis TN (2005) The organization of the core proteins of the yeast spindle pole body. Mol Biol Cell 6:3341–3352.
84. Bullitt E, Rout MP, Kilmartin JV, Akey CW (1997) The yeast spindle pole body is assembled around a central crystal of Spc42p. Cell 89:1077–1086.
85. Elliott S, Knop M, Schlenstedt G, Schiebel E (1999) Spc29p is a component of the Spc110p subcomplex and is essential for spindle pole body duplication. Proc Natl Acad Sci USA 96:6205–6210.
86. Klenchin VA, Frye JJ, Jones MH, Winey M, Rayment I (2011) Structure-function analysis of the C-terminal domain of CNM67, a core component of the Saccharomyces cerevisiae spindle pole body. J Biol Chem 286: 18240–18350.
87. Sugita Y, Okamoto Y (1999) Replica-exchange molecular dynamics method for protein folding. Chem Phys Lett 314:141–151.
88. Bonomi M, Parrinello M (2010) Enhanced sampling in the well-tempered ensemble. Phys Rev Lett 104: 190601.
89. Daura X, van Gunsteren WF, Mark AE (1999) Folding-unfolding thermodynamics of a beta-heptapeptide from equilibrium simulations. Proteins 34:269–280.
90. Ulrich EL, Akutsu H, Doreleijers JF, Harano Y, Ioannidis YE, Lin J, Livny M, Mading S, Maziuk D, Miller Z, Nakatani E, Schulte CF, Tolmie DE, Kent Wenger R, Yao H, Markley JL (2008) BioMagResBank. Nucleic Acids Res 36:D402–D408.
91. Mesirov JP (2010) Computer science. Accessible reproducible research. Science 327:415–416.
92. Barnes N (2010) Publish your computer code: it is good enough. Nature 467:753.
93. Merali Z (2010) Computational science: ...error. Nature 467:775–777.
94. Sali A, Berman H, Schwede T, Trewhella J, Kleywegt G, Burley S, Markley J, Nakamura H, Adams P, Bonvin A, Chiu W, Dal Peraro M, Di Maio F, Ferrin T, Grunewald K, Gutmanas A, Henderson R, Hummer G, Iwasaki K, Johnson G, Lawson C, Meiler J, Marti-Renom M, Montelione G, Nilges M, Nussinov R, Patwardhan A, Rappsilber J, Read R, Saibil H, Schroder G, Schwieters C, Seidel C, Svergun D, Topf M, Ulrich E, Velanker S, Westbrook J (2015) Outcome of the first wwPDB hybrid/integrative methods task force workshop. Structure 23:1156–1167.