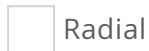


Supplying a user-defined bulk solvent map for refinement

Published Jun 17, 2026



DOI: 10.82153/2me0-hd96

Purpose

A significant portion of a macromolecular crystal is occupied by solvent. Since X-ray scattering arises from all atoms in a crystal, accurate solvent modeling is essential for reliable crystal structure refinement. In most crystallographic model-building protocols, water molecules of the first hydration shells are modeled as discrete, ordered atoms positioned in well-resolved electron density, and the remaining disordered solvent is approximated with a flat, binary mask. However, this approach has well-known shortcomings. Bulk solvent is not truly flat, the boundary between ordered and disordered solvent is nebulous, and regions near partially occupied protein, ligand, or solvent molecules are not appropriately accounted for, as these regions cannot fit cleanly into a binary bulk solvent. The inability to accurately model solvent remains one of the central unsolved challenges in macromolecular crystallography and is widely considered a major driver of the persistent gap between model and experiment. Alternative methods for generating bulk solvent masks include molecular dynamics simulations, 3D-RISM, holographic methods such as EDEN, and neural networks. But integrating these resulting models into standard refinement pipelines has remained a practical barrier. Here, we present a patch to Phenix refinement that allows users to supply their own bulk solvent map in place of the default flat mask. The modification is compact and applied locally to the user's Phenix installation, requiring no changes to the broader software infrastructure.

Background

Macromolecular crystals typically contain 20–90% solvent by volume, with an average of approximately 50% [1]. Since X-ray scattering arises from all atoms in

the crystal, solvent contributions must be accounted for to achieve accurate agreement between observed and calculated scattering intensities. Solvent molecules within the first hydration shells around the protein are often sufficiently ordered to be modeled explicitly as discrete water molecules [2]. Beyond these ordered shells, the remaining bulk solvent is disordered and cannot be modeled atomistically; instead, its scattering contribution must be approximated using a bulk-solvent model.

Most refinement programs account for disordered solvent using a flat, mask-based bulk solvent model [3][4]. In this approach, the region occupied by the atomic model is masked using a probe-and-shrink algorithm, and the remaining solvent-accessible volume is assigned a constant electron density. This implicitly assumes that disordered solvent is uniformly distributed throughout the non-masked region of the asymmetric unit.

However, this assumption imperfectly reflects physical reality. First, water molecules, ions, lipids or membrane components, and buffer components produce a structured density that a flat mask cannot represent [5][6]. Second, the binary mask only approximately models solvent at the boundary between explicitly modeled atoms and bulk solvent, especially around partially occupied atoms. In multiconformer models or with partially occupied solvent molecules [7][8][9][10], any atom with occupancy below 1 is fully excluded from the bulk solvent mask, regardless of its partial occupancy, thereby treating the unoccupied fraction as vacuum rather than solvent. The tempting remedy of adding additional low-occupancy waters to capture this density introduces its own problems, inflating the parameter count and risking instability in refinement. Third, there are also issues when the atomic model is incomplete. As model building is an iterative process, incorrectly assigning solvent density to unmodeled regions can produce smeared-out electron density with artificially reduced peak heights. This can be corrected with Polder maps [11], but these are typically not used in refinement programs. At high resolution, where the data are most sensitive to small errors, these approximations leave systematic signals in difference maps, including Fourier ripples introduced by the sharp edges of the mask boundary [12]. All of these contribute to the large gap observed between refinement R values and data quality [13].

The new mosaic bulk-solvent model [15] represents a meaningful advance over the flat model by treating spatially isolated solvent regions independently, each with its own occupancy, rather than imposing a single uniform density across the entire unit cell. However, it does not resolve partially occupied or partially modeled problems, leaving the unoccupied fraction as vacuum. More fundamentally, all mask-based approaches share the same underlying approximation of solvent as a structureless continuum defined by a binary boundary, rather than modeling its physical distribution. Alternative approaches, such as 3D-RISM [14] or MD-derived solvent densities from tools like AMBER [15], offer physically richer descriptions but remain difficult to integrate into Phenix [16]. While the optimal strategy for bulk-solvent representation remains an open question, integrating different models into refinement can help accelerate this research. Here we describe methods for incorporating alternative solvent representations into the Phenix refinement [16], with the goal of improving this systematically underserved component of macromolecular structure determination.

The flat-mask bulk-solvent model is an approximation of structured solvent

`phenix.refine` models the total structure factor as a scaled sum of a contribution from the atomic model (F_{calc}) and a contribution from bulk solvent (F_{mask}). The relative weight of each term is controlled by the refineable scale factors k_{total} and k_{mask} , which are optimized during refinement to minimize the discrepancy between observed and calculated intensities [16][17][18][19]:

$$F_{\text{model}} = k_{\text{total}} (F_{\text{calc}} + k_{\text{mask}} F_{\text{mask}})$$

The flat-mask model treats the crystallographic asymmetric unit as binary: areas with explicitly modeled molecules (protein, heteroatoms, or explicit solvent) and those without. Areas without explicitly modeled atoms are regions of constant electron density outside the macromolecule [4].

Other refinement programs, such as Buster [20], Refmac5 [21], SHELXL [22], PLATON [23], AMBER [24], and X-PLOR/CNS [4] allow a user to supply a precomputed solvent or partial-structure contribution [21]. In Refmac5, the `FPART` / `PHIPART` columns

allow the user to add up to five arbitrary partial structures, scaled to the data, alongside the model [21]. The patch described here closes that gap by allowing the user's solvent structure factors to serve as `F_mask`, while `k_mask` continues to be fit analytically to the supplied map.

A user-supplied bulk solvent map can replace the flat mask as the source of bulk-solvent structure factors

The user provides solvent structure factors as amplitude and phase columns in an MTZ file [25]. These can be derived from any method that predicts the solvent density distribution throughout the unit cell. For example, an explicit-solvent molecular dynamics simulation can be run with the crystal unit cell repeat as a periodic boundary, and the resulting solvent density can be averaged over trajectory frames and symmetry operations, then Fourier-transformed to yield structure factors [26][27]. 3D-RISM solves an integral equation theory of liquids on a crystallographic grid and outputs the solvent density directly, which can then be transformed into structure factors [14]. Holographic methods such as EDEN reconstruct the solvent electron density from experimental difference maps [28], producing a real-space density that can be converted into an MTZ file. Future methods will include neural network approaches that predict solvent density from the protein structure alone (<https://github.com/jmholton/cowcat>) and output a density map that can be Fourier-transformed in the same way.

The feature is provided as a patch to phenix.refine and invoked by adding bulk-solvent map parameters to an otherwise standard refinement command. When an MTZ file with appropriate column types is supplied, refinement proceeds as usual, just using the solvent contribution from the provided map rather than the default flat mask. Omitting these parameters reverts to standard behavior of using the default flat mask. If solvent amplitude and phase columns are already present in the experimental MTZ, no separate solvent file is needed. Two current limitations apply: joint X-ray/neutron refinement does not support the supplied map, and twin refinement has not been tested.

Input specification. A `refinement.input.bulk_solvent_map` PHIL scope accepts an `.mtz` file path and the labels of the columns to be read, via three parameters: `file_name`, `amplitudes_label` (default `FPART`), and `phases_label` (default `PHIPART`). The default column labels follow the Refmac convention, meaning that a solvent MTZ file

prepared for use with Refmac can typically be passed directly to Phenix without relabeling columns[21].

```
phenix.refine model.pdb data.mtz \  
  
refinement.input.bulk_solvent_map.file_name=solvent.mtz \  
  
refinement.input.bulk_solvent_map.amplitudes_label=Fpart \  
  
refinement.input.bulk_solvent_map.phases_label=PHIpart
```

Map ingestion and indexing. When a file is supplied, the amplitude and phase columns are combined into a complex Miller array (via `phase_transfer`, with phases in degrees) and reduced to the asymmetric unit. The array is then re-expressed onto the exact index set of the observed data: reflection indices are matched to those of `F_obs`, ensuring the two arrays share identical indices in the same order. Reflections present in `F_obs` but absent from the supplied map are assigned a zero solvent contribution. The aligned array is stored on the `f_model` manager and used in place of the mask-manager output for the remainder of the run.

Scaling and treatment during refinement. The supplied map replaces the flat mask solely as the source of the solvent structure factors. The `k_mask` scale factor is determined analytically for each resolution bin using the supplied map, and the overall scaling remains unchanged. Because `k_mask` still places the map within the existing scaling machinery, a user may supply it on an arbitrary scale. The map is held fixed throughout refinement and is not recomputed as the atomic model moves. No hybrid mode is provided: the supplied map fully replaces the internally computed mask rather than supplementing it.

Reflection outlier rejection. During scaling, `phenix.refine` removes outlier reflections several times, and each removal rebuilds the `f_model` manager. Absent any intervention, the rebuild discards the stored map, and the flat mask is recomputed from the atomic model, so `k_mask` is fit to the flat mask, and the

supplied map has no effect on the final result. The modification propagates the supplied map through the manager's `select()` step, including the in-place path used by outlier removal, so that the supplied solvent persists across the full refinement. This is required for the supplied map to take effect, and it is noted here for others attempting comparable modifications.

Installation. The modification is distributed as three patches applied to an existing Phenix installation. For Phenix 2.x releases, the relevant modules are located in `lib/python3.*/site-packages/` :

```
cd /programs/phenix-2.x-NNNN/lib/python3.*/site-packages
patch -p0 < ~/phenix_user_solvent/f_model.patch
patch -p0 < ~/phenix_user_solvent/phenix_refine.patch
patch -p0 < ~/phenix_user_solvent/__init__.params.patch
```

For Phenix 1.21 and earlier, which use a source-tree layout, the same patches apply under `modules/` (`modules/cctbx_project/mmtbx/f_model/f_model.py` , `modules/phenix/phenix/programs/phenix_refine.py` , and the corresponding `__init__.params`).

Of note, several files are named `phenix_refine.py` (for example, `phenix/command_line/phenix_refine.py`). Only `phenix/programs/phenix_refine.py` should be patched.

Modified source files

File	Change
<code>phenix/refinement/__init__.params</code>	New <code>bulk_solvent_map</code> PHIL scope (file name, amplitude/phase labels).
<code>phenix/programs/phenix_refine.py</code>	<code>inject_user_bulk_solvent()</code> helper, invoked from <code>validate()</code> once the <code>f_model</code> is built (single-refinement path).
<code>mmtbx/f_model/f_model.py</code>	<code>get_user_f_masks()</code> method; use of the stored masks in <code>update_xray_structure()</code> ; propagation through <code>select()</code> .

Default scaling is left in place and only the source of the solvent contribution changes.

Example of AMBER bulk solvent

As an example, we present a test case built on toxin II from the scorpion *Androctonus australis* Hector (PDB 1AHO), whose deposited data extend to 0.96 Å [29]. The refinement used 29,557 reflections over 0.96–15.8 Å, with 5% (1,444 reflections) reserved as a free set. The high-resolution, ambient-temperature data collection and nearly pure-water solvent make this a favorable system for probing the solvent model.

Generating the solvent map. As described in the example documentation, the supplied map was computed from an AMBER molecular dynamics trajectory that included 1aho crystal packing [15] and was then improved by density editing. In this editing, differences between F_o – F_c maps were converted into electron-density divots and added back into the solvent map. The ensemble model of the protein was optimized in the context of this solvent mask and explicit solvent using the AMBER 24 modern sander (msander; <https://github.com/Amber-MD/AmberClassic>) package, with X-ray data fit energy ramped upward while the simulation temperature was lowered, as in Mikhailovskii et al [24]. All maps and refinements were computed with the free-flagged reflections excluded, so the supplied map was never fit to the cross-validation set [30]. The repository includes the optimized solvent map (example/solvent_Fpart.mtz) to ensure reproducible refinement. Map generation by MD trajectory density editing is labor-intensive and not automated by this resource. For individual map generation, users will need to reproduce the pipeline separately (https://github.com/jmholton/bespoke_amber_restraints) and provide the amplitude and phase columns.

Effect on refinement. The two protocols supplied in the repository are identical except for the solvent source. Both start from the same multi-conformer ensemble model and experimental data and run three macro-cycles, refining individual sites and ADP ligands. Substituting the supplied flat mask with the MD-derived sander map lowered both R-factors:

Bulk-solvent source	R-work	R-free
Default flat mask	0.1278	0.1473
MD-derived map	0.0928	0.1092

The R-work improvement is, in part, expected, since the map was refined against working-set difference density. The free-flagged reflections, by contrast, were excluded from the difference-map editing, so the R-free improvement (0.147 to 0.109) is cross-validated and is a meaningful improvement [30]. Two further observations suggest that this reflects a better solvent model rather than overfitting. First, the model geometry was essentially unchanged between the two runs (bond r.m.s.d. of 0.007 Å in both), and the R-free/R-work gap remains small. Note that molprobity (Mpro)_[31] does not like 48 altlocs in an ASU. For this reason, msander and other 48-copy runs were done as a supercell ensemble (2x2x3 in P212121). That is, the deposited data and bulk solvent structure factors were re-indexed by 2h,2k,3l, the unit cell was re-defined to be 2x2x3 fold larger, and populated with 48 full-occupancy copies of the protein monomer. These supercell ensembles have the desirable property of resolving ambiguities in non-bonded interactions and are just as stable to refine in Phenix or Refmac as conventional single-ASU refinements. However, because supercell refinements are not currently widely adopted, the 48-member ensemble model here was condensed to a single ASU with 48 altloc letters for purposes of this demonstration.

A brief summary of how this solvent map was improved alongside the ensemble model is given in Table 1:

Table 1: Solvent models improve R-factors (1aho)			
Method of refinement	Rwork	Rfree	MolProbity
Refmac5: defaults, isotropic B	.1617	.1797	1.34
Flat bulk + add water & occupancy-refine	.1474	.1664	2.94
3D-RISM + occ-refine water	.1105	.1302	2.94
2-8 conformers/residue MD average solvent + occ-refine water	.1089	.1239	2.13
48-copy supercell refinement MD average solvent + ordered + watershed	.0938	.1181	1.66
+ map_touchup density editing	.0928	.1161	1.65
msander + refmac5 in supercell	.0812	.1107	1.06
phenix.refine 48-altloc, single-ASU + further density editing as described above	.0813	.1089	crash

We present these as the outcome of this single test case rather than as a general benchmark; the magnitude of any improvement will depend on resolution, solvent content, and the quality of the supplied map. Moreover, map editing contributed rather small gains in R-free relative to the ensemble modeling. A full table of results with different solvent and protein models is supplied below. Future analysis will benchmark this on a larger set of structures.

The patch, example data, and reference logs are openly available

The patches, documentation, the complete 1AHO example (input model and data, solvent map, and reference logs), and the commands needed to reproduce the results are available in a public repository:

https://github.com/jmholton/phenix_user_solvent. Users apply the patches to their own Phenix installation.

LLM Disclosure Statement

Writing: During the preparation of this work, the authors used Claude (Opus 4.8, Anthropic) to assist with drafting and editing. All AI-generated content was reviewed, edited, and verified by

the authors, who take full responsibility for the accuracy and integrity of the published work.

Code generation: Claude Code (Opus 4.8) was prompted to add this feature to the latest nightly build of the Phenix suite using the previous Refmac5 results as testing and validation requirements. Several rounds of debugging were required to isolate the minimal set of edits. This patch is intended as a feature request to the Phenix developers, but also serves as an example of how an end user can now use agentic AI to quickly and inexpensively add a desired feature to their own copy of an open-source software package.

Contributors (A-Z)

- **James Holton:** Conceptualization, Formal analysis, Investigation, Methodology, Software, Writing
- **Stephanie A Wankowicz:** Writing

References

1. Matthews B. (1968). Solvent content of protein crystals. [https://doi.org/10.1016/0022-2836\(68\)90205-2](https://doi.org/10.1016/0022-2836(68)90205-2)
2. Nittinger E, Schneider N, Lange G, Rarey M. (2015). Evidence of Water Molecules—A Statistical Evaluation of Water Molecules Based on Electron Density. <https://doi.org/10.1021/ci500662d>
3. Urzhumtsev A, Adams P, Afonine P. (2024). Universal parameters of bulk-solvent masks. <https://doi.org/10.1107/s2053273324000299>
4. Jiang J-S, Brünger AT. (1994). Protein Hydration Observed by X-ray Diffraction. <https://doi.org/10.1006/jmbi.1994.1633>
5. Afonine PV, Adams PD, Sobolev OV, Urzhumtsev AG. (2024). Accounting for nonuniformity of bulk solvent: A mosaic model. <https://doi.org/10.1002/pro.4909>
6. Lunin VY, Lunina NL, Ritter S, Frey I, Berg A, Diederichs K, Podjarny AD, Urzhumtsev A, Baumstark MW. (2001). Low-resolution data analysis for low-density lipoprotein particle. <https://doi.org/10.1107/s0907444900014608>

7. Flowers J, Echols N, Correy GJ, Jaishankar P, Togo T, Renslo AR, van den Bedem H, Fraser JS, Wankowicz SA. (2025). Expanding automated multiconformer ligand modeling to macrocycles and fragments. <https://doi.org/10.7554/elife.103797>
8. Wankowicz SA, Ravikumar A, Sharma S, Riley B, Raju A, Hogan DW, Flowers J, van den Bedem H, Keedy DA, Fraser JS. (2024). Automated multiconformer model building for X-ray crystallography and cryo-EM. <https://doi.org/10.7554/elife.90606>
9. Stachowski TR, Fischer M. (2023). FLEXR: automated multi-conformer model building using electron-density map sampling. <https://doi.org/10.1107/s2059798323002498>
10. (2023). Refinement of multiconformer ensemble models from multi-temperature X-ray diffraction data. <https://doi.org/10.1016/bs.mie.2023.06.009>
11. Liebschner D, Afonine PV, Moriarty NW, Poon BK, Sobolev OV, Terwilliger TC, Adams PD. (2017). Polder maps: improving OMIT maps by excluding bulk solvent. <https://doi.org/10.1107/s2059798316018210>
12. Fenn TD, Schnieders MJ, Brunger AT. (2010). A smooth and differentiable bulk-solvent model for macromolecular diffraction. <https://doi.org/10.1107/s0907444910031045>
13. Holton JM, Classen S, Frankel KA, Tainer JA. (2014). The R factor gap in macromolecular crystallography: an untapped potential for insights on accurate structures. <https://doi.org/10.1111/febs.12922>
14. Gray JG, Giambaşu GM, Case DA, Luchko T. (2022). Integral equation models for solvent in macromolecular crystals. <https://doi.org/10.1063/5.0070869>
15. Case DA, Cerutti DS, Cruzeiro VWD, Darden TA, Duke RE, Ghazimirsaeed M, Giambaşu GM, Giese TJ, Götz AW, Harris JA, Kasavajhala K, Lee T-S, Li Z, Lin C, Liu J, Miao Y, Salomon-Ferrrer R, Shen J, Snyder R, Swails J, Walker RC, Wang J, Wu X, Zeng J, Cheatham III TE, Roe DR, Roitberg A, Simmerling C, York DM, Nagan MC, Merz KM. (2025). Recent Developments in Amber Biomolecular Simulations. <https://doi.org/10.1021/acs.jcim.5c01063>
16. Afonine PV, Grosse-Kunstleve RW, Echols N, Headd JJ, Moriarty NW, Mustyakimov M, Terwilliger TC, Urzhumtsev A, Zwart PH, Adams PD. (2012). Towards automated crystallographic structure refinement with phenix.refine. <https://doi.org/10.1107/s0907444912001308>
17. Afonine PV, Grosse-Kunstleve RW, Adams PD, Urzhumtsev A. (2013). Bulk-solvent and overall scaling revisited: faster calculations, improved results. <https://doi.org/10.1107/s0907444913000462>

18. Afonine PV, Grosse-Kunstleve RW, Adams PD, Urzhumtsev A. (2023). Bulk-solvent and overall scaling revisited: faster calculations, improved results. Corrigendum. <https://doi.org/10.1107/s2059798323004825>
19. Liebschner D, Afonine PV, Baker ML, Bunkóczi G, Chen VB, Croll TI, Hintze B, Hung L-W, Jain S, McCoy AJ, Moriarty NW, Oeffner RD, Poon BK, Prisant MG, Read RJ, Richardson JS, Richardson DC, Sammito MD, Sobolev OV, Stockwell DH, Terwilliger TC, Urzhumtsev AG, Videau LL, Williams CJ, Adams PD. (2019). Macromolecular structure determination using X-rays, neutrons and electrons: recent developments in Phenix. <https://doi.org/10.1107/s2059798319011471>
20. Blanc E, Roversi P, Vonrhein C, Flensburg C, Lea SM, Bricogne G. (2004). Refinement of severely incomplete structures with maximum likelihood in BUSTER-TNT. <https://doi.org/10.1107/s0907444904016427>
21. Murshudov GN, Skubák P, Lebedev AA, Pannu NS, Steiner RA, Nicholls RA, Winn MD, Long F, Vagin AA. (2011). REFMAC5 for the refinement of macromolecular crystal structures. <https://doi.org/10.1107/s0907444911001314>
22. Sheldrick GM. (2015). Crystal structure refinement with SHELXL. <https://doi.org/10.1107/s2053229614024218>
23. Spek AL. (2015). PLATONSQUEEZE: a tool for the calculation of the disordered solvent contribution to the calculated structure factors. <https://doi.org/10.1107/s2053229614024929>
24. Mikhailovskii O, Izmailov SA, Xue Y, Case DA, Skrynnikov NR. (2023). X-ray Crystallography Module in MD Simulation Program Amber 2023. Refining the Models of Protein Crystals. <https://doi.org/10.1021/acs.jcim.3c01531>
25. Glykos NM. (2011). On the application of structure-specific bulk-solvent models. <https://doi.org/10.1107/s090744491102258x>
26. Wych DC, Aoto PC, Vu L, Wolff AM, Mobley DL, Fraser JS, Taylor SS, Wall ME. (2023). Molecular-dynamics simulation methods for macromolecular crystallography. <https://doi.org/10.1107/s2059798322011871>
27. Cerutti DS, Case DA. (2018). Molecular dynamics simulations of macromolecular crystals. <https://doi.org/10.1002/wcms.1402>
28. Szöke A, Szöke H, Somoza JR. (1997). Holographic Methods in X-ray Crystallography. V. Multiple Isomorphous Replacement, Multiple Anomalous Dispersion and Non-crystallographic Symmetry. <https://doi.org/10.1107/s010876739601584x>
29. Smith GD, Blessing RH, Ealick SE, Fontecilla-Camps JC, Hauptman HA, Housset D, Langs DA, Miller R. (1997). Ab Initio Structure Determination

and Refinement of a Scorpion Protein Toxin.

<https://doi.org/10.1107/s0907444997005386>

30. Brünger AT. (1992). Free R value: a novel statistical quantity for assessing the accuracy of crystal structures. <https://doi.org/10.1038/355472a0>
31. Williams CJ, Headd JJ, Moriarty NW, Prisant MG, Videau LL, Deis LN, Verma V, Keedy DA, Hintze BJ, Chen VB, Jain S, Lewis SM, Arendall WB, Snoeyink J, Adams PD, Lovell SC, Richardson JS, Richardson DC. (2017). MolProbity: More and better reference data for improved all atom structure validation. <https://doi.org/10.1002/pro.3330>