http://pubs.acs.org/journal/acsodf

Article

warpDOCK: Large-Scale Virtual Drug Discovery Using Cloud Infrastructure

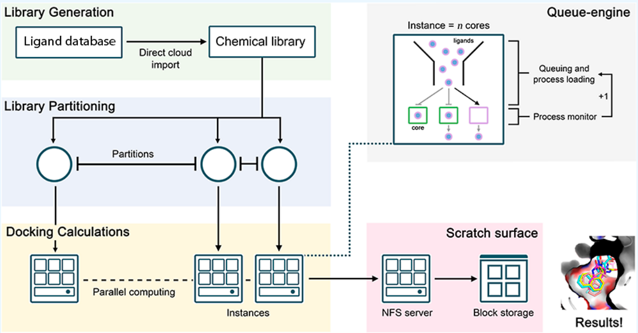
Daniel P. McDougal, Harinda Rajapaksha, Jordan L. Pederick, and John B. Bruning*

Cite This: *ACS Omega* 2023, 8, 29143−29149

Read Online

ACCESS | Metrics & More | Article Recommendations | SI Supporting Information

ABSTRACT: warpDOCK is an open-source pipeline for virtual small-molecule drug discovery using cloud infrastructure. warpDOCK is designed from the ground up for the Oracle Cloud Infrastructure (OCI), enabling harmonious parallelism of docking calculations over thousands to hundreds of thousands of cores. This enables cost-effective sampling of ultra-large chemical libraries, potentially reducing the time to identify lead drug candidates by orders of magnitude. By utilizing established docking software and automating each step of the process, warpDOCK makes large-scale virtual screening accessible to a broad user group. The warpDOCK code can be found at the BruningLab GitHub repository (https://github.com/BruningLab/warpDOCK).



■ INTRODUCTION

Recent advances in virtual screening programs and approaches have enabled the discovery of promising new drug candidates.[1−4] Traditionally, modest virtual screening experiments (up to 3−5 million compounds) have been performed using onsite high-performance computing clusters, typically requiring days to weeks of computation time.[5,6] Recently, an increasing number of published studies have performed large-scale virtual screens using chemical libraries of more than 100 million,[3,4,7] or even a billion ligands.[1] However, to perform such screens, access to considerable computational resources, experience in managing HPC clusters, and scripting knowledge are essential—a limiting factor for most prospective users. The benefit of upscaling CPU cores is clear when considering that the overall computation time (T) of a virtual screen is proportional to the number of ligands (N), processing time per ligand (P), and the number of cores (C) available

$$T \propto \frac{(P{\cdot}N)}{C} \qquad (1)$$

For example, on a modern desktop PC with 8 cores, it would take approximately 4 years to screen 100 million ligands, assuming a typical processing time of 10 s per ligand, which is not practical. Compared to onsite HPCs, cloud computing is advantageous as the user has access to virtual and bare-metal machines, enormous core scalability, and data storage. Cloud infrastructure is also globally accessible and does not require prior HPC experience. Therefore, incorporating cloud infrastructures with large-scale virtual screening is beneficial for the broader scientific community, enabling increased access and usability for this powerful drug discovery approach.

Here, we introduce warpDOCK, an open-source pipeline for virtual screening using cloud network infrastructure (Figure 1). To provide a smooth user experience and best performance, we designed warpDOCK for the Oracle Cloud Infrastructure (OCI) and automated tasks that would otherwise be complex and time-consuming with subsidiary programs (Figure 2; Table 1). Because of the design architecture of the queue-engine, warpDOCK offers practically limitless scaling capabilities whilst maintaining the maximum efficiency of docking calculations on all CPU cores across all compute instances, and is compatible with multiple different docking algorithms. As an exercise to demonstrate the capabilities and cost-effectiveness of the pipeline and to compute behavioral characteristics, we performed a virtual screen against AmpC β-lactamase (PDB: 1L2S) from *Escherichia coli* using 1.28 million ligands and a large-scale ensemble virtual screen with over 100 million docking calculations against *Staphylococcus aureus* D-alanine−D-alanine ligase (*Sa*Ddl; PDB: 7U9K). Collectively, these exercises demonstrate that the warpDOCK platform is accessible, cost-effective, and simple to use.

■ RESULTS AND DISCUSSION

In cloud environments, a computer ("shape" according to OCI terminology) is launched as an object called an "instance",

Received: April 5, 2023
Accepted: July 11, 2023
Published: July 31, 2023

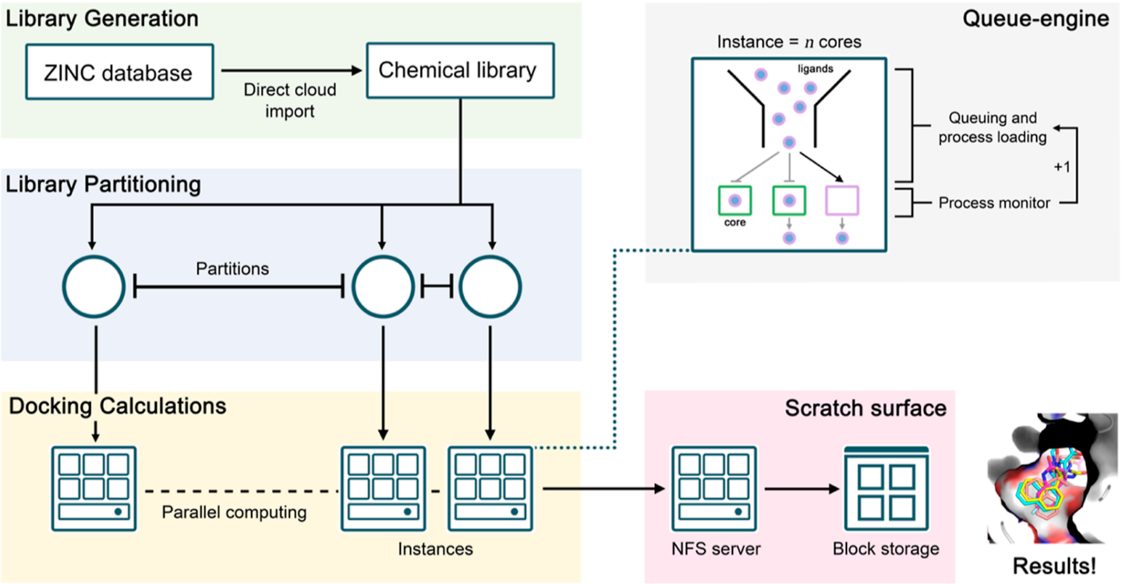


Figure 1. Overview of the warpDOCK pipeline. Chemical libraries are curated from the ZINC database in the PDBQT format (3D) and imported into cloud storage devices. The chemical library is partitioned into sub-folders proportional to the number of compute instances. Docking calculations are performed on compute instances and are managed by the queue engine. Results are funneled through the NFS server to buffer I/O operations and into block storage.

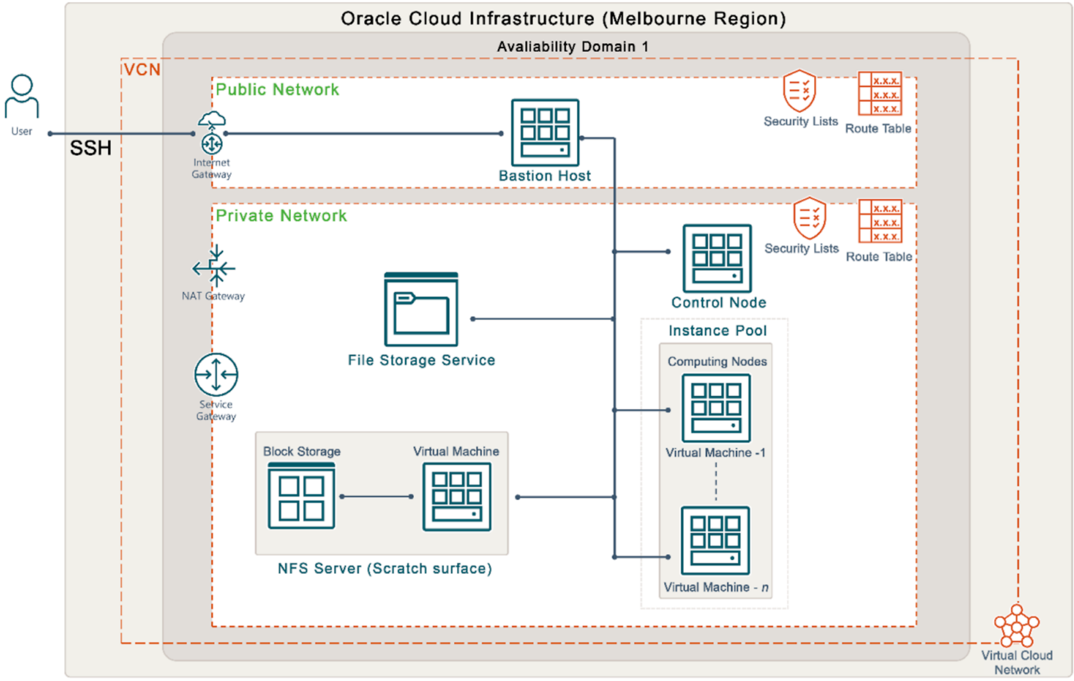


Figure 2. Schematic architecture of warpDOCK on the Oracle Cloud Infrastructure (OCI). The framework is hosted within the user's private-subnet of the VCN. To access the private-subnet, the user first uses SSH to access the Bastion login node in the public-subnet from a local machine with the private key file. The Bastion login node is used as a lily pad into the private-subnet using SSH and the private key file. In the private-subnet, all operations are managed from the Control node. In the private-subnet, compute instances are launched together as an instance pool. Each instance is installed with a custom image and has the same shape configuration. The custom image is cloned from a base canonical Ubuntu OS installed with warpDOCK, Qvina2, and dependencies. The custom image is permanently mounted to block storage devices via the NFS server. Thus, all operations can be managed harmoniously.

having varied core allocations and memory. Depending on the task, the number of instances could vary significantly, but this creates complexity as jobs must be managed harmoniously between cores and simultaneously between instances. Because the docking calculation time per ligand is not consistent, core utilization can drop if the next ligand is not pre-loaded. Rather than using BASH shell scripting to load ligands,[1] which is computationally expensive (thus affecting process loading time), we developed the warpDOCK queue-engine in python (https://www.python.org/) to circumvent this issue. Here, the queue-engine actively monitors the number of docking calculations allocated to each core, and if the values drop

Table 1. warpDOCK Programs

ZincDownloader	chemical library import
Splitter	.PDBQT file preparation
FileDivider	chemical library partitioning
FetchResults	retrieval of docking scores
ReDocking	binding pose retrieval, chemical library handling
WarpDrive	queue-engine
Conductor	network navigation

below a threshold, a new ligand is loaded to the queue from a pre-partitioned chemical library sub-folder. The number of compounds in the sub-folder is determined by partitioning the chemical library by the number of compute instances. The processing threshold (L) is determined by the number of cores multiplied by the scaling factor (S)

$$L = C \cdot S \qquad (2)$$

Therefore, if $C = 128$ and $S = 3$, a new ligand will be added to the queue if the number of processes drops below 384. Multiple parallel instances are controlled via SSH from a secure control node in the user's private-subnet. When the warpDOCK queue-engine is operating with over 250 physical CPUs (500 virtual CPU cores), many input and output requests can occur between the computing instances and the data storage device. This may push some processes onto the blocked queue on the operating system, resulting in idling of instances. To reduce this risk, data from the virtual screen is funneled through a customized networked file system server (NFS) with multi-path enabled uber-high performance block storage devices (Figure 2).

For the best possible trade-off between speed and accuracy, warpDOCK is intended to use Qvina2[8] but is also compatible with AutoDock Vina and other Vina-based docking algorithms. Chemical library handling and management of large results datasets are likely to be major hurdles for many prospective users. Hence, we have developed a series of programs to automate these tasks increasing usability (Table 1).

To test compatibility with different docking algorithms (Qvina2, Qvina-W, AutoDock Vina, Smina vinardo, and Vina-Carb), we performed virtual screens against two targets, AmpC β-lactamase and *Sa*Ddl, using a drug-like library of 1.28 million compounds and a modest allocation of 1024 AMD E4.Flex virtual CPUs (2048 virtual CPU cores) and 64 GB RAM/instance (Figure 3a). Each program was compatible with the warpDOCK pipeline. Qvina2 and Qvina-W were fastest and performed at similar speed for both proteins, each recording wall-clock times of approximately 80 min (which includes boot-up of instances and shutdown of queue)[8,9] However, in practice, Qvina-W is recommended only for docking very large search areas. In contrast to Qvina2 and Qvina-W, completion times for each of the other docking algorithms varied between protein. These differences are likely a product of the different search algorithms and scoring functions; e.g., Vina docked all compounds to *Sa*Ddl in ∼113 min versus ∼156 min for AmpC β-lactamase. Nonetheless, the warpDOCK pipeline efficiently handled all docking algorithms tested.

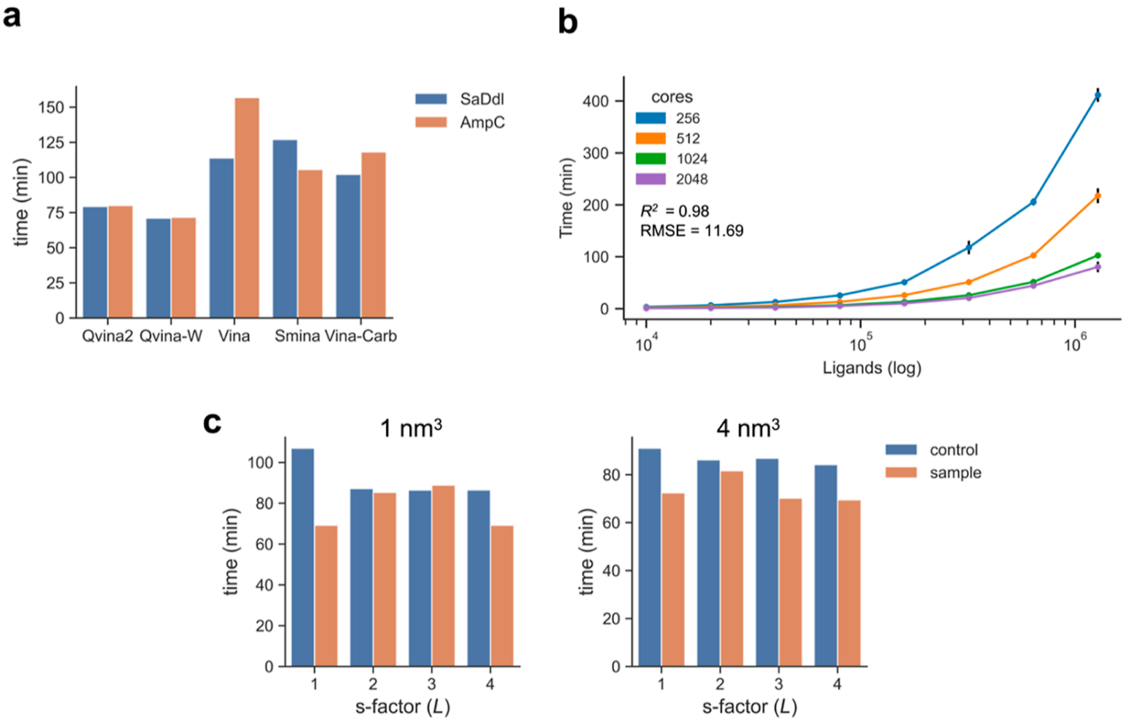


Figure 3. Performance characteristics of the queue-engine. (A) Recorded wall-clock time (minutes) for different dock algorithms to virtually screen 1.28 million drug-like ligands against the receptor targets *Sa*Ddl and AmpC β-lactamase. Each virtual screen was performed using 1024 VM.E4.Flex CPUs with 64 GB RAM/instance and $L = 3$. (B) Recorded wall-clock time (minutes) for 160 individual screening experiments against *Sa*Ddl ($L = 3$) in combinations of VM.E4.Flex CPUs (256, 512, 1024, and 2048 virtual CPUs) with 64 GB RAM/instance, search grid volume (1, 2, 4^3, 8^3, and 16 nm^3), and the number of drug-like ligands exponentially increasing from 10,000 to 1.28 million. Times are averaged over search grid volume; error bars represent the standard deviation for search grid volume. Slope fit using multiple non-linear regression (root mean square error (RMSE) = 11.69, $R^2 = 0.98$). (C) Comparison of recorded wall-clock times of 1.28 million drug-like ligands and an equally sized prototypical drug control library (Ampicillin) virtually screened against *Sa*Ddl for scaling factors $L = 1$ to 4 and search grid volumes of 1 nm^3 (left) and 4 nm^3 (right).

To test the scaling behavior of the pipeline on OCI, we performed virtual screens in combinations of exponentially increasing library sizes (Figure S2), cores (ergo instances), and search-grid volume using Qvina2 and recorded wall-clock times. This data fit well to a non-linear regression model (RMSE = 11.69, r^2 = 0.98), with computation time exponentially increasing and decreasing with the number of ligands and number of cores, respectively. For smaller libraries, a gain of diminishing returns was evident when comparing core speed-up, e.g., 1024 vs 2048 virtual CPUs (Figure 3b)—a trait of multicore processing (see Amdahl's law).[10] Additionally, varying the search grid volume does not majorly affect the overall computation time. For example, using 2048 virtual CPUs and L = 1, a library of 1280K randomly sampled ligands could be screened in 69.2 and 72.2 min for search-grid volumes of 1^3 and 4 nm^3, a four-fold increase in volume (Figure 3c). In comparisons between the randomly sampled library and an equally sized prototypical drug "control" library (Ampicillin), the queue-engine scaled similarly for the sample library over the control library (Figure 3d). Collectively, these data demonstrate the adaptability of the queue-engine and pipeline to respond efficiently to varied experimental parameters and provide predictable performance for each virtual screen on OCI.

Because computation times scale linearly with the number of CPUs, it is possible to estimate time and, therefore, compute costs. For example, using 1024 AMD VM.E4.Flex CPUs (2048 virtual CPUs) with 64 GB RAM and L = 3 and a search-grid of 4 nm^3 the cost to screen 640K ligands versus 1280K ligands against *Sa*Ddl was approximately $USD 18.94 and $USD 32.68 including boot-volumes, respectively. By fitting a slope to these data (for all points; R^2 = 0.99), the estimated costs to screen 10M ligands, 100M ligands, and 1B ligands using 1024 AMD VM.E4.Flex CPUs (2048 virtual CPUs) with 64 GB RAM are $USD 258.5, $USD 2580.88, and $USD 25,804.6. In practice, however, the number of CPUs used to screen 100M or 1B ligands would much greater and thus require less time, which may vary cost estimations due to Amdahl's law. It is important to also note that the costs discussed above are for compute hardware only, i.e., CPUs and memory, and instance boot volumes. Storage costs vary depending on capacity and are billed monthly. All costs were estimated at the time of manuscript preparation.

We also sought to benchmark and compare the performance of warpDOCK against the closest docking pipeline, VirtualFlow, which uses BASH job scheduling.[1] To compare performance, the 1.28 million benchmarking libraries were screened against AmpC β-lactamase with Qvina2 (exhaustiveness = 1) using both VirtualFlow and warpDOCK (L = 1). Docking calculations were performed using 1024 AMD VM.E4.Flex CPUs (2048 virtual CPU cores) and 64 GB RAM/ instance. warpDOCK completed all docking calculations in 80 min, compared to 295 min for VirtualFlow, a 3.7-fold increase in completion time. The compute costs for each pipeline for the two virtual screens also differed: the virtual screen compute costs for warpDOCK were $USD 35.45 versus $USD 131 for VirtualFlow. Thus, the warpDOCK queue-engine demonstrates the benefits of ground-up design for increasing the efficiency of docking calculations and reducing compute costs.

As a practical example to demonstrate capabilities and cost-effectiveness for users, we used the warpDOCK pipeline to perform two virtual screening exercises. First, we docked the 1.28 million benchmarking library against AmpC β-lactamase with Qvina2 using 1024 AMD VM.E4.Flex CPUs with 64 GB of RAM. A coarse- to fine-grain approach was used: in the first pass, all ligands were docked with an exhaustiveness of 1; then the top 1% were extracted and redocked with an exhaustiveness of 25. To help validate the screening approach and Qvina2 scoring function, a library of active inhibitors was also docked to the receptor (Figure S3). Comparative analysis of the docked co-crystallized ligand (binding affinity = −7.4 kcal/mol) to that of the crystal structure revealed strong conservation of binding orientation and molecular interactions. We also found that the top 0.25% of ligands from the virtual screen (n = 3200) were significantly enriched over the active inhibitors (n = 62) by Welch's unequal variances t-test (p < 0.0005). The compute costs for the AmpC β-lactamase virtual screen cost were approximately $USD 63, demonstrating efficiency and affordability.

Next, for the second virtual screening exercise, we performed a larger and more exhaustive ensemble virtual screen targeting the ATP and di-peptide binding sites of *Sa*Ddl, an antibiotic drug target essential for cell wall biosynthesis, using a library of 4.75 million "drug-like" ligands (Figure S4). The Ω-loop, which forms a peripheral shell surrounding the ATP and di-peptide binding sites, is dynamic in the absence of substrates and does not appear in the electron density of crystal structures.[11] As such, we performed three independent 250 ns all-atom molecular dynamics simulations with substrate-free *Sa*Ddl and used RMSD-based clustering to identify representative conformations (or centroids). Next, 21 of the top representative conformations of the protein (including the crystal structure) were screened, totaling ∼101 million docking calculations using a coarse-/fine-grain approach. There are currently no available inhibitors that target both the ATP and dipeptide binding sites of D-alanine—D-alanine ligase enzymes; therefore, it is challenging to generate decoys. Appropriate validation of docking calculations and interpretation of the results are important to mitigate the risk of false positives. As an alternative, we tested in silico whether a subset of the top clustered hits from the ensemble screen could stably bind to *Sa*Ddl over the duration of a 100 ns molecular dynamics (MD) simulation. We found that, using the poses docked to the crystal structure as starting coordinates, 2 of the 31 ligands dissociated after an initial equilibration period (Figure S3d,e). These data highlight that even when predicted binding affinities are calculated using exhaustive ensemble-based approaches, false positives may still be enriched. Therefore, users should be aware of and have controls in place to reduce risks. The virtual screen was performed using 1920 AMD VM.E4.Flex CPUs (3840 virtual CPU cores) with an approximate wall-clock time of 3.5 h per receptor conformation. At the time of manuscript preparation, the entire ensemble virtual screen compute costs were approximately $USD 3675.

■ CONCLUSIONS

Here, we show that the warpDOCK pipeline provides a powerful framework for virtual screening using cloud infrastructure which is both accessible and simple to use. Given the ever-expanding depth of ultra-large chemical libraries and protein structures and the recent successes of RoseTTAFold and Alphafold2,[12−16] large-scale virtual screening is more relevant than ever. Importantly, because of the nature of cloud computing, users can perform virtual screening experiments using as little or as many resources available. As the warpDOCK code is written in Python, we encourage the scientific community to contribute toward future development. All code, usage examples, and documentation can be found free and open source (*GNU General Public License v3.0*) at our GitHub repository (https://github.com/BruningLab/warpDOCK).

■ EXPERIMENTAL METHODS

warpDOCK Programs. Here, we provide a description of the warpDOCK modules and their functions. For instructions on how to set up a virtual cloud network (VCN) on OCI, install warpDOCK, and implement all programs, we have provided a detailed step-by-step tutorial, which can be found in the GitHub repository (https://github.com/BruningLab/warpDOCK).

ZincDownloader. Following curation of the desired chemical library in ZINC, the program *ZincDownloader* receives as input a text file containing raw URLs that link to compressed tranches which host all the multi-PDBQT ligand files. The contents of each URL are downloaded and decompressed in a two-step process directly onto the cloud storage devices. This process saves double handling and bypasses the need to download/upload from the user's local machine.

Splitter. Each multi-PDBQT file contains hundreds to thousands of individual ligand files. To extract each ligand file, *Splitter* reads each multi-PDBQT file in the directory and writes a new file for each ligand to a new directory.

FileDivider. In order to manage concurrency and queueing, the chemical library is partitioned into batches proportional to the number of instances.

FetchResults. *FetchResults* is used to read the log file of each ligand and write the calculated affinity (kcal/mol) to a CSV file.

ReDocking. To dock selected ligands with higher exhaustiveness, ReDocking is evoked. This module uses the CSV from *FetchResults* to select the top *n* ligands specified by the user and copies them to a new directory. Additionally, ReDocking can be used to retrieve top binding poses *en masse*.

WarpDrive. The queue-engine at the heart of the warpDOCK pipeline. To manage the concurrency of processes across multiple cores and instances, *WarpDrive* monitors active docking processes and loads a new ligand to the queue if the number of active processes drops below the processing threshold (as described in the main text) until the queue is complete.

Conductor. Designed to orchestrate the whole pipeline. To begin and manage docking calculations, arguments are parsed by the *Conductor,* a program which uses SSH connections to enter each instance within the private-subnet of the VCN. The parsed arguments are then used to evoke the *WarpDrive* queue-engine.

Docking Algorithm Compatibility, Scaling Behavior, and Benchmarking. To test the compatibility of the warpDOCK pipeline with other docking algorithms, we assembled a 1.28 million "drug-like" ligand library from ZINC (https://zinc.docking.org/) and virtually screened all ligands against two receptor targets: AmpC -lactamase and *Sa*Ddl. The docking algorithms used were AutoDock Vina (https://vina.scripps.edu/downloads/), Qvina2 and Qvina-W (https://qvina.github.io/), Smina vinardo (https://sourceforge.net/projects/smina/), and Vina-Carb (https://glycam.org/). Each virtual screen was performed using an exhaustiveness of 1 and $L = 3$; flexible side chain docking was not enabled.

To test the performance of the queue-engine on OCI, we assembled random libraries increasing exponentially from 10,000 to 1.28 million in size from the benchmarking library. First, we performed a series of virtual screens using the *Sa*Ddl active-site as the target (exhaustiveness = 1, scaling factor = 3) and changed parameters until all possible combinations were screened, e.g., 640K ligands, 512 cores, and a search-grid volume of 8 nm^3. The number of AMD VM.E4.Flex CPUs used was either 256, 512, 1024, or 2048, each with 64 GB of RAM per instance. The search-grid volume was either 1, 2, 4, 8, and 16 nm^3. For each screen, we recorded wall-clock times, which include, for instance, boot-up time, embarrassingly parallel operations, shutting down of the queue, and parallel processes. To derive a slope, we implemented multiple polynomial regressions using the *Sklearn* library in Python

$$y = \beta_0 + \beta_1 a + \beta_2 b + \beta_3 c + \beta_4 a^2 + \beta_5 ab + \beta_6 b^2 + \beta_7 bc + \beta_8 c^2 + \cdots + \beta_{19} c^3 \quad (3)$$

where y is the dependent variable (time in minutes), β_0 is the y intercept, β_n are the coefficients, and a, b, and c are the dependent variables of ligands, cores, and search-grid volume, respectively. The polynomial is of the 3rd degree. For a complete list of the polynomial features, coefficients, and intercepts at different degrees, please refer to the accompanying Supporting Information. Root-mean-squared-error and R^2 were calculated with the *Sklearn.metrics* library.

Second, to test characteristics of the scaling factor L, we compared the 1.28 million benchmarking library ("sample") to an equally sized "control" library, which was the ligand Ampicillin (SMILES: CC1(C(N2C(S1)C(C2=O)NC(=O)C(C3=CC=CC=C3)N)C(=O)O)C). We compared wall-clock times for L = 1 to 4 for search-grid volumes of 1 and 4 nm.^{3}

To benchmark the performance of warpDOCK against VirtualFlow, a pipeline which uses BASH job scheduling, we installed VirtualFlow in the private subnet and set up the pipeline using SLURM as the job scheduler.[1] The 1.28 million ligand benchmarking library was virtually screened against AmpC with Qvina2 using both warpDOCK and VirtualFlow on 1024 AMD VM.E4.Flex CPUs (2048 virtual CPU cores) with 64 GB RAM per instance. Wall-clock times were recorded as above.

Virtual Screening Exercise 1 (Mid-scale): AmpC *β*-Lactamase. The crystal structure of AmpC (PDB: 1L2S) was stripped of all small-molecules and water, and a search-grid was drawn around the substrate binding site. AmpC was virtually screened using a coarse-to-fine grain approach. First, the substrate binding site was screened using the 1.28 million benchmarking libraries with an exhaustiveness of 1 using Qvina2. Next, the top 1% scoring ligands (kcal/mol) were re-docked with an exhaustiveness of 25. To help validate the virtual screening method, a subset of active-lactamase inhibitors was also screened against the substrate binding site. Statistical analysis was performed using Welch's unequal variance *t*-test.

Virtual Screening Exercise 2 (Large Scale): *Sa*Ddl. *Atomistic Simulations.* Molecular dynamics simulations and RMSD-based clustering of wild-type *Sa*Ddl were performed using GROMACS (2020.4) with the Charmm-27 all-atom forcefield. The ATP and dipeptide-bound co-crystal structure of *Sa*Ddl (2 Å resolution, PDB: 7U9K, intact Ω-loop) was stripped of all waters, metals, and small molecules. Missing residues of other loops were modeled using the ICM-Pro Molsoft software suite (3.9-1b) and subsequently refined. Refined *Sa*Ddl was placed in a dodecahedral box at least 1.0 nm from the periodic edge boundary and solvated with the TIP3P water model. Na^+ and Cl^- were added until the net neutral charge of the system was achieved. The system was then energy minimized using the steepest-descent method until F_{max} < 1000 kJ/mol. We performed sequential 250 ps restrained *NVT* and *NPT* simulations to appropriately equilibrate the system with Particle Mesh Ewald electrostatics and Berendsen thermostat coupling, prior to three 250 ns unrestrained production simulations. All equilibration and simulation steps were performed at 300 K.

RMSD-Based Clustering. To generate an ensemble of probable structural conformations for ensemble docking calculations, we performed RMSD-based clustering of a single unrestrained trajectory using the GROMOS algorithm.[17] Coordinates were extracted per 10 ps across the trajectory, producing 25,001 simulated structures aligned by α-C atoms to remove translational and rotational variance. Representative structures were then clustered by pairwise RMSD calculations across all protein heavy atoms with a 0.15 nm cut-off, resulting in 122 clusters, with the top 20 representing ∼77% of the conformational ensemble. We defined a representative structure, or "centroid," from each cluster as the lowest (or middle) RMSD relative to all other structures in the cluster.

Ensemble Virtual Ligand Screening and Refinement with MD. Docking calculations were performed using a library of ∼4.75 million commercially available drug-like compounds curated from ZINC (https://zinc.docking.org/), of which all could be successfully docked. The library was designed such that compounds had a molecular weight of up to 450 Da and a Log *P* score of 5.0 or less. A search grid was created around the ATP and dipeptide binding sites, large enough to encompass conformational variance for both the crystal structure and ensemble structures. For both the crystal and ensemble VLS, we utilized a coarse-to-fine grain approach. Initially, ligands were screened against each structure with low exhaustiveness, equating to ∼101 million unique docking calculations in the first pass. For the ensemble VLS, binding affinities (in kcal/mol) were weighted per the contribution (%) of each cluster to the sampled conformational ensemble. Weighted binding affinities were then summed to provide a final binding affinity for each ligand. Results from the ensemble and crystal virtual screens were aligned by ligand ID and ranked according to the ensemble-weighted score. Next, using the top 0.01% of compounds from the ensemble screen, we repeated docking calculations with higher exhaustiveness (E = 25). To further the reduce dimensionality of the data, we introduced a binding affinity threshold of −10 kcal/mol along the *X* and *Y* axes, resulting in four quadrants. From these, we further analyzed ligands belonging to the 2nd quadrant (Q2), thus including ligands that have the best binding affinities for both the ensemble and the crystal structure. We then clustered each ligand according to pairwise Tanimoto coefficient distance (Tc = 0.3) using the Butina algorithm, an unsupervised statistical approach optimized for clustering based on chemical similarities.[18] Following this, the top-scoring ligands were extracted from each cluster and subsequently ranked. For in silico validation using molecular dynamics, we cherry-picked compounds based on binding affinity, the number of molecules belonging to each cluster, and favorable receptor interactions, resulting in an initial 31 compounds. The starting coordinates for each of the 31 ligands were those of the ligand docked to the crystal structure. The system was equilibrated as described above, and a single 100 ns unrestrained production simulation was performed for each ligand. To calculate relative solvation, we averaged ligand heavy-atom RMSDs across the trajectory per 10 ps and calculated RMSDs relative to the starting coordinates post-energy minimization.

Docking calculations were performed with Qvina2. Chemical similarity clustering of compounds was conducted using the RDkit Python library using in-house scripts (available on request).

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.3c02249.

> Data pertaining to the scaling behavioral characteristics of warpDOCK, benchmarking library molecular weight distributions, and data pertaining to each virtual screening exercise (AmpC and *Sa*Ddl) can be found in the associated Supporting Information (PDF)

■ AUTHOR INFORMATION

Corresponding Author

John B. Bruning − *Institute for Photonics and Advanced Sensing, (IPAS), School of Biological Sciences, The University of Adelaide, Adelaide, South Australia 5005, Australia;* orcid.org/0000-0002-6919-1824; Email: john.bruning@adelaide.edu.au

Authors

Daniel P. McDougal − *Institute for Photonics and Advanced Sensing, (IPAS), School of Biological Sciences, The University of Adelaide, Adelaide, South Australia 5005, Australia;* orcid.org/0000-0003-4499-6789

Harinda Rajapaksha − *Oracle for Research, Japan & Asia Pacific Region, Oracle Australia, Melbourne, Victoria 3000, Australia*

Jordan L. Pederick − *Institute for Photonics and Advanced Sensing, (IPAS), School of Biological Sciences, The University of Adelaide, Adelaide, South Australia 5005, Australia*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.3c02249

Author Contributions

D.P.M. conceptualized the study, wrote the code, performed computational experiments, analyzed the data, and prepared the manuscript. H.R. conceptualized the study, wrote the code, and performed computational experiments. J.L.P. contributed to the analysis of the data and the preparation of the manuscript. J.B.B. conceptualized and supervised the study. All authors contributed to the revision of the manuscript.

Notes

The authors declare no competing financial interest.

All code and supporting documentation are available for free and open source (*GNU General* Public License v3.0) at the Bruning Lab GitHub repository found at https://github.com/BruningLab/warpDOCK.

■ ACKNOWLEDGMENTS

We acknowledge the Kaurna people, the original custodians on whose land this study was conducted. We graciously thank Oracle for Research for providing computational resources and members of the Bruning Laboratory for thoughtful feedback regarding the preparation of the manuscript.

■ REFERENCES

(1) Gorgulla, C.; Boeszoermenyi, A.; Wang, Z.-F.; Fischer, P. D.; Coote, P. W.; Padmanabha Das, K. M.; Malets, Y. S.; Radchenko, D. S.; Moroz, Y. S.; Scott, D. A.; Fackeldey, K.; Hoffmann, M.; Iavniuk, I.; Wagner, G.; Arthanari, H. An open-source drug discovery platform enables ultra-large virtual screens. *Nature* **2020**, *580*, 663−668.

(2) Alon, A.; Lyu, J.; Braz, J. M.; Tummino, T. A.; Craik, V.; O'Meara, M. J.; Webb, C. M.; Radchenko, D. S.; Moroz, Y. S.; Huang, X.-P.; Liu, Y.; Roth, B. L.; Irwin, J. J.; Basbaum, A. I.; Shoichet, B. K.; Kruse, A. C. Structures of the $\sigma 2$ receptor enable docking for bioactive ligand discovery. *Nature* **2021**, *600*, 759−764.

(3) Stein, R. M.; Kang, H. J.; McCorvy, J. D.; Glatfelter, G. C.; Jones, A. J.; Che, T.; Slocum, S.; Huang, X.-P.; Savych, O.; Moroz, Y. S.; Stauch, B.; Johansson, L. C.; Cherezov, V.; Kenakin, T.; Irwin, J. J.; Shoichet, B. K.; Roth, B. L.; Dubocovich, M. L. Virtual discovery of melatonin receptor ligands to modulate circadian rhythms. *Nature* **2020**, *579*, 609−614.

(4) Zheng, Z.; Huang, X.-P.; Mangano, T. J.; Zou, R.; Chen, X.; Zaidi, S. A.; Roth, B. L.; Stevens, R. C.; Katritch, V. Structure-Based Discovery of New Antagonist and Biased Agonist Chemotypes for the Kappa Opioid Receptor. *J. Med. Chem.* **2017**, *60*, 3070−3081.

(5) Irwin, J. J.; Shoichet, B. K. Docking Screens for Novel Ligands Conferring New Biology. *J. Med. Chem.* **2016**, *59*, 4103−4120.

(6) Bender, B. J.; Gahbauer, S.; Luttens, A.; Lyu, J.; Webb, C. M.; Stein, R. M.; Fink, E. A.; Balius, T. E.; Carlsson, J.; Irwin, J. J.; Shoichet, B. K. A practical guide to large-scale docking. *Nat. Protoc.* **2021**, *16*, 4799−4832.

(7) Kaplan, A. L.; Confair, D. N.; Kim, K.; Barros-Álvarez, X.; Rodriguiz, R. M.; Yang, Y.; Kweon, O. S.; Che, T.; McCorvy, J. D.; Kamber, D. N.; Phelan, J. P.; Martins, L. C.; Pogorelov, V. M.; DiBerto, J. F.; Slocum, S. T.; Huang, X.-P.; Kumar, J. M.; Robertson, M. J.; Panova, O.; Seven, A. B.; Wetsel, A. Q.; Wetsel, W. C.; Irwin, J. J.; Skiniotis, G.; Shoichet, B. K.; Roth, B. L.; Ellman, J. A. Bespoke library docking for 5-HT2A receptor agonists with antidepressant activity. *Nature* **2022**, *610*, 582−591.

(8) Alhossary, A.; Handoko, S. D.; Mu, Y.; Kwoh, C.-K. Fast, accurate, and reliable molecular docking with QuickVina 2. *Bioinformatics* **2015**, *31*, 2214−2216.

(9) Hassan, N. M.; Alhossary, A. A.; Mu, Y.; Kwoh, C.-K. Protein-Ligand Blind Docking Using QuickVina-W With Inter-Process Spatio-Temporal Integration. *Sci. Rep.* **2017**, *7*, 15451.

(10) Gustafson, J. L. Reevaluating Amdahl's Law. *Commun. ACM* **1988**, *31*, 532−533.

(11) Pederick, J. L.; Thompson, A. P.; Bell, S. G.; Bruning, J. B. , d-Alanine−d-alanine ligase as a model for the activation of ATP-grasp enzymes by monovalent cations. *J. Biol. Chem.* **2020**, *295*, 7894−7904.

(12) Irwin, J. J.; Tang, K. G.; Young, J.; Dandarchuluun, C.; Wong, B. R.; Khurelbaatar, M.; Moroz, Y. S.; Mayfield, J.; Sayle, R. A. ZINC20—A Free Ultralarge-Scale Chemical Database for Ligand Discovery. *J. Chem. Inf. Model.* **2020**, *60*, 6065−6073.

(13) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; Bridgland, A.; Meyer, C.; Kohl, S. A. A.; Ballard, A. J.; Cowie, A.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Adler, J.; Back, T.; Petersen, S.; Reiman, D.; Clancy, E.; Zielinski, M.; Steinegger, M.; Pacholska, M.; Berghammer, T.; Bodenstein, S.; Silver, D.; Vinyals, O.; Senior, A. W.; Kavukcuoglu, K.; Kohli, P.; Hassabis, D. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583−589.

(14) Baek, M.; DiMaio, F.; Anishchenko, I.; Dauparas, J.; Ovchinnikov, S.; Lee, G. R.; Wang, J.; Cong, Q.; Kinch, L. N.; Schaeffer, R. D.; Millán, C.; Park, H.; Adams, C.; Glassman, C. R.; DeGiovanni, A.; Pereira, J. H.; Rodrigues, A. V.; van Dijk, A. A.; Ebrecht, A. C.; Opperman, D. J.; Sagmeister, T.; Buhlheller, C.; Pavkov-Keller, T.; Rathinaswamy, M. K.; Dalwadi, U.; Yip, C. K.; Burke, J. E.; Garcia, K. C.; Grishin, N. V.; Adams, P. D.; Read, R. J.; Baker, D. Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **2021**, *373*, 871−876.

(15) Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. The Protein Data Bank. *Nucleic Acids Res.* **2000**, *28*, 235−242.

(16) Congreve, M.; de Graaf, C.; Swain, N. A.; Tate, C. G. Impact of GPCR Structures on Drug Discovery. *Cell* **2020**, *181*, 81−91.

(17) Daura, X.; Gademann, K.; Jaun, B.; Seebach, D.; van Gunsteren, W. F.; Mark, A. E. Peptide Folding: When Simulation Meets Experiment. *Angew. Chem., Int. Ed.* **1999**, *38*, 236−240.

(18) Butina, D. Unsupervised Data Base Clustering Based on Daylight's Fingerprint and Tanimoto Similarity: A Fast and Automated Way To Cluster Small and Large Data Sets. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 747−750.