VirtualFlow 2.0 - The Next Generation Drug Discovery Platform Enabling Adaptive Screens of 69 Billion Molecules

Christoph Gorgulla[1,2,3,*], AkshatKumar Nigam[4,24], Matt Koop[9], Süleyman Selim Çınaroğlu[21], Christopher Secker[8,18], Mohammad Haddadnia[19], Abhishek Kumar[23], Yehor Malets[22], Alexander Hasson[20], Minkai Li[25], Ming Tang[26,27], Roni Levin-Konigsberg[4], Dmitry Radchenko[5,7], Aditya Kumar[10], Minko Gehev[12], Pierre-Yves Aquilanti[9], Henry Gabb[11], Amr Alhossary[13], Gerhard Wagner[1], Alán Aspuru-Guzik[14-17], Yurii S. Moroz[6,7], Konstantin Fackeldey[8,10], Haribabu Arthanari[1,3,*]

Early-stage drug discovery has been limited by initial hit identification and lead optimization and their associated costs (1). Ultra-large virtual screens (ULVSs), which involve the virtual evaluation of massive numbers of molecules to engage a macromolecular target, have the ability to significantly alleviate these problems, as was recently demonstrated in multiple studies (2–7). Despite their potential, ULVSs have so far only explored a tiny fraction of the chemical space and of available docking programs. Here, we present VirtualFlow 2.0, the next generation of the first open-source drug discovery platform dedicated to ultra-large virtual screenings. VirtualFlow 2.0 provides the REAL Space from Enamine containing 69 billion drug-like molecules in a "ready-to-dock" format, the largest library of its kind available to date. We provide an 18-dimensional matrix for intuitive exploration of the library through a web interface, where each dimension corresponds to a molecular property of the ligands. Additionally, VirtualFlow 2.0 supports multiple techniques that dramatically reduce computational costs, including a new method called Adaptive Target-Guided Virtual Screening (ATG-VS). By sampling a representative sparse version of the library, ATG-VS identifies the sections of the ultra-large chemical space that harbors the highest potential to engage the target site, leading to substantially reduced computational costs by up to a factor of 1000. In addition, VirtualFlow 2.0 supports the latest deep learning- and GPU-based docking methods, allowing further speed-ups by up to two orders of magnitude. VirtualFlow 2.0 supports 1500 unique docking methods providing target-specific and consensus docking options to increase accuracy and has the ability to screen new types of ligands (such as peptides) and target receptors (including RNA and DNA). Moreover, VirtualFlow 2.0 has many advanced new features, such as enhanced AI and cloud support. We demonstrate a perfectly linear scaling behavior up to 5.6 million CPUs in the AWS Cloud, a new global record for parallel cloud computing. Due to its open-source nature and versatility, we expect that VirtualFlow 2.0 will play a key role in the future of early-stage drug discovery.

Introduction

One of the major challenges in early-stage drug discovery is the identification and optimization of initial hit and lead compounds that target biological macromolecules. Traditionally, experimental high-throughput screens have been the primary method for discovering initial hit compounds (8). However, this traditional approach has several disadvantages, including high cost, considerable time requirements, and the need for a robust assay. The chemical space of small molecules suitable for drug discovery is estimated to contain over 10^{60} molecules (9). Experimental screens often only sample a minuscule fraction (typically hundreds of thousands of molecules) of the possible chemical space which is not conducive to challenging target sites such as protein-protein interactions or allosteric sites, which often have a rather flat binding surface. Ultra-large virtual screenings (ULVSs) have the potential to address many of the challenges faced in early-stage drug discovery. Indeed, these virtual screenings evaluate over 100 million ligands, enabling the sampling of a substantially larger portion of the chemical space of compounds suitable for drug discovery. ULVSs can be significantly more efficient in terms of cost and time compared to experimental high-throughput screens, enabling the discovery of highly potent hits and providing mechanistic insights into how the molecules interact with biological targets. Additionally, ULVSs also allow for the targeting of challenging target sites (e.g. flat surfaces often found in protein-protein interfaces) due to their ability to find exceptionally tight-fitting molecules.

In the past few years, several studies have demonstrated the extraordinary power of ULVSs in identifying potent hits for diverse sets of target proteins, including enzymatic active sites, orthosteric sites of GPCRs, and protein-protein interactions. In 2019, Lyu and colleagues (2) targeted the D4 dopamine receptor and AmpC using ULVSs and obtained picomolar binders directly from the screen. In 2020, the melatonin receptor MT1 was successfully targeted with ULVS, resulting in compounds with sub-nanomolar potency (4). In the same year, ULVSs were used for the first time to target protein–protein interactions. That

[1]Department of Biological Chemistry and Molecular Pharmacology, Harvard Medical School, Harvard University, Boston, USA. [2]Department of Physics, Faculty of Arts and Sciences, Harvard University, Cambridge, USA. [3]Department of Cancer Biology, Dana-Farber Cancer Institute, Boston, USA. [4]Department of Computer Science, Stanford University, Stanford, USA. [5]Enamine Ltd, Kyiv, Ukraine. [6]Chemspace LLC, Kyiv, Ukraine. [7]Taras Shevchenko National University of Kyiv, Kyiv, Ukraine. [8]Zuse Institute Berlin (ZIB), Berlin, Germany. [9]Amazon Web Services, Seattle, USA. [10]Institute of Mathematics, Technical University Berlin, Berlin, Germany. [11]Intel, Santa Clara, USA. [12]Google, Mountain View, USA. [13]Wesleyan University, Middletown, USA. [14]Department of Computer Science, University of Toronto, Toronto, Ontario, Canada. [15]Vector Institute for Artificial Intelligence, Toronto, Ontario, Canada. [16]Department of Chemistry, University of Toronto, Toronto, Ontario, Canada. [17]Canadian Institute for Advanced Research, Toronto, Ontario, Canada. [18]Max Delbrück Center for Molecular Medicine, Berlin, Germany. [19]Department of Molecular Genetics, University of Toronto, Toronto, Canada. [20]Medical Sciences Divisional Office, University of Oxford, John Radcliffe Hospital, Oxford, UK. [21]Department of Biochemistry, University of Oxford, UK. [22]V. P. Kukhar Institute of Bioorganic Chemistry and Petrochemistry, National Academy of Science of Ukraine, Kyiv, Ukraine. [23]Department of Biosciences and Bioengineering, Indian Institute of Technology Guwahati, India. [24]Department of Genetics, Stanford University, Stanford, USA. [25]Harvard College, Cambridge, USA. [26]The University of Queensland, Frazer Institute, Faculty of Medicine at the Translational Research Institute Australia, Brisbane, Australia. [27]Queensland University of Technology, School of Biomedical Sciences, Centre for Genomics and Personalised Health at the Translational Research Institute Australia, Brisbane, Australia.
*email: c.gorgulla@gmail.com, hari@hms.harvard.edu

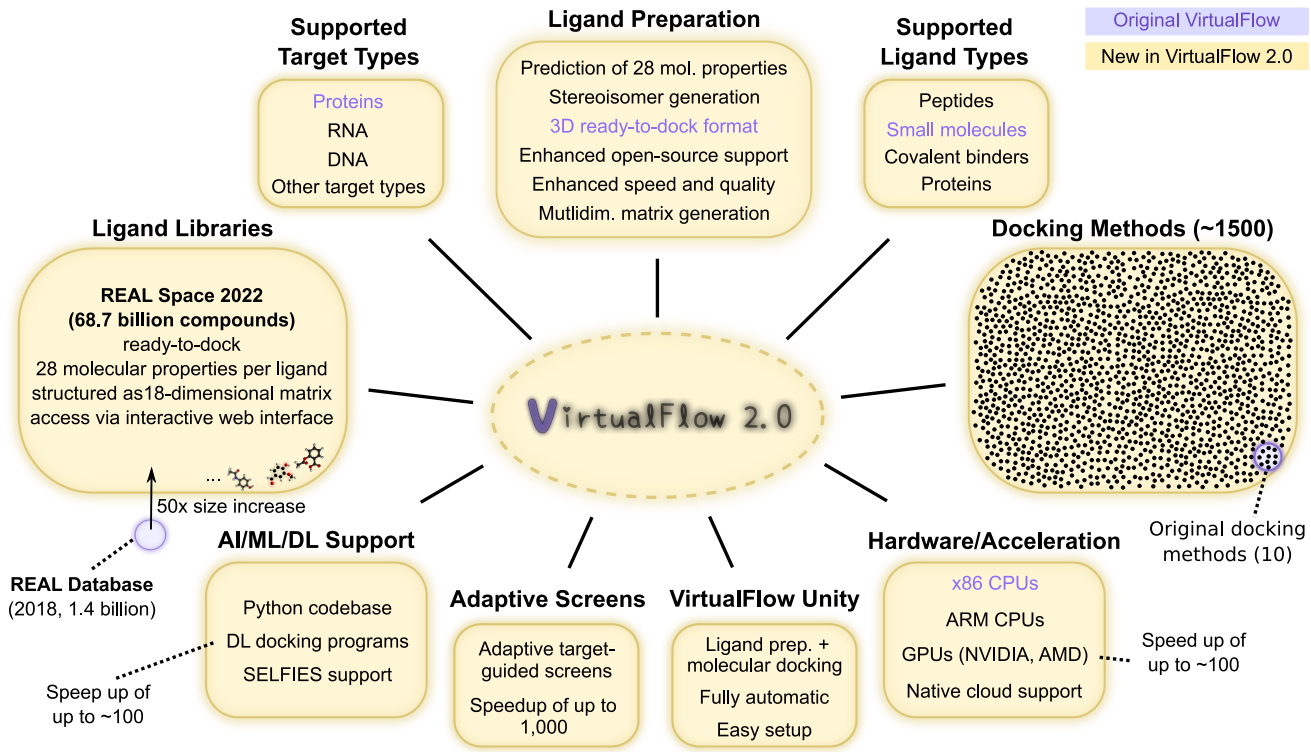


Fig. 1. Overview of VirtualFlow 2.0 (gold) and comparison to the original VirtualFlow (violet). Most features of the original VirtualFlow version are still supported by VirtualFlow 2.0, and thus VirtualFlow can be seen as a major extension of the original VirtualFlow. Among the new features of VirtualFlow 2.0 are: i) a new much larger ligand library (the REAL Space of Enamine containing 69 billion molecules), ii) a much larger number of docking methods (approximately 1500), allowing to target new receptor types (such as RNA and DNA), iii) supporting new hardware (e.g. ARM CPUs), iv) hardware acceleration via GPUs, v) providing more advanced ligand preparation capabilities (e.g. stereoisomer enumeration and calculation of 28 molecular properties), vi) features to support machine learning and deep learning approaches (e.g. support of SELFIES and deep learning-based docking methods), and vii) VirtualFlow 2.0 has a new module, VirtualFlow Unity, that allows the preparation and dock of ligands in a single workflow. Importantly, VirtualFlow 2.0 allows for a new screening paradigm called Adaptive Target-Guided Virtual Screens (ATG-VSs), which allows effective sampling of ultra-large ligand libraries, in a target-focused manner, for a small fraction of the cost compared to standard ULVSs.

study performed the first screen of one billion compounds to target the KEAP1-NRF2 interaction, leading to the identification of inhibitors in the sub-micromolar range (3). In 2021, the $\sigma 2$ receptor was targeted by ULVSs, leading to low-nanomolar ligands (5). Notably, in 2020, the first open-source, fully automated platform dedicated to ULVSs called VirtualFlow was published (3, 10). This platform allows users to routinely perform ULVS with a library of approximately 1.4 billion compounds in a "ready-to-dock format". In 2020, there were approximately 2 billion on-demand molecules available, but with the recent advances in manufacturing and computational technologies, this number has increased significantly (11), with estimates as high as 90 billion on-demand molecules available from multiple compound vendors.

Here, we present VirtualFlow 2.0, a next-generation *in silico* platform for ULVSs that can intelligently and efficiently sample the most recent ultra-large libraries. VirtualFlow 2.0 has vastly extended features spanning the ligand preparation module (VFLP - VirtualFlow for Ligand Preparation), the virtual screening module (VFVS - VirtualFlow for Virtual Screening), and VF Unity, a new module that unifies both VFLP and VFVS, aimed at providing a streamlined workflow for the computational drug discovery, computational biology, and the machine learning communities. These expanded capabilities of the platform were directly applied to curating the world's largest commercially available ligand library (Enamine REAL Space (12)) in a "ready-to-dock" format, consisting of 69 billion compounds, utilizing the linear scaling behavior of the platform and simultaneously employing 5.6 million vCPUs. This next-generation platform also contains a novel module, Adaptive Target-Guided Virtual Screens (ATG-VSs), which performs a pilot screen of representative molecules from diverse subsections of the 69 billion library to narrow down promising regions of the chemical space efficiently for a given target receptor. One can then expand on these target-specific regions of the chemical space in a final ultra-large screen. We observed that the ATG-VS approach can reduce computational costs by a factor of up to 1000 as compared to exhaustive screening.

VirtualFlow 2.0

Ultra-large virtual screens involve billions of molecules and files that need to be prepared, processed, and handled efficiently using high-performance computing infrastructure. For the preparation and docking of single ligands, a large number of tools and methods exist. In the context of ultra-large virtual screens, these tools need to work in concert and in parallel on different types of HPC systems (e.g. computer clusters and the cloud) and be aware of platform-based error checks and restrictions. This calls for a single unified software platform that runs on diverse HPC systems and brings together all the available tools,

allowing them to work together in a highly parallelized way. We have created the VirtualFlow open-source platform to support the above-mentioned demands.

VirtualFlow 2.0 is the next generation of virtual screening platforms, more powerful than the original version (VirtualFlow 1.0, for a detailed comparison see Extended Data Table 1). The new features (Figure 1) can be classified into three categories: platform-wide improvements (affecting all or multiple modules of VirtualFlow), module-specific new features, and new modules related to the ligand libraries that we provide. One of the major improvements in VirtualFlow 2.0 is its native cloud support, which allows for more efficient use of computational resources on AWS and improves its scaling abilities. We have demonstrated that VirtualFlow 2.0 scales linearly up to an unprecedented number of CPUs (5.6 million). Additionally, VirtualFlow 2.0 includes several new features that use machine learning and deep learning approaches for docking, which can be considerably faster than traditional docking methods and more accurate in some modalities such as blind docking (13, 14).

The ligand preparation module in VirtualFlow 2.0, called VirtualFlow for Ligand Preparation (VFLP 2.0), has several new features. In addition to ligand preparation for docking, VFLP 2.0 enables the enumeration of stereoisomers, tautomers, and the calculation of 28 molecular properties (for a complete list, see Extended Data Table 3). Notably, VFLP 2.0 also offers enhanced open-source support and the ability to partition newly created libraries into multidimensional grids, with up to 28 dimensions. This allows the user to subselect a portion of the ultra-large library with desired molecular properties. More information about these features can be found in the Methods section.

The virtual screening module in VirtualFlow 2.0, called VirtualFlow for Virtual Screening (VFVS 2.0), has been significantly enhanced. Specifically, VFVS now supports approximately 1500 docking methods, as listed in Table 1. These methods are provided by over 40 individual docking programs, which are listed in Extended Data Table 5, as well as by scoring functions (Extended Data Table 6) and pose prediction methods (Extended Data Table 7) that are assembled by VFVS 2.0 into complete docking methods. VFVS 2.0 supports the latest deep learning-based docking and pose-prediction methods and can handle new types of receptors (such as DNA and RNA) and peptide ligands. A detailed description of these and other features of VFVS 2.0 can be found in the "Methods" section.

In addition to the updates to VFLP 2.0 and VFVS 2.0, we engineered a new module called VirtualFlow Unity (VFU). VFU seamlessly combines many features of VFLP and VFVS, allowing users to prepare molecules and dock them in a single, automated workflow. Details about VFU can be found in the "Methods" section.

REAL Space

Virtual screens with larger libraries of molecules tend to produce substantially better-quality results compared to small-scale virtual screens. Research has shown that both the potency and hit rate (i.e., the number of experimentally confirmed hits divided by the number of tested compounds) improve significantly with the number of compounds screened (2, 3, 5, 10). Currently, the largest ready-to-dock libraries are the REAL Database (2018 version) (3) and the ZINC20 library (15), both of which contain approximately 1.5 billion ready-to-dock molecules. To allow for screening substantially larger ligand libraries, here we have prepared the REAL Space of Enamine into a ready-to-dock format, containing 69 billion commercially available molecules, by far the largest library for structure-based virtual screens.

The current version of the Enamine REAL Space (2022q1-2, accessed November 15, 2022, https://enamine.net/compound-collections/real-compounds) as provided by Enamine has 31,507,987,117 enumerated molecules. The molecules are derivatives of 137,000 qualified building blocks assembled via 167 in-house validated one-pot synthetic protocols (16–25). The above combination of the validated features supports the high synthetic feasibility of the REAL molecules with a success rate of 82 percent across 386,000 experiments set in 2021. The REAL Space is drug-like, and 99 percent of the molecules meet Lipinski's Rule of Five (26). The space is represented by 1,098,811,629 Murcko scaffolds. The scaffold diversity in the REAL Space is similar to the in-stock compound collection, 3,024,947 compounds are based on 1,110,956 Murcko scaffolds (27).

After ligand preparation with VFLP 2.0, the number of ready-to-dock molecules increased to 68.7 billion. This preparation process included several steps, such as enumerating stereoisomers and relevant tautomers for each molecule, desalting, neutralizing, calculating protonation states, generating 3D conformers, and converting the molecules to the desired target formats. Using the VFLP 2.0 workflow we calculated 28 different molecular properties of each compound (see Extended Data Table 3 for a complete list). Based on 18 of these molecular properties, the REAL Space library, in a ready-to-dock format was partitioned into an 18-dimensional matrix. Each dimension corresponds to one of the following molecular properties: molecular weight, logP, hydrogen bond donor count, hydrogen bond acceptor count, rotatable bond count, topological polar surface area (TPSA), logS, aromatic ring count, molecular refractivity (MR), formal charge, positive charge count, negative charge count, fsp3, chiral center count, halogen atom count, sulfur atom count, and stereoisomer count. Each dimension was subsequently partitioned into multiple intervals, resulting in a matrix-like structure (see also Figure 2). This method of partitioning results in 24.4 billion tranches (matrix elements). The molecules from the REAL Space populate approximately 12 million tranches of the 18-dimensional matrix. The 18 properties and intervals were selected based on their expected importance for selecting subsets of the library in the context of drug discovery (a significantly higher number would have resulted in

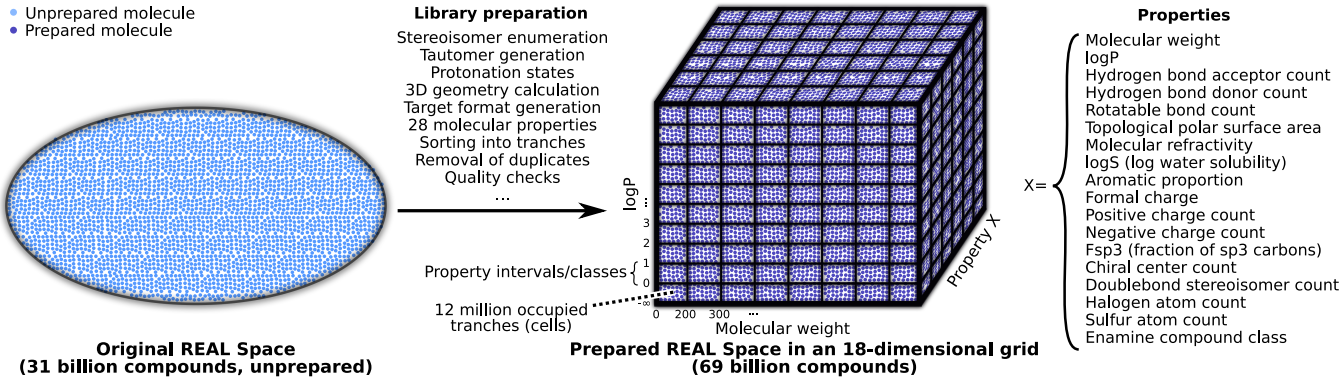


Fig. 2. VirtualFlow version of the Enamine REAL Space. The initial REAL Space from Enamine (version 2022q1-2) contained 31 billion SMILES in an unprepared form (e.g. not all stereo-isomers were enumerated). The VFLP module of VirtualFlow was used to prepare the REAL Space into a ready-to-dock format. The preparation steps included stereo-isomer enumeration, tautomerization state calculation, protonation state prediction, 3D-format conversion, and target format conversion. In addition, for each molecule, 28 properties were computed that are relevant for drug discovery. 18 of these properties were used to partition the library into an 18-dimensional grid, where each dimension is partitioned into multiple intervals. In the figure, this 18-dimensional grid is represented schematically as a 3-dimensional grid, with each of the three chosen dimensions partitioned into 6-10 subsets/intervals. This leads to $8 \cdot 6 \cdot 6 = 288$ tranches. Each tranche is shown as a small box, and within each tranche, all the ligands are contained that have the molecular properties associated with that tranche. This partitioning allows for a flexible selection of subsets of the library and enables adaptive virtual screens.

an unfavorable tranche-to-ligand ratio for efficient workflows). An overview of the partitions of the 18 dimensions/properties and their occupancy is illustrated in Extended Data Figure 3 as well as in Extended Data Table 4.

The REAL Space is an extensive library of drug-like molecules that is rich in chemical diversity and offers a number of non-flat and stereocenter-containing molecules (15, 28). With over 50.5 billion molecules (74% of the library), the library contains at least one chiral center, rendering these molecules nonflat. Nonflatness has emerged as a favorable characteristic of druglike molecules (29, 30). Specific binding of ligand to its receptor involves leveraging the unique aspects of the receptor's shape as well as the molecular and atomic properties of the binding site. In this case, in a Goldilocks manner, if a compound has multiple stereoisomers, the various stereoisomers should have different docking scores due to the different three-dimensional properties to specifically engage the target. The difference in docking scores among the stereoisomers can be considered a trait of specific binding and should aid in the selection of hits. The REAL Space includes a significant number of halogen-containing compounds (20.2 billion molecules), of which 17.4 billion harbor fluorine atoms. These molecules can be used in 19F-NMR experiments to validate binding without the need for isotopically labeled proteins. Despite the diversity of the REAL Space, vast areas of the 18-dimensional matrix are empty, allowing chemical providers to explore and discover new molecules that may selectively engage specific proteins in the proteome.

We have designed two interactive interfaces for the exploration of the REAL Space library (Extended Data Figures 5 and 6), that use different implementation techniques and thus can differ in their performance dependent on the device/browser used. This web interface allows users to select specific subsets of the library for virtual screens using VFVS. The interface displays the library as a two-dimensional table, which is a projection of the other 16 dimensions of the complete 18-dimensional matrix. Users can choose which property or dimension to display on each of the two axes and use interactive sliders to select a specific range for each property or dimension. We make the REAL Space available in PDB, PDBQT, MOL2, and SDF ready-to-dock formats, which are compatible with the docking format requirements of almost all docking programs that use small molecules as ligands. In addition, the library is available in the SMILES, SELFIES, and Parquet formats, which can be useful for tasks complementary to virtual screens (e.g. generative models). The entire library or user-defined sub-sections of the library can be accessed through the VirtualFlow project homepage (`https://virtual-flow.org/`) or via a dataset on AWS Open Data, where the library can be directly used for virtual screenings in the cloud without the need to locally download it.

Adaptive Ultra-Large Virtual Screenings

Screening ultra-large ligand libraries containing billions of molecules is computationally intensive, especially when using traditional docking programs. As the number of compounds in the library increases into the tens of billions, the cost of screening the library becomes prohibitively expensive.

Virtual screening is a process of searching a vast chemical space for the best candidate that matches the structural and biophysical properties of the target site in a given receptor. However, the fraction of compounds in an ultra-large screen that will optimally engage a specific site in the receptor is often very small and share specific molecular characteristics. To be cost- and time-effective, it is desirable to focus the screen on the optimal chemical space that is likely to contain hits for a given target. To rapidly narrow the search space for a given target, we introduce the Adaptive Target-Guided Virtual Screening (ATG-VS) technique. This approach consists of the following steps:

1. **Library Decomposition.** Each tranche in the current library has approximately 5600 molecules, and for each tranche

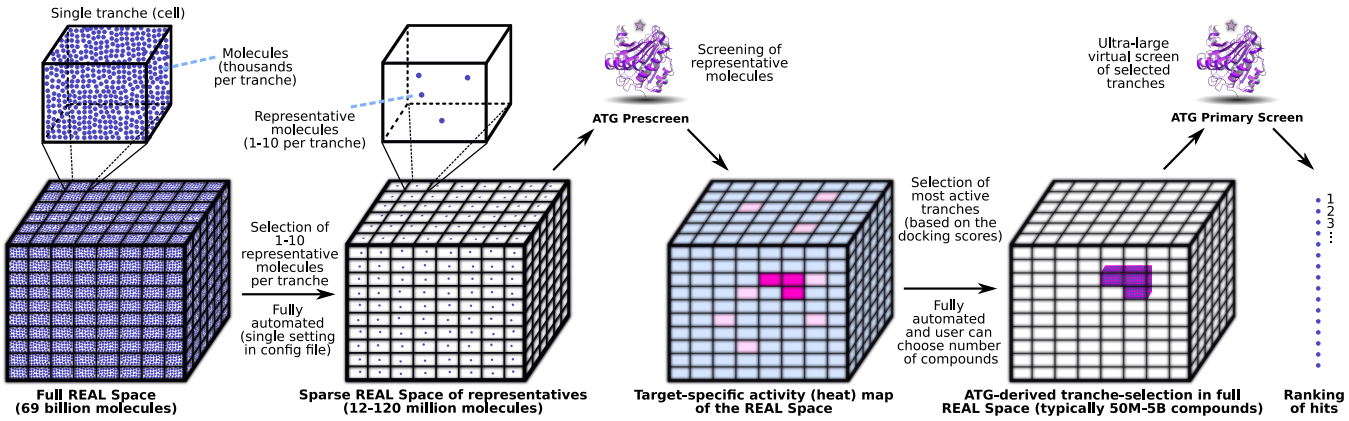


Fig. 3. Adaptive Target-Guided Virtual Screens (ATG-VSs). The REAL Space, divided into an 18-dimensional grid, populates over 12 million individual tranches (cells) represented in the figure schematically as a 3-dimensional matrix and each of these tranch has on average 5600 molecules. During the first step of ATG-VSs, the VFVS module of VirtualFlow selects a few (1-10) representative molecules in each tranche and docks these molecules to the target receptor. This screening mode (and stage) is referred to as a prescreen and allows VirtualFlow to detect the tranches of the library that contain molecules of high potential binding affinity to the given target receptor/binding site. VirtualFlow then selects these tranches and screens them fully against the target receptor.

up to ten representative ligands are selected for a pilot screen. The number of representatives per tranche (1-10) can be selected by the user. Since there are approximately 12 million tranches (that the molecules from the REAL Space populate), if one representative per tranche is selected, the total number of molecules screened in the pilot screen would be approximately 12 million for the entire library. If more than one representative molecule, per tranche, is chosen by the user then these additional molecules are automatically selected to be diverse from each other within the given tranche.

2. **Screening Representative Molecules (ATG Prescreen).** Instead of screening the entire library, a prescreen of only the representative ligands is performed. After the pre-screen, the cells (tranches) of the ligand library that have the best docking scores among the representative ligands are selected for the next screening stage.

3. **Selection of Active Tranches.** VirtualFlow 2.0 automatically selects the most promising tranches based on docking scores of the ATG Prescreen, and the user can select the weight of each of the two criteria. The total number of ligands to be screened in the next stage can be specified by the user.

4. **Active Cell Screening (ATG Primary Screen).** In the next stage, the tranches that were selected after the prescreen are fully screened (e.g. in Figure 3, the three tranches that contain the best ligands are marked purple).

ATG-VSs significantly reduce the dimension of the chemical space to be screened, thereby reducing the required computation time. VFLP 2.0 can also prepare new and/or proprietary ligand libraries with ATG-VS support, which requires that the prepared ligand libraries are structured into a multidimensional tranche matrix with pre-assigned representative molecules for each tranche. An optional third-stage screen can be performed to rescore the best hits with higher accuracy, such as by including receptor flexibility. VirtualFlow can carry out all three stages in a fully automated workflow.

To investigate the effectiveness of the various parameters including the number of representative molecules from each tranche, and the docking exhaustiveness (which is a measure of how the ligands' conformational space is sampled), we carried out multiple ATG prescreens on an array of targets. We examined the following scenarios, i) one representative molecule per tranche and exhaustiveness of 1, ii) 10 representative molecules and exhaustiveness of 1, and iii) 1 representative molecule with exhaustiveness of 10. The higher the exhaustiveness parameter the more thoroughly the confirmational space is searched, but also the more computation time is needed. The results of these trials can be seen in Extended Data Figure 8. As shown in the figure, the results (based on the distribution of the top docking scores, and the ranking of the tranches within each property category) are relatively similar for all three scenarios. Therefore, ATG prescreens with exhaustiveness 1 and one representative molecule per tranche are expected to be sufficient for many applications. In this case, an ATG prescreen involves docking 12 million molecules to explore all parts of the 69 billion compound library, which is over 5000 times less computational effort than screening all molecules of the library.

To investigate the performance of ATG-VSs in comparison to standard ultra-large virtual screens, we screened 10 different target proteins of various types: including kinases, phosphatases, GPCRs, and protein-protein interfaces (cf. Extended Data Figure 4). We investigated the advantages of an ATG screen using two different pilot studies on an array of different protein targets. i) We compared the docking scores of the top 50 hits from a screen of either 100 million random molecules using standard ULVSs with 100 million molecules chosen by the ATG prescreen, the ligands chosen after the ATG prescreen consistently performed substantially better, and in most cases resulted in hits with the predicted binding constant approximately five-fold better than those obtained with the random selection (see Extended Data Figure 9). ii) We investigated if the ATG screen can be used to effectively and efficiently sample vast regions of the chemical space. Here we show that the ATG primary

screen, consisting of only ten thousand to one million compounds, often yields top-scoring compounds with similar docking scores as when screening 100 million random compounds in standard ultra-large virtual screens (ULVSs), representing a reduction of 100-10,000 in the number of compounds screened (see Figure 4). Compared to CB1 where we see 10,000-fold time saving, we observe a 100-fold reduction for NTR1. The performance of the ATG screen depends on the target and the nature of the library screened. Some of the target surfaces are rich in molecular features, harboring a combination of hydrophobic and electrostatic surfaces and the docking area is large in size. In such instances, the ATG will identify a rather diffused area in the 18-dimensional tranch. It should also be noted that the molecules of the REAL Space have a molecular weight below 450 Da, and if the library would contain larger ligands (e.g. 450-700 Da) ATG-VSs might be even more effective for certain targets. These results demonstrate the effectiveness of ATG-VSs in comparison to standard ULVSs for identifying high-quality compounds for various target proteins.

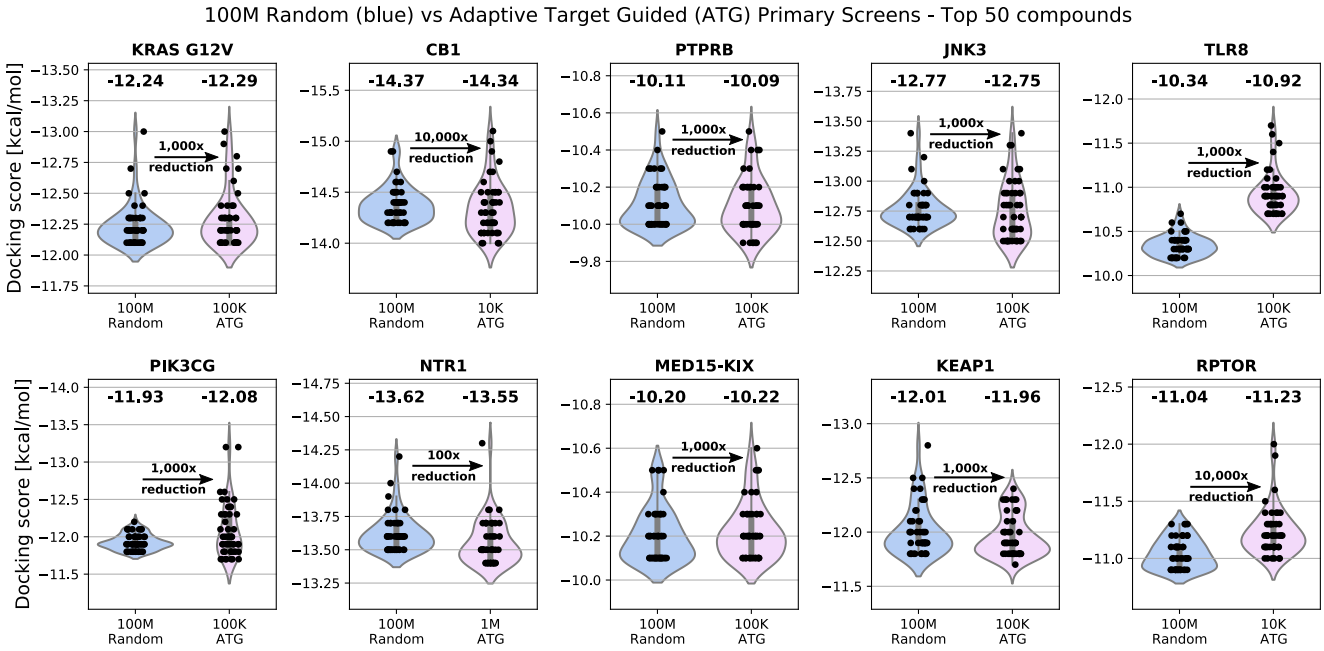


Fig. 4. Benchmarks Involving Adaptive Target-Guided (ATG) Primary Virtual Screens and Standard ULVSs. Violin plots of the docking scores (the more negative the better) of the top 50 virtual screening hits obtained by standard ULVS using 100 million random compounds (blue) and by considerably smaller ATG primary screens (rose). Stripplots of the data points are overlayed, and the numbers above the violin plots are the mean docking scores of the top 50 hits. For most target proteins, ATG primary screens in which 100,000 molecules were screened after the ATG prescreen performed similarly well as standard ULVSs in which 100 million random compounds were screened, resulting in a reduction of 100-10,000x regarding the number of compounds screened in the primary virtual screening stage.

Machine Learning and Cloud Support

AI Support. VirtualFlow 2.0 has significantly enhanced support for artificial intelligence (AI) and machine learning (ML) approaches in the fields of molecular biology and drug discovery. In addition to leveraging ML-based docking programs such as EquiBind (31), DiffDock(14), and TANKBind (13), VirtualFlow 2.0 has features designed to support ML workflows. For example, VFLP supports the calculation of properties such as the Quantitative Estimate of Druglikeness (QED) (32), and octanol solubility (logP) (33), both properties have been heavily used in ML communities for optimization of small molecules. Another new feature of VFLP 2.0 is its support for the SELFIES molecular representation (34, 35). Due to their robustness, SELFIES are often used in generative models to create new molecules (36). The new module VirtualFlow Unity allows for easier training data generation, and seamless integration of docking programs to be utilized as objective functions (37) when creating new machine-learning models. VFU makes molecular docking also more accessible to drug discovery communities since it fully automates ligand preparation and molecular docking within a single and simple unified command-line interface able to access approximately 1500 docking methods, eliminating the need for users to learn how to use these methods individually. VFVS 2.0 provides many new docking methods by assembling existing deep learning-based scoring functions and sampling algorithms into full docking programs. Overall, VirtualFlow 2.0 can serve as a hub for developing new ML methods for drug discovery and make existing ML-based structure-based drug discovery methods, such as docking methods, more accessible to the drug discovery community.

Cloud Support. VirtualFlow 2.0 is designed to run efficiently in the cloud. VirtualFlow runs on top of batch systems such as Slurm (38) and AWS Batch (39). Slurm is supported by many cloud providers such as Google Cloud and Microsoft Azure, and AWS Batch is tailored to AWS cloud. In this study, we demonstrate the use of VirtualFlow 2.0 with over 5.6 million vCPUs in

AWS, setting a new global record for the number of vCPUs used in parallel in the cloud. This impressive scalability in terms of the number of CPUs allows for the completion of ultra-large-scale virtual screenings (ULVSs) involving billions of molecules in a matter of hours, which would take weeks to months to complete on shared university computer clusters.

Discussion

The size of the chemical space available for drug discovery is rapidly expanding, with the number of on-demand molecules increasing from around 2 billion in 2020 to around 90 billion in 2022, and is expected to reach trillions in the near future. However, even this vast chemical space is still a minuscule fraction of the potential space of drug-like molecules. To access a significant portion of the expanding chemical space for drug discovery, we have created the largest ready-to-dock ligand library available to date, the REAL Space from Enamine, with a total of 69 billion molecules. This library allows researchers to perform virtual screens on a vast number of molecules, increasing the chances of discovering potent hits and optimizing lead compounds. Notably, the scale of the screen improves the docking scores of the best virtual screening hits, which can lead to higher potency and higher true-hit rates (2, 3, 5, 10, 40). By using a larger library, researchers can increase their chances of finding potent and reliable hits, making the screening process more efficient and effective.

The rapid increase in accessible chemical space requires an efficient and flexible platform for efficient access and navigation in the context of ULVSs. While we have access to unprecedented computational power through cloud-based service providers, the cost of screening billions of molecules is often prohibitive for academic researchers. To address this challenge, VirtualFlow 2.0 introduces a new adaptive target-based virtual screening (ATG-VS) technique, which allows researchers to efficiently explore the largest ligand libraries available to date. For the Enamine REAL Space, an ATG prescreen of 12-120 million compounds can dramatically narrow the search space in a target-dependent manner, providing a speed-up (and cost reduction) of up to 1000. In addition, VirtualFlow 2.0 offers fast deep learning- and GPU-based docking methods that can speed up screens by up to two orders of magnitude when compared to traditional docking programs and CPUs, respectively.

VirtualFlow 2.0 is a platform that brings together a wide range of diverse docking methods, deep learning-based scoring functions, and pose-prediction methods in a single unified framework in which these methods can be used in concert and complement each other synergistically. Individual scoring functions, pose prediction methods, and full docking programs are assembled into new complete docking methods, leading to a large number of possible choices for the user. Additional methods can be easily added in the future as new methods become available. Closely related, we posit that generative models will soon be able to generate large libraries of molecules with the potential to act as therapeutics (41). To train and evaluate these models, VirtualFlow 2.0 offers a streamlined software backend that can be easily utilized by the machine-learning communities.

In conclusion, VirtualFlow 2.0 is a platform that offers a comprehensive and flexible approach to virtual screening, with the ability to access the largest ready-to-dock library of drug-like molecules currently available. The platform employs techniques to reduce computational costs, making it an effective tool for hit identification and lead optimization in drug discovery. Its open-source nature and impressive scaling capabilities make it a valuable resource for researchers. Despite the significant progress made in the field of ultra-large virtual screenings in recent years, there is still much room for improvement and development in the coming years.

Online Content. Any methods, additional references, Nature Research reporting summaries, source data, statements of data availability, and associated accession codes are available online at https://doi.org

References

1. Joseph A. DiMasi, Henry G. Grabowski, and Ronald W. Hansen. Innovation in the pharmaceutical industry: New estimates of R&D costs. Journal of Health Economics, 47:20–33, may 2016. ISSN 18791646. doi: 10.1016/j.jhealeco.2016.01.012.
2. Jiankun Lyu, Sheng Wang, Trent E Balius, et al. Ultra-large library docking for discovering new chemotypes. Nature, 2019. ISSN 1476-4687. doi: 10.1038/s41586-019-0917-9.
3. Christoph Gorgulla, Andras Boeszoermenyi, Zi-fu Wang, Patrick D Fischer, Paul W. Coote, Krishna M. Padmanabha Das, Yehor S Malets, Dmytro S Radchenko, Yurii S Moroz, David A Scott, Konstantin Fackeldey, Moritz Hoffmann, Iryna Iavniuk, Gerhard Wagner, and Haribabu Arthanari. An open-source drug discovery platform enables ultra-large virtual screens. Nature, 580(7805): 663–668, apr 2020. ISSN 0028-0836. doi: 10.1038/s41586-020-2117-z.
4. Reed M Stein, Hye Jin Kang, John D McCorvy, Grant C Glatfelter, Anthony J Jones, Tao Che, Samuel Slocum, Xi-Ping Huang, Olena Savych, Yurii S Moroz, Benjamin Stauch, Linda C Johansson, Vadim Cherezov, Terry Kenakin, John J Irwin, Brian K Shoichet, Bryan L Roth, and Margarita L Dubocovich. Virtual discovery of melatonin receptor ligands to modulate circadian rhythms. Nature, feb 2020. ISSN 0028-0836. doi: 10.1038/s41586-020-2027-0.
5. Assaf Alon, Jiankun Lyu, Joao M. Braz, Tia A. Tummino, Veronica Craik, Matthew J. O'Meara, Chase M. Webb, Dmytro S. Radchenko, Yurii S. Moroz, Xi-Ping Huang, Yongfeng Liu, Bryan L. Roth, John J. Irwin, Allan I. Basbaum, Brian K. Shoichet, and Andrew C. Kruse. Structures of the $\sigma 2$ receptor enable docking for bioactive ligand discovery. Nature, 600(7890):759–764, dec 2021. ISSN 0028-0836. doi: 10.1038/s41586-021-04175-x.
6. Anat Levit Kaplan, Danielle N Confair, Kuglae Kim, Ximena Barros-Álvarez, Ramona M Rodriguiz, Ying Yang, Oh Sang Kweon, Tao Che, John D. McCorvy, David N Kamber, James P Phelan, Luan Carvalho Martins, Vladimir M Pogorelov, Jeffrey F. DiBerto, Samuel T Slocum, Xi-ping Huang, Jain Manish Kumar, Michael J. Robertson, Ouliana Panova, Alpay B. Seven, Autumn Q. Wetsel, William C Wetsel, John J Irwin, Georgios Skiniotis, Brian K Shoichet, Bryan L. Roth, and Jonathan A. Ellman. Bespoke library docking for 5-HT2A receptor agonists with antidepressant activity. Nature, 610(7932), sep 2022. ISSN 0028-0836. doi: 10.1038/s41586-022-05258-z.
7. Elissa A Fink, Jun Xu, Harald Hübner, Joao M Braz, Philipp Seemann, Charlotte Avet, Veronica Craik, Dorothee Weikert, Maximilian F Schmidt, Chase M Webb, Nataliya A Tolmachova, Yurii S Moroz, Xi-Ping Huang, Chakrapani Kalyanaraman, Stefan Gahbauer, Geng Chen, Zheng Liu, Matthew P Jacobson, John J Irwin, Michel Bouvier, Yang Du, Brian K Shoichet, Allan I Basbaum, and Peter Gmeiner. Structure-based discovery of nonopioid analgesics acting through the α 2A -adrenergic receptor. Science, 377(6614):eabn7065, sep 2022. ISSN 0036-8075. doi: 10.1126/science.abn7065.
8. Steven M Mennen, Carolina Alhambra, C Liana Allen, Mario Barberis, Simon Berritt, Thomas A Brandt, Andrew D Campbell, Jesús Castañón, Alan H Cherney, Melodie Christensen, et al. The evolution of high-throughput experimentation in pharmaceutical development and perspectives on the future. Organic Process Research & Development, 23(6):1213–1242, 2019.
9. Regine S. Bohacek, Colin McMartin, and Wayne C. Guida. The art and practice of structure-based drug design: A molecular modeling perspective. Medicinal Research Reviews, 16(1):3–50, 1996. ISSN 01986325. doi: 10.1002/(SICI)1098-1128(199601)16:1<3::AID-MED1>3.0.CO;2-6.

10. Christoph Gorgulla. Free Energy Methods Involving Quantum Physics , Path Integrals , and Virtual Screenings: Development, Implementation and Application in Drug Discovery. PhD thesis, Freie Universität Berlin, 2018.
11. Oleksandr O. Grygorenko, Dmytro S. Radchenko, Igor Dziuba, Alexander Chuprina, Kateryna E. Gubina, and Yurii S. Moroz. Generating Multibillion Chemical Space of Readily Accessible Screening Compounds. iScience, 23(11):101681, 2020. ISSN 25890042. doi: 10.1016/j.isci.2020.101681.
12. Enamine. REAL Space: Billions of make-on-demand molecules. https://enamine.net/compound-collections/real-compounds/real-space-navigator, 2022.
13. Wei Lu, Qifeng Wu, Jixian Zhang, Jiahua Rao, Chengtao Li, and Shuangjia Zheng. TANKBind: Trigonometry-Aware Neural NetworKs for Drug-Protein Binding Structure Prediction. bioRxiv, 2022. doi: 10.1101/2022.06.06.495043v1.
14. Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. arXiv preprint arXiv:2210.01776, 2022.
15. John J Irwin, Khanh G Tang, Jennifer Young, Chinzorig Dandarchuluun, Benjamin R Wong, Munkhzul Khurelbaatar, Yurii S Moroz, John Mayfield, and Roger A Sayle. ZINC20—A Free Ultralarge-Scale Chemical Database for Ligand Discovery. Journal of Chemical Information and Modeling, 60(12):6065–6073, dec 2020. ISSN 1549-9596. doi: 10.1021/acs.jcim.0c00675.
16. Andrey V Bogolubsky, Yurii S Moroz, Pavel K Mykhailiuk, Dmitry S Granat, Sergey E Pipko, Anzhelika I Konovets, Roman Doroschuk, and Andrey Tolmachev. Bis (2, 2, 2-trifluoroethyl) carbonate as a condensing agent in one-pot parallel synthesis of unsymmetrical aliphatic ureas. ACS Combinatorial Science, 16(6):303–308, 2014.
17. Andrey V Bogolubsky, Yurii S Moroz, Pavel K Mykhailiuk, Sergey E Pipko, Anzhelika I Konovets, Irina V Sadkova, and Andrey Tolmachev. Sulfonyl fluorides as alternative to sulfonyl chlorides in parallel synthesis of aliphatic sulfonamides. ACS Combinatorial Science, 16(4):192–197, 2014.
18. Andrey V Bogolubsky, Yurii S Moroz, Pavel K Mykhailiuk, Dmitriy M Panov, Sergey E Pipko, Anzhelika I Konovets, and Andrey Tolmachev. A one-pot parallel reductive amination of aldehydes with heteroaromatic amines. ACS Combinatorial Science, 16(8):375–380, 2014.
19. Andrey V Bogolubsky, Yurii S Moroz, Pavel K Mykhailiuk, Eugeniy N Ostapchuk, Alexander V Rudnichenko, Yurii V Dmytriv, Anna N Bondar, Olga A Zaporozhets, Sergey E Pipko, Roman A Doroschuk, et al. One-pot parallel synthesis of alkyl sulfides, sulfoxides, and sulfones. ACS Combinatorial Science, 17(6):348–354, 2015.
20. Andrey V Bogolubsky, Yurii S Moroz, Pavel K Mykhailiuk, Sergey E Pipko, Anton V Zhemera, Anzhelika I Konovets, Olena O Stepaniuk, Inna S Myronchuk, Yurii V Dmytriv, Roman A Doroschuk, et al. 2, 2, 2-trifluoroethyl chlorooxoacetate universal reagent for one-pot parallel synthesis of n 1-aryl-n 2-alkyl-substituted oxamides. ACS Combinatorial Science, 17(10):615–622, 2015.
21. Andrey Tolmachev, Andrey V Bogolubsky, Sergey E Pipko, Alexander V Grishchenko, Dmytro V Ushakov, Anton V Zhemera, Oleksandr O Viniychuk, Anzhelika I Konovets, Olga A Zaporozhets, Pavel K Mykhailiuk, et al. Expanding synthesizable space of disubstituted 1, 2, 4-oxadiazoles. ACS combinatorial science, 18(10):616–624, 2016.
22. Andrey V Bogolubsky, Yurii S Moroz, Olena Savych, Sergey Pipko, Angelika Konovets, Maxim O Platonov, Oleksandr V Vasylchenko, Vasyl V Hurmach, and Oleksandr O Grygorenko. An old story in the parallel synthesis world: an approach to hydantoin libraries. ACS Combinatorial Science, 20(1):35–43, 2018.
23. Olena Savych, Yuliya O Kuchkovska, Andrey V Bogolyubsky, Anzhelika I Konovets, Kateryna E Gubina, Sergey E Pipko, Anton V Zhemera, Alexander V Grishchenko, Dmytro N Khomenko, Volodymyr S Brovarets, et al. One-pot parallel synthesis of 5-(dialkylamino) tetrazoles. ACS combinatorial science, 21(9):635–642, 2019.
24. OO Grygorenko, DS Radchenko, I Dziuba, A Chuprina, KE Gubina, and YS Moroz. Generating multibillion chemical space of readily accessible screening compounds. iscience 2020, 23 (11), 101681, 2020.
25. CA Lipinski, F Lombardo, BW Dominy, and PJ Feeney. In vitro models for selection of development candidatesexperimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. Adv Drug Deliv Rev, 23(1):3–25, 1997.
26. Christopher A Lipinski, Franco Lombardo, Beryl W Dominy, and Paul J Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. Advanced drug delivery reviews, 23(1-3):3–25, 1997.
27. Enamine. Enamine In-Stock Screening Collection. https://enamine.net/compound-collections/screening-collection, 2022.
28. Anna Tomberg and Jonas Boström. Can easy chemistry produce complex, diverse, and novel molecules? Drug Discovery Today, 00(00):1–8, oct 2020. ISSN 13596446. doi: 10.1016/j.drudis.2020.09.027.
29. Frank Lovering, Jack Bikker, and Christine Humblet. Escape from flatland: increasing saturation as an approach to improving clinical success. Journal of medicinal chemistry, 52(21):6752–6756, 2009.
30. Frank Lovering. Escape from flatland 2: complexity and promiscuity. MedChemComm, 4(3):515–519, 2013.
31. Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. Equibind: Geometric deep learning for drug binding structure prediction. In International Conference on Machine Learning, pages 20503–20521. PMLR, 2022.
32. G. Richard J. Bickerton, Gaia Valeria Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L. Hopkins. Quantifying the chemical beauty of drugs. Nature chemistry, 4 2:90–8, 2012.
33. Scott A Wildman and Gordon M Crippen. Prediction of physicochemical parameters by atomic contributions. Journal of chemical information and computer sciences, 39(5):868–873, 1999.
34. Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. Self-referencing embedded strings (selfies): A 100% robust molecular string representation. Machine Learning: Science and Technology, 1(4):045024, oct 2020. doi: 10.1088/2632-2153/aba947.
35. Mario Krenn, Qianxiang Ai, Senja Barthel, Nessa Carson, Angelo Frei, Nathan C Frey, Pascal Friederich, Théophile Gaudin, Alberto Alexander Gayle, Kevin Maik Jablonka, et al. Selfies and the future of molecular string representations. Patterns, 3(10):100588, 2022.
36. Daniel Flam-Shepherd, Kevin Zhu, and Alán Aspuru-Guzik. Language models can learn complex molecular distributions. Nature Communications, 13(1):1–10, 2022.
37. AkshatKumar Nigam, Robert Pollice, Gary Tom, Kjell Jorner, Luca A Thiede, Anshul Kundaje, and Alan Aspuru-Guzik. Tartarus: A benchmarking platform for realistic and practical inverse molecular design. arXiv preprint arXiv:2209.12487, 2022.
38. SLURM. https://slurm.schedmd.com, Accessed: 2018-12-22.
39. AWS Batch. https://aws.amazon.com/batch/, Accessed: 2022-12-31.
40. David E. Gloriam. Bigger is better in virtual drug screens. Nature, 566(7743):193–194, feb 2019. ISSN 0028-0836. doi: 10.1038/d41586-019-00145-6.
41. Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. Inverse molecular design using machine learning: Generative models for matter engineering. Science, 361(6400):360–365, 2018.
42. Noel M O'Boyle, Michael Banck, Craig A James, Chris Morley, Tim Vandermeersch, and Geoffrey R Hutchison. Open Babel: An open chemical toolbox. Journal of Cheminformatics, 3(1):33, December 2011. ISSN 1758-2946. doi: 10.1186/1758-2946-3-33.
43. David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. Journal of Chemical Information and Computer Sciences, 28(1):31–36, 1988. doi: 10.1021/ci00057a005.
44. Greg Landrum et al. Rdkit: Open-source cheminformatics, 2006.
45. Christoph Gorgulla, Konstantin Fackeldey, Gerhard Wagner, and Haribabu Arthanari. Accounting of Receptor Flexibility in Ultra-Large Virtual Screens with VirtualFlow Using a Grey Wolf Optimization Method. Supercomputing Frontiers and Innovations, 7(3):4–12, sep 2020. ISSN 23138734. doi: 10.14529/jsfi200301.
46. Christoph Gorgulla, Süleyman Selim Çınaroğlu, Patrick D Fischer, Konstantin Fackeldey, Gerhard Wagner, and Haribabu Arthanari. VirtualFlow Ants—Ultra-Large Virtual Screenings with Artificial Intelligence Driven Docking Algorithm Based on Ant Colony Optimization. International Journal of Molecular Sciences, 22(11):5807, may 2021. ISSN 1422-0067. doi: 10.3390/ijms22115807.
47. Amr Alhossary, Stephanus Daniel Handoko, Yuguang Mu, and Chee-Keong Kwoh. Fast, accurate, and reliable molecular docking with QuickVina 2. Bioinformatics, 31(13):2214–2216, jul 2015. ISSN 1460-2059. doi: 10.1093/bioinformatics/btv082.
48. Diogo Santos-Martins, Leonardo Solis-Vasquez, Andreas F Tillack, Michel F Sanner, Andreas Koch, and Stefano Forli. Accelerating autodock4 with gpus and gradient-based local search. Journal of chemical theory and computation, 17(2):1060–1073, 2021.
49. Shidi Tang, Ruiqi Chen, Mengru Lin, Qingde Lin, Yanxiang Zhu, Ji Ding, Haifeng Hu, Ming Ling, and Jiansheng Wu. Accelerating autodock vina with gpus. Molecules, 27(9):3041, 2022.
50. Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. arXiv preprint arXiv:2210.01776, 2022.
51. Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. Equibind: Geometric deep learning for drug binding structure prediction. In International Conference on Machine Learning, pages 20503–20521. PMLR, 2022.
52. Andrew T McNutt, Paul Francoeur, Rishal Aggarwal, Tomohide Masuda, Rocco Meli, Matthew Ragoza, Jocelyn Sunseri, and David Ryan Koes. Gnina 1.0: molecular docking with deep learning. Journal of cheminformatics, 13(1):1–20, 2021.
53. Jacob D Durrant and J Andrew McCammon. Nnscore 2.0: a neural-network receptor–ligand scoring function. Journal of chemical information and modeling, 51(11):2897–2903, 2011.
54. Haiping Zhang, Linbu Liao, Konda Mani Saravanan, Peng Yin, and Yanjie Wei. Deepbindrg: a deep learning based method for estimating effective protein–ligand affinity. PeerJ, 7:e7362, 2019.
55. Mostafa Karimi, Di Wu, Zhangyang Wang, and Yang Shen. Deepaffinity: interpretable deep learning of compound–protein affinity through unified recurrent and convolutional neural networks. Bioinformatics, 35(18):3329–3338, 2019.
56. Anthony J Clark, Pratyush Tiwary, Ken Borrelli, Shulu Feng, Edward B Miller, Robert Abel, Richard A Friesner, and Bruce J Berne. Prediction of protein–ligand binding poses via a combination of induced fit docking and metadynamics simulations. Journal of chemical theory and computation, 12(6):2990–2998, 2016.
57. Edward B Miller, Robert B Murphy, Daniel Sindhikara, Kenneth W Borrelli, Matthew J Grisewood, Fabio Ranalli, Steven L Dixon, Steven Jerome, Nicholas A Boyles, Tyler Day, et al. Reliable and accurate solution to the induced fit docking problem for protein–ligand binding. Journal of Chemical Theory and Computation, 17(4):2630–2639, 2021.
58. Ercheng Wang, Huiyong Sun, Junmei Wang, Zhe Wang, Hui Liu, John ZH Zhang, and Tingjun Hou. End-point binding free energy calculation with mm/pbsa and mm/gbsa: strategies and applications in drug design. Chemical reviews, 119(16):9478–9508, 2019.
59. Dominykas Lukauskis, Marley L Samways, Simone Aureli, Benjamin P Cossins, Richard D Taylor, and Francesco Luigi Gervasio. Open binding pose metadynamics: An effective approach for the ranking of protein–ligand binding poses. Journal of Chemical Information and Modeling, 2022.
60. Garrett M Morris, Ruth Huey, William Lindstrom, Michel F Sanner, Richard K Belew, David S Goodsell, and Arthur J Olson. Autodock4 and autodocktools4: Automated docking with selective receptor flexibility. Journal of computational chemistry, 30(16):2785–2791, 2009.

61. Diogo Santos-Martins, Stefano Forli, Maria João Ramos, and Arthur J Olson. Autodock4zn: an improved autodock force field for small-molecule docking to zinc metalloproteins. Journal of chemical information and modeling, 54(8):2371–2379, 2014.
62. Jerome Eberhardt, Diogo Santos-Martins, Andreas F Tillack, and Stefano Forli. Autodock vina 1.2. 0: New docking methods, expanded force field, and python bindings. Journal of Chemical Information and Modeling, 61(8):3891–3898, 2021.
63. Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. Journal of computational chemistry, 31(2):455–461, 2010.
64. Pradeep Anand Ravindranath, Stefano Forli, David S Goodsell, Arthur J Olson, and Michel F Sanner. Autodockfr: advances in protein-ligand docking with explicitly specified binding site flexibility. PLoS computational biology, 11(12):e1004586, 2015.
65. Junkai Ji, Jin Zhou, Zhangfan Yang, Qiuzhen Lin, Jianqiang Li, and Carlos A Coello Coello. Autodock koto: A gradient boosting differential evolution for molecular docking. IEEE Transactions on Evolutionary Computation, 2022.
66. Liang Rao, Ning-Xin Jia, Jun Hu, Dong-Jun Yu, and Gui-Jun Zhang. Atpdock: a template-based method for atp-specific protein–ligand docking. Bioinformatics, 38(2):556–558, 2022.
67. Kai Zhu, Kenneth W Borrelli, Jeremy R Greenwood, Tyler Day, Robert Abel, Ramy S Farid, and Edward Harder. Docking covalent inhibitors: a parameter free approach to pose prediction and scoring. Journal of chemical information and modeling, 54(7):1932–1940, 2014.
68. William J Allen, Trent E Balius, Sudipto Mukherjee, Scott R Brozell, Demetri T Moustakas, P Therese Lang, David A Case, Irwin D Kuntz, and Robert C Rizzo. Dock 6: Impact of new features and current docking performance. Journal of computational chemistry, 36(15):1132–1156, 2015.
69. Bernd Kramer, Matthias Rarey, and Thomas Lengauer. Evaluation of the flexx incremental construction algorithm for protein–ligand docking. Proteins: Structure, Function, and Bioinformatics, 37 (2):228–241, 1999.
70. Xiaocong Yang, Yang Liu, Jianhong Gan, Zhi-Xiong Xiao, and Yang Cao. Fitdock: protein–ligand docking by template fitting. Briefings in Bioinformatics, 23(3), 2022.
71. Woong-Hee Shin and Chaok Seok. Galaxydock: protein–ligand docking with flexible protein side-chains. Journal of chemical information and modeling, 52(12):3225–3232, 2012.
72. Thomas A Halgren, Robert B Murphy, Richard A Friesner, Hege S Beard, Leah L Frye, W Thomas Pollard, and Jay L Banks. Glide: a new approach for rapid, accurate docking and scoring. 2. enrichment factors in database screening. Journal of medicinal chemistry, 47(7):1750–1759, 2004.
73. Marcel L Verdonk, Jason C Cole, Michael J Hartshorn, Christopher W Murray, and Richard D Taylor. Improved protein–ligand docking using gold. Proteins: Structure, Function, and Bioinformatics, 52(4):609–623, 2003.
74. Kin Meng Wong, Hio Kuan Tai, and Shirley WI Siu. Gwovina: A grey wolf optimization approach to rigid and flexible receptor docking. Chemical biology & drug design, 97(1):97–110, 2021.
75. Yumeng Yan, Huanyu Tao, Jiahua He, and Sheng-You Huang. The hdock server for integrated protein–protein docking. Nature protocols, 15(5):1829–1852, 2020.
76. Yumeng Yan, Huanyu Tao, and Sheng-You Huang. Hsymdock: a docking web server for predicting the structure of protein homo-oligomers with cn or dn symmetry. Nucleic acids research, 46 (W1):W423–W431, 2018.
77. Hongjian Li, Kwong-Sak Leung, and Man-Hon Wong. idock: A multithreaded virtual screening tool for flexible ligand docking. In 2012 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), pages 77–84. IEEE, 2012.
78. Kai-Cheng Hsu, Yen-Fu Chen, Shen-Rong Lin, and Jinn-Moon Yang. igemdock: a graphical environment of enhancing gemdock using pharmacological interactions and post-screening analysis. BMC bioinformatics, 12(1):1–11, 2011.
79. Ni Liu and Zhibin Xu. Using ledock as a docking tool for computational drug design. In IOP Conference Series: Earth and Environmental Science, volume 218, page 012143. IOP Publishing, 2019.
80. Cherayathumadom M Venkatachalam, Xiaohui Jiang, Tom Oldfield, and Marvin Waldman. Ligandfit: a novel method for the shape-directed rapid docking of ligands to protein active sites. Journal of Molecular Graphics and Modelling, 21(4):289–307, 2003.
81. Brian Jiménez-García, Jorge Roel-Touris, Miguel Romero-Durana, Miquel Vidal, Daniel Jiménez-González, and Juan Fernández-Recio. Lightdock: a new multi-scale approach to protein–protein docking. Bioinformatics, 34(1):49–55, 2018.
82. Zhiwei Ma and Xiaoqin Zou. Mdock: A suite for molecular inverse docking and target prediction. In Protein-Ligand Interactions and Drug Design, pages 313–322. Springer, 2021.
83. Ming Liu and Shaomeng Wang. Mcdock: a monte carlo simulation approach to the molecular docking problem. Journal of computer-aided molecular design, 13(5):435–451, 1999.
84. Gabriela Bitencourt-Ferreira and Walter Filgueira de Azevedo. Molegro virtual docker for docking. In Docking screens for drug discovery, pages 149–167. Springer, 2019.
85. Fang Bai, Sha Liao, Junfeng Gu, Hualiang Jiang, Xicheng Wang, and Honglin Li. An accurate metalloprotein-specific scoring function and molecular docking program devised by a dynamic sampling and iteration optimization strategy. Journal of Chemical Information and Modeling, 55(4):833–847, 2015.
86. Marcus CK Ng, Simon Fong, and Shirley WI Siu. Psovina: The hybrid particle swarm optimization algorithm for protein–ligand docking. Journal of bioinformatics and computational biology, 13 (03):1541007, 2015.
87. Oliver Korb, Thomas Stützle, and Thomas E Exner. Plants: Application of ant colony optimization to structure-based drug design. In International workshop on ant colony optimization and swarm intelligence, pages 247–258. Springer, 2006.
88. Bin Li and Daisuke Kihara. Protein docking prediction using predicted protein-protein interface. BMC bioinformatics, 13(1):1–17, 2012.
89. Dima Kozakov, Ryan Brenke, Stephen R Comeau, and Sandor Vajda. Piper: an fft-based protein docking program with pairwise potentials. Proteins: Structure, Function, and Bioinformatics, 65 (2):392–406, 2006.
90. Amr Alhossary, Stephanus Daniel Handoko, Yuguang Mu, and Chee-Keong Kwoh. Fast, accurate, and reliable molecular docking with quickvina 2. Bioinformatics, 31(13):2214–2216, 2015.
91. Nafisa M Hassan, Amr A Alhossary, Yuguang Mu, and Chee-Keong Kwoh. Protein-ligand blind docking using quickvina-w with inter-process spatio-temporal integration. Scientific reports, 7(1): 1–13, 2017.
92. Sergio Ruiz-Carmona, Daniel Alvarez-Garcia, Nicolas Foloppe, A Beatriz Garmendia-Doval, Szilveszter Juhos, Peter Schmidtke, Xavier Barril, Roderick E Hubbard, and S David Morley. rdock: a fast, versatile and open source program for docking ligands to proteins and nucleic acids. PLoS computational biology, 10(4):e1003571, 2014.
93. Gordon Lemmon and Jens Meiler. Rosetta ligand docking with flexible xml protocols. In Computational Drug Discovery and Design, pages 143–155. Springer, 2012.
94. Li-Zhen Sun, Yangwei Jiang, Yuanzhe Zhou, and Shi-Jie Chen. Rldock: a new method for predicting rna–ligand interactions. Journal of chemical theory and computation, 16(11):7173–7183, 2020.
95. Nicolas Majeux, Marco Scarsi, Joannis Apostolakis, Claus Ehrhardt, and Amedeo Caflisch. Exhaustive docking of molecular fragments with electrostatic solvation. Proteins: Structure, Function, and Bioinformatics, 37(1):88–105, 1999.
96. David Ryan Koes, Matthew P Baumgartner, and Carlos J Camacho. Lessons learned in empirical scoring with smina from the csar 2011 benchmarking exercise. Journal of chemical information and modeling, 53(8):1893–1904, 2013.
97. Anita K Nivedha, David F Thieker, Spandana Makeneni, Huimin Hu, and Robert J Woods. Vina-carb: improving glycosidic angles during carbohydrate docking. Journal of chemical theory and computation, 12(2):892–901, 2016.
98. Mathew R Koebel, Grant Schmadeke, Richard G Posner, and Suman Sirimulla. Autodock vinaxb: implementation of xbsf, new empirical halogen bond scoring function, into autodock vina. Journal of cheminformatics, 8(1):1–8, 2016.
99. Ruth Huey, Garrett M Morris, Arthur J Olson, and David S Goodsell. A semiempirical free energy force field with charge-based desolvation. Journal of computational chemistry, 28(6):1145–1152, 2007.
100. Jayashree Srinivasan, Thomas E Cheatham, Piotr Cieplak, Peter A Kollman, and David A Case. Continuum solvent studies of the stability of dna, rna, and phosphoramidate- dna helices. Journal of the American Chemical Society, 120(37):9401–9409, 1998.
101. Maciej Wójcikowski, Pedro J Ballester, and Pawel Siedlecki. Performance of machine-learning scoring functions in structure-based virtual screening. Scientific Reports, 7(1):1–10, 2017.
102. Hao-Yang Liu, Irwin D Kuntz, and Xiaoqin Zou. Pairwise gb/sa scoring function for structure-based drug design. The Journal of Physical Chemistry B, 108(17):5453–5462, 2004.
103. Rodrigo Quiroga and Marcos A Villarreal. Vinardo: A scoring function based on autodock vina improves scoring, docking, and virtual screening. PloS one, 11(5):e0155183, 2016.
104. Susanta Das, Mor Shimshi, Keren Raz, Neta Nitoker Eliaz, Anil Ranu Mhashal, Tamar Ansbacher, and Dan T Major. Enzydock: Protein–ligand docking of multiple reactive states along a reaction coordinate in enzymes. Journal of Chemical Theory and Computation, 15(9):5116–5134, 2019.
105. Mark McGann. Fred pose prediction and virtual screening accuracy. Journal of chemical information and modeling, 51(3):578–596, 2011.
106. Noel M. O'Boyle, Michael Banck, Craig A. James, et al. Open Babel: An open chemical toolbox. Journal of Cheminformatics, 3(1):1–14, 2011. ISSN 1758-2946. doi: 10.1186/1758-2946-3-33.

Methods

VirtualFlow 2.0. VirtualFlow is an in silico platform consisting of three modules (VFLP, VFVS, and VFU) and ready-made ligand libraries for virtual screens. In this section, general platform-wide features have been described that span multiple modules.

Python Code-base. VirtualFlow 2.0 has been entirely rewritten in Python, a programming language that enables the implementation of more sophisticated and advanced code while reducing maintenance work. This rewrite has also made it easier for scientists and programmers to join the open-source project and/or extend VirtualFlow 2.0 independently, as Python is the most widely used programming language in scientific communities. In contrast, the previous version of VirtualFlow was written in Bash.

Parallelism. In addition to being rewritten in Python, a key goal for VirtualFlow 2.0 was to improve parallelism in order to decrease the overall time required for screening. VirtualFlow 2.0 has increased the levels of parallelism compared to the previous version. To help manage this increased parallelism, it maintains the use of *collections,* that group around 1,000 ligands together, reducing the overhead of scheduling individual ligands for processing.

VirtualFlow 2.0 also introduces the concept of *subjobs,* which are atomic units of computational work that include one or more collections for processing. These subjobs are automatically generated by VirtualFlow 2.0 to match the runtime requirements specified in the job configuration. Subjobs are then combined into larger groups called *workunits,* which are designed to allow schedulers to efficiently process the high levels of parallelism generated by VirtualFlow 2.0. These work units are sent as job arrays to the end schedulers for processing. By optimizing for parallelism, the team was able to achieve 5.6 million concurrent vCPUs running VirtualFlow 2.0.

Native Cloud Support. The VFLP and VFVS modules of VirtualFlow can run on computer clusters as well as on the cloud by using a batch system. Currently, two batch systems are supported, Slurm (38) and AWS Batch (39). Slurm is able to run on many cloud infrastructures, including Google Cloud, Microsoft Azure, as well as AWS. Within AWS, VirtualFlow 2.0 uses AWS Batch and S3 object storage. CloudFormation is used to set up VirtualFlow and all necessary components. The AWS-native architecture allows VirtualFlow 2.0 to utilize millions of virtual CPUs (vCPUs) in parallel, while the previous version was only demonstrated to scale up to 160,000 vCPUs.

VirtualFlow 2.0 is able to utilize various features from AWS in addition to supporting traditional clusters, making it easier to use and increasing the scale of computations available to researchers. Some of these features include:

- Scheduling and Orchestration: AWS Batch is a cloud-native scheduler and orchestrator for containerized jobs. As part of running a job, VirtualFlow can make use of AWS Batch to automatically scale cloud infrastructure up and down as needed and run all of the individual subjobs. VirtualFlow can also automatically generate the Docker container used for processing, eliminating the need to understand containerization or other infrastructure.

- Object Storage (Amazon S3): In addition to the usage of standard cluster filesystems, VirtualFlow 2.0 also supports the usage of object storage for both ingesting of datasets and the output of the resulting data from the processing.

- Spot Capacity Awareness and Spot Resilience: Spot capacity is unused computing capacity that cloud providers make available at significantly discounted prices compared to normal on-demand access. In exchange for a lower price, a workload running on a spot instance may be reclaimed (stopped) with a short warning. VirtualFlow 2.0 is designed to take advantage of this capacity model by using shorter running subjobs (approximately 30 minutes) and automatically re-running jobs that have prematurely exited through integration with AWS Batch. Additionally, through integration with AWS Batch, VirtualFlow is able to select instances that are less likely to be interrupted through the "spot capacity aware" scheduling method.

The architecture of VirtualFlow 2.0 when deployed on AWS is shown in Extended Data Figure 1. The architecture makes use of multiple *Compute Environments*, each of which may contain thousands of instances (compute nodes). The instances are automatically allocated and deallocated when the job starts and ends, respectively. Individual subjobs are allocated to the instances by AWS Batch until all subjobs have been completed. These subjobs run as Docker containers on the instances.

ARM. VirtualFlow is also able to run on the compute ARM architecture when using ARM-compatible choices in the settings. VFLP is fully ARM compatible, and VFVS is ARM compatible when docking programs are used that either make the source code available or provide binaries for ARM. ARM-based computing infrastructure can be more cost-efficient than traditional x86-based architectures.

Enamine REAL Space. The REAL Space of Enamine (version 2022q1-2) was prepared using VFLP 2.0. JChem Suite was used for neutralizing the molecules, stereoisomer enumeration, tautomer generation, protonation state prediction, 3D coordinate calculation, and the calculation of multiple molecular properties, JChem Suite 18.20.0, ChemAxon (https:

//www.chemaxon.com). Open Babel (version 2.3.2) was used as backup program for protonation state prediction, 3D coordinate calculation (42), and target format conversion (all formats except SELFIES). The pip-installable SELFIES package (https://github.com/aspuru-guzik-group/selfies) was used for the conversion of molecules between SMILES and SELFIES. An overview of the processing steps and preparation methods used is shown in Extended Data Table 2.

Multiple Formats. VirtualFlow provides the REAL Space library in ready-to-dock 3D formats including PDB, PDBQT, MOL2, and SDF. These formats are compatible with a wide range of docking programs for small molecules. The total size of the ready-to-dock library is approximately 50TB in compressed form for each of these formats, totaling around 200 terabytes in compressed form (uncompressed, the library is around 2 petabytes). In addition to these ready-to-dock formats, the REAL Space library is also provided in the SELFIES (34), SMILES (43), and Parquet formats. These formats are useful for postprocessing the results of virtual screening (e.g., for easy representation of each molecule), and they can also be used as standalone libraries for other purposes such as in the AI/ML community, where SMILES and SELFIES are the standard molecular input file formats. The Parquet format of the REAL Space only stores the properties of the molecules, which can be useful for fast library-wide queries.

VirtualFlow for Ligand Preparation (VFLP). VFLP 2.0 is the module of VirtualFlow 2.0 dedicated to the preparation of ligand libraries into a ready-to-dock format and can handle libraries of any size, including ultra-large libraries. VFLP requires a batch system and supports SLURM as well as AWS Batch as batch systems. VFLP is configured via a configuration file (an example configuration file is shown in Supplementary Listing G). Details on the workflow are available in Supplementary Information Section B. The new version of VFLP has the following features.

Preparation into Ready-To-Dock Format. The primary task of VFLP 2.0 is to prepare ligands into a ready-to-dock format. To accomplish this, VFLP 2.0 desalts the original ligands (in case they are salts), neutralizes them, generates stereoisomers and tautomers, predicts the protonation states, computes the minimum-potential energy conformation of the ligand (3D coordinates), and converts them into the desired target formats. In total over 100 output formats are supported. A detailed overview of the processing steps can be seen in Extended Data Figure 7. The support input formats are SMILES, SELFIES, as well as amino acid sequences. VFLP 2.0 uses external tools such as ChemAxon's JChem, Open Babel, and RDKit (44), to carry out the preparation steps. For many of the preparation steps, the user can choose which external tool should be used. Details on which tools are available for each processing step can be found in Extended Data Table 3.

Molecular Properties Calculation. VFLP 2.0 enables researchers to calculate a variety of physicochemical properties for the molecules in the input ligand library, in addition to its primary task of preparing them for molecular docking. By calculating properties such as QED (Quantitative Estimate of Druglikeness), logP, molecular weight, and others (a full list of supported properties can be found in Extended Data Table 2), researchers can more effectively select and prioritize ligands, which can help reduce costs and the time required for screens using VFVS. VFLP 2.0 uses mostly external tools to calculate the molecular properties, such as ChemAxon's JChem (https://chemaxon.com/calculators-and-predictors, Open Babel (42), and RDKit (for details see Extended Data Table 3). By combining the results of VFLP with those of VFVS, researchers can gain valuable insights into the structural and chemical characteristics of their ligands relative to specific target proteins, which can help them design and optimize new molecules with improved affinity and specificity.

Automatic Tranche Assignments. In VirtualFlow, ligand libraries are organized into tranches. Each tranche corresponds to a property of the ligands and is divided into multiple intervals or categories. For example, the molecular weight of the ligands can be used as a tranche type, and can then be partitioned into several categories or intervals such as 0-200, 200-300, 300-400, 400-500, and 500-∞ daltons. Multiple tranches allow for the partitioning of a ligand library into a multidimensional grid or table, with one dimension corresponding to each tranche type (ligand property).

While the previous version of VirtualFlow required that the input ligand collection was already organized and structured in the tranche format, VirtualFlow 2.0 can automatically reorganize the input ligand library into a new tranche output format based on the computation of specific molecular properties of the ligands (see the corresponding paragraph above). This feature is useful because most ligand libraries (e.g., from chemical compound vendors) are initially in the SMILES format and not in the tranche format. VFLP can then be used to prepare the entire library in the multidimensional grid/tranche format for VFVS.

Support of SELFIES. SELF-referencing Embedded Strings (SELFIES) (34, 35) has become one of the most popular line notation formats for small molecules within machine-learning communities. The primary reason is that SELFIES are fully robust, meaning that every SELFIES string corresponds to a valid molecule, which is not the case for SMILES. Thus generative ML models cannot create invalid molecules when using SELFIES. VFLP now supports SELFIES as an input format for the molecules in addition to SMILES. Additionally, the entire REAL Space is free to be downloaded in the SELFIES format, enabling the creation of powerful datasets.

Stereoisomer Enumeration. One of the main tasks of VFLP is to prepare ligand libraries from unprepared formats (e.g., SMILES) into ready-to-dock formats. The new version of VFLP enables the enumeration of stereoisomers for the ligands,

which was not possible in the previous version. Accurate stereochemistry is essential for virtual screenings using structure-based molecular docking, but many ligand libraries in unprepared formats (e.g., from chemical compound vendors) do not contain stereochemical information. The ability to enumerate stereoisomers allows VFLP to include this information in the prepared ligand library, ensuring that the virtual screening results are accurate.

Extended Open Source Support. The previous version of VFLP required external tools (from ChemAxon) that are not open source for some of the steps when preparing ligands in a ready-to-dock format. The new version provides the option to choose open-source alternatives for each of the steps to prepare molecules into a ready-to-dock format. In particular, Open Babel and RDKit are supported by VirtualFlow 2.0 as open-source alternatives.

Enhanced Speed. VFLP 2.0 has been improved in terms of speed compared to the original version. Firstly, this has been achieved by VFLP 2.0 using its own Java code to call the necessary functions of ChemAxon's JChem package via Nailgun, which reduces the number of calls to Nailgun significantly. Additionally, timeout parameters that can be adjusted by users have been added to the control file, allowing VFLP to skip ligands that require unusually long processing times during preparation. These improvements make VFLP more efficient and able to handle larger ligand libraries.

Enhanced Quality Checks. VFLP checks the validity of each ligand that it prepares in the ready-to-dock format by calculating the potential energy of the ligand in 3D format using Open Babel. This allows VFLP to automatically remove ligands that have an excessively high potential energy, which can be an indication that the compound was damaged or corrupted during the preparation process. This helps to ensure the integrity and accuracy of the prepared ligand library.

Application. VFLP was used to prepare the REAL Space (version 2022q1-2). It was run on AWS using over 5.6 million Intel vCPUs in parallel. Spot instances were used, and the used capacity was sustained without significant preemption. In total less than 0.1 percent of the used CPU hours were interrupted due to preemption. The scaling behavior of VFLP 2.0 was found to be perfectly linear (as shown in Extended Data Figure 2). This demonstrates the efficiency and effectiveness of VFLP in preparing large ligand libraries in a ready-to-dock format.

VirtualFlow for Virtual Screening (VFVS). VFVS is the module of VirtualFlow dedicated to ultra-large virtual screens but can also screen libraries of smaller sizes. It requires a batch system and supports SLURM as well as AWS Batch as batch systems. VFVS is configured via a configuration file (an example configuration file is shown in Supplementary Listing H). VFVS and VFU share many virtual screening features, which are described in a later section. VFLP 2.0 and VFVS 2.0 share the same parallelization mechanisms when using Slurm or AWS Batch, and therefore the scaling behavior of VFVS 2.0 is expected to be the same as for VFLP, which was shown to be perfectly linear up to 5.6 million CPUs (see also Extended Data Figure 2).

Parquet Output Format. VFVS has the ability to store output data in the Apache Parquet format, an open-source column-oriented data storage format. When using this format, VFVS can take advantage of any database and query system that can read this format, including Amazon Athena, a serverless service that allows users to efficiently query output score files stored in Amazon S3 (object storage) on AWS. Additionally, VFVS has a feature for automatic postprocessing ATG Prescreens using Amazon Athena, enabling researchers to efficiently prepare the input data for the ATG primary screens. Tools are provided with VFVS to facilitate common queries, but researchers can also create their own queries using the open-source format of the dataset.

ATG-VSs and Benchmarks After an ATG Prescreen is completed, the tranches to be selected for the ATG Primary Screen can be selected in multiple ways. Either the tranches of the REAL Space can be selected individually, allowing for arbitrary selection patterns within the multidimensional grid. Alternatively, the shape of the selected tranches in the multidimensional tranche matrix can be restricted to hyperrectangles, which corresponds to selecting the best intervals/tranches of each dimension/property separately, and then multiplying/combining the selected tranches of each dimension to obtain the hyperrectangular selection. The total number of molecules to be selected for the ATG Primary Screen can be chosen by the user.

The ATG-VS benchmarks shown in Figure 4, and Extended Data Figures 8 and 9 were run using the target proteins shown in Extended Data Table 4. The docking program used was QuickVina 2. The benchmarks were run in AWS using Spot instances using Intel CPUs, and up to 400,000 CPUs were used in parallel. After the ATG Prescreen, the tranches for the ATG Primary Screen were selected using the hyperrectangle method.

VirtualFlow Unity. VirtualFlow Unity is the all-in-one version of VirtualFlow, combining both ligand preparation and virtual screening into a single streamlined workflow. It is the third module besides VFLP and VFVS. This new module adds many conveniences for workflows where both ligand preparation and virtual screenings have to be carried out in tandem. Notably, VFU is designed to be run as a package, supporting approximately 1500 docking methods that can be run independently of job schedulers (see Supplementary Data Table 1). A conceptual overview of the workflow of VFU can be found in Extended Data Figure 10, and a detailed UML workflow diagram is shown in Supplementary Figure 10. Specifically, as input, a user

specifies the molecule to be docked (as a SMILES, SELFIES, or amino-acid sequence), the protein file, the docking site, and the docking program to be run. Based on the user's choices, error checks in the form of missing docking-choice-specific files are performed, and calculations are run. As a result of a successful calculation, the docking score and pose are returned. VFU can be run in standalone mode, where it is run via the command line and specifies the options via a configuration file (an example configuration file is shown in Supplementary Data Figure I). This mode is for example useful for scientists in the drug discovery communities who want to carry out molecular dockings in the most convenient and simple way, without requiring computational expertise, while having a large number of docking methods available. In addition, VFU can be run in API mode to directly interface with other programs and code, which can for instance be useful for the machine learning communities that develop new methods for drug discovery.

Platform-wide Virtual Screening Features (VFU and VFVS). The VFU and VFVS modules support the same docking methods and therefore share all docking-specific features.

Supported Docking Methods. The initial version of VirtualFlow supported 7 different docking programs: AutoDock Vina, Vina Carb, VinaXB, QuickVina 02, QuickVina W, Smina, and ADFR. Shortly after, support for GWO Vina and PLANTS was added (45, 46)). In VirtualFlow 2.0, this has been extended to approximately 40 docking methods, listed in Tables 5, 6, 7. These include software that performs both pose prediction and scoring (Table 5), solely pose prediction (Table 7) and only scoring (Table 6). Notably, within VirtualFlow 2.0, the scoring functions can be combined with pose prediction methods, giving rise to approximately 1500 methods (listed in Table 1). We note that many of the newly supported docking software possess special features such as protein-protein/RNA docking, explicit parameterization of solvents/metals, and covalent docking. In Tables 5,6,7, we describe these as "Special features". Additionally, a key consideration for ULVS is the timing required for carrying out a single calculation. As such, in Figure 11, we provide timing estimates for running a single calculation, averaging over 25 independent runs for all docking programs.

Special Types of Ligands. In addition to traditional small molecule ligands, VirtualFlow 2.0 now supports the use of peptides, protein ligands, and covalently-bonded ligands in VFVS. Peptides are of increasing interest as therapeutics, and several docking programs already support their use. For protein-protein/RNA docking, specialized programs require two protein/RNA targets in order to generate co-complexes. Covalent docking involves reduced flexibility for certain parts of the input molecule during pose sampling, and a few specialized docking programs support this ligand type (see Extended Data Table 5). These new ligand types expand the scope of VirtualFlow and allow for even more diverse virtual screenings.

Enhanced Performance. The QuickVina 2 molecular docking tool is used to accelerate AutoDock Vina with no loss of accuracy (47). QuickVina 2 execution was profiled in an effort to improve performance even further. To establish baseline performance, the open-source GitHub repository was cloned and the qvina_1buffer branch was compiled using the GNU C++ compiler (version 9.4), the Boost C++ libraries (version 1.79, available from `https://www.boost.org`), and the default parameters (see `https://qvina.github.io/compilingQvina2.html` for the build instructions). A single docking experiment of Avanafil binding to unsolvated 5FNQ was run through the Intel VTune Profiler to look for performance bottlenecks. Unfortunately, the execution profile was flat. The most time-consuming function accounted for less than 20% of the total run-time. Tuning the source code would give rapidly diminishing returns at the risk of affecting accuracy, so compiler-level optimization was the best option to improve performance. QuickVina 2 was rebuilt using the Intel oneAPI DPC++/C++ Compiler (version 2022.2.0.20220730) and aggressive optimization: automatic vectorization, interprocedural optimization, and relaxed floating-point constraints. Profile-guided optimization was also tried, but it did not improve performance. The final executable was 18% faster than the default builds with no loss of accuracy. The new binaries are provided on the QuickVina and VF homepages.

GPU Support. Several of the docking programs support graphics processing units (GPUs), which can strongly reduce computation time and cost. For pose prediction, these include softwares: AutoDock-GPU (48), QVina2-GPU (49), QVina2-W-GPU (49), Vina-GPU (49). Additionally, deep-learning methods such as TANKBind (13), DiffDock (50), EquiBind (51) use GPU-compatible neural networks for direct pose prediction of several ligands in parallel. Additionally, Gnina (52), NNScore 2.0 (53), DeepAffinity (54) and DeepBindRG (55) use GPU-compatible deep learning models for scoring protein-ligand complexes. VirtualFlow 2.0 supports GPU compatibility of all these software. The relative speedups of these methods for single calculations can be compared within Figure 11.

Molecular Dynamics-Based Methods. Molecular docking is a computationally fast way to predict binding poses and binding-free energies, but it has sometimes trouble to distinguish between compounds with similar binding affinities. Molecular docking algorithms often prioritize speed over accuracy by using fast pose generation and approximate scoring functions to save time, especially when screening extensive collections of compounds against a protein target. This trade-off reduces the effectiveness of the algorithms. More precise algorithms based on molecular dynamics simulations are sometimes used downstream of docking in drug discovery processes to improve the number of accurate predictions. The combination of molecular docking and molecular dynamics can yield results near experimentally determined crystal structures (56, 57). VFVS 2.0 includes

methods for selecting and scoring poses based on molecular dynamics simulations, such as MM/PBSA and MM/GBSA, for evaluating docking poses and predicting binding affinities (58). MM/PBSA and MM/GBSA are popular for binding free energy prediction because they are more accurate than molecular docking and less computationally intensive than alchemical methods. VirtualFlow 2.0 also supports Binding Pose Metadynamics (BPMD), which uses a stability score based on the resistance of the ligand to perturbation away from the initial pose to rerank predicted binding poses from docking (56, 59). The OpenBPMD algorithm can accurately predict binding poses within 2 Å RMSD of the crystallographic complex structure for many systems and efficiently rerank docked poses. The MM/PBSA, MM/GBSA, and BPMD methods can be used within VirtualFlow as rescoring and pose refinement methods.

Enhanced Quality Checks. The docking pose of each ligand is checked for corruption by evaluating the coordinates and computing the potential energy of the docked compound with Open Babel. Any corrupted compounds are automatically removed.

Code Availability. The source of VirtualFlow 2.0 is released under the GNU GPL v2.0, a free and open-source license. The code can be accessed via GitHub. The VirtualFlow GitHub Organization can be found at https://github.com/VirtualFlow, which contains multiple repositories. The VFLP 2.0 and the VFVS 2.0 modules of VirtualFlow 2.0 can be found in the branches "python-develop2" of the VFLP and the VFVS GitHub repositories, respectively. The module VirtualFlow Unity can be found in the GitHub repo named VF-Unity.]

Data Availability. The ready-to-dock REAL Space from Enamine can be accessed online via an interactive web interface available on the VirtualFlow homepage http://virtual-flow.org/, or via AWS Open Data (https://aws.amazon.com/opendata.

Supplementary Information. Supplementary information is available in the online version of the paper.

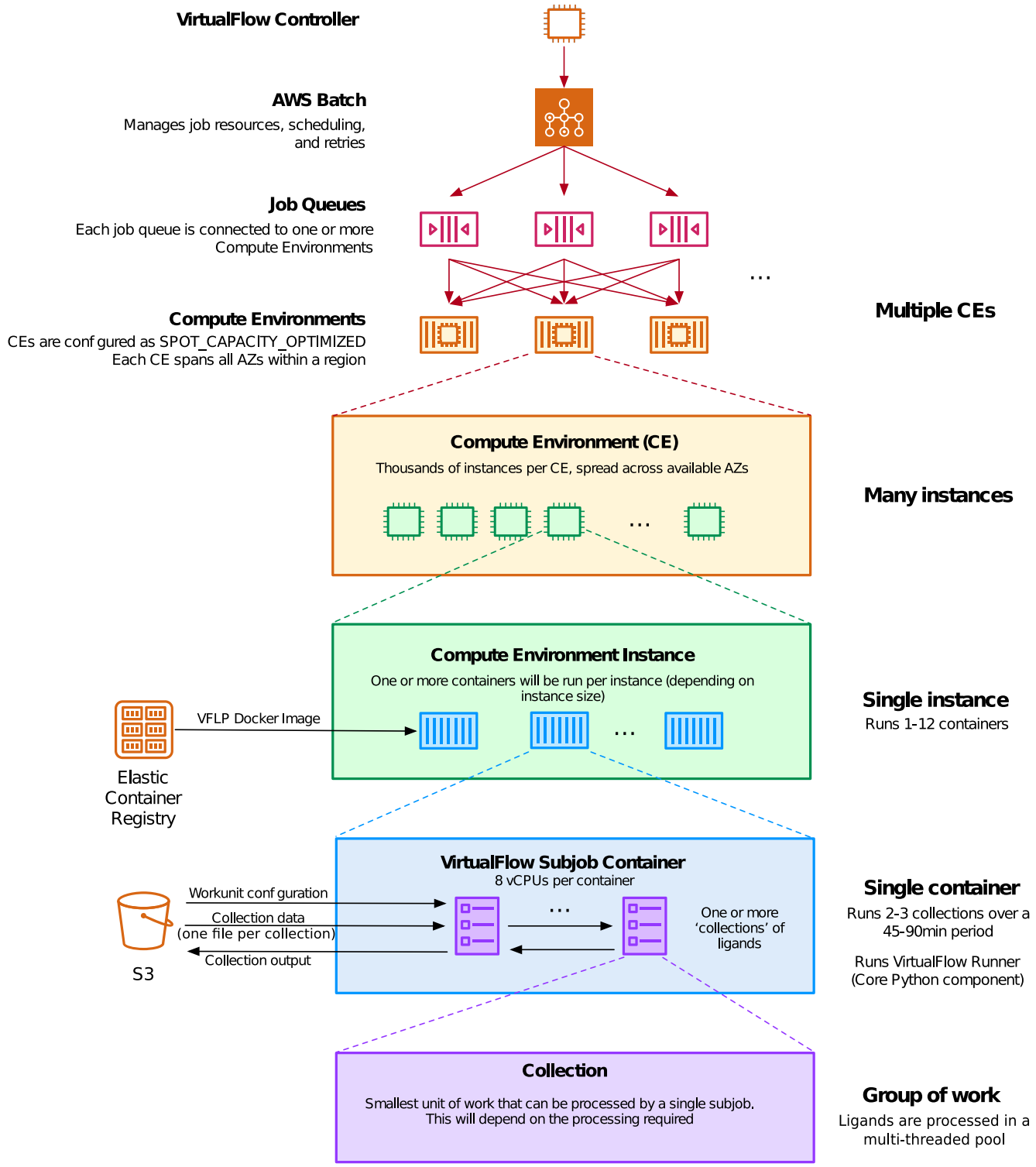
Acknowledgements. The authors would like to thank ChemAxon for a free academic license for JChem. AK.N. acknowledges funding from the Bio-X Stanford Interdisciplinary Graduate Fellowship (SGIF). AK.N. thanks Anshul Kundaje and Michael Bassik for their support. The authors thank AWS for research credits for the AWS Cloud and thank Nelson Gonzales, Stephen Litster, Mayank Thakkar, Robert Wang, Arun Subramaniyan, and Christine Tsien Silvers for technical and administrative support within AWS. The authors would like to thank Akanksha Bilani and Caroline Rhoades from Intel for their support, and Brigitte Klein for proofreading the manuscript.

Competing Interests. The authors declare the following competing interests: D.S.R. works for Enamine, a company that is involved in the synthesis and distribution of drug-like compounds. Y.S.M is a scientific advisor to Enamine and CEO of Chemspace LLC. C.G., H.A., and G.W. are cofounders of Quantum Therapeutics Inc., a company that uses computational methods for drug development. C.G. and G.W. are cofounders of Virtual Discovery Inc., which is a fee-for-service company for computational drug discovery. M.G. works for Google, a company providing cloud computing services. M.K. and P.-Y.A. work for Amazon Web Services, a company providing cloud computing services.

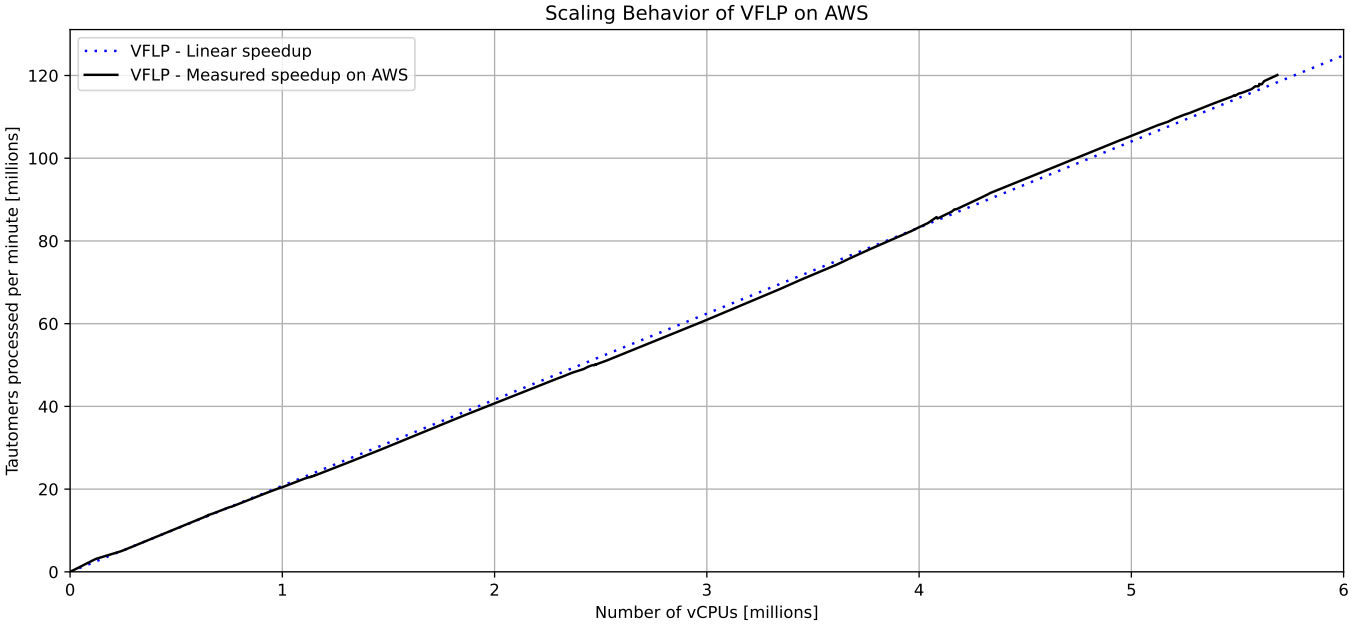
Extended Data

Feature	VirtualFlow 1.0	VirtualFlow 2.0
General Features		
Codebase	Bash	Python
Native cloud support	not available	available (AWS)
Linear scaling behavior	demonstrated up to 160,000 CPUs	demonstrated up to 5.6 million CPUs
VirtualFlow Ligand Libraries		
Total size of ligand library	1.4 billion	68.7 billion
Additional predicted ligand properties	0	28
PDB format	not available	available
PDBQT format	available	available
SDF format	not available	available
MOL2 format	not available	available
SMILES format	not available	available
SELFIES format	not available	available
Parquet format	not available	available
Sparse version (for ATG-VSs)	not available	available
Interactive web interface	yes (6 dimensions)	yes (18 dimensions)
Accessability	VirtualFlow website	VirtualFlow website and AWS Open Data
VirtualFlow For Ligand Preparation (VFLP) Module		
Stereoisomer enumeration	not supported	supported
External open-source tools for ligand preparation	limited support	extended support
SMILES as input format	supported	supported
SELFIES as input format	not supported	supported
Amino acid sequences as input format	not supported	supported
SELFIES as output format	not supported	supported
Prediction of physical molecular properties	not supported	supported (27 properties)
Automatic multidimensional matrix generation	not supported	extended support
Quality control	limited support	extended support
Enhanced speed	limited support	extended support
VirtualFlow For Virtual Screening (VFVS) Module		
Adaptive target-guided virtual screens	not supported	supported
Docking methods	10	1500
Deep learning-based scoring functions	not supported	supported
Deep learning-based pose prediction methods	not supported	supported
Molecular dynamics-based methods	not supported	supported
Protein as target	supported	supported
RNA as target	not supported	supported
DNA as target	not supported	supported
Covalent docking	limited support	extended support
Protein ligands (protein-receptor docking)	not supported	supported
Peptide ligands	not supported	supported
Parquet output format	not supported	supported
Docking quality control	not supported	supported
ARM supported	not supported	supported
Enhanced x86 performance	not available	available (for several docking methods)
GPU support	not supported	supported
VirtualFlow Unity (VFU) Module		
Standalone mode	not available	available
API mode	not available	available

Extended Data Table 1. Comparision of Features Supported by VirtualFlow 1.0 and VirtualFlow 2.0. Features are grouped by the VirtualFlow module: General, VFLP, VFVS, and VF Unity. VirtualFlow 2.0 has significantly more features than the original version.



Extended Data Fig. 1. Compute Stack deployed by VirtualFlow in AWS. The login node is used by the user to control the workflow in AWS via the workload manager AWS Batch. AWS Batch has multiple Job Queues to which jobs are submitted by VirtualFlow, and each job queue deploys multiple Compute Environments that can spin up and use a large number of virtual machines (instances) of specific types. On each instance, one or more subjobs are run, and each subjobs deploys a docker container that carries out the computations of VirtualFlow by executing VirtualFlow Runner (a core python component of VirtualFlow). Each subjob typically processes a few ligand collections, where each ligand collection contains multiple ligands (e.g. 1000).



Extended Data Fig. 2. Scaling behavior of VFLP of on AWS. The scaling behavior is perfectly linear even when using extremely large numbers of CPUs. Up to 5.6 million vCPUs were used in parallel across multiple AWS regions. As can be seen in the plot, the scaling behavior is perfectly linear, implying there is no slowdown even when using extremely large numbers of vCPUs.

Index	Property	VirtualFlow Parameter	Preparation Method	Used for 18-Dimensional Matrix
1	molecular weight	mw_file	Input library	yes
2	logP	logp_jchem	JChem	yes
3	hydrogen bond acceptor count	hba_jchem	JChem	yes
4	hydrogen bond donor count	hbd_jchem	JChem	yes
5	rotatable bond count	rotb_jchem	JChem	yes
6	topological polar surface area	tpsa_jchem	JChem	yes
7	molecular refractivity	mr_jchem	JChem	yes
8	solubility [molar]	logs	JChem	yes
9	number of bonds	bondcount_jchem	JChem	no
10	aromatic proportion	ap	JChem	yes
11	ring count	ringcount	JChem	no
12	aromatic ring count	aromaticringcount	JChem	no
13	logD	logd	JChem	no
14	formal charge	formalcharge	VirtualFlow	yes
15	positive charge count	positivechargecount	VirtualFlow	yes
16	negative charge count	negativechargecount	VirtualFlow	yes
17	fraction of sp3 carbon atoms	fsp3_file	Input library	yes
18	total atom count	atomcount_jchem	JChem	no
19	heavy atom count	heavyatomcount_file	Input library	no
20	chiral center count	chiralcentercount	JChem	yes
21	doublebond stereoisomer count	doublebondstereoisomercount	JChem	yes
22	halogen atom count	halogencount	JChem	yes
23	sulfur atom count	sulfurcount	JChem	yes
24	nitrogen and oxygen atom count	NOcount	JChem	no
25	electronegative atom coun	electronegativeatomcount	JChem	no
26	Enamine compound class	enamine_compound_class	Input library	yes
27	quantitative estimate of druglikeness (QED)	qed_rdkit	RDKit	no
28	Murcko scaffold	scaffold_rdkit	RDKit	no

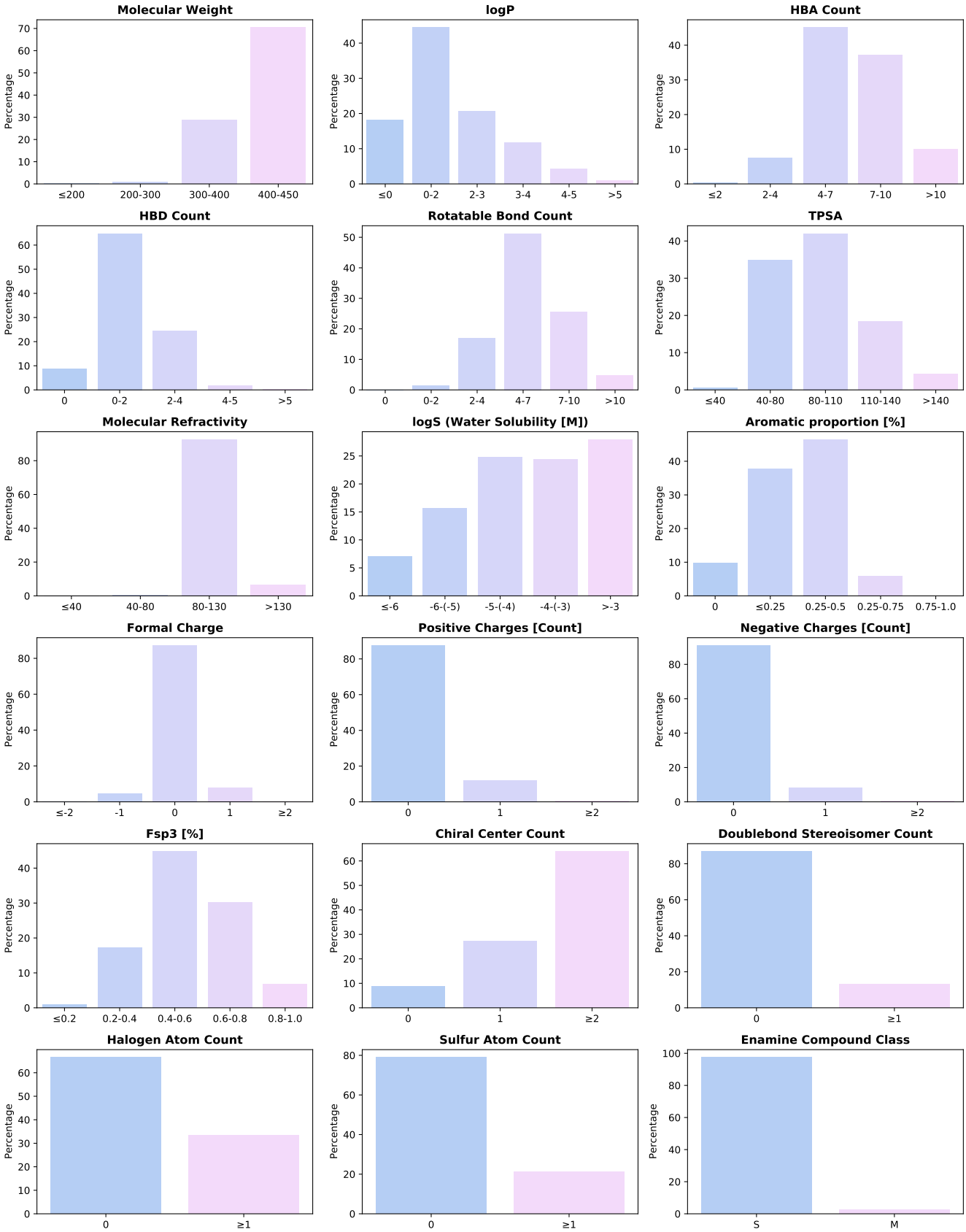
Extended Data Table 2. Calculated Properties of Ligands in REAL Space. All properties of the ligands that were determined/calculated during the preparation of the REAL Space (version 2022q1-2) with VFLP 2.0. A few properties (molecular weight, fsp3, heavy atom count, Enamine compound class) were available in the input library and thus could be directly used without any calculations.

Molecular Property (X)	Tranche ID	A	B	C	D	E	F
MW	1	**X ≤ 200**	**200 < X ≤ 300**	**300 < X ≤ 400**	**400 < X ≤ 450**	**X > 450**	
		3.8E+6	564.9E+6	19.8E+9	48.4E+9	0.0E+0	
logP	2	**X ≤ 0**	**0 < X ≤ 2**	**2 < X ≤ 3**	**3 < X ≤ 4**	**4 < X ≤ 5**	**X > 5**
		12.5E+9	30.5E+9	14.2E+9	8.0E+9	2.8E+9	706.3E+6
HBA count	3	**X ≤ 2**	**2 < X ≤ 4**	**4 < X ≤ 7**	**7 < X ≤ 10**	**X > 10**	
		270.9E+6	5.1E+9	31.0E+9	25.5E+9	6.8E+9	
HBD count	4	**X = 0**	**0 < X ≤ 2**	**2 < X ≤ 4**	**4 < X ≤ 5**	**X > 5**	
		6.0E+9	44.5E+9	16.8E+9	1.2E+9	277.2E+6	
Rotatable bond count	5	**X = 0**	**0 < X ≤ 2**	**2 < X ≤ 4**	**4 < X ≤ 7**	**7 < X ≤ 10**	**X > 10**
		2.0E+6	966.3E+6	11.6E+9	35.2E+9	17.6E+9	3.3E+9
TPSA	6	**X ≤ 40**	**40 < X ≤ 80**	**80 < X ≤ 110**	**110 < X ≤ 140**	**X > 140**	
		377.6E+6	24.0E+9	28.9E+9	12.6E+9	2.9E+9	
Molecular refractivity	7	**X ≤ 40**	**40 < X ≤ 80**	**80 < X ≤ 130**	**X > 130**		
		79.2E+3	540.6E+6	63.6E+9	4.6E+9		
logS	8	**X ≤ -6**	**-6 < X ≤ -5**	**-5 < X ≤ -4**	**-4 < X ≤ -3**	**X > -3**	
		4.9E+9	10.7E+9	17.1E+9	16.8E+9	19.2E+9	
Aromatic proportion	9	**X = 0**	**0 < X ≤ 0.25**	**0.25 < X ≤ 0.5**	**0.25 < X ≤ 0.75**	**0.75 < X ≤ 1.0**	
		6.8E+9	25.9E+9	31.9E+9	4.0E+9	49.7E+6	
Formal charge	10	**X ≤ -2**	**X = -1**	**X = 0**	**X = 1**	**X ≥ 2**	
		158.3E+6	3.1E+9	59.9E+9	5.4E+9	140.9E+6	
Positive charge count	11	**X = 0**	**X = 1**	**X ≥ 2**			
		60.1E+9	8.2E+9	341.7E+6			
Negative charge count	12	**X = 0**	**X = 1**	**X ≥ 2**			
		62.5E+9	5.7E+9	484.1E+6			
Fsp3	13	**0 ≤ X ≤ 0.2**	**0.2 < X ≤ 0.4**	**0.4 < X ≤ 0.6**	**0.6 < X ≤ 0.8**	**0.8 < X ≤ 1.0**	
		647.0E+6	11.9E+9	30.8E+9	20.7E+9	4.7E+9	
Chiral center count	14	**X = 0**	**X = 1**	**X ≥ 2**			
		6.1E+9	18.7E+9	43.9E+9			
Doublebond stereoisomer count	15	**X = 0**	**X ≥ 1**				
		59.6E+9	9.1E+9				
Halogen atom count	16	**X = 0**	**X ≥ 1**				
		45.7E+9	23.0E+9				
Sulfur atom count	17	**X = 0**	**X ≥ 1**				
		54.2E+9	14.5E+9				
Enamine compound class	18	**X = S**	**X = M**				
		66.9E+9	1.8E+9				

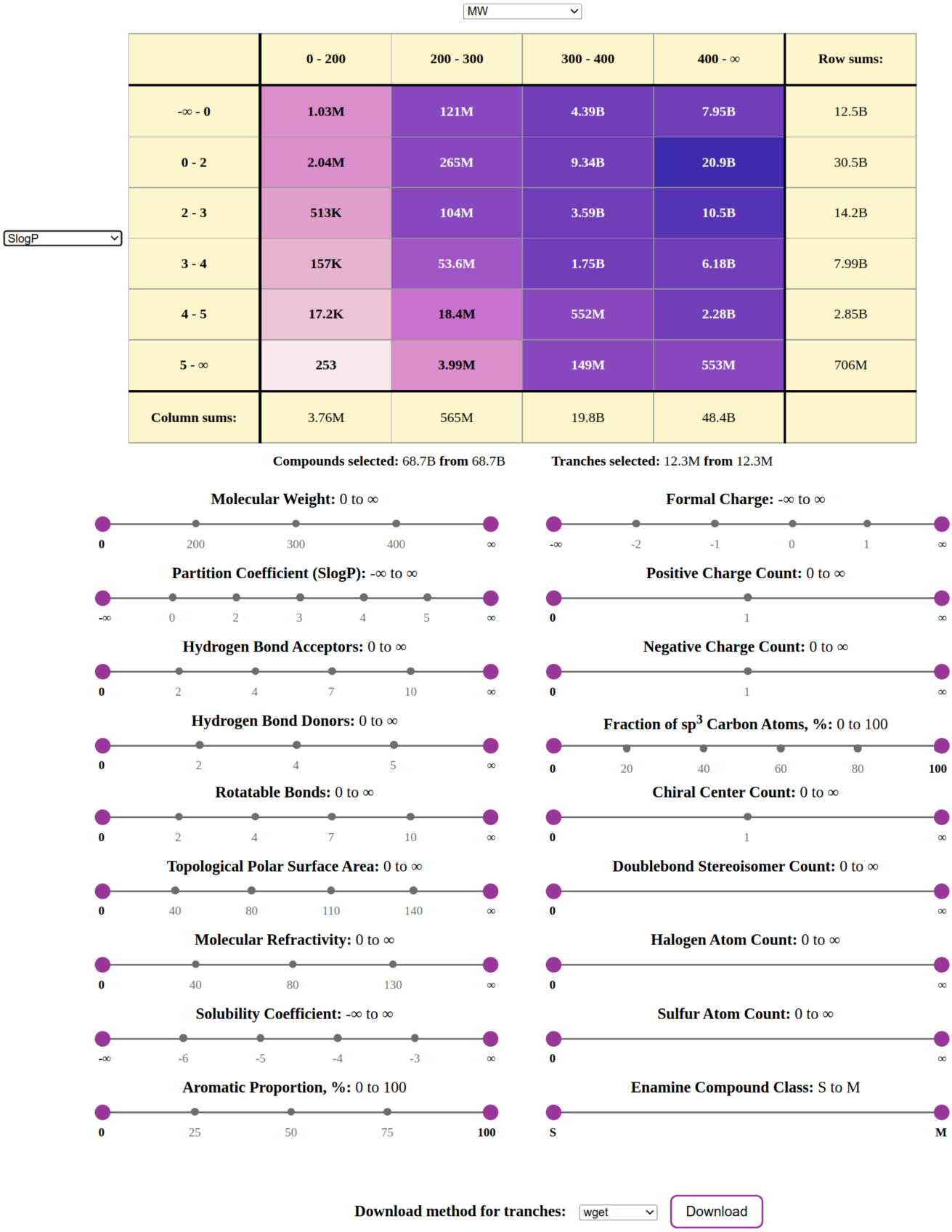
Color Scale

0 100M 50B

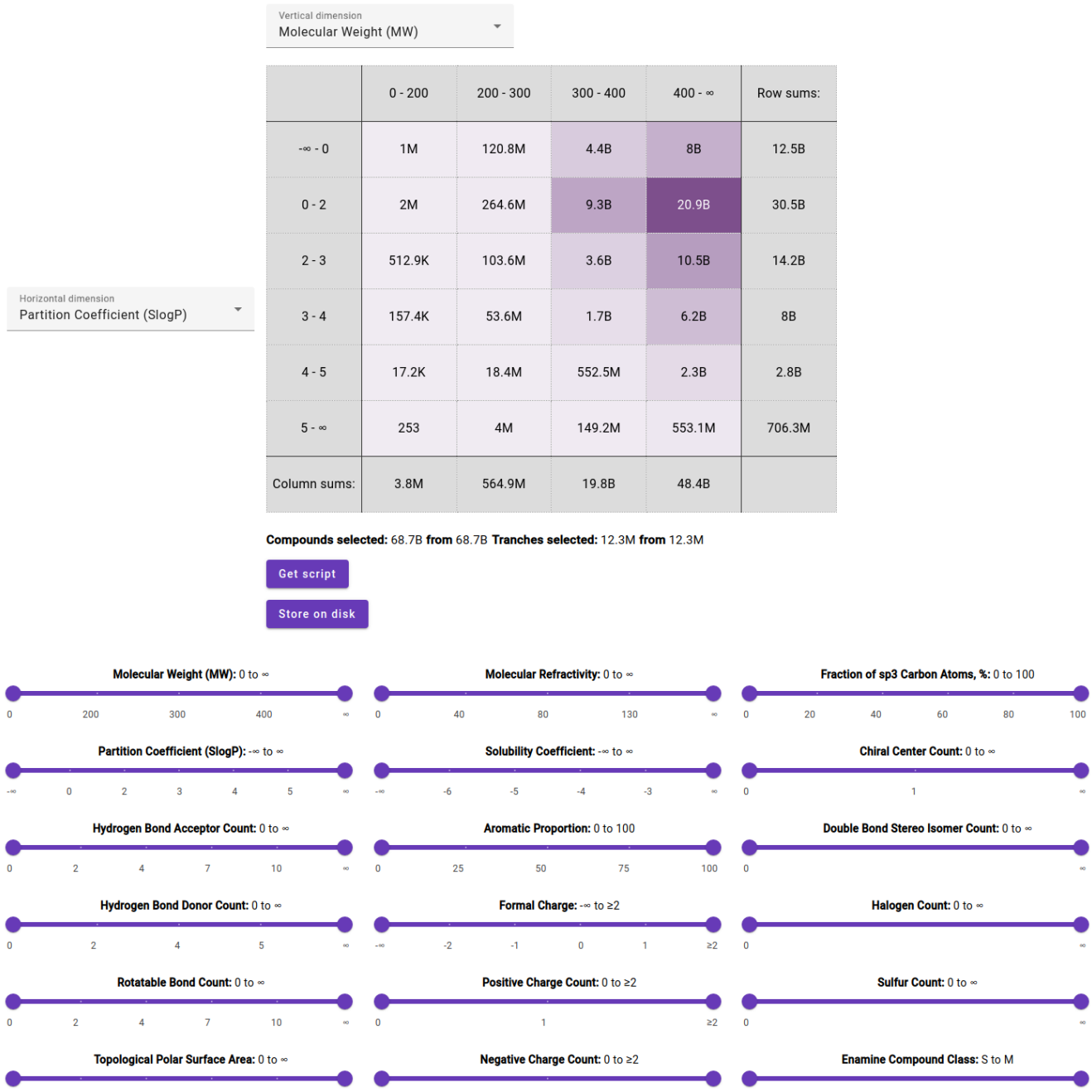
Extended Data Fig. 3. Overview of 18-dimensional tranche table. Molecular properties associated with each tranche, the partitions that are used for each tranche property, and the number of compounds per partition interval for each property (colored as a heat map).



Extended Data Fig. 4. Overview of the distribution of the molecular properties of REAL Space library. Each subplot shows the distribution of the molecular property associated with each dimension of the 18-dimensional grid into which the REAL Space was partitioned.

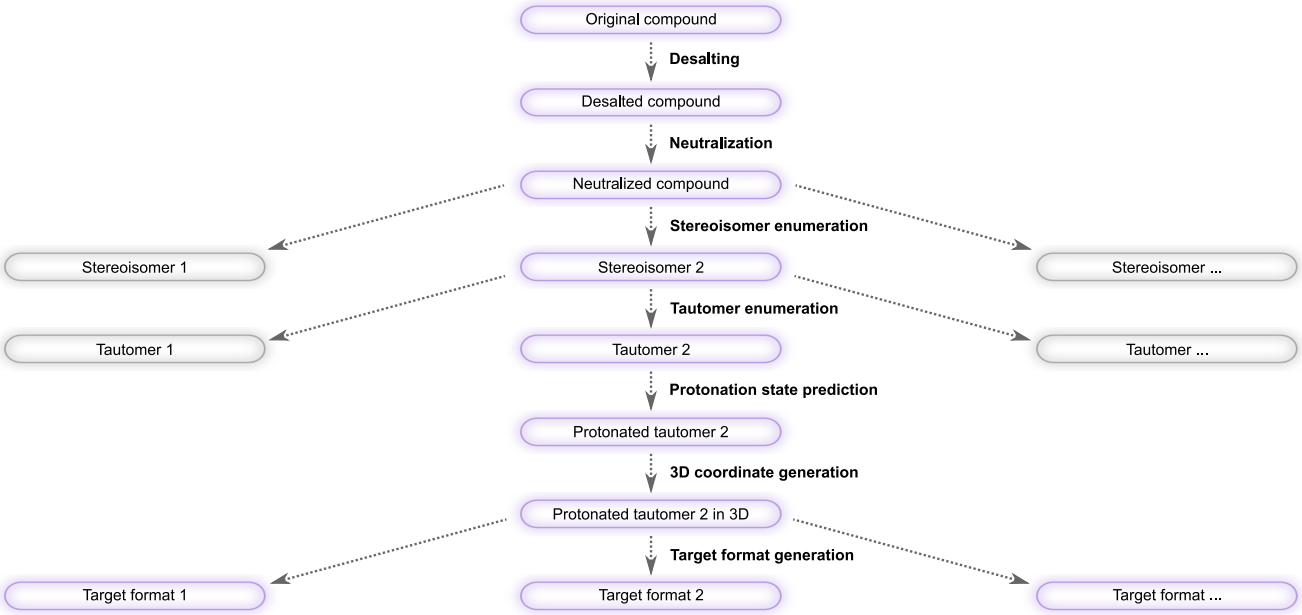


Extended Data Fig. 5. Webinterface (version 1) of the REAL Space. Shown is an interactive web interface that is made available on the VirtualFlow homepage (https://virtual-flow.org/) that allows selecting subsets of the REAL Space partitioned into an 18-dimensional matrix. Each dimension corresponds to a molecular property and has a slider that can be adjusted to select the desired property ranges. The number of selected molecules is updated automatically in the web interface. Effectively, the sliders allow the selection of multidimensional rectangles within the 18-dimensional matrix.



	0 - 200	200 - 300	300 - 400	400 - ∞	Row sums:
-∞ - 0	1M	120.8M	4.4B	8B	12.5B
0 - 2	2M	264.6M	9.3B	20.9B	30.5B
2 - 3	512.9K	103.6M	3.6B	10.5B	14.2B
3 - 4	157.4K	53.6M	1.7B	6.2B	8B
4 - 5	17.2K	18.4M	552.5M	2.3B	2.8B
5 - ∞	253	4M	149.2M	553.1M	706.3M
Column sums:	3.8M	564.9M	19.8B	48.4B	

Extended Data Fig. 6. Webinterface (version 2) of the REAL Space. Shown is an interactive web interface that is made available on the VirtualFlow homepage (https://virtual-flow.org/) that allows selecting subsets of the REAL Space partitioned into an 18-dimensional matrix. Each dimension corresponds to a molecular property and has a slider that can be adjusted to select the desired property ranges. The number of selected molecules is updated automatically in the web interface. Effectively, the sliders allow the selection of multidimensional rectangles within the 18-dimensional matrix.



Extended Data Fig. 7. VFLP - Overview of the Ligand Preparation Procedure. Ligands are initially in a format in line notation (e.g. SMILES, SELFIES, or amino acid sequences). The preparation steps include desalting, neutralization, stereoisomer enumeration, tautomer enumeration, protonation state prediction, 3D coordinate generation, and target format generation. Stereoisomer and tautomer generation can result in multiple chemical species. Highlighted in purple are the steps of a molecule from the beginning to the end of the preparation procedure. Colored in white are the intermediate molecules/tautomers whose subsequent preparation steps are not shown in the diagram.

Property	Available Calculation Methods
Preparation of Ligands into Ready-To-Dock Format	
Desalting	VFLP (internal)
Neutralization	JChem, Open Babel
Stereoisomer Generation	JChem, RDKit
Tautomer Generation	JChem, Open Babel
Protonation State Calculation	JChem, Open Babel
3D Coordinate Conversion	JChem, Open Babel
Target Format Conversion	Open Babel (for all formats but SELFIES) and SELFIES Python Package (only for SELFIES)
Calculation of Additional Molecular Properties	
Molecular weight	JChem, Open Babel, Input Library
logP	JChem, Open Babel, Input Library
HBA count	JChem, Open Babel, Input Library
HBD count	JChem, Open Babel, Input Library
Rotatable bond count	JChem, Open Babel, Input Library
TPSA	JChem, Open Babel, Input Library
Molecular refractivity	JChem, Open Babel, Input Library
Solubility [molar]	JChem, Open Babel, Input Library
Bond count	JChem, Input Library
Aromatic proportion	JChem, Input Library
Ring count	JChem, Input Library
Aromatic ring count	JChem, Input Library
logD	JChem, Input Library
Formal charge	VFLP (internal)
Positive charge count	VFLP (internal)
Negative charge count	VFLP (internal)
Fraction of sp3 carbon atoms (fsp3)	JChem, Input Library
Total atom count	JChem, Open Babel
Heavy atom count	Input Library
Chiral center count	JChem, Input Library
Doublebond stereoisomer count	JChem
Halogen atom count	JChem, Input Library
Sulfur atom count	JChem, Input Library
NO atom count	JChem, Input Library
Electronegative atom count	JChem, Input Library
Enamine compound class	Input Library
Quantitative Estimate of Druglikeness (QED)	RDKit
Murcko scaffold	RDKit

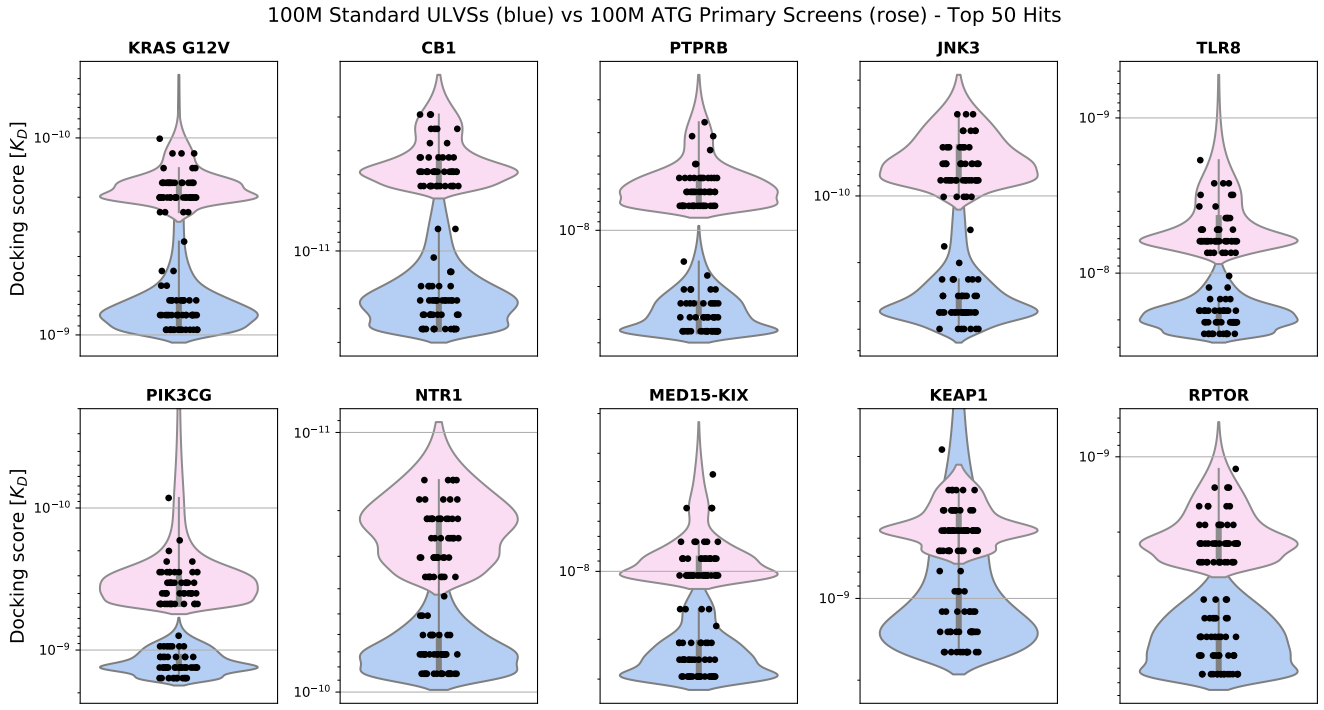
Extended Data Table 3. Supported Processing Steps of VFLP 2.0 VFLP 2.0 can prepare molecules into a ready-to-dock format, and calculate additional properties of each molecule. For many of these calculations, external tools (JChem from ChemAxon, RDKit, and Open Babel) are used, while for a few VFLP 2.0 calculates them internally. Many properties can also be used directly from the input library if available, avoiding the need to calculate them during the runtime.

Property (X)	Tranche		Sensors/Cell=1, Exh=1		Sensors/Cell=10, Exh=1		Sensor/Cell=1, Exh=10	
	Letter	Range	Average Score	Rank	Average Score	Rank	Average Score	Rank
Molecular weight	**A**	X ≤ 200	-5.69868	4	-5.66977	4	-5.75664	4
	B	200 < X ≤ 300	-7.2417	3	-7.20452	3	-7.31959	3
	C	300 < X ≤ 400	-7.79181	1	-7.76165	1	-7.95967	2
	D	400 < X ≤ 450	-7.76098	2	-7.74938	2	-8.0356	1
logP	**A**	X ≤ 0	-7.67313	5	-7.66211	6	-7.86196	6
	B	0 < X ≤ 2	-7.68707	4	-7.6674	5	-7.87787	5
	C	2 < X ≤ 3	-7.7057	3	-7.68094	3	-7.90026	4
	D	3 < X ≤ 4	-7.71324	2	-7.69173	2	-7.90955	2
	E	4 < X ≤ 5	-7.7178	1	-7.70489	1	-7.92332	1
	F	X > 5	-7.67281	6	-7.66768	4	-7.90419	3
HBA count	**A**	X ≤ 2	-7.22028	5	-7.20911	5	-7.32189	5
	B	2 < X ≤ 4	-7.38033	4	-7.36258	4	-7.53528	4
	C	4 < X ≤ 7	-7.61231	3	-7.59707	3	-7.8139	3
	D	7 < X ≤ 10	-7.79858	2	-7.78421	2	-8.01877	2
	E	X > 10	-8.00413	1	-7.9967	1	-8.22418	1
HBD count	**A**	X = 0	-7.6844	3	-7.66601	4	-7.86789	4
	B	0 < X ≤ 2	-7.70107	2	-7.68555	1	-7.89879	2
	C	2 < X ≤ 4	-7.70723	1	-7.68371	2	-7.90943	1
	D	4 < X ≤ 5	-7.68301	4	-7.6664	3	-7.88586	3
	E	X > 5	-7.63874	5	-7.62658	5	-7.83489	5
Rotatable bond count	**A**	X = 0	-7.71737	4	-7.7189	4	-7.85219	4
	B	0 < X ≤ 2	-7.94585	2	-7.90628	2	-8.06441	2
	C	2 < X ≤ 4	-7.99004	1	-7.9602	1	-8.11996	1
	D	4 < X ≤ 7	-7.81481	3	-7.79696	3	-7.97132	3
	E	7 < X ≤ 10	-7.35753	5	-7.36396	5	-7.6019	5
	F	X > 10	-6.72586	6	-6.74264	6	-7.2096	6
TPSA	**A**	X ≤ 40	-7.18856	5	-7.18394	5	-7.2971	5
	B	40 < X ≤ 80	-7.489	4	-7.47825	4	-7.66254	4
	C	80 < X ≤ 110	-7.72677	3	-7.71709	3	-7.94197	3
	D	110 < X ≤ 140	-7.90033	2	-7.88201	2	-8.13229	2
	E	X > 140	-8.10076	1	-8.08286	1	-8.32545	1
Molecular refractivity	**A**	X ≤ 40	-5.44523	4	-5.38075	4	-5.52602	4
	B	40 < X ≤ 80	-7.31169	3	-7.25362	3	-7.40034	3
	C	80 < X ≤ 130	-7.73766	2	-7.71934	2	-7.9346	2
	D	X > 130	-7.79962	1	-7.79242	1	-8.07744	1
logS	**A**	X ≤ -6	-7.89126	1	-7.87113	1	-8.125	1
	B	-6 < X ≤ -5	-7.83936	2	-7.82017	2	-8.04633	2
	C	-5 < X ≤ -4	-7.75784	3	-7.73812	3	-7.95445	3
	D	-4 < X ≤ -3	-7.67255	4	-7.65238	4	-7.86527	4
	E	X > -3	-7.56383	5	-7.55104	5	-7.75079	5
Aromatic proportion	**A**	X = 0	-7.00124	5	-6.97872	5	-7.2258	5
	B	0 < X ≤ 0.25	-7.47794	4	-7.4629	4	-7.71839	4
	C	0.25 < X ≤ 0.5	-7.69386	3	-7.687	3	-7.88349	3
	D	0.25 < X ≤ 0.75	-8.07776	2	-8.07503	2	-8.23953	2
	E	0.75 < X ≤ 1.0	-8.38983	1	-8.39619	1	-8.51087	1
Formal charge	**A**	X ≤ -2	-8.26463	1	-8.28754	1	-8.43463	1
	B	X = -1	-7.95327	2	-7.94973	2	-8.1436	2
	C	X = 0	-7.66391	3	-7.64593	3	-7.85809	3
	D	X = 1	-7.53418	4	-7.50755	4	-7.7401	4
	E	X ≥ 2	-7.26159	5	-7.21597	5	-7.4746	5
Positive charge count	**A**	X = 0	-7.64767	3	-7.62676	3	-7.83845	3
	B	X = 1	-7.73326	2	-7.72519	2	-7.93725	2
	C	X ≥ 2	-7.7854	1	-7.78236	1	-7.9847	1
Negative charge count	**A**	X = 0	-7.47956	3	-7.46405	3	-7.677	3
	B	X = 1	-7.85469	2	-7.86841	2	-8.05519	2
	C	X ≥ 2	-8.23447	1	-8.26297	1	-8.41197	1
Fsp3	**A**	0 ≤ X ≤ 0.2	-8.25291	1	-8.24892	1	-8.39246	1
	B	0.2 < X ≤ 0.4	-7.9144	2	-7.91246	2	-8.09817	2
	C	0.4 < X ≤ 0.6	-7.57134	3	-7.5644	3	-7.78127	3
	D	0.6 < X ≤ 0.8	-7.27781	4	-7.26214	4	-7.50539	4
	E	0.8 < X ≤ 1.0	-7.02762	5	-6.99253	5	-7.26721	5
Chiral center count	**A**	X = 0	-7.68648	2	-7.66472	3	-7.88864	2
	B	X = 1	-7.68156	3	-7.66601	2	-7.87407	3
	C	X ≥ 2	-7.72642	1	-7.70654	1	-7.91929	1
Doublebond stereoisomer count	**A**	X = 0	-7.7409	1	-7.72163	1	-7.92645	1
	B	X ≥ 1	-7.58906	2	-7.56916	2	-7.81019	2
Halogen atom count	**A**	X = 0	-7.6665	2	-7.65526	2	-7.88305	2
	B	X ≥ 1	-7.7349	1	-7.70825	1	-7.90392	1
Sulfur atom count	**A**	X = 0	-7.79078	1	-7.77764	1	-7.98746	1
	B	X ≥ 1	-7.5662	2	-7.53497	2	-7.7627	2
Enamine compound class	**A**	X = S	-7.67793	2	-7.65586	2	-7.89126	2
	B	X = M	-7.72188	1	-7.71391	1	-7.89299	1

Extended Data Fig. 8. Adaptive Target-Guided Prescreening Benchmarks. Results of several prescreens for a test system within the Adaptive Target-Guided Virtual Screening (ATG-VS) approach, where different settings were used. In the first run (left), one representative per tranche was used with docking exhaustiveness 1. In the second run (middle), 10 representatives per tranche are used with exhaustiveness 1. In the third run (right), one representative per tranche was used with exhaustiveness 10. The average docking scores were computed for each tranche of each property (averaging over all other properties/dimensions). For each property, the tranches have been ranked within their property realm based on the average docking scores for the tranches. As can be seen, the patterns and rankings are very similar to each other, indicating that one representative with exhaustiveness 1 can be sufficient for the prescreenings.

Target Protein	Structure ID	Structure Source	Target Site	Target Site Type
$KRAS^{G12V}$	4TQ9	Protein Data Bank (RCSB)	Active site	Enzymatic site
CB1	6KPG	Protein Data Bank (RCSB)	Primary ligand pocket	Orthosteric site
PTPRB	2H04	Protein Data Bank (RCSB)	Active site	Enzymatic site
JNK3	7S1N	Protein Data Bank (RCSB)	Active site	Enzymatic site
TLR8	6ZJZ	Protein Data Bank (RCSB)	Dimerization site	Protein-protein interaction
PIK3CG	2CHZ	Protein Data Bank (RCSB)	Active site	Enzymatic site
NTR1	4GRV	Protein Data Bank (RCSB)	Primary ligand pocket	Orthosteric site
MED15-KIX	2GUT	Protein Data Bank (RCSB)	SREBP interaction site	Protein-protein interaction
KEAP1	5FNQ	Protein Data Bank (RCSB)	NRF2 interaction site	Protein-protein interaction
Raptor	AFQ8N122_F1	AlphaFold Structure Database	TOS motif	Protein-protein interaction

Extended Data Table 4. ATG-VS Benchmark Systems. The target proteins and binding sites used in the benchmark studies comparing ATG-VSs with standard ULVSs.



Extended Data Fig. 9. Comparison of Adaptive Target-Guided (ATG) Primary Virtual Screens and standard ULVS. 100 million compounds were screened with standard ULVS of randomly selected compounds, as well as with ATGs where the 100 million compounds to be screened were selected based on the ATG prescreen. Violin plots of the docking scores of the top 50 virtual screening hits obtained by standard ULVSs (blue) and ATG-VS (rose). Stripplots of the data points are overlayed. The docking scores of the top 50 compounds of the adaptive screens are considerably better for all target proteins than for the standard ULVSs.

Docking program	License model	Special features	DL-based	Hardware support	Reference
AutoDock-GPU	Open Source	GPU-accelerated version of AutoDock4	✗	GPU	(48)
AutoDock4	Open Source	Improved modelling of waters and macrocycles	✗	CPU	(60)
AutoDockZN	Open Source	Special support for zinc-coordinating ligands	✗	CPU	(61)
AutoDock Vina 1.2.0	Open Source	Python support; macromolecule support	✗	CPU	(62)
AutoDock Vina 1.1.2	Open Source	Improved optimization, scoring and multithreading	✗	CPU	(63)
AutoDockFR (ADFR)	Open Source	Improved receptor side-chain flexibility modelling	✗	CPU	(64)
AutoDock Koto	Open Source	Improved sampling	✗	CPU	(65)
ATPdock	Open Source	ATP-specific protein-ligand docking	✗	CPU	(66)
CovDock	Licence Required	Covalent docking	✗	CPU	(67)
DOCK 6.0	Free for academics	RNA-ligand, DNA-ligand, protein-protein docking.	✗	CPU	(68)
FlexX	Licence Required	Covalent docking. Ensemble docking	✗	CPU	(69)
FitDock	Academic Licence	Incorporation of template fitting for docking	✗	CPU	(70)
GalaxyDock3	Licence Required	Incorporation of side chain flexibility	✗	CPU	(71)
Glide (HTVS)	Paid Academic Licencing	Less exhaustive sampling and scoring	✗	CPU	(72)
Glide (SP)	Paid Academic Licencing	More exhaustive sampling and scoring (compared to Glide HTVS) & different levels of accuracy	✗	CPU	(72)
Glide (XP)	Paid Academic Licencing	Most accurate sampling and scoring for Glide & different levels of accuracy	✗	CPU	(72)
Gnina	Open Source	Deep learning scoring of complex	✓	GPU/CPU scoring	(52)
GOLD	Licence Required	Genetic algorithm based docking; Covalent docking	✗	CPU	(73)
GWOVina	Open Source	Grey wolf optimization	✗	CPU	(74)
HDOCK	Academic Licence	Protein–RNA/DNA docking	✗	CPU	(75)
HSYMDOCK	Academic Licence	Protein–protein/rna docking with Cn and Dn symmetry	✗	CPU	(76)
idock	Open Source	AutoDock Vina derivative with improved speed	✗	CPU	(77)
iGEMDOCK	Open Source	-	✗	CPU	(78)
LeDock	Freeware for academic use	Excellent Sampling power (comparable to GOLD)	✗	CPU	(79)
LigandFit	Open Source	Identification of active site regions for docking	✗	CPU	(80)
LightDock	Open Source	Protein-protein docking	✗	CPU	(81)
MDock	Licence Required	Ensemble docking for multiple protein conformations	✗	CPU	(82)
MCDock	Open Source	Use of Monte Carlo simulation technique	✗	CPU	(83)
Molegro	Licence Required	-	✗	CPU	(84)
MpsDockzn	Licence Required	Zinc metalloprotein-specific scoring function	✗	CPU	(85)
PSOVina	Open Source	Improved pose sampling and speed	✗	CPU	(86)
PLANTS	Open Source	Use of ant colony optimization	✗	CPU	(87)
PI-LZerD	Open Source	Protein-protein docking	✗	CPU	(88)
PIPER	Open Source	Use of pairwise interaction potentials for protein-protein docking	✗	CPU	(89)
QuickVina 2	Open Source	Extremely fast	✗	CPU	(90)
QuickVina-W	Open Source	Improved speed for blind docking	✗	CPU	(91)
QVina2-GPU	Open Source	GPU-accelerated version of QVina2	✗	GPU	(49)
QVina2-W-GPU	Open Source	GPU-accelerated version of QVina2-W	✗	GPU	(49)
rDock	Open Source	Support for RNA targets	✗	CPU	(92)
Rosetta Ligand	Licence Required	-	✗	CPU	(93)
RLDOCK	Open Source	RNA-ligand docking	✗	CPU	(94)
SEED	Open Source	Fragment-based docking	✗	CPU	(95)
SMINA	Open Source	User specific scoring functions	✗	CPU	(96)
TANKBind	Open Source	Fast neural network based complex prediction	✓	CPU/GPU	(13)
VinaCarb	Open Source	Improved glycosidic modelling	✗	CPU	(97)
VinaXB16	Open Source	Improved halogen bonding scoring function	✗	CPU	(98)
Vina-GPU	Open Source	GPU-accelerated version of Vina	✗	GPU	(49)
Vina-GPU-2.0	Open Source	GPU-accelerated version of Vina-2.0	✗	GPU	(49)

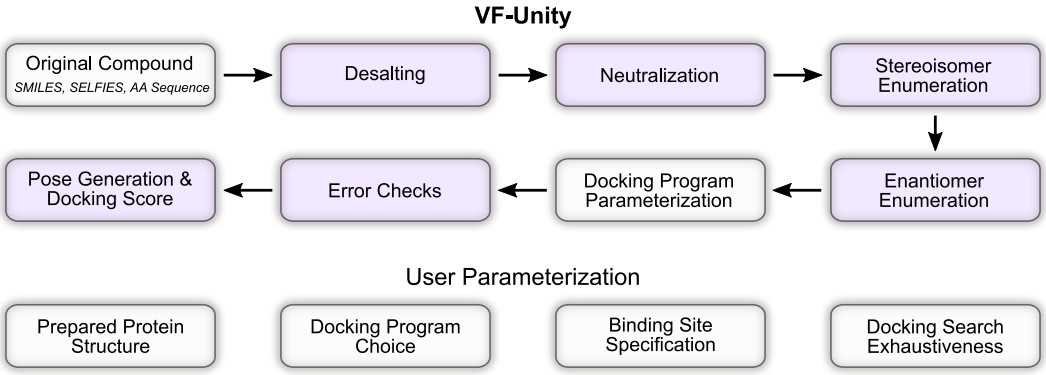
Extended Data Table 5. Supported docking programs. The collection of docking programs supported within VFVS and VF Unity that perform both pose prediction and scoring.

Docking program	License model	Special features	DL-based	Hardware support	Reference
AD4 Scoring	Open Source	Improved over AutoDock 3 forcefield	✗	CPU	(99)
ASP Scoring	Licence required	Atom-atom distance potential	✗	CPU	(73)
Bump Filter Scoring	Free for academics	Consideration of molecule overlap with receptor atoms	✗	CPU	(68)
ChemScore	Licence required	Improved speed compared to GoldScore	✗	CPU	(73)
Contact Scoring	Free for academics	Incorporates the scoring performed with the DISTMAP program	✗	CPU	(68)
Continuous Scoring	Free for academics	No grid calculations	✗	CPU	(68)
ChemPLP	Licence required	Slightly faster than ChemScore and up to four times faster than GoldScore	✗	CPU	(73)
DOCK3.5 Scoring	Free for academics	Incorporation of solvation effects	✗	CPU	(68)
DeepAffinity	Open Source	Use of convolutional neural networks	✓	CPU/GPU	(54)
DeepBindRG	Open Source	Use of recurrent and convolutional neural networks	✓	CPU/GPU	(55)
Footprint Scoring	Free for academics	Calculates intermolecular hydrogen bonds and footprint comparisons, in addition to standard interactions	✗	CPU	(68)
Gnina Scoring	Open Source	Convolutional neural network scoring	✓	CPU/GPU	(52)
GOLD Scoring	Licence required	Use of the chemscore function	✗	CPU	(73)
GoldScore	Licence required	Original scoring function provided with GOLD	✗	CPU	(73)
Grid-Based Scoring	Free for academics	Based on the non-bonded terms of the molecular mechanic force field	✗	CPU	(68)
Hawkins GB/SA Scoring	Free for academics	Use of pairwise GB solvation model	✗	CPU	(100)
NNScore 2.0	Open Source	Use of a fully-connected neural networks	✓	CPU/GPU	(53)
MM/PBSA	Open Source	Calculation of end-state free energies of molecules in solution	✗	CPU	(58)
MM/GBSA	Open source	Improved scoring accuracy	✗	CPU	(58)
MultiGrid FPS Scoring	Free for academics	Pair-wise interaction energies are computed over multiple grids rather than in Cartesian space	✗	CPU	(68)
OpenBPMD	Open Source	Improved scoring accuracy	✗	CPU	(56)
PLP Scoring	Open source	Pairwise additive scoring function	✗	CPU	(87)
PLP95 Scoring	Open source	Origin of PLP	✗	CPU	(87)
PB/SA Scoring	Free for academics	Application of the ZAP electrostatics toolkit from OpenEye	✗	CPU	(68)
Pharmacophore Matching Similarity Scoring	Free for academics	Calculates the level of pharmacophore overlap between a reference molecule and a candidate molecule	✗	CPU	(68)
Rosetta Scoring	Open Source	Improved scoring for side-chain flexibility	✗	CPU	(93)
RF-Score-VS	Open Source	Random Forest-based scoring function	✗	CPU	(101)
SASA Scoring	Free for academics	Calculates the percent exposure of a ligand	✗	CPU	(102)
SEED Scoring	Open Source	Continuum dielectric approximation of electrostatic contribution	✗	CPU	(95)
SMINA Scoring	Open Source	Improved AutoDock Vina scoring	✗	CPU	(96)
Vinardo Scoring	Open Source	Improved Vina scoring	✗	CPU	(103)
Vina Scoring	Open Source	Improved scoring compared to AD4	✗	CPU	(63)
Zou GB/SA Scoring	Free for academics	Fast algorithm, pairwise free energy model, for ligand binding affinity calculations	✗	CPU	(102)

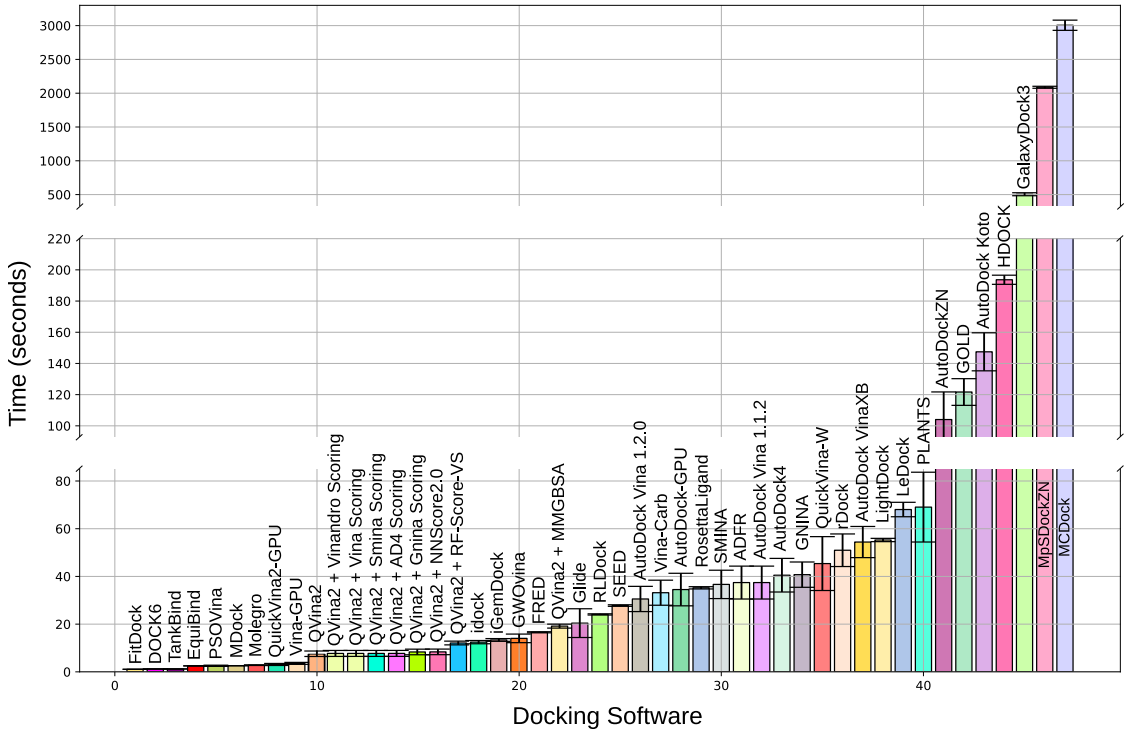
Extended Data Table 6. Supported Protein-Ligand Scoring Functions. The collection of docking programs supported within VFVS and VF Unity that perform both pose prediction and scoring.

Docking program	License model	Special features	DL-based	Hardware support	Reference
DiffDock	Open Source	Blind docking pose prediction	✓	GPU/CPU	(50)
EquiBind	Open Source	Blind docking pose prediction	✓	GPU/CPU	(51)
EnzyDock	Licence Required	Identifies probable binding modes of multiple ligand states in an enzyme active site	✗	CPU	(104)
FRED	Academic Licence	Improved pose sampling	✗	CPU	(105)

Extended Data Table 7. Supported pose prediction methods. The collection of docking programs supported within VFVS and VF Unity that perform both pose prediction and scoring.



Extended Data Fig. 10. Conceptual Overview of VirtualFlow Unity. VF Unity takes as input a user-specified string representation of a molecule and processes the provided molecule by desalting, neutralizing, and generating stereoisomer (with RDKit (44)). Subsequently, based on the docking program choice, all ligands are converted into 3D conformers (with OpenBabel(42)) using a default protonation state of 7.4. Finally, built-in error checks are performed depending on a user's docking program choice, returning the docking pose and scoring function value. Steps in which the parameters of the user are involved are colored in white.



Extended Data Fig. 11. Docking time estimates. Timing estimates for running a single calculation averaged over 25 independent runs for all docking programs. The calculations were run using VFU.