nature machine intelligence

Article

https://doi.org/10.1038/s42256-023-00712-7

ResGen is a pocket-aware 3D molecular generation model based on parallel multiscale modelling

Received: 13 December 2022

Accepted: 1 August 2023

Published online: 7 September 2023

Check for updates

Odin Zhang[1,2], Jintu Zhang[1], Jieyu Jin[1], Xujun Zhang[1], RenLing Hu[1], Chao Shen[1], Hanqun Cao[3], Hongyan Du[1], Yu Kang[1], Yafeng Deng[2], Furui Liu[4] ✉, Guangyong Chen[4] ✉, Chang-Yu Hsieh[1] ✉ & Tingjun Hou[1] ✉

Most molecular generative models based on artificial intelligence for de novo drug design are ligand-centric and do not consider the detailed three-dimensional geometries of protein binding pockets. Pocket-aware three-dimensional molecular generation is challenging due to the need to impose physical equivariance and to evaluate protein–ligand interactions when incrementally growing partially built molecules. Inspired by multiscale modelling in condensed matter and statistical physics, we present a three-dimensional molecular generative model conditioned on protein pockets, termed ResGen, for designing organic molecules inside of a given target. ResGen is built on the principle of parallel multiscale modelling, which can capture higher-level interaction and achieve higher computational efficiency (about eight-times faster than the previous best art). The generation process is formulated as a hierarchical autoregression, that is, a global autoregression for learning protein–ligand interactions and atomic component autoregression for learning each atom's topology and geometry distributions. We demonstrate that ResGen has a higher success rate than existing state-of-the-art approaches in generating novel molecules that can bind to unseen targets more tightly than the original ligands. Moreover, retrospective computational experiments on de novo drug design in real-world scenarios show that ResGen successfully generates drug-like molecules with lower binding energy and higher diversity than state-of-the-art approaches.

Humanity has always been struggling with diseases over the course of civilization's development. The inventions of chemical-compound-based drugs and other therapeutics not only symbolize our victory to rein in diseases but also showcase some of the most remarkable scientific achievements in human history. This victory does not come by easily and we are certainly still far from eradicating all diseases. More advanced techniques are still urgently needed to boost the success rate of drug development. Early drug discovery

[1]Innovation Institute for Artificial Intelligence in Medicine of Zhejiang University, College of Pharmaceutical Sciences, Zhejiang University, Hangzhou, China. [2]Hangzhou Carbonsilicon AI Technology Co., Ltd, Hangzhou, China. [3]Department of Mathematics, Chinese University of Hong Kong, Hong Kong, China. [4]Zhejiang Lab, Zhejiang University, Hangzhou, China. ✉e-mail: liufurui@zhejianglab.com; gychen@zju.edu.cn; kimhsieh@zju.edu.cn; tingjunhou@zju.edu.cn

is essentially a blind process by searching for potential drug candidates in a vast chemical space[1]. Many of them were generally inspired by accidental events or observations in daily life or ancient recipes, such as the discovery of cephalosporins[2] and artemisinin[3]. However, advances in molecular biology and computational chemistry have profoundly transformed the paradigm of drug design, facilitating the transition from blind screening to rational design and enabling efficient navigation in a focused chemical space. Particularly, one can now design a drug candidate by analysing and optimizing drug–target interactions at the molecular level.

Molecular generation is a hot topic in rational drug design, and many works demonstrate that sound molecular generation models can greatly accelerate lead compound discovery[4–6]. However, so far most works[7–10] could be categorized as ligand-based molecular generative (LBMG) models, that is, learning a chemical space, in which potential positive molecules are located, and then interpolating it for molecular generations. Although there is an abundance models designed to explore chemical space, LBMG models still suffer from low generalization issues and are non-trivial for real-world lead compound discovery[11]. The main problem is that LBMG models do not facilitate the information of protein structures for designing new hits and thus cannot efficiently generate target-specific molecules. To put more physical and chemical constraints in chemical space exploration (as denoted in Fig. 1c), molecular generative models are required to not only learn from the topological data in compound libraries[12] (typically provided with two-dimensional graph representations), but also distil the underlying interaction principles from the geometric data[13]. In fact, structure-based molecular generative (SBMG) models, which hold the lofty goal to directly generate pocket-aware ligands with appropriate topology and compatible three-dimensional (3D) geometry (as denoted by Fig. 1c) are receiving increasing attention. Although we may now build reliable machine learning models to estimate protein–ligand binding affinity on the fly[14,15], there has been small efforts invested in developing SBMG models. That is because this non-trivial generative procedure can only be optimally achieved when the models obey SE(3)-equivariance, a topic which has only recently come to the forefront[16–19].

There has been some exploration in the SBMG field recently. Some pioneers[20–22] adopted a geometry-aware framework and considered protein pockets as conditions for LBMG, but they did not explicitly model the interactions between generated molecules and target pockets. LIGAN[23], GraphBP[24] and SBDD[25] were subsequently proposed to directly generate molecules inside pockets. These pioneers have much room for improvement. Meanwhile, another state-of-the-art method, Pocket2Mol[26], took a step further, and was the first to design hits with tighter binding on the basis of geometric neural networks, but was only proven on the CrossDock2020 (ref. 27) dataset. Demonstrated by the results in the following real-world scenario experiments, existing state-of-the-art methods tend to generate random molecules rather than active hits[28]. Apparently—despite many encouraging results being reported—SBMG is still a vastly understudied field with many technical obstacles waiting to be overcome.

In this study we proposed a novel SE(3)-equivariant multiscale generative model, termed ResGen as illustrated in Fig. 1a, to approach this conditional task. Specifically, we introduced a two-level autoregression protocol for molecular generation to better incorporate the geometry of protein pockets into consideration. The global autoregression is to generate atoms one by one in pockets, and the atomic autoregression is to produce the coordinates and topology of the newly added atoms sequentially. Furthermore, to better capture higher-level interactions and reduce computational cost, we introduced a carefully designed parallel multiscale modelling technique to this 3D conditional generation problem, that is, the residue-atom scheme, which is inspired by multiscale modelling—such as coarse-grained molecular dynamics[29] and hybrid quantum mechanics/molecular mechanics[30]—in condensed matter and statistical physics. This characterization is illustrated in Fig. 1b. Furthermore, to maintain the local coordinate features constructed from multiscale modelling as much as possible and fulfil the SE(3)-equivariance demands at the same time, we merged geometric vector neurons[31] and vector perceptrons[18] as the building blocks of ResGen. Overall, extensive experiments provided in this study have proven that ResGen is a powerful tool to generate molecules with closer and even lower binding energy distributions than the experimentally determined actives, and can generate plausible 3D ligand–protein binding poses directly. We believe ResGen is a powerful tool for structure-based drug design.

Results and discussion

The two design criteria for a pocket-aware 3D molecule generative model are: (1) it is able to learn the characteristic topological distributions of ligands in distinct protein pockets, that is, learn a target-dependent distribution of molecular graphs, including the topology associated with new interaction patterns with protein pockets, to achieve tighter binding; and (2) it is able to learn the geometric distributions of ligands inside pockets (that is, the atomic positions and the plausibility of conformations) to directly grow physically sensible poses inside pockets. To assess and compare our method against existing state-of-the-art models with respect to these two design criteria, we proposed the following evaluation scheme. To assess the first design criterion, we evaluated the binding energies and drug-like properties (explained below) on the generated molecules designed for the targets in the test set and real-world therapeutic targets. Furthermore, we computed the molecular similarity between the generated molecules and the actives in the training or test sets, or actives on the therapeutic targets for an integrated evaluation of drug-likeness. To assess the quality of the generated geometry, we performed the "quality of conformations" experiments and analysed the molecular interaction patterns. Particularly, in the conformational quality experiments, we compared the bond-length distributions between the generated samples and training molecules, using the root-mean-square-deviations (r.m.s.d.) between the generated geometries and ground-truth conformations. Furthermore, for the analysis of interaction patterns, we randomly selected several cases and extracted the protein–ligand interaction profiles to illustrate whether ResGen can successfully learn these geometry-dependent interactions.

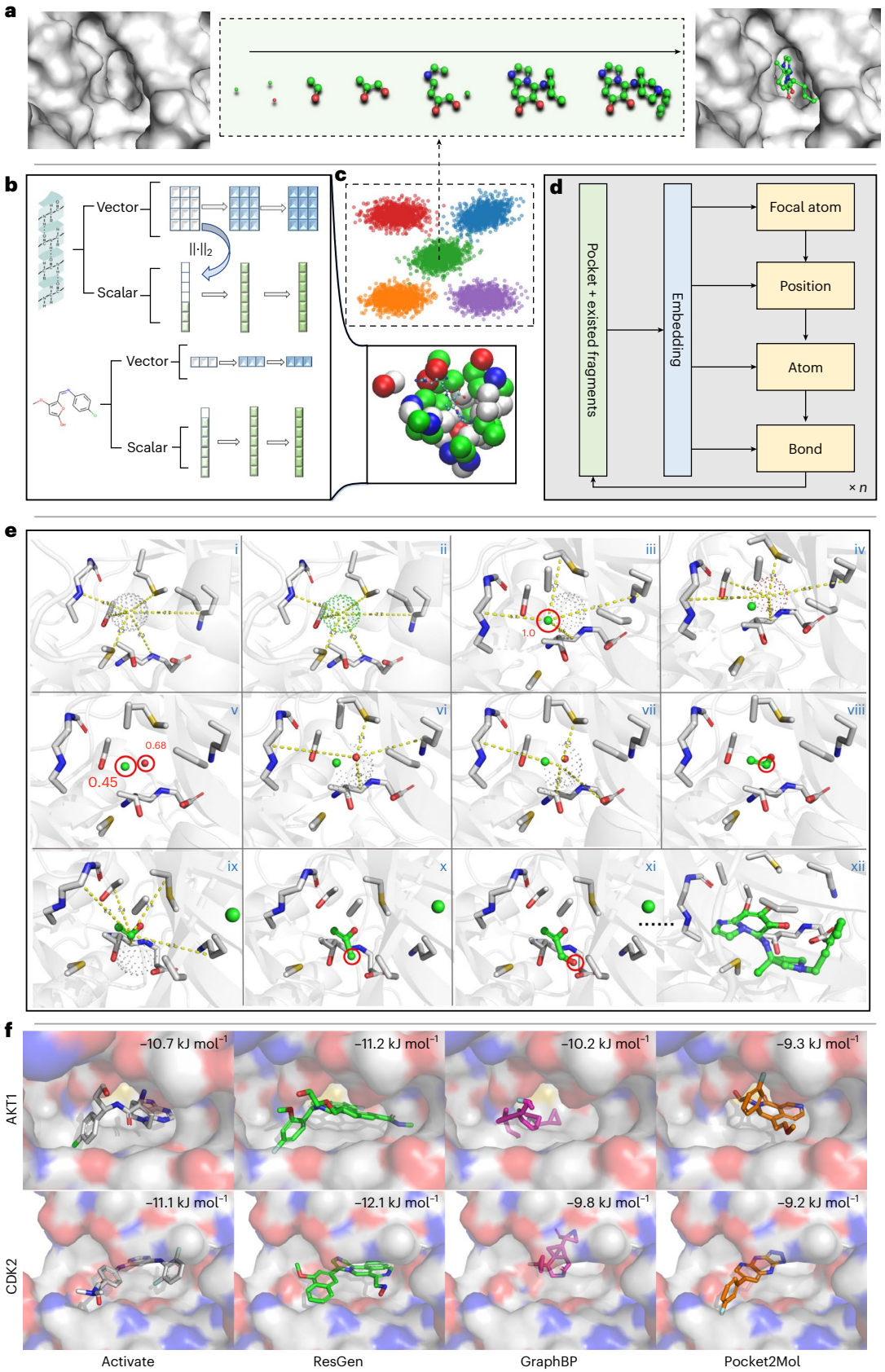
Molecular properties of generated molecules on test set

We first want to evaluate the generalization capability of the models to design potentially active molecules for distinct protein pockets with no close analogues in the training set. To this end, we adopted the following

Fig. 1 | Workflow and architecture of ResGen. a, Illustration of growing a molecule inside the protein pocket for ResGen. **b**, Residue-atom parallel multiscale modelling scheme. The large spheres represent different residues, whereas the small spheres inside the pockets represent atoms. **c**, Clustered distributions of the active molecules of different drug targets in the chemical space. **d**, Simplified framework of ResGen. Inside the block is the autoregression of the atomic component scale, meanwhile, the autoregression of the global scale performs each iteration of the block. **e**, An illustrative example to explain how the atom is generated through the atomic component autoregression and how the molecule is grown by the global autoregression. The grey points cloud, such as that in i, represents the newly generated atom only with positional information. The coloured points cloud, such as the green one in ii, is the newly generated atom complemented with the atomic type. The red circles denote the focal atom at each step, whereas the numbers are the probabilities of becoming the focal atom for each atom. **f**, Molecules generated by different methods on two real-world targets. The numbers in the upper-right corners denote the binding energy estimated by Vina score.

https://doi.org/10.1038/s42256-023-00712-7

data splitting strategy: the protein sequence identity between the test and training sets should be less than 40%. The mean values of various metrics are summarized in Table 1. In general, ResGen outperforms the other state-of-the-art methods, including GraphBP[24] and Pocket2Mol[26], on almost all metrics covering binding energies and drug-likenesses. For the evaluation of binding energy, the goal is to inspect whether a model can generate pocket-aware 3D molecules that are more likely to strongly bind to a given pocket and elicit therapeutic effects. As a



https://doi.org/10.1038/s42256-023-00712-7

Table 1 | The top 1/3/5/10 molecules mean binding energies and drug-likeness properties

	Test set	GraphBP	Pocket2Mol	ResGen
			Top 1	
Vina score (↓)	−7.158	−9.332	−9.247	**−9.622**
Hit Pocket	–	87.07%	92.10%	**93.15%**
QED (↑)	0.484	0.560	**0.562**	0.536
SA (↑)	0.732	0.468	0.707	**0.753**
Lipinski (↑)	4.367	4.821	4.921	**4.958**
log*P*	0.947	1.552	0.8249	1.809
			Top 3	
Vina score (↓)	−7.158	−8.809	−9.042	**−9.453**
QED (↑)	0.484	0.567	**0.574**	0.537
SA (↑)	0.732	0.474	0.712	**0.750**
Lipinski (↑)	4.367	4.787	4.938	**4.954**
log*P*	0.947	1.470	0.786	1.763
			Top 5	
Vina score (↓)	−7.158	−8.515	−8.924	**−9.343**
QED (↑)	0.484	0.563	**0.572**	0.546
SA (↑)	0.732	0.478	0.719	**0.751**
Lipinski (↑)	4.367	4.776	4.928	**4.953**
log*P*	0.947	1.430	0.778	1.679
			Top 10	
Vina score (↓)	−7.158	−8.060	−8.730	**−9.077**
QED (↑)	0.484	0.569	**0.575**	0.553
SA (↑)	0.732	0.485	0.721	**0.748**
Lipinski (↑)	4.367	4.778	4.944	**4.950**
log*P*	0.947	1.366	0.787	1.549

Vina score, the binding energy of ligands to protein pockets; Hit Pocket, the probability that generated molecules will have higher binding affinity than their original ligands for a given target; QED[71], the quantitative estimation of drug-likeness, an integrative score to evaluate compounds' favorability to become a hit; SA[72], the synthetic accessibility evaluation, a score to assess the ease of synthesis of compounds (a higher score corresponds to an easier synthesized molecule); Lipinski[73], Lipinski's rule-of-five, a rule of thumb to evaluate drug-likeness (the higher the better); log*P* (ref. 74), the octanol-water partition coefficient (log*P* values are generally between −0.4 and 5.6, and should ideally be between 1.35 and 1.8 for good oral intestinal absorption). Bold indicates best peformance.

Table 2 | Molecular similarity analysis on the test set

Test set	GraphBP	Pocket2Mol	ResGen
Sim. test.	0.1496±0.1049	0.1774±0.1281	**0.2075±0.1503**
Sim. train.	0.1430±0.0951	0.1915±0.1309	**0.2542±0.1423**
Diversity (↑)	0.1568±0.1008	0.1752±0.1281	**0.2061±0.1527**

crude thermodynamics estimation[32], for every 1 kcal mol^{-1} decrease in energy, the activity of the bound molecule increases by roughly fivefold. It is therefore reasonable to use the docking score (predicted by a reputable docking program called Vina[33]) to estimate binding energy. As shown in Table 1, the top 1/3/5/10 molecules generated by ResGen exhibit the lowest binding energies, implying that ResGen has a better chance to produce more tightly binding molecules than the other two state-of-the-art models. This superior performance can be attributed to its ability to capture higher-level interactions—such as the fragment–residue interaction—between protein pockets and ligands based on multiscale modelling. The Hit Pocket metric validates that ResGen could work on most of the unseen targets (~93%) in the test set. In practice, whether a hit could be advanced to a lead compound depends not only on the strength of its interaction with the protein but also on its drug-likeness and synthesizability. We therefore evaluated various proxy indicators—such as the quantitative estimation of drug-likeness (QED), synthetic accessibility evaluation (SA), Lipinski's rule-of-five, and log*P*, all of which were computed by RDKit[34]—which provide hints on the drug-likeness profile and synthesizability. As listed in Table 1, ResGen scored best in SA and Lipinski, and exhibited competitive QED scores (only ~0.6% lower than the best score) across all sub-tables, indicating that ResGen is more likely to generate drug-like and easily synthesized molecules for these unseen protein pockets.

Molecular similarity analysis on test set

As a generative model, the similarity analysis should compare the generated samples with the training and test sets, as well as internally. As shown in Table 2, the molecules generated by ResGen not only display higher similarity with the molecules in the training and test sets, but they also possess higher diversity. The former observation implies the superior ability of ResGen to learn a complex and multimodel topological and geometric probability conditioned on protein pockets. The higher diversity of the outputs proves that our model not just memorizes some prominent structures in the training set and repeats it again and again during generation. We provide some convincing examples to illustrate ResGen's superior performance (Fig. 1f).

Structure-based drug design for specific targets

Most existing 3D pocket-conditioned molecular generation papers[23–26] focus on demonstrating the capabilities of their models from the viewpoint of algorithmic design but overlooking the assessment of how these models perform in real-world drug design projects. Here we fill this gap and provide an alternative view on how these models perform in a real-world setting. To make a realistic evaluation, we carefully selected several representative and well-studied therapeutic targets, properly prepared their corresponding protein crystal structures, curated experimentally active compounds, and randomly selected other molecules as the random set in the subsequent discussion for performing the binding energy and drug-likeness analyses as we did before. Figure 2a shows the binding affinity distributions of various groups of the molecules, the more left-sided the distribution, the lower the binding energy and the higher the affinity. Molecules generated by ResGen not only score better than the random set and those generated by the other state-of-the-art models, but also achieve better scores than the experimentally confirmed active molecules. By contrast, the distributions of the molecules generated by Pocket2Mol are close to those of the random set despite its competitive performance in past assessment for unseen targets. As for GraphBP, the generated molecules actually bind more loosely to the targets than the random set, which is also implied by the distorted generated examples shown in Fig. 1f. As shown in Table 3, it is clear that the molecules generated by ResGen are the most promising to be further developed into leads, due to their lowest binding energies (~125 times higher affinity than its competitors) and comparable QED scores with the active compounds. It should be noted that the molecules generated by GraphBP score the best on the SA metric as they are much smaller than those produced by the other methods, leading to potentially higher synthesizability. Finally, considering all of the metrics in Table 3, especially the log*P* and molecular weight, we conclude that the molecules generated by Pocket2Mol are most similar to the random set, whereas the molecules generated by ResGen mostly resemble the actives. Figure 2a provides additional corroboration in which the binding affinity distributions of the random and Pocket2Mol molecules almost overlap, and distributions of ResGen fall on the far-left side. In conclusion, the molecules generated by ResGen are closest to the experimentally verified active

https://doi.org/10.1038/s42256-023-00712-7

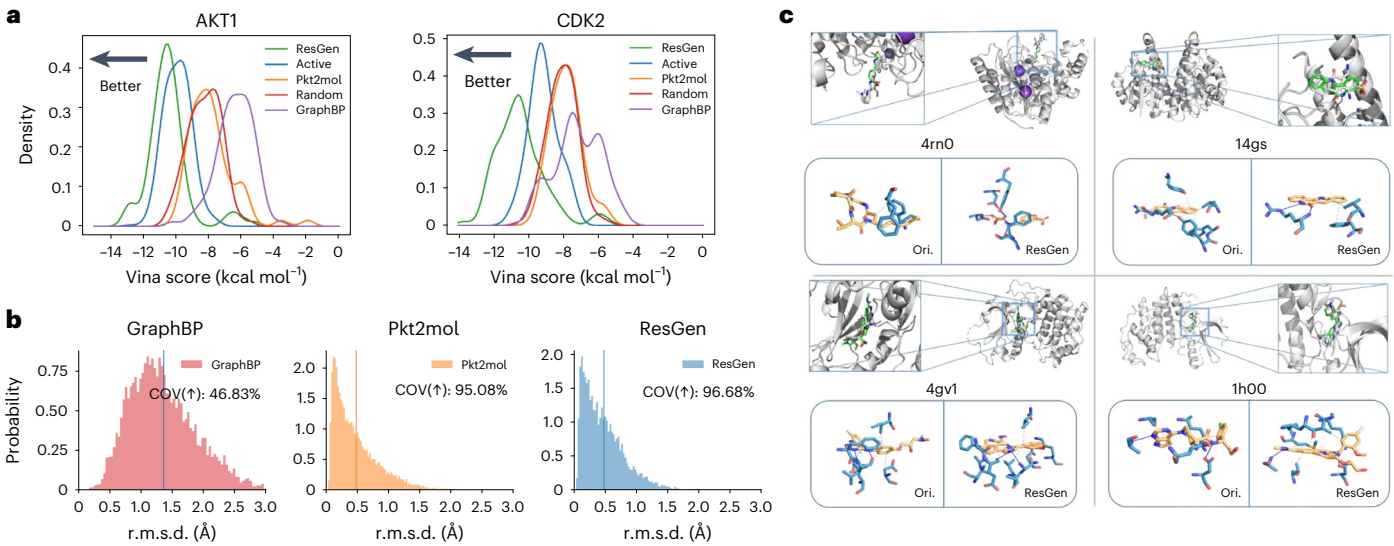


Fig. 2 | Evaluation of generated molecules. a, The binding affinity distributions of three methods, as well as negative (random) and positive (active) controls. **b**, The r.m.s.d. between generated geometry and 20 ETKDG-predicted {Riniker, 2015 #219} conformations. **c**, Protein–ligand interaction analysis. The top quadrants represent show superposition of the original ligand and ResGen's generated ligand, whereas the below quadrants denotes the protein–ligand interaction patterns for them.

compounds with better attributes, such as the highest binding energies (~125 times higher affinity) and high drug-likeness (best scores for Lipinski's rule-of-five). More examples on other therapeutic target could be found in Supplementary Fig. 1.

Molecular similarity analysis on drug targets

In addition to analyses of the binding energies and drug-likenesses, we analysed the similarity between the generated molecules and true actives. As shown in Supplementary Table 3, as expected, ResGen could generate molecules most closely resembling the experimentally positive compounds (that is, ResGen 0.4282 versus Pocket2Mol 0.2808 on CDK2). By contrast, Pocket2Mol seems to generate decoy-like molecules instead of the true active-like molecules. The similarity of the Pocket2Mol's generation has a high match with the random set (for example, Pocket2Mol 0.2808 versus Random 0.2981 on CDK2). A similar conclusion was also made on the molecules generated by GraphBP, which have very low similarity to the known actives (for example, GraphBP 0.1573 versus Random 0.2981 on CDK2), indicating that this method fails to generate drug-like molecules. Although ResGen delivers the most active-like molecules, it also generates the most diverse set of molecules among the three models. This is a desirable trait as drug design is an iterative process that has a greater chance of finding novel hits from a more diverse pool of candidates. In summary, this experiment proves that ResGen could generate drug-like and diverse molecules, whereas Pocket2Mol and GraphBP are more likely to generate random-like molecules.

Quality of generated conformation

Another dimension to assess the learning ability of a 3D generation model is the rationality of the proposed conformations. We conducted the sub-structural and global comparison experiments in this study. For the assessment of the sub-structural geometry, we computed the difference in the bond lengths of the common types between the generated molecules and training molecules. The adopted quantitative metric is Jensen–Shannon divergence[35], which is a symmetric version of the Kullback–Leibler divergence[36]; a lower value denotes that two distributions are more similar. As shown in Table 4, ResGen achieves the best performance in five out of the seven types of bond lengths, outperforming GraphBP by a large margin (approximately tenfold), showing that the bond-length distributions of ResGen are closest to those of the ground-truth conformations. Another evaluation scheme, the global experiment, compares the conformations directly extracted from the 3D generation models with those obtained by the classical conformation generation algorithm, ETKDG[37], followed by the UFF force-filed relaxation[38]. In detail, we generated 20 conformations per molecule using the above method and then computed their r.m.s.d. with the original conformations generated directly from the 3D models. Figure 2b provides the r.m.s.d. distributions, the mean r.m.s.d. scores, and the coverage (COV)[39,40]—a commonly used metric in conformation generation. The lowest mean r.m.s.d. result demonstrates that ResGen could generate more smooth conformations compared with the other two state-of-the-art methods, highlighting its superior ability to capture the complex and multimodel geometric distributions inside protein pockets. Another interesting observation is that the COV metric obtained by ResGen (96.68%) is comparable to those state-of-the-art conformation generation models[41–43] (~90%), implying that ResGen could be directly used for another closely related task—molecular conformation generation.

Interaction pattern analysis

We next analysed whether the pocket-conditioned 3D generative models can learn the detailed microscopic interaction patterns hidden in the data of the 3D conformations for protein–ligand complexes. This analysis was conducted on two randomly selected targets from the test set and two well-studied therapeutic targets, AKT1 (4gv1) and CDK2 (1h00). We used PLIP[44] to characterize the protein–ligand interactions between these protein targets and a few generated ligands (without force-field optimizations) and compared the results with the interaction patterns between these targets and some associated experimental actives. As illustrated in Fig. 2c, it is obvious that the molecules generated by ResGen overlay very well with the original ligands, implying that ResGen learns how a hit localizes in the pocket. The quantitative results of the detailed interaction patterns could be found in Supplementary Table 4. It is worth noting that the molecules generated by ResGen not only can reproduce most of the non-bonded interactions occurred between the experimental actives and protein

pockets (4/4 for 4gv1 and 4/6 for 1h00) but also form other physically sensible interaction patterns that are not present in the limited set of the experimental actives, for example, 4rn0 forms four extras and 4gv1 forms five extras. Specifically, for the 4rn0 target, the example generated by ResGen not only retains the key interaction pattern, such as the hydrophobic interaction with PHE152 and the original hydrogen bonds, but also forms four additional hydrogen-bond-mediated interactions with the pockets compared with the original reference ligand. For the 4gvq target, some key interaction patterns—such as the hydrophobic interactions with ALA177 and LYS179, and the hydrogen bonds with ALA230 and ASP292—are retained while two more hydrophobic and three more hydrogen bonding interactions are formed to improve the binding affinity between the generated molecules and the therapeutic target. This observation strongly suggests that ResGen learns some principles of geometry-depednent interactions and explains why it could achieve surprisingly good results by generalizing these interaction principles on therapeutic targets in the test set. In summary, by analysing the interaction patterns between the generated molecules and protein pockets, we may draw the conclusion that ResGen actually learns how the topological structures should distribute in the protein pockets and the rules of higher-level interactions from the structure data.

Table 3 | Binding energies and drug-likeness properties of the top 5 molecules designed for two real-world targets

	Ori.	Decoys	GraphBP	Pocket2Mol	ResGen
AKT1/top 5					
Vina score (↓)	−11.940	−10.340	−9.059	−9.600	**−12.800**
QED (↑)	0.264	0.506	0.601	**0.745**	0.527
SA (↑)	0.266	0.314	**0.454**	0.264	0.238
Lipinski (↑)	4.200	4.800	**5.000**	**5.000**	**5.000**
log*P*	2.475	1.208	1.417	0.967	2.028
Weight	544.2	470.8	367.1	309.0	476.2
CDK2/top 5					
Vina score (↓)	−11.020	−9.920	−9.720	−9.300	**−12.700**
QED (↑)	0.407	0.563	0.581	**0.667**	0.535
SA (↑)	0.362	0.316	**0.586**	0.336	0.266
Lipinski (↑)	4.600	**5.000**	4.600	**5.000**	**5.000**
log*P*	−0.893	0.821	1.765	0.397	2.768
Weight	448.1	426.0	331.8	249.0	428.6

Graphics processing unit memory consumption

Consumption of computational resources is another issue we should judge a model in real-world scenarios. One way to optimize the consumption is to exert inductive biases on the model using the domain knowledge we have. For instance, if a scientist aims to design a hit molecule toward a specific target that only has limited experimentally verified active molecules. One approach to enhance the performance of a universal molecule generation model is to fine-tune the model trained on other structurally similar targets with a rich set of active molecules. It is generally believed that the more data you load at a time, the faster you can fine-tune the model. As a consequence, the more samples you can load into a batch, the easier it can be applied to real scenarios. Here we used an affordable graphics processing unit (GPU), NVIDIA-3090 with 24 GB GPU memory to illustrate the superior memory efficiency of ResGen. Let us first look at how competitive models perform in this regard. GraphBP could train with eight samples in a batch, Pocket2Mol could train with six samples in a batch, and ResGen could train with 42 samples in a batch. The key is the residue-atom parallel multi-scale modelling we introduced, where the final nodes for the message passing scheme are at the residue level. Let's take a nuclear receptor (PDB entry: 5jjm) as an example. The pocket is a box of radius 10 Å centred on its original ligand. If we trained it with the other two methods, GraphBP and Pocket2Mol, the protein graph could contain 247 nodes, but if we trained it with ResGen, the protein graph would only contain 16 nodes, thus reducing the learning space by a factor of 15. Although the computational cost is much lower than that of synthesizing molecules, a rapid algorithm with minimal computational resource requirements can expedite the feedback loop for models, thereby streamlining the drug development process.

Conclusion

Generating 3D molecules in protein pockets is of great interest but still remains challenging. Inspired by the technique of multiscale modelling in condensed matter and statistical physics, we proposed ResGen, a novel SE(3)-equivariant conditional generative model, to generate 3D molecules for any given protein pocket. Based on our comprehensive analysis from both perspectives of benchmarking models and fulfilling the needs of real-world drug design projects, we conclude that ResGen could generate molecules that bind more tightly to unseen and therapeutically relevant protein pockets with potentially better drug-likeness. Moreover, the 3D molecular structures generated by ResGen are physically sensible and comparable with the conformations obtained from conformation generation models (COV ≈ 96%, which is comparable with the state-of-the-art performance). By analysing the protein–ligand interaction patterns for several cases, we confirmed that ResGen can generate promising hits that not only retain favourable higher-level interactions that reference ligands have but also explore more possibilities based on pocket structures. Moreover, ResGen is a computationally efficient model as it reduces the GPU consumption by almost an order of magnitude in comparison to other state-of-the-art methods, owing to its adoption of parallel multiscale modelling technique. In short, ResGen convincingly outperforms recent baselines in generating 3D molecules conditioned on protein pockets with improved generalization capability for novel targets, learning essential molecular interactions and appropriately porting these interaction patterns across distinct protein pockets. This feat is achieved while consuming less computational resources than existing methods.

Methods

Unlike past studies, which treated a protein pocket as a scalar feature for conditional generation[20–22], ResGen is explicitly designed for growing 3D molecules on the basis of the topology and geometry of protein pockets. Inspired by past works in graph generation and 3D molecular generation[24,25,43,45], we formulate the 3D molecule generation conditioned on protein pockets into a two-scale autoregressive problem, that is, the global scale and the atom-component scale. To be specific, the global scale autoregression refers to:

$$\begin{aligned} \mathcal{C}^{(0)} &= \mathcal{P} \\ (a_t, b_t, r_t) &= \mathrm{GVP}\left(\mathcal{C}^{(t-1)}\right) \\ \mathcal{C}^{(t)} &= \mathcal{C}^{(t-1)} \cup \{a_t, b_t, r_t\} \end{aligned} \tag{1}$$

where GVP is geometric vector perceptron, $\mathcal{C}^{(t)}$ represents the context information at tth step, consisting of geometric and topologic information of protein pockets, and existing parts of ligand atoms, whereas $\mathcal{C}^{(0)}$ only includes information on protein pockets; $\{f_t, \Delta r_t, a_t, b_t, r_t\}$ is a set of the required components to generate an atom, where f_t is the focal atom, Δr_t is the relative coordinate vector difference to the focal atom, a_t is the atom type and b_t is the bonding relationship. The global scale autoregression explicitly models how molecular generation is conditioned on a given protein pocket. At the same time, the

Table 4 | Jensen–Shannon divergence between the bond length distributions of different methods and that of the training set

Method	Jensen–Shannon divergence (×10^{-3}) (↓)						
	C–C	C–N	C–O	N–O	S–O	C–Cl	C–F
GraphBP	10.02	8.093	10.57	26.20	8.114	22.14	15.89
Pkt2Mol	1.066	0.9987	**1.523**	1.128	0.9358	**0.5932**	0.5274
ResGen	**1.059**	**0.9528**	1.708	**1.090**	**0.3871**	1.067	**0.4624**

atomic-component scale autoregression empowers ResGen to generate 3D molecules directly. In particular, the global autoregression means that each atom we generate is based on the generated partial molecule and protein pocket in the previous steps. The atom-component scale autoregression refers to the decomposition of the next generated atom according to its components for each generation. As described above, we define the components of each atom as $\{f_t, \Delta r_t, a_t, b_t\}$. Each time an atom is generated it is essentially sampled from $p(f_t, \Delta r_t, a_t, b_t, |, \mathcal{C}^{(t-1)})$, and those four variables belong to different modes. To sample from $p(f_t, \Delta r_t, a_t, b_t, |, \mathcal{C}^{(t-1)})$ effectively, we decompose this joint distribution as a product of multiple conditional probabilities:

$$\begin{aligned} p(f_t, \Delta r_t, a_t, b_t, |, \mathcal{C}^{(t-1)}) &= p(\Delta r_t, a_t, b_t, |, f_t, \mathcal{C}^{(t-1)}) \times p(f_t, |, \mathcal{C}^{(t-1)}) \\ &= p(a_t, b_t, |, \Delta r_t, f_t, \mathcal{C}^{(t-1)}) \times p(\Delta r_t, |, f_t, \mathcal{C}^{(t-1)}) \times p(f_t, |, \mathcal{C}^{(t-1)}) \end{aligned} \quad (2)$$

ResGen is constructed with four blocks following the above decomposition. The illustrative workflow is depicted in Fig. 1d.

Multiscale modelling of proteins and ligands

As a famous saying goes, "more is different"[46,47]; this saying describes a well-established observation whereby new phenomena may emerge when the physical scale of interest changes. As a result, multiscale modelling was developed to reduce computational costs while still delivering reasonable accuracy and even finding new physics hidden in the higher dimensions. The famous examples in molecular modelling are coarse-grained molecular dynamics[29] and hybrid quantum mechanics/molecular mechanics[30]. Furthermore, for newly developed data-driven methods, it is also discovered that multiscale-based representation does not compromise the model's accuracy and even may exceed some all-atom-based modelling in binding affinity prediction[15]. Inspired by works in protein modelling and computational protein design[31,48], we build the protein models based on residue-level features and local coordinates system at each residue to capture higher-level and long-range interactions. In particular, each amino acid is abstracted to the pattern of backbone plus residues, whose vector features are structural and scalar features are structural and topological. The vector features consist of the mutual and global positional relationships of amino acids among proteins, that is $\{C_{\alpha i}–C_{\alpha i+1}; C_{\alpha i}–C_{\alpha i-1}; C_{\alpha i}–\mathrm{Res}_i; C_{\alpha i}\}$, where $C_{\alpha i}$ and Res_i are the positions of α-carbon atom and the centre of gravity of the residue in the ith amino acid. The scalar features are dihedral angles computed from $\{C_{i-1}, C_{\alpha i}, N_i, C_{i+1}, N_{i+1}\}$ and the one-hot encoding of amino acid residue types (N_i is the position of the nitrogen atom in the ith amino acid).

For the modelling of ligands, the vector features are 3D Cartesian coordinates, and the scalar features are the atomic types, the degree of the atom as a node and the number of valence bonds for the atom. As the representations of residues and ligands belong to two distinct scales, we use the geometric vector linear layer to map them into a unified representation space.

Focal atom prediction

As discussed in the previous section, the first subtask for each atom generation is to predict the focal atom, whereby a new atom is to be attached. A discrete distribution consisting of Dirac distributions is utilized to model this conditional probability, that is, $p_f = \sum_{i=1}^{n} w_i \delta(x - x_i)$, where n is the number of the already generated atoms for the partially grown molecule; however, at the initial stage when there is no ligand, n refers to the number of the atoms of the protein pocket. The neural architecture for this prediction goes as follows:

$$(s_f, V_f) = \mathrm{GVP}(s_h, V_h) \quad (3)$$

$$p_f = \sigma(\mathrm{MLP}_{\mathrm{foc}_2}(\mathrm{MLP}_{\mathrm{foc}_1}(s_f) +, |, |V_f|\,|_2)) \quad (4)$$

where (s_h, V_h) are the scalar and vector features of the encoded latent space; MLP is the multilayer perceptron[49]; foc1 and foc2 denote two different MLP networks; and σ is the sigmoid function, which outputs a probability value between [0,1].

Coordinate distribution modelling

In comparison with generating coordinates directly with the regression scheme, Méndez-Lucio[50] and Shen[15] used the mixture density network[51] to fit the distribution of positions between atoms with a multivariate Gaussian mixture distribution and achieved impressive results. This model transforms a regression problem into a generative problem, which enhances the robustness of the model to some extent[52,53]; we therefore also adopted a mixture density network as the underlying network for the coordinate prediction, that is, $p(\Delta \mathbf{r}_i) = \sum_{i=1}^{K} \pi_i^{(k)} \mathcal{N}\left(\boldsymbol{\mu}_i^{(k)}, \boldsymbol{\Sigma}_i^{(k)}\right)$, where π, μ, Σ are the combination coefficients, mean and variance of the multivariate Gaussian mixture distribution, respectively, k defines the number of mixed Gaussians and $\Delta \mathbf{r}_i$ represents the coordinate vector difference between the $(i-1)$th and ith atoms.

$$(s_p, V_p) = \mathrm{GVP}_p(s_h, V_h) \quad (5)$$

$$(\mu_s, \mu_v) = \mathrm{GVP}_\mu(s_h, V_h) \quad (6)$$

$$(\Sigma_s, \Sigma_v) = \mathrm{GVP}_\Sigma(s_h, V_h) \quad (7)$$

$$(\Sigma, \mu) = (\Sigma_v, \mu_v) \quad (8)$$

$$\pi = \mathrm{MLP}_{\pi_2}(\mathrm{MLP}_{\pi_1}(s_p) + \|V_p\|_2) \quad (9)$$

where (μ_s, μ_v), (Σ_s, Σ_v) are the scalar and vector features of mean and variance; and (Σ, μ, π) is a final set determining a Gaussian mixture distribution as defined above; thus, the distribution of the newly generated atom position is $p(\Delta \mathbf{r}_i + r_{f_i}) = \sum_{k=1}^{K} \pi_i^{(k)} \mathcal{N}(\boldsymbol{\mu}_i^{(k)} + r_{f_i}, \boldsymbol{\Sigma}_i^{(k)})$, where r_{f_i} is the coordinate of the focal atom in ith step.

Atomic type and bonding relationship prediction

Once we have the location of the new atom, we can predict its type from its neighbouring atoms' structural and geometric information.

$$(s_{m(i)}, V_{m(i)}) = \sum_{k \in \mathrm{kNN}(i)} \mathrm{Message}\left(s_{h(k)}, V_{h(k)}, e_{ki}, U_{ki}\right) \quad (10)$$

$$p_{a_i} = \mathrm{MLP}_{at_2}\left(\mathrm{MLP}_{a_2}\left(\mathrm{s}_{m(i)}\right) + , | , |\mathrm{V}_{m(i)}|\,|_2\right) \quad (11)$$

where 'Message' is the Message passing block; $s_{h(k)}$ and $Vh_{(k)}$ are the scalar and vector features of the latent space of k-nearest neighbour atoms, respectively, whereas e_{ki} and U_{ki} are scalar and vector features of corresponding bonds, respectively; a_i is also a discrete distribution, that is, $p_{a_i} = \sum_{i=1}^{A} w_i \delta(x - x_i)$, where A is the total number of atomic types.

Numerous molecular generation works[7,23–25,54] do not predict bonding relationships but instead predict the possible bonding patterns by a set of chemical rules, such as bond lengths between atoms; however, this approach is not reliable and may cause additional errors as it is not a trivial task[54,55]. We therefore add an extra module to predict the bonding relationship between existing atoms and the next generated atom, thus helping the entire model to capture more meaningful chemical features and reducing the error introduced from the extra bond type determination.

$$\mathrm{e}'_{ij}, \mathrm{U}'_{ij} = \mathrm{GVP}_{\mathrm{bp}_1}\left([s_i \| s_j \| e_{ij}], [\mathrm{V}_i \| \mathrm{V}_j \| \mathrm{U}_{ij}]\right) \quad (12)$$

$$\mathrm{e}''_{ij}, \mathrm{U}''_{ij} = \mathrm{Attention}\left(\mathrm{e}'_{ij}, \mathrm{U}'_{ij}\right) + \left(\mathrm{e}'_{ij}, \mathrm{U}'_{ij}\right) \quad (13)$$

$$e_{ij}^{\mathrm{type}} = \mathrm{MLP}_{bp_2}\left(\mathrm{MLP}_{bp_1}\left(\mathrm{e}''_{ij}\right) + , | , \left|\mathrm{U}''_{ij}\right|\,|_2\right) \quad (14)$$

where the (e_{ij}, U_{ij}) are the hidden features for the prediction of bonding relationships, (s_i, V_i) are the scalar and vector features of the atoms, || is the concatenation operation, Attention is the trigonometry self-attention module[26,56] and e_{ij}^{type} is the final edge type.

Trigonometry self-attention

The attention mechanism has been shown to be effective in a variety of scenarios[26,57–60], and one of the reasons for the success of AlphaFold2 is the introduction of a tri-edge self-attention mechanism[56]. In our case, we utilized this mechanism to predict the bonding relationship between the newly generated atom i and the already generated part of the ligands. The rationale for bringing trigonometry into the self-attention mechanism is explained with the octet rule[61], that is, the tendency of common atoms in organic chemistry to reach eight electrons in the valence shell. As a result, the max number of the electron pairs a carbon can form with other atoms is four in general. The intuition of this module goes: if the new atom i forms a triple bond with atom j and a single bond with another atom, it will have no access to form a bond with atom k. Trigonometry would help to capture these chemical rules automatically. The detail of this module goes as follows:

$$e_{ij}^{\mathrm{query}}, \mathrm{U}_{ij}^{\mathrm{query}} = \mathrm{GVL}_{\mathrm{bt}_1}\left(\left[s_i \| s_j \| e_{ij}\right], \left[\mathrm{V}_i \| \mathrm{V}_j \| \mathrm{U}_{ij}\right]\right) \quad (15)$$

$$e_{jk}^{\mathrm{key}}, \mathrm{U}_{jk}^{\mathrm{key}} = \mathrm{GVL}_{\mathrm{bt}_2}\left(\left[s_j \| s_k \| e_{jk}\right], \left[\mathrm{V}_j \| \mathrm{V}_k \| \mathrm{U}_{jk}\right]\right) \quad (16)$$

$$e_{jk}^{\mathrm{value}}, \mathrm{U}_{jk}^{\mathrm{value}} = \mathrm{GVL}_{\mathrm{bt}_3}\left(\left[s_j \| s_k \| e_{jk}\right], \left[\mathrm{V}_j \| \mathrm{V}_k \| \mathrm{U}_{jk}\right]\right) \quad (17)$$

$$e_{jk}^{\mathrm{bias}}, \mathrm{U}_{jk}^{\mathrm{bias}} = \mathrm{GVL}_{\mathrm{bt}_4}\left(\left[s_j \| s_k \| e_{jk}\right], \left[\mathrm{V}_j \| \mathrm{V}_k \| \mathrm{U}_{jk}\right]\right) \quad (18)$$

where $e_{ij}^{\mathrm{query}}, \mathrm{U}_{ij}^{\mathrm{query}}, e_{jk}^{\mathrm{key}}, \mathrm{U}_{jk}^{\mathrm{key}}, e_{jk}^{\mathrm{value}}, \mathrm{U}_{jk}^{\mathrm{value}}$ and $e_{jk}^{\mathrm{bias}}, \mathrm{U}_{jk}^{\mathrm{bias}}$ are the corresponding scalar and vector features of query, key, value and bias, respectively. The scalar and vector attention scores would then be computed:

$$\alpha_{ijk} = e_{ij}^{\mathrm{query}} \times e_{ik}^{\mathrm{key}} \quad (19)$$

$$\mathrm{A}_{ijk} = \mathrm{tr}\left(\mathrm{U}_{ij}^{\mathrm{query}} \times \mathrm{U}_{jk}^{\mathrm{key}^{\mathrm{T}}}\right) \quad (20)$$

$$\mathrm{b}_{jk} = \mathrm{e}_{jk}^{\mathrm{bias}} \times e_{jk}^{\mathrm{bias}} \quad (21)$$

$$\mathrm{B}_{jk} = \mathrm{tr}\left(\mathrm{U}_{jk}^{\mathrm{bias}} \times \mathrm{U}_{jk}^{\mathrm{bias}^{\mathrm{T}}}\right) \quad (22)$$

$$\alpha'_{ijk} = \mathrm{softmax}\left(\alpha_{ijk} + \mathrm{b}_{jk}\right) \quad (23)$$

$$\mathrm{A}'_{ijk} = \mathrm{softmax}\left(\mathrm{A}_{ijk} + \mathrm{B}_{jk}\right) \quad (24)$$

$$\mathrm{att}_{ij} = \sum_{k \in V_{\mathrm{lig}}} \alpha'_{ijk} \times e_{jk}^{\mathrm{value}} \quad (25)$$

$$\mathrm{Att}_{ij} = \sum_{k \in V_{\mathrm{lig}}} \mathrm{A}'_{ijk} \times \mathrm{U}_{jk}^{\mathrm{value}} \quad (26)$$

where $\mathrm{V}_{\mathrm{lig}}$ is the set of already generated atoms, att_{ij} is the scalar attention score of edge ij, and Att_{ij} is the vector attention score of that edge. The tr(.) operation is also called the Frobenius inner product, that is, $\langle A, B \rangle_F := \sum_{i,j} a_{ij} b_{ij}$.

Training protocol

In the classical pre-training task, a self-supervised learning strategy exists that randomly gives a node mask and predicts the linkage and properties of the node from its neighbours[62–64]. We therefore also adopted this protocol for the generation of each atom. We randomly mask atoms in the ligand and recover their positions, atomic types and bonding relationships for training ResGen. We also randomly sample the blank atoms around the real existing mask atoms to introduce negative samples, enhancing the learning ability of ResGen. Moreover, we also employed the teacher-forcing strategy[65,66] to help the autoregression of the global and atom-component levels converge, that is, replacing the prediction with the ground truth in the intermediate conditions variables for the next generation step.

In summary, the overall loss function is the summation of the above four terms:

$$\mathscr{L}_{\mathrm{fro}} = -\frac{1}{n}\sum_{i=1}^{n} f_i \times \log \tilde{f}_i + (1 - f_i) \times \log\left(1 - \tilde{f}_i\right) \quad (27)$$

$$\mathscr{L}_{\mathrm{pos}} = -\log P\left(\Delta \mathbf{r}_i + r_{f_i}\right) = -\log \sum_{k=1}^{K} \pi_i^{(k)} \mathcal{N}\left(\boldsymbol{\mu}_i^{(k)} + r_{f_i}, \boldsymbol{\Sigma}_i^{(k)}\right) \quad (28)$$

$$\mathscr{L}_{\mathrm{ele}} = -\sum_{i=1}^{n} \mathrm{a}_i \log \tilde{a}_i \quad (29)$$

$$\mathscr{L}_{\mathrm{bond}} = -\sum_{i=1}^{n} \mathrm{b}_i \log \tilde{b}_i \quad (30)$$

The frontiers prediction is essentially a binary classification problem for already generated atoms, with binary cross-entropy as the loss function. The prediction of atoms and bonding could be attributed to a multiclassification problem with cross-entropy loss function utilized. The prediction of positions is designed to fit a multivariate Gaussian mixture distribution with negative log-likelihood loss used.

Generation protocol

The overall generation process can be split into the global and atom-component levels, as we stated at the beginning of the Methods. Figure 1e illustrates the whole process. First, at the initial stage (Fig. 1ei), we predict the focal atom in the protein pocket and then determine the position of the first atom on the basis of the context $C^{(0)}$,

that is, the pure protein environment. Second, the atomic type is determined based on the initial context and generated atomic components $\{\mathscr{C}^{(0)}, f_{t_0}, r_0\}$, where f_{t_0} and r_0 are the focal atom and position of generated atom, respectively (Fig. 1eii). After the first-generation process is complete, the context information is updated to $\mathscr{C}^{(1)} = \{\mathscr{C}^{(0)}, f_{t_0}, r_0, a_0, b_0\}$. The next atom generation is then followed in the same order, selecting the focal atom (as shown in Fig. 1eiii, there is only one choice as there is only one generated atom), determining the position of the next atom and predicting its atomic type and bonding with the context information updated as $\mathscr{C}^{(2)} = \{\mathscr{C}^{(1)}, f_{t_1}, r_1, a_1, b_1\}$ (Fig. 1eiv). The next step is the generation of the third atom. There is a slight difference between the generation processes for the first and second atoms: the model is required to decide which atom is the focal atom. Thus, as depicted in Fig. 1ev, we score the already generated atoms and sample from this distribution for selecting f_{t_2}. The other atomic components then go as follows, position (Fig. 1evi), atomic type (Fig. 1evii) and bonding relationship (Fig. 1eviii), with the context information updated simultaneously, $\{\mathscr{C}^{(2)}, f_{t_1}\}$, $\{\mathscr{C}^{(2)}, f_{t_1}, r_1\}$, $\{\mathscr{C}^{(2)}, f_{t_1}, r_1, a_1\}$, $C^{(3)}$. This process is repeated (Fig. 1eix–xiii) until the molecule grows to a predetermined maximum molecular weight or the probability of selecting the focal atom is below a given value. Finally, a complete molecule is created by this global and atom-component autoregressive scheme.

Dataset

The training dataset we used here is CrossDock2020[27], a massive collection of small molecules docked into receptors. The initial set of the collection contains over 22 million protein–ligand pairs and is clustered based on 40% sequence similarity in its original paper. Following Pocket2mol[26] we ignored any poses with r.m.s.d. greater than 2 Å and split the train and test data to ensure the sequence similarity between the training and test sets is less than 40% for a fair comparison on the model side. As a result, the remaining train set contains about 100,000 protein–ligand pairs, whereas the test set contains 100 pockets.

In real-world drug discovery dataset, the protein structures were downloaded from PDB[13], the experimentally active molecules were extracted from BindingDB[12], and the randomly selected molecules were taken from AICures[67,68].

Other set-ups

Molecular similarity: all molecular similarity is evaluated by the Tanimoto coefficient[67] with the RDKit's default topological fingerprint[34]. The Tanimoto coefficient is defined as follows:

$$\text{Tanimoto}\,(G_1, G_2) = \frac{|G_1 \cap G_2|}{|G_1 \cup G_2|} = \frac{|G_1 \cap G_2|}{|G_1| + |G_2| - |G_1 \cap G_2|} \tag{31}$$

where G is the molecular fingerprint.

The r.m.s.d. is:

$$\text{r.m.s.d.}(\hat{R}, R) = \min_{\theta}\left(\frac{1}{n}\sum_{i=1}^{n}\|\Phi(\hat{R}_i) - R_i\|^2\right)^{\frac{1}{2}} \tag{32}$$

where n is the number of heavy atoms and Φ is an alignment function that aligns two conformations by rotation and translation.

Coverage is a well-acknowledged metric in molecular conformation generation[69,70], and is computed as follows:

$$\text{COV}\,(S_g, S_r) = \frac{|R \in S_r, |, \text{r.m.s.d.}\,(R, \hat{R}) < \delta, \hat{R} \in S_g|}{|S_r|} \tag{33}$$

where S_g, S_r are the generated and reference molecular conformation ensembles, respectively; δ is a given r.m.s.d. threshold, and we choose $\delta = 1.25$ Å here. In general, a high COV metric represents higher diversity performance.

Data availability

The train and test data of this study is available at Zenodo (https://doi.org/10.5281/zenodo.7759114).

Code availability

The source code of this study is freely available at GitHub (https://github.com/HaotianZhangAI4Science/ResGen) to allow replication of the results.

References

1. Mandal, S. & Mandal, S. K. Rational drug design. *Eur. J. Pharmacol.* **625**, 90–100 (2009).
2. Bo, G. Giuseppe Brotzu and the discovery of cephalosporins. *Clin. Microbiol. Infection* **6**, 6–9 (2000).
3. Kong, L. Y. & Tan, R. X. Artemisinin, a miracle of traditional Chinese medicine. *Nat. Prod. Rep.* **32**, 1617–1621 (2015).
4. Zhavoronkov, A. et al. Deep learning enables rapid identification of potent DDR1 kinase inhibitors. *Nat. Biotechnol.* **37**, 1038–1040 (2019).
5. Stokes, J. M. et al. A deep learning approach to antibiotic discovery. *Cell* **180**, 688–702.e613 (2020).
6. Godinez, W. J. et al. Design of potent antimalarials with generative chemistry. *Nat. Mach. Intell.* **4**, 180–186 (2022).
7. Zang, C. & Wang, F. MoFlow: An invertible flow model for generating molecular graphs. In *Proc. 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* 617–626 (ACM, 2020).
8. Jin, W., Barzilay, R. & Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. In *Proc. 35th International Conference on Machine Learning* 2323–2332 (PMLR, 2018).
9. Shi, C. et al. GraphAF: a flow-based autoregressive model for molecular graph generation. Preprint at https://arxiv.org/abs/2001.09382 (2020).
10. Gao, K., Nguyen, D. D., Tu, M. & Wei, G.-W. Generative network complex for the automated generation of drug-like molecules. *J. Chem. Inf. Model.* **60**, 5682–5698 (2020).
11. Xie, W., Wang, F., Li, Y., Lai, L. & Pei, J. Advances and challenges in de novo drug design using three-dimensional deep generative models. *J. Chem. Inf. Model.* **62**, 2269–2279 (2022).
12. Liu, T., Lin, Y., Wen, X., Jorissen, R. N. & Gilson, M. K. BindingDB: a web-accessible database of experimentally determined protein–ligand binding affinities. *Nucleic Acids Res.* **35**, D198–D201 (2007).
13. Sussman, J. L. et al. Protein Data Bank (PDB): database of three-dimensional structural information of biological macromolecules. *Acta Crystallogr. D* **54**, 1078–1084 (1998).
14. Jiang, D. et al. InteractionGraphNet: a novel and efficient deep graph representation learning framework for accurate protein–ligand interaction predictions. *J. Med. Chem.* **64**, 18209–18232 (2021).
15. Shen, C. et al. Boosting protein–ligand binding pose prediction and virtual screening based on residue–atom distance likelihood potential and graph transformer. *J. Med. Chem.* **65**, 10691–10706 (2022).
16. Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A. & Müller, K.-R. SchNet–a deep learning architecture for molecules and materials. *J. Chem. Phys.* **148**, 241722 (2018).
17. Cohen, T. S., Geiger, M., Köhler, J. & Welling, M. Spherical CNNs. Preprint at https://arxiv.org/abs/1801.10130 (2018).
18. Deng, C. et al. Vector neurons: a general framework for SO(3)-equivariant networks. In *Proc. IEEE/CVF International Conference on Computer Vision* 12200–12209 (IEEE, 2021).
19. Thomas, N. et al. Tensor field networks: rotation-and translation-equivariant neural networks for 3D point clouds. Preprint at https://arxiv.org/abs/1802.08219 (2018).

20. Grechishnikova, D. Transformer neural network for protein-specific de novo drug generation as a machine translation problem. *Sci. Rep.* **11**, 321 (2021).
21. Li, C. et al. Geometry-based molecular generation with deep constrained variational autoencoder. In *IEEE 21st International Conference on Bioinformatics and Bioengineering (BIBE)* (IEEE, 2021).
22. Kang, S.-G. et al. In-pocket 3D graphs enhance ligand–target compatibility in generative small-molecule creation. Preprint at https://arxiv.org/abs/2204.02513 (2022).
23. Ragoza, M., Masuda, T. & Koes, D. R. Generating 3D molecules conditional on receptor binding sites with deep generative models. *Chem. Sci.* **13**, 2701–2713 (2022).
24. Liu, M., Luo, Y., Uchino, K., Maruhashi, K. & Ji, S. Generating 3D molecules for target protein binding. Preprint at https://arxiv.org/abs/2204.09410 (2022).
25. Luo, S., Guan, J., Ma, J. & Peng, J. A 3D generative model for structure-based drug design. In *Advances in Neural Information Processing Systems* Vol. 34, 6229–6239 (NeurIPS, 2021).
26. Peng, X. et al. Pocket2Mol: efficient molecular sampling based on 3D protein pockets. Preprint at https://arxiv.org/abs/2205.07249 (2022).
27. Francoeur, P. G. et al. Three-dimensional convolutional neural networks and a cross-docked data set for structure-based drug design. *J. Chem. Inf. Model.* **60**, 4200–4215 (2020).
28. Isert, C., Atz, K. & Schneider, G. Structure-based drug design with geometric deep learning. *Curr. Opin. Struct. Biol.* **79**, 102548 (2023).
29. Rudd, R. E. & Broughton, J. Q. Coarse-grained molecular dynamics and the atomic limit of finite elements. *Phys. Rev. B* **58**, R5893 (1998).
30. Senn, H. M. & Thiel, W. QM/MM methods for biomolecular systems. *Angew. Chem. Int. Ed.* **48**, 1198–1229 (2009).
31. Jing, B., Eismann, S., Suriana, P., Townshend, R. J. & Dror, R. Learning from protein structure with geometric vector perceptrons. Preprint at https://arxiv.org/abs/2009.01411 (2020).
32. Moon, S., Zhung, W., Yang, S., Lim, J. & Kim, W. Y. PIGNet: a physics-informed deep learning model toward generalized drug–target interaction predictions. *Chem. Sci.* **13**, 3661–3673 (2022).
33. Trott, O. & Olson, A. J. AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *J. Comput. Chem.* **31**, 455–461 (2010).
34. *RDKit Documentation* (RDKit, 2021); https://www.rdkit.org/docs/
35. Menéndez, M., Pardo, J., Pardo, L. & Pardo, M. The Jensen–Shannon divergence. *J. Franklin Inst.* **334**, 307–318 (1997).
36. Kullback, S. & Leibler, R. A. On information and sufficiency. *Ann. Inst. Stat.* **22**, 79–86 (1951).
37. Riniker, S. & Landrum, G. A. Better informed distance geometry: using what we know to improve conformation generation. *J. Chem. Inf. Model.* **55**, 2562–2574 (2015).
38. Rappé, A. K., Casewit, C. J., Colwell, K., Goddard, W. A. III & Skiff, W. M. UFF, a full periodic table force field for molecular mechanics and molecular dynamics simulations. *JACS* **114**, 10024–10035 (1992).
39. Ganea, O. et al. Geomol: torsional geometric generation of molecular 3D conformer ensembles. In *Advances in Neural Information Processing Systems* Vol. 34, 13757–13769 (NeurIPS, 2021).
40. Simm, G. N. & Hernández-Lobato, J. M. A generative model for molecular distance geometry. Preprint at https://arxiv.org/abs/1909.11459 (2019).
41. Shi, C., Luo, S., Xu, M. & Tang, J. in *Proc. 38th International Conference on Machine Learning* Vol. 139 (eds Melia, M. & Zhang, T.) 9558–9568 (PMLR, 2021).
42. Xu, M., Luo, S., Bengio, Y., Peng, J. & Tang, J. Learning neural generative dynamics for molecular conformation generation. Preprint at https://arxiv.org/abs/2102.10240 (2021).
43. Luo, S., Shi, C., Xu, M. & Tang, J. Predicting molecular conformation via dynamic graph score matching. In *Advances in Neural Information Processing Systems* Vol. 34 (NeurIPS, 2021).
44. Salentin, S., Schreiber, S., Haupt, V. J., Adasme, M. F. & Schroeder, M. PLIP: fully automated protein–ligand interaction profiler. *Nucleic Acids Res.* **43**, W443–W447 (2015).
45. Huang, Y., Peng, X., Ma, J. & Zhang, M. 3DLinker: an E(3) equivariant variational autoencoder for molecular linker design. Preprint at https://arxiv.org/abs/2205.07309 (2022).
46. Anderson, P. W. More is different: broken symmetry and the nature of the hierarchical structure of science. *Science* **177**, 393–396 (1972).
47. Comez, L. et al. More is different: experimental results on the effect of biomolecules on the dynamics of hydration water. *J. Phys. Chem. Lett.* **4**, 1188–1192 (2013).
48. Ingraham, J., Garg, V., Barzilay, R. & Jaakkola, T. Generative models for graph-based protein design. In *33rd Conference on Neural Information Processing Systems* Vol. 32 (NeurIPS, 2019).
49. Gardner, M. W. & Dorling, S. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmos. Environ.* **32**, 2627–2636 (1998).
50. Méndez-Lucio, O., Ahmad, M., del Rio-Chanona, E. A. & Wegner, J. K. A geometric deep learning approach to predict binding conformations of bioactive molecules. *Nat. Mach. Intell.* **3**, 1033–1039 (2021).
51. Bishop, C. M. *Mixture Density Networks* (Aston Univ., 1994).
52. Zou, L. et al. GMDN: a lightweight graph-based mixture density network for 3D human pose regression. *Comput. Graph.* **95**, 115–122 (2021).
53. Chen, J., Yu, Y. & Liu, Y. Physics-guided mixture density networks for uncertainty quantification. *Reliab. Eng. Syst. Saf.* **228**, 108823 (2022).
54. Hoogeboom, E., Satorras, V. G., Vignac, C. & Welling, M. Equivariant diffusion for molecule generation in 3D. In *Proc. 39th International Conference on Machine Learning* 8867–8887 (PMLR, 2022).
55. Sproul, G. Electronegativity and bond type: predicting bond type. *J. Chem. Educ.* **78**, 387 (2001).
56. Jumper, J. et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
57. Vaswani, A. et al. Attention is all you need. In *Advances in Neural Information Processing Systems* Vol. 30 (NeurIPS, 2017).
58. Shen, Z., Zhang, M., Zhao, H., Yi, S. & Li, H. in *Proc. IEEE/CVF Winter Conference on Applications of Computer Vision* 3531–3539 (IEEE, 2021).
59. Wang, G., Ying, R., Huang, J. & Leskovec, J. Multi-hop attention graph neural network. Preprint at https://arxiv.org/abs/2009.14332 (2020).
60. Lu, W. et al. TANKBind: trigonometry-aware neural networks for drug–protein binding structure prediction. Preprint at https://www.biorxiv.org/content/10.1101/2022.06.06.495043v1 (2022).
61. Lewis, G. N. The atom and the molecule. *J. Am. Chem. Soc.* **38**, 762–785 (1916).
62. Yu, L., Su, Y., Liu, Y. & Zeng, X. Review of unsupervised pretraining strategies for molecules representation. *Brief. Funct. Genom.* **20**, 323–332 (2021).
63. Hu, W. et al. Strategies for pre-training graph neural networks. Preprint at https://arxiv.org/abs/1905.12265 (2019).
64. Zhu, J. et al. Unified 2D and 3D pre-training of molecular representations. In *Proc. 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* 2626–2636 (ACM, 2022).
65. Lamb, A. M. et al. Professor forcing: a new algorithm for training recurrent networks. In *Advances in Neural Information Processing Systems* Vol. 29 (NeurIPS, 2016).

66. Drossos, K., Gharib, S., Magron, P. & Virtanen, T. Language modelling for sound event detection with teacher forcing and scheduled sampling. Preprint at https://arxiv.org/abs/1907.08506 (2019).
67. Tanimoto, T. T. *An Elementary Mathematical Theory of Classification and Prediction* (International Business Machines Corporation, 1958).
68. Axelrod, S. & Gómez-Bombarelli, R. GEOM, energy-annotated molecular conformations for property prediction and molecular generation. *Sci. Data* **9**, 185 (2022).
69. Zhang, H. et al. SDEGen: learning to evolve molecular conformations from thermodynamic noise for conformation generation. *Chem. Sci.* **14**, 1557–1568 (2023).
70. Xu, M. et al. An end-to-end framework for molecular conformation generation via bilevel programming. In *Proc. 38th International Conference on Machine Learning* 11537–11547 (PMLR, 2021).
71. Clark, D. E. & Pickett, S. D. Computational methods for the prediction of 'drug-likeness'. *Drug Discov. Today* **5**, 49–58 (2000).
72. Ertl, P. & Schuffenhauer, A. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *J. Cheminform.* **1**, 1–11 (2009).
73. Ganesan, A. The impact of natural products upon modern drug discovery. *Curr. Opin. Chem. Biol.* **12**, 306–317 (2008).
74. Sangster, J. Octanol-water partition coefficients of simple organic compounds. *J. Phys. Chem. Ref. Data* **18**, 1111–1229 (1989).

Acknowledgements

This study was supported by the National Key Research and Development Program of China (grant no. 2022YFF1203000), the National Natural Science Foundation of China (grant no. 22220102001), the Fundamental Research Funds for the Central Universities (grant no. 226-2022-00220) and the Hong Kong Innovation and Technology Fund (project no. ITS/241/21).

Author contributions

O.Z. contributed to the main idea and code. J.Z. contributed to the manuscript writing and code reorganization. X.Z. contributed to the collection of the dataset and the corresponding experiment. R.H. and J.J. contributed to the curation of the real-world dataset. C.S. and H.D. contributed to the data analysis and drawing. H.C. and Y.K. contributed to the instruction in physical concepts. Y.D. contributed to the visualization and technical support. F.L. contributed to the suggestion of the geometry analysis metric. G.C. and C.-Y.H. contributed to manuscript revision and experimental design. T.H. contributed to the essential financial support, the conceptualization, and was responsible for the overall quality.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-023-00712-7.

Correspondence and requests for materials should be addressed to Furui Liu, Guangyong Chen, Chang-Yu Hsieh or Tingjun Hou.

Peer review information *Nature Machine Intelligence* thanks Arne Elofsson and Guo-Wei Wei for their contribution to the peer review of this work.