Noname manuscript No.
(will be inserted by the editor)

AI-driven prediction of SARS-CoV-2 variant binding trends from atomistic simulations

Sara Capponi[1,2,a], **Shangying Wang**[1,2,a], **Simone Bianco**[1,2,b]

[1]IBM Almaden Research Center, 650 Harry Rd, San Jose, CA 95120, USA
[2]Center for Cellular Construction, San Francisco, CA, 94158, USA

Received: date / Accepted: date

Abstract We present a novel technique to predict binding affinity trends between two molecules from atomistic molecular dynamics simulations. The technique uses a neural network algorithm applied to a series of images encoding the distance between two molecules in time. We demonstrate that our algorithm is capable of separating with high accuracy mutations with low binding affinity from mutations with high binding affinity. Moreover, we show high accuracy in prediction using a small subset of the simulation, therefore requiring a much shorter simulation time. We apply our algorithm to the binding between several variants of the the SARS-CoV-2 spike protein and the human receptor ACE2.

1 Introduction

The ongoing COVID-19 pandemic, caused by the infection with the RNA betacoronavirus SARS-CoV-2, has already resulted in over 110 million cases and 2 million deaths around the world [1]. As the global response to the pandemic moves from non-pharmaceutical (e.g., social distancing, periodic lockdowns, mandate of personal protective equipment) to pharmaceutical (e.g., vaccination) forms of intervention, important questions arise about the evolution of the virus and the emergence of genetic variants which may escape the drug. Coronavirus viral particles contain four structural proteins [2]: the glycoprotein spike (S), the membrane (M), the envelope (E), and nucleocapsid (N) proteins. The S protein, which is the main focus of much of the ongoing research, is a trimeric class I membrane fusion protein responsible for binding the host cell receptor and triggering the fusion between the viral membrane and the host cell membrane [3–6]. The S protein exists in a prefusion metastable state and undergoes large conformational changes transitioning to a stable postfusion conformation once it binds to a host cell receptor [3, 6–8]. Each S protomer (see Fig.1A) comprises two functional subunits: S1 contains the receptor binding domain (RBD) and is responsible for binding the host cell receptor; S2 contains the fusion machinery and is responsible for the membrane fusion between the virus and the host cell. The first cryo-EM structures of the SARS-CoV-2 S protein in the prefusion conformation, published in March 2020 [9, 10], revealed fundamental structural details providing the basis for identifying conserved and accessible epitopes for future antibody isolation and vaccine design efforts. In addition, several structures of the SARS-CoV-2 RBD in association with ACE2, published between March and May 2020, helped uncover the molecular mechanisms of the interaction between the RBD and ACE2 and to identify residues that are crucial for the binding process [11–14].

Being the RBD the most variable part of the viral genome [15], several studies have focused on understanding the effects mutations have on the binding affinity to ACE2 [16–18]. Atomistic molecular dynamics (MD) simulations have been used since early in the pandemic to study and visualize the binding between the S and the ACE2 receptor and explore potential targets for drugs [19–21]. An important limit of these tools is the exceptional computational burden necessary to carry out the simulations to an extent that is meaningful to answer biological questions. To estimate the free energy of binding between the RBD and ACE2, and more in general between two proteins, one would need to carry out one simulation of the association process starting from separate protein molecules, and another simulation sampling the dissociation process. However, the association and dissociation rates can be very slow, much beyond the current atomistic MD simulation accessible time scale, making it

[a]These authors contributed equally to this work.
[b]e-mail: sbianco@us.ibm.com

hard to estimate directly the binding free energy from sampling association and dissociation events *in silico*. Therefore, several theoretical approaches have been developed to solve this issue. Among others, Markov state modeling [22–24], Hamiltonian replica exchange methods [25], weighted ensemble approaches [26, 27], or adding a bias potential (umbrella sampling) to guide the association or dissociation process along a progress coordinate [28, 29] are among the most common tools to study the binding process between two molecules. Once an approach has been selected, the following biological question arises: how can we probe efficiently *in silico* the effects of site mutations on the protein-protein interaction and on the binding affinities? The answer to this and other important questions is increasingly relying on machine learning (ML) methods, whose application in computational biochemistry is gaining momentum.

Indeed, artificial intelligence (AI) has recently emerged as a potential accelerator of MD simulations [30–32]. The majority of ML methods have focused on learning the force field [31], that is, the complex set of rules and parameters governing the interaction among atoms and thus between molecules. Recently, an increasing amount of works devoted to the prediction of experimentally measurable quantities from MD has emerged [30, 33, 34]. In this paper we present a neural network-based method to predict molecular binding trends, that is, whether or not the binding affinity of a target molecular interaction is higher or lower than a reference interaction, and reduce the computational burden of the molecular simulations. Rather than learning the sequence to structure relationship as most AI methods in this space do [35], our method learns the relative atomic distance between molecules over the simulation time and uses this information to predict the molecular association or dissociation trend. We apply our method to evaluate binding between mutants of the SARS-CoV-2 RBD and the ACE2 receptor.

Our goals are twofold: First, we aim to demonstrate that the combination of AI and MD simulations allows for the prediction of the binding affinity trend between two molecules. While it is difficult to measure affinity through MD for systems of the size of the RBD-ACE2 complex, we prove that, if that information is experimentally known for a set of mutants, a neural algorithm is able to predict the correct affinity trend of an unknown mutation. Second, we prove that our AI algorithm can reveal the binding affinity trend using the information extracted from simulations significantly shorter than those used for training, thereby significantly reducing the time to obtain the prediction.

Our paper is structured as follows: The next section introduces our MD and AI methodologies, as well as the data. Next we report the results our analysis. Finally we discuss the potential impact of our work and future directions.

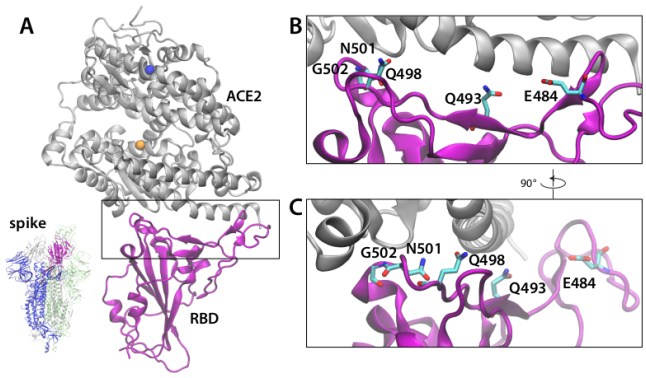

Fig. 1 **A**. Representative snapshopt of the chimeric spike receptor binding domain (RBD) (purple) in association with ACE2 (gray). Proteins are represented in New Cartoon format. Zinc (orange) and chloride (blue) ions are represented in van der Waals format. Bottom left: representation of the spike trimeric complex from [9], one chain is blue with the RBD colored purple, the other 2 chains are transparent gray and green for clarity. **B-C**. Closeup view of the contact region between RBD and ACE2. The residues that are mutated and studied in this paper are indicated. They are represented in licorice format colored according to the ion type (red for oxygen, cyan for carbon, blue for nitrogen). Hydrogen atoms are not displayed for clarity. **C**. Same representation of panel **B** rotated of 90°.

2 Data and Methods

2.1 Reference data

We obtain binding affinity for twelve mutations of the S protein RBD - ACE2 receptor from the recent literature [16], six with affinity higher and six with affinity lower than the wild type (WT), which we use as reference. The list is reported in Table 1 according to their decreasing value of binding affinity. $\Delta \log_{10}(K_{D,app})$ quantifies the relative affinity value compared to the unmutated SARS-CoV-2 RBD, with positive values indicating stronger binding, and negative values indicating weaker binding. We choose two medically important mutations, N501Y (present in the English variant) and E484K (present in the Brazilian variant), two mutations which are present in SARS-CoV, N501T and Q493N, and two mutations from the bat coronavirus RaTG13. Finally, we randomly select five additional variants according to their affinity, for the purpose of balancing our training set.

2.2 Molecular dynamics simulations

To carry out the simulations of the SARS-CoV-2 RBD in association with the ACE2 receptor, we used the atomic coordinates extracted from the X-ray crystal structure of the SARS-CoV-2 chimeric RBD in complex with the human receptor ACE2 (PDB ID 6VW1) [36]. The structure has a resolution of 2.68 Å and is a chimera of the RBD from SARS–CoV and the receptor binding motif (RBM) from SARS–CoV–2, with the exception of an RBM loop. For our work,

we considered this structure as reference for the WT binding affinity, in analogy with the WT used as reference for the experimental work [16]. We used VMD software [37] to model the disulfide bonds, to solvate the system, add the zinc and chloride ions present in the crystal structure, and add the potassium ions to maintain the system charge neutrality. The final system contains approximately 132000 atoms, with $\approx$ 40000 water molecules.

From the equilibrated structure of the RBD-ACE2 system, we extracted the last configuration and used VMD software [37] to build the 12 mutated systems used for this study and reported in Table 1. The procedure to minimize and equilibrate the system was the same for WT and all mutations, and it is described in Section 2.2.1. The simulations had variable lengths, ranging from 160 to 270*ns* (Table 2).

All simulations were performed using NAMD 2.14 [38, 39] with the CHARMM36m force field for the protein and ions [40] and the TIP3P model for water [41]. We used a Langevin dynamics scheme to keep the temperature constant at 303.15 K and an anisotropic coupling in conjunction with Nosé-Hoover-Langevin piston algorithm to keep the pressure constant at 1 atm [42, 43]. Periodic boundary conditions were applied in three dimensions. We employ the smooth particle-mesh Ewald summation method to calculate the electrostatic interactions [44, 45] and the short-range real-space interactions were cutoff at 10 Å using a switching function between 8 Å and 10 Å. The equations of motion were integrated with a time step of 4 *fs* for the long-range electrostatic forces, 2 *fs* for the short-range non-bonded forces, and 2 *fs* for the bonded forces by means of a reversible, multiple time-step algorithm [46]. The SHAKE algorithm [47] was used to constrain the length of the bonds involving hydrogen atoms. The simulations were visualized using VMD software [37].

2.2.1 System Equilibration Protocol

We used the following procedure to minimize and equilibrate all the systems. To minimize the system, we used the conjugate gradient algorithm for 8,000 steps and then gradually heated the simulated cell from 25 K to 300 K. To equilibrate the positions of ions, water molecules, and protein complex, harmonic restraints were used in a series of five consecutive simulation runs of 1 *ns*, as follows: We ran 1 NVT (constant number of particles N, volume V, and temperature T) and 4 NPT (constant number of particles N, pressure P, and temperature T) equilibration runs. In the first equilibration run, restraint force constants of 20 kcal mol^{-1} Å^{-2} were applied to the protein backbone atoms and 5 kcal mol^{-1} Å^{-2} to ions and water. In the subsequent runs, the restraints on the protein backbone were decreased to 10, 8, 4, 2, and then to 1 kcal mol^{-1} Å^{-2}, while restraints on ions and water were decreased to 2, 1, 0.5, 0.1 and then 0.01 kcal mol^{-1} Å^{-2}, respectively. After the last 1 ns equilibration run, we began the production runs. The simulations were carried out under NPT conditions and the coordinates were saved every 20*ps*.

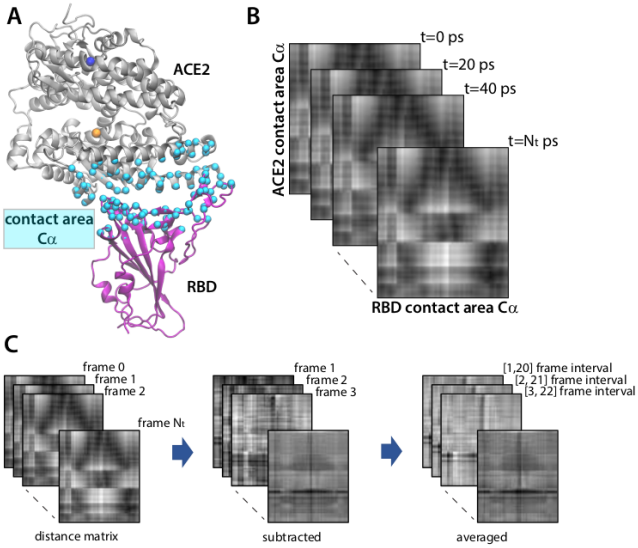

Fig. 2 Conversion of the MD data to input matrices for a neural algorithm. **A**. Representative snapshot of the chimeric RBD-ACE2 complex indicating the Cα carbon atoms used to calculate the distance between the RBD and ACE2. The color code is as in Figure 1 and the Cα carbon atoms are represented in cyan, van der Waals format. The distances between the atoms are then represented as matrix, are normalized by each of their minima and maxima, and converted to gray-scaled images. **B**. Representative distance matrices of the chimeric RBD-ACE2 complex simulations taken at different steps of the simulation showing the effect of data preprocessing. **C**. Representation of the data processing to obtain machine learning inputs. We first subtract the distance matrix at $t = 0$ from the distance matrices from the simulations at each time steps (central panel). We then use a moving average to average the subtracted matrices and use these as input to the algorithm (right-hand panel).

For all systems, we monitored visually the simulations and calculated the root mean square deviation (RMSD) of all the Cα carbon atoms belonging to both proteins and of the Cα carbon atoms belonging to the RBD and ACE2 contact area (see Figures 6 and 7; Section 2.2.2). The contact area appears to be rather rigid in almost all systems with Cα RMSDs $<$ 2 Å. Only in the mutation N501D the RMSD value reaches $\approx$ 3 Å between 180 ns and the end of the simulation. In few simulations (N501D, N501S, Q493N, Q498Y, Q498N, Q498K) and at different time points of the simulation, we observe escape of the Cl ion from the ACE2 binding pocket. We do not observe reduced stability due to loss of Cl, and verify that it does not impact the RMSD throughout the affected simulations.

2.2.2 Training set preparation

The result of the MD simulations is a time series of molecular snapshots. We use the following procedure to convert the data into a training set for our machine learning algorithm. From the long simulation of the RBD in association with ACE2, we extracted the residue identification number of the Cα carbon atoms of the RBD that happen to be within 6 Å of ACE2 receptor at least for a single frame during the whole simulation length. We repeated the same analysis for ACE2, obtaining two lists of residues (55 amino acids for the RBD and 62 for ACE2 respectively) that represent the amino acids found in one protein within 6 Å from the other and viceversa (Fig. 2A); for the complete list of amino acids see Appendix A. We called this region as RBD and ACE2 contact area. For each system, we calculated the distances among all the Cα carbon atoms of the residues belonging to the two lists for each frame. For this, we used the contact and distances modules from MDAnalysis [48, 49]. Finally, we used the matrices of the distances to construct images that can be interpreted by our machine learning algorithm as follows.

We recorded the distance matrices every 20ps to up to 160ns, i.e., 8000 molecular snapshots were collected and preprocessed. Simulations of under 160ns were not sufficiently long to directly reveal the binding trend. The dissociation constants of typical protein–protein complexes are in the nanomolar range and the dissociation rate can be of the order of minutes or longer, much beyond the timescale of current MD simulations. The time evolution of the distance matrices shows very similar patterns for all mutations (Fig. 2B and also the supp videos). Therefore, we perform the following procedure to highlight the difference in the molecular dynamic due to the mutations. The signals are buried in the structural restrictions of the positions of the amino acids. In order to observe how the atoms move due to the contact between RBD and ACE2, we first subtract from all matrices the initial frame of coordinates after completion of minimization and equilibration procedures, which we indicate as $t = 0$ (see 2.2.1). We then average the signal using a moving average of 20 frames (or 400 ps) to reduce noise (Fig. 2C). We then normalize the averaged matrices using min-max normalization to make sure all values in the matrices are between 0 and 1. Finally, we converted them to gray-scale images and use them as neural network inputs for training, validation and testing. In Figure 3 we show a representative image for each mutation analyzed. The training and validation sets were built using a randomly chosen 90/10 split of all input images from three mutations higher than the reference (namely, N501Y, N501V, and N501T) and three mutations lower than the reference (namely, N501S, Q498N, and N501D). All images from the remaining six mutations were used for testing.

Mutation	**Origin**	Δ **log**$_{10}$(**K**$_{D,app}$)
N501Y	English variant	0.24
Q498Y	SARS-CoV	0.16
N501V	-	0.15
Q493Y	Bat RaTG13	0.12
N501T	SARS-CoV	0.1
E484K	Brazilian variant	0.06
N501S	-	-0.13
Q493N	SARS-CoV	-0.21
Q498N	-	-0.5
Q498K	-	-2.26
N501D	Bat RaTG13	-2.42
G502P	-	-4.55

Table 1 List of the mutations of the spike RBD studied in this paper. The second column reports the origin of the mutation and the third column the binding affinity extracted from Ref [16]. The mutations are listed according to decreasing values of the binding affinity.

2.3 Neural Network Framework

2.3.1 Neural network Architecture

In this work we used a Convolutional Neural Network (CNN) implemented in Python 3.7.8 and TensorFlow v2.4.1. The neural network architecture is illustrated in Figure 4A. The neural network uses 62×55 grayscale images as inputs, and outputs two values corresponding to the probabilities of higher and lower affinity compared to the WT (P_1 and P_2, respectively). Between the input and output layers, we have 2 convolutional layers and 2 maximum pooling layers. Convolutional layers detect spatial correlations in input data and maximum pooling layers down-sample input data to reduce dimensionality and the number of model parameters. We used RandomNormal initializer and ReLU activation function for the convolutional layers. The outputs of the second maximum pooling layer was then flattened. We used dropout regularization to the flattened array with dropout rate equal to 0.5 to prevent over-fitting. The flattened layer then connected to a dense layer containing 5 nodes. We used softmax activation function to connect this dense layer and the final output layer to convert the outputs into categorical probabilities, i.e. $P_1 + P_2 = 1$. If $P_1 > P_2$, the input will be categorized as having higher than WT affinity and vice versa (Fig. 4C). For training, we used sparse categorical cross entropy loss and Adam optimization algorithm with the initial learning rate equals 0.0001, and training over 30 epochs.

2.3.2 Ensemble Predictions

Due to the stochastic nature of network initialization and dropout, as well as the availability of a limited training set,

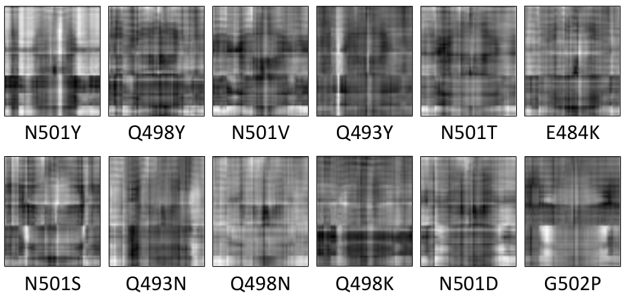

Fig. 3 Sample gray-scaled images representing the averaged matrices for all the mutations studied in this paper at $t = 20ns$.

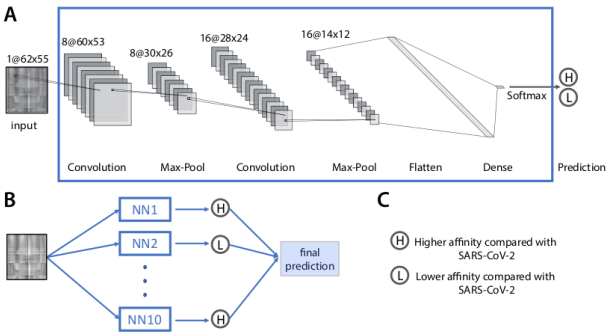

Fig. 4 Structure of the convolutional neural network and ensemble prediction used in this study. **A**. The convolutional neural network architecture. **B**. The ensemble prediction consisting of N independently trained convolutional neural networks. The final prediction is based on the simple majority of all neural networks. **C**. Definition of the output categories.

every neural network is unique in terms of the parameterization of the network connections [50]. To mitigate the potential impact of this issue on the classification, we implemented an ensemble decision method to get consensus prediction: For each mutation, we train ten identical neural networks. We then use a simple majority to assign the final class to the mutation (Figure 4B). This method has been proven to further reduce prediction errors and bias and increase accuracy and reliability, especially with limited training data.

3 Results

3.1 An ensemble of CNNs can reliably predict binding trends

The results of our ensemble procedure are reported in Tables 3 to 8, and summarized in Table 9. Every row represents a different network run, while every column is the result of testing using only a subset of the data. The row labeled FP is the final prediction for the ensemble, summarized by the accuracy of prediction in the row labeled PPC. We first focus on the predictions obtained using the whole MD simulation for testing (last column in the Tables, $t = 160ns$).
Nine out of ten trained neural networks can predict the six

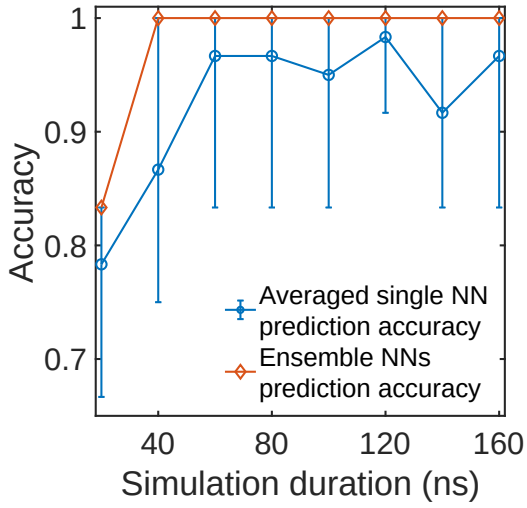

Fig. 5 The overall test accuracy of CNNs varies with the simulation time. The blue dots are the average prediction accuracy from ten independently trained CNNs. The error bars represent the 10th percentile and 80th percentile. The orange diamonds represent the prediction accuracy of the ensemble prediction, voted by 10 NN assembles.

mutations in the test set with 100% accuracy, while one neural network can predict four out of six mutations in the test set, with an overall average accuracy of 0.9667. Thus, we conclude that our methods if optimal in predicting binding trend with respect to a reference with exceptionally high accuracy. Moreover, by using the ensemble prediction method, we show that we successfully eliminate the possibility of an incorrect prediction due to faulty training, and increase our confidence when making a call on a novel mutation.

3.2 An ensemble of CNNs can accurately predict binding trends using shorter MD simulation times

We now discuss the problem of how much MD data is needed to correctly predict whether a mutation has higher or lower than WT affinity using a CNN trained in the way we describe. As stated above, MD simulations of minutes or longer are needed to observe binding or dissociation. In the previous sections, we showed that, when the ground truth is known, simulations of $160ns$ of systems of the RBD-ACE2 complex size are sufficient to train neural networks which can reliably predict binding trends for unknown mutations.

However, MD simulations of 160*ns* still require extraordinarily long computational times and resources. This limits the ability to simulate important molecular processes in situations where rapid responses would be necessary. To give an estimate of the necessary resources, on the Artificial Intelligence Multiprocessing Optimized System (AiMOS) supercomputer we are able to simulate $\approx 33ns$ per simulation per day using 8 GPUs, while we are able to simulate $\approx 7ns$ per day on a cloud server provisioned with 2 latest generation GPUs. In this section, we test whether or not it is possible to use our CNN to accurately predict binding trends using shorter MD simulations. We varied the simulation duration from 20*ns* to 160*ns*, with increments of 20*ns*. Our results are reported in Tables 3 to 8 for single mutations predictions, and summarized in Table 9 and Figure 5.

Figure 5 reports the average prediction accuracy of ten independently trained neural networks as well as the ensemble accuracy as a function of simulation time. The error bars for the averaged predictions represent the 10th percentile and 80th percentile of the prediction accuracy (the 90th percentile is equal to the 80th percentile, see Table 9 for details). We can see that the average prediction accuracy for independently trained CNNs are consistently over 90% with the simulation duration equal or longer than 60*ns*. Taking advantage of the ensemble prediction method, the accuracy can be as high as 100% with simulation duration as short as 40*ns*. The ensemble prediction significantly increases the prediction accuracy and reliability while allowing for a $4x$ shorter simulation time. Furthermore, w observe that some mutations such as E484K, Q493N, and G502P can be correctly classified with high confidence ($> 75\%$) with only 20*ns* of simulation. While a deeper understanding of the reasons behind this result is beyond the scope of this work, we suspect that their molecular snapshots have some similarity to the snapshots that are already included in the training set. And vice versa, the mutations that are relatively more challenging for the network to classify at shorter time horizons may show more differences with respect to those included in the training set. A training set composed of a more diverse set of mutations will possibly result in more reliable predictions. Nonetheless, our results prove that our CNN is capable of generalizing with sufficiently high accuracy at very short simulation times when the training set is composed of only a handful of mutations.

4 Conclusions and future directions

The emergence of viral variants in a pandemic epitomizes the evolutionary arm race between the virus and the host immune system. As vaccines are deployed to a growing fraction of the population, questions regarding drug resistance and escape will likely dominate the public and scientific discourse of the near future. Therefore, time is of the essence to determine whether a specific genetic variant has a higher or lower potential to bind to the human cellular receptor. MD simulations are an important tool to address these issues. Methods that accelerate discovery are now necessary for these computational approaches to yield timely responses. Once trained our CNN takes less than a second to perform a classification task.

Beyond the application presented in this paper, we believe that a method like the one we propose may be applied generally to other problems where assessing binding affinity trends using only MD simulations may be important, like drug design or cellular engineering. The generalization of these methods is an important argument of research which we are planning to explore in the future. Concurrently, we realize that our methods does not provide an exact measure of the affinity. Recent papers have proposed methods capable of achieving that goal, with mixed but encouraging results [51]. Another future objective of our research is to generalize our method to obtain estimates of binding affinity using limited MD simulations.

Appendix A: MD simulation details

In table 2 we report the length of the simulations we carried out in this work. To monitor the simulations, we calculated the RMSD of the C_α atoms of the whole proteins and of the *contact area* between RBD and ACE2. For the *contact area* of the RBD, we considered the C_α of the following residues: 403, 405, 406, 408, 416 to 418, 421, 437, 439, 440, 444 to 449, 452, 453, 455 to 458, 472 to 478, 484 to 508. For the *contact area* of ACE2, we considered the C_α of the following residues: 19 to 39, 41, 42, 45, 48, 49, 61, 72, 75 to 84, 97, 323 to 332, 351 to 357, 386 to 390, 393.

simulation	lenght (ns)
cRBD-ACE2	270
N501Y-ACE2	187
Q498Y-ACE2	206.84
N501V-ACE2	204.86
Q493Y-ACE2	197.28
N501T-ACE2	186.58
E484K-ACE2	204.56
N501S-ACE2	205.2
Q493N-ACE2	197.18
Q498N-ACE2	198.84
Q498K-ACE2	204.46
N501D-ACE2	210.44
G502P-ACE2	160.58

Table 2 List of the simulations carried out in this work and of the corresponding length.

Appendix B: Details of CNN predictions

Table 3, 4, 5, 6, 7 and 8 list the predictions from 10 CNNs for mutation Q498Y (H), Q493Y (H), E484K (H), Q493N (L), Q498K (L) and G502P (L) respectively. These 10 neural networks have the same architecture and are independently trained. The neural network weights are randomly initialized from a normal distribution. Neural network prediction is the probability of higher affinity (P_1) and the probability of lower affinity (P_2) compared to the WT, with the constraint of $P_1 + P_2 = 1$. H and L is the neural network prediction. H represents higher the WT affinity ($P_1 > P_2$) and L represents lower than WT affinity ($P_1 < P_2$). The value in the brackets is the probability value of the predicted affinity. (P_1 for H and P_2 for L). FP is short for final prediction and PPC represents the percentage of CNNs predicting correctly. When training for much smaller data size (simulation ends at 20ns and 40ns), we trained the CNNs over 60 epochs, while for the rest 30 epochs are used.

Table 9 lists the prediction accuracy of 80 CNNs for all 6 test mutations with different simulation duration, i.e., Q498Y, Q493Y, E484K, Q498K, Q493N and G502P. This table also contains the average accuracy of 10 CNNs and the ensemble prediction accuracy per simulation end time.

Appendix C: Data availability

The distance matrices used to train the neural network are available at the following url:
https://ibm.box.com/v/DistanceMatricesMD
Supplementary videos are at the following url:
https://ibm.box.com/v/SupplementaryVideos

Acknowledgements We want to thank Alex Lenail for developing the online drawing tool called NN-SVG (https://alexlenail.me/NN-SVG/AlexNet.html). We used it to plot the architecture of the neural network we used in this study. We acknowledge support from the IBM Research AI Hardware Center, and the Center for Computational Innovation at Rensselaer Polytechnic Institute for computational resources on the AiMOS Supercomputer. This material is based upon work supported by the National Science Foundation under Grant No. DBI-1548297.

References

1. WHO Coronavirus Disease (COVID-19) Dashboard. https://covid19.who.int/
2. H. J. Maier, E. Bickerton, and P. Britton, *Coronaviruses: Methods and Protocols*, 1–282 (2015). https://doi.org/10.1007/978-1-4939-2438-7
3. F. Li, Annu. Rev. Virol. **3**, 237-61 (2016)
4. S. Belouzard, J. K. Millet, B. N. Licitra,G. R. Whittaker. Viruses. **4**, 1011-1033 (2012)
5. S. Duquerroy, A. Vigouroux, P. J. M. Rottier, F. A. Rey, B. J. Bosch. Virology **335**, 176-185 (2005)
6. A. C. Walls, M. A. Tortorici, J. Snijder, X. Xiong, B.-J. Bosch, F. A. Rey, D. Veesler. Proc Natl Acad Sci USA **114**, 11157-11162 (2017)
7. B. J. Bosch, R. van der Zee, C. A. de Haan, P. J. Rottier, J. Virol. **77**, 8801–8811 (2003)
8. M. A. Tortorici, D. Veesler. Advances in Virus Research, **105**, 94-108 (2019)
9. D. Wrapp, N. Wang, K. S. Corbett, J. A. Goldsmith, C.-L. Hsieh, O. Abiona, B. S. Graham, J. S. McLellan, Science **367**, 1260-1263 (2020)
10. A. C. Walls, Y.-J. Park, M. A. Tortorici, A. Wall, A. T. McGuire, D. Veesler. Cell **180**, 1-12 (2020)
11. J. Lan, J. Ge, J. Yu, S. Shan, H. Zhou, S. Fan, Q. Zhang, X. Shi, Q. Wang, L. Zhang, X. Wang. Nature **581** 215-220 (2020)
12. J. Shang, G. Ye, K. Shi, Y. Wan, C. Luo, H. Aihara, Q. Geng, A. Auerbach, F. Li. Nature **581** 221-224 (2020)
13. Q. Wang, Y. Zhang, L. Wu, S. Niu, C. Song, Z. Zhang, G. Lu, C. Qiao, Y. Hu, K.-Y. Yuen, Q. Wang, H. Zhou, J. Yan, J. Qi. Cell **181**, 894-904 (2020)
14. R. Yan, Y. Zhang, Y. Li, L. Xia, Y. Guo, Q. Zhou. Science **367**, 1444-1448 (2020)
15. P. Zhou, X.L. Yang, XG. Wang, *et al.* Nature **579**, 270–273 (2020). https://doi.org/10.1038/s41586-020-2012-7
16. T. N. Starr, *et al.* Cell, **182**(5), 1295-1310.e20 (2020). https://doi.org/10.1016/j.cell.2020.08.012
17. A.J. Greaney, *et al.*, Cell Host and Microbe, **29**(1), 44-57.e9 (2021). https://doi.org/10.1016/j.chom.2020.11.007
18. A. K. Padhi and T. TripathiACS Pharmacology and Translational Science, **3**(5), 1023–1026 (2020). https://doi.org/10.1021/acsptsci.0c00113
19. P.R. Arantes, A. Saha, and G. Palermo, ACS Central Science, 6(10), 1654–1656 (2020). https://doi.org/10.1021/acscentsci.0c01236
20. C. Peng, *et al*, ChemrXiv (2020). https://doi.org/10.26434/chemrxiv.11877492.v1
21. G. Deganutti, F. Prischi, and C.A. Reynolds, Journal of Computer-Aided Molecular Design, **35**(2), 195–207 (2020). https://doi.org/10.1007/s10822-020-00356-4
22. N. Plattner, S. Doerr, G. De Fabritiis G, F. Noé. Nat Chem. **9**, 1005-1011 (2017).
23. G. R. Bowman, V. S. Pande, F. Noé. Dordrecht: Springer Science and Business Media. **148** (2014).
24. J. D. Chodera, F. Noé. Curr Opin Struct Biol. **25**, 135-144 (2014)
25. H. Fukunishi, O. Watanabe, S. Takada. J. Chem. Phys. **116**, 9058-9067 (2002).
26. G. A. Huber, S. Kim. Biophys J. **70**, 97-110 (1996)

NNs	Simulation end time							
	20ns	40ns	60ns	80ns	100ns	120ns	140ns	160ns
NN1	H(51%)	H(64%)	H(69%)	H(88%)	H(63%)	H(94%)	H(84%)	H(69%)
NN2	H(54%)	H(63%)	H(80%)	H(86%)	H(94%)	H(62%)	H(63%)	L(58%)
NN3	L(64%)	H(75%)	H(99%)	H(87%)	H(70%)	H(69%)	H(78%)	H(78%)
NN4	L(61%)	H(59%)	H(86%)	H(73%)	H(82%)	H(56%)	H(61%)	H(98%)
NN5	L(64%)	H(74%)	H(73%)	L(61%)	H(84%)	H(64%)	H(89%)	H(85%)
NN6	H(53%)	H(77%)	H(70%)	H(94%)	H(84%)	H(58%)	H(63%)	H(84%)
NN7	H(51%)	L(62%)	H(54%)	H(84%)	H(97%)	H(94%)	H(60%)	H(77%)
NN8	L(65%)	H(85%)	H(84%)	H(60%)	H(83%)	H(86%)	H(84%)	H(61%)
NN9	H(51%)	H(80%)	H(70%)	H(60%)	H(81%)	H(57%)	L(60%)	H(65%)
NN10	H(69%)	L(74%)	H(89%)	H(93%)	H(64%)	H(99%)	H(80%)	H(69%)
FP	H	H	H	H	H	H	H	H
PPC	60%	80%	100%	90%	100%	100%	90%	90%

Table 3 CNN predictions for Q498Y: This is the table listing the predictions from 10 neural networks for mutation Q498Y, which has higher affinity than the unmutated SARS-CoV-2 RBD.

NNs	Simulation end time							
	20ns	40ns	60ns	80ns	100ns	120ns	140ns	160ns
NN1	H(100%)	L(59%)	H(99%)	H(67%)	H(71%)	H(96%)	H(86%)	H(97%)
NN2	H(78%)	L(64%)	H(98%)	H(99%)	H(100%)	H(82%)	H(99%)	H(83%)
NN3	H(74%)	H(63%)	H(84%)	H(69%)	H(70%)	H(98%)	H(99%)	H(100%)
NN4	H(87%)	L(67%)	H(97%)	H(97%)	H(99%)	H(75%)	H(86%)	H(100%)
NN5	H(60%)	L(67%)	H(51%)	H(78%)	H(95%)	H(87%)	H(99%)	H(97%)
NN6	H(85%)	H(99%)	H(95%)	H(96%)	H(68%)	H(99%)	H(99%)	H(98%)
NN7	H(99%)	L(57%)	H(70%)	H(97%)	H(99%)	H(91%)	H(99%)	H(99%)
NN8	H(79%)	H(82%)	H(58%)	H(72%)	H(97%)	H(99%)	H(100%)	H(92%)
NN9	H(90%)	H(85%)	L(53%)	H(75%)	H(95%)	H(56%)	H(80%)	H(94%)
NN10	H(84%)	H(98%)	L(58%)	H(95%)	H(76%)	H(100%)	H(97%)	H(82%)
FP	H	H*	H	H	H	H	H	H
PPC	100%	50%	80%	100%	100%	100%	100%	100%

Table 4 CNN predictions for Q493Y: This is the table listing the predictions from 10 neural networks for mutation Q493Y, which has higher affinity than the unmutated SARS-CoV-2 RBD. *The number of H is the same as the number of L. The final prediction is based on comparing the averaged P_1 (=0.854) for H and averaged P_2 ($= 0.628$) for L and choosing the larger one.

NNs	Simulation end time							
	20ns	40ns	60ns	80ns	100ns	120ns	140ns	160ns
NN1	H(75%)	H(96%)	H(98%)	H(98%)	H(98%)	H(99%)	H(100%)	H(100%)
NN2	H(88%)	H(97%)	H(99%)	H(99%)	H(100%)	H(99%)	H(100%)	H(99%)
NN3	H(75%)	H(96%)	H(100%)	H(100%)	H(97%)	H(99%)	H(99%)	H(100%)
NN4	H(82%)	H(95%)	H(97%)	H(99%)	H(98%)	H(99%)	H(99%)	H(100%)
NN5	H(73%)	H(94%)	H(88%)	H(95%)	H(99%)	H(98%)	H(99%)	H(99%)
NN6	H(82%)	H(94%)	H(97%)	H(99%)	H(99%)	H(89%)	H(100%)	H(100%)
NN7	H(83%)	H(91%)	H(85%)	H(99%)	H(100%)	H(100%)	H(100%)	H(100%)
NN8	H(73%)	H(90%)	H(99%)	H(96%)	H(100%)	H(100%)	H(100%)	H(96%)
NN9	H(79%)	H(96%)	H(87%)	H(95%)	H(98%)	H(95%)	H(98%)	H(88%)
NN10	H(74%)	H(58%)	H(97%)	H(98%)	H(95%)	H(100%)	H(99%)	H(97%)
FP	H	H	H	H	H	H	H	H
PPC	100%	100%	100%	100%	100%	100%	100%	100%

Table 5 CNN predictions for E484K: This is the table listing the predictions from 10 neural networks for mutation E484K, which has higher affinity than the unmutated SARS-CoV-2 RBD.

NNs	Simulation end time							
	20ns	40ns	60ns	80ns	100ns	120ns	140ns	160ns
NN1	L(100%)	L(97%)	L(100%)	L(80%)	L(95%)	L(100%)	L(99%)	L(95%)
NN2	L(100%)	L(84%)	L(97%)	L(100%)	L(99%)	L(100%)	L(99%)	L(100%)
NN3	L(100%)	L(97%)	L(100%)	L(94%)	L(100%)	L(100%)	L(100%)	L(100%)
NN4	L(100%)	L(100%)	L(100%)	L(100%)	L(99%)	L(100%)	L(100%)	L(100%)
NN5	L(100%)	L(100%)	L(100%)	L(99%)	L(98%)	L(100%)	L(99%)	L(99%)
NN6	L(100%)	L(100%)	L(100%)	L(100%)	L(100%)	L(100%)	L(100%)	L(100%)
NN7	L(100%)	L(98%)	L(100%)	L(99%)	L(85%)	L(100%)	L(92%)	L(100%)
NN8	L(100%)	L(100%)	L(100%)	L(100%)	L(99%)	L(100%)	L(100%)	L(100%)
NN9	L(100%)	L(100%)	L(100%)	L(100%)	L(100%)	L(99%)	L(100%)	L(100%)
NN10	L(100%)	L(100%)	L(100%)	L(99%)	L(100%)	L(100%)	L(100%)	L(100%)
FP	L	L	L	L	L	L	L	L
PPC	100%	100%	100%	100%	100%	100%	100%	100%

Table 6 CNN predictions for Q493N: This is the table listing the predictions from 10 neural networks for mutation Q493N, which has lower affinity than the unmutated SARS-CoV-2 RBD.

NNs	Simulation end time							
	20ns	40ns	60ns	80ns	100ns	120ns	140ns	160ns
NN1	H(68%)	L(85%)	L(72%)	L(77%)	H(60%)	L(76%)	H(54%)	L(75%)
NN2	H(56%)	L(69%)	L(70%)	L(81%)	H(56%)	L(66%)	L(68%)	H(61%)
NN3	H(78%)	L(78%)	L(56%)	L(65%)	L(87%)	L(84%)	L(77%)	L(66%)
NN4	H(89%)	L(59%)	L(60%)	L(67%)	L(68%)	L(63%)	L(71%)	L(62%)
NN5	L(66%)	L(64%)	L(57%)	H(88%)	L(77%)	L(76%)	L(70%)	L(53%)
NN6	H(90%)	L(84%)	L(70%)	L(66%)	L(71%)	L(93%)	H(63%)	L(55%)
NN7	H(90%)	L(82%)	L(77%)	L(73%)	H(71%)	L(56%)	L(56%)	L(57%)
NN8	H(80%)	L(57%)	L(77%)	L(83%)	L(58%)	L(72%)	H(54%)	L(68%)
NN9	H(78%)	H(64%)	L(74%)	L(83%)	L(57%)	H(69%)	L(80%)	L(64%)
NN10	H(80%)	L(58%)	L(74%)	L(87%)	L(76%)	L(55%)	L(67%)	L(70%)
FP	H	L	L	L	L	L	L	L
PPC	10%	90%	100%	90%	70%	90%	70%	90%

Table 7 CNN predictions for Q498K: This is the table listing the predictions from 10 neural networks for mutation Q498K, which has lower affinity than the unmutated SARS-CoV-2 RBD.

NNs	Simulation end time							
	20ns	40ns	60ns	80ns	100ns	120ns	140ns	160ns
NN1	L(98%)	L(100%)	L(98%)	L(85%)	L(91%)	L(90%)	L(76%)	L(91%)
NN2	L(100%)	L(94%)	L(94%)	L(88%)	L(87%)	L(86%)	L(78%)	L(86%)
NN3	L(100%)	L(98%)	L(90%)	L(87%)	L(95%)	L(91%)	L(71%)	L(85%)
NN4	L(94%)	L(100%)	L(96%)	L(93%)	L(91%)	L(88%)	L(83%)	L(72%)
NN5	L(99%)	L(99%)	L(90%)	L(95%)	L(92%)	L(96%)	L(80%)	L(87%)
NN6	L(91%)	L(92%)	L(97%)	L(91%)	L(94%)	L(95%)	L(69%)	L(65%)
NN7	L(98%)	L(100%)	L(96%)	L(94%)	L(85%)	L(86%)	L(62%)	L(83%)
NN8	L(97%)	L(99%)	L(87%)	L(94%)	L(91%)	L(93%)	L(79%)	L(69%)
NN9	L(91%)	HL(97%)	L(95%)	L(89%)	L(93%)	L(94%)	L(89%)	L(96%)
NN10	L(94%)	L(100%)	L(88%)	L(88%)	L(95%)	L(88%)	L(66%)	L(81%)
FP	L	L	L	L	L	L	L	L
PPC	100%	100%	100%	100%	100%	100%	90%	100%

Table 8 CNN predictions for G502P: This is the table listing the predictions from 10 neural networks for mutation G502P, which has lower affinity than the unmutated SARS-CoV-2 RBD.

Accuracy	Simulation end time							
	20ns	40ns	60ns	80ns	100ns	120ns	140ns	160ns
NN1	5/6	5/6	6/6	6/6	5/6	6/6	5/6	6/6
NN2	5/6	5/6	6/6	6/6	5/6	6/6	6/6	4/6
NN3	4/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6
NN4	4/6	5/6	6/6	6/6	6/6	6/6	6/6	6/6
NN5	5/6	5/6	6/6	4/6	6/6	6/6	6/6	6/6
NN6	5/6	6/6	6/6	6/6	6/6	6/6	5/6	6/6
NN7	5/6	4/6	6/6	6/6	5/6	6/6	5/6	6/6
NN8	4/6	6/6	6/6	6/6	6/6	6/6	5/6	6/6
NN9	5/6	5/6	5/6	6/6	6/6	5/6	5/6	6/6
NN10	5/6	5/6	5/6	6/6	6/6	6/6	6/6	6/6
Averaged NN prediction	0.7833	0.8667	0.9667	0.9667	0.95	0.9833	0.9167	0.9667
Ensemble Prediction	5/6	1	1	1	1	1	1	1

Table 9 This is the table listing the accuracy of the 10 independently trained neural network per simulation duration, i.e., we have trained 80 CNNs in total. This table also list the average CNN accuracy and accuracy of ensemble predictions.

27. D. M. Zuckerman, L. T. Chong. Annu Rev Biophys. **46**, 43-57 (2017)
28. H. Kokubo, T. Tanaka, Y. Okamoto. J. Comput. Chem. **13**, 2810-2821 (2011)
29. J. Curuksu, J. Sponer, M. Zacharias. Biophys J. **7**, 2004-2013 (2009)
30. O. V. Prezhdo. J. Phys. Chem. Lett. **11**, 9656-9658 (2020)
31. F. Noé, Lecture Notes in Physics, **968**, 331–372 (2020). https://doi.org/10.1007/978-3-030-40245-7_16
32. L. Casalino, *et al*, BioRxiv (2020). https://doi.org/10.1101/2020.11.19.390187
33. A. S. Kelkar, B. C. Dallin, R. C. Van Lehn. J. Phys. Chem. b. **124**, 9103-9114 (2020)
34. F. Zhu, X. Zhang, J.E. Allen, D. Jones, and F.C. Lightstone, Journal of Chemical Information and Modeling, **60**(6), 2766–2772 (2020). https://doi.org/10.1021/acs.jcim.0c00026
35. A. Trabelsi, M. Chaabane, and A Ben-Hur, Bioinformatics, **35**(14), i269–i277 (2019). https://doi.org/10.1093/bioinformatics/btz339
36. J. Shang, *et al.* Nature, **581**, 221-226 (2020).
37. W. Humphrey, A. Dalke, K. Schulten, J. Mol. Graph., **14**, 33-38 (1996).

38. J.C. Phillips, *et al.* J. Comput. Chem., **26**, 1781–1802 (2005).
39. J.C. Phillips, *et al.* J. Chem. Phys., **153**, 044130 (2005).
40. J. Huang, *et al*, Nat. Methods, **14**, 71-73 (2017).
41. W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein. J. Chem. Phys, **79**, 926 (1983).
42. S. E. Feller, Y. Zhang, R. W. Pastor, and B. R. Brooks, J. Chem. Phys, **103**, 4613–4621 (1993).
43. G. J. Martyna, D. J. Tobias, and M. L. Klein, J. Chem. Phys., **101**, 4177–4189 (1994).
44. T. Darden, D. York, and L. Pedersen. J. Chem. Phys., **98**, 10089–10092 (1993).
45. U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen, J. Chem. Phys., **103**, 8577–8593 (1995).
46. H. Grubmüller, H. Heller, A. Windemuth, and K. Schulten. Mol. Simul., **6**, 121–142 (1991).
47. J.-P.Ryckaert, G. Ciccotti, and H. J. C. Berendsen, J. Comput. Phys., **23**, 327–341 (1977).
48. R. J. Gowers, *et al.* In S. Benthall and S. Rostrup, editors, *Proceedings of the 15th Python in Science Conference*,98-105 (2016).
49. N. Michaud-Agrawal, E. J. Denning, T. B. Woolf, and O. Beckstein. J. Comput. Chem.,**32**, 2319-2327 (2011).
50. S. Wang, K. Fan, N. Luo, Y. Cao, F. Wu, C. Zhang, K. Heller and L. You, Nature communications,**10**(1), 1-9 (2019).
51. M. Wang, Z. Cang, and G.-W. Wei, Nature Machine Intelligence, **2**(2), 116–123 (2020). https://doi.org/10.1038/s42256-020-0149-6

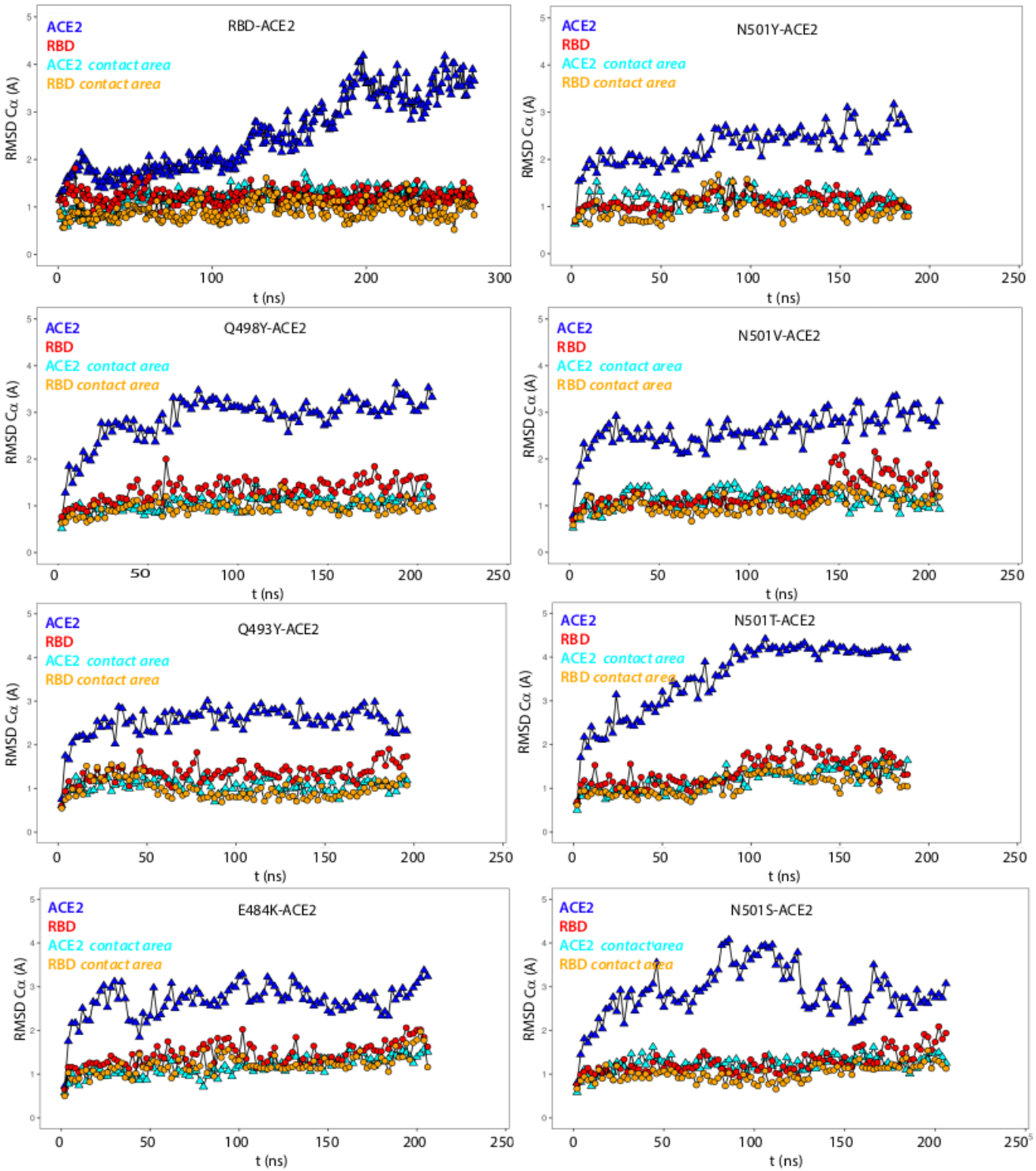

Fig. 6 RMSD calculated for the C_α carbon atoms of the RBD and ACE2 and for the contact area for the simulations reported in the panels.

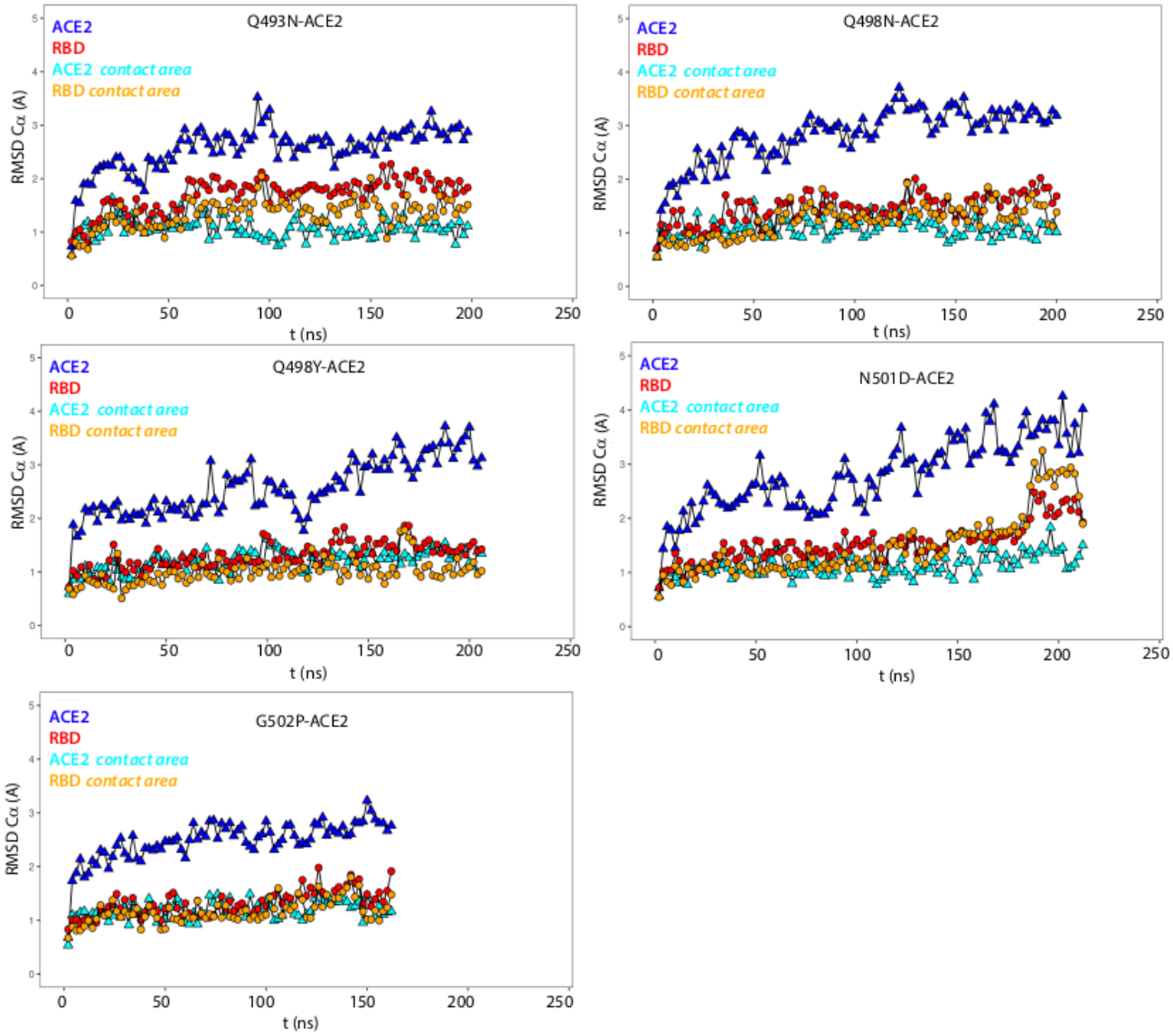

Fig. 7 RMSD calculated for the C_α carbon atoms of the RBD and ACE2 and for the contact area for the simulations reported in the panels.