Supplementary Materials for

# Machine Learning-driven Multiscale Modeling: bridging the scales with a next generation simulation infrastructure

Helgi I. Ingólfsson[1], Harsh Bhatia[2], Fikret Aydin[1], Tomas Oppelstrup[1], Cesar A López[3], Liam G. Stanton[4], Timothy S. Carpenter[1], Sergio Wong[1], Francesco Di Natale[2], Xiaohua Zhang[1], Joseph Y. Moon[2], Christopher B. Stanley[5], Joseph R. Chavez[2], Kien Nguyen[3], Gautham Dharuman[1], Violetta Burns[3], Rebika Shrestha[6], Debanjan Goswami[6], Gulcin Gulten[6], Que N. Van[6], Arvind Ramanathan[7], Brian Van Essen[2], Nicolas W. Hengartner[3], Andrew G. Stephen[6], Thomas Turbyville[6], Peer-Timo Bremer[2], S. Gnanakaran[3], James N. Glosli[1], Felice C. Lightstone[1], Dwight V. Nissley[6,*], Frederick H. Streitz[1,*]

[1]Physical and Life Sciences (PLS) Directorate, Lawrence Livermore National Laboratory, Livermore, CA 94550. [2]Computing Directorate, Lawrence Livermore National Laboratory, Livermore, CA 94550. [3]Theoretical Biology and Biophysics Group, Los Alamos National Laboratory, Los Alamos, NM 87545. [4]Department of Mathematics and Statistics, San José State University, San José, CA 95192. [5]Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37830. [6]RAS Initiative, The Cancer Research Technology Program, Frederick National Laboratory, Frederick, MD 21701. [7]Computing, Environment & Life Sciences (CELS) Directorate, Argonne National Laboratory, Lemont, IL 60439.

[*]Corresponding authors: Dwight V. Nissley, nissleyd@mail.nih.gov, and Frederick H. Streitz, streitz1@llnl.gov

# S1 Supplementary Methods

## S1.1 Continuum macro model parameterization

The continuum macro model was updated and extended to support both RAS (KRAS4b) and RAS-RBDCRD (KRAS4b bound to RBD and CRD domains from RAF1) proteins. The parameterization was done largely along the same lines as the earlier RAS-only macro model described in.[1] This section describes the updated parameterization: starting with estimates of the relative RAS and RAF concentration, short descriptions of the different CG reference simulations using in the parameterization, how the protein states and rates were derived, as well as the protein-lipid interactions.

A variety of western blot and proteomic analyses have been used to determine the amount of RAS and RAF proteins in cell lines and tissue, with concentrations to ranging from 0.005 – 1.3 μM.[2-7] Depending on the cell line and method used the ratio of RAS:RAF ranged from 1:5 to 180:1. These results suggest that in most cell lines analyzed there is a higher concentration of RAS compared with RAF. We chose a concentration of 150 RAF molecules per $\mu m^2$ to allow us to have sufficient RAF particles in the simulation to allow significant sampling but keeping the level to 50% of that of RAS (300 RAS molecules per $\mu m^2$). The kinetics of the binding between RAS and the RBD domain of RAF has been measured with an association rate of $4.5 \times 10^7$ $M^{-1}$ $s^{-1}$ and a dissociation rate of 7.4 $s^{-1}$.[8] In a single molecule co-immunoprecipitation analysis, the binding rate of RBDCRD domain of RAF1 with RAS was measured as 0.28 $s^{-1}$ with a dissociation rate of 2.5 $s^{-1}$.[9] These affinities were used to inform the RAS-RBDCRD rates discussed below.

The main set of parameterization simulations, as mentioned in the main text, were Martini coarse-grained simulation of RAS and RAS-RBDCRD on an 8-lipid type plasma membrane mimic[10] sampled from a previous $\mu m^2$ macro model simulation.[1] The details of these two sets of simulations are described in Nguyen et al.[11] and here we give a short summary. 732 simulations were run for RAS-only and 970 for RAS-RBDCRD. All the simulations started from initial patches, 30 nm x 30 nm lipid membranes with a RAS in the middle cut out from a larger $\mu m^2$ macro model simulation.[1] These were selected randomly within five groups along the low, average, high RAS-count latent coordinate.[1] For each simulation: diverse initial protein structures were selected, the membrane initialized based the accompanying macro model patch, the protein pulled to the membrane, initial equilibrium run followed with 10 μs of projection simulation. Frames were saved every 2 ns but most of the analysis was done at a 10 ns frame rate. For lipid macro model parameterization of both lipid-lipid RDFs and lipid diffusions (see below) a separate set of lipids only simulations were run. These simulations were selected, set up, and run exactly like the RAS and RAS-RBDCRD simulations (see[11] and above) except the proteins were omitted. 50 simulations were selected, 10 in each of the 5 lipid groups, and each simulated for 10 μs.

The RAS and RAS-RBDCRD are described in the macro model as one or two beads, respectively. Each protein/bead is in a specific ‘state’, with unique properties, representing a protein confirmational state with respect to the membrane. In order to define the protein bead ‘states’ for the macro model, the population in RAS tilt-rotation space of configurations near the end of the main parameterization simulations were analyzed. For the RAS-only system, three clear basins were found, and two major and one minor basin found for the RAS-RBDCRD system (see Fig. 2 in main paper). For the RAS-RBDCRD system the minor basin in rotation space combined with a threshold on the CRD-membrane distance were used to define a third elevated ‘Z’ state called za. The state definitions are shown in Fig. 2 of the main text, and each point in tilt-rotation space was assigned to the state (basin) to which there is a strictly uphill path in the population histogram. The RAS-RBDCRD za-state was defined by the RAS rotation in the (140º, 315º) interval and the CRD-membrane distance > 4.8 nm. The population in the different states were as follows:

| **RAS-only system** | **RAS-RBDCRD system** |
|---|---|
| state α: 21.5% | state ma: 52.9% |
| state β: 65.4% | state mb: 46.6% |
| state β': 13.1% | state za: 0.5% (high Z-state) |

To get an indication that these populations are also close to what the equilibrium populations would be, we defined a two state Markov model for the RAS-RBDCRD system (here ignoring the relatively low population and short-lived za-state) and fit this model to the time evolution of the relative populations in states ma and mb. The simulation data and model evolution are shown in Fig. S1. Solving for the equilibrium of the so-defined model yields 56.3% in state ma and 43.7% in state mb, which is close to what was observed at the end, the 9-10 μs interval, of the simulations from[11] and listed in the above table.

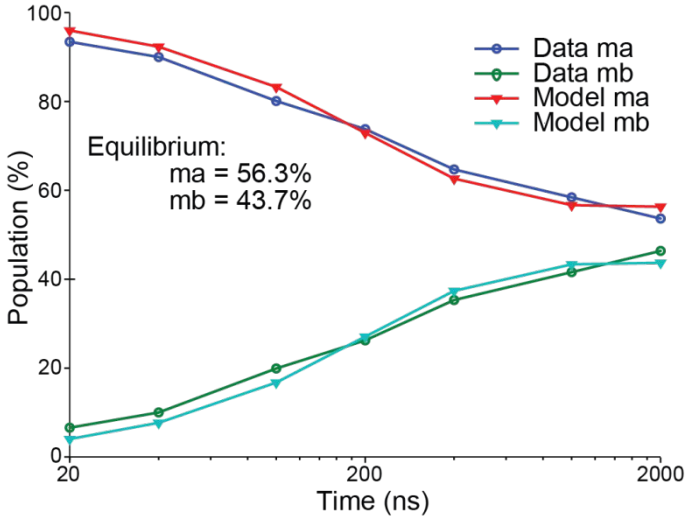



*Fig. S1 Two-state Markov model for population evolution between the two major states in the RAS-RBDCRD system. The rate unit is 1/μs. Blue and green show the time evolution between the two major population basins in the main parametrization simulations. Red and cyan show the corresponding model evolution. We can see a quite good agreement between this simple model and the real simulation data.*

In order to quantify the rate of transitions among basins, we utilized a set of much longer simulation, done on a similar lipids mixture. A single RAS-RBDCRD complex was built on a two-lipid mixture and deposited on its surface. The lipid membrane was composed of a 70:30 POPC:POPS mixture, as described in Travers et al.[12] The Protein complex was modeled following the same protocol as previously published.[12] Ten initial replicates were started in order to populate the most equilibrated configurations of the RAS-RAF complex bound to the anionic membrane. This first round of simulations was run for 100 μs each, until convergency of equilibration was observed. Then, 500 representative configurations were extracted from each basin (three basins = 1500 sims) and parallel simulations were continued. Each trajectory was run for 30 μs and frames saved every 30 ns used for analysis. To further validate the relative population sizes of the different protein states Hidden Markov Model analysis of these longer simulations yields roughly equal equilibrium populations in the two main basins for a wide range of lag times, see Fig. S2.

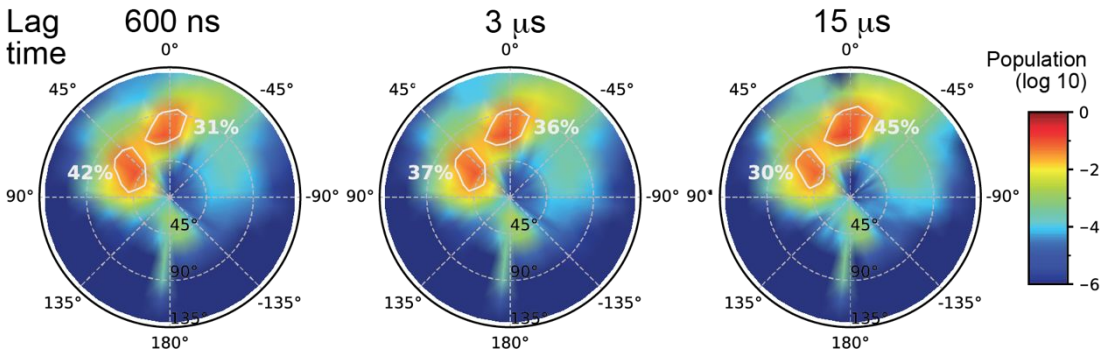



*Fig. S2 Equilibrium populations from HMM analysis of longer two-lipid mixture simulations. For a range of lag times, the relative populations of the two main basins remain of similar magnitude. The lag times shown, from left to right, are 20 frames (600 ns), 100 frames (3 μs), and 500 frames (15 μs).*

The orientation of the complex was measured using a set of reaction coordinates, the 'tilt' and 'rotation' angles. The tilt angle is defined as the tilt of the RAS G-domain away from the bilayer normal and the azimuthal angle (the rotation) is the angle at which that tilt occurs, as previously described.[1, 12, 13] These tilt and rotation values were projected into polar coordinates, see Fig. 2 in main text. For RAS-RBDCRD the tilt and rotation were calculated in the same manner as well as the depth of CRD penetration into the membrane, which was calculated using the center of mass (COM) distance of the residues belonging to the hydrophobic loop of CRD, residues THR145-LYS148 and PHE158-ASN161, with respect to the center of mass of the membrane COM. Only the Z component of the distance is taken into account.

The transition rates for the macro model were determined through a continuous-time Markov chain (CTMC) model. Estimates based on MD simulations were used to constrain the parameter space of the CTMC calculations, and the rates were constructed to recover both the equilibrium state populations from the MD as well as the experimentally observed residency time distributions of RBDCRD in HeLa cells (with endogenous KRAS4b). The values of these rates along with their associated state pathways can be found in Fig. 2C in the main document.

For the lipids the 8-lipid type plasma membrane mimic was used as described in[1, 10] except the PIP2 (Martini lipid name PAP6) lipid charge was changed from -5 to -4 as described in.[11] Due to the changes in PAP6 lipid parameters, we updated the lipid-lipid RDFs using the 50 lipid-only control simulations described above. RDFs were calculated from the same Martini beads as previously used[1] (T1A, C1A, D1A, R1) and in the macro model the same scaling for the PIP2 interactions as described in[1] was used. For all the lipids the same lipids diffusion values were also used as determined in.[1]

For the proteins the macro model parameters were calculated from the reference simulation described above. The lipid-RAS RDFs were calculated using the same reference beads as previously described[1] (F1 bead of farnesyl and T1A/C1A/D1A/R1 beads of lipids). The reference point for the CRD RAF bead for the lipid-CRD RDFs was the COM of the CRD loops (as defined above in the tilt/rotation/depth section above). Lateral diffusion for the RAS-only and RAS-RBDCRD complex were computed using the *gmx msd* tool as provided by GROMACS. First, periodic conditions of the simulations were reconstructed allowing a full displacement of the proteins on the surface of the membrane. This was achieved by using the *pbc nojump* option in GROMACS. Then the COM displacement of the membrane was removed and the lateral mean square displacement (MSD) of the proteins was measured. Means were taken for trajectories corresponding to all the defined states. Distances between RAS and CRD RAF beads in the macro model are state specific and set to 0, 3.2 and 3.3 nm for the za, ma, and mb states, respectively. In the za state the RAF-CRD is not attached to the membrane and the CRD bead is located on top of the RAS bead. For ma and mb states the average distance between RAS farnesyl bead F1 to the COM of RAF-CRD loops residues THR145-LYS148 and PHE158-ASN161 were estimated from Martini CG

simulations to determine the bead separation in the continuum macro model. The RAS and RAF (CRD bead) RDF's for the ma and mb states include interference from RAF and RAS, respectively. While not done for the present campaign, deconvolving this interference could improve the protein-lipid potentials for these states.

In the current macro model, we wanted to focus on the induced effects of lipids. Protein-protein interactions are complex and specific to the relative protein-protein confirmation, and the macro model as a large-scale model is not expected to have a good opinion of what happens with close protein-protein interactions as they are only each represented by a rotationally averaged point particle (bead). For this reason, we define the protein-protein interactions to be a soft-core repulsion to prevent overlap, so that for the most part the protein-protein interactions are implicitly defined through the lipids only. The form of the soft-core repulsion was:

$$U(r) = \left(\frac{\sigma}{r - r_c}\right)^{12} - a - b(r - r_c),$$

with $\sigma = 1.375\ nm$, $r_c = 7\ nm$, and a and b chosen so that $U(r_c) = U'(r_c) = 0$.

The model created as described in the previous sections results in too high a concentration of PIP2 around the proteins, particularly around the RAS-RBDCRD complex, more than the input RDF's show. In addition, strong aggregation of the proteins occurs. In order to adjust the number of PIP2 around the proteins we introduce a charge screening function that scales the lipid-protein interaction as a function of lipid charge in the neighborhood of each protein. We define first the charge $Q$ around a protein at position $\boldsymbol{r}$ by:

$$Q(\boldsymbol{r}) = \sum_{j=1}^{n_{lipids}} \int_{\boldsymbol{r}' \in \mathbb{R}^2} q_j \rho_j(\boldsymbol{r} - \boldsymbol{r}') \frac{e^{-\kappa|\boldsymbol{r} - \boldsymbol{r}'|}}{|\boldsymbol{r} - \boldsymbol{r}'|} d^2\boldsymbol{r}'.$$

Next, given the protein-lipid potential $\phi(r)$ we define the scaled protein-interaction $U$ as:

$$U(\boldsymbol{r}) = \sum_{j=1}^{n_{lipids}} \int_{\boldsymbol{r}' \in \mathbb{R}^2} \rho_j(\boldsymbol{r} - \boldsymbol{r}') \phi_j(|\boldsymbol{r} - \boldsymbol{r}'|) S\left(\frac{Q(r)}{Q_0}\right) d^2\boldsymbol{r}'.$$

With the scaling function $S(q) = \frac{1}{1 + |q|^{\alpha}}$. $Q_0$ is a constant related to the preferred charge around a given protein as determined by the equilibrium lipid density around a protein. $\alpha$ and $\kappa$ are fitting parameters. We used the same $\alpha$ and $\kappa$ for all protein types. We set $q_{PIPs} = -4$, $q_{PAPS} = -1$, and all other lipid charges to zero. In our simulations we used $\alpha = 20$, $\kappa = 1$, and $Q_0 = 14$. Fig. S3 shows the effect of this charge screening on the lipid profiles around different protein beads in the macro model.

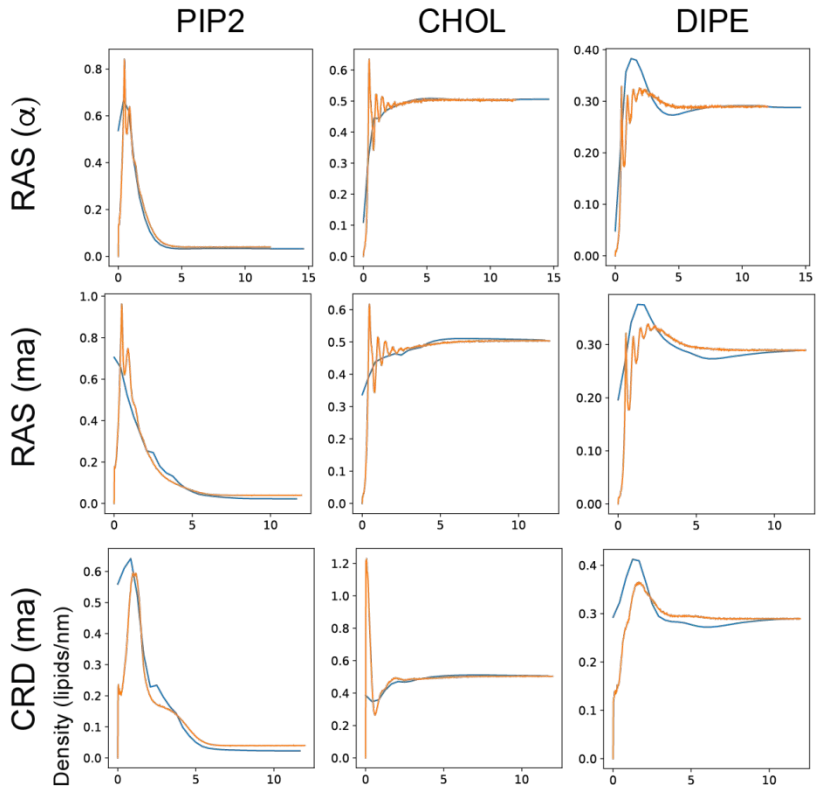



*Fig. S3 Lipid profiles around protein beads in CG and the macro model for PIP2 (left), CHOL (middle), and DIPE (right). CG density profile shown in yellow, and macro model profile in blue. Top row: Lipids around RAS in the α state. Middle row: Lipids around RAS for the RAS-RBDCRD complex in the ma state. Bottom row Lipids around RAF-CRD for the RAS-RBDCRD complex in the ma state.*

## S1.2 Protein structure input libraries

Libraries of initial structure in each state for both RAS-only and RAS-RBDCRD were sampled from the pre-campaign single protein CG simulation ensample described in Nguyen et al.[11] This pre-campaign simulation ensample was itself set up from diverse sets of protein structures and lipid configurations and consisted of 732 RAS-only (KRAS4b) and 970 RAS-RBDCRD (KRAS4b with RBD and CRD domains from RAF1) simulations with an aggregated simulation time over 21 ms. For all the simulations each frame was assigned to a given state based on tilt/rotation (RAS-only) or tilt/rotation/depth (RAS-RBDCRD), see section S1.1, and for each protein and each state protein structures were saved from 5000 random frames.

## S1.3 *Createsims*: Mapping Continuum-to-CG

The MuMMI *Createsims* module creates a CG Martini simulation form a given macro model patch in an automatic and fault tolerant manner. The initial *Createsims* module was described in Ingólfsson et al.,[1] here we give a short summary and detail the changes made. The *Createsims* module runs on CPUs only for MuMMI to utilize available CPU resources and not to compete with concurrently running MD production runs that use the GPUs. First, the selected macro model patch is parsed and converted from macro model resolution to CG Martini resolution. Macro model lipid densities are discretized to whole lipids and protein structures, from specified conformational sates, are sampled from input libraries and

palace as dictated by the macro model patch. Second, the CG simulation is built using a modified version of the *insane* membrane building tool.[14] Third, proteins are pulled from solution to the membrane and initial system minimization and equilibration is performed using the GROMACS MD package v2019.06.[15] Fourth and last, the resulting simulation is converted from GROMACS to ddcMD[16-18] format using the ddcMDconverter [19] and now ready for running with ddcMD on GPU resources.

The macro model represents each protein with one (RAS-only) or two (RAS-RBDCRD) beads. RAS-RBDCRD has one bead for RAS and one for the attached CRDRBD of RAF. Additionally, each protein can exist in one of three possible confirmational states. Protein initial structures are randomly sampled from input libraries as defined in Section S1.2. For the placement of proteins, as before, the macro model RAS bead dictated the farnesyl position of RAS in the bilayer x,y plane. RAS-only proteins are randomly rotated around the z-axes while for RAS-RBDCRD proteins the position of the second RBDCRD bead is used to rotate the selected RAS-RBDCRD accordingly along the z-axes. The angle placement applies to the two RAS-RBDCRD confirmational states (ma and mb) were the CRD is membrane bound, in the third (za) state the CRD bead is above the RAS bead and the structures z-axes rotation is randomized. As before, MDAnalysis[20, 21] is used to detect overlap between proteins, with protein CG bead <0.8 nm from other proteins. If an overlap is detected, new random rotations and initial structures are tried, currently up to 500 times before all proteins are spaced away from each other by 0.5, 1, 1.5 or 2 nm and the progress repeated. If no placement of all proteins can be found the setup of this patch is stopped and marked as failed. To guard against bias in the build procedure we updated the method to place all proteins at the same time, if any two proteins have overlap the placement of all proteins is repeated. The proteins, except for the RAS fanesyl's, are initially places in solution 2.25 nm above the membrane and later pulled to the membrane.

For the lipid placement, we want to place the lipids to match the number of lipids represented by the macro model patch density field, both in terms of overall composition and in spatial distribution. We use the following stochastic method to place discrete lipid molecules in close approximation of the macro model density field at individual lipid resolution instead to the 5×5 subgrid (64 lipids each) approximation we used before. We place the lipids in a 40x40 grid, to select one for each grid cell for a total of 1600 lipids in each of the inner and outer layers. The process first divides the density in each grid cell so that it total number of molecules (represented by the density in that grid cell) equals one. Then a grid point is selected at random, and a lipid type is selected according to the relative densities in that grid cell. The left-over density is distributed to all not yet sampled grid points by selecting a multiplier for each lipid type so that the total remaining amount of that lipid is the expected amount. The process is repeated until all grid points have been sampled. After this process is complete, some lipids are removed by selecting a set of locations at random to create the desired overall density and asymmetry between the outer/inner leaflets. The full lipid placement grid is passed to a modified version of the *insane* membrane building tool[14] which we use to construct each lipid and place water and ions.

After placement of all molecules the system is energy minimized, initial equilibrium is run, and proteins pulled to the membrane. The GROMACS MD package v2019.06[15] is used to run the same sequence of minimization and equilibrium steps as described before except now the pulling is done slightly slower (0.0005 nm/ps with 1,000 kJ $mol^{-1}$ $nm^{-2}$ force constant along the z-axes only) and an short (10000 steps) in-between equilibrium run with position restraints on protein beads (100 kJ $mol^{-1}$ $nm^{-2}$ restraints on protein beads in x and y dimensions) added to let the membrane better adapt to the proteins before the weaker (10 kJ $mol^{-1}$ $nm^{-2}$) run.

The Martini 2.2v CG force field parameters[14, 22-28] were used as described in[1] with updated protein parameters and PIP2 (specifically Martini lipid PAP6) lipid charge are described in[11] and using the same simulation parameters as described in[1, 11]; notably, following the new-rf Martini parameter set[29] with final simulations run at 20 fs time step, 1 bar semiisotropic pressure coupling and a temperature of 310 K, as

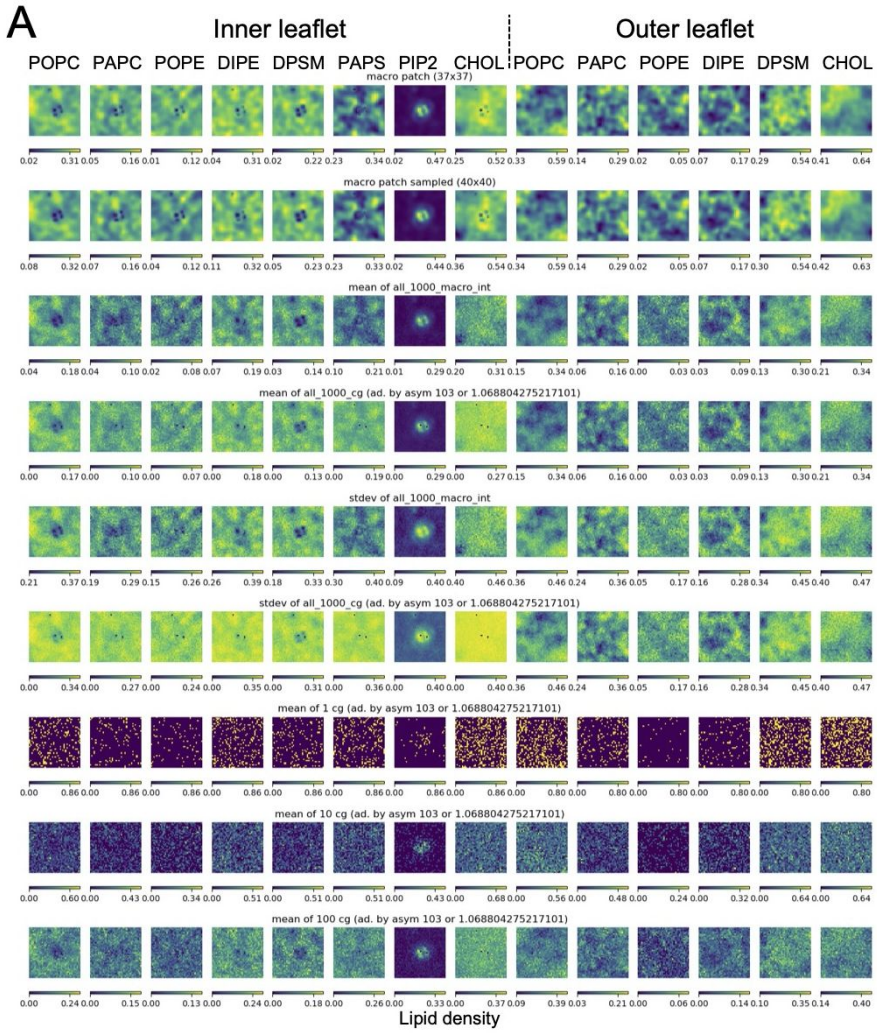
well as weak (2 kJ $mol^{-1}$ $nm^{-2}$) harmonic potentials applied to the outer leaflet POPC PO4 beads along the z-direction to keep limit large bilayer undulations.

After simulation set up and initial equilibrium the GROMACS inputs were converted to ddcMD format using the ddcMDconverter[19], ddcMD run files and select setup file are saved and the setup success or failure marked for the MuMMI workflow. Current configuration of *Createsims* is tuned to run on 24 CPU cores on *Lassen* and *Summit* and takes ~1.5 h to run for the ∽140,000 particles CG systems.

## S1.4 CG Simulations and Analysis

During a MuMMI multiscale simulation all CG simulations are converted from continuum macro model patches as described in section S1.3. After conversion the MuMMI workflow launches a *CG Simulation and Analysis* module for each CG simulation when GPU compute resources become available.[30] Each *CG Simulation and Analysis* module requires three CPUs and one GPU and manages the running and on-the-fly analysis of a single CG simulation. The module is Python based and runs and manages its files locally on the compute node it is launched on, with periodic updates of simulation files to the global filesystem adding frames to the simulation trajectory and more frequent passing of data for online feedback. The module runs the CG MD simulation using a GPU enabled version of ddcMD.[16] ddcMD runs the full MD loop on the GPU only utilizes CPU resources only for I/O, coordination and logging; delivering ~1 μs per day for the ∽140,000 particles CG systems simulated here.[16, 30] The simulation run conditions CG parameters are as explain in Ingolfsson et al.[1] and in section S1.3 with updates described in.[11, 31] The module monitors the running simulation restarting and stopping (currently set at 5 μs) as needed. ddcMD outputs simulation snapshots every 0.5 ns locally which are all analyzed as they come in using a parallel multithreads analysis routine with batched writes to global filesystem, passing data to macro model feedback as well as well as possible conversion to AA simulation, and saving simulation trajectories at a 2 ns interval. The on-the-fly analysis is an updated version of the analysis described in,[1] were each frame is parsed using MDAnalysis[20, 21] and can be analysis for a number of properties, including assignment of lipids to leaflet, degerming protein state and calculating protein-lipid RDFs. The analysis methods were updated to support different types of proteins, including RAS-RBDCRD.

# S2 Supplementary Figures



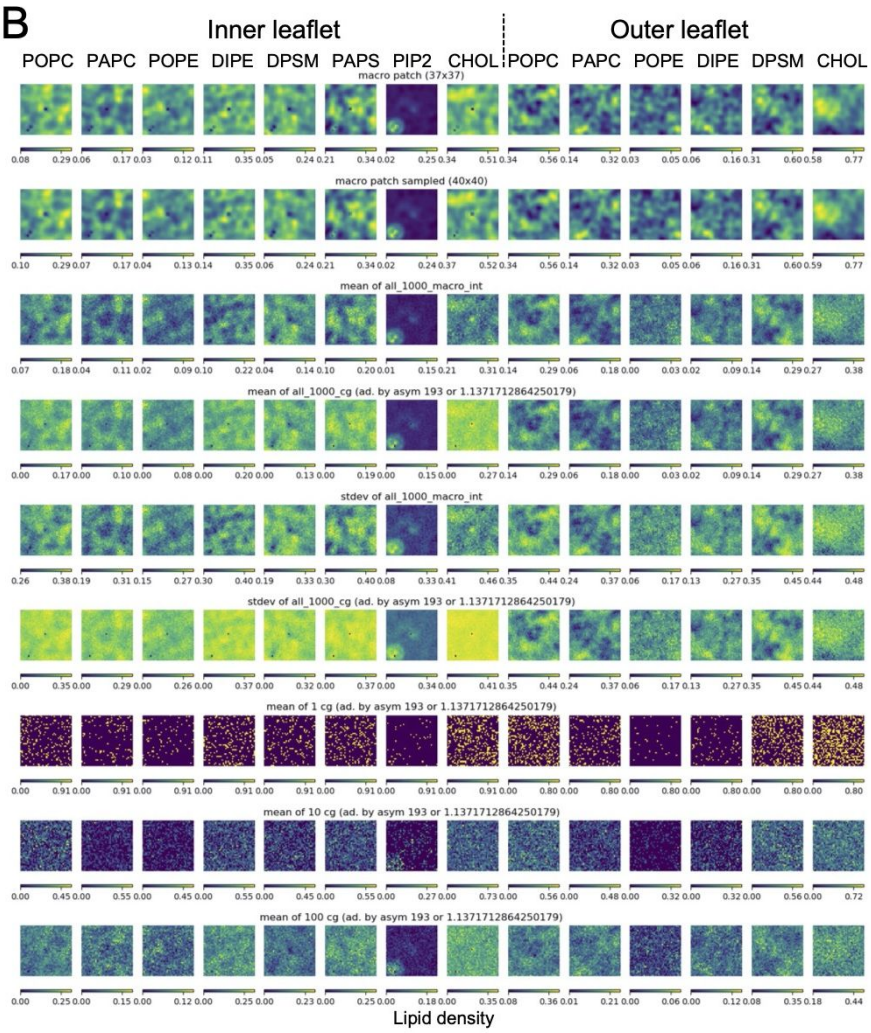



*Fig. S4 Converting continuum macro model patch lipid densities to individual lipids for CG simulation build. The lipid described are converted as described in section S1.3 above. Here we show a comparison of the macro lipids densities to a CG lipid instantiation and averaging of 10, 100, or 1000 instantiations. A) Shows a patch with two RAS-RBDCRD and one RAS-only and (B) a patch with one RAS-only and one RAS-RBDCRD, all lipid types (eight inner and six outer leaflet) are shown for both. Note continuum macro model densities are in lipids per $\mu m^2$ while the CG setup are counts and averaged counts per cell.*

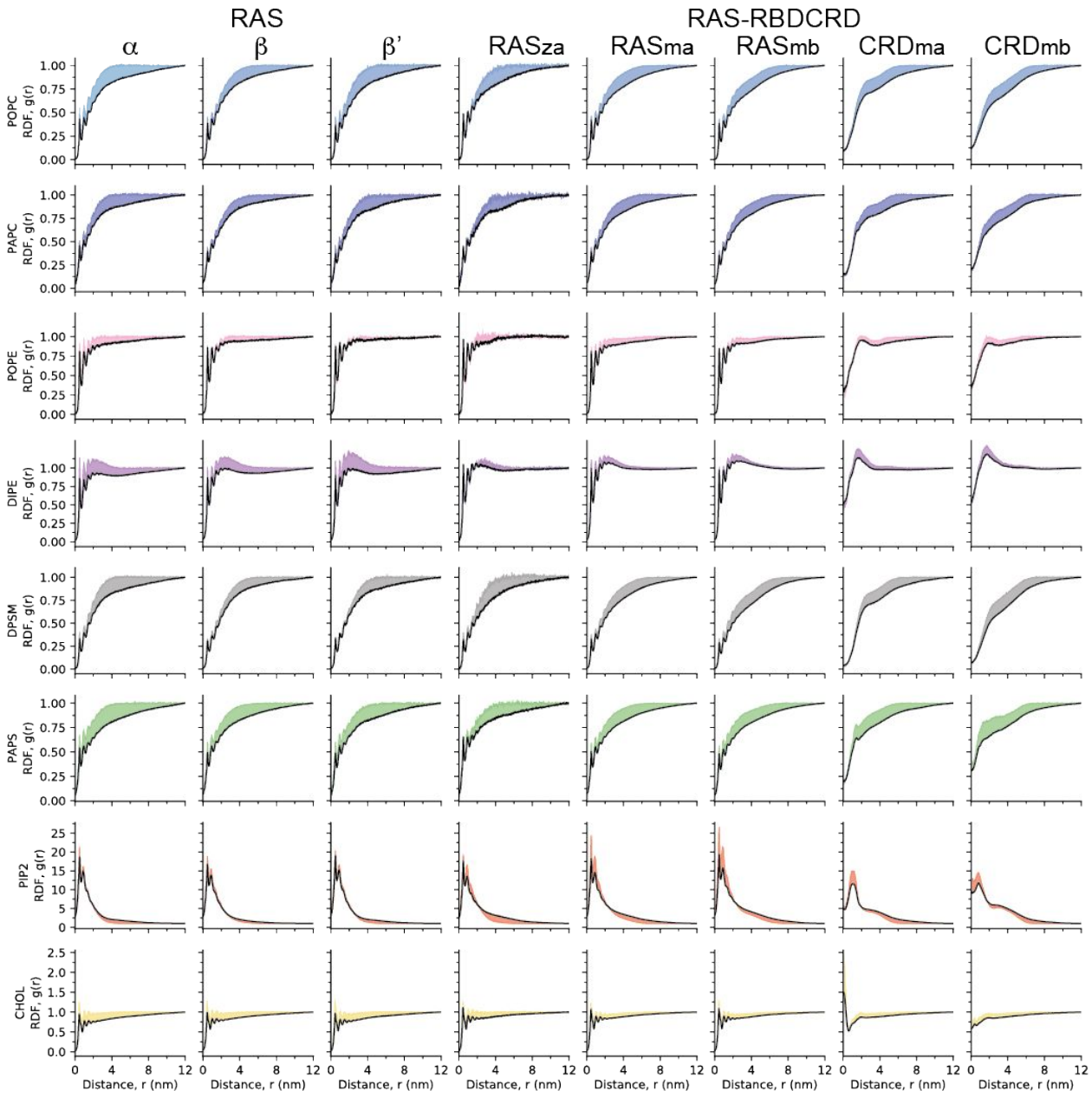



*Fig. S5 The effect of the on-the-fly feedback on protein-lipid radial distribution functions (RDFs) is shown by comparing the initial (colored) and final (black) RDFs. The shaded region highlights the difference achieved through the feedback. Same data as shown in main text Fig. 6 except here shown for all lipids and all conditions.*

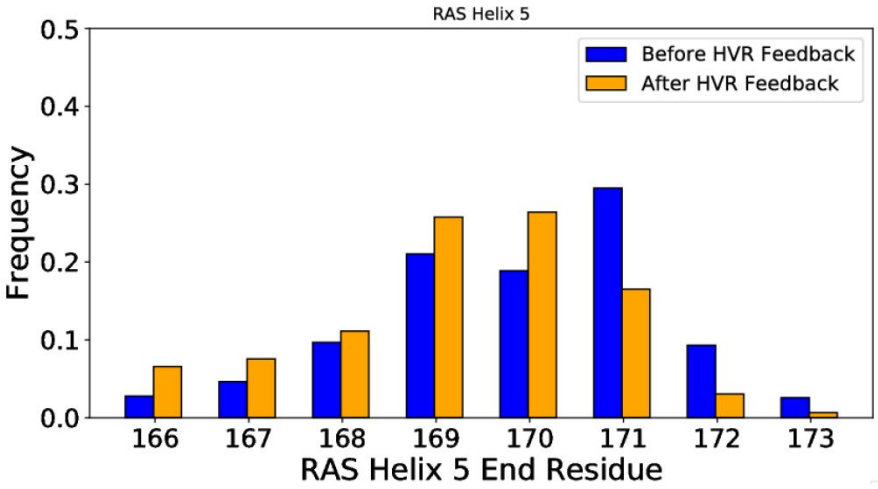



*Fig. S6 RAS helix 5 length distribution in AA simulations before (helix 5 ends at 172) and after (helix 5 ends at 171) HVR feedback. The last parts of the AA simulations (>40 ns) were analyzed to examine the length of RAS helix 5. Before HVR Feedback on average helix 5 ends at 169.8 (CG HVR Position: 172) with an average shift of 2.2 residues. After HVR Feedback on average helix 5 ends at 169.1 (CG HVR Position: 171) average shift of 1.9 residues.*

# S3 Supplementary Tables

| | | RAS proteins | | | | | |
|---|---|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 | 4 | 5 |
| RAS-RBDCRD complexes | 0 | 3598 | 3660 | 4507 | 1185 | 213 | 0 |
| | 1 | 7204 | 3398 | 3368 | 56 | 10 | 5 |
| | 2 | 1476 | 671 | 208 | 56 | 20 | 0 |
| | 3 | 796 | 808 | 235 | 56 | 0 | 0 |
| | 4 | 709 | 651 | 148 | 0 | 0 | 0 |
| | 5 | 662 | 354 | 3 | 0 | 0 | 0 |
| | 6 | 459 | 2 | 1 | 0 | 0 | 0 |

*Table S1: Distribution of CG systems by composition. The values along the left-most column are the number of RAS-RBDCRD complexes and the top-most row describing the number of RAS proteins.*

## S4 Supplementary References


(1) Ingolfsson, H. I.; Neale, C.; Carpenter, T. S.; Shrestha, R.; Lopez, C. A.; Tran, T. H.; Oppelstrup, T.; Bhatia, H.; Stanton, L. G.; Zhang, X.; Sundram, S.; Di Natale, F.; Agarwal, A.; Dharuman, G.; Kokkila Schumacher, S. I. L.; Turbyville, T.; Gulten, G.; Van, Q. N.; Goswami, D.; Jean-Francois, F.; Agamasu, C.; Chen; Hettige, J. J.; Travers, T.; Sarkar, S.; Surh, M. P.; Yang, Y.; Moody, A.; Liu, S.; Van Essen, B. C.; Voter, A. F.; Ramanathan, A.; Hengartner, N. W.; Simanshu, D. K.; Stephen, A. G.; Bremer, P. T.; Gnanakaran, S.; Glosli, J. N.; Lightstone, F. C.; McCormick, F.; Nissley, D. V.; Streitz, F. H. Machine learning-driven multiscale modeling reveals lipid-dependent dynamics of RAS signaling proteins. *Proc Natl Acad Sci U S A* **2022**, *119* (1). DOI: 10.1073/pnas.2113297119.
(2) Fujioka, A.; Terai, K.; Itoh, R. E.; Aoki, K.; Nakamura, T.; Kuroda, S.; Nishida, E.; Matsuda, M. Dynamics of the Ras/ERK MAPK cascade as monitored by fluorescent probes. *J Biol Chem* **2006**, *281* (13), 8917-8926. DOI: 10.1074/jbc.M509344200.
(3) Bhalla, U. S. Signaling in small subcellular volumes. II. Stochastic and diffusion effects on synaptic network properties. *Biophys J* **2004**, *87* (2), 745-753. DOI: 10.1529/biophysj.104.040501.
(4) Schoeberl, B.; Eichler-Jonsson, C.; Gilles, E. D.; Muller, G. Computational modeling of the dynamics of the MAP kinase cascade activated by surface and internalized EGF receptors. *Nat Biotechnol* **2002**, *20* (4), 370-375. DOI: 10.1038/nbt0402-370.
(5) Sasagawa, S.; Ozaki, Y.; Fujita, K.; Kuroda, S. Prediction and validation of the distinct dynamics of transient and sustained ERK activation. *Nat Cell Biol* **2005**, *7* (4), 365-373. DOI: 10.1038/ncb1233.
(6) Hatakeyama, M.; Kimura, S.; Naka, T.; Kawasaki, T.; Yumoto, N.; Ichikawa, M.; Kim, J. H.; Saito, K.; Saeki, M.; Shirouzu, M.; Yokoyama, S.; Konagaya, A. A computational model on the modulation of mitogen-activated protein kinase (MAPK) and Akt pathways in heregulin-induced ErbB signalling. *Biochem J* **2003**, *373* (Pt 2), 451-463. DOI: 10.1042/BJ20021824.
(7) Wisniewski, J. R.; Dus-Szachniewicz, K.; Ostasiewicz, P.; Ziolkowski, P.; Rakus, D.; Mann, M. Absolute Proteome Analysis of Colorectal Mucosa, Adenoma, and Cancer Reveals Drastic Changes in Fatty Acid Metabolism and Plasma Membrane Transporters. *J Proteome Res* **2015**, *14* (9), 4005-4018. DOI: 10.1021/acs.jproteome.5b00523.
(8) Sydor, J. R.; Engelhard, M.; Wittinghofer, A.; Goody, R. S.; Herrmann, C. Transient kinetic studies on the interaction of Ras and the Ras-binding domain of c-Raf-1 reveal rapid equilibration of the complex. *Biochemistry* **1998**, *37* (40), 14292-14299. DOI: 10.1021/bi980764f.
(9) Lee, H. W.; Kyung, T.; Yoo, J.; Kim, T.; Chung, C.; Ryu, J. Y.; Lee, H.; Park, K.; Lee, S.; Jones, W. D.; Lim, D. S.; Hyeon, C.; Heo, W. D.; Yoon, T. Y. Real-time single-molecule co-immunoprecipitation analyses reveal cancer-specific Ras signalling dynamics. *Nat Commun* **2013**, *4*, 1505. DOI: 10.1038/ncomms2507.
(10) Ingólfsson, H. I.; Bhatia, H.; Zeppelin, T.; Bennett, W. F. D.; Carpenter, K. A.; Hsu, P.-C.; Dharuman, G.; Bremer, P.-T.; Schiott, B.; Lightstone, F. C.; Carpenter, T. S. Capturing Biologically Complex Tissue-Specific Membranes at Different Levels of Compositional Complexity. *The Journal of Physical Chemistry B* **2020**, *124* (36), 7819-7829. DOI: 10.1021/acs.jpcb.0c03368 (acccessed November).
(11) Nguyen, K.; Lopez, C. A.; Neale, C.; Van, Q. N.; Carpenter, T. S.; Di Natale, F.; Travers, T.; Tran, T. H.; Chan, A. H.; Bhatia, H.; Frank, P. H.; Tonelly, M.; Zhang, X.; Gulten, G.; Reddy, T.; Burns, V.; Oppelstrup, T.; Hengartner, N.; Simanshu, D. K.; Bremer, P.-T.; Chen, D.; Glosli, J. N.; Shrestha, R.; Turbyville, T.; Streitz, F. H.; Nissley, D. V.; Ingólfsson, H. I.; Stephen, A. G.; Lightstone, F. C.; Gnanakaran, S. Exploring CRD mobility during RAS/RAF engagement at the membrane. *Biochem. J.* **2022**. DOI: 10.1016/j.bpj.2022.06.035.
(12) Travers, T.; López, C.; Van, Q.; Neale, C.; Tonelli, M.; Stephen, A.; Gnanakaran, S. Molecular recognition of RAS/RAF complex at the membrane: Role of RAF cysteine-rich domain. *Scientific Reports* **2018**, *8*, 8461. DOI: 10.1038/s41598-018-26832-4.

(13) Neale, C.; García, A. E. The Plasma Membrane as a Competitive Inhibitor and Positive Allosteric Modulator of KRas4B Signaling. *Biophysical Journal* **2020**, *118* (5), 1129-1141. DOI: 10.1016/j.bpj.2019.12.039.
(14) Wassenaar, T. A.; Ingolfsson, H. I.; Bockmann, R. A.; Tieleman, D. P.; Marrink, S. J. Computational Lipidomics with insane: A Versatile Tool for Generating Custom Membranes for Molecular Simulations. *J Chem Theory Comput* **2015**, *11* (5), 2144-2155. DOI: 10.1021/acs.jctc.5b00209.
(15) Abraham, M. J.; Murtola, T.; Schulz, R.; Páll, S.; Smith, J. C.; Hess, B.; Lindahl, E. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX* **2015**, *1-2*, 19-25. DOI: 10.1016/j.softx.2015.06.001.
(16) Zhang, X.; Sundram, S.; Oppelstrup, T.; Kokkila-Schumacher, S. I. L.; Carpenter, T. S.; Ingólfsson, H. I.; Streitz, F. H.; Lightstone, F. C.; Glosli, J. N. ddcMD: A fully GPU-accelerated molecular dynamics program for the Martini force field. *J Chem Phys* **2020**, *153* (4), 045103. DOI: 10.1063/5.0014500.
(17) Glosli, J. N.; Richards, D. F.; Caspersen, K. J.; Rudd, R. E.; Gunnels, J. A.; Streitz, F. H. Extending Stability Beyond CPU Millennium: A Micron-scale Atomistic Simulation of Kelvin-Helmholtz Instability. In *The 2007 ACM/IEEE Conference on Supercomputing*, Reno, Nevada, 2007; ACM: p 58.
(18) Streitz, F. H.; Glosli, J. N.; Patel, M. V. Beyond Finite-Size Scaling in Solidification Simulations. *Phys. Rev. Lett.* **2006**, *96*, 225701. (acccessed Jun).
(19) *ddcMDconverter*; github.com/LLNL/ddcMDconverter (accessed.
(20) Michaud-Agrawal, N.; Denning, E. J.; Woolf, T. B.; Beckstein, O. MDAnalysis: a toolkit for the analysis of molecular dynamics simulations. *J Comput Chem* **2011**, *32* (10), 2319-2327. DOI: 10.1002/jcc.21787.
(21) Gowers, R.; Linke, M.; Barnoud, J.; Reddy, T.; Melo, M.; Seyler, S.; Domański, J.; Dotson, D.; Buchoux, S.; Kenney, I.; Beckstein, O. MDAnalysis: A Python Package for the Rapid Analysis of Molecular Dynamics Simulations. In *The 15th Python in Science Conference*, Austin, Texas, 2016; pp 102-109.
(22) Marrink, S. J.; Risselada, H. J.; Yefimov, S.; Tieleman, D. P.; de Vries, A. H. The MARTINI Force Field: Coarse Grained Model for Biomolecular Simulations. *The Journal of Physical Chemistry B* **2007**, *111* (27), 7812-7824. DOI: 10.1021/jp071097f.
(23) Monticelli, L.; Kandasamy, S. K.; Periole, X.; Larson, R. G.; Tieleman, D. P.; Marrink, S.-J. The MARTINI Coarse-Grained Force Field: Extension to Proteins. *Journal of Chemical Theory and Computation* **2008**, *4* (5), 819-834. DOI: 10.1021/ct700324x.
(24) Periole, X.; Cavalli, M.; Marrink, S. J.; Ceruso, M. A. Combining an Elastic Network With a Coarse-Grained Molecular Force Field: Structure, Dynamics, and Intermolecular Recognition. *J Chem Theory Comput* **2009**, *5* (9), 2531-2543. DOI: 10.1021/ct9002114.
(25) de Jong, D. H.; Singh, G.; Bennett, W. F.; Arnarez, C.; Wassenaar, T. A.; Schafer, L. V.; Periole, X.; Tieleman, D. P.; Marrink, S. J. Improved Parameters for the Martini Coarse-Grained Protein Force Field. *J Chem Theory Comput* **2013**, *9* (1), 687-697. DOI: 10.1021/ct300646g.
(26) Lopez, C. A.; Sovova, Z.; van Eerden, F. J.; de Vries, A. H.; Marrink, S. J. Martini Force Field Parameters for Glycolipids. *J Chem Theory Comput* **2013**, *9* (3), 1694-1708. DOI: 10.1021/ct3009655.
(27) Ingólfsson, H. I.; Melo, M. N.; van Eerden, F. J.; Arnarez, C.; Lopez, C. A.; Wassenaar, T. A.; Periole, X.; de Vries, A. H.; Tieleman, D. P.; Marrink, S. J. Lipid Organization of the Plasma Membrane. *Journal of American Chemical Society* **2014**, *136* (41), 14554-14559. DOI: 10.1021/ja507832e.
(28) Melo, M. N.; Ingolfsson, H. I.; Marrink, S. J. Parameters for Martini sterols and hopanoids based on a virtual-site description. *J Chem Phys* **2015**, *143* (24), 243152. DOI: 10.1063/1.4937783.
(29) de Jong, D. H.; Baoukina, S.; Ingólfsson, H. I.; Marrink, S. J. Martini straight: Boosting performance using a shorter cutoff and GPUs. *Computer Physics Communications* **2016**, *199*, 1-7. DOI: 10.1016/j.cpc.2015.09.014.
(30) Bhatia, H.; Di Natale, F.; Moon, J. Y.; Zhang, X.; Chavez, J. R.; Aydin, F.; Stanley, C.; Oppelstrup, T.; Neale, C.; Schumacher, S. K.; Ahn, D.; Herbein, S.; Carpenter, T. S.; Gnanakaran, S.; Bremer, P.-T.; Glosli, J. N.; Lightstone, F. C.; Ingólfsson, H. I. Generalizable Coordination of Large Multiscale

Ensembles: Challenges and Learnings at Scale. In *The International Conference for High Performance Computing, Networking, Storage and Analysis*, 2021; ACM: p 10. DOI: 10.1145/3458817.3476210.
(31) Lopez, C. A.; Zhang, X.; Aydin, F.; Shrestha, R.; Van, Q. N.; Stanley, C. B.; Carpenter, T. S.; Nguyen, K.; Patel, L. A.; Chen; Burns, V.; Hengartner, N. W.; Reddy, T. J. E.; Bhatia, H.; Di Natale, F.; Tran, T. H.; Chan, A. H.; Simanshu, D. K.; Nissley, D. V.; Streitz, F. H.; Stephen, A. G.; Turbyville, T. J.; Lightstone, F. C.; Gnanakaran, S.; Ingolfsson, H. I.; Neale, C. Asynchronous Reciprocal Coupling of Martini 2.2 Coarse-Grained and CHARMM36 All-Atom Simulations in an Automated Multiscale Framework. *J Chem Theory Comput* **2022**, *18* (8), 5025-5045. DOI: 10.1021/acs.jctc.2c00168.