Supplementary Materials for

Machine Learning-driven Multiscale Modeling Reveals Lipid-Dependent Dynamics of RAS Signaling Proteins

Helgi I. Ingólfsson, Chris Neale, Timothy S. Carpenter, Rebika Shrestha, Cesar A López, Timothy H. Tran, Tomas Oppelstrup, Harsh Bhatia, Liam G. Stanton, Xiaohua Zhang, Shiv Sundram, Francesco Di Natale, Animesh Agarwal, Gautham Dharuman, Sara I. L. Kokkila Schumacher, Thomas Turbyville, Gulcin Gulten, Que N. Van, Debanjan Goswami, Frantz Jean-Francios, Constance Agamasu, De Chen, Jeevapani J. Hettige, Timothy Travers, Sumantra Sarkar, Michael P. Surh, Yue Yang, Adam Moody, Shusen Liu, Brian C. Van Essen, Arthur F. Voter, Arvind Ramanathan, Nicolas W. Hengartner, Andrew G. Stephen, Dhirendra K. Simanshu, Peer-Timo Bremer, S. Gnanakaran, James N. Glosli, Felice C. Lightstone, Frank McCormick*, Dwight V. Nissley*, Frederick H. Streitz*

*Corresponding authors. Email: frank.mccormick@ucsf.edu; nissleyd@mail.nih.gov; streitz1@llnl.gov

Foreword

This Supplementary Materials for the manuscript “Machine Learning-driven Multiscale Modeling Reveals Lipid-Dependent Dynamics of RAS Signaling Proteins” covers a range of topics and we advise accessing it: directly by referenced section from the main text, through the contents lists below, or by topic.

The document is split up into two main sections the supplementary materials and methods (pages 6-30) and the supplementary notes and discussion (pages 31-74). These are followed by supplementary References, and Appendix A (pages 75-80). Sections one and two are further split up into several distinct subsections by topic.

Section one is the method section and split in four main parts:

- **Section 1.1.** The Multiscale Machine-Learned Modeling Infrastructure (MuMMI) (page 6), gives a complete introduction to MuMMI and all its subcomponents, detailing those that have not been described before and summarizing and referenced those that have.
- **Section 1.2.** Creation of Models (page 12), goes through all the preliminary work needed for starting the multiscale simulation campaign described in this work. This includes the selection of lipids mixture, determinization of RAS structure, CG and macro model parameterization and training of ML framework.
- **Section 1.3.** Analytics details (page 22), details the different methods used for data analysis of the simulations.
- **Section 1.4.** Experimental methods (page 26), details all the different experimental methods used for accessing RAS structure and dynamics.

Section two goes into more detail on the different results presented in the manuscript. This section has nine subsections which roughly belong to two catagories:

- **Sections 1-3** (pages 31-45) cover details on the multiscale simulation, lengths, sampling and comparison between the macro and CG scales.
- **Sections 4-9** (pages 45-74) discuss different aspects of RAS membrane dynamics and aggregation.

Contents

List of Figures

List of Tables

1. Supplementary Materials and Methods

1.1. The Multiscale Machine-Learned Modeling Infrastructure (MuMMI)

The multiscale simulation of RAS-lipid biology discussed in this paper is facilitated by the MuMMI framework. MuMMI is capable of orchestrating massive simulations using a sophisticated workflow (1), and utilizes a dynamic-importance sampling approach (2) based on machine learning (ML) to couple macro (continuum) and micro (CG) model simulations. Fig. S1 shows the conceptual schema that MuMMI implements to coordinates macro and micro scale simulations. The macro model simulates biologically relevant time- and length- scales, which are currently intractable using higher-fidelity models. From the macro simulation, MuMMI extracts smaller neighborhoods of RAS, called *patches*, which are candidates for simulation at the CG resolution. Patches with highest importance (greatest novelty) are then selected using the ML-driven sampler and simulated at a higher-fidelity using CG molecular dynamics (MD).

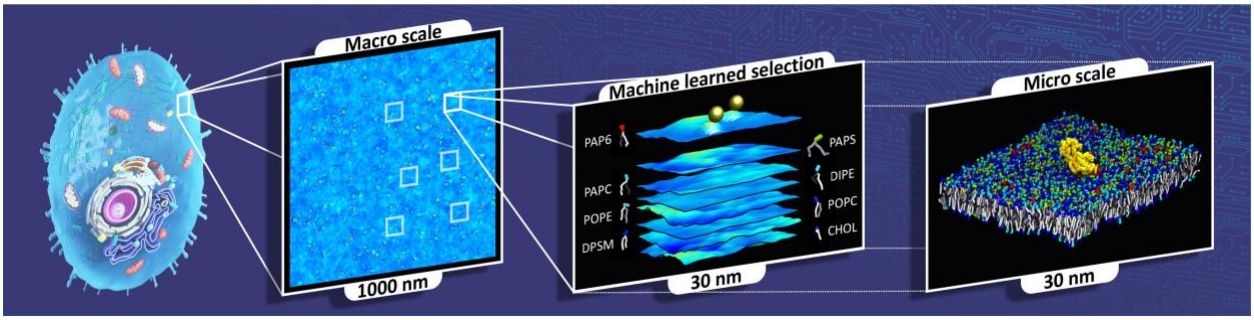

Fig. S1: MuMMI overview. MuMMI performs massively parallel multiscale simulations using an ML-driven sampling framework. The first layer is a macro scale (DDFT model) with an overlaid MD simulation of RAS particles. 30×30 nm^2 patches are extracted from the 1×1 μm^2 macro snapshots and are simulated at the CG MD level. Each selected patch is run concurrently occupying available resources as much as possible.

A MuMMI simulation is seeded using a dynamic density functional theory (DDFT) simulation that is coupled with MD responsible for moving RAS beads along the DDFT-generated concentration plane. As the macro model runs, patches are extracted around the mobile RAS beads, creating a set of local subregions that can be sampled at the CG scale. The ML-driven dynamic-importance sampling framework is pre-trained to prioritize these local subregions based on criteria of scientific interest. Each CG simulation comprises two major phases: set up and simulation. For CG set up, each selected macro model patch is mapped to a corresponding CG molecular configuration and then equilibrated in preparation for simulation. Each CG simulation is accompanied with in situ analysis and runs for at least one μs of simulation time. The analyses are responsible for saving full-system coordinate data to disk at a specified cadence because a higher frequency would otherwise be infeasible to store. Finally, MuMMI implements an on-the-fly feedback mechanism that improves macro model parameters using data collected from the the analyses of the higher-fidelity, CG simulations.

In the remainder of this section, we cover the different components of MuMMI. We start with the central workflow, hardware, and ddcMD (the MD simulation engine) (3) used to run our

campaign, followed by descriptions of the macro model, micro model, ML-based dynamic-importance sampler, and in situ analysis.

1.1.1. Workflow Management and Hardware

A new workflow management tool (1) was developed to coordinate MuMMI because no existing workflow tools are capable of supporting our target, ML-driven multiscale simulations. The MuMMI workflow manager (WM) manages the state and execution of the simulation, including generation and ingestion of patches, their selection using ML, management of CG simulations, and feedback from in sity analysis of CG simulations to the macro model.

The WM continuously polls for incoming new data from a running macro model simulation and uses them to generate patches, which are local neighborhoods of RAS proteins. These patches are then evaluated by a pre-trained ML model and selected for novelty (see Section 1.1.5) to spawn new CG simulations when resources become available. The WM monitors the available resources, starts new simulation tasks when needed, monitors running tasks, and restarts any that fail due to hardware issues or simulation instability. The WM is designed to provide robustness against hardware failures through extensive checkpointing and restoring capabilities. The WM also manages the feedback mechanism that updates macro model parameters. The WM collects the RAS-lipid radial distribution functions (RDFs) from the in situ analyses of CG simulations through the filesystem, aggregates them using appropriate ML weighting (see Section 1.2.7), and converts to the free-energy functionals needed for the macro model (see Section 1.1.3).

Fig. S2 highlights how MuMMI couples several diverse components to create the unique capabilities required by our scientific campaign. The WM is written in Python and interfaces with Maestro (4), an open-source workflow tool with abstracted scheduler APIs to support portability. Maestro further interfaces with Flux (5), a scalable resource manager that provides hierarchical scheduling and supports submitting and monitoring tens of thousands of jobs.

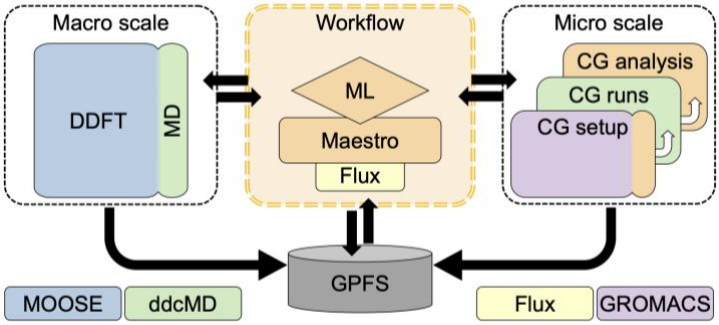

Fig. S2: MuMMI component scheme. MuMMI couples the macro scale (DDFT and MD) model with the micro scale (CG model) using a ML-based dynamic-importance sampling framework. Data resulting from the macro scale simulation is analyzed by ML, and interesting subregions are simulated at the micro scale. CG simulations are analyzed in situ and used to improve the macro model via on-the-fly feedback. The central workflow uses Flux as the resource manager, as abstracted using Maestro, and coordinates with each of the software components using in-memory and on-disk communication. Modules in orange are the core, specially-developed components of the MuMMI framework, and other colors represent external software extended for MuMMI.

MuMMI allows the creation of a simulation campaign of unprecedented scale. The work presented in this paper effectively utilized all of the 4000 nodes of *Sierra*, the second most-powerful supercomputer in the world (6) when the campaign was ran. MuMMI is easily scalable and can be deployed on small clusters as well with as few as 5 computational nodes. A *Sierra* node contains four NVIDIA® Tesla® V100 GPUs and two IBM® POWER9™ CPUs with 22 cores each. When sufficient work is available, MuMMI utilizes all four GPUs and 40 out of the 44 CPU cores available on each node by assigning them to the various components in our framework. Out of the 44 CPU cores available, four are used for the four CG simulations (ddcMD; see Section 1.1.4), three cores each for analysis of these simulations (see Section 1.1.6), and 24 cores for a single CG setup process (see Section 1.1.4). The remaining four cores were reserved for handling system processes and filesystem interaction for CG simulations. A breakdown of a typical 2,040 node run is shown in Fig. S3.

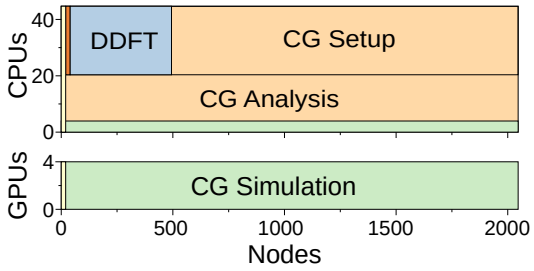

Fig. S3: MuMMI node breakdown. A breakdown of a typical 2,040 node run of MuMMI on *Sierra*. MuMMI utilizes the heterogeneous mix of GPUs and CPUs on *Sierra* by running CG setup and analysis on the POWER9 CPUs while simultaneously running ddcMD CG simulations on the GPUs.

1.1.2. The ddcMD Molecular Dynamics Engine

The main MD simulation software used within MuMMI is ddcMD (3, 7). This code has been applied in studying various problems in different research areas including biology, material science, fluid flows, and plasma physics (1, 8) and has twice won the Gordon Bell Prize (8, 9). For this work, ddcMD was significantly extended to take advantage of heterogenous architecture capabilities.

MuMMI utilizes ddcMD in two different ways. An existing CPU-only version is used to integrate protein equations in the macro model (see Section 1.1.3) and a specially-designed GPU-enabled ddcMD for micro model simulations (see Section 1.1.4). The new GPU capabilities were added to accelerate the Martini CG force field (10). The code not only supports GPU-enabled high-throughput MD but crucially also minimizes CPU utilization. Minimizing CPU usage is critical when working on architectures with low CPU to GPU resources or when executing frameworks, like MuMMI, with high CPU demand from other tasks (Fig. S3). The GPU-enabled ddcMD (3) has been released on GitHub (11).

1.1.3. Macro Model

In order to rapidly explore the bilayer system and the associated RAS proteins over long time- and length-scales, a macro scale model was designed where this macro model uses a continuum description of the lipids that is less-detailed than the Martini MD model. The macro model feeds

into the ML-based, dynamic-importance sampling framework, which identifies important patches to promote to CG MD simulations.

The macro model formalism and equations are described by Stanton et al. (12). In the macro model, the equations of motion for the lipid species are based on a DDFT formalism (13), where the corresponding free energy functional was constructed entirely from measurements of Martini MD (CG) simulations. The Ornstein-Zernike (OZ) equation (14) is used to convert the lipid-lipid radial distribution functions (RDFs) from CG simulations to direct correlation functions (DCFs), which are used to compute lipid-lipid interactions in the continuum DDFT formulation. Lipid self-diffusion coefficients measured in CG training simulations (see Section 1.2.6) are used to obtain mobility parameters for the macro model. The macro model represents each RAS protein as a single bead, which interact with the lipids through potentials of mean force (PMFs) that are also extracted from CG simulations using the OZ equation and the Hypernetted Chain (HNC) closure relation (15). Finally, RAS-RAS interactions are modeled by a pair potential, which is discussed below. In contrast to the multiplicative noise in (12) we chose a additive noise term (still mass-conserving). To solve the system of partial differential equations resulting from the DDFT formalism, the parallel finite element code MOOSE (16) was used in conjunction with the original CPU-only version of the ddcMD (see Section 1.1.2) MD code to integrate the equations of motion for RAS particles.

In this simulation campaign, the macro model included 8 lipid types that mimic the plasma membrane (PM) composition (see Section 1.2.1) and comprised a 1×1 μm^2 bilayer at a resolution of 1200×1200 cubic-order elements. The membrane had 300 RAS molecules. The macro model was run on a relatively small number of nodes (50-500) and used only CPUs. On 900 cores, it can perform 6.3 μs per day in this setting.

In order to support on-the-fly feedback, the implementation of the macro model allows for updating the parameter set of the macro model in real time. By reading the improved RDFs accumulated by the workflow via CG simulations, PMFs are calculated using the OZ and HNC equations, and periodically loaded into the macro model simulation.

1.1.4. Micro Model

When a region of the macro model is defined to be of interest, it is scheduled for CG MD simulation setup and initial equilibration. After a simulation is setup and equilibrated, it is run using a CUDA-based GPU version of the ddcMD MD program.

The setup module transforms a selected patch of the macro model into a particle-based micro representation (see Fig. S4). Each selected macro model patch is 30×30 nm^2 with one RAS molecule at the center and may include additional RAS. The patch is instantiated and equilibrated for a Martini simulation. Within a patch, the macro model indicates the concentration and asymmetry of all membrane lipids, which are resolved down to a 5×5 subgrid (with a subgrid spacing of 6 nm), and indicates the number, states (Section 1.2.5), and locations of included RAS proteins. The proteins, lipids, ions, and water molecules are placed in their initial **x**,**y**,**z** coordinates using a modified version of the *insane* membrane building tool (17). The modification to the *insane* tool allows lipid concentrations to be specified with subgrid resolution in each membrane leaflet. For the 30×30 nm^2 patches a maximum of 3,200 lipids were placed, 1,600 in each leaflet, with 64 lipids randomly arranged within each subgrid. The macro model

lipid densities were averaged within the 5×5 subgrids and rounded to integer values, with the residuals amortized between the subgrids. The difference in total densities between the inner and outer leaflets in the macro model for that patch were used to adjust the total lipid count between the two leaflets, accounting for the difference in lipid area between the two leaflets due to the composisional asymmetry of the patch. Initial RAS conformations were generated from a set of 30 CG MD simulations of standard patch size, with one RAS molecule each, and run for ~33 μs each (see Section 1.2.4 for details). RAS states (Section 1.2.5) were defined for this simulation set and a library of 1000 confirmations was saved for each state. For each RAS in a patch during MuMMI CG simulation, the initial conformation is randomly sampled from these preconstructed libraries based on the RAS molecules' conformational state. The location dictated by the macro model determines the farnesyl position in the bilayer plane (**x**,**y**). Apart from the farnesyl, all RAS proteins are moved away from the membrane by 2.25 nm. The RAS proteins are placed one at a time. Each protein is randomly rotated at the farnesyl in place around the **z** axes (same as the membrane normal for the initial flat bilayer). If there is no overlap with other proteins, the RAS is placed with that orientation. Otherwise, the rotation is repeated for as many as 20 attempts for placements. If all attempts are unsuccessful, the new RAS is translated out along the bilayer plane away from the already placed RAS by 0.5 nm and the rotation process is repeated. After all RAS have been placed, the initial coordinates are energy minimized, equilibrated, and the RAS molecules are pulled towards the membrane. The GROMACS MD package v5.1.4 (18) (CPU-only version) is used for energy minimization, equilibration, and pulling (using only the CPU cores allows the workflow to set up new CG simulations without competing with production CG simulations, which utilize the GPUs).

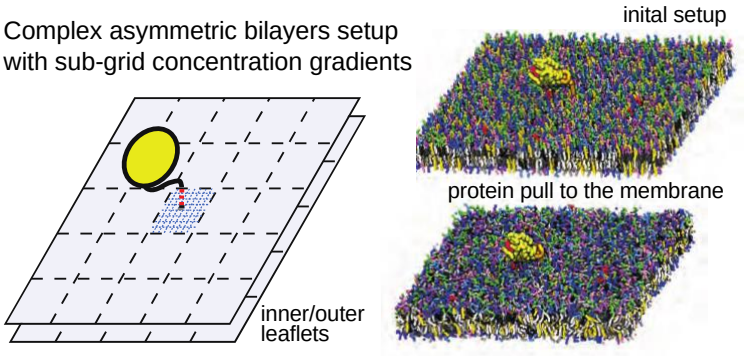

Fig. S4: Macro to micro simulation setup. Particle-based micro simulations are created based on input from the macro model. Micro model, CG MD simulations are instantiated based on protein locations, protein states (Section 1.2.5), and spatially resolved lipid concentrations. Snapshots of a representative patch with one RAS molecule are shown after construction and initial protein/bilayer equilibration.

The setup process includes particle creation, 1,500 steps of energy minimization, 5,000 steps with a short 1 fs time step and 20,000 steps with a 5 fs time step, where the proteins are away from the bilayers and both proteins and lipid anchors are constrained. The proteins are then pulled to the bilayer with each protein having its own umbrella pull group (**z**-axes only, 0.0006 nm/ps, 1,000 kJ mol^{-1} nm^{-2}) in a 400,000 step MD simulation with a 10 fs time step. After pulling, a further 2 ns of equilibration is run (100,000 time steps with a 20 fs time step) with only weak (10 kJ mol^{-1} nm^{-2}) position restraints on protein beads in **x** and **y** dimensions. For these system sizes, ∽140,000 particles, one CG setup takes ~1.5±0.1 h on up to 24 POWER9 CPU cores.

After setup and initial equilibration, GPU-enabled ddcMD (see Section 1.1.2) is used for MD production simulations. All CG simulations (equilibrium and production runs) were run using the new-rf Martini parameter set (19) with a final time step of 20 fs, at 310 K and 1 bar semiisotropic pressure coupling. The nonbonded interaction is calculated using Lennard-Jones potential with a cut-off radius of 11 Å. The electrostatic interaction is treated by the reaction field method (20). The dielectric constant within the cut-off 11 Å is 15 and beyond the cut-off is infinite. The velocity Verlet algorithm is employed in integrating the Newtonian equations (21). The RATTLE method is used for the molecules with constraints (22). The Langevin thermostat is used with a friction coefficient of 1 ps^{-1} (23). The Berendsen barostat is used for the pressure calculation with a compressibility constant of 3.0×10^{-4} bar^{-1}. Position restraints are applied to the POPC lipids of the outer leaflet to limit large scale bilayer undulations, these are weak (2 kJ mol^{-1} nm^{-2}) harmonic potentials applied to the **z**-direction of each lipid PO4 bead. The orthorhombic periodic boundary condition is applied to the system. The MD trajectory is saved every 25,000 steps or 0.5 ns. For these system sizes, ∽140,000 particles, ddcMD produces on average 1.02±0.002 μs of MD simulation per day using 1 CPU core and 1 GPU.

1.1.5. Machine Learning Based Dynamic-Importance Sampling

Given limited computational resources and millions of potentially important patches (local macro configurations), MuMMI must decide how to best invest its resources by simulating at the micro scale only those patches that are most likely to provide new insights. At the core of our multiscale simulation lies the ML-based dynamic-importance (DynIm) sampling approach (2) that investigates all patches and decides which micro simulations to spawn.

The DynIm framework (2) comprises two key steps. First, it uses an autoencoder — an unsupervised deep neural network — that is trained to encode lipid configurations (within patches) into a reduced dimensional latent space that decouples the inherent correlations and captures the characteristic behavior of lipids. This latent space provides a more-pertinent similarity metric that can be used to compare patches. Second, DynIm uses this similarity metric to identify in real-time the most "novel" lipid configurations, i.e., the most dissimilar to the ones that have already been selected. A pretrained autoencoder, along with the sampling framework, was deployed in MuMMI for real-time selection of patches for CG simulations.

Exploring the space of macro configurations includes two related but separate aspects. First, the macro model explores common, low-energy lipid configurations significantly-more frequently than others. Our ML-based sampling approach facilitates a more uniform sampling of the underlying configuration space (as compared to a random sampling) and, therefore, identifies infrequently-occurring configurations. Understanding the likelihood of any given configuration is crucial for any subsequent analysis, and the DynIm framework records this likelihood for all selected and nonselected configurations. Second, the resulting likelihood may be used to reconstruct the distribution in the configuration space at any instant in time. MuMMI utilizes this information to remove the bias introduced by novelty sampling and aggregate statistics from ongoing MD simulations. This results in MuMMI's capability to augment the macro model information with insights from the micro scale to accumulate higher-fidelity data.

Note that the selection algorithm works dynamically as the multiscale simulation proceeds; the framework makes the best selection at a given point a time, according to the current state of the previous selections. Consequently, different resource constraints, e.g., choosing 5 patches per minute vs. 10 patches per minute, might result in somewhat different sequences of patches being selected. Nevertheless, given enough time and resources, the framework converges to a uniform sampling of the configuration space. As a result, the sampling framework can adapt to the scale of the multiscale simulation, covering a wide range from ~5 computational nodes all the way through several thousands.

1.1.6. In situ Analysis

A key enabling technology in MuMMI is the in situ analysis capability for the MD simulations, allowing for high-frequency analysis of specific properties of interest and on-the-fly feedback from the micro simulations to the running macro model. In situ analysis is particularly useful when dealing with such vast numbers of simulations as network filesystems may struggle to cope with the required I/O bandwidth or storage requirements. MuMMI contains a custom Python analysis module that is run for each running micro simulations locally on the same node. Newly generated simulation snapshots are saved to an on-node RAM disk, which provides high I/O throughput, and consumed immediately by the corresponding analysis. Each frame is read using an extended version of the MDAnalysis package (24, 25) that is able to parse the native ddcMD binary and ASCII data formats. In the current campaign, in situ analysis was performed every 0.5 ns of the CG MD simulations and simulation frames were saved for offline analysis every 2 ns. The saved frames, results of online analysis and simulation restart files, were synced from local RAM disks to the network filesystem every 40 ns of simulation time. The online analyses performed were chosen to support the on-the-fly feedback to update the macro model, and a number of RAS and lipid properties of interest. The features extracted were lipid leaflet location, lipid concentration fields per leaflet and for each RAS:RAS state, RAS lipid RDFs, RAS-RAS contacts, and RAS-lipid contacts. The different analyses are described in Section 1.3. We note that the RAS states are defined with Markov state analysis of the RAS tilt and rotation with respect to the membrane (see Section 1.2.5). The RAS-membrane tilt and rotation are defined as shown in Fig. 5a. We use two angles to define the orientation of RAS' G-domain with respect to the membrane surface. The tilt angle represents the deflection of the long axis of helix 5 (α 5) away from the bilayer normal Fig. 5a (top panel) and the rotation angle represents the direction in which that tilt occurs Fig. 5a (bottom panel). All analysis routines are optimized such that using 3 POWER9 CPU cores for each simulation the online analysis can keep up with the frequency of incoming frames from ddcMD.

1.2. Creation of Models

Akihiro Kusumi has proposed a model of the plasma membrane that is hierarchical and dynamic in nature (26). Within different length-scales from microns to nanometers, lipids, proteins, such as RAS, and associated structures interact and form transitory complexes that regulate or activate major cell processes including cell signaling. Existing structural biology and biophysical techniques, including imaging, are unable to resolve the precise molecular details of these interactions. It is crucial to bridge the temporal and resolution gaps to understand the biology, and the multiscale simulation framework, MuMMI is designed for this purpose.

For this framework to provide biologically relevant information, care needs to be taken in identifying the input parameters for the model. A variety of estimates for the number of RAS molecule/cell are available based on several experimental approaches, providing a range of 46,000 – 820,000 KRAS molecules per cell (27-29). Assuming the diameter of these cells to be 20 μm and all RAS proteins are localized to the PM, the total number of PM-localized KRAS molecules is between 38 and 678 KRAS molecules/μm^2. In this work 300 RAS molecules/μm^2 is selected to be within the range of experimentally measured values and allows for sampling of RAS-RAS interactions.

RAS membrane dynamics are captured at the micro scale by using the CG Martini force field (10, 28, 29) with a created PM model (Section 1.2.1). Martini has been shown to be well suited to capture membrane dynamics and membrane lipid interactions (31-33), but as with any CG force field there is a balance of pros and cons; when evaluating results, all model limitations need to be carefully considered (10, 34-36). RAS protein parameters were optimized (Section 1.2.3) from our solved active state structure of RAS (Section 1.2.2). In the following subsections we list the control and parameterization simulations that were run in advance of the MuMMI campaign (Section 1.2.4), how the parameters for the macro model were derived (Section 1.2.5), and the optimizations needed for the machine learning guided selection of patches (Section 1.2.6).

1.2.1. Plasma Membrane Model

Cellular PMs consist of hundreds of different types of lipids that are actively regulated by the cell (37, 38) and asymmetrically distributed between the leaflets (39). Recent MD efforts using biologically relevant complex lipid models at the CG Martini resolution have started to reveal details of the lipid organization of PMs (40-43). Due to their high complexity (~60 lipid types) these compositions are not directly applicable to either modeling at the macro scale or to in vitro experimental investigation. Therefore, we used a simpler CG PM mimic using 8 lipid types that maintains the overall PM character and properties (44).

The RAS minimal PM mimic model is composed of 8 different lipid types: two phosphocholine (PC) lipids (1-palmitoyl-2-oleoyl-*sn*-glycero-3-phosphocholine; POPC and 1-palmitoyl-2-arachidonoyl-*sn*-glycero-3-phosphocholine; PAPC), two phosphoethanolamine (PE) lipids (1-palmitoyl-2-oleoyl-*sn*-glycero-3-phosphoethanolamine; POPE and 1,2-dilinoleoyl-*sn*-glycero-3-phosphoethanolamine; DIPE), one sphingomyelin (SM) lipid (N-stearoyl-D-erythro-sphingosylphosphorylcholine, DPSM), one phosphatidylserine (PS) lipid (1-palmitoyl-2-arachidonoyl-*sn*-glycero-3-phosphatidylserine, PAPS), one phosphatidylinositol (PIP) lipid (phosphatidylinositol 4,5-bisphosphate with a stearoyl and arachidonoyl tails, hereon referred to as PIP2 or PAP6), and cholesterol (CHOL). Note, in Martini both stearoyl and palmitoyl are mapped to the same tail denoted with P. We refer to the global macro model lipid composition as the average-RAS lipid composition or ARC (Fig. S5).

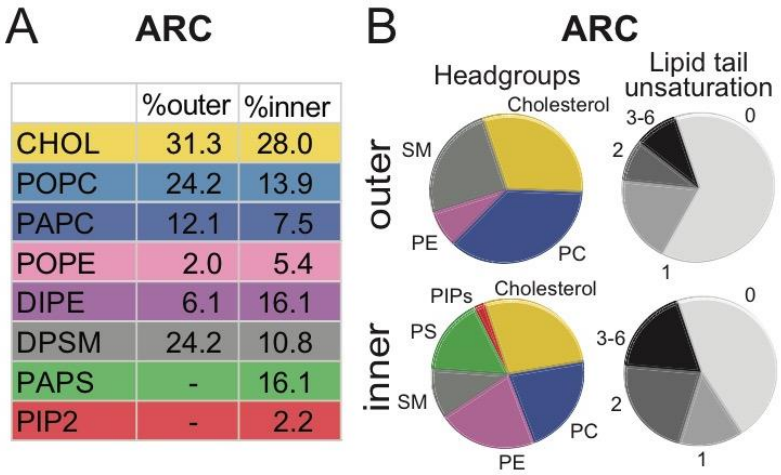

Fig. S5: Average-RAS minimal plasma membrane mixture (ARC). (A) The percentage of each of the lipid species is given within each leaflet in the ARC. Due to the leaflet asymmetry and the difference in the sizes of different types of lipids, the inner leaflet has fewer lipids than the outer leaflet (in a 0.926:1 ratio). Each lipid type is given a default color, which will be maintained throughout this manuscript. (B) The pie-charts show the lipid headgroup distribution and the levels of tail unsaturation for the outer and inner leaflets.

The Martini lipid parameters for the RAS minimal PM model are available online at https://bbs.llnl.gov/data.html as well as on the Martini portal, http://cgmartini.nl. The lipid parameters were originally constructed according to the standard Martini 2.0 lipid building blocks and rules (10, 45) and are detailed by Ingólfsson et al. (46) and Wassenaar et al. (17) The updated Martini cholesterol model was used (47), except for production MD simulations in ddcMD. Currently, virtual sites are not supported in ddcMD; therefore, a hybrid model was constructed based on the original non-virtual site Martini cholesterol (10), but with updates to the cholesterol shape to reflect that of the never-virtual sites model (47) in order to maintain the improved lipid phase behavior of the newer model. Using a 20 fs timestep in ddcMD, the updated cholesterol model was generally stable and when instabilities emerged they could always be resolved by restarting from the last checkpoint.

PIP2 is a phosphatidylinositol 4,5-bisphosphate lipid with stearoyl and arachidonoyl tails. The tail parameters are from the standard Martini building blocks (10, 17) and the headgroup parameters are the newly extended Martini PIP parameters described by Sun et al. (48), which are based on the original PIP parameters (49).

1.2.2. RAS Structure

Considering that a structure of active KRAS4b was not available in the Protein Data Bank at the start of this work, we solved the crystal structure of KRAS bound to GMPPNP at 2.5 Å resolution (PDB 6VC8). The overall structure resembles the structures of other RAS isoforms and contains a central β-sheet formed by six β-strands (β1-β6), which is surrounded by five α-helices (α1-α5) (50). In this active KRAS structure, the switch I region (residues 30-38) is present in the conformation (state 2; (51)) that is compatible for binding to effector proteins (Fig. S6). In contrast to a recently resolved structure of wild-type KRAS (52), this structure represents an active conformation in which switch I adheres to the bound GTP analog and magnesium ion,

making RAS compatible with effector binding (state 2; (51)), similar to another recent active state KRAS structure (53) (Fig. S6A).

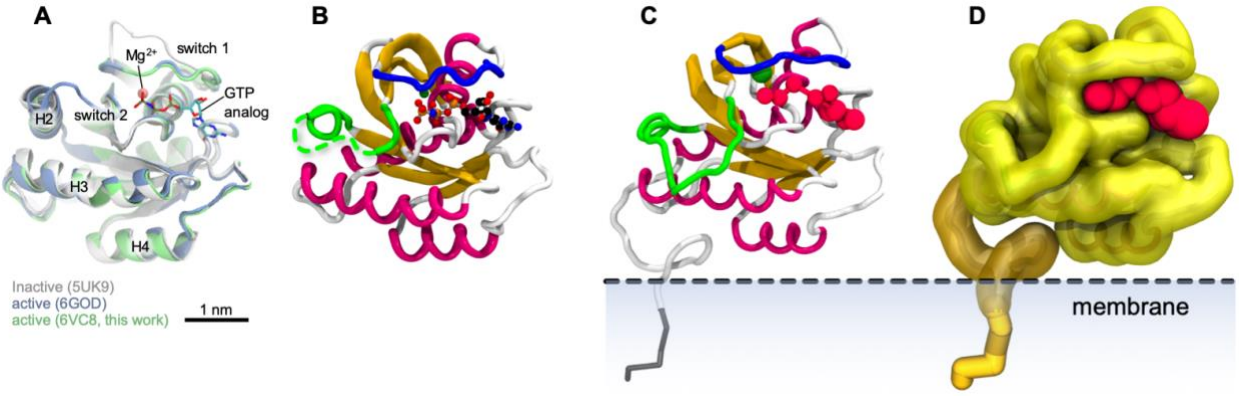

Fig. S6: RAS structures. (A) The crystal structures of wild-type KRAS in active (green and blue; GppNHp-bound) and inactive (grey; GppCH2p-bound configurations). The nucleotide analogue and Mg^{2+} are shown in stick and sphere representation, respectively. (B) The all-atom structure conformation constructed from 6VC8. The nucleotide analogue and Mg^{2+} are shown in ball-and-stick representations and colored according to the CPK atom colors. The protein secondary structure elements β-strands, helices and loops are colored in yellow, magenta and white, respectively. The switch I and II regions are colored blue and green, respectively. (C) The same coloring scheme and representation for (B) is used to show the equivalent coarse-grained Martini structure, as used in the simulations. The HVR and farnesyl tail modeled into the structure. (D) The same structure as (C) is illustrated in a globular, volume-filling representation. The G-domain is colored yellow, and the HVR and farnesyl tail are orange. For both (C) and (D), the location of the membrane is illustrated.

For a few residues in the helix 2 and switch II region of chain B, no electron density is observed. Using the Loop Modeler function integrated in the commercial software Molecular Operation Environment (MOE) (54), these missing residues are modeled to complete the G-domain of the KRAS structure. First, the missing sequence (Q61-R68) is added, followed by modeling the secondary structure. Both a PDB (database) template approach and a *de novo* approach are used, and the model associated with the best score is chosen for subsequent modeling and MD simulations. For optimal loop modeling performance, A59, G60, Q70 and Y71 are also included, resulting in a total of 12 modeled residues. The backbone atoms of selected residues are constructed as indicated by the Loop Modeler and subsequently sidechains are added. The resulting system is then modeled using CHARMM36 FF (55), with explicit solvent modeled using TIP3P water model (56) with CHARMM modification. A short minimization followed by 100 ns equilibration was performed using AMBER PMEMD (57), with positional restraints applied to backbone heavy atoms, except for those modeled using Loop Modeler.

1.2.3. RAS CG Model

The initial coordinates used for KRAS4b are as described above (Section 1.2.2). These coordinates are transformed into CG beads using *martinize.py* v2.6 (see http://cgmartini.nl) and standard parameters based on the Martini 2.2 protein force field (58). The first residue, G1, which is retained after cleavage of the expression tag, is removed as this is a non-native amino acid. Furthermore, mass spectrometry indicates that the N-terminal methionine residue of KRAS

is cleaved in mammalian cells, with subsequent N-terminal backbone acetylation of threonine 2 (59). Therefore, the N-terminal residue of our CG RAS model is the biologically relevant backbone-acetylated threonine 2. In addition to C-terminal side chain farnesylation (see below), the C-terminus is also backbone methylated. Therefore, in our Martini model, both the N and C terminal are capped using a CG representation of acetylation and methylation chemical modifications. Thus, the N-terminal Qd bead is replaced by a P3 bead and the C-terminal Qa bead is replaced by a C5 bead. In either case, the capped regions are maintained neutral (zero charge).

Stability of folded RAS is preserved using an internal elastic network, which is applied to the residues forming the G-domain (threonine 2 to histidine 166), bound guanosine triphosphate (GTP), and Mg^{2+}, using the default elastic network (cutoff of 0.9 nm and restraint force constant of 500 kJ mol^{-1} nm^{-2}). Parameters for the Martini representation of GTP were provided by Carsten F. E. Schroer (University of Groningen) and were derived from the Martini DNA/RNA parameters (60, 61). The accompanying Mg^{2+} was modeled with a Martini Qd particle of +2 charge.

The protein is anchored to the membrane via a farnesyl group, which is post-translationally attached to C-terminal residue cysteine 185. Parameters (62) for such chemical modification are obtained via thorough calibration using the general Martini philosophy. Thus, internal CG dynamics is iteratively incorporated using atomistic derived data using the general CHARMM36 force field. In order to consistently represent the behavior of the molecule according to the Martini approach, its preferential octanol/water partition coefficient was calibrated and directly compared with the partitioning obtained with the CHARMM36 force field. Preferential partitioning and localization of the farnesyl group was also tested in the context of a simple POPC membrane. Both properties proved to be in very good agreement when compared with the AA resolutions, giving enough confidence and reliability for its application in combination with the protein topology. Note that a different set of parameters for the CG simulation of farnesylated cysteine is currently available (63). However, for our purpose, parameters were derived from an improved atomistic representation, which has been previously published (64).

1.2.4. Control and Parameterization Simulations

Extensive sets of CG simulations were carried out in order to validate the behavior of mixed lipid systems with and without RAS, as well as to provide input parameters for the macro model. Simulation systems were composed of the 8 lipid ARC mimic described in Section 1.2.1, using the RAS model parameters described in Section 1.2.3, and the simulation parameters defined in Section 1.1.4. Four distinct sets of simulations were completed: (a) 'standard' size (~3,200 lipids, 30×30 nm^2) systems of the ARC, (b) standard size systems of the ARC with a single RAS molecule, (c) 'smaller' size (~430 lipids, 11×11 nm^2) systems of the ARC, and (d) smaller size systems of the ARC with a single RAS molecule (See Table S1). A mixture of different-sized systems was used as a way to both replicate the environment for the production run simulations, as well as to generate as much data as possible using smaller representations. For certain parameters (such as RAS-lipid RDFs), the standard-size systems were required to achieve better sampling of low populations of different components of the system. The standard-size system was also required to allow long-range measurements of the RDFs. Cross validation was also carried out to ensure that parameters calculated using the smaller systems were not subject to

size effects. Table S1 summarizes the different simulations used to generate the initial macro model parameters; lipid-lipid RDFs, RAS-lipid RDFs, RAS states, initial RAS conformations, RAS and lipid diffusion coefficients.

Table S1: Sets of parameterization simulations.

Set Name	# lipids	# RAS	# simulations	Length (μs)	Parameters calculated from simulations
Standard PM	~3,200	0	30	30	Lipid-lipid RDFs
Standard PM + RAS	~3,200	1	30	30	Initial RAS conformation libraries, RAS-lipid RDFs, RAS diffusion
Small PM	~430	0	100	10	Lipid diffusions
Small PM + RAS	~430	1	200	10	Definition of states, RAS diffusion

1.2.5. RAS States

In order to describe the orientational states of RAS, we used Hidden Markov Models (HMMs) (65), which build upon the architecture of Markov state models (MSMs) (66-68). In MSMs, the state space is discretized into *n* discrete states (also called *microstates*), and the system's dynamics is modeled by a $n \times n$ transition probability matrix, where an element *ij* in the transition matrix represents the probability of switching from state *i* to state *j* at time *t* (also called the *lag-time*). The lag-time is chosen to ensure that the system has lost its memory, inferred using the shape of the relaxation time-scale vs. lag-time plot, also called the *implied time-scales plot*. The analysis of the MSM transition matrix gives information on the slowest time-scales and processes in the system, the metastable states, and the transition pathways. Since MSMs can consist of several hundreds (or thousands) of microstates, they are post-processed to provide a coarse-grained model that can be explained in terms of a few metastable states and is more easily interpreted. Therefore, Perron Cluster-Cluster analysis (PCCA) (69) is usually applied to obtain the most-metastable set of macrostates. The number of metastable sets is specified by the user based on a separation of time-scales between the $(M-1)^{\text{th}}$ process and the M^{th} process, then *M* metastable sets are sufficient for understanding the system's kinetics. However, MSM's accuracy hinges on the input subspace and the quality of discretization. Thus, results obtained from MSMs may differ with different order parameters and clustering methods.

In this work, this limitation was overcome by using HMMs, which consists of a $M \times M$ transition matrix describing the dynamics between the metastable states and an output probability matrix with dimensions $M \times n$, where the row vector gives the probability that the metastable state will output to one of the *n* discrete states. HMMs can be estimated by Baum-Welsch Expectation-Maximization algorithm (70). All thermodynamic and kinetic properties calculated from MSMs can also be computed using HMMs. Furthermore, even with a poor discretization quality, it has been shown that the metastable dynamics can be exactly described using HMMs (65). All MSM/HMM construction and analysis in this work is performed with PyEmma software package (71).

In order to apply the HMM technique to our preliminary CG simulation data, the orientational state of RAS in the 8 lipid PM mimic system was analyzed based on both the tilting and rotation angles obtained from the training data Fig. 5a. K-means clustering algorithm (72) was used to discretize the tilt-rotation space into 2,000 microstates, and the implied time-scales were calculated as a function of lag-time. Fig. S7A shows the population map in the tilting and rotation space, and Fig. S8 shows the implied time-scales plot obtained from MSM analysis of the training data, where the relaxation time-scales (calculated from the eigenvalues of the transition matrix) are plotted as a function of the lag-time. The figure shows a separation of time-scales between the first (solid blue line) and the second relaxation time-scale (solid red line), indicating that the dynamics in our preliminary simulations can be fully described by two metastable states, which we refer to as α and β states. Therefore, a maximum-likelihood HMM was constructed using two states and a lag time of 1 μs. Fig. S7B shows HMM macrostate boundaries obtained by crisp assignment of microstates to metastable sets along with their relative equilibrium populations and Fig. S7C shows the effect of adding a third macrostate to the HMM calculation. The third state, refered to as state *t*, has a population of just ~6% implying that, from our preliminary simulations, only two metastable states are necessary to describe RAS' orientational dynamics on the membrane.

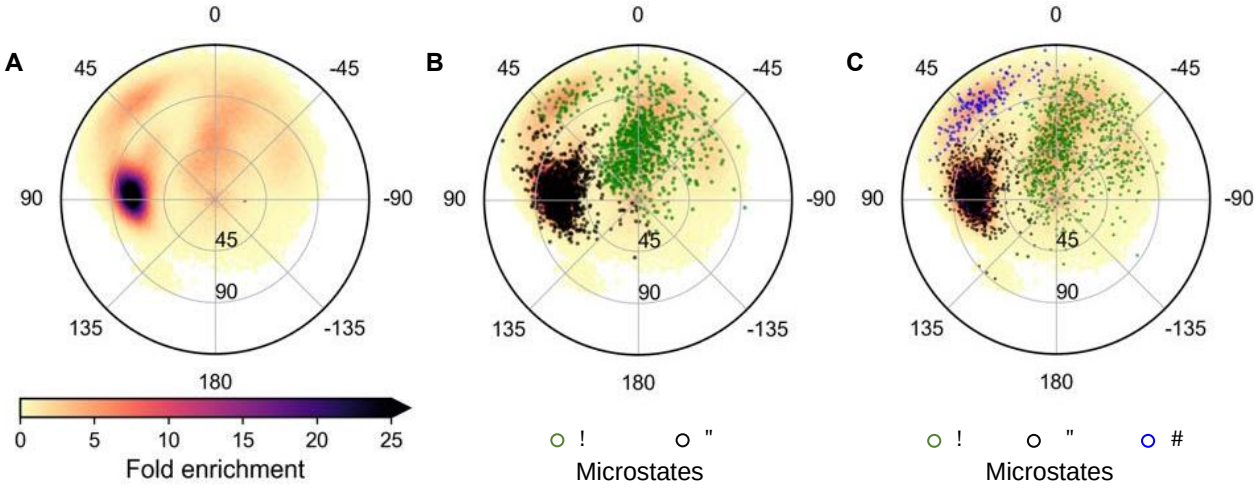

Fig. S7: HMM analysis of training data. (A) Population map in tilt-rotation space. (B) Two metastable states obtained from HMM. The microstates are colored according to the macrostates they belong to. The populations of the states are: α: 43%, β: 57%. (C) Three metastable states obtained from HMM, with the corresponding populations as: α: 40%, t: 6%, β: 54%.

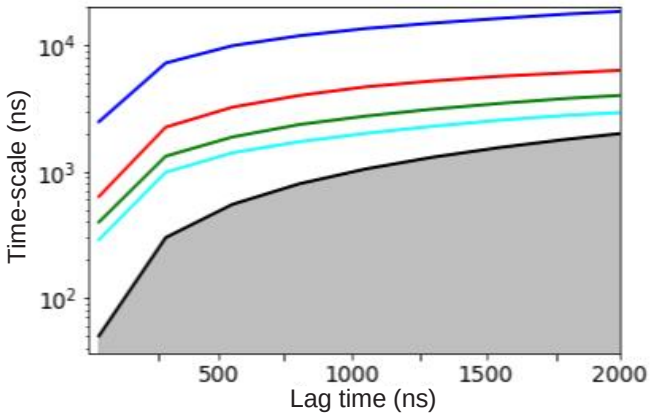

Fig. S8: Implied time scales from training data. Relaxation times calculated as a function of lag time are shown in different colors. Any relaxation process below the black line (y=x) cannot be reliably estimated as the time-scales of these processes have already decayed.

1.2.6. Macro Model Parameterization

Initial parameters for the macro model were calculated from CG MD Martini parameterization simulations (described in Section 1.2.4). The parameters to the macro model consist of:

a) diffusion coefficients for the different lipids,
b) additive noise magnitude for the lipids,
c) diffusion coefficients for RAS in the two different orientational states,
d) lipid-lipid correlation functions,
e) potentials for lipid-RAS and RAS-RAS interactions, and
f) state change rates for RAS.

For the RAS-RAS interactions we used a 9-6 Kihara potential parameterized to be mildly attractive and with distance parameters estimated from Martini simulations with two RAS molecules (see Section 1.1.3). Based on the RAS state analysis (Section 1.2.5), the macro model has two states for KRAS, labeled α and β. State changes are modeled by a Markov model with transition rates computed from observed state changes in 2-RAS Martini simulations. The transition rates used are: $\alpha \rightarrow \beta = 0.03\ \mu s^{-1}$, and $\beta \rightarrow \alpha = 0.0042\ \mu s^{-1}$.

Diffusion coefficients for RAS and the different lipids were calculated from the mean-square displacement of the respective molecule as a function of time. Our diffusion coefficients given in $nm^2/\mu s$ are as follows.

Table S2: Macro model diffusion coefficients ($nm^2/\mu s$).

Leaflet	**Species**									
	CHOL	POPC	PAPC	POPE	DIPE	DPSM	PAPS	PIP2	α-state	β-state
Inner	42.9	46.0	44.0	39.0	49.0	45.0	49.0	32.0	8.0	18.0
Outer	42.9	36.0	36.0	31.0	34.0	35.0	n/a	n/a	n/a	n/a

The additive noise terms for the lipids were adjusted so that the density fluctuations were similar to observations in Martini simulations.

The lipid-lipid correlation functions were computed from lipid-lipid radial distribution functions (RDF's) using the OZ equation (see Section 1.1.3). Our macro model is two dimensional; the 2D correlation functions required by the model were computed for each lipid type by selecting a bead that stays at a relatively constant height close to the membrane surface in CG simulations (see Section 1.3.1). The horizontal positions of these selected beads were used to compute radially-averaged RDF's to use in the 2D OZ relations. Given these RAS-lipid RDFs, the RAS-lipid PMFs are derived using the HNC. The horizontal position of the F1 bead of the farnesylated cysteine is used to represent the RAS position for RDF calculations.

Initial testing revealed that the PIP2-PIP2 DDFT model from the lipid-lipid RDF's creates islands of PIP2 with very sharp interfaces and with unphysically high densities — over 5 lipids/nm^2, which is more than twice as high as a pure PIP2 membrane patch. Inspection of the PIP2-PIP2 RDF shows a secondary tail feature extending to over 3 nm within which PIP2 has enhanced probability of PIP2 neighbors. This kind of long-range tail is not found in any other lipid-lipid RDFs in our Martini simulations of the ARC mixture, and all lipids except PIP2 are well-behaved in our DDFT model without intervention. In order to resolve the PIP2-aggregation and create a physically reasonable model, we chose to replace the Martini PIP2-PIP2 RDF in the OZ equations (see Section 1.1.3) with a scaled version of a more normally appearing lipid-lipid RDF, see Fig. S9. We chose the PIP2-DIPE as a source for this scaling, since it has a very similar undulation wavelength and primary structure. We then applied a logarithmic scaling of it to match the height of the first peak to the original PIP2-PIP2 first peak while keeping zeros and ones at zero and one respectively. Specifically, for the PIP2-PIP2 RDF in the OZ equations (see Section 1.1.3) we used:

$$RDF^{OZ}_{PIP2-PIP2} = (RDF_{PIP2-DIPE})^{\log(max\ RDF_{PIP2-PIP2})/\log(max\ RDF_{PIP2-DIPE})}$$

The resulting DDFT model is well behaved, and, with the exception of PIP2-PIP2 neighbors, represents lipid neighbors counts with an average error of 4%, and 9% in the worst case. PIP2-PIP2 neighbors are underrepresented (Fig. S23), preseumably largely due to the scaling procedure. Details can be seen in Section 2.3.3.

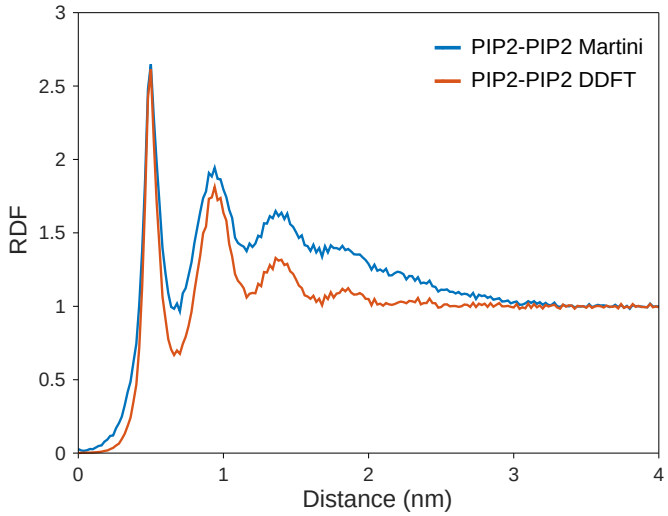

Fig. S9: PIP2-PIP2 RDFs. The lipid-lipid PIP2-PIP2 RDF calculated from Martini lipid parameterization simulations (blue curve) and the scaled PIP2-DIPE RDF used to replace the PIP2-PIP2 RDF in the OZ equations (red curve).

The macro model is run at a large length-scale and long time-scales. Therefore, its spatial resolution must be limited. In running the macro model, third-order rectangular elements and 16 Gauss quadrature points per element were used to describe the membrane. Each element was a square with a side of 5/6 nm. The correlations functions from CG MD, however, have features at finer scales, which cannot be represented at this resolution. Therefore, the correlation functions were smoothed for better representation using our finite elements. The smoothed correlation functions were determined so that the interaction between two points is equal to the interaction that the original unsmoothed functions would yield for the source points smeared into Gaussians. The Gaussian width was 0.707 nm. Specifically, the smoothed function was calculated from the given function as

$$f_{\text{smooth}}(r) = \frac{1}{\sqrt{\omega^2\pi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f\left(\sqrt{(r+x)^2 + y^2}\right) \exp\left(-\frac{(r+x)^2 + y^2}{\omega^2}\right) dx\, dy$$

with ω = 0.707 nm.

1.2.7. Autoencoder Design for ML-based Sampling

As detailed by Bhatia et al. (2), a variational autoencoder (VAE) (73) was designed to facilitate novelty sampling of patches for CG simulations. A VAE is a type of deep neural network that learns a reduced representation of data through attempting to reconstruct the data from the reduced representation. A suitable VAE was trained prior to the MuMMI simulation using 302,000 patches generated from similarly-parameterized macro model. The VAE learned to encode a 350-dimensional (5×5×14) patch into a 15D latent space and represents the best trade-off between preserving the spatial correlations among lipid concentrations, the saliency of the resulting latent space, and computational benefits of the reduced dimensionality. A key benefit to reducing the dimensionality using an autoencoder is that the Euclidian distance in the latent space defines a more-pertinent similarity metric between patches than any pixel-wise norm --- a feature leveraged by MuMMI for performing novelty sampling.

1.3. Analytics Details

This section describes the specific details and criteria used to calculate many parameters and measurements discussed within the main text of this paper.

1.3.1. Lipid-lipid and Lipid-RAS Radial Distribution Functions

Two-dimensional (within the plane of the membrane) RDFs are calculated between RAS molecule(s) and the eight types of lipids, as well as between the lipids themselves. For the RAS protein, the F1 bead of the farnesylated cysteine is used as the reference for the RDFs. For the phospholipid species, the first bead of the Martini 'A' tail is chosen (C1A, D1A, or T1A), whereas for cholesterol the R1 bead is chosen. The same criteria are used to calculate the initial RDFs from the parameterization simulations (Section 1.2.4) as the in situ analysis (Section 1.1.6).

1.3.2. Topological Analysis of Spatial Dynamics of Lipids

Topological techniques are useful in analyzing threshold-based segmentations, i.e., connected regions of high or low value. For example, they correspond to a friends-of-friends clustering when applied to distance fields or describe regions of lipid enrichment as areas of high concentration fields. To easily explore such segmentation with respect to different thresholds and across time, and to assemble summary statistics, the *Topological Analysis of Large-Scale Simulations (TALASS)* framework (74, 75) is used to process the data. TALASS-based analysis is applied to explore lipid-dependent RAS clustering (Sections 1.3.3 and 2.2.2).

1.3.3. Lipid-dependent RAS Clustering

To investigate the relationship between local lipid composition and the clustering of RAS, we explore the distributions of average lipid densities underneath each RAS using topological techniques. In particular, the PM, as expressed by the macro simulation, is decomposed into RAS clusters. Neighborhoods of RAS are extracted and RAS clusters are defined if two or more neighborhoods overlap. In order to identify an appropriate size of neighborhoods, TALASS (see Section 1.3.2) is used to explore different neighborhood sizes and the corresponding clusters. Through this analysis, a 5 nm distance cutoff was chosen. Next, these RAS clusters are described by the average densities of the 8 inner leaflet lipids as well as the number of RAS within the cluster.

To explore whether different lipid compositions encourage or discourage the clustering behavior, a two-step process based on Function Preserving Projections (FPP) (76) is used. First, the concentrations of the 8 inner leaflet lipids are extracted for all neighborhoods of RAS captured from the macro model. Each set of concentrations is labeled with the corresponding number of RAS within the selected neighborhood. Lipid-dependent clustering is then phrased as a regression function from 8-dimensional space representing the densities of the eight inner leaflet lipid types to the number of RAS proteins. FPP is designed to find optimal linear projections, such that a given function appears as a simple, low-order signal in the projected space. This approach can produce a two-dimensional embedding of the patches that focuses on preserving the relationship between the number of RAS proteins and the lipid composition.

1.3.4. Lipid-dependent RAS State Prediction

A key step in exploring the relationship between lipids and RAS is to understand whether the state of RAS can be predicted using the lipid concentrations. We remind the reader that the macro model distinguishes between two RAS states (Section 1.2.5), whereas the CG simulations allowed us to refine this parameterization into three states (Fig. 5). For the purpose of state prediction, analysis is performed on data from both the macro model and the CG simulations to classify RAS states from the lipid densities around RAS using supervised ML.

In the case of the macro model, the simulation provides, for about 7000 time steps, the lipid concentrations for the 14 types of lipids (both the 8 inner and 6 outer leaflet lipid types) as well as the state of each of the 300 RAS proteins. To construct the ML dataset, regions spanning 30×30 nm^2 regions on the PM centered on the RAS are extracted and expressed as a 36×36 grid. These regions occupy the same physical area as a patch, but they are resolved at the native resolution of the macro model's grid. To predict the state of individual RAS, only the regions with a single RAS protein are considered. Each data sample, thus, consists of a $36\times36\times14$ data grid of lipid concentration values and a label that denotes the state of the RAS. To understand the relationship between different lipid species and RAS states, we develop several ML models to predict the states using individual or subset of all lipids. During training, the trajectories for 90% of the qualified RAS proteins are randomly selected as a training set, and the remaining 10% are assigned to the test set.

The supervised ML model for the macro model data consists of two consecutive convolutional layers, each with 5×5 kernels and 20 features, and a 2×2 max-pooling layer, then two consecutive convolutional layers with 3×3 kernels and 20 features and a 2×2 max-pooling layer, followed by a 50D dense layer, a 10D dense layer, and terminating with a 2-way softmax. Each of the dense layers uses ReLU activation and is followed by a 50%-dropout layer. Variations with more or fewer convolutional layers, dense layers, features, kernel sizes, and maxpooling layers were also explored.

To explore CG simulations, the resulting molecular data is converted into a format that is more amenable to ML. First, all CG simulation frames are translated to a consistent frame of reference by bringing the RAS C185 backbone bead to the center of the simulation frame. Next, the center of mass of the RAS G-domain is aligned with a unique direction across all frames (the positive x direction) by rotating the RAS and the PM about the normal to the membrane surface. Finally, the lipid positions are used to define lipid concentrations through the kernel density estimation (KDE) approach, where the positional coordinates are convolved with a Gaussian kernel. Using a kernel with variance of 1 nm^2, 13×13 concentration grids are generated, spanning 10×10 nm^2 areas (matching the native resolution of the macro model) for all 14 types of lipids (see Fig. S10). To identify the state of RAS, its tilt and rotation angles relative to the PM are used to define the likelihood of being in any of the three states using the HMM approach described in Section 1.2.5.

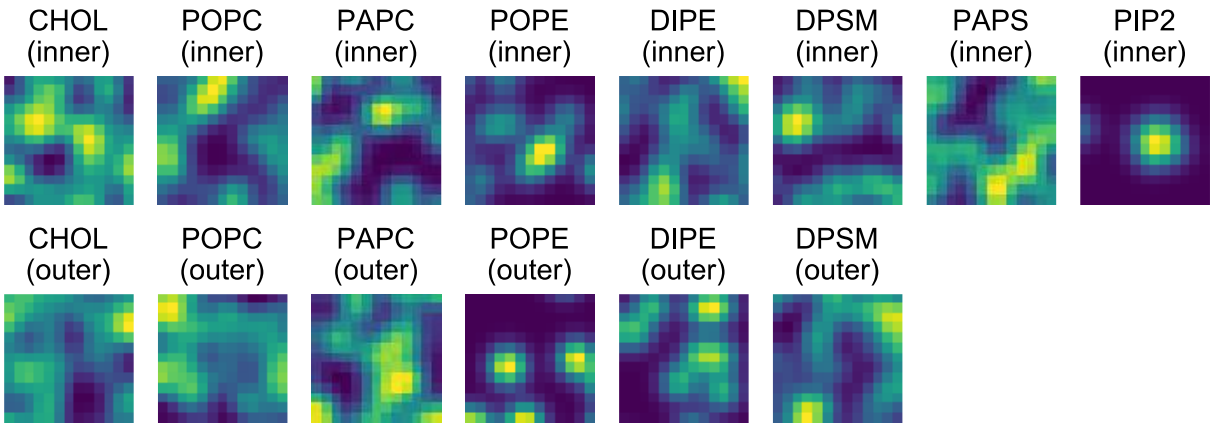

Fig. S10: Representation of CG data as concentration fields. Supervised ML is applied to the CG data by converting the positional coordinates of lipids (shown as red dots) into concentration fields (color mapped from blue to yellow) using KDE. The resulting concentrations are represented as 13×13 grids for all 14 types of lipid species (8 inner leaflet and 6 outer leaflet lipids).

As in the case of macro model, only the CG simulations with a single RAS are chosen. Each simulation consists of ~500 to ~1200 time steps, spanning 1000 to 2400 ns. For each dataset, 10% of the CG simulations are randomly selected as test data, 20% as validation data, and the remaining 70% are used as training data.

The supervised ML model for CG simulations is trained for the three-state RAS configurations. A convolutional neural network model is trained as a supervised image classification problem. The model consists of two convolutional layers each with 3×3 kernels and 20 features, followed by a 100D dense layer, a 20D dense layer, and terminating with a 2- or 3-way softmax. Each of the fully-connected layers is followed by a 50%-dropout layer.

All models are trained for 20 epochs using categorical cross-entropy loss and rmsprop optimizer. A data parallel approach was adopted to train the model since the total size of the dataset exceeds the memory capacity of a modest computational resource. In particular, the training data is distributed across several computational nodes (ranging between 8 and 16, depending on the number of lipid channels chosen), with each node training a copy of the same model. Such an approach is realized by defining the same model, initialized with similar weights, on all nodes. At the end of each training epoch, a reduction operation is performed to average over the gradients across all nodes (different data), and the averaged gradients are used to update the weights of all the models on all nodes. This data parallel approach was employed as implemented in the Horovod framework with the models constructed using the Keras framework (77) included with TensorFlow v1.12, and the training was performed on 4 NVIDIA Volta 100 GPUs per node.

1.3.5. RAS Effector Binding

To evaluate the orientational dependence of RAS-RAF binding inhibition, we combine CG simulation snapshots and a RAS-RAF crystal structure to identify residues in RAF's RAS binding domain (RBD) that would overlap with membrane lipids in a RAS-RAF complex and thereby impede RAF binding. Specifically, we orient a CG version (transformed into a Martini

representation using *martinize.py* v2.6 as described for RAS in Section 1.2.3) of the co-crystallized complex of H-RAS with the RBD of C-RAF (PDB: 4G0N) (78) to minimize the sum of squared displacement between G-domain backbone beads of residues T2-N26, Y40-L56, and G75-K165 in crystallized H-RAS and CG-simulated KRAS4b (K165 in KRAS is Q165 in HRAS). We then count the number of backbone beads in the RAF RBD that are closer than 1.8 nm to the global bilayer center along its normal, N_{clash}. To allow for the possible existence of limited membrane accommodation, we define a CG configuration of RAS to be membrane occluded for RAF binding when $N_{\text{clash}}>5$.

1.3.6. Lipid Space and Time Correlations

For the macro model, the correlations were calculated by discretizing a 1000×1000 nm^2 plane into a 1200×1200 grid. For the micro model (CG simulations) the 30×30 nm^2 area was discretized into a 14×14 grid. Correlation was measured as $C(x) = \frac{\langle \delta\rho(0)\delta\rho(x) \rangle}{\sigma_0 \sigma_x}$, where x is time (t) for autocorrelation and is radial distance (r) for spatial correlation measurements. $\delta\rho(x)$ is the difference between the local density of the lipid, $\rho(x)$, and its global average $\bar{\rho}$. σ_0 and σ_x are the standard deviation of density fluctuations at 0 and at x. To measure autocorrelation, the covariance was averaged over all the grid points. To measure spatial correlation, the covariance was averaged over multiple independent samples of the configurations. For the CG simulations, the samples were chosen at 100 ns to 1 μs with 50 ns interval using the average lipid density within the interval for each of the different independent simulations. For the macro model, the samples were chosen by picking 2000 consecutive time points starting at 3.826 μs and ending at 7.826 μs.

1.3.7. Preferential Binding Coefficients

Preferential binding coefficients of lipids to RAS, d_{Lipid}, are computed for each inner-leaflet lipid type from the 2,037 CG simulations that meet the following criteria: (i) the patch has two RAS proteins, (ii) the Cartesian xy (global membrane plane) component of the initial intermolecular backbone-backbone bead distance is >4.5 nm (see Section 2.9), and (iii) at least one frame in the simulation exhibits RAS-RAS contact (d_{min}<0.6 nm). Values of d_{Lipid} are computed according to

$$\delta_{\text{Lipid}} = \langle C_{\text{Lipid}} - C_{\text{other}} \times \left(N_{\text{Lipid}}/N_{\text{other}}\right) \rangle,$$

where N_{Lipid} and N_{other} are the number of lipids of the lipid species of interest and the number of other lipid molecules in the CG patch, respectively, C_{Lipid} and C_{other} are the number of lipids of the lipid species of interest and the number of other lipids within 1 nm of RAS (closest approach of all lipid-protein bead combinations, including farnesyl beads with RAS), respectively, and angular brackets denote averaging over independent snapshots. Only lipids from the inner leaflet are considered in the evaluation of d_{Lipid}. Leaflet selection is accomplished by using only those molecules whose PO4 bead (lipids) or ROH bead (cholesterol) is on the same side of the bilayer's center of mass along its global normal as RAS, evaluated per-frame. Values of d_{Lipid} are computed separately for each value of N_{Lipid}. Two RAS molecules are defined to be a dimer in frames where the minimum intermolecular distance between protein beads, d_{min}, is less than 0.6 nm, and are defined as monomers otherwise. d_{Lipid} values for all lipid species to RAS monomers and RAS dimers are shown in Section 2.8.5.

To quantify the influence of lipid concentration on RAS dimerization, we fit separate linear functions of the form $d_{\text{Lipid}} = m \acute{} N_{\text{Lipid}} + b$ to the preferential binding coefficient profiles of

monomers and dimers. To reduce noise, this fitting excludes data with N_{Lipid} values corresponding to the lowest and highest 10% of the sampling. Subsequently, the impact of each lipid type on the free energy of RAS dimerization is computed as $\Delta\Delta G = -k_B \cdot T \cdot (m_{\text{dimer}} - m_{\text{monomer}})$, for Boltzmann constant k_B and absolute temperature T = 310 K. To estimate the uncertainty of the fitted function, simulations are divided into three interleaved groups, fitted separately, and the uncertainty of m is defined by the standard error of its three fitted values.

1.4. Experimental Methods

1.4.1. Cell Culture, Transfection and Labeling of HaloTag-Ras

HeLa cells (ATCC) were cultured in phenol red-free Dulbecco's modified Eagle's medium (DMEM) supplemented with 10% FBS. HeLa cells were seeded at a density of 2×10^5 cells per well in six well culture plates. Transfection was performed 24 hrs. after seeding with HaloTag fusion construct of KRAS4b using Fugene (Promega) reagent and 1.5 µg DNA per well. Next day, cells were then transferred on to ultraclean 24 mm glass coverslips (#1.5, plasma-cleaned) in 6-well culture plates and allowed to grow for another 24 hrs. On the day of imaging, cells were labeled with 25 pM fluorescent JF646 HaloTag ligand for 20-25 mins within incubator, followed by multiple washes with phosphate buffer saline to remove unbound ligands. JF646 HaloTag ligand, which is highly photostable and covalently binds to the HaloTag-RAS molecules, was obtained from Dr. Luke Lavis at (HHMI, Janelia Farm, Ashburn, VA) (79).

1.4.2. Single Molecule Microscopy of Live Cells

Cultured and labeled HeLa cells were imaged on a Nikon N-STORM microscope (Nikon, Japan) equipped with an APO ×100 TIRF (Total Internal Reflection Fluorescence) objective (1.49 NA). The cells were maintained at 37°C and 5% CO_2 in a Tokai hit stage incubator (Tokai Hit Co., Ltd, Japan). Halo-KRAS4b proteins were labeled with chloroalkane JF646 dyes (which covalently links to Halo proteins) and the membrane associated Halo-KRAS4b molecules were illuminated under TIRF mode using the 647 nm laser line. Nikon software (NIS- Elements AR 4.4) was used to change the laser angle to achieve TIRF illumination. A thermoelectric-cooled EMCCD camera with 16 µm pixel size was used to capture and record fluorescent signals (iXon Ultra DU-897, Andor Technologies, USA). A region of interest covering an area of 16×16 μm^2 was chosen in the lamellipodia of the cell because it is flat to avoid artefacts due to deformities in the cell membrane. Continuous illumination of the cells at a 10 ms exposure rate for up to 1000 frames was used to collect single molecule tracks. Membrane bound molecules appear as transient, diffraction-limited fluorescence spots.

1.4.3. Single Molecule Tracking Data Processing

Time lapse movies of single molecules moving in the membrane were analyzed by an ImageJ-based single molecule tracking plugin, TrackMate, to create tracks (80). A point spread function (PSF) of 0.5 µm was used to identify single molecules; and furthermore, sub-diffraction limited resolution localization was achieved by using a 2D Laplacian of Gaussian (LoG) fit function for estimating the position of each PSF in each frame. Single molecules between frames were linked into tracks by thresholding criteria and cut off values, and the single molecule spot detection and tracking parameters were kept consistent across all experiments. These tracks were organized

and exported for InferenceMAP software (81) using a semi-automated workflow developed in Matlab (Mathwork, Natick, MA), on a multi-core Mac Pro.

1.4.4. Spatial Mapping of KRAS4b

Spatial maps of diffusivity (Fig. 1a) were obtained using the InferenceMAP software (81) based on Bayesian inference, considering a physical model of diffusion in a potential field. The analysed areas were partitioned into small regions of variable size by Voronoi tessellation (81) and presented as heat map that corresponds to the diffusion co-efficient (indicated in the colorbar, Fig. 1a).

1.4.5. Cloning, Expression, and Purification of Wild-type KRAS

Gateway Entry clones for *E. coli* produced KRAS4b (1-169) was generated by standard cloning methods and incorporate an upstream tobacco etch virus (TEV) protease cleavage site followed by the KRAS. Sequence validated Entry clones were sub-cloned into pDest-566, a Gateway Destination vector containing a His6 and maltose-binding protein (MBP) tag to produce the final *E. coli* expression clones (82). The BL21 STAR (rne131) *E. coli* strain containing the DE3 lysogen and rare tRNAs (pRare plasmid CmR) was transformed with the expression plasmid (His6-MBP-TEV-KRAS, AmpR). The expression and purification of wild-type KRAS was carried out using the procedure described previously (59). Briefly, the expressed protein of the form His6-MBP-TEV-KRAS was purified from clarified lysates by IMAC, treated with His6-TEV protease to release the target protein, and the target protein separated from other components of the TEV protease reaction by the second round of IMAC. Positive fractions were pooled, the pools concentrated to an appropriate volume for injection onto a 26/60 Superdex S-75 (GE Healthcare) column equilibrated and run in 20 mM HEPES, pH 7.3, 150 mM NaCl, 2 mM $MgCl_2$ and 1 mM TCEP. The peak fractions containing pure protein were pooled, flash-frozen in liquid nitrogen and stored at -80 ºC.

1.4.6. Nucleotide Exchange, Crystallization, and Structure Determination of GMPPNP-bound Wild-type KRAS

To crystallize active KRAS (1-169) bound to non-hydrolysable GTP analog, GMPPNP, we carried out nucleotide exchange to replace GDP with GMPPNP using the protocol described previously (59). Crystallization screenings were carried out using the sitting-drop vapor diffusion method using sparse matrix screens. The initial hits obtained from screening were further optimized. The best diffracting crystals of wild-type KRAS bound to GMPPNP and Mg were obtained in crystallization condition consisting of 100 mM Tris pH 8.5, 32% PEG 4000, 800 mM LiCl and 100 mM $MgCl_2$. Crystals were harvested for data collection and cryoprotected with a 25% (v/v) solution of ethylene glycol in the crystallization condition, before being flash-cooled in liquid nitrogen. The diffraction data set was collected on 21-ID-F beamline at the Advanced Photon Source (APS), Argonne National Laboratory. Crystallographic datasets were integrated and scaled using XDS (83). The crystal parameters and the data collection statistics are summarized in Supplementary Appendix A.

1.4.7. Protein Production of Prenylated KRAS

Cloning, expression, and purification of fully processed wild type prenylated KRAS (GG-Hs.KRAS4b (2-185-FMe)) and mutated prenylated KRAS (GG-Hs.KRAS4b(2-185-FMe)

S106C/C118S) protein was described before (84). The mutations C118S was chosen to block the native cysteine in the protein and S106C was created to provide an artificial site accessible for site-specific fluorescent labeling via maleimide chemistry. The mutations were introduced via site-directed mutagenesis (QuikChange -Agilent). Final protein purity was verified by SDS-PAGE and ESI-MS analysis indicated that 78% of the purified KRAS4b S106C/C118S protein was farnesylated-methylated, while the remaining 22% was farnesylated only. No unlipidated species were observed by ESI-MS. Prior to fluorescence experiments, KRAS4b S106C/C118S was labeled with Alexa Fluor 647 C2 maleimide dye (ThermoFisher Scientific) for fluorescence lifetime correlation spectroscopy experiments and Janelia Fluor 646, Maleimide (Tocris, MN) for TIRF single particle tracking experiments.

1.4.8. Liposome Preparation

The lipids: 1-palmitoyl-2-oleoyl-*sn*-glycero-3-phosphocholine (POPC), 1-palmitoyl-2-arachidonoyl-*sn*-glycero-3-phosphocholine (PAPC), 1-palmitoyl-2-oleoyl-*sn*-glycero-3-phosphoethanolamine (POPE), 1,2-dilinoleoyl-*sn*-glycero-3-phosphoethanolamine (DIPE), N-stearoyl-D-erythro-sphingosylphosphorylcholine (DPSM), 1-palmitoyl-2-arachidonoyl-*sn*-glycero-3-phosphatidylserine (PAPS), L-α-phosphatidylinositol-4,5-bisphosphate (Brain PI(4,5)P2) and Cholesterol were purchased from Avanti Polar Lipids, Inc. (Alabaster, AL) and used without further purification. The desired volume of each lipid was aliquoted from the stock solution using the molar ratios described in Table S4.

1.4.9. Supported Lipid Bilayer Preparation

The supported lipid bilayers (SLB) were prepared on a glass coverslip using the vesicle fusion technique (85). Borosilicate glass coverslips (#1.5, 40 mm German Degas 263 purchased from Bioptechs Inc., Butler, PA) were subjected to a rigorous cleaning procedure prior to use. Briefly, the coverslips were first sonicated in 200 proof ethanol for 30 minutes followed by base etching in 1% Hellmanex III solution (Sigma Aldrich, USA) for at least 3 hours. The coverslips were thoroughly cleansed with copious amount of ultrapure water and again sonicated in 200% ethanol for another 30 minutes before placing them in a plasma cleaner overnight. The plasma cleaned coverslips were used immediately. 2 μL of liposome sample was spread onto a clean glass coverslip and assembled in a FCS2 flow cell chamber (Bioptechs Inc., Butler, PA) and incubated at room temperature for at least half an hour. The extra uncollapsed vesicles were washed off by flowing at least 10 mL of 20 mM Hepes, 200 mM NaCl buffer at pH 7.4. For samples containing RAS, about 700 μL of 1 μM unlabeled full length farnesylated and methylated KRAS4b combined with 50 nM of JF646 labeled KRAS4b S106C/C118S was flowed through the flow cell, incubated for at least an hour at room temperature and then washed off with Hepes buffer.

1.4.10. Surface Plasmon Resonance Spectroscopy Experiments

Surface plasmon resonance (SPR) binding experiments were performed on a Biacore S200 Instrument from (GE Healthcare). The temperature was 25 °C for all experiments. The binding of KRAS4b to liposomes of different composition were carried out as follows. The Series 5 sensor chip L1 (GE Healthcare) surface was activated with three injections of 20 mM CHAPS at a flow rate of 30 μL/min. 5 mM of the HRC, ARC and LRC liposomes were captured on flow cells 2, 3 and 4 respectively at a flow rate of 5 μL/min. Flow cell 1 was used for referencing

purposes. The capture response unit (RU) values were 4000 RU for HRC, 3500 RU for ARC and 8000 RU for the LRC liposomes. After capture, a series of buffer injections were performed in the running buffer 20 mM Hepes, pH 7.2, 150 mM NaCl and 1 mM $MgCl_2$ to establish a stable baseline. KRAS4b was diluted in running buffer from 60 – 0.05 μM and injected onto the captured liposomes from the lowest to the highest concentration at a flow rate of 30 μL/min. Association response data were collected for KRAS4b to the liposomes for 120 s. Dissociation response data were collected for 900 s. The Series 5 sensor chip L1 was regenerated using 3 injections of 20 mM CHAPS at 30 μL/min. The data was double referenced by subtracting binding to the reference flow cell and buffer response using the Biaevaluation software.

1.4.11. Atomic Force Microscopy Experiments

Atomic force microscopy (AFM) experiments were carried out on an Asylum Cypher VRS Video Rate AFM (Oxford Instruments Asylum Research, Santa Barbara, CA). 80 μL of liposome samples were deposited on a freshly cleaved mica surface (grade V1, Ted Pella, Inc., CA) and incubated at room temperature for at least an hour. During incubation, the liposomes rupture and collapse onto the mica surface forming a single layer of planar supported lipid bilayer. Any uncollapsed liposomes were washed off by rinsing the sample with approximately 2 mL of 20 mM Hepes pH 7.4, 200 mM NaCl buffer. Special care was taken to make certain that the samples were always maintained under aqueous conditions throughout preparation and data collection. The SLB was imaged using a Biolever mini silicon nitride tip (BL-AC40TS) (Oxford Instruments Asylum Research) with a spring constant of 0.09 N/m under tapping mode in aqueous environment with tip-sample force of <100 pN. The images were analyzed for topography information using Gwyddion, an open source software. (86)

1.4.12. Fluorescence Lifetime Correlation Spectroscopy Experiments

Fluorescence lifetime correlation spectroscopy (FLCS) experiments were performed on an Olympus Fluoview FV1000 (IX81, 60x, 1.42 N.A oil immersion) inverted confocal microscope equipped with Picoquant LSM upgrade kit and Picoharp 300 TCSPC module. The samples were illuminated with a picosecond pulsed diode laser (LDH-D-C-640, LDH-D-TA-560 and LDH-D-C-485) with a repetition rate of 40 MHz controlled by a multichannel picosecond laser driver PDL 828-L "SEPIA II" and the fluorescence signal was detected with a PMA Hybrid detector. Data acquisition was performed with a Picoharp 300 TCSPC module in Time-Tagged Time-Resolved (TTTR) mode. The data was analysed using Picoquant's SymPhoTime 64 software. First, the fluorescence lifetime filters for each diffusing species were defined based on a multi-exponential decay curve fit followed by the autocorrelation of the filtered fluorescence intensities. The FLCS curves were fitted to a 1-component Triplet 2D diffusion model defined in SymPhoTime 64 (Picoquant, Germany) with triplet species set to zero. The size of the confocal volume was calibrated by measuring diffusion time of Rhodamine 6G dye purchased from Sigma Aldrich with known diffusion coefficient of 550 $\mu m^2/s$.

1.4.13. Single Molecule Tracking Experiments on Supported Lipid Bilayers

As in cells above, single molecule tracking (SMT) experiments were performed on the Nikon N-STORM Ti-81 inverted microscope equipped with APO ×100 1.49 N.A. oil immersion TIRF objective (Nikon, Japan) and Andor iX EMCCD camera. The samples were first photobleached using the highest power setting of the appropriate laser line and immediately followed by

acquisition of series of time lapse images up to 5000 frames in total under continuous illumination with 10 ms exposure time and zero time delay between two subsequent frames. The lipid diffusion was tracked before and after addition of KRAS4b by illuminating single molecules of ATTO550 DOPE under TIRF mode using 561 nm laser and KRAS4b diffusion was tracked by exciting JF646 dye under TIRF mode using 647 nm laser line. For each experiment, minimum of 15 TIRF movies were acquired. To reduce photosensitization-based cross-linking of RAS proteins (87), the imaging buffer for all SPT and FCS experiments included 5 mM β-mercaptoethanol.

1.4.14. Preprocessing Single Molecule Tracking Data

Igor pro software (WaveMetrics, Inc. Portland, USA) was used to read the single molecule image stack frame by frame. The embedded Localizer (88) software localized single molecules in each frame and linked them through all the frames to create single tracks from the time-lapse movies. Single molecules in each frame appear as diffraction limited patches (Point Spread Function, PSF) and super-resolution accuracy of localization was achieved with a 2D Gaussian fit to the PSF of each molecule. The detected single molecule trajectories were organized and exported for HMM analysis on a high-performance batch cluster (ABCC, FNLCR), followed by other complement analyses such as mean square displacement (MSD) plots, and single step-length distribution analysis to extract the possible information of molecular diffusion.

1.4.15. Single Molecule Tracking Analyzed by HMM Method with vbSPT Software

Single molecule tracks after extraction from each imaged area of 20×20 μm^2 on the membrane were organized into a single matlab data file to be input into the variational Bayes SPT (vbSPT) (89) software for HMM analysis. The analytical vbSPT software identifies discrete diffusive states of molecules from the single molecule trajectories and the transition rates between diffusive states during diffusion in the membrane. vbSPT analysis was initialized with three states.

1.4.16. Mean Square Displacement Analysis

The Mean Square Displacement (MSD) analysis was performed with the Matlab based TrackArt (90) software. Organized single molecule trajectories of the same type of molecules from multiple samples were input into the TrackArt to yield MSD curves with standard deviations and parameter errors, assuming two-dimensional diffusion of the particle. It serves as a qualitative tool to compare confinement of diffusion between different conditions, which provides a means of evaluating the degree of anomaly in diffusion.

1.4.17. Single Step-length Distribution

From all single molecule tracks recorded from a cell, the single jump step-length distribution was extracted, which contains information about the diffusion states of molecules. Here the distribution was used to judge the over/under inclusion of all tracks for analysis. Very short and long step-length must approach zero in probability to indicate full inclusion of tracks. At 10 ms time resolution, the maximum travel distances between 3 and 5 pixels (0.16 μm/pixel) (91) were set for linking molecules in consecutive frames, depending on the membrane composition and diffusion molecule measured.

2. Supplementary Notes and Discussion

2.1. Simulation Summary

Our simulation campaign, conducted using MuMMI, generated 119,686 independent CG MD simulations that totaled ~206 ms of simulated time. Note that CG Martini simulations have a higher effective time-scale compared to atomistic simulations, so a standard conversion factor of 4 has been proposed (10), which, in our case, would translate to ~0.8 s of sampling. All reported times in this manuscript are unscaled, except as indicated in comparison with diffusion rates. Each selected 30×30 nm^2 patch was simulated for 1.724±0.764 μs (mean ± sd) with a minimum of 1 μs. Fig. S11B shows a histogram of the lengths of simulations. These CG simulations were spawned from a single macro model simulation that explored ~150 μs of RAS-lipid dynamics on a 1×1 μm^2 membrane that contained 300 KRAS molecules. Each simulation was started using the ML-driven criteria of patch interest, irrespective of the originating macro model timestamp, and resulted in the distribution seen in Fig. S11A. The macro model simulation generated 7,481 snapshots with 2,061,900 patches, of which approximately 5.8% were chosen by ML to create corresponding CG MD simulations. Generation of the complete dataset required efficient use of ~5.6 million GPU hours and ~56 million CPU core hours for macro model, CG MD set up, and CG MD simulations. The resulting dataset contains patches comprising 1 to 4+ RAS proteins with the majority consisting of a single RAS as seen in Fig. S11C.

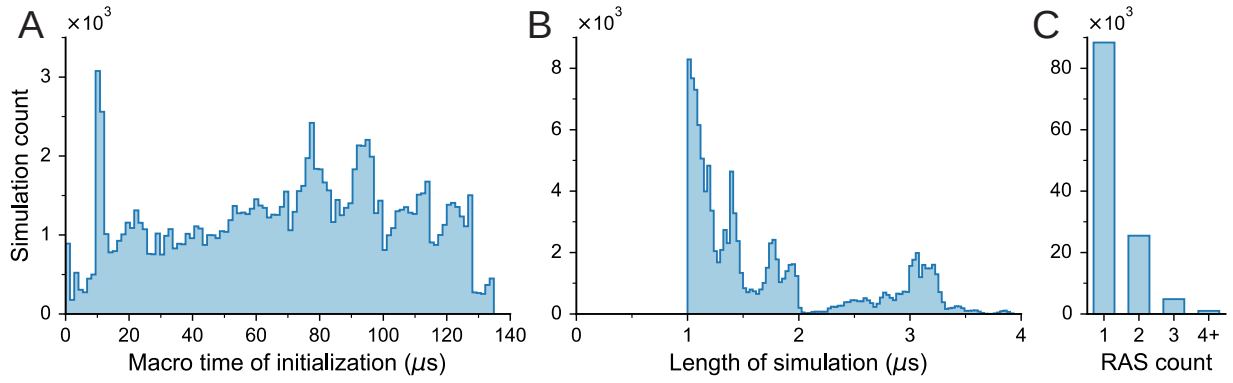

Fig. S11: Simulation summary. Summary distributions for the 119,686 CG MD simulations. (A) Histogram of CG simulations binned in accordance to their starting macro timestamp, (B) histogram of CG simulations binned by their duration, and (C) histogram of CG simulations binned by the number of RAS in the patch.

Of the 119,686 CG simulations, all of which were completely automated, only two were found to have erroneous constructions, resulting in two intertwined RAS proteins. Due to the farnesyl-centric nature of the macro model, when two RAS in close (farnesyl) proximity are converted to the micro scale, steric hindrances between RAS cause G-domains to favor initial orientations directed away from one another (see Section 2.9). Therefore, RAS that were constructed in close proximity need to be excluded for some analyses.

The simulation campaign was run in two segments. Segments 1 and 2 contained ~29K (29,191) and ~90K (90,495) CG simulations, respectively. The two segments used different macro to micro conversion procedures for lipid placement, resulting in different lipid concentrations for

low frequency lipid types. These differences and their implications are discussed in Section 2.3.2. Additionally, on average, simulations in Segment 1 are 2.2-fold longer than simulations in Segment 2. The two segments were combined for most analyses and, when indicated, CG simulations were weighted based on the original macro model patch from which they were constructed (see Section 2.3.2). As Segment 2 was a continuation of Segment 1, approximately 10K (10,407) patches were selected during the Segment 2 run that had already been selected during the Segment 1 run; thus, those two corresponding CG simulations share the weight of the single macro model patch.

For all of the ~120K CG simulations, frames were saved every 2 ns, resulting in >100M saved frames. Due to file transfer and/or disk file corruption, 113 simulations were found to contain corrupt frames; these were pruned and excluded from analysis. Overall, corrupted frames affect less than ~0.1% of all CG simulations and <0.001% of all frames. For the online analysis, Segment 1 data was analyzed every 2 ns, whereas Segment 2 data was analyzed every 0.5 ns, together resulting in ~300M online-analyzed frames.

In the following sections, we discuss the sampling observed in the macro model and several observed characteristics of RAS behavior observed from the CG MD simulations such as orientation, effector binding, and lipid dependence.

2.2. Sampling at Macro Scale

The macro model (see Section 1.1.3) is designed to explore the interactions between the lipid PM and RAS proteins over long time- and length-scales. This section illustrates how the macro model successfully enables the sampling of the space of lipid densities, RAS states, and the interplay between them – both directly from the resulting macro simulation and using ML-based sampling of the lipid configurations generated by the macro simulation.

2.2.1. Lipid Diversity in the Macro Model

The macro model enables the realization of spatial diversity in lipids in a PM both in the vicinity of and away from RAS, as well as understand the correlations between different lipid species in the PM. The results presented in this section highlight this diversity, which can be exploited to both adaptively sample lipid configurations of scientific interest using ML (see Section 2.2.3) and experimentally validate the resulting data (see Section 2.3).

In particular, Fig. S12 visualizes the 1×1 μm^2 PM simulated by the macro model, and illustrates the spatial variability in the concentration of the different types of lipids in both inner and outer membranes. Fig. S12A shows a top-down view of the full PM colored by the concentration of cholestrol, and highlights the formation of lipid domains. The figure also provides a broader context in the presence of all the lipids present in the inner leaflet (Fig. S12B) and both leaflets (Fig. S12C) of the PM for different levels of zoom.

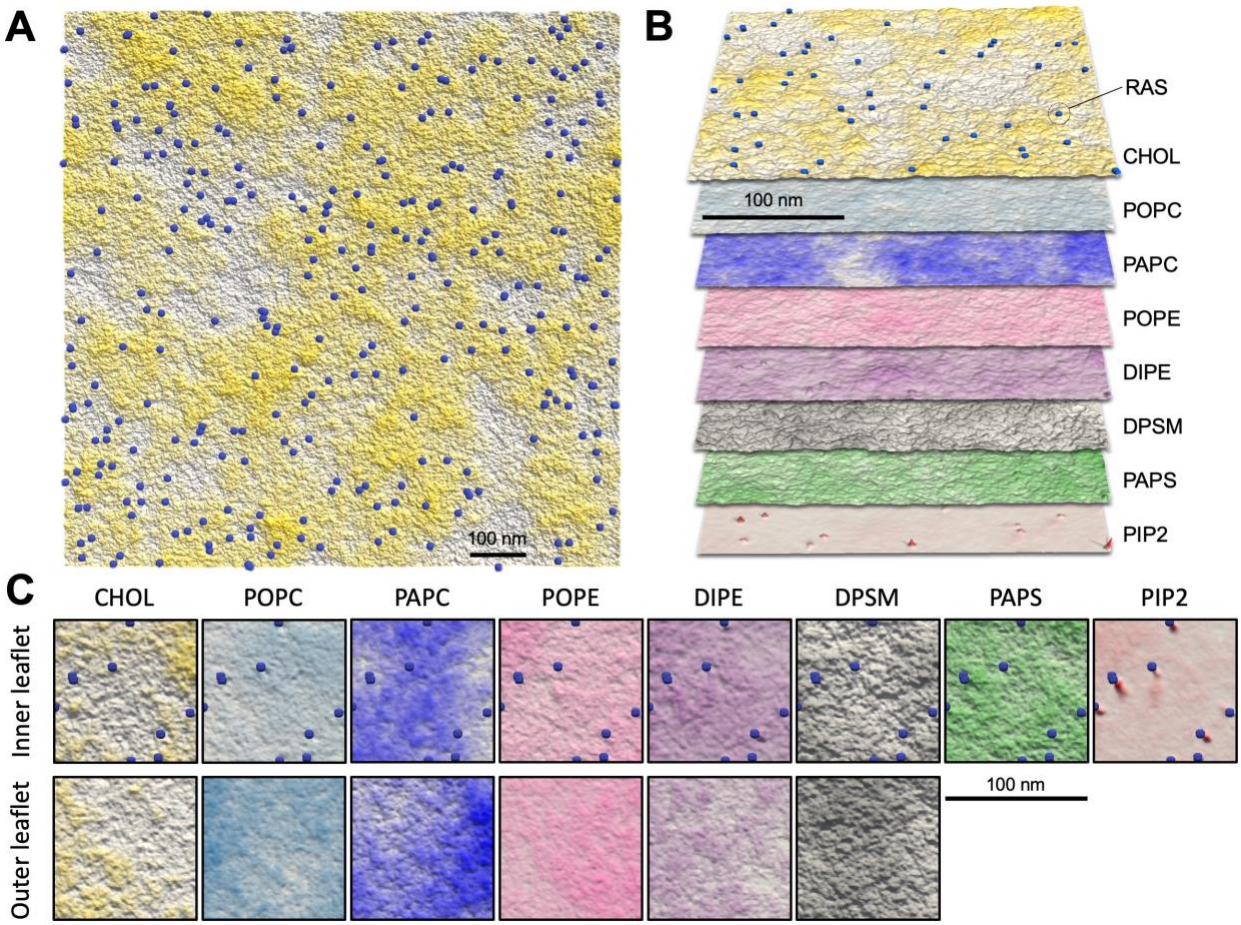

Fig. S12: Macro model lipids diversity. (A) The 1×1 μm^2 PM simulated by the macro model colored for cholesterol density, with 300 RAS molecules (blue dots). (B) Illustration of the layers of the different lipid densities over a 300×300 nm^2. (C) Example of lipid density fluctuations over a 100×100 nm^2 region for the inner and outer leaflets, with the overall inner and outer densities also shown. The color saturation of each lipid represents the density at that position of the membrane.

Focusing on the lipids in the vicinity of RAS, Fig. S13 shows the mean proportion of the different types of lipids in the inner and outer bilayers. To illustrate the influence of the presence of RAS, the figure considers 18×18 nm^2 regions around RAS only when there exists no other RAS within the considered neighborhood, and compares them against randomly selected regions of the same size but without any RAS. The comparison indicates a strong correlation between the the presence of RAS and PIP2 enhancement in the inner membrane. Other correlations are also observed, e.g., enhancement of DPSM and depletion of PAPC in the outer membrane.

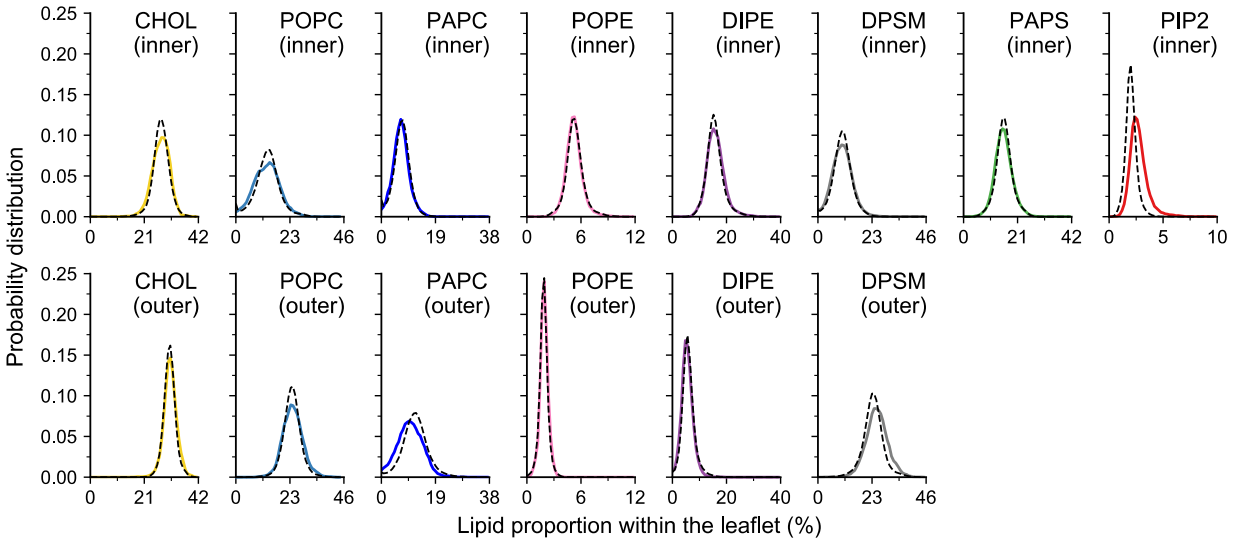

Fig. S13: Macro model lipid distributions. Distributions of the proportions of the mean lipid concentrations on the inner and outer leaflets of the PM. The distributions are computed using 18×18 nm^2 regions around RAS (colored, shaded regions) and randomly selected regions of same size without any RAS (dashed lines). The figure shows characteristic differences for lipids that are more strongly correlated with RAS, especially PIP2 with its mean lipid concentration enhanced in the vicinity of RAS.

2.2.2. RAS Aggregation

Within the macro model simulation, we next study the aggregation of RAS that can be explored by CG simulations. Therefore, we compute the enhancement of RAS with respect to uniformly random spatial distribution of (i.e., noninteracting) RAS molecules within patches.

Each patch is centered around a RAS; however, each patch can contain several RAS molecules. We compute the histogram of patches with different numbers of RAS (1 through 5) both for the macro model simulation as well as for randomly distributed RAS molecules. Fig. 2b shows the ratio of these two histograms, i.e., the frequency of *n*-RAS patches in the macro model simulation divided by the frequency of *n*-RAS patches in a uniformly random RAS distribution. The figure shows that multi-RAS patches are much more prevalent in the macro model simulation than for non-interacting molecules and that the effective RAS-RAS interactions lead them to aggregate.

Next, the correlation between RAS aggregation and individual states of RAS (the macro model uses two states, α and β) is explored using topological anlaysis. To this end, a RAS distance field is computed, which describes the distance to the nearest RAS for each grid point in the macro model simulation box. The distance field is used here because it decomposes the simulation box into regions associated with a given RAS as well as readily highlights the neighborhoods affected by multiple RAS. Next, TALASS (introduced in Section 1.3.2) is used to determine clusters in the distance field with respect to the state of the corresponding RAS. For clustering, only the RAS within 3 nm distance of each other are considered. The distribution of the resulting clustering is shown in Fig. S14. There are two trends that are evident from the figure: (1) there are more clusters with one or more RAS in state β across different extents of RAS aggregation, with the difference being about one to two orders of magnitude between the clusters dominated

by RAS in state α and those dominated by RAS in state β; and (2) the number of clusters decrease with an increase in RAS aggregation, e.g., there is about an order of magnitude decrease in the number of clusters as we go from monomers to dimers and from dimers to trimers.

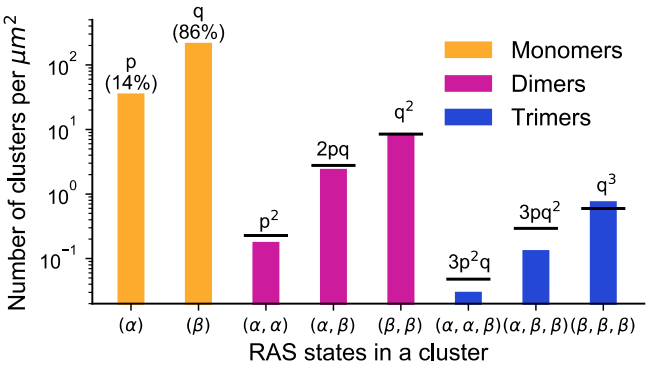

Fig. S14: RAS aggregation vs association. Histogram of clusters per unit area on RAS distance field for RAS monomers, dimers, and trimers with different combinations of RAS states. Given the probabilities of RAS existing state α and state β (p and q, respectively), the expected state distribution is labeled and marked as black lines.

2.2.3. ML-based Sampling of Macro Configurations

As stated earlier, the goal of the MuMMI framework is to sample macro lipid configurations as uniformly as possible—a task performed using ML-based, dynamic-importance sampling (2) (DynIm – see Sections 1.1.5 and 1.2.7). In this campaign, MuMMI used DynIm to select a total of 119,686 important patches from a set of 1,918,500 candidate patches. To demonstrate the significance of ML-based sampling, Fig. S15 compares the density distribution of ML-selected patches and randomly-selected patches against that of the set of all candidate patches. The figure shows five pairs of 2D marginal distribution in the 15D latent space. As expected, the random selection reproduces the original distribution and, if used to spawn CG simulations, would invest computational resources in simulating similar configurations while missing out on critical information in infrequently-occuring configurations. The ML-selection, on the other hand, is designed to reduce the selection of similar configurations and favor the selection of rare configurations. The figure shows that the density distributions for ML sampling are “flatter” (i.e., suppressed modes) as well as “wider” (i.e., capturing infrequent configurations), allowing for a more-uniform sampling. Finally, the DynIm framework allows reproducing the original density distribution by appropriately weighting the ML-selection, as highlighted in the figure.

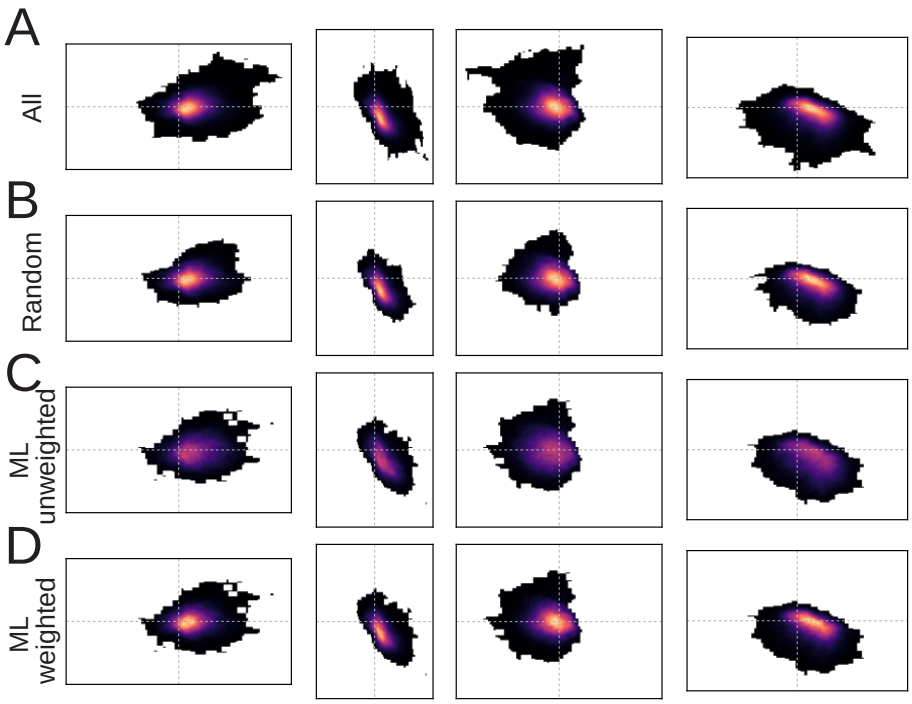

Fig. S15: Demonstration of the wider coverage of DynIm sampling. Comparison of ML-based dynamic-importance sampling (C) with a random sampling (B) of patches from a given "true" distribution of all patches (A). The figures show five pairs of marginal distributions of density in the latent space with zeros of the corresponding latent dimensions marked. The random sampling closely replicates the input distribution, whereas the ML-based sampling produces a flatter and wider distribution, indicating a wider coverage of the phase space. Both sampling approaches select the same number of patches (~5.8% of the total), and the corresponding figures are color-mapped to the same range. The figure also demonstrates that the true distribution (A) can be reconstructed (D) using the ML-based sampling and ML-based simulation weights. Additional comparisons are provided by Bhatia et al. (2).

Next, Fig. S16 shows the distribution of weights associated with each CG simulation (patch selected through ML). The histogram shows that many patches have low weights, indicating the coverage of infrequent configurations. The figure also shows a significant number of patches (hundreds of thousands) with relatively high weights (above a few tens). It is such frequently-occurring patches whose similar lipid configurations are suppressed by ML.

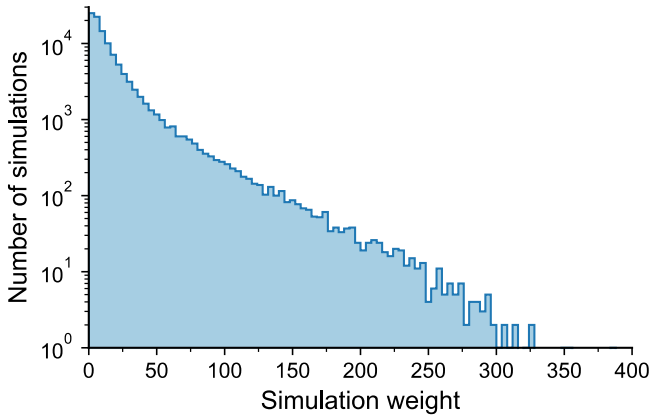

Fig. S16: Histogram of patch weights. Histogram of weights associated with CG simulations (patches selected by ML). (2).

2.2.4. On-the-fly Feedback to Macro Model Parameters

A key characteristic of the MuMMI framework is its use of in situ analysis of CG simulations to update the parameters of the macro model. In the current simulation campaign, the framework aggregates the RAS-lipid RDFs from the CG simulations and transforms them to RAS-lipid potentials. These potentials are fed to the macro model periodically, thus resulting in an updated macro model that more-closely resembles the behavior captured from CG simulations. Fig. S17 illustrates the improvement in the RAS-lipid RDFs as they approach convergence through on-the-fly feedback.

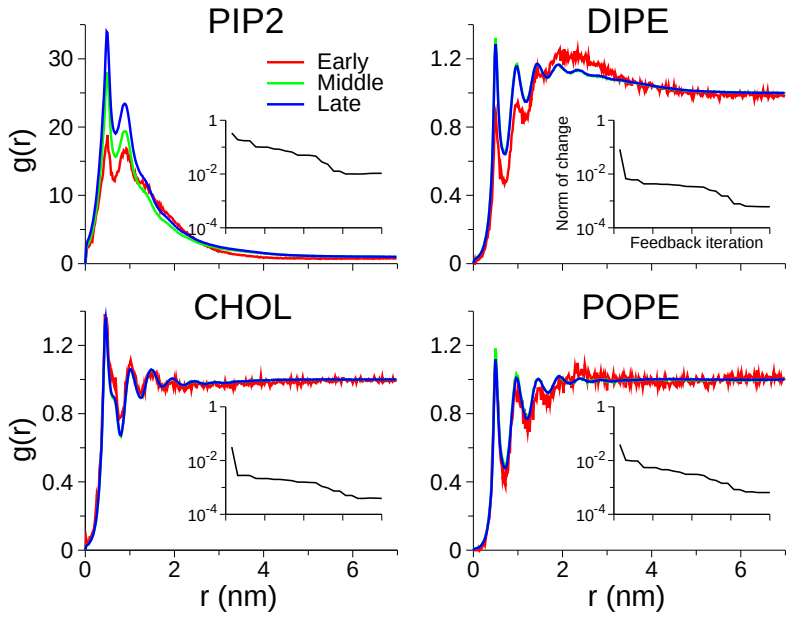

Fig. S17: Feedback and convergence. Each main panel shows the lipid-RAS β state RDF's for a particular lipid, for 3 different instances in the simulation: Shortly after the beginning, after several feedback interactions, and toward the end of the simulation. For all lipids except PIP2, the RDF's converge relatively quickly, and the green (intermediate time) lines are almost on top of the blue (late time) lines. The inset shows in log-scale the relative error (in L_2 norm) in the current compared to the final RDF's.

We see that over the course of the simulation, all lipid-RAS RDF's (and hence potentials) converge to <1%, and in most cases to all digits recorded in our output files.

2.3. The RAS-Plasma Membrane Mimic

The RAS minimal PM mimic (ARC) is an 8 component asymmetric mammalian PM mimic designed to capture lipid dynamics relevant for RAS biology, see Section 1.2.1. The macro model simulated the ARC mixture with 300 RAS molecules at a large (1×1 μm^2) length- and time-scale (>150 μs), sampling a vast distribution of possible lipid configurations of which representative patches were selected for further exploration using CG MD simulations (see Section 2.2). Here we explore the overall applicability of the ARC for forming PM-like membranes and binding RAS (Section 2.3.1) and how the overall lipid dynamics in the different lipid compositions of the smaller CG simulations compare to the full macro model simulation (Section 2.3.3).

2.3.1. Experimental Characterization of the Inner RAS-Plasma Membrane Mimic

We created the symmetric inner version of the ARC lipid mixture, see Section 1.4.5, and verified the lipid composition using HPLC technique as described in our earlier publication (92). The black spectra in Fig. S18 shows representative spectra collected for liposomes composed of the 8 lipids symmetric inner ARC. We assigned the different peaks in the HPLC spectra to the specific lipid species based on spectra collected from standard stock samples of each lipid types indicated by different colors. The spectra show that all eight different lipid types are incorporated in the system with relative intensities in overall agreement with expected concentrations, see Table S4.

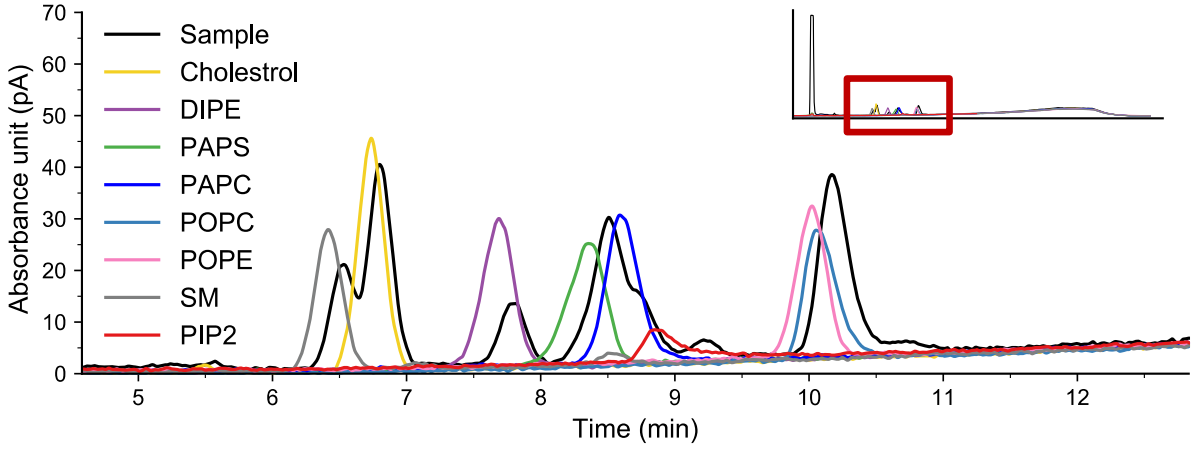

Fig. S18: HPLC chromatograms. The black line shows the spectrum of the symmetrical inner ARC 8 lipid mixture, as described in Table S4. Colored lines represent reference spectra collected from standard stock samples of each lipid type.

For an initial biophysical characterization of the 8 lipid bilayer and its interaction with RAS, we first investigated the lateral organization of the ARC 8 lipid mixture by imaging with atomic force microscopy (AFM) under the tapping mode. The topography image shown in Fig. S19A displays that the 8 lipid mixture segregates into liquid ordered and liquid disordered domains

with a height difference of approximately 1 nm between them as seen in the corresponding height profile in Fig. S19B. In order to verify that the phase separation exists because of the interaction between sphingomyelin and cholesterol included in the complex 8 lipid mixture and is not an artifact of our experimental setup, we performed AFM experiment on lipid bilayer composed of 7 lipid mixture without cholesterol and no phase separation was observed (data not shown). Next, we systematically increased the temperature from 25 °C up to 65 °C and observed significant differences in the domain features (Fig. S19C). Quantitative analysis of the domain heights (Fig. S19D) reveals a melting pattern in the ordered domain as a function of temperature. This indicates that our 8 lipid mixture is entropically driven and biologically relevant.

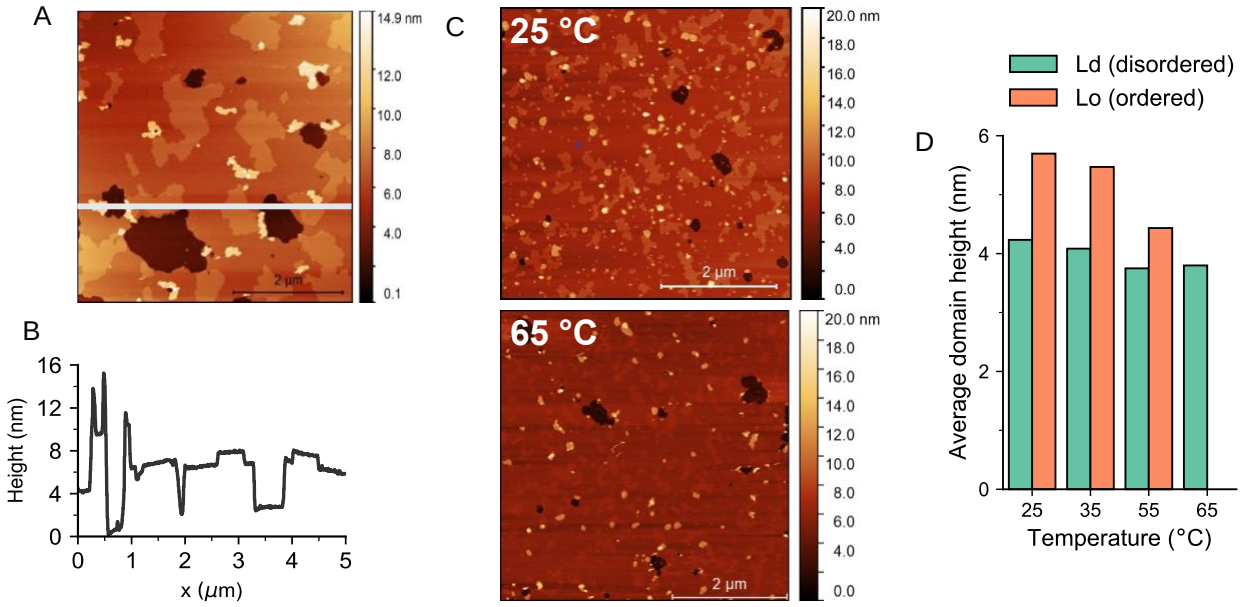

Fig. S19: AFM of the 8 lipid ARC mixture. (A) AFM image of the symmetrical inner ARC 8 lipid mixture prepared on mica surface acquired using tapping mode. (B) Height profile corresponding to the blue line marked on the top image. (C) AFM images of ARC obtained at 25 °C (top) and 65 °C (bottom). (D) Normalized histogram of the height distribution corresponding to the two AFM images on top (blue, 25 °C; red, 65 °C).

2.3.2. Macro and Micro Scale Lipid Consistency

The simulation campaign was run in two segments that used different macro to micro conversion procedures for lipid placement. The first segment consisted of ~29K (29,191) CG simulations and used rounding to convert continuous lipid concentrations to discrete numbers of lipids in each sub-grid (64 lipids). This rounding led to undersampling of lipids with small absolute values and ranges in the macro model (e.g., the small PIP2 concentration distant from RAS was further reduced). The second segment consisted of ~90K (90,495) CG simulations and probabilistically adjusted lipid placement in each sub-grid based on lipid distribution. A coding error in Segment 2 implementation resulted in all non-integer remainders being inaccurately assigned. This procedure, while resulting in improved macro model/CG agreement in mol% for many lipid types, disproportionately underrepresented rare lipids and led to a further reduction in the average number of PIP2 lipids (Fig. S20).

Conversion from continuous to discrete models necessitates a discretization which can lead to discrepancies between the models. When converting from the macro model to CG patches, the discretization was compounded with sub-grid rounding (in Segment 1), and a coding error (in Segment 2) resulted in lower sampling of low frequency lipids in each sub-grid. Fig. S20 compares the average lipid compositions in all CG patches selected in Segments 1 and 2 to their respective macro model patch compositions. The lipid compositional distributions are captured reasonably well for all inner leaflet lipids with the exception of low frequency lipids (POPE in Segment 1 and PIP2 in both segments), and for all outer leaflet lipids except POPE. The largest difference was the reduction of PIP2 concentration in the bulk lipid phase (distant from RAS where PIP2 concentration is lower). Averaged across all ML-selected patches, the inner leaflet concentration of PIP2 decreased from 2.4 mol% in the macro model to 1.7 and 0.8 mol% in Segments 1 and 2, respectively. Due to the broad tails of the PIP2 distribution and large number of simulations, both segments contain significant numbers of relatively high (above 2.2%) PIP2 concentration. Additionally, in comparison to experimentally determined concentrations for PIP2, the 0.8 mol% average inner leaflet PIP2 concentration in Segment 2 is similar to the 1.0 mol% phosphatidylinositol identified in the erythrocyte PM inner leaflet by mass spectrometry (93) and is, therefore, biologically relevant.

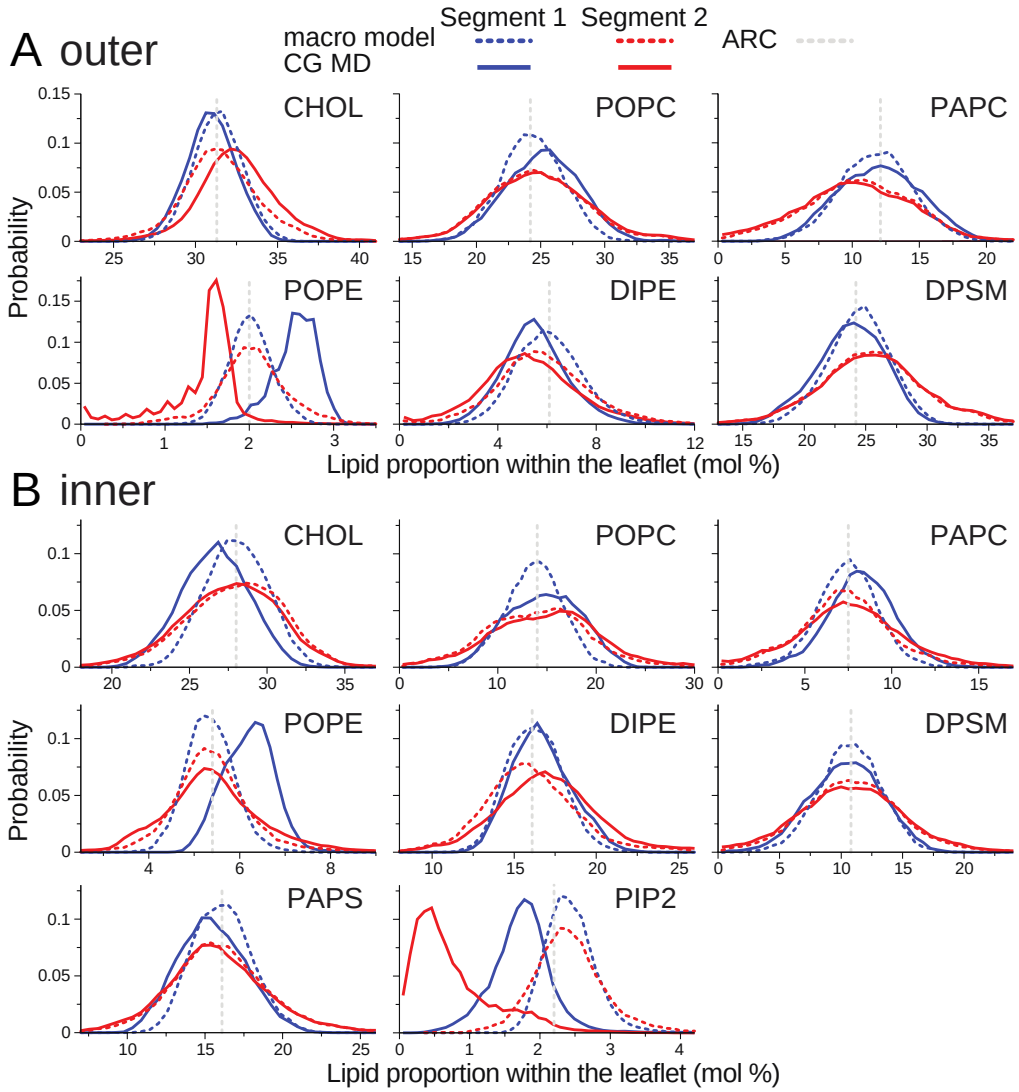

Fig. S20: Lipid concentrations in macro model and CG patches. Panels show probability distributions of the molar concentration of each outer (A) and inner (B) leaflet lipid species in macro model patches selected for CG simulation (broken lines), and CG patches (solid lines). Segments 1 and 2 are colored in blue and red, respectively. Dashed grey vertical line shows the global lipid concentation in the macro model (ARC).

Separate analysis of the two segments show similar observables in both ensembles. Two instances where small differences are found are shown in Fig. S21. The RAS G-domain adopts the same three orientational states with only subtle changes in favored orientations between Segments 1 and 2 (Fig. S21A). The fact that RAS orientation is robust to changes in average PIP2 concentration between 0.8 and 1.7 mol% is consistent with our conclusion that >2% PIP2 is required to substantially perturb the orientation of the RAS G-domain in a manner that relieves membrane-based occlusion of RAF binding (Fig. 6d and Fig. S21B).

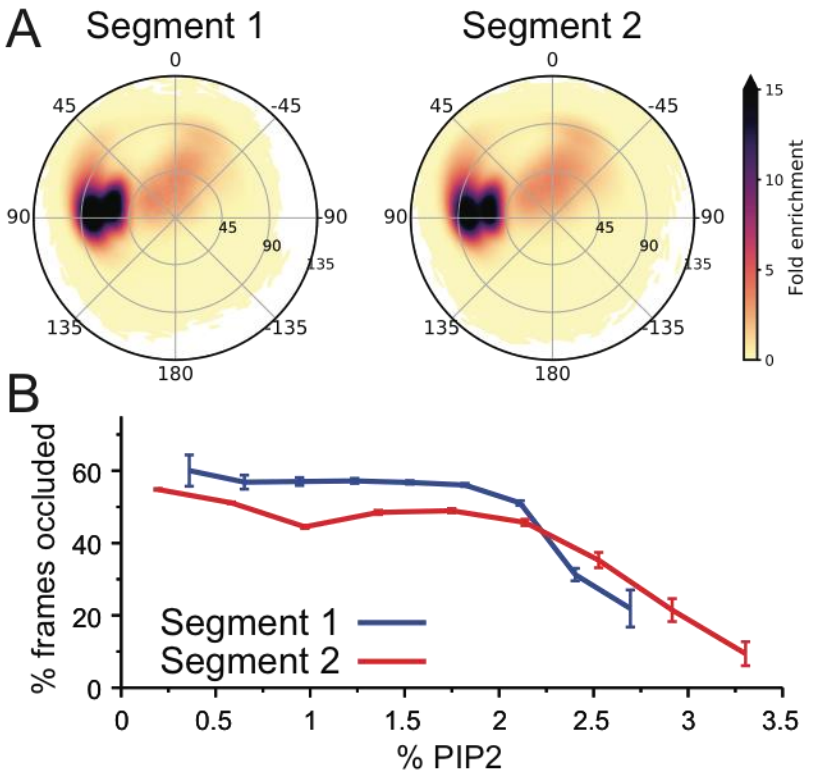

Fig. S21: Comparsion of Segments 1 and 2. Selected RAS observables in CG simulations from Segments 1 and 2. (A) G-domain disposition in simulations with one RAS. Analogous to Fig. 5c, but without patch weighting and only including data from the first 1 μs/simulation. (B) RAF-occlusion vs. PIP2 content. Analogous to Fig. 6d, but only including data from the first 1 μs/simulation.

For most analyses, Segments 1 and 2 are combined and when indicated the CG simulations are weighted based on the weight of the original macro model patch from which they were constructed. Due to the non-ideal converstion of low frequency lipids, the macro weights might be biased compaired to their CG conterparts. To evaluate the possible implications of this bias, all weighted analyses were also evaluated in an unweighted manner and, for this campaign, the weighted vs. unweighted results were similar and we therefore conclude that this bias does not affect any of our conclusions.

2.3.3. Macro and Micro Scale Lipid-Dynamics of the RAS-Plasma Membrane Mimic

The ARC mixture exhibits a large range of lipid fluctuations of various time- and length-scales both in the macro and micro simulations. Phase separation of the ARC mixture is not observed at either scale, but regions of lipid enrichment/depletion can be seen spanning simulation boxes, indicating that phase separation might be observed in longer and/or larger simulations. The macro simulation sampled a large range of local lipid compositions, resulting in broad compositional heterogeneity across the 120K unique CG MD simulations (Section 2.2). For each selected macro patch (30×30 nm^2 region), a CG simulation is created (Section 1.1.4) based on the local macro model lipid composition resolved at a 5×5 sub-grid resolution (each sub-grid ~6×6 nm^2). This mapping preserves the local lipid composition and overall/larger-scale spatial distribution of the selected macro patch.

Fig. S22 shows a selected region of the macro simulation, where MuMMI has selected two patches to be of interest; the spatial enrichment/depletion of cholesterol is shown for the macro patches and CG simulations. In both the macro and micro models the lipid dynamics are overall similar to those described for the complex mammalian PM, and its eight lipid mimic (44, 46) and described briefly in Section 1.2.1, with colocalization of more saturated lipids and cholesterol and exclusion of polyunsaturated lipids. All selected patches and, therefore, all CG simulations contain at least one RAS molecule which affects the lipid properties in its proximity (Fig. 6a and Section 2.6).

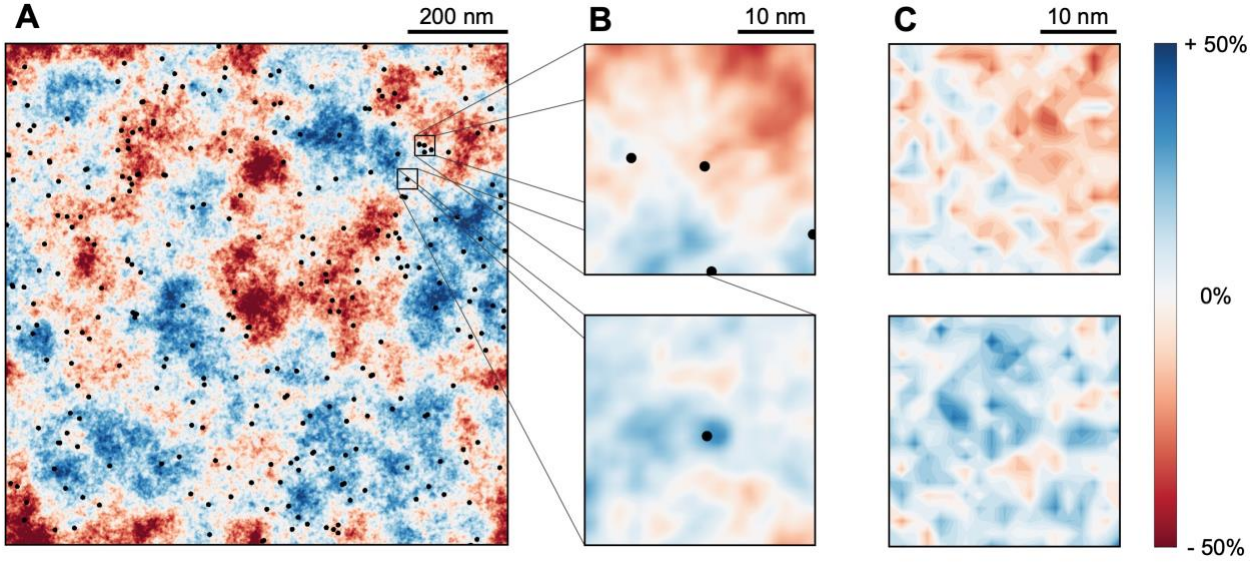

Fig. S22: Comparison of lipid densities in the macro model and CG simulation. A snapshot of the macro model inner leaflet (A), colored to show relative cholesterol density, with RAS positions indicated as black dots. The two highlighted patches (B) represent regions of both high and low average cholesterol content. The equivalent CG simulations (C) present the same distribution of cholesterol, both in terms of magnitudes and spatial arrangement.

The lipid properties are explored by calculating the lipid enrichment/depletion of each lipid type for all the lipid types (in both leaflets). Fig. S23A shows the lipid-lipid affinities in the macro model, demonstrating the various lipid-lipid affinities and nonhomogeneous lipid mixing. Fig. S23B compares the macro model lipid-lipid affinities to the micro models lipid-lipid neighbor counts in the ARC lipid mixture (44) and demonstrates the similarity between the two models. To explore the consistency between different patches with varying lipid compositions, 100 simulations are randomly selected, and the same lipid neighbor analysis is performed. Each of the 100 simulations has at least one RAS, 0.2-1 μs used for the analysis, and individually normalized to the lipid concentration in that patch. The average behavior of all 100 simulations is remarkably similar, within error, to the lipid neighbor preferences of the average ARC mixture, except for a higher PIP2-PIP2 interaction which is due to the RAS-PIP colocalization.

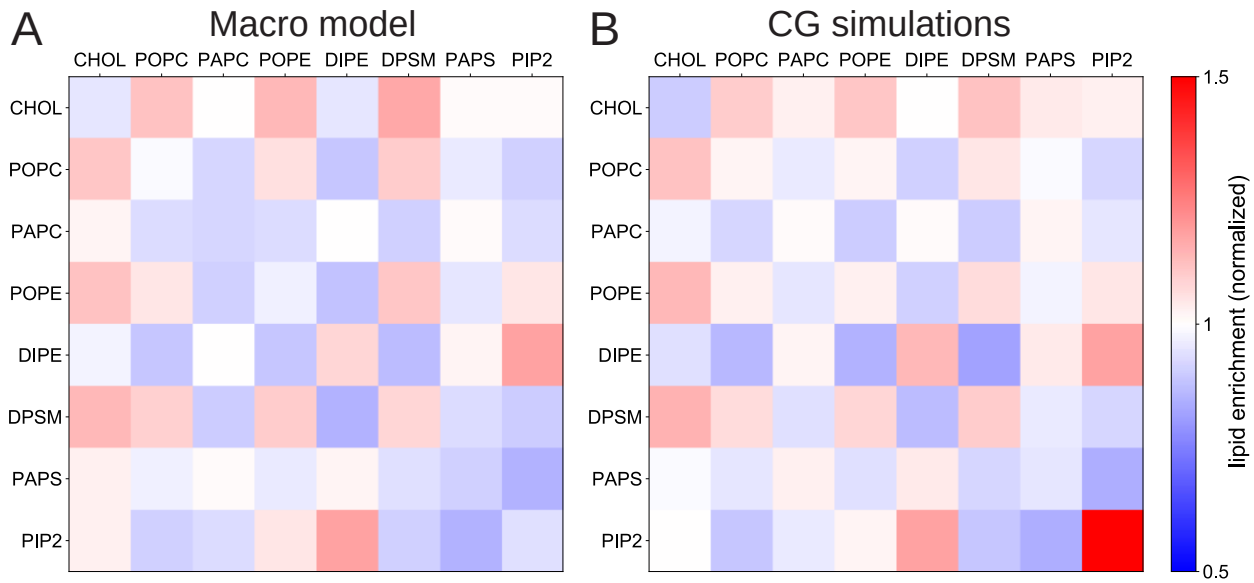

Fig. S23: Average neighbor analysis. Results for the macro model (A) and ARC from CG simulations (B) are shown. Comparison between the two reveals extremely comparable behavior. In general, the more ordered lipids are enriched around other ordered lipids, but depleted around more disordered lipids. The largest difference between the two is the PIP2-PIP2 interaction. The macro model underpredicts the number of PIP2-PIP2 neighbors within the small 1.5 nm radius by about 0.15 PIP2 neighbors, which is due in part to the scaling introduced in the PIP2-PIP2 RDF when parametrizing the macro model, Section 1.2.6, and in part to the limited resolution at which the macro model is run. The low average concentration of PIP2 amplifies this to a large relative error, which is why there is a very considerable difference in color of the macro PIP2-PIP2 relative neighbor count compared to the Martini result.

For analysis of the lipid-lipid dynamics in both models, the lipid space and time correlations are explored. Fig. S24 shows the time and space autocorrelation for cholesterol (CHOL) and DIPE in the macro model and the average space/time autocorrelation over the same 100 random simulations as above; the correlations were calculated as described in Section 1.3.6. For cholesterol, the autocorrelation functions are nonexponential at short times, with exponential tails at long times, as the inset shows. The spatial correlation function show similar behavior with nonexponential decay at short distances and exponential decay at longer distances. The nonexponential decays are not algebraic decays, as we have verified by plotting the correlation functions in log-log plots (not shown). Although the correlation functions show qualitatively similar behavior, there is significant quantitative differences. For example, the autocorrelation function for the micro model decays to zero much before 1 μs, but for the macro model, the autocorrelation function does not decay to zero even after 2 μs, indicative of the longer range correlations in the larger macro model. We observe similar behavior for DIPE as well. For DIPE, the macro model autocorrelation function for the upper leaflet (dashed line) is significantly different from the macro model autocorrelation function for the lower leaflet (solid line).

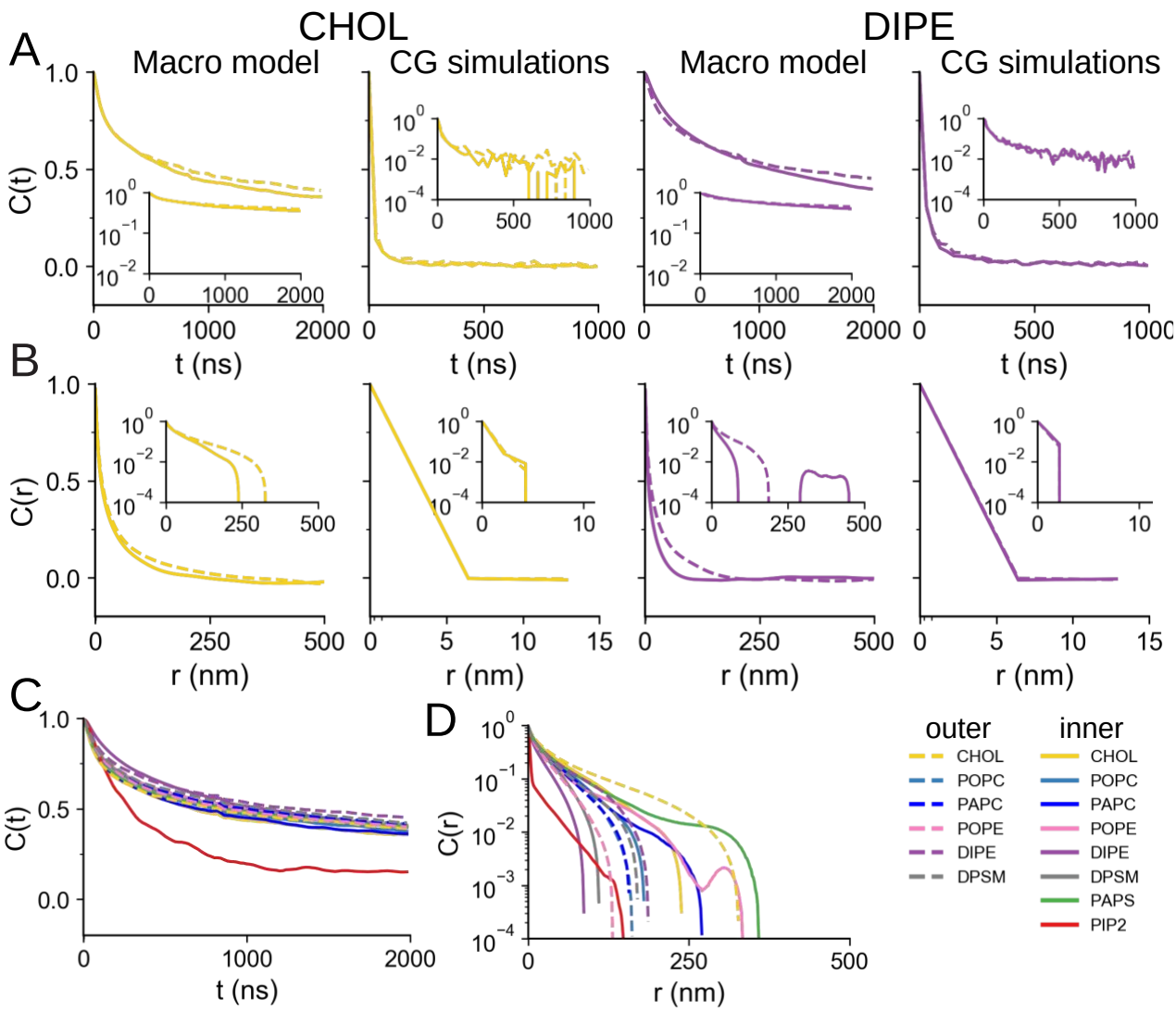

Fig. S24: Lipid space and time correlations in the macro and micro models. Comparison of autocorrelation (A) and spatial correlations (B) calculated from the macro model and the micro model (CG simulations). Inset in each panel shows the variation of the correlation in log-linear plots. (C) Plots of autocorrelation functions for all lipids (same-same) using macromodel. All lipids show decay patterns similar to CHOL and DIPE (shown in A), except PIP2, which decays much faster than the other lipids. (D) Log-linear plots of spatial correlation functions for all lipids (same-same) using the macro model. The decay pattern is clustered in three groups with no particular preference for inner or outer lipids.

2.4. RAS State Analysis

As discussed in Section 1.2.5, our training system consisting of RAS in the 8 lipid ARC can be described adequately by two metastable states. We repeat the same procedure on results generated from our novel ML-based importance sampling approach, with over two orders of magnitude more simulations and, instead of a fixed average ARC, a large ensemble of relevant lipid compositions sampled by the macro model, see Section 2.2. The HMM analysis was performed on the tilting and rotation angles of RAS in 88,392 patches containing a single RAS

protein. The tilting rotation space was transformed into 2000 microstates using the *k*-means clustering method, see Section 1.2.5. Fig. S25A shows the population map, and Fig. S26 shows the implied time-scales plot; as can be seen, there is a separation of time-scales between the first (solid blue line) and the second relaxation time-scale (solid red line). Subsequently, we obtain a maximum likelihood HMM using two states and a lag time of 100 ns. The two macrostates can be seen in Fig. S25B. However, adding a third state (in HMM) definition splits the region previously defined in Fig. S25B into two sub states, each one having a significant population (29±1 % and 40±1 %), as seen in Fig. S25C. Importantly, this result clearly highlights the benefit of our ML-based sampling in revealing hidden protein configurations that are not sampled properly in the training data.

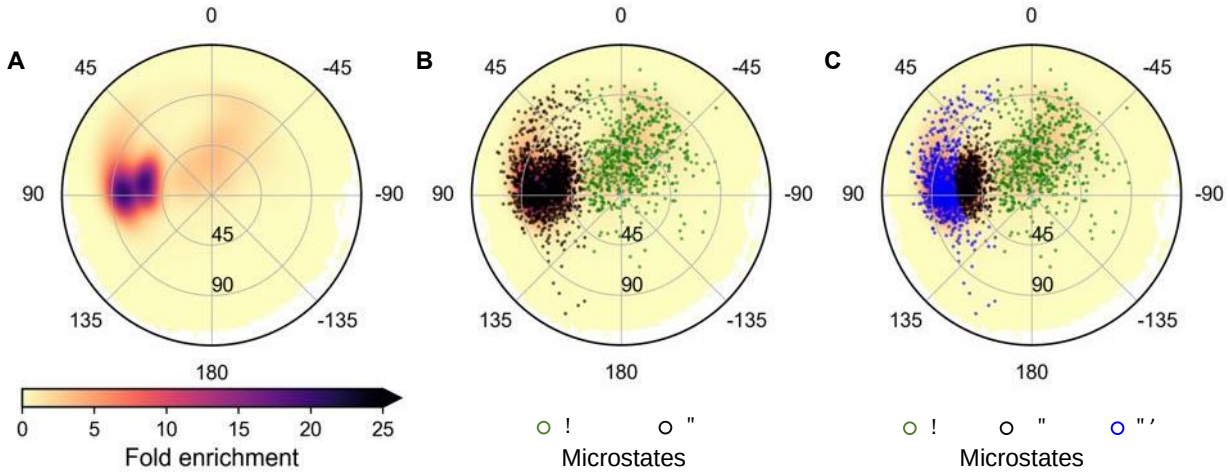

Fig. S25: RAS tilt/rotation states. (A) Population map in tilt-rotation subspace. (B) Two metastable states obtained from HMM. The microstates are colored according to the macrostates they belong to. The populations of the states are β: 65%, α: 35%. (C) Three metastable states obtained from HMM. The microstates are colored according to the macrostates they belong to. The populations of the states are: β: 29%, β': 40%, α: 31%.

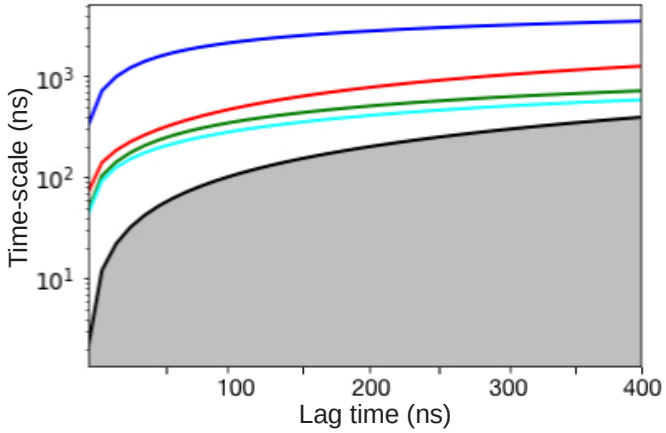

Fig. S26: Implied time scales. Relaxation times calculated as a function of lag time are shown in different colors. Any relaxation process below the black line (y=x) cannot be reliably estimated as the time-scales of these processes have already decayed.

2.5. RAS Orientation and Effector Binding

Patch-weighted sampling of G-domain orientations in CG MD simulations with one RAS per patch favor 45°-90° tilting with rotation angles between 50° and 110° (Fig. 5c). This represents the β and β' states (Fig. 5b), which bring β strands 1-3 and switch I toward the membrane so as to occlude the G-domain's RAF binding interface (Figs. 5f, 5g). This occlusion is especially apparent at the larger tilting angles that characterize the β' state in comparison to the β state (Figs. 5f, 5g). The balance between β and β' states appears to be strongly influenced by the extent of HVR-membrane association (Figs. 5i, 5j). This correlation suggests that kinetic separation of β and β' states may arise from opposing tendencies for extensive tilting toward switch I and transient fluctuation of the G-domain proximal end of the HVR away from the bilayer surface. Conversely, RAS' RAF binding interface remains accessible when the G-domain is oriented such that α helix 5 is perpendicular to the global bilayer normal (Fig. 5d) or adopts other α state orientations that tilt α helices 3-5 toward the membrane (Fig. 5e).

Although membrane-based occlusion of RAS' effector binding interface is largely predicted by the G-domain orientation in these CG simulations (Fig. 5c), the G-domain's displacement from the membrane, d_z^G, is dynamic (Fig. S27A) and influences its effector binding competence (Fig. S27B).

Lipid dependence of membrane-based occlusion is shown in Fig. S27C. Notably, this occlusion decreases as PIP2 concentration increases beyond 2% (Fig. 6d and Fig. S27C). Other relationships are complex (Fig. S27C) and difficult to interpret because lipid concentrations are sampled collectively (Fig. S15).

Although the large number of simulations afforded by MuMMI dramatically reduces statistical sampling errors that are common shortcomings of MD simulation-based studies, the populations at which G-domain orientational states are sampled continue to drift with time throughout the CG simulation ensemble (Fig. S28).

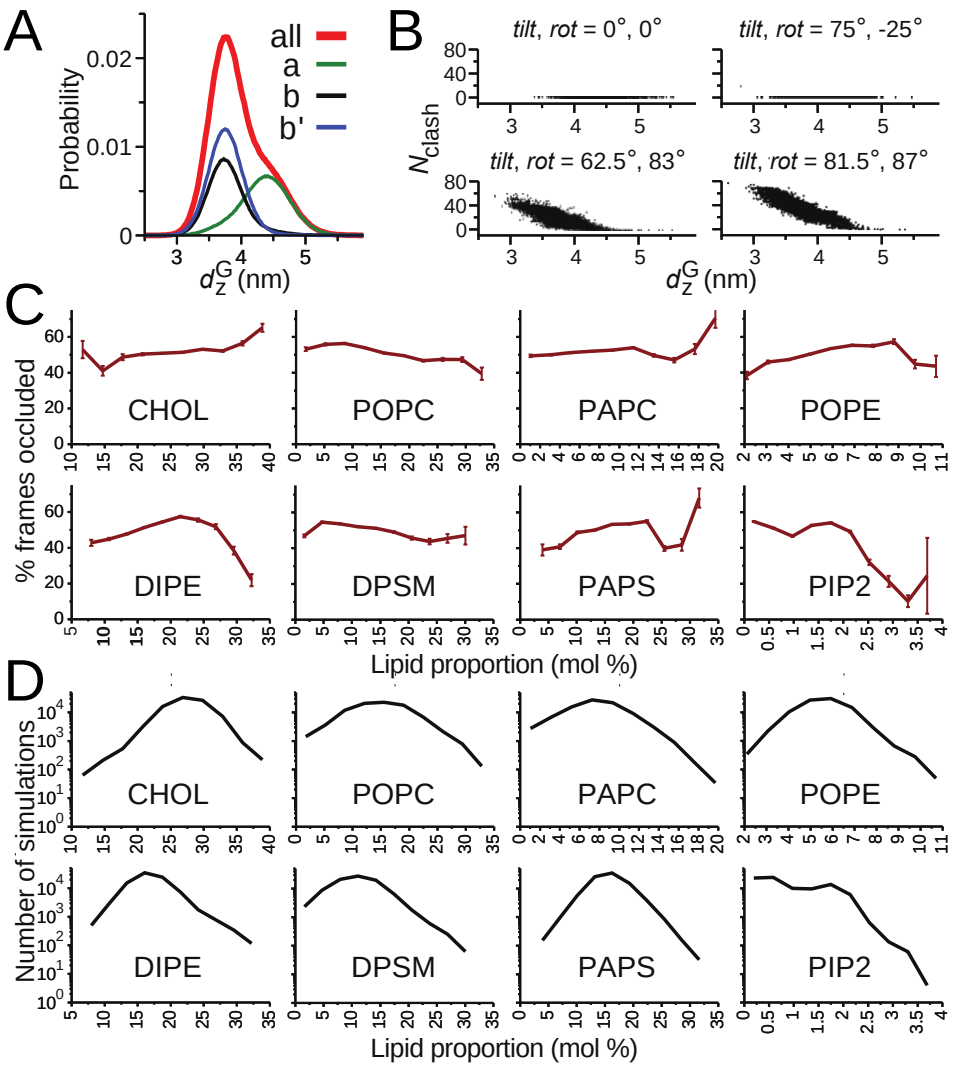

Fig. S27: G-domain disposition and competence for RAF binding in CG simulations with one RAS. (A) Probability histogram of the center of mass distance between the RAS G-domain backbone beads and the bilayer along its global normal, d_z^G, overall and separately for configurations in α, β, and β' orientational states. Probabilities in state-specific histograms are multiplied by the proportion of that state in the overall ensemble. (B) Number of C-RAF RBD backbone beads modeled closer than 1.8 nm to the bilayer center along its global normal, N_{clash}, as a function of d_z^G near the G-domain orientations depicted in Figs. 5d-5g. Orientations are included for ±0.5° tilt and ±1° rotation, except near (tilt,rot) = (0°,0°), which includes 0° to 1° tilt and all rotation values. (C) Percentage of frames consistent with membrane-based occlusion of RAF binding (N_{clash}>5) as a function of the molar percent of each lipid type in the CG simulation patch. (D) Number of simulations in each histogram bin for part C. Patch weighting is applied in part A, but not in parts C and D. Error bars in part C are from bootstrapping.

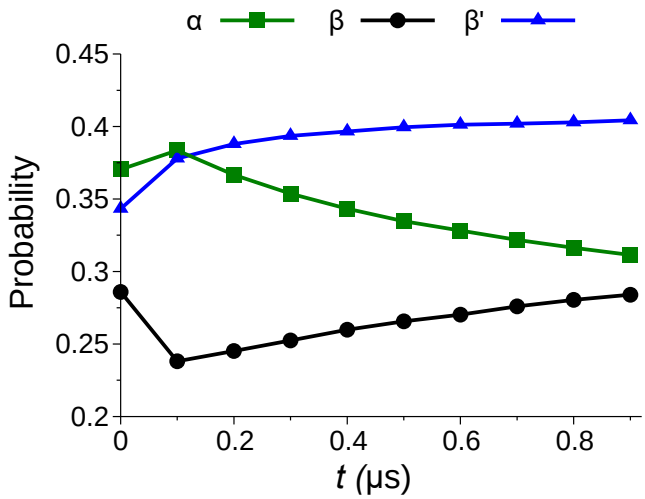

Fig. S28: Sampling of one RAS states. Probabilities at which the three orientational states of RAS are sampled from time *t* to time *t*+0.1 μs across all 1-RAS CG simulations as a function of *t*.

The above analysis suggest that membrane-based occlusion of the binding site is common, even for a relatively small binding partner such as the C-RAF RBD (78). Full-length RAF likely exerts additional steric hinderances to binding. We anticipate similar results for other structurally similar RAS-binding motifs such as the RBDs of Byr2 (94), RalGDS (95), and NORE1A (96), and the RAS associating domains of PLCε (97) and Grb14 (98). Occlusion is also expected to be prominent for large RAS-binding proteins such as PI3Kγ (99), p120GAP (100), and the GAP SOS1 (101). In contrast, the HVR-binding protein PDEδ (102) is expected to be completely incompetent to bind the configurations of RAS sampled in these CG simulations, where the farnesyl group is consistently embedded in the membrane (103, 104).

2.6. Lipid dependence of RAS states

RAS is known to interact with lipids in the membrane through electrostatic and hydrophobic interactions (105). These interactions are observed in our simulations, somewhat in the phase field model and in particular in the corse-grained MD models. Since the different states have differing orientation and proximity to the membrane of the G-domain, it is reasonable to presume that the total number of lipids in contact with an isolated RAS protein will show a strong association with the protein's orientational state. Beyond this type of relatively simple relationship with the total number of contacting lipids, we make use of bilayer's compositional complexity to evaluate the coupling of RAS orientational state to the local lipid composition in terms of the relative frequencies at which specific lipid arrangements occur.

2.6.1. Morphology of Lipid Composition Near RAS

RAS interacts with membrane lipids. Fig. S29 shows how the relative concentration of lipids vary as a function of distance to center of the G-domain and residue 185. The fact that these curves are not constant are indicative of lipid-RAS interactions. Furthermore, the concentration at very long distances is near the average concentration of each lipid in the membrane. Observe

that concentration of PIP2, which has the lowest concentration in membrane, has concentrations near RAS that exceed that of other lipids (Fig. S29C and D).

Visually, we note some state dependence of the radial lipid concentration function (Fig. S30). As previously noted, the concentration of PIP2 is enriched near the G-domain in state β and β' than in α. Our explanation for this is that PIP2 preferentially gets close to the G-domain (which in state β and β' lies at the membrane surface) thanks to its strong negative charge. There are fewer options for PIP2 to get in proximity of the G-domain in the α state, as the G-domain is further away from the membrane. We also observe that cholesterol and POPC are enriched in the outer leaflet near the G-domain, but only in states α and β. We conjecture that with the G-domain being farther from the membrane, fewer charged lipids will be near the farnesyl, leaving room for small lipids (such as Cholesterol) to sneak in. Fig. S31 provides evidence supporting that conjecture. It shows spatial dependence for lipid concentrations once the membrane is rotate around a particular residue to align the axis of the G-domain to the x-axis. We interpret that plot as saying that the distributions of lipids around RAS are also influenced by the orientation of G-domain.

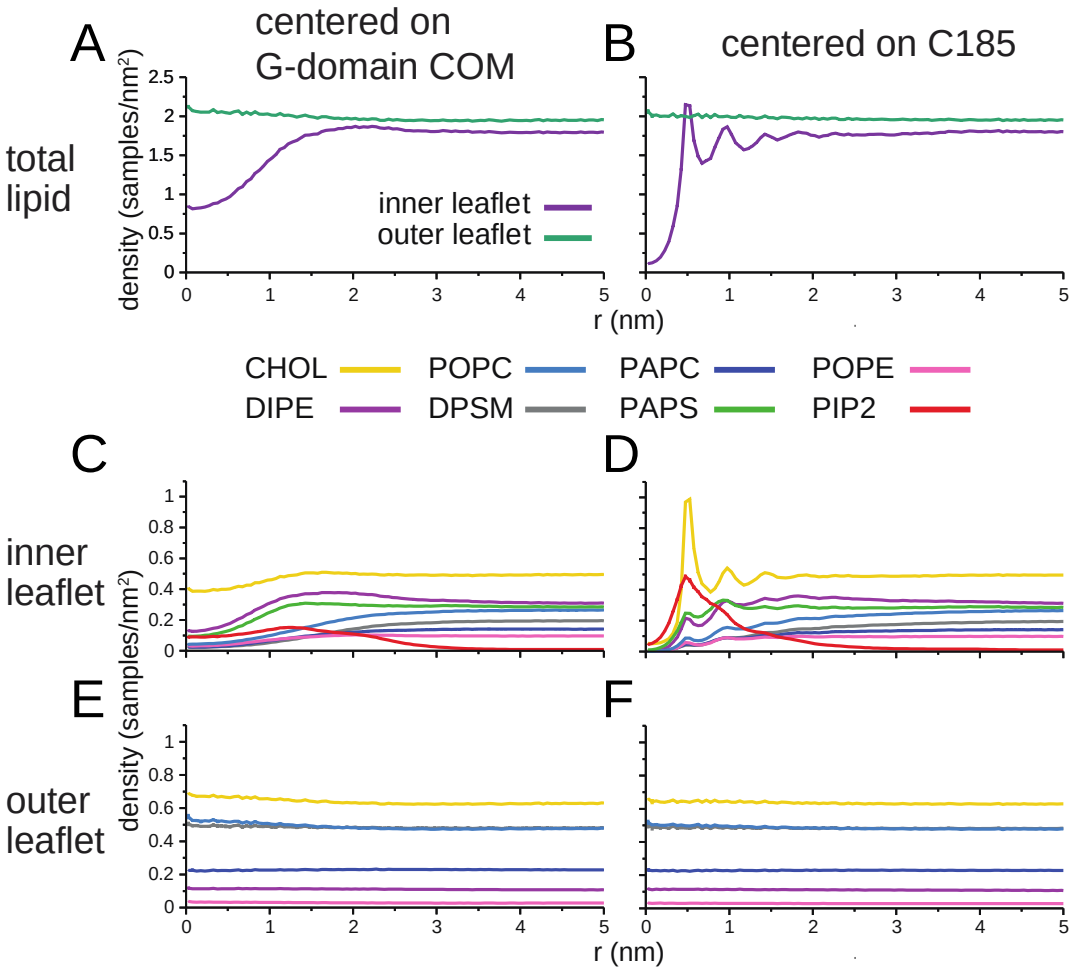

Fig. S29: 1D radial lipid distributions around RAS in CG simulations with one RAS. Radial densities of lipid headgroup beads (PO4 for lipids and ROH for cholesterol) in the global bilayer plane are shown (A, B) collectively, and by lipid type in the (C, D) inner and (E, F) outer leaflets. The group at $r = 0$ is (A, C,

E) the center of mass of G-domain backbone residues, or (B, D, F) the backbone bead of residue C185. Data is patch weighted. Error bars obtained from three sets of interleaved CG simulations are too small to see. Data from the second segment of CG simulations only.

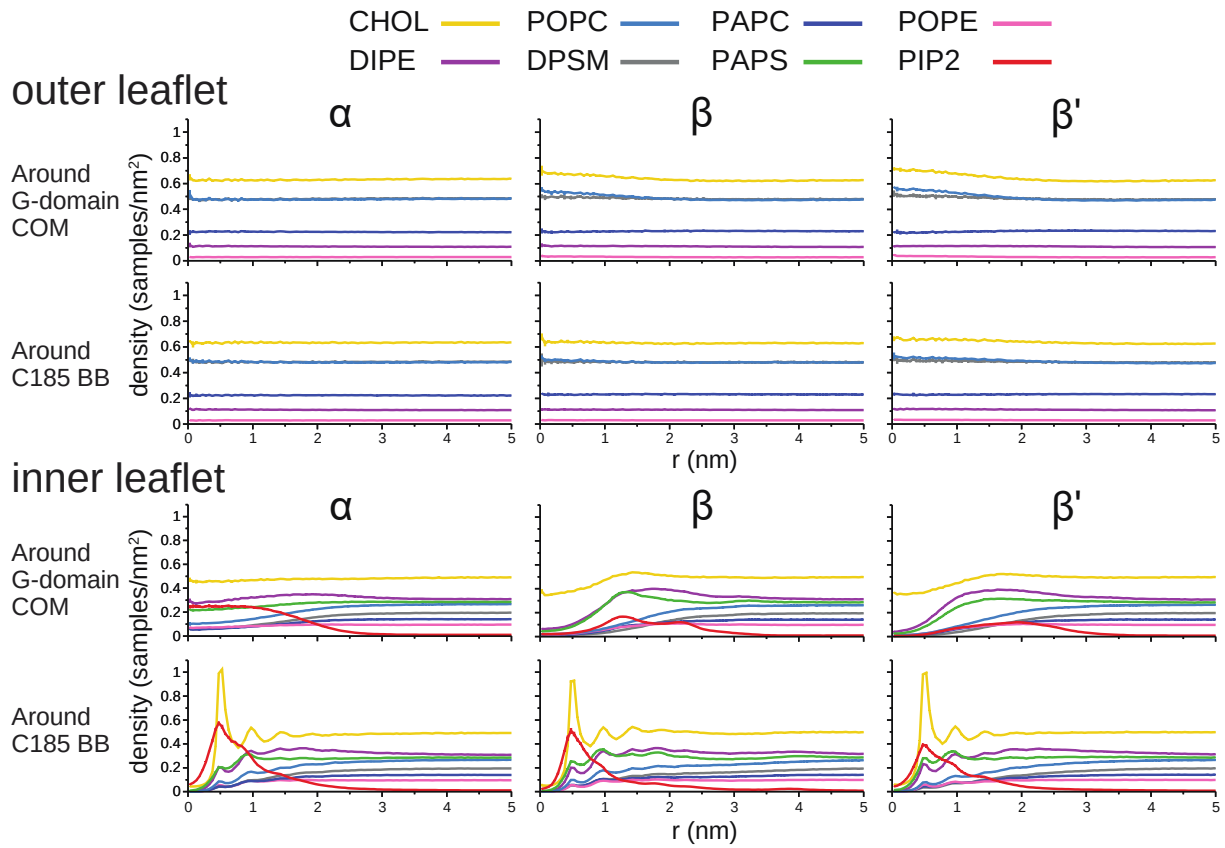

Fig. S30: Lipids around RAS G-domain in different RAS states. G-domain orientation-state-specific 1D radial lipid distributions around RAS in CG simulations with one RAS. Radial densities of lipid headgroup beads (PO4 for lipids and ROH for cholesterol) in the global bilayer plane are shown when the RAS G-domain is in the (left) α, (middle) β, or (right) β' orientational state. Data shown for (top two rows) outer and (bottom two rows) inner membrane leaflets. Within each of the two vertical sections of this plot, the group at $r = 0$ is (top row) the center of mass of G-domain backbone residues, or (bottom row) the backbone bead of residue C185. Data is patch weighted. Error bars obtained from three sets of interleaved CG simulations are too small to see.

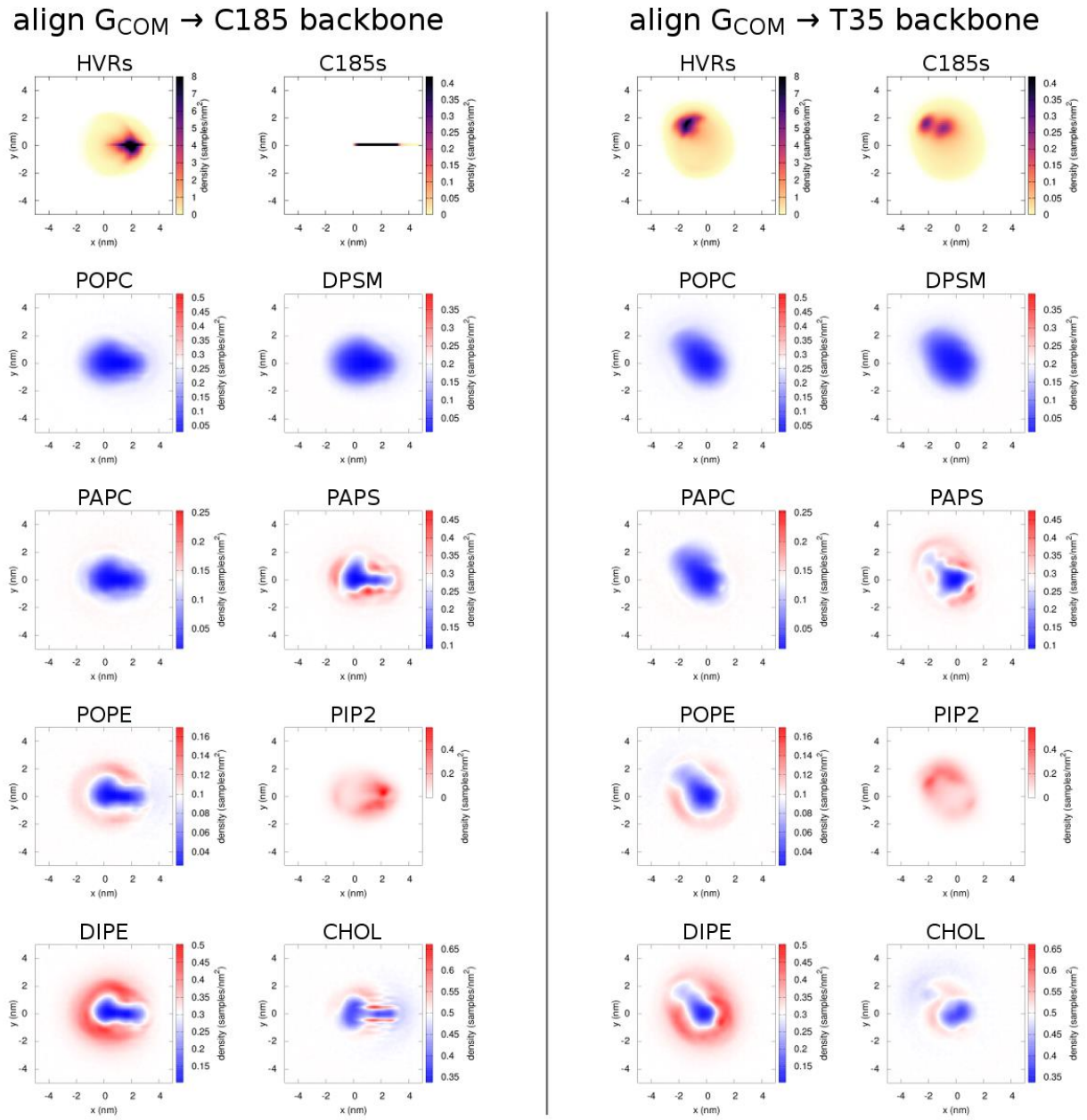

Fig. S31: Oriented 2D distributions of protein and lipids in CG simulations with one RAS. Densities shown separately for HVR beads, the C185 backbone bead, and lipid headgroup beads (PO4 for lipids and ROH for cholesterol). Densities computed after placing the G-domain center of mass at the origin and rotating all coordinates to place the (left) C185 backbone bead, or (right) Thr35 backbone bead on the positive **x**-axis. Color scales differ among groups for which densities are plotted, and are determined as follows. The (white) midpoint is the average value over radial distances $6 < r < 10$ nm, where $r = \mathrm{sqrt}(x^2+y^2)$; the initial low is the minimum density for $r < 5$ nm; the initial high is the maximum density for all values of x and y; the (blue) low and (red) high are then set such that they are equidistant from the midpoint according to the largest absolute value from that midpoint. During this process, data from different alignments are considered together. Data from the second segment of CG simulations only and not patch weighted.

2.6.2. Estimating RAS State from Lipid Density with Supervised Classification Models

In the previous section, we have shown a co-dependence of membrane lipid composition and RAS state. We next investigate correlations between RAS and neighboring lipids using a supervised ML model, which is trained to detect whether one can accurately label the RAS state by observing its surrounding lipid densities. As described in Section 1.3.4, this analysis is conducted for both CG simulations (three-state RAS model) and macro model (two-state RAS model). Furthermore, to capture the correlation between lipids and the state of a (single) RAS, we focus only on the regions/simulations with a single RAS.

Prediction for CG simulations. Convolutional neural network models (see Section 1.3.4) are trained with each lipid type individually and also with combinations of different lipids. When using multiple lipids, the input is treated as a multi-channel image with one channel devoted to each lipid type. Fig. S32 shows the accuracy of predicting the RAS state from the lipid density on training and test data, respectively. The plots show the average and standard deviation of the resulting accuracy of models trained on a random training set selection from the entire dataset. The training and test datasets are equally balanced between the three RAS states, so the values higher than ~33% accuracy indicate that the model is doing better than a random chance. We observe that some lipid types are better predictors of RAS state than others. In particular, models trained on PIP2 deliver the best accuracy among single-lipid models, followed by inner leaflet DIPE, POPC, and PAPS. The models trained on outer lipids largely do not perform better than random chance. The best model is obtained when combining all eight lipids of the inner membrane.

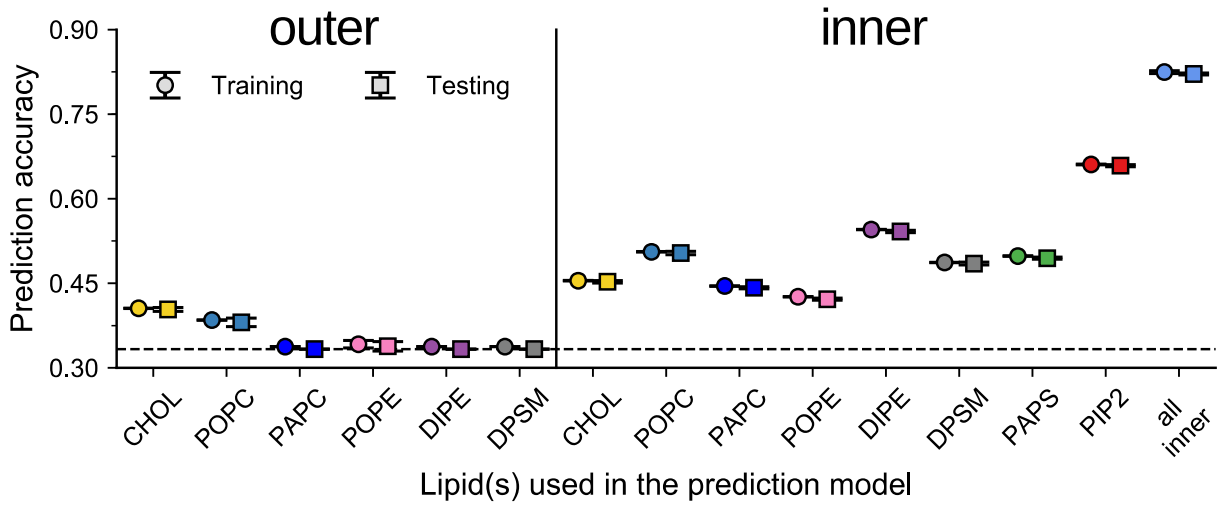

Fig. S32: Training and testing accuracy for the three-state prediction from CG data. Mean and standard deviation of training and testing accuracies are shown when predicting RAS state from lipid densities for different lipid types individually as well using all eight inner lipids combined. The results show that outer lipids perform just about as good as a random chance (dashed line), whereas considering all lipids on the inner leaflet is able to predict RAS states with ~80% accuracy.

Fig. S33 shows the average density for each lipid for all frames in the test data that the model correctly predicts to be in each state. These images, therefore, represent spatial patterns of lipid densities that the model learns to be indicative of the RAS being in the corresponding state.

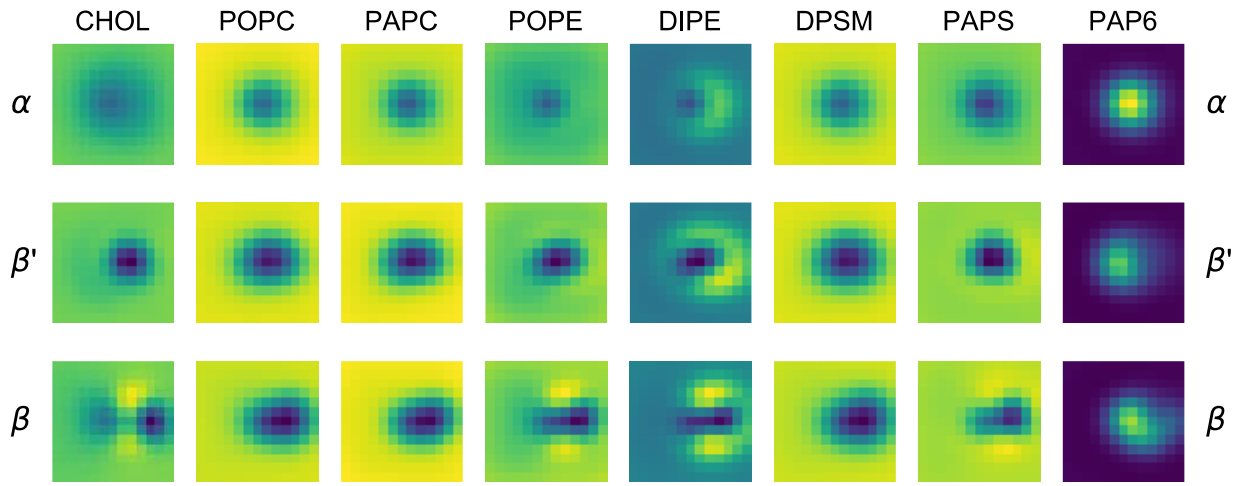

Fig. S33: Average ML prediction of lipid fingerprints for the three RAS states in CG data. Average densities for inner lipids from frames correctly predicted by the model to be in one of the three states. The three rows show the lipid densities averaged over 182,000, over 113,000, and over 32,000 frames correctly predicted to be in states α, β', and β, respectively. Color represents (blue) low to (yellow) high lipid density. Note that these images differ from those in Fig. 6a and Fig. S31 in part because trajectory frames for ML-based state prediction are processed to place the farnesyl at the origin and rotate the G-domain center of mass onto the positive **x**-axis, whereas the representation used for Fig. 6a and Fig. S31 is the inverse.

Prediction for Macro Model. A similar analysis with models trained to predict RAS state from lipid densities is conducted for the macro model. In the macro model, a Markov model governs when the RAS transitions from one state to the other. The lipids then react to the RAS state according to forces dictated by the particular RDF assigned to the RAS state for each lipid, as described in Section 1.1.3. For each frame of the macro model simulation, the RAS state is known, and the lipid densities are computed for a 36×36 data grid spanning a 30×30 nm^2 area centered on the RAS. A convolutional neural network model similar to the model described for the CG data is trained for state classification. Further details about the model and the dataset are provided in Section 1.3.4.

As with the models trained on CG data, models trained on the macro model data successfully identify spatial patterns in the lipid densities to accurately predict the RAS state. Fig. S34 shows a plot of the average predictions of the models trained on each of the individual inner lipids in time windows centered around a RAS state transition. In particular, the model trained on PIP2 densities proves to be the most accurate, followed by PAPS, DIPE, CHOL, and POPE. Another observation is that the models trained on different lipids demonstrate a varying degree of time lag in adapting to predict the new RAS state after a transition. For instance, PIP2 and DIPE adapt more quickly than PAPS or POPE, which improve their average prediction more gradually after more timesteps. Presumably, the different lipids diffuse through the membrane in the macro model at different rates according to the different forces implied by the RDFs used in the model. The patterns in lipid density that represent each RAS state as learned by the ML model require different amounts of time to settle for the different lipids.

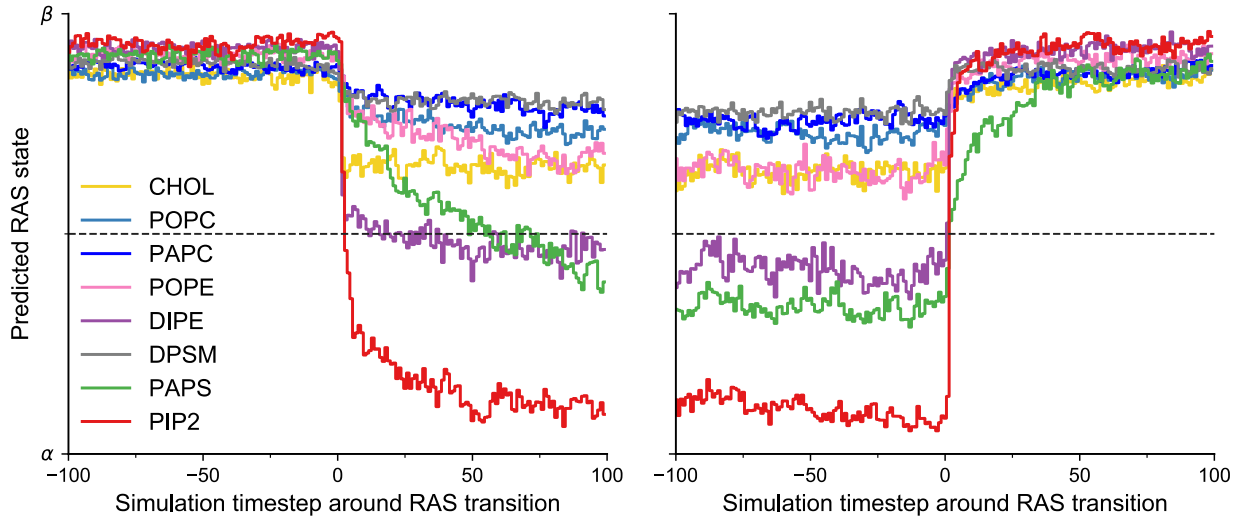

Fig. S34: Average ML prediction of RAS states for the macro model data. Average of model prediction of RAS state for models trained on individual lipids from macro model data in a time window around RAS transitions. Each plot shows the results for a model trained only with the concentrations of one of the eight inner lipids.

To illustrate the types of patterns that the different lipid densities form under different RAS states, Fig. S35 shows the densities of several inner lipids for a sequence of time steps around a transition from state β to state α in the macro model. While the RAS is in state α, the PIP2 and DIPE densities intensify around the RAS as those lipids are pulled in more closely to the RAS, while PAPS and CHOL tend to be pushed away from the RAS creating voids in the densities of those lipids. These spatial density patterns develop from the different RDFs associated with the RAS states in the macro model.

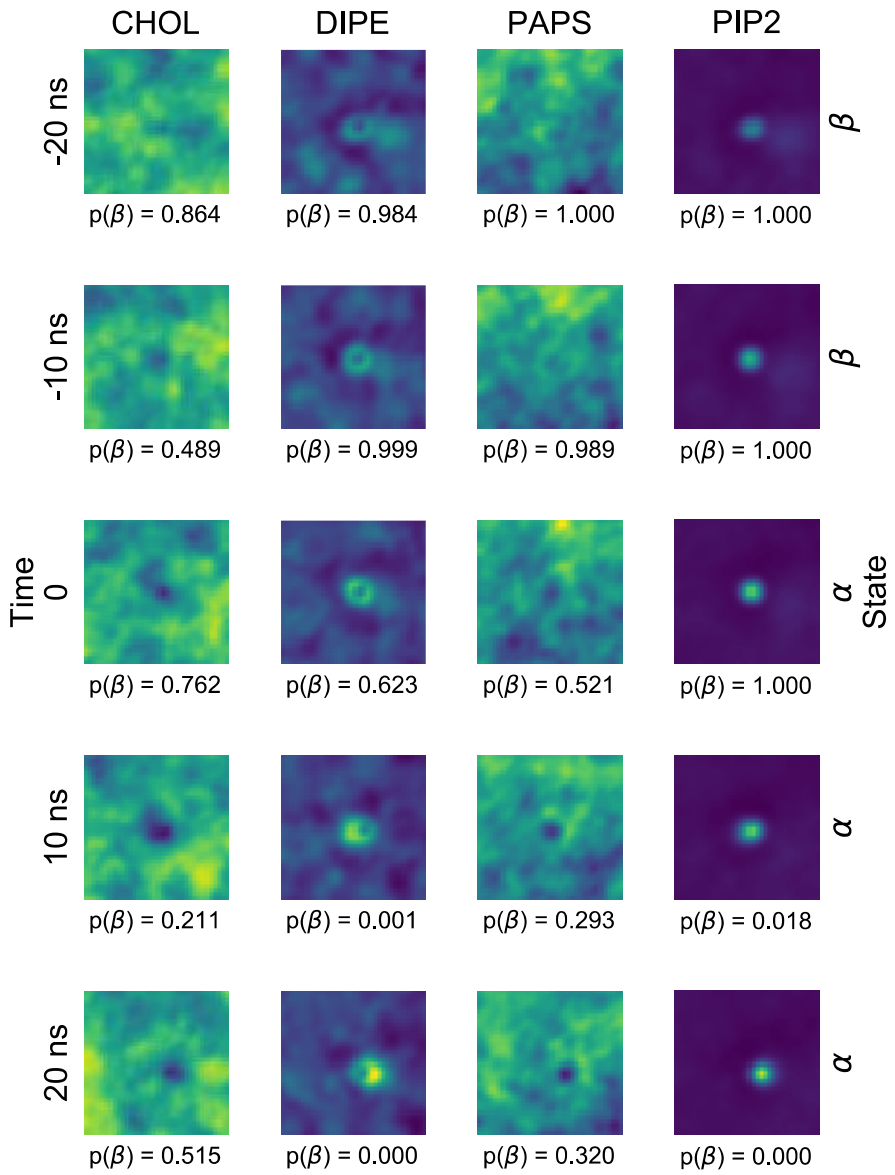

Fig. S35: Examples of ML prediction of RAS state transition for the macro model data. Lipid densities for CHOL, DIPE, PAPS, and PIP2 taken from a select macro model simulation that spans 20 timesteps (40 ns) around a point where RAS transitions from β to α. Each row shows the relative simulation time and the known RAS state. Each image is labeled with its estimated probability of the RAS being in β as predicted by an ML model that is trained only on densities of that respective lipid.

In summary, there is a correlation between RAS state, as defined by its tilt and rotation angles relative to the membrane surface, and the lipid densities within the inner leaflet of the membrane that are near the RAS. The strongest correlation occurs when considering the densities for all eight inner lipids together. Individually, the density patterns of some lipids show a stronger correlation to RAS state than other lipids. These patterns are detectable by a ML model in both the CG simulations and the macro model simulations.

2.6.3. Predicting Future RAS State Changes from Past Changes in Lipid Density

Predicting RAS state changes based on past lipid composition is of significant interest because state changes may implicate changes in down stream effects. This section presents analysis that may provide insights into how lipids could mediate RAS state changes. Fig. S36 provides evidence that lipid composition changes after a transition. This section presents further evidence and analysis showing how changes in lipid composition predict RAS state change.

This analysis only considers CG simulations that contain a single RAS protein in the patch. For every observed state transition in a given trajectory, a randomly selected "nontransition" is selected from the same trajectory. For example, if a given trajectory shows a transition from state α to β, then a random α to α transition from that trajectory is selected, if such a transition exists in the trajectory. For each selected transition and nontransition, the lipids in a backward-looking window (lag time) of length going from 10 to 250 ns are considered. These are used to compute the average total number of lipids and the average composition in each window, which are used as features to predict the probability of change.

For each possible transition (say from state α to β), a logistic regression is performed, using average total and the square-root of the average fraction of lipids as the explanatory variable. The use of the square root helps to "self-normalize" the compositional frequency vector, as that vector has norm one. It also removes the co-linearity with the constant. To help interpret the results, features to be included into the model were selected using the Bayesian Information Criterion (BIC). The BIC criterion is known to be conservative, and only includes variables that would be judged as highly significant by other analysis approaches. However, we do not include a formal p-value for the fitted parameters, as their nominal significance as reported by standard statistical software are distorted by the model selection procedure we used. Thus, the presented results should be considered in the framework of hypothesis generation instead of significant findings. Finally, the analysis was done separately for lipid composition near the G-domain. RAS' local lipidic environment was defined by the numbers of each type of lipid within 1.2 nm of any protein BB bead. Lipids were represented by the positions of their C1A, D1A, T1A, or R1 bead, as was done for CG RDF construction to parameterize the macro model (Section 1.3.1).

In all cases, the total number of lipids was a significant predictor. As a result, we will only describe how lipid composition helps to predict RAS state change. Fig. S36 shows how lipid composition near the G-domain impacts prediction of state changes as a function of state of origin, end state, and the size of the window. The size of the window seems to matter to some extent. It is somewhat surprising that the average composition, even for larger windows, remains significant, as it was initially conjectured that local short-term variations would mediate state changes. While Fig. 6c summarizes this plot by reporting the most important factor for each change, Fig. S36 shows that multiple membrane lipids are helpful to predict state change.

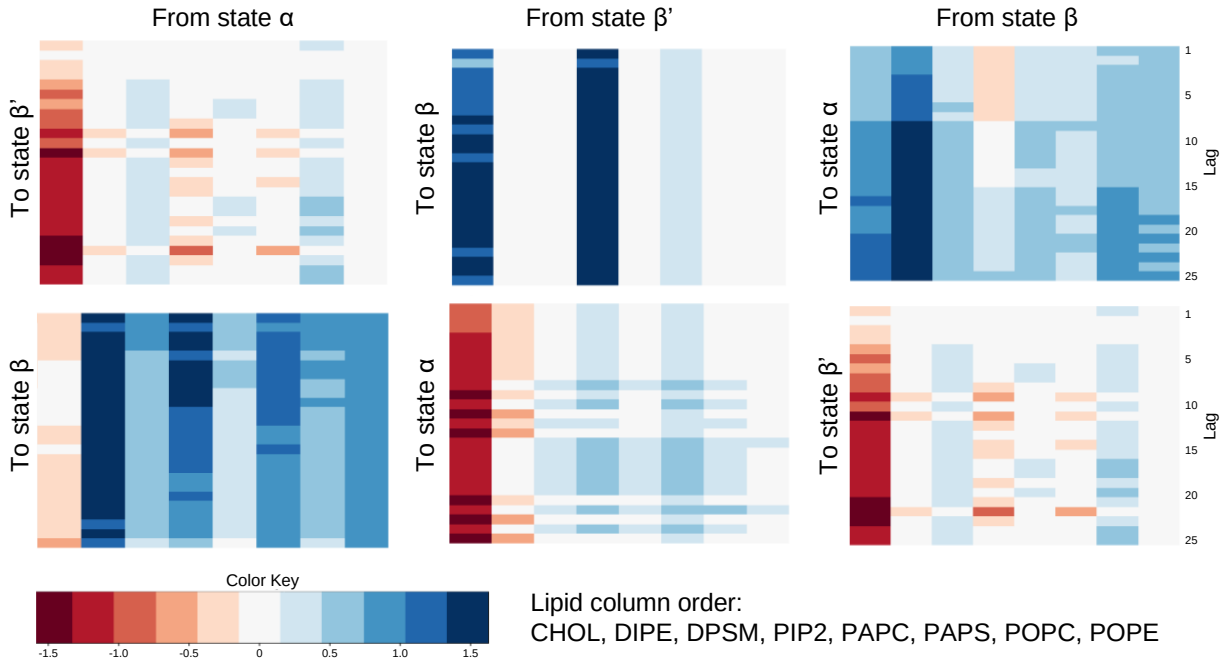

Fig. S36: Lipids predict RAS state change. Figure shows the statistically significant coefficients of lipid composition near the G-domain prior to the change to predict state change using a logistic regression. Each logistic regression estimates the logarithm of the odds ratio of the probability of moving from a stating state to a final state. Large positive coefficients (blue) indicate that increasing the proportion of that lipid increases the probability of making the change. Conversely, large negative coefficients (red) indicate that an increase in that lipid decreases the probability of a change. Average lipid composition were calculated over windows of length 1 to 25 prior to the change (10 to 250 ns).

It is interesting to note that whereas PIP2 is noted to be the most important predictor for predicting the state of RAS, changes in its state sometimes involve DIPE and Cholesterol as well.

2.7. RAS Dynamics

Next, we focus on the dynamics of both RAS and lipids in our PM mimics, corresponding to the CG micro simulations. In this section all diffusion values are corrected taking into account the about four times faster diffusion of CG Martini (10). First, we determine whether or not our simulation time-scales are long enough to de-correlate the initial anomalous diffusion (106). Fig. S37A clearly shows that in general, lipids show a linear mean square displacement (MSD) within the range of 0-1 μs. In fact, by taken the derivative of the MSD with respect to time, we are able to successfully provide reliable lipid dynamics within this time-scale (Fig. S37B). Interestingly, same lipid types show a slightly different lateral diffusion coefficient (*D*), among the different CG membrane patches and highly correlated with the change in total cholesterol concentration (Fig. S37C). We compute such correlation index for all the different lipid types, markedly agreeing regardless of the RAS content on the membrane (Fig. S37). However, such correlation is somehow affected in the case of PIP2. In fact, these lipids do not follow the same trend, and seems to be unaffected by the cholesterol content in the membrane, or at least hardly detectable. Nevertheless, we conclude that the micro-simulations (CG simulations) are effectively reproducing the change in membrane viscosity, as a result of cholesterol increase (107, 108),

clearly suggesting that the local lipid environment dynamically affects lipids. Again, and provided by our results, both RAS and PIP2 are somehow unaffected by the change in membrane viscosity, which proves intriguing.

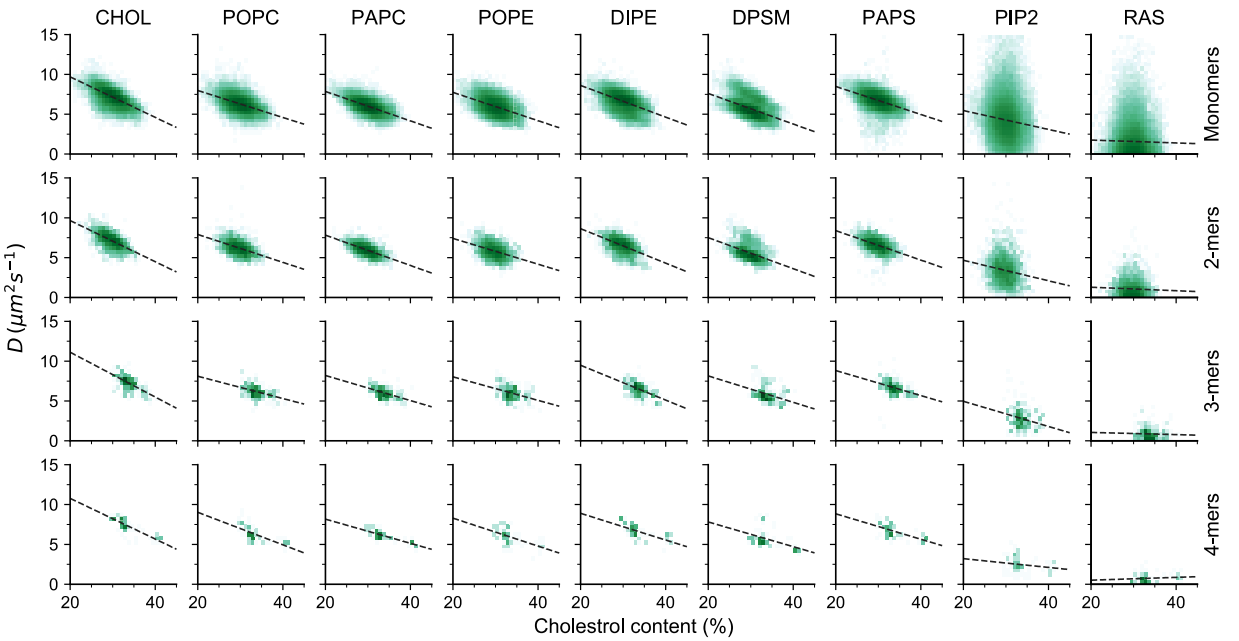

Fig. S37: Direct correlation of lipid lateral diffusion and cholesterol concentration. All lipids but PIP2 are negatively impacted by the total concentration of cholesterol in the membrane in terms of their lateral displacement. Similarly to PIP2, RAS proteins lateral displacement are not directly influenced, suggesting a de-correlated effect of cholesterol-modulating membrane viscosity.

Next, we investigate whether the lipids are also affected by the concentration of RAS molecules as extracted from the micro model simulations (Fig. S38). From the total membrane patches analyzed (~70K) all the lipids exhibit on average the same *D* (6 $\mu m^2\ s^{-1}$ under PBC conditions), regardless of RAS concentration in the simulation box. Even for higher concentrations (4 RAS in a box) the lipids resemble the features of lower RAS content. The conclusion of this graph can mislead towards a wrong interpretation of the membrane dynamics. Thus, to more thoroughly understand the behavior of the lipids in the membrane, we split the lateral diffusion coefficients by the different lipid types, as provided in Fig. S38B. In general, and as already mentioned, the similar dynamics between the lipids suggests a homogeneous behavior of them in every different patch. We should point that in case of DPSM, clearly, we can detect two subspecies, one of fast diffusion (inner leaflet) and one of slower dynamics (outer leaflet), which cannot be described by the same *D* (Fig. S38). Classifying *D* as a function of RAS aggregation state does not change the previous conclusion (Fig. S38C). However, a particular feature immediately rises as an effect of RAS concentration. Clearly, PIP2 is correlated with the lateral dynamics of RAS aggregates (Fig. S38C and D), in which we can at least distinguish four different subspecies, with PIP2 in systems with four RAS being the slowest. Although expected, our conclusive data points towards a direct modulatory effect of RAS on the dynamics of PIP2, which is correlated with the strong association of this lipid for the protein.

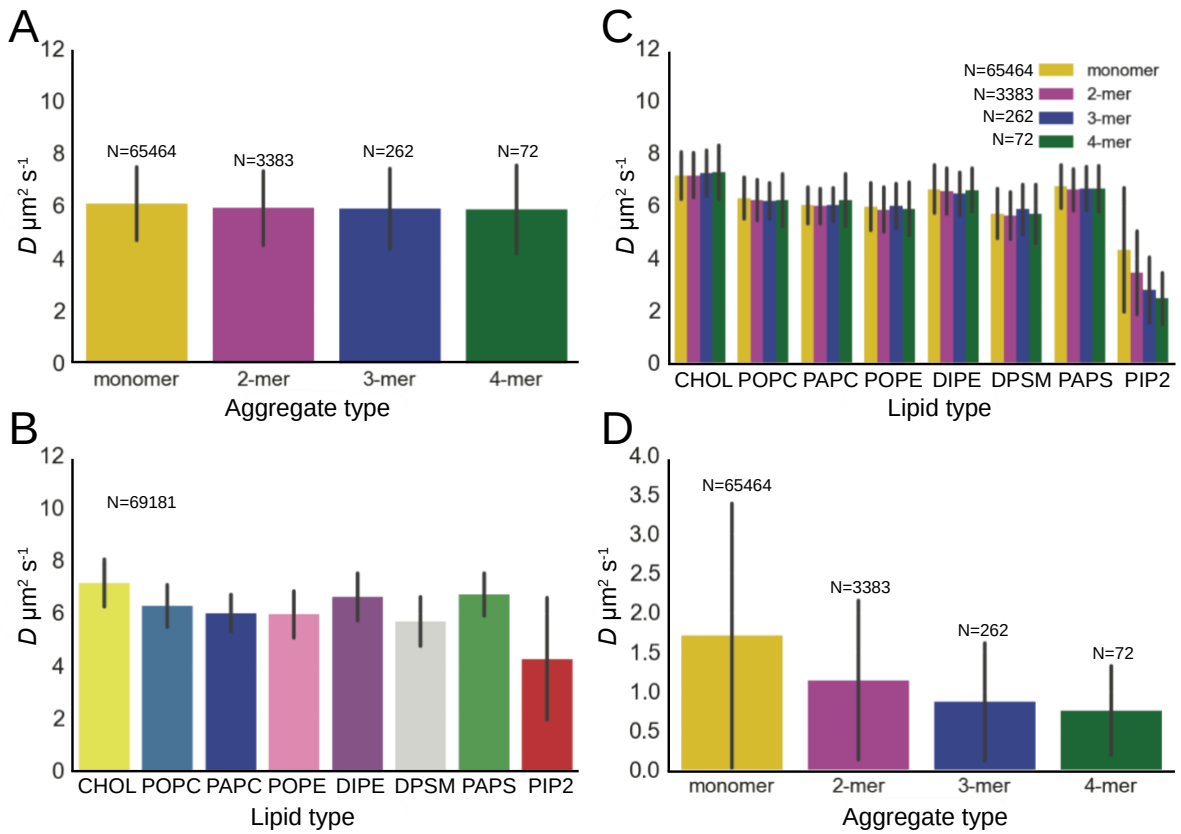

Fig. S38: PIP2 behavior is affected by RAS concentration. (A) Mean lateral diffusion coefficient (*D*) of all the lipids as a function of RAS concentration. (B) Mean lateral diffusion value as a function of lipid type. (C) *D* per lipid and split by RAS concentration, showing PIP2 as the only lipid affected. (D) Mean value of lateral diffusion for RAS protein. Bars denote standard deviation from the mean and all values are corrected taking into account the about four times faster diffusion of CG Martini.

Comparison between the simulated distances and the ones measured experimentally is not straightforward. Primarily, effects coming from the truncation via PBC plus changes in shear viscosity given by cholesterol are difficult to take into account. However, we can provide an estimate of lateral diffusion at infinite size (D^{∞}), using a PBC correction function as previously published by Venable et al. (109). We should stress that outcomes can drastically change due to two important variables; both the membrane viscosity (ηm) and hydrodynamic radius (Rh) can potentially affect the interpretation of the results.

Given the difficulty in tracking the change in membrane viscosity due to cholesterol content, we set a fix value in the formulation, which is able to recapitulate the average PBC dynamics of lipids. We find that in particular for our system, membrane viscosity needs to be increased 3 fold, thus agreeing with the higher viscosity property of our membranes when compared with pure DPPC bilayers. By doing this, our average lateral lipid diffusion constant agrees very well with the hydrodynamics Saffman-Delbrück model Table S3. After the Periodic boundary corrections, most of the lipids (except PIP2) in our microsimulations will diffuse at ~9 $\mu m^2\ s^{-1}$, a value that is very close to experimental measurements in membrane models.

Table S3: Average lateral diffusion.

Molecule	$Rh_{(nm)}$[b]	$D_{(PBC)}$($\mu m^2 s^{-1}$) raw[a]	$D_{(PBC)}$($cm^2 s^{-1} 10e^{-7}$) raw[a]	$D_{(PBC)}/D_{(\infty)}$ ($cm^2 s^{-1} 10e^{-7}$) raw[a] predicted from[b]	$D_{(PBC)}/D_{(\infty)}$ ($cm^2 s^{-1} 10e^{-7}$) raw[a] adjusted Rh[c]	$D_{(\infty)}$($\mu m^2 s^{-1}$) Martini correction[d] from adjusted Rh[c]
RAS 1-mer	1.26	6.8	0.68	2.49/3.94	1.39/2.83	7
RAS 2-mer	1.8	4.8	0.48	1.8/3.33	1/2.44	6
RAS 3-mer	2.42	2.7	0.27	1.49/2.94	0.7/2.12	5.25
RAS 4-mer	3.04	2.1	0.21	1.21/2.65	0.5/1.9	4.75
Lipids	0.53	24	2.4	2.4/3.84	-	9.5

[a]Not accounting for faster diffusion at the CG resolution
[b]As predicted from the Hydropro (110) program
[c]Rh is adjusted based on $Rh_{(experimental)}/Rh_{(computed)}$ difference
[d]Corrected to account for the four times faster diffusion at the Martini CG resolution (10)

However, correcting the lateral diffusion for RAS is more difficult. As given in the formulation, the Hydrodynamics radius (Rh) can change the properties of lateral diffusion. We use the Hydropro (110) program in order to provide an estimate of such a parameter, as listed in Table **S3** for all the different RAS aggregates. Yet, this value is unable to recapitulate the PBC dynamics, according to the theorem. In fact, we find that Rh is underestimated when compared to experimentally measured values (111), revealing the limitations of computing this property. The use of the experimental Rh still overestimates *D* obtained from our simulations, implying that other factors (not considered here) can vary the results. Nevertheless, we provide a list of *D* after PBC corrections, using an extrapolation of experimental ηm for the different RAS aggregates. Surprisingly, we find these values also in good agreement with experimental measurements (112).

Importantly, MSD coefficients for the whole series of lipids are ~30% smaller than other published values for PC lipids (109, 113). However, different integrators, as well as coupling factors, can indeed affect the lateral mobility of the lipids. In fact, ddcMD, using the Langevin thermostat and a friction coefficient of 1 ps^{-1}, compared to GROMACS, using velocity rescaling thermostat (114) and a coupling constant of 1 ps, give different diffusion values. Four identical 1 μs simulations of the ARC lipid mixture in ddcMD and GROMACS, using the parameters above, resulted in an average (±se) lipid diffusion of 26±2 $\mu m^2 s^{-1}$ from ddcMD and 42±2 $\mu m^2 s^{-1}$ from GROMACS, explaining the difference in diffusion. Note, using a higher coupling constant for the Langevin thermostat in ddcMD will move towards the weak-coupling limit and a value of 5 ps, within error, results in the same diffusion as measured with GROMACS.

We also investigated whether cholesterol-dependent membrane viscosity can affect transition rates between the different RAS states, in particular for the monomeric cases. These transition rates are computed from using the maximum likelihood HMM. Fig. S39 shows the distribution of cholesterols in the trajectories with a single RAS protein. In order to define the boundaries of our analysis for low and high cholesterol content, we characterize two different sets of trajectories: i) number of cholesterols <830 and ii) number of cholesterols >1010. These values

were chosen such that the two sets have around ~5000 trajectories and enough to obtain reliable outcomes form the HMM state analysis. Fig. S40 and

Fig. S41 show the population map, states and the transition rates for low cholesterol and high cholesterol respectively. For reference, we also show the rate kinetics obtained from HMM analysis on all the RAS (Fig. S42). It can be seen that the rate kinetics is almost identical in all the different cases implying that cholesterol content has no direct influence on the rates between the different states. The transition between states β and α is faster relative to the transition between states β' and α. Furthermore, the transition from state β to β' is faster relative to the transition β' to β. This implies that the trajectory likes to remain in state β, which is also evident from the equilibrium populations.

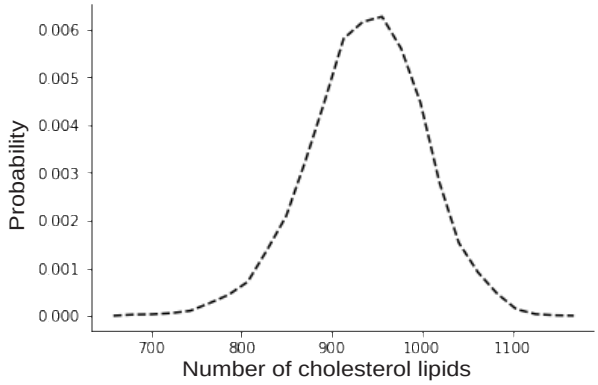

Fig. S39: Distribution of number of cholesterols in simulations with one RAS.

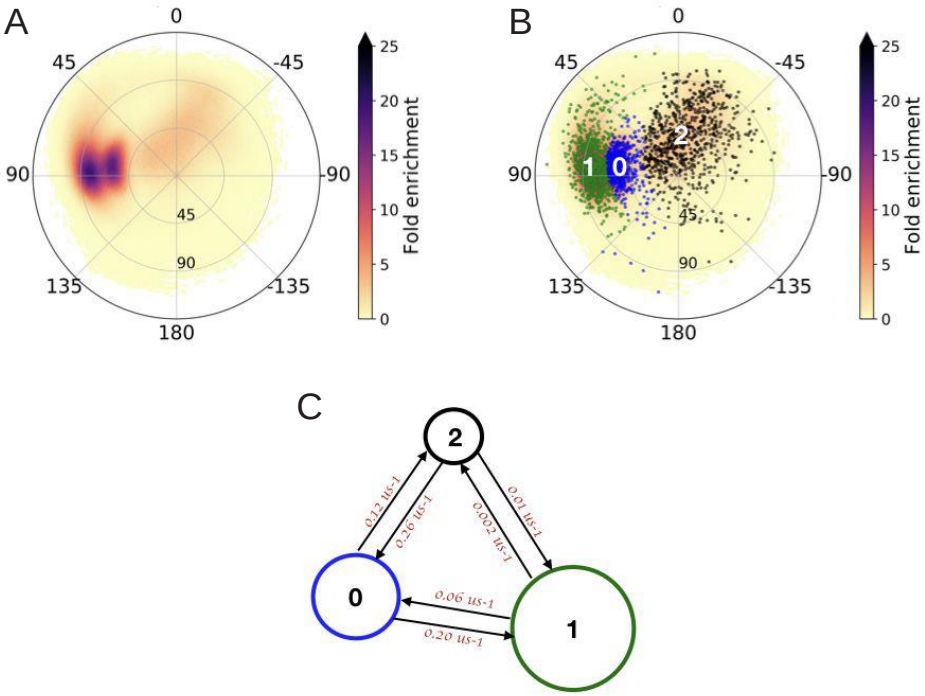

Fig. S40: Low cholesterol RAS states. (A) Population map in tilt-rotation subspace. (B) Three metastable states obtained from HMM. The microstates are colored according to the macrostates they belong to. (C) Rate kinetics obtained from HMM analysis. The RAS state names are β = 0, β' = 1, and α = 2.

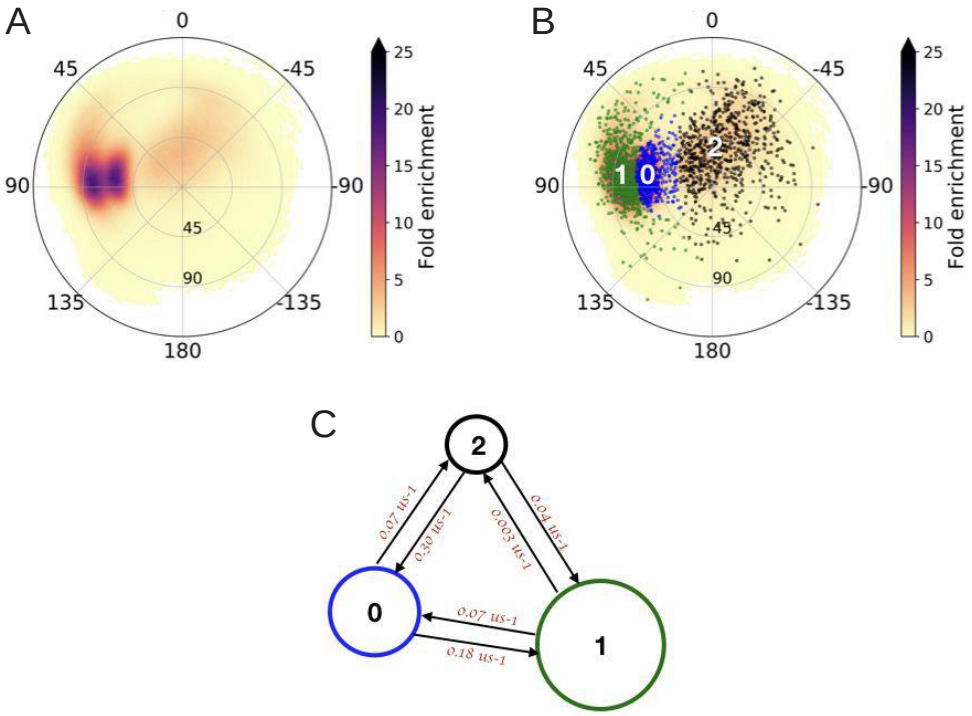

Fig. S41: High cholesterol RAS states. (A) Population map in tilt-rotation subspace. (B) Three metastable states obtained from HMM. The microstates are colored according to the macrostates they belong to. (C) Rate kinetics obtained from HMM analysis. The RAS state names are β = 0, β' = 1, and α = 2.

Fig. S42: Rate kinetics obtained from HMM analysis of all RAS monomer trajectories. The population map and the three metastable states are shown in Fig. S25. The RAS state names are β = 0, β' = 1, and α = 2.

Next, to provide an experimental verification that we are reliably capturing the dynamic behavior of RAS and lipids in our CG simulations, we measured the lateral diffusion of RAS and lipids on a reconstituted model membrane using fluorescence lifetime correlation spectroscopy (FLCS). We prepared supported lipid bilayer composed of the symmetric 8 lipid ARC mixture labeled with trace amount of fluorescent lipid and fully processed, farnesylated and methylated, KRAS4b S106C/C118S covalently labeled with Alexa647 dye. We probed the translational mobility of the different lipid classes: Cholesterol, POPC and PAPC, DIPE and POPE, DPSM,

PAPS, and PIP2 in the bilayer with Topluor Cholesterol, TopFluor TMR PC, TopFluor TMR PE, TopFluor Sphingomyelin, TopFluor PS, and Topluor PIP2 respectively. Fig. 3b shows *D* measured for the different lipid types. In general, the lipid diffusion follows a trend like the one observed in simulations as shown in Fig. S38B. The diffusion coefficient for all lipid types remains relatively the same, on average ~7 $\mu m^2/s$, except for PIP2. PIP2 moves significantly slower, a distinct feature evaluated from simulations directly correlating PIP2 dynamics to that of RAS aggregates. Likewise, RAS diffusion also remains homogeneous in all our experiments, ~4 $\mu m^2/s$, shown in Fig. 3b. It is important to note that FLCS is an average ensemble measurement technique and unlike in simulations, it cannot isolate diffusion based on RAS aggregation state such as monomer, dimer and so on, therefore, the measured *D* is a weighted average of the mobility of all states in a system (91). However, under our experimental conditions, i.e., a micromolar concentration of RAS, the probability of RAS aggregation is high. Thus, the quantitative values determined by our experiments for RAS and lipid diffusion conclusively support the CG simulations and prove a strong correlation between PIP2 and RAS.

2.8. RAS Lipid-Dependent Aggregation

The results discussed in Section 2.6 demonstrate that in both the micro and the macro model, one can detect a strong correlation between the lipid configurations and the state of the RAS protein. A related question is how the arrangement of lipids is connected to RAS aggregation. This is especially interesting as aggregation can be observed more easily in experiments.

2.8.1. Lipid-Dependent RAS Clustering in the Macro Model

One of the overarching goals of the project is to identify new hypotheses that can be experimentally verified. An initial target for such experiments is the lipid-dependent clustering behavior of RAS as predicted by the macro model. In particular, when one computes the average lipid densities of the inner leaflet underneath each RAS within a small (5 nm radius) neighborhood, clusters of multiple RAS have a systematically different lipid environment than single RAS molecules. For example, Fig. S43 shows the marginal distributions of average lipid densities underneath RAS computed from neighborhoods of the macro model simulation for isolated RAS compared to clusters with three or more RAS.

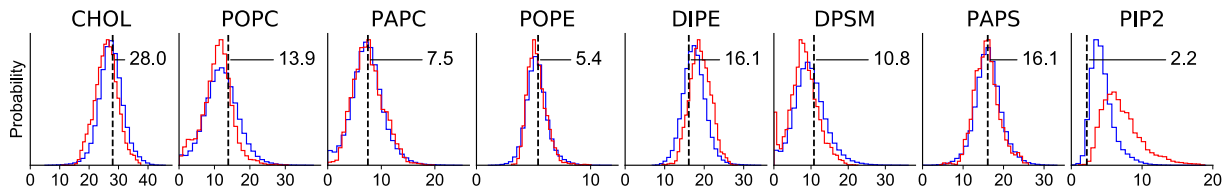

Fig. S43: Distributions of inner leaflet lipid densities in neighborhoods underneath RAS molecules. Distributions for isolated RAS are shown in blue and for clusters with 3 or more RAS shown in red. The ARC values for each lipid are shown as a dashed vertical line.

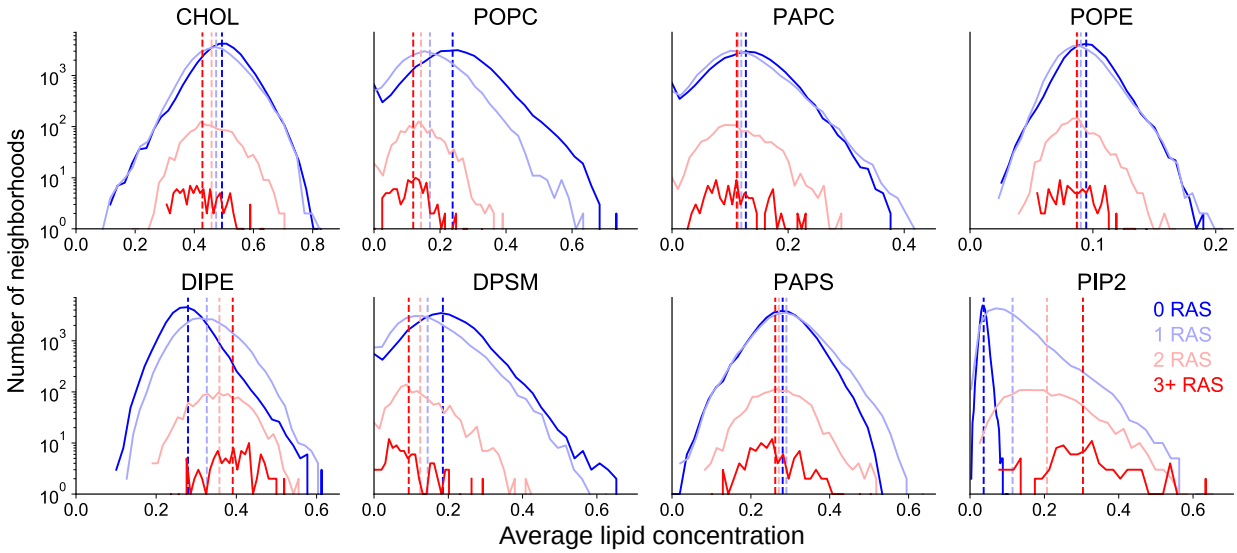

Fig. S44: Distribution of RAS counts with changes in local lipid concentration. Distribution of the number of RAS neighborhoods corresponding to the mean inner leaflet lipid concentration within 5 nm of the proteins as the RAS count varies from 0 to 3 or more. The curves for 0 RAS are shown in blue and for 3 or more RAS are shown in red with the intermediate 1 RAS and 2 RAS curves shown in light blue and light red, respectively. Also shown are the mean values marked by the vertical dashed lines. The increase in RAS count corresponds to an increase in mean PIP2 and mean DIPE concentrations with PIP2 showing a more pronounced increase. But, the mean CHOL, POPC and DPSM concentrations show a decrease for increasing RAS count with the effect being more pronounced for POPC followed by DPSM and CHOL. For PAPC, POPE, and PAPS, the increase in RAS count shows a minimal effect on their mean concentrations compared to the effect for the other inner lipids.

2.8.2. Finding Lipid Compositions that Modulate RAS Aggregation

The initial analysis of data in Section 2.8.1 shows that RAS clusters have a different lipid environment than RAS monomers or regions without RAS. Further examination is required to fully ascertain lipid compositions that will moderate RAS aggregation. Given the eight-component nature of the lipid composition, they represent a complex high-dimensional distribution that is not easily interpretable. Thus, we use a two-step process based on Function Preserving Projections (FPP) (76) to determine appropriate compositions. First, each neighborhood beneath a RAS (within 5 nm of RAS), i.e., each set of lipid concentrations, is labeled with the corresponding number of RAS. Furthermore, a set of equivalent neighborhoods without RAS are selected from the macro model at random to form a no-RAS baseline. Subsequently, one can phrase the lipid-dependent clustering as a regression function from eight-

dimensional space representing the densities of the eight inner lipid types, to the number of RAS. FPP is designed to find optimal linear projections, such that a given function appears as a simple, low-order signal in the projected space. Applying this approach to all neighborhoods produces the two-dimensional embedding of the patches shown in Fig. S45, which indicates a strong relationship between the number of RAS and the lipid composition.

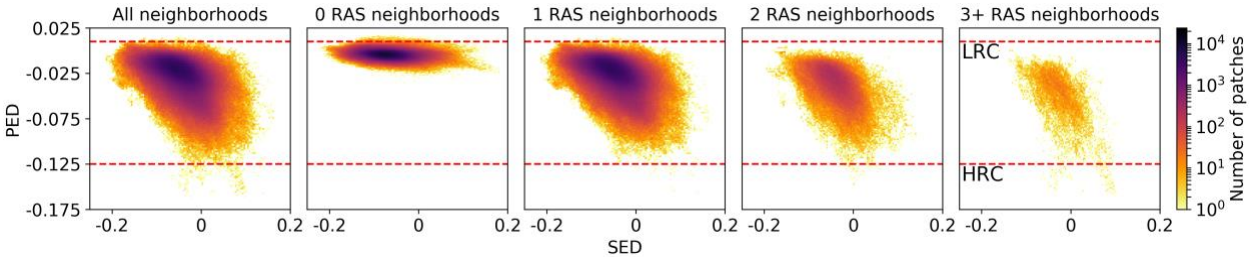

Fig. S45: FPP two-dimensional embedding of patches. Optimal linear projection as computed by FPP of the eight-dimensional space of lipid compositions. The data is plotted using the primary embedding dimension (PED) and the secondary embedding dimension (SED). The complete set of neighborhoods is shown on the left, followed by data separated by RAS count. The red dashed lines represent the chosen thresholds, which are used to define the 'high RAS colocalization' (HRC) and 'low RAS colocalization' (LRC) lipid compositions.

In particular, the vertical direction – termed the 'primary embedding dimension' (PED) – in the resulting plot correlates strongly with the number of RAS in a neighborhood. Consequently, one can sub-select test configurations by selecting horizontal cuts in the embedding space and selecting all neighborhoods above the higher line as low RAS colocalization (LRC) and all patches below the lower line as high RAS colocalization (HRC). Note that, since FPP computes a linear projection, these lines represent hyperplanes in the original eight-dimensional input space.

The corresponding selections (either above the higher line, or below the lower line) show stronger differences in compositions yet remain multi-modal. In the second step we place additional constraints on the sub-selection based on individual marginals to reduce the degree of multi-modal behavior as much as possible without sacrificing sample populations (Fig. S46). The average molar percentages for these lipids are then normalized to sum to 100%. Thus, the final lipid distributions are shown Fig. 2d and Table S4.

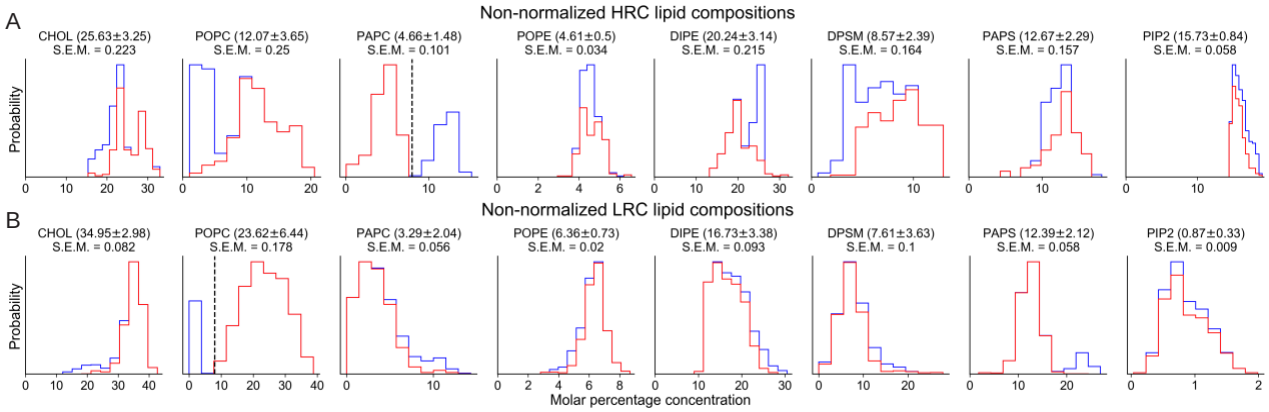

Fig. S46: PED threshold cuts. Initial PED threshold cuts produces two sets of lipid distributions (blue lines); HRC (A) and LRC (B). Several of these individual lipid distributions are multi-modal in nature. A single secondary cut (dashed black line) is made in one of these lipid distributions to reduce the multi-modal nature of the data as much as possible (red lines).

Table S4: Compositions of defined inner leaflet lipid mixtures (Lipid mol % ± sd)

Lipid	LRC	ARC	HRC
CHOL	33.0 ± 2.8	28.0	23.6 ± 2.3
POPC	22.3 ± 6.1	13.9	11.0 ± 3.3
PAPC	3.1 ± 1.9	7.5	4.7 ± 1.5
POPE	6.0 ± 0.7	5.4	4.5 ± 0.5
DIPE	15.8 ± 3.2	16.1	21.0 ± 3.1
DPSM	7.2 ± 3.4	10.8	7.5 ± 2.0
PAPS	11.7 ± 2.0	16.1	12.2 ± 2.6
PIP2	0.8 ± 0.3	2.2	15.5 ± 0.9

2.8.3. Construction of the High/Low RAS Colocalization (HRC/LRC) Lipid Compositions

To test the correlation between the lipid compositions, HRC and LRC, identified in Section 2.8.2 with the level of RAS aggregation, we made various biophysical measurements of RAS on liposomes composed of HRC and LRC as defined in Table S4. We created symmetrical inner leaflet version of the HRC/LRC RAS-PM 8 lipid mixtures and tested them using HPLC technique (92), Fig. S18 for ARC and Fig. S47 for LRC and HRC lipid compositions. We assigned the different peaks in the HPLC spectra to the specific lipid species based on spectra collected from standard stock samples of each lipid types (see Section 2.3.1). Clearly, the spectra show that all 8 different lipid types are incorporated in both lipid systems, and in addition, the relative intensity differences for each lipid types between LRC and HRC compositions match the corresponding theoretical differences in the lipid content. This confirms the desired compositional variations in LRC and HRC liposomes used in the experiments henceforth (Sections 1.4.10 and 1.4.13).

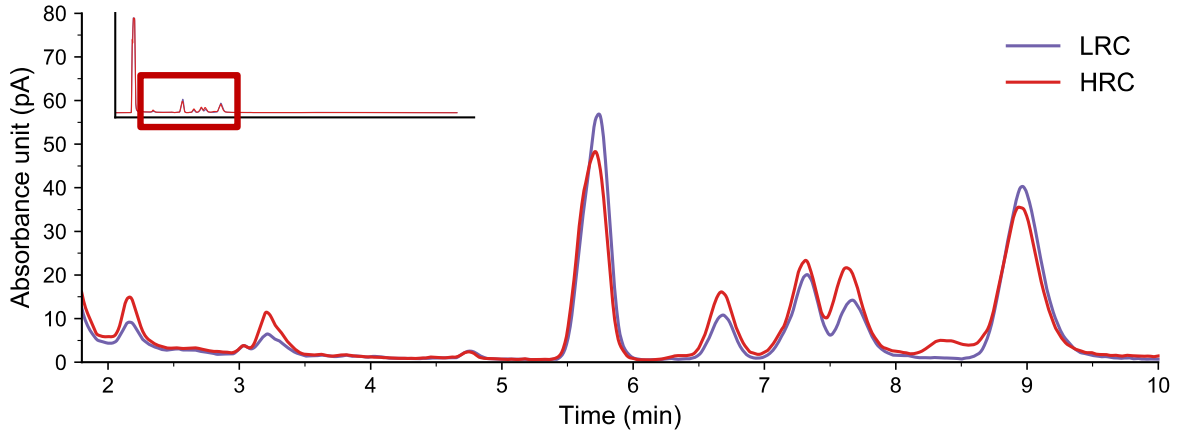

Fig. S47: HPLC spectra of LRC and HRC. HPLC chromatogram of the 8 lipid type inner leaflet mixtures LRC (blue) and HRC (red) as described in Table S4. HPLC chromatogram for ARC is shown in Fig. S18.

2.8.4. Single Molecule Tracking Studies of RAS on HRC/LRC Supported Lipid Bilayer

We tested the lipid dependent clustering of RAS on reconstituted membranes by measuring the mobility of RAS using single molecule tracking (SMT) studies facilitated by numerous statistical analysis tools including hidden Markov modeling (HMM) and single step-length distribution analysis. Recently, using live cell SMT studies and HMM analysis on a very rich set of trajectories, Goswami et al. (115) and Lee et al. (116) reported three distinct diffusion states for RAS on the plasma membrane of live HeLa and U2OS cells; fast, intermediate, and immobile states with unique inter-state transition paths. Here, we created reconstituted supported lipid bilayer with 8 lipids LRC and HRC compositions that favored RAS monomers and RAS multimers respectively, and tracked the diffusion of JF646 labeled RAS and Atto550 DOPE on the bilayer using TIRF microscopy. Both lipids and RAS were mobile and recovered after photobleaching confirming a mobile lipid bilayer. We also counted the number of bright particles in each frame to test if RAS reaches equilibrium on the membrane after washing. Fig. S48 shows the particle count obtained from the localization processing of the stacked images for up to 1000 frames collected during a representative movie for LRC, ARC and HRC. The number of particles per frame remains same on average with minor variations. The slight change in molecule counts is due to RAS molecules diffusing in and out of the confocal plane.

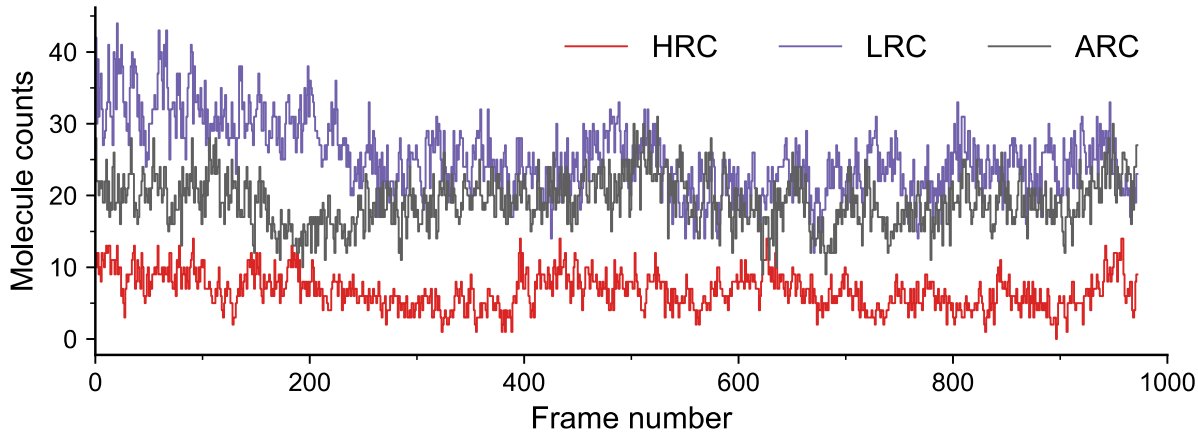

Fig. S48: RAS particle counts per frame. The number of molecules counted for each frame (1-1000 frames) in the single particle tracking movies obtained for KRAS on HRC, LRC and ARC.

Fig. S49A, B and C represent the mean square displacement (MSD) plots calculated from several thousands of trajectories measured for lipid and RAS respectively and step-length distribution for RAS on HRC (red) and LRC (blue) bilayer. The shape of the MSD plot characterizes the dynamic behavior of the particle (117). A straight line is indicative of free diffusion whereas a bent line represents confinement. Similarly, step-length distribution shows the histogram of the step sizes that single particles take in subsequent frames (91). If particles undergo free diffusion, they take larger steps whereas if particles follow confined diffusion, they take shorter steps. In our studies, the relatively straight MSD plots calculated for lipids indicate free diffusion of lipids on both HRC and LRC. This confirms that our working experimental lipid bilayer is a flat and simple reconstituted lipid bilayer lacking the complexity of real plasma membrane such as transmembrane proteins and actin cytoskeleton. On the other hand, the bent MSD plots for RAS on both HRC and LRC indicate confined diffusion of RAS with greater degree of confinement on HRC. Likewise, the step-length distribution of RAS on HRC features shorter step jumps compared to RAS on LRC eluding greater confinement, complimentary to the MSD plots. We hypothesize that the source of confinement comes from RAS organization into clusters on complex 8 lipid bilayer and not from lipid organization.

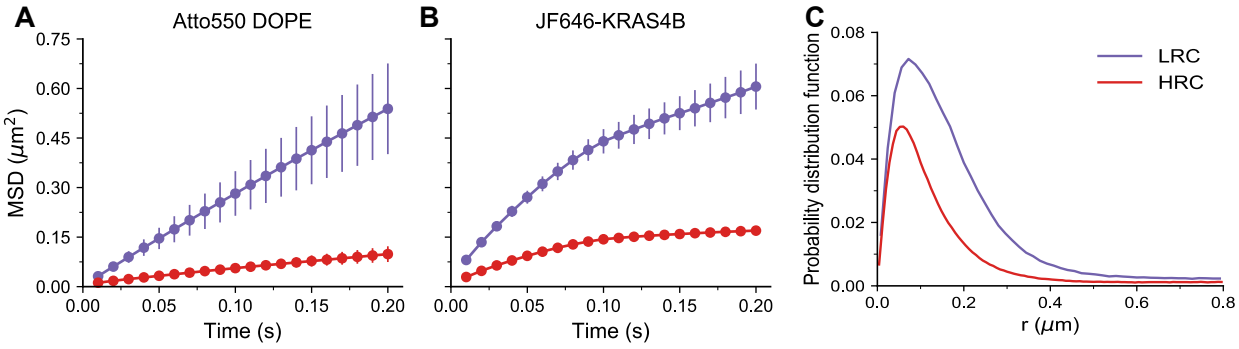

Fig. S49: Single Molecule Tracking of RAS on HRC/LRC. (A) Mean-square displacement (MSD) plots calculated from tracks obtained for Atto550 DOPE on LRC (blue) and HRC (red). (B) Mean square displacement plots calculated from tracks obtained for JF646 labeled KRAS4b on LRC (blue) and HRC (red). (C) Step-length distribution analysis of tracks collected for JF646 labeled KRAS4b on LRC (blue) and HRC (red).

Since the translational diffusion of RAS and lipids on two-dimensional lipid-bilayer is a stochastic process, both the MSD plot and step size distribution provide an ensemble-average value of the mobility of the particles on the membrane. In order to explore the underlying multiple diffusion states within the trajectory ensemble and their corresponding diffusion coefficients, inter-state transition probability, and dwell time, we performed HMM analysis for RAS tracking data. The results are summarized in Fig. S50. The HMM analysis predicted 3-state diffusion model—a fast state, an intermediate state, and a slow state for both LRC (Fig. S50A) and HRC (Fig. S50B). Although the 3-state diffusion model resembles the live cell data, the nature of the individual diffusion state is quite different. In LRC, the intermediate state has the highest occupancy whereas in HRC, the slow state is the most dominant state, as shown in Fig. S50D and Fig. S50E . In both lipid systems, the fast sate is very short lived followed by the intermediate and then the slow state. RAS undergoes rapid transition from the fast state to the intermediate state while the transition probability from the intermediate to fast state decreases by

4-fold and barely exist from the intermediate to the slow state, and from the slow to the fast state and vice versa (Fig. S50F). The rapid transitory nature of the fast state led us to attribute the fast state comes from the free diffusion of RAS on the lipid bilayer whereas based on the longer dwell time and smaller diffusion coefficient, we propose that the intermediate state and the slow state arise from RAS molecules that reside on the membrane and organize into nanoclusters.

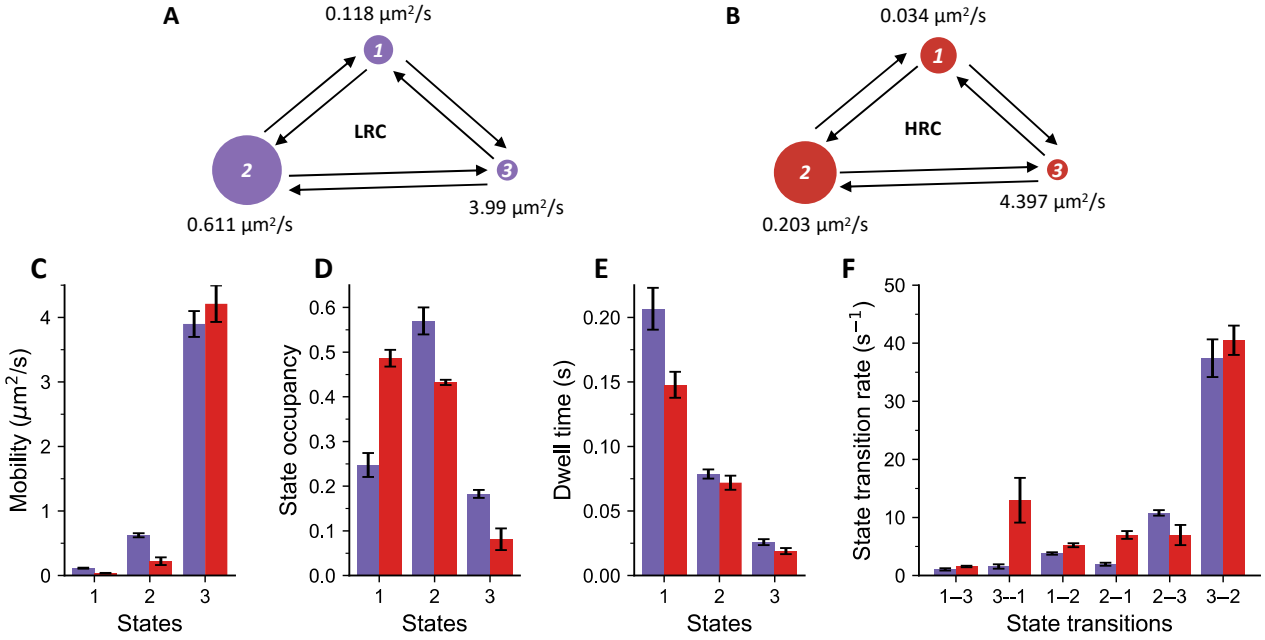

Fig. S50: HMM analysis of RAS diffusion states on LRC and HRC. (A) A representative 3-state diffusion model predicted by HMM analysis for RAS on LRC. (B) A representative 3-state diffusion model predicted by HMM analysis for RAS on HRC. (C) The average diffusion coefficients calculated for the three diffusive states of RAS on LRC (blue) and HRC (red). (D) The fractional occupancy of three diffusion states of percent state occupancy calculated for RAS on LRC (blue) and HRC (red). (E) The dwell time that RAS spends at each diffusive state on LRC (blue) and HRC (red). (F) The probability that RAS transitions between the three states for RAS on LRC (blue) and HRC (red).

We also observed differences in the diffusion coefficient of all three states between HRC and LRC (Fig. S50C). The diffusion coefficients of the intermediate and slow states reduced by more than 60% on HRC compared to LRC. The slowed and more confined diffusion suggests increased molecular crowding of RAS on HRC, partly driven by the more negatively charged HRC. Also, the state occupancy of the slow state increased substantially. This could be either from increased number of immobilized RAS on glass surface due to unspecific interaction with glass surface or from the formation of relatively large nanodomains or clusters of RAS mediated by the intermediate state. Although it is not a direct evidence of increased RAS multimerization on HRC, it does possess an attractive molecular assembly process driving towards RAS multimerization.

To test if RAS multimerization does occur on HRC, we carried out RAS density dependence experiments. We incorporated RAS at increasing concentrations of 10, 50, 70, 100, 200 and 1000 nM onto HRC and performed SPT studies. We used HMM analysis to determine the changes in the fractional occupancy of the three states. The results are shown in Fig. S51 along with result

for 1000 nM RAS on LRC. At all concentrations of KRAS, the state occupancies of the slow and the intermediate state are greater than the state occupancy of the fast state on HRC compared to LRC. Most importantly, in HRC, as RAS concentration increased, the state occupancies of slow state increased as well. This change was most significant between 10 nM and 50 nM RAS concentrations and only marginal at concentration higher than 50 nM. Hence, we can assume that RAS multimerization process reaches equilibrium at 50 nM.

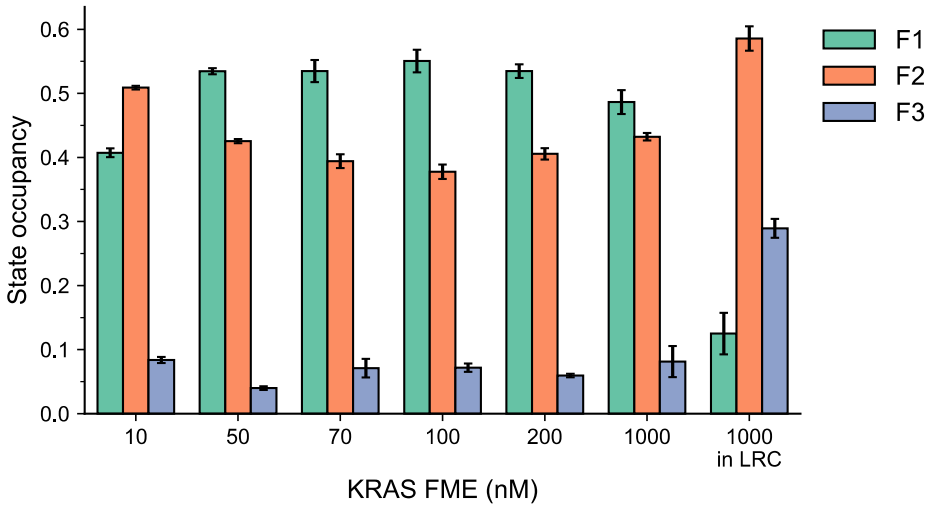

Fig. S51: Fractional occupancy of the three states from HMM analysis. The state occupancies of the slow (F1), intermediate (F2) and fast (F3) diffusion state calculated from the vbSPT HMM analysis of the trajectories collected for KRAS FME at various concentration on HRC.

2.8.5. Preferential Binding Coefficients of RAS Monomers vs. Dimers

Computations of the preferential binding coefficient, δ_{Lipid}, are defined in Section 1.3.7. Profiles of δ_{PIP2} are shown in Fig. 4d, and δ_{Lipid} profiles for other inner-leaflet lipids are shown in Fig. S52.

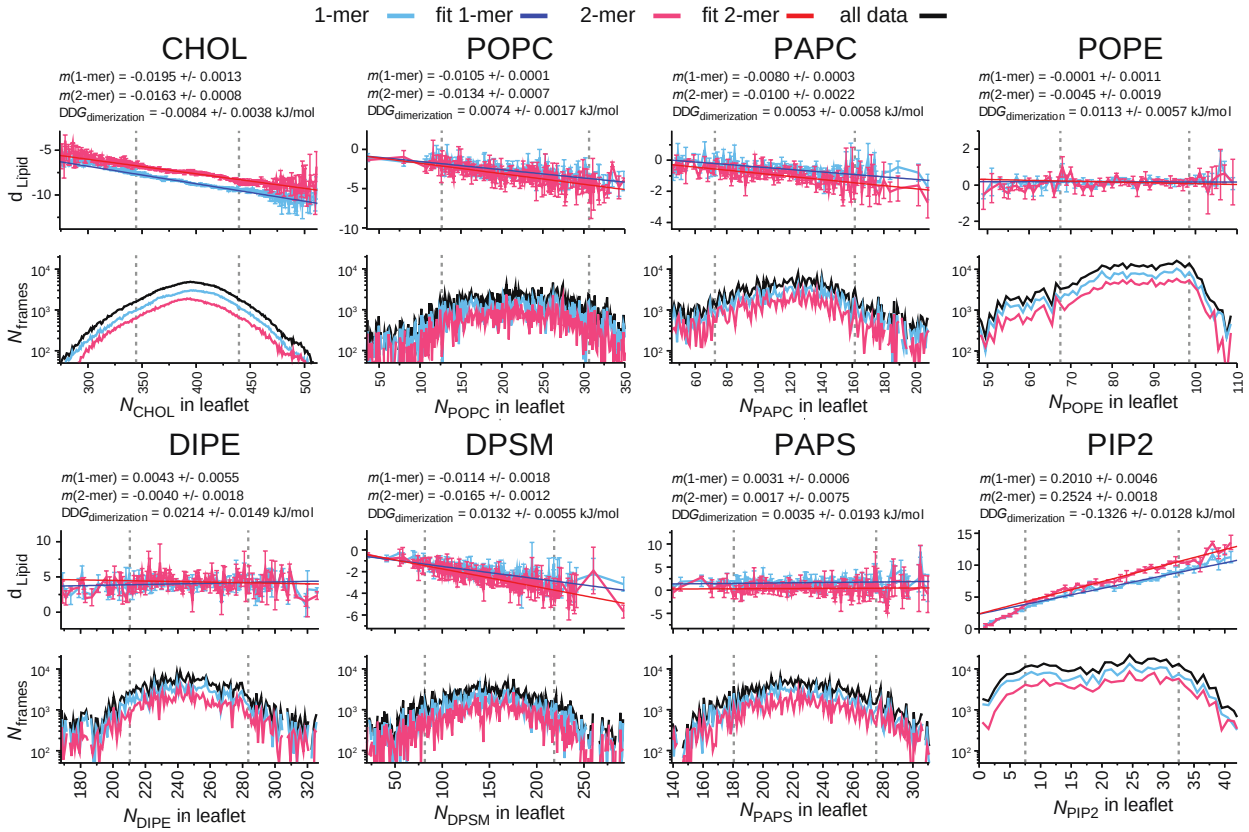

Fig. S52: Preferential binding coefficients of inner-leaflet lipids to RAS. For each lipid species, the top panel shows δ_{Lipid} as a function of N_{Lipid}, along with linear fits, fitted slopes (m), and $\Delta\Delta G$ values, as outlined in Section 1.3.7. Bottom panels show histograms of the number of frames used for each value of N_{Lipid}. Vertical dashed grey lines in all figures enclose the range on which linear fits were conducted.

2.9.RAS-RAS Interactions

In principle, CG simulations with more than one RAS per patch are capable of identifying preferred interfaces and lipid dependence of protein-protein interactions. However, the generation of CG systems from the macro model presents at least two challenges in cases where two RAS molecules are initially constructed in contact or close proximity. The first challenge relates to difficulties in sampling protein-protein association and dissociation events due to the long time-scales required (118), coupled with the fact that Martini proteins with elastic networks are somewhat too sticky (35). In the microsecond time-scale Martini simulations reported here, sampled configurations frequently remain similar to the initial pose when two RAS proteins are built in contact (
Fig**. S53**).

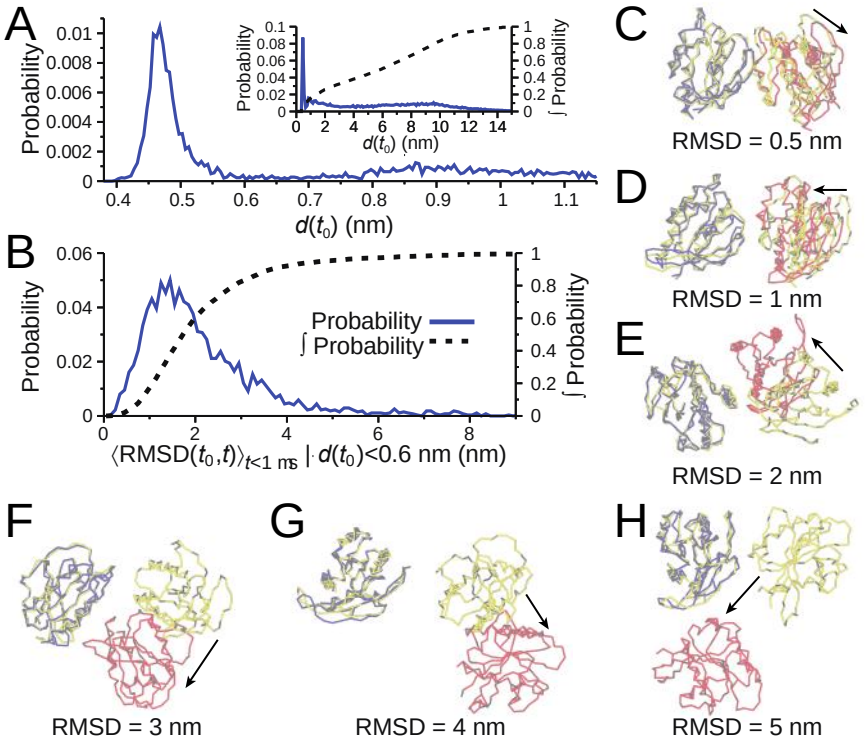

Fig. S53: Patches initiated with two RAS in contact have relatively static protein-protein interfaces. (A) Probability distribution of initial minimum inter-protein bead distance, $d(t_0)$, in simulations with two RAS. Inset shows a longer range of $d(t_0)$ and includes cumulative probability from 0 to $d(t_0)$, ∫Probability. (B) Probability distribution of the average RMSD of G-domain backbone beads in RAS molecule *j* between configurations at time 0 and time *t*, averaged over values of *t* in the first μs, after least-squares fitting on the G-domain backbone beads of RAS molecule *i*. Data only includes simulations for which $d(t_0) < 0.6$ nm (i.e., simulations with initial RAS-RAS contact). (C-H) Representative configurations of two RAS in simulations with $d(t_0) < 0.6$ nm. Initial arrangement of both RAS proteins shown in yellow. Configurations of the (blue) fitted RAS molecule *i* and (red) other RAS molecule *j* for snapshots with indicated RMSD values.

The second challenge relates to the lack of orientational information in the macro model. In these MuMMI simulations, the macro model RAS bead was parameterized to represent the farnesyl group. During the CG build procedure, candidate orientations of nearby RAS proteins are more likely to be rejected due to molecular overlap when the G-domain of one RAS is directed toward the other RAS. Hence, two RAS initially built close together have a built-in bias that favors HVR-to-HVR association over G-domain-to-G-domain association (Fig. S54A and B). Because the CG build procedure iterated random attempts to replace the second RAS molecule in event of unresolved clashes, most (but not all) of this bias is exerted on the second RAS, which tends to point its HVR toward the G-domain of the first RAS molecule when the two RAS are built in or close to contact (Fig. S54B). If, in contrast, the macro model point represented the G-domain, then we expect that the bias would be inverted such that it favored G-G contacts. One approach to removing this bias is to define a cutoff distance between the farnesy groups of two RAS proteins that is large enough to make RAS-RAS contact or spatial overlap theoretically impossible (even with an extended HVR) and exclude all simulations in which the initial farnesyl-farnesyl distance was below this cutoff. However, doing so is undesirable because it

excludes > 95% of the CG simulations that underwent RAS-RAS contact. Therefore, we use an empirical approach to eliminate the apparent bias. Specifically, RAS orientation appears unbiased in initial system construction when we exclude all simulations in which the Cartesian xy (global membrane plane) component of the initial intermolecular backbone-backbone bead distance is <= 4.5 nm (Fig. S54A and B). Fitting the probability distribution of this initial minimum distance to an expected linear function with a biased Gaussian component also indicates that the 4.5 nm cutoff removes most of the bias (Fig. S54C). Finally, the orientations of the two RAS molecules appear unbiased and without relation to initial separation when this exclusion criterion is applied (Fig. S54D). From a total of 25,489 simulations with two RAS, of which 9,611 exhibited RAS-RAS contact, this unbiased set retains 15,828 2-RAS simulations, of which 2,037 undergo RAS-RAS contact. We used this set of 15,828 CG simulations with 2 RAS molecules to evaluate changes in RAS arrangement and local lipid composition as a function of RAS separation.

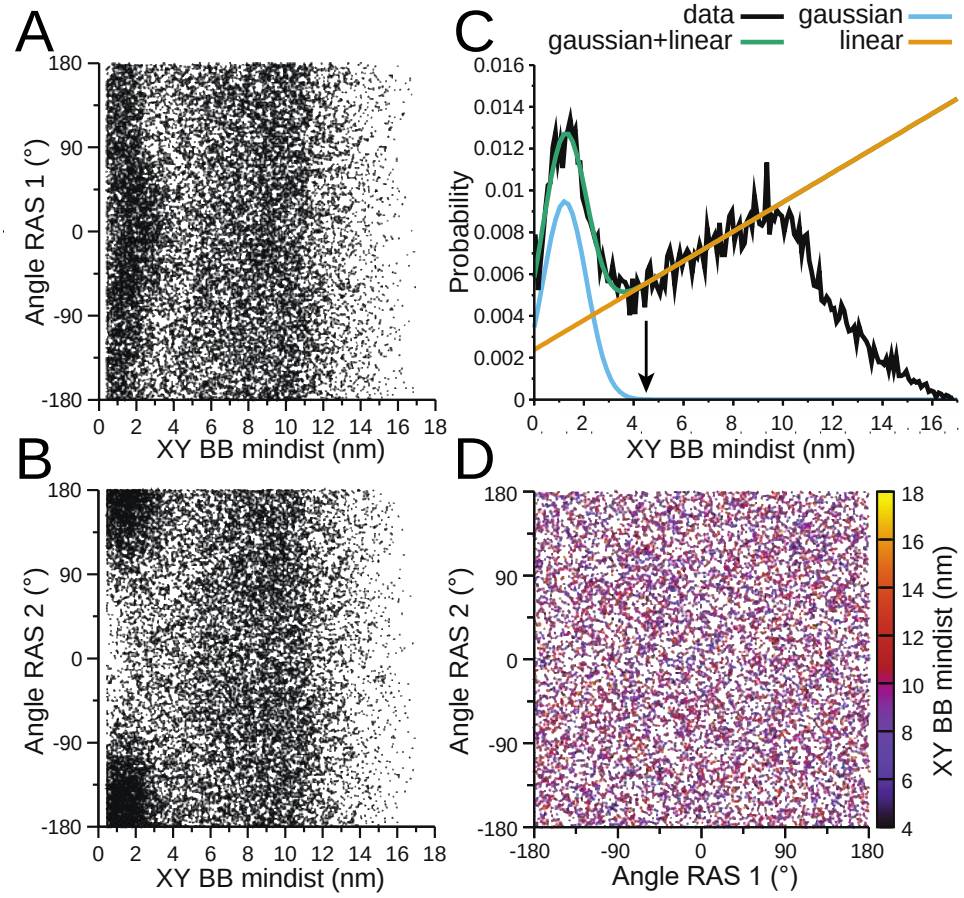

Fig. S54: Selection of CG simulations with two RAS for analysis. (A and B) Arrangements of the RAS molecule whose CG model was constructed (A) first and (B) second as a function of initial minimum intermolecular BB bead distance in the global bilayer plane (XY BB mindist). Angle is the directed angle from the $G_{COM}(1) \rightarrow G_{COM}(2)$ vector to the $G_{COM}(N) \rightarrow C185(N)$ BB bead vector, where the subscript COM indicates center of mass. (C) Probability distribution of XY BB mindist from all segment 2 simulations with 2 RAS. A Gaussian + linear function is fit for mindist <= 9 nm. (D) Angle RAS 1 vs. angle RAS 2 for all Segment 2 2-RAS initial configurations with XY BB mindist >4.5 nm. Points are colored by XY BB mindist.

3. Supplementary References

1. F. Di Natale *et al.* (2019) A massively parallel infrastructure for adaptive multiscale simulations: modeling RAS initiation pathway for cancer. in *The International Conference for High Performance Computing, Networking, Storage and Analysis* (ACM, Denver, Colorado), p 57.
2. H. Bhatia *et al.*, Machine Learning Based Dynamic-Importance Sampling for Adaptive Multiscale Simulations. *Nature Machine Intelligence* **3**, 401-409 (2021).
3. X. Zhang *et al.*, ddcMD: A fully GPU-accelerated molecular dynamics program for the Martini force field. *J. Chem. Phys.* **153**, 045103 (2020).
4. F. Di Natale (2019) Maestro Workflow Conductor. (Lawrence Livermore National Laboratory). https://github.com/LLNL/maestrowf
5. D. H. Ahn *et al.* (2018) Flux: Overcoming Scheduling Challenges for Exascale Workflows. in *IEEE/ACM Workflows in Support of Large-Scale Science (WORKS)* (IEEE, Dallas, Texas), pp 10-19.
6. TOP500 (2019) TOP500 Supercomputer Sites | November 2019. https://www.top500.org/lists/2019/11/
7. F. H. Streitz, J. N. Glosli, M. V. Patel, Beyond Finite-Size Scaling in Solidification Simulations. *Phys. Rev. Lett.* **96**, 225701 (2006).
8. J. N. Glosli *et al.* (2007) Extending Stability Beyond CPU Millennium: A Micron-scale Atomistic Simulation of Kelvin-Helmholtz Instability. in *The 2007 ACM/IEEE Conference on Supercomputing* (ACM, Reno, Nevada), p 58.
9. F. H. Streitz *et al.* (2005) 100+ TFlop Solidification Simulations on BlueGene/L. in *The 2005 ACM/IEEE Conference on Supercomputing* (ACM, Washington DC).
10. S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, A. H. de Vries, The MARTINI Force Field: Coarse Grained Model for Biomolecular Simulations. *J. Phys. Chem. B* **111**, 7812-7824 (2007).
11. J. Glosli *et al.* (2021) ddcMD. (Lawrence Livermore National Laboratory). https://github.com/LLNL/ddcmd
12. L. Stanton *et al.*, Dynamic Density Functional Theory of Multicomponent Cellular Membranes. *arXiv*:2112.08651 [physics.bio-ph] (2021).
13. U. M. B. Marconi, P. Tarazona, Dynamic density functional theory of fluids. *The Journal of chemical physics* **110**, 8032-8044 (1999).
14. L. S. Ornstein, Accidental deviations of density and opalescence at the critical point of a single substance. *Proc. Akad. Sci.* **17**, 793-806 (1914).
15. D. A. McQuarrie, *Statistical Mechanics* (University Science Books, 2000).
16. D. Gaston, C. Newman, G. Hansen, D. Lebrun-Grandié, MOOSE: A parallel computational framework for coupled systems of nonlinear equations. *Nucl. Eng. Des.* **239**, 1768-1778 (2009).
17. T. A. Wassenaar, H. I. Ingólfsson, R. A. Böckmann, D. P. Tieleman, S. J. Marrink, Computational Lipidomics with insane : A Versatile Tool for Generating Custom Membranes for Molecular Simulations. *Journal of Chemical Theory and Computation* **11**, 2144-2155 (2015).
18. M. J. Abraham *et al.*, GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX* **1-2**, 19-25 (2015).
19. D. H. de Jong, S. Baoukina, H. I. Ingólfsson, S. J. Marrink, Martini straight: Boosting performance using a shorter cutoff and GPUs. *Comput. Phys. Commun.* **199**, 1-7 (2016).
20. J. A. Barker, R. O. Watts, Monte Carlo studies of the dielectric properties of water-like models. *Molecular Physics* **26**, 789-792 (1973).
21. W. C. Swope, H. C. Andersen, P. H. Berens, K. R. Wilson, A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *The Journal of Chemical Physics* **76**, 637-649 (1982).

22. H. C. Andersen, Rattle: A ``velocity'' version of the shake algorithm for molecular dynamics calculations. *Journal of Computational Physics* **52**, 24-34 (1983).
23. M. P. Allen, D. J. Tildesley, *Computer Simulation of Liquids* (Clarendon Press, 1989).
24. N. Michaud-Agrawal, E. J. Denning, T. B. Woolf, O. Beckstein, MDAnalysis: A toolkit for the analysis of molecular dynamics simulations. *J. Comput. Chem.* **32**, 2319-2327 (2011).
25. R. Gowers *et al.* (2016) MDAnalysis: A Python Package for the Rapid Analysis of Molecular Dynamics Simulations. in *The 15th Python in Science Conference* (Austin, Texas), pp 102-109.
26. A. Kusumi, K. G. N. Suzuki, R. S. Kasai, K. Ritchie, T. K. Fujiwara, Hierarchical mesoscale domain organization of the plasma membrane. *Trends Biochem. Sci.* **36**, 604-615 (2011).
27. F. D. Tsai *et al.*, K-Ras4A splice variant is widely expressed in cancer and uses a hybrid membrane-targeting motif. *Proc. Natl. Acad. Sci. U. S. A.* **112**, 779-784 (2015).
28. C. J. Mageean, J. R. Griffiths, D. L. Smith, M. J. Clague, I. A. Prior, Absolute quantification of endogenous Ras isoform abundance. *PLoS One* **10**, e0142674 (2015).
29. N. Castillo, L. Monticelli, J. Barnoud, D. P. Tieleman, Free energy of WALP23 dimer association in DMPC, DPPC, and DOPC bilayers. *Chemistry and Physics of Lipids* **169**, 95-105 (2013).
30. Q. Wang *et al.*, Mutant proteins as cancer-specific biomarkers. *Proceedings of the National Academy of Sceinces* **108**, 2444-2449 (2011).
31. S. J. Marrink, D. P. Tieleman, Perspective on the Martini model. *Chem. Soc. Rev.* **42**, 6801-6822 (2013).
32. S. J. Marrink *et al.*, Computational Modeling of Realistic Cell Membranes. *Chemical Reviews* **119**, 6184-6226 (2019).
33. V. Corradi *et al.*, Emerging Diversity in Lipid–Protein Interactions. *Chemical Reviews* **119**, 5775-5848 (2019).
34. H. I. Ingólfsson *et al.*, The power of coarse graining in biomolecular simulations. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **4**, 225-248 (2014).
35. R. Alessandri *et al.*, Pitfalls of the Martini Model. *Journal of Chemical Theory and Computation* **15**, 5448-5460 (2019).
36. P. C. T. Souza *et al.*, Martini 3: a general purpose force field for coarse-grained molecular dynamics. *Nature Methods* **18**, 382-388 (2021).
37. G. van Meer, D. R. Voelker, G. W. Feigenson, Membrane lipids: where they are and how they behave. *Nature Rev. Mol. Cell Biol.* **9**, 112-124 (2008).
38. J. L. Sampaio *et al.*, Membrane lipidome of an epithelial cell line. *Proceedings of the National Academy of Sciences of the United States of America* **108**, 1903-1907 (2011).
39. J. H. Lorent *et al.*, Plasma membranes are asymmetric in lipid unsaturation, packing and protein shape. *Nature Chemical Biology* **16**, 644-652 (2020).
40. E. Sezgin, I. Levental, S. Mayor, C. Eggeling, The mystery of membrane organization: composition, regulation and roles of lipid rafts. *Nature Rev. Mol. Cell Biol.* **18**, 361-374 (2017).
41. O. G. Mouritsen, M. Bloom, Mattress model of lipid-protein interactions in membranes. *Biophysical Journal* **46**, 141-153 (1984).
42. D. Lingwood, K. Simons, Lipid rafts as a membrane-organizing principle. *Science* **327**, 46-50 (2010).
43. J. N. Israelachvili, Refinement of the fluid-mosaic model of membrane structure. *Biochimica et Biophysica Acta* **469**, 221-225 (1977).
44. H. I. Ingólfsson *et al.*, Capturing Biologically Complex Tissue-Specific Membranes at Different Levels of Compositional Complexity. *J. Phys. Chem. B* **124**, 7819-7829 (2020).
45. S. J. Marrink, A. H. De Vries, A. E. Mark, Coarse grained model for semiquantitative lipid simulations. *The Journal of Physical Chemistry B* **108**, 750-760 (2004).
46. H. I. Ingólfsson *et al.*, Lipid Organization of the Plasma Membrane. *Journal of American Chemical Society* **136**, 14554-14559 (2014).
47. M. N. Melo, H. I. Ingólfsson, S. J. Marrink, Parameters for Martini sterols and hopanoids based on a virtual-site description. *The Journal of Chemical Physics* **143**, 243152 (2015).

48. F. Sun *et al.*, Molecular Mechanism for Bidirectional Regulation of CD44 for Lipid Raft Affiliation by Palmitoylations and PIP2. *J. Biol. Chem.* **278**, 10831-10841 (2019).
49. C. A. Lopez, Z. Sovova, F. J. van Eerden, A. H. De Vries, S. J. Marrink, Martini force field parameters for glycolipids. *Journal of Chemical Theory and Computation* **9**, 1694-1708 (2013).
50. D. K. Simanshu, D. V. Nissley, F. McCormick, RAS Proteins and Their Regulators in Human Disease. *Cell* **170**, 17-33 (2017).
51. M. Spoerner, C. Herrmann, I. R. Vetter, H. R. Kalbitzer, A. Wittinghofer, Dynamic properties of the Ras switch I region and its importance for binding to effectors. *Proceedings of the National Academy of Sceinces* **98**, 4944-4949 (2001).
52. J. A. Parker, A. Y. Volmar, S. Pavlopoulos, C. Mattos, K-Ras Populates Conformational States Differently from Its Isoform H-Ras and Oncogenic Mutant K-RasG12D. *Structure* **26**, 810-820 (2018).
53. A. Cruz-Migoni *et al.*, Structure-based development of new RAS-effector inhibitors from a combination of active and inactive RAS-binding compounds. *Proc. Natl. Acad. Sci. USA* **116**, 2545-2550 (2019).
54. Anonymous (2019) Molecular Operating Environment (MOE), v2019.01. (Chemical Computing Group ULC, Montreal, QC, Canada, H3A 2R7).
55. J. Huang, A. D. MacKerell Jr, CHARMM36 all-atom additive protein force field: Validation based on comparison to NMR data. *Journal of Computational Chemistry* **34**, 2135-2145 (2013).
56. W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, M. L. Klein, Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **79**, 926-935 (1983).
57. D. A. Case *et al.*, AMBER 2016 Reference Manual. *University of California: San Francisco, CA, USA*, 1-923 (2016).
58. L. Monticelli *et al.*, The MARTINI Coarse-Grained Force Field: Extension to Proteins. *J. Chem. Theory Comput.* **4**, 819-834 (2008).
59. S. Dharmaiah *et al.*, Structures of N-terminally processed KRAS provide insight into the role of N-acetylation. *Scientific Reports* **9** (2019).
60. J. J. Uusitalo, H. I. Ingólfsson, P. Akhshi, D. P. Tieleman, S. J. Marrink, Martini Coarse-Grained Force Field: Extension to DNA. *J. Chem. Theory Comput.* **11**, 3932-3945 (2015).
61. J. J. Uusitalo, H. I. Ingólfsson, S. J. Marrink, I. Faustino, Martini Coarse-Grained Force Field: Extension to RNA. *Biophysical Journal* **113**, 246-256 (2017).
62. T. Travers *et al.*, Molecular recognition of RAS/RAF complex at the membrane: Role of RAF cysteine-rich domain. *Scientific Reports* **8**, 8461 (2018).
63. Y. Atsmon-Raz, D. P. Tieleman, Parameterization of Palmitoylated Cysteine, Farnesylated Cysteine, Geranylgeranylated Cysteine, and Myristoylated Glycine for the Martini Force Field. *The Journal of Physical Chemistry B* **121**, 11132-11143 (2017).
64. C. Neale, A. E. García, Methionine 170 is an Environmentally Sensitive Membrane Anchor in the Disordered HVR of K-Ras4B. *J. Phys. Chem. B* **122**, 10086-10096 (2018).
65. F. Noé, H. Wu, J.-H. Prinz, N. Plattner, Projected and hidden Markov models for calculating kinetics and metastable states of complex molecules. *The Journal of Chemical Physics* **139**, 184114 (2013).
66. C. Schütte, A. Fischer, W. Huisinga, P. Deuflhard, A direct approach to conformational dynamics based on hybrid Monte Carlo. *Journal of Computational Physics* **151**, 146-168 (1999).
67. J.-H. Prinz *et al.*, Markov models of molecular kinetics: Generation and validation. *The Journal of chemical physics* **134**, 174105 (2011).
68. V. S. Pande, K. Beauchamp, G. R. Bowman, Everything you wanted to know about Markov State Models but were afraid to ask. *Methods* **52**, 99-105 (2010).
69. P. Deuflhard, M. Weber, Robust Perron cluster analysis in conformation dynamics. *Linear algebra and its applications* **398**, 161-184 (2005).
70. L. R. Welch, Hidden Markov models and the Baum-Welch algorithms. *IEEE Trans. Theory Soc. Newsl.* **53**, 1-13 (2003).

71. M. K. Scherer *et al.*, PyEMMA 2: A software package for estimation, validation, and analysis of Markov models. *Journal of Chemical Theory and Computation* **11**, 5525-5542 (2015).
72. J. A. Hartigan, M. A. Wong, Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* **28**, 100-108 (1979).
73. C. Doersch (2016) Tutorial on Variational Autoencoders. (arXiv:1606.05908).
74. P. T. Bremer *et al.*, Interactive Exploration and Analysis of Large Scale Simulations Using Topology-based Data Segmentation. *IEEE Trans. on Visualization and Computer Graphics* **17**, 1307-1324 (2011).
75. W. Widanagamaachchi , C. Christensen, V. Pascucci, P.-T. Bremer (2012) Interactive exploration of large-scale time-varying data using dynamic tracking graphs. in *IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, pp 9-17.
76. S. Liu, R. Anirudh, J. J. Thiagarajan, P.-T. Bremer, Uncovering interpretable relationships in high-dimensional scientific data through function preserving projections. *Machine Learning: Science and Technology* **1**, 045016 (2020).
77. F. Chollet (2015) Keras. https://github.com/fchollet/keras
78. S. K. Fetics *et al.*, Allosteric effects of the oncogenic RasQ61L mutant on Raf-RBD. *Structure* **23**, 505-516 (2015).
79. J. B. Grimm, T. D. Gruber, G. Ortiz, T. A. Brown, L. D. Lavis, Virginia Orange: A Versatile, Red-Shifted Fluorescein Scaffold for Single- and Dual-Input Fluorogenic Probes. *Bioconjugate Chemistry* **27**, 474-480 (2016).
80. J.-Y. Tinevez *et al.*, TrackMate: An open and extensible platform for single-particle tracking. *Methods* **115**, 80-90 (2017).
81. M. E. Beheiry, M. Dahan, J.-B. Masson, InferenceMAP: mapping of single-molecule dynamics with Bayesian inference. *Nature Methods* **12**, 594-595 (2015).
82. S. A. Doyle, Ed., *High Throughput Protein Expression and Purification: Methods and Protocols* (Humana Press, 2009), 1 Ed10.1007/978-1-59745-196-3.
83. W. Kabsch, XDS. *Acta Crystallographica Section D* **66**, 125–132 (2010).
84. W. K. Gillette *et al.*, Farnesylated and methylated KRAS4b: high yield production of protein suitable for biophysical studies of prenylated protein-lipid interactions. *Sci. Rep.* **5**, 15916 (2015).
85. P. S. Cremer, S. G. Boxer, Formation and Spreading of Lipid Bilayers on Planar Glass Supports. *The Journal of Physical Chemistry B* **103**, 2554-2559 (1999).
86. D. Necas, P. Klapetek, Gwyddion: An open-source software for SPM data analysis. *Open Physics* **10**, 181 (2012).
87. J. K. Chung, Y. K. Lee, H. Y. Lam, J. T. Groves, Covalent Ras Dimerization on Membrane Surfaces through Photosensitized Oxidation. *J Am Chem Soc* **138**, 1800-1803 (2016).
88. P. Dedecker, S. Duwe, R. K. Neely, J. Zhang, Localizer: fast, accurate, open-source, and modular software package for superresolution microscopy. *Journal of Biomedical Optics* **17** (2012).
89. F. Persson, M. Lindén, C. Unoson, J. Elf, Extracting intracellular diffusive states and transition rates from single-molecule tracking data. *Nature Methods* **10**, 265-269 (2013).
90. A. Matysik, R. S. Kraut, TrackArt: the user friendly interface for single molecule tracking data analysis and simulation applied to complex diffusion in mica supported lipid bilayers. *BMC Research Notes* **7** (2014).
91. W.-C. Lin *et al.*, H-Ras forms dimers on membrane surfaces via a protein-protein interface. *Proceedings of the National Academy of Sceinces* **111**, 2996-3001 (2014).
92. B. Lakshman *et al.*, Quantitative biophysical analysis defines key components modulating recruitment of the GTPase KRAS to the plasma membrane. *JBC* **294**, 2193-2207 (2019).
93. J. H. Lorent *et al.*, The Molecular and Structural Asymmetry of the Plasma Membrane. *bioRxiv* **preprint** (2019).
94. K. Scheffzek *et al.*, The Ras-Byr2RBD complex: structural basis for Ras effector recognition in yeast. *Structure* **9**, 1043-1050 (2001).

95. L. Huang, F. Hofer, G. S. Martin, S.-H. Kim, Structural basis for the interaction of Ras with RaIGDS. *Nature Structural Biology* **5**, 422 (1998).
96. B. Stieglitz *et al.*, Novel type of Ras effector interaction established between tumour suppressor NORE1A and Ras switch II. *The EMBO Journal* **27**, 1995-2005 (2008).
97. T. D. Bunney *et al.*, Structural and mechanistic insights into ras association domains of phospholipase C epsilon. *Mol. Cell* **21**, 495-507 (2006).
98. R. Qamra, S. R. Hubbard, Structural basis for the interaction of the adaptor protein grb14 with activated ras. *PloS One* **8**, e72473 (2013).
99. M. E. Pacold *et al.*, Crystal Structure and Functional Analysis of Ras Binding to Its Effector Phosphoinositide 3-Kinase γ. *Cell* **103**, 931-944 (2000).
100. K. Scheffzek *et al.*, The Ras-RasGAP Complex: Structural Basis for GTPase Activation and Its Loss in Oncogenic Ras Mutants. *Science* **277**, 333-339 (1997).
101. S. M. Margarit *et al.*, Structural Evidence for Feedback Activation by Ras·GTP of the Ras-Specific Nucleotide Exchange Factor SOS. *Cell* **112**, 685-695 (2003).
102. S. Dharmaiah *et al.*, Structural basis of recognition of farnesylated and methylated KRAS4b by PDEδ. *Proceedings of the National Academy of Sciences* **113**, E6766-E6775 (2016).
103. P. Prakash, A. A. Gorfe, Membrane orientation dynamics of lipid-modified small GTPases. *Small GTPases* **8**, 129-138 (2017).
104. S. Muratcioglu *et al.*, GTP-Dependent K-Ras Dimerization. *Structure* **23**, 1325-1335 (2015).
105. T. G. Bivona *et al.*, PKC Regulates a Farnesyl-Electrostatic Switch on K-Ras that Promotes its Association with Bcl-Xl on Mitochondria and Induces Apoptosis. *Mol. Cell* **21**, 481-493 (2006).
106. M. Vögele, G. Hummer, Divergent Diffusion Coefficients in Simulations of Fluids and Lipid Membranes. *The Journal of Physical Chemistry B* **120**, 8722-8732 (2016).
107. A. Filippov, G. Oradd, G. Lindblom, The Effect of Cholesterol on the Lateral Diffusion of Phospholipids in Oriented Bilayers. *Biophysical Journal* **84**, 3079-3086 (2003).
108. M. Collins, Interleaflet Coupling Mechanisms in Bilayers of Lipids and Cholesterol. *Biophysical Journal* **94**, L32-34 (2008).
109. R. M. Venable *et al.*, Lipid and Peptide Diffusion in Bilayers: The Saffman–Delbrück Model and Periodic Boundary Conditions. *The Journal of Physical Chemistry B* **121**, 3443-3457 (2017).
110. A. Ortega, D. Amores, J. Torre, Prediction of Hydrodynamic and Other Solution Properties of Rigid Proteins from Atomic- and Residue-Level Models. *Biophys. J.* **101**, 892-898 (2011).
111. R. Hermann Alfons Winter, S. Patra, N. Erwin, Translational Dynamics of Lipidated Ras Proteins in the Presence of Crowding Agents and Compatible Osmolytes. *Chem. Phys. Chem.* **17** (2016).
112. A. Werkmuller, G. Triola, H. Waldmann, R. Winter, Rotational and Translational Dynamics of Ras Proteins upon Binding to Model Membrane Systems. *Chem. Phys. Chem.* **14** (2013).
113. C. Arnarez, S. J. Marrink, X. Periole, Molecular mechanism of cardiolipin-mediated assembly of respiratory chain supercomplexes. *Chemical Science* **7**, 4435-4443 (2016).
114. G. Bussi, D. Donadio, M. Parrinello, Canonical sampling through velocity rescaling. *The Journal of Chemical Physics* **126**, 014101 (2007).
115. D. Goswami *et al.*, Membrane interactions of the globular domain and the hypervariable region of KRAS4b defines its unique diffusion behavior. *eLife* **9**, e47654 (2020).
116. Y. Lee *et al.*, High-throughput single-particle tracking reveals nested membrane nanodomain organization that dictates Ras diffusion and tracking. *bioRxiv* **preprint** (2019).
117. M. J. Saxton, K. Jacobson, SINGLE-PARTICLE TRACKING:Applications to Membrane Dynamics. *Annual Review of Biophysics and Biomolecular Structure* **26**, 373-399 (1997).
118. A. C. Pan *et al.*, Atomic-level characgterization of protein-protein association. *Proceedings of the National Academy of Sciences* **116**, 4244-4249 (2019).

4. Supplementary Appendix A

Crystallographic data collection and refinement statistics. Statistics for the highest-resolution shell are shown in parentheses.

Data collection parameters	GMPPNP bound KRAS4b (1-169)
Wavelength (Å)	0.97872
Resolution range (Å)	50.0 - 2.50 (2.56 - 2.50)
Space group	*C* 1 2 1
Unit cell (Å, º)	*a* = 69.21 *b* = 82.33, *c* = 88.14 90 β = 112.92
Unique reflections	15477 (1571)
Multiplicity	2.6
Completeness (%)	97.3 (97.3)
Mean I/sigma(I)	15.59 (2.02)
Wilson B-factor	46.28
R-merge (%)	5.3 (56.5)
Refinement parameters	
Resolution range (Å)	37.91 - 2.50 (2.69 - 2.50)
Reflections used in refinement	15475 (2965)
Reflections used for R-free	749 (138)
R-work	0.181 (0.269)
R-free	0.26.8 (0.345)
RMSD bonds (Å)	0.008
RMSD angles (deg)	1.071
Ramachandran favoured (%)	96.08
Ramachandran allowed (%)	2.40
Ramachandran outliers (%)	1.52
Average B-factor (Å^3)	64.79
macromolecules	65.40
ligands	46.64
solvent	49.10