

Article

Coupling Bulk Phase Separation of Disordered Proteins to Membrane Domain Formation in Molecular Simulations on a Bespoke Compute Fabric

Julian C. Shillcock^{1,*} , David B. Thomas², Jonathan R. Beaumont³, Graeme M. Bragg², Mark L. Vousden² and Andrew D. Brown²

¹ Blue Brain Project and Laboratory of Molecular and Chemical Biology of Neurodegeneration, Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland

² Department of Electronics and Computer Science, University of Southampton, Highfield, Southampton SO17 1BJ, UK; D.B.Thomas@southampton.ac.uk (D.B.T.); gmb@ecs.soton.ac.uk (G.M.B.); M.Vousden@soton.ac.uk (M.L.V.); adb@ecs.soton.ac.uk (A.D.B.)

³ Department of Electronic Engineering, Imperial College London, London SW7 2AZ, UK; jonathan.r.beaumont@gmail.com

* Correspondence: julian.shillcock@epfl.ch; Tel.: +41-21-693-9679

Abstract: Phospholipid membranes surround the cell and its internal organelles, and their multicomponent nature allows the formation of domains that are important in cellular signalling, the immune system, and bacterial infection. Cytoplasmic compartments are also created by the phase separation of intrinsically disordered proteins into biomolecular condensates. The ubiquity of lipid membranes and protein condensates raises the question of how three-dimensional droplets might interact with two-dimensional domains, and whether this coupling has physiological or pathological importance. Here, we explore the equilibrium morphologies of a dilute phase of a model disordered protein interacting with an ideal-mixing, two-component lipid membrane using coarse-grained molecular simulations. We find that the proteins can wet the membrane with and without domain formation, and form phase separated droplets bound to membrane domains. Results from much larger simulations performed on a novel non-von-Neumann compute architecture called POETS, which greatly accelerates their execution compared to conventional hardware, confirm the observations. Reducing the wall clock time for such simulations requires new architectures and computational techniques. We demonstrate here an inter-disciplinary approach that uses real-world biophysical questions to drive the development of new computing hardware and simulation algorithms.

Keywords: membrane domains; intrinsically-disordered protein; phase separation; biomolecular condensate; coarse-grained simulation; event-based computing; hardware-accelerated simulation; application-specific hardware



Citation: Shillcock, J.C.; Thomas, D.B.; Beaumont, J.R.; Bragg, G.M.; Vousden, M.L.; Brown, A.D. Coupling Bulk Phase Separation of Disordered Proteins to Membrane Domain Formation in Molecular Simulations on a Bespoke Compute Fabric. *Membranes* **2022**, *12*, 17. <https://doi.org/10.3390/membranes12010017>

Academic Editors: Peter G. Petrov and C. Peter Winlove

Received: 26 November 2021

Accepted: 21 December 2021

Published: 23 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Soft surfaces are ubiquitous in living cells and are typified by phospholipid membranes that bound the cell and its internal organelles [1]. Membranes allow compositionally-distinct environments for incompatible biochemistry and provide specialized surfaces to mediate material and information transport. Although viewed originally as merely structural boundaries, they are now recognized as active participants in cellular dynamics [2,3]. They undergo morphological transitions between distinct shapes in response to weak external influences [4], and in turn exert forces on membrane-associated particles or materials [5]. Their multicomponent nature allows regulated demixing of a subset of their component lipids and proteins to assemble into localized domains to carry out transient functions [3,6]. Pathological domain formation occurs when bacteria such as *E. Coli* and Shigelle release nm-sized rigid toxin particles in order to infect the cell [7]. Toxin particles adsorb to the

cellular plasma membrane, cluster into domains, and invaginate into the cell carrying their toxic payload that ultimately destroys the cell and aids the life cycle of the bacterium [8].

Another form of fluid interface that is increasingly recognized as crucial for compartmentalizing the cellular and nuclear cytoplasm is the surface of biomolecular condensates or membraneless organelles [9–12]. These are fluid droplets that form by liquid–liquid phase separation (LLPS) of a wide range of intrinsically-disordered proteins (IDPs) [13]. They have numerous functions in the cell, including regulation of biochemical reactions [14], sequestering stalled RNA translation in stress granules [15], chromatin organization in the nucleus [16], and organizing the presynaptic axon and postsynaptic density in neurons [17,18]. Unlike lipid membranes, the formation of biomolecular condensates (BCs) *in vivo* is usually reversible: The cell actively assembles them to perform a function and they subsequently melt away [19]. However, they can transition irreversibly into pathological rigid states in experiments [20–22], a transition that is hypothesized to be important for neurodegenerative diseases such as Alzheimer’s, Parkinson’s and Huntington’s disease [23]. Evidence is emerging that they are implicated in the dysregulation of cellular processes occurring in cancer [24–26].

The cellular cytoplasm is a crowded place: It is estimated that nowhere in a cell is more than 50 nm from a lipid membrane [1]. The combination of two-dimensional fluid lipid membranes, which can be relatively easily deformed, and three-dimensional fluid protein droplets could theoretically lead to a variety of morphologies [4,27–29]. Membranes regulate the formation of BCs that help assemble protein signaling clusters [30–32], and mediate immune cell signalling [33–35]. Recent experiments show that phase separation of T cell proteins at the surface of giant unilamellar lipid vesicles drives lipid domain formation that in turn is able to recruit additional proteins to the domains [36]. This provides a starting point for rational design of artificial organelles [37].

In this work, we qualitatively explored the equilibrium morphologies of a dilute phase of self-associating model IDPs near a well-mixed, two-component lipid bilayer using the coarse-grained, explicit-solvent simulation technique of dissipative particle dynamics (DPD) [38–40], which is commonly used for simulating membranes [41,42], and complex fluids [43]. In our model, the two lipid species have identical molecular structure and interactions with each other, so that only their interaction with the IDPs can drive domain formation. The IDPs are modeled as soluble telechelic polymers (linear molecules with self-associating endcaps [44]). We first investigated how the IDPs organize the minority phase lipid by direct interaction with their headgroups, to simulate wetting phenomena. We then extended our exploration by attaching a polymeric linker to the minority phase lipid headgroups to better mimic the flexible cytoplasmic tails of transmembrane signalling proteins [32,35]. The linkers contain a linear sequence of sticky and inert domains, and we explored the morphological outcome of the system when the IDP endcaps are attracted to the sticky domains.

We *a priori* envisaged several morphological outcomes for the IDP/membrane system as the relative strengths of the IDP self-attraction and IDP-lipid/linker attraction are varied, and these are illustrated in Figure 1.

When both interactions are weak, the membrane is well mixed and the IDPs are dispersed in the solvent (a, IDPs not shown). In the absence of an attraction of the IDPs for the minority phase lipids, we expected the membrane to be well mixed, and the IDPs dispersed in the bulk solvent if their self-attraction is low (as in a), or phase separated, if their self-attraction is high (b). Increasing the attraction of the IDPs for the lipid headgroups leads to a morphology that is controlled by their self-attraction: if this is low, they may form a thin wetting layer (c), while a strong self-attraction may lead to a phase separated droplet adhering to a minority phase lipid domain (d). We observed all these results in the simulations, indicating that the equilibrium states of the membrane and IDPs are accessible to DPD simulations. Theoretical calculations have predicted a range of wetting states resulting from the adsorption of polymeric solute molecules on a membrane surface. In particular, it was found that surface phase transitions can occur under conditions in

which bulk phase separation is impossible at any solute concentration [27]. Monte Carlo simulations have been used to study the cooperativity of receptor-ligand binding and domain formation in membrane binding [45]. But a molecular-scale picture of the structural organization of membrane-associated BCs is still missing.

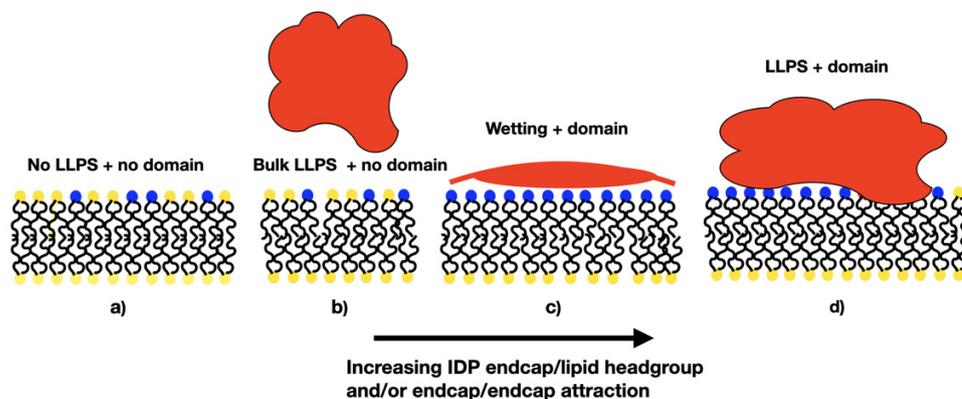


Figure 1. Cartoon showing potential morphologies of the bulk IDPs and two-component lipid membrane as the relative interactions between the IDPs (red) and the minority lipid headgroups (blue) are varied. (a) IDPs dispersed in the solvent; (b) high self-attraction; (c) thin wetting layer; (d) minority phase lipid domain.

Although our system contains only three molecular species (not including the solvent), a large number of parameters must be specified to define its state. Additionally, the simulated system is relatively small (~ 10 – 100 nanometres in linear dimension), and we may legitimately ask: What are the prospects of improving its biological fidelity? In considering this question, we encounter the barrier of the rapidly increasing computational cost of simulating a system of increasing size while retaining near-molecular resolution.

Biological membranes evolve on multiple timescales that are hard to span even with coarse-grained simulation techniques [46–49]. The resources needed to simulate a membrane of linear dimension L on a single compute core scale at least as L^5 , allowing for the spatial increase and time for diffusive processes to propagate, so it is challenging to extend such simulations to near-cell scale problems (\sim microns). The Folding@home consortium recently harnessed more than one million devices across the world to explore conformational transitions of the SARS-CoV-2 spike protein with an effective performance of 1.01 exaflops [50]. Even this extraordinary power is not sufficient to simulate the interaction of, for example, a complete virus particle with the host cell membrane [49].

This is not a new problem. Conventional parallel programming paradigms using, e.g., the message passing interface (MPI) system, increase the simulated volume by dedicating many compute cores to one simulation [51]. However, they are limited by the requirement that the compute cost per core must exceed the messaging cost, which necessitates a large number of particles to be assigned to each core. This limitation is not removed by adding more cores, but it can be eliminated by the elegant scheme we describe below, and which we have used to carry out large-scale simulations to validate our main results.

Since 2016, the Engineering and Physical Sciences Research Council of the UK has funded a project called POETS—Partially-Ordered Event-Triggered Systems (POETS)—that applies the non-von-Neumann computer architecture called event-based computing to large-scale computational problems [52]. POETS is not a general-purpose computer: Each application must be transformed into a graph in order to run on POETS, and reimplemented in the POETS domain specific language. We took the open source DPD code Osprey-DPD [53], converted it to run in POETS, and used it to explore the larger system size presented here. POETS-DPD allowed us to simulate larger systems at a reduced cost—both in terms of wall clock time and electricity consumed—compared to conventional CPU/GPU hardware [54].

In POETS-DPD, the simulation volume is divided into a grid of cells similar to that used in conventional MPI-based parallel schemes. Crucially, in contrast to MPI-based simulations, each grid cell contains only a few particles and is assigned to its own compute core, so each grid cell's computing step is very fast. Messaging between cores is executed in hardware and is also fast. Because the limitation of compute cost \gg messaging cost has been removed, the system as a whole can be scaled to any volume by simply adding more cores, and it exhibits almost perfect weak scaling.

We showcase in this special issue results obtained with conventional DPD and POETS-DPD, and demonstrate that event-based computing opens up exciting pathways to near cell-scale computational exploration. We use an inter-disciplinary approach that combines the development of new hardware and simulation algorithms on the bespoke computational hardware with the real-world biophysical problem of predicting the influence of protein phase separation on membrane organization. The POETS-DPD hardware and software is still under development, but the initial version used here provides functionally correct answers at speeds which are substantially faster than existing systems, and allows us to validate the dynamical phenomena we observe between small and large systems over long timescales.

The principal contributions of this paper are two-fold:

1. We present a simulation-based exploration of the morphological outcomes resulting from the 3D phase separation of intrinsically-disordered proteins near a 2D lipid membrane;
2. We describe the novel POETS compute architecture and the algorithm used in POETS-DPD, and demonstrate the power of this approach by simulating selected systems at a fraction of the wall clock time: a 14 cpu-day conventional DPD simulation is completed by POETS-DPD in 8 h, a speedup of more than a factor of 40.

The remainder of the paper is organized as follows. In Section 2, we briefly describe the dissipative particle dynamics simulation technique and the event-based hardware and POETS-DPD implementation, referring the reader to the literature for more details. Section 3 presents our main results on the equilibrium morphologies observed in the simulations. Section 4 discusses the powerful advantage of POETS-DPD by highlighting much larger systems. Finally, in Section 5, we illuminate the path ahead for using massively parallel POETS-DPD to explore near-cell scale dynamic phenomena.

2. Materials and Methods

2.1. Dissipative Particle Dynamics Simulations

Dissipative particle dynamics (DPD) is a coarse-grained, explicit-solvent form of molecular dynamics originally designed to simulate the hydrodynamic behavior of complex fluids [38–40]. In DPD, atoms and atomic groups are grouped into beads, of mass m , which interact via three non-bonded, short-ranged (vanish beyond a cut-off distance d_0), soft, momentum-conserving forces. A conservative force between each bead type pair (characterized by a strength parameter a_{ij} for beads of type i and j) gives beads a chemical identity, such as hydrophobic oil particles or hydrophilic amino acid residues. Additionally, a dissipative force (characterized by the parameter γ_{ij}) and random force (whose strength σ_{ij} is related to the dissipative force parameter by the fluctuation-dissipation theorem $\sigma_{ij}^2 = 2\gamma_{ij}k_B T$) provide a thermostat that keeps the system temperature $k_B T$ constant. More details of the force field are given in previous work [55]. Beads are connected into molecules with Hookean springs, and a bending stiffness may be applied via an additional bond angle-dependent potential. Once the forces are specified for all bead and molecule types in a simulation, the Newtonian equations of motion for the beads are integrated using a modified Velocity Verlet scheme. Because the non-bonded forces are soft, the integration scheme is able to use a larger time-step than is possible in atomistic molecular dynamics, and the grouping of several atomic groups into each bead reduces the number of degrees of freedom needing to be integrated. These two factors make DPD several orders of magnitude faster than traditional atomistic and coarse-grained molecular dynamics.

DPD has been extensively used to simulate lipid membranes [41,56], domain formation in two-component membranes [57,58], nanoparticles interacting with membranes [59,60], and many other soft matter systems [43]. We adopt here the model of Grafmüller et al. for the lipid molecules [61], and their molecular shape is shown in Figure 2. Briefly, each lipid has a headgroup composed of four hydrophilic beads (H_A for the major species, H_B for the minor species), and two tails each containing four hydrophobic beads (T_A for the major species, T_B for the minor species). The tails are connected to the final two adjacent head beads as shown in Figure 2. The IDPs are modeled as linear, semi-flexible polymers (backbone beads of type B) with self-associating endcaps containing four beads (bead type E) [55]. Water was represented by a single bead W. Both backbone and endcap beads in the IDPs are hydrophilic, and it is only the self-attraction of their endcaps that drives their phase separation. The full set of DPD conservative force parameters is given in Table 1. The two parameters a_{EE} and a_{EM} set the attraction between the IDP endcaps and themselves and the minority lipid headgroups, respectively, and are varied systematically in Section 3.1. The parameter a_{ES} sets the attraction of the IDP endcaps and the sticky domains in the linker attached to the minority phase lipid in Section 3.4. The dissipative force parameter γ_{ij} for all beads was 4.5 (in units of $\sqrt{mk_B T/d_0^2}$). Beads were bonded into molecules using Hookean springs with potential parameters of 128 and 0.5 (in units of $k_B T/d_0^2$ and d_0 respectively). A three-body bending potential of the form $k_3(1 - \cos(\varphi - \varphi_0))$ was applied to adjacent triples of backbone beads (B) in the IDPs, and the linker beads L and S when they were present, with parameters $k_3 = 5 k_B T$ and $\varphi_0 = 0$. A similar potential was applied to the lipid tails with parameters $k_3 = 15 k_B T$ and $\varphi_0 = 0$. Further details of the simulation technique are given in the literature [40,55,61].

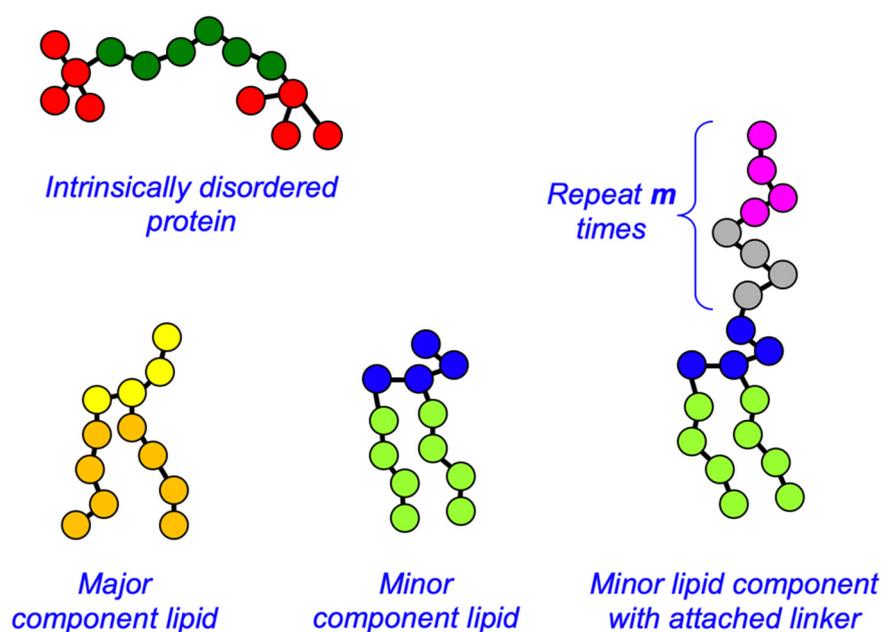


Figure 2. Cartoon of the molecular shape of the IDPs and lipid species. The conservative interactions for all bead types are given in Table 1. Note that the head beads and tail beads of the two lipid species are colored differently for clarity but are otherwise the same in the simulations. The number of linker elements can be varied, but is fixed at $m = 8$ for simplicity. The number of backbone beads in the IDPs is shown as six, but is varied as part of the investigation.

Table 1. Bead–bead conservative force parameters a_{ij} (in units of $k_B T/d_0$) for all bead types. The table is symmetric. The parameters a_{EE} , a_{EM} , and a_{ES} specify the attraction between the IDP endcaps and themselves, the minority lipid headgroup, and the sticky domains in the linkers respectively as described in the text.

a_{ij}	W	E	B	H _A	T _A	H _B	T _B	L	S
W	25								
E	25	a_{EE}							
B	23	25	25						
H _A	30	30	30	30					
T _A	75	35	35	35	10				
H _B	30	a_{EM}	30	30	35	30			
T _B	75	35	35	35	10	35	10		
L	30	30	30	30	35	30	35	30	
S	30	a_{ES}	30	30	35	30	35	30	30

All simulations took place in a box of size $40 \times 40 \times 48 (d_0)^3$ unless otherwise stated and periodic boundary conditions were applied in all three dimensions. A single planar membrane was pre-assembled with its normal along the long axis, and the minority phase lipids were randomly distributed in the upper monolayer only to localize IDP/membrane interactions to one side. A given number of IDPs were dispersed randomly throughout the remaining space and the simulation box was filled with solvent beads to the average bead density $\rho d_0^3 = 3$. The DPD length scale is set to $d_0 = 1$ nm from the size of the lipid headgroups [41]. We set the reduced system temperature to $k_B T = 1$ [40]. Each simulation was run for 3 million time-steps unless otherwise stated using an integration step size of 0.02τ , where $\tau = \sqrt{m d_0^2 / k_B T}$ is the DPD timescale. Because we are interested in equilibrium properties, we did not attempt to fix the simulation timescale more precisely.

Results in the $40 \times 40 \times 48 (d_0)^3$ simulation box are generated using the open-source, single-thread DPD code OSPREY-DPD [53]. Simulations in the larger box $100 \times 100 \times 48 (d_0)^3$ are performed using POETS-DPD as described next.

2.2. Massively Parallel Event-Based Simulations

The larger simulations in this work were performed on a new computer technology called POETS—Partially Ordered Event-triggered Systems [52]. POETS represents computational problems as a graph: vertices in the graph are independent computational threads; and edges between vertices represent channels that transport messages between threads. The abstract, arbitrary computational graph is mapped onto a fixed hardware network of RISC-V processors [62], which provides tens of thousands of hardware threads in the space and power envelope that would usually support only a few hundred conventional x86 threads. POETS is an ongoing research project (<https://www.poets-project.org>; accessed on 18 December 2021) but preliminary results show that it is highly suited to DPD simulations [54]. We used it both as an efficient way to produce the large-scale results presented here, and to evaluate its potential for production use in mesoscale research and simulation.

POETS is not a general-purpose computer. Each application must be transformed into a graph in order to run on POETS, and reimplemented in the POETS domain specific language. For our case of DPD simulations, we follow the same discretization of space approach used to parallelize DPD for the MPI framework. The simulation volume is spatially subdivided into a cubic array of unit cells for which the beads in each cell are managed by a single computational thread. Each cell becomes a vertex in the graph, and each vertex has edges to its 26 neighboring cells to represent information flows within each simulated time-step. A key point of POETS messaging is that it is very fast and consumes few cycles: sending a message is just one machine instruction, with message latencies

ranging from 1–10 micro-seconds, depending on the physical distance between cores. This contrasts with MPI messages that are typically very slow compared to a local function call, requiring hundreds of instructions to prepare, and with latencies ranging from 10–1000 micro-seconds, depending on the message size.

A big algorithmic difference between DPD on POETS and MPI is that the volume of space managed by a cell in the POETS implementation is comparable to the DPD non-bonded cut-off distance (d_0), so each cell contains only a few particles. In comparison, an MPI-enabled DPD must use much larger cells containing thousands of beads to ensure that the messaging cost is a small fraction of the total runtime. The difference between these two approaches is shown graphically in Figure 3. Effectively, because messaging in POETS is so cheap, we are able to use very large numbers of light-weight CPUs with each managing only a few beads.

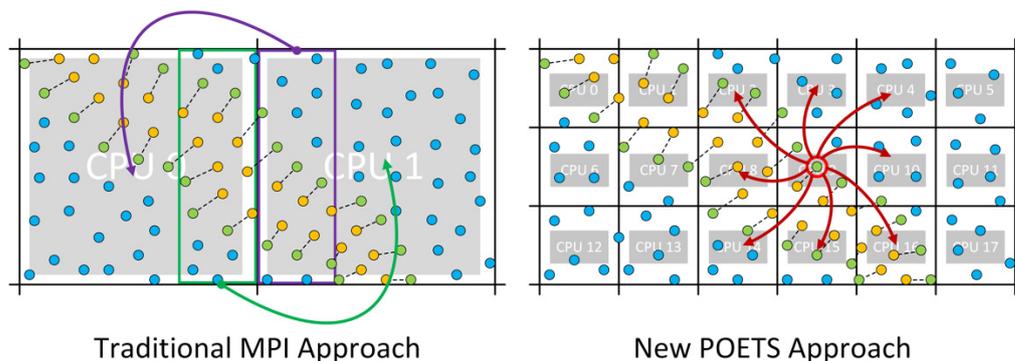


Figure 3. Contrast between the traditional MPI-based parallelization approach (left panel) in which each CPU owns a volume of space containing hundreds or thousands of beads, and the POETS approach (right panel) where space is sub-divided down to unit-volume cells containing only 3 or 4 beads, each of which is managed by one light-weight CPU.

Once ownership of all the unit cells in the simulation space has been assigned to physical POETS CPUs, the DPD algorithm executes in a highly concurrent fashion, with each thread operating independently and synchronized only by the messages containing beads. We only sketch the main ideas here, in order to give a sense of how it works and refer the read to the literature for more details [54]. POET-DPD executes in in two phases:

1. **Force calculation**

- a **Sharing:** Broadcast each resident bead position and velocity to neighbors;
- b **Integration:** For each received bead, calculate non-bonded DPD and Hookean bond interactions with resident cells;
- c **AngleForces:** Once head and tail beads for an angle bond are received by middle bead, calculate angle forces and broadcast to neighbors (which must include the owner of the angle bond’s head and tail);
- d **AngleAddition:** If an angle force is received *and* this cell contains the related head or tail bead, apply angle-bond forces to that head/tail.

2. **Bead movement**

- a **Movement:** Apply equations of motion to all beads resident in the cell;
- b **BeadExit:** If any bead leaves the cell, broadcast it to neighbors and remove from resident set;
- c **BeadEntrance:** If a bead is received from a neighbor *and* its new position is in the current cell, add it to the resident set.

3. Go back to step 1 for next step of simulation

This is a very high-level overview, but it highlights some of the important computational characteristics:

Concurrent: Each thread is running independently, so this approach is concurrent, where lots of threads do different things with loose synchronization, rather than parallel, where lots of threads do the same thing with tight synchronization.

Fine-grained: The simulation is decomposed into one computational thread per unit-volume cell, with each thread potentially only handling 3 or 4 beads, and resulting in thousands or millions of concurrent compute threads.

Event-triggered: Messages trigger computation, rather than computation being used to schedule messages. Within each phase, the sending and receiving of messages can be interleaved, overlapping compute and communication.

Broadcasts: Rather than aiming messages only at the correct receiver, we broadcast messages to all possible receivers. This might seem counter-intuitive, but due to the tight binding between processors and network in POETS this is very efficient.

Single algorithm: All calculations, including Hookean and angle bonds, are integrated into the same algorithm on the same compute thread, and handled concurrently with DPD force calculation.

There is obviously a lot more computational detail in precisely how this is implemented, not least because the experimental goals of the paper were used to drive development of the algorithms and methods. This meant that we had to handle the edge-cases needed in practical simulations, rather than just assuming-away the difficult parts in the pursuit of headline performance figures that do not translate to real-world performance.

For example, the need to handle angle bonds completely changed the way that we handled bonds within the algorithm. Our original algorithmic approach used a faster method with fewer algorithmic steps, but ultimately could only handle Hookean bonds efficiently. Though it was very fast, this was for a problem which is less useful in practice for end-users. By having computer scientists, hardware engineers, and bio-physics end-users working together we tried to avoid that trap, and tackle the more difficult but useful problems head on.

3. Results

3.1. Influence of the IDP-Membrane Affinity on the Equilibrium Morphology

We were interested in how compositionally-distinct domains in a multicomponent membrane may be induced by the nearby bulk phase separation of intrinsically-disordered proteins. As stated, the problem has too many parameters to be efficiently presented, and it is unfeasible to exhaustively explore its high-dimensional parameter space. We therefore studied the simplest system that may exhibit interesting behavior—a two-component membrane and a single type of IDP—and qualitatively enumerate its equilibrium states. The membrane contains a major lipid species and a smaller fraction of a minor lipid species, and we fixed their number fractions to give 1989 and 397 major and minor lipid molecules, respectively, corresponding to a minor fraction of 0.166. This ensures that a domain (if formed) is much larger than the lipid molecules but smaller than the membrane area, thereby reducing the effects of the boundary conditions. The lipid molecules possess identical structures and interactions and are well mixed in the absence of any influence of the IDPs. We initially fixed the backbone length of the IDPs to 6 beads and refer to them by their backbone length in an obvious notation as B_6 . We started with 397 B_6 molecules in the bulk phase. Note that the IDP backbone and endcap beads are hydrophilic, and only their endcap self-attraction drives their phase separation. We varied the IDP self-attraction and their affinity to the minority phase lipid headgroups by changing the conservative DPD interaction parameters a_{EE} , a_{EM} respectively. We defined dimensionless parameters to quantify the IDP endcap–endcap attraction (ε_{EE}), and the endcap–minority lipid headgroup (ε_{EM}) attraction by:

$$\varepsilon_{EE} = (a_{WE} - a_{EE})/a_{WE}$$

$$\varepsilon_{EM} = (a_{WE} - a_{EM})/a_{WE}$$

where a_{WE} is the DPD conservative interaction parameter for the water/IDP-endcap interaction. We explored the range $0 \leq \varepsilon_{ij} \leq 1$, for which 0 corresponds to no net attraction between bead types i, j (compared to their interaction with the solvent beads) and 1 to a very strong attraction.

When there was no attraction of the IDPs for themselves nor the membrane, $\varepsilon_{EE} = \varepsilon_{EM} = 0$, they remained dispersed in the solvent and the membrane was well mixed (Figure 4, Supplementary Video S1). In the following figures, the left panel shows the whole system while the right panel shows just the membrane for clarity.

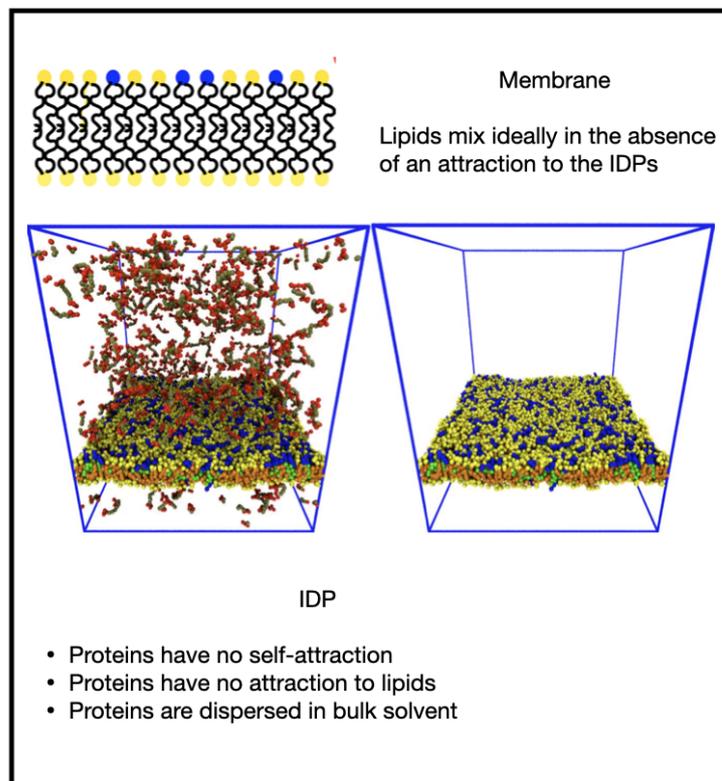


Figure 4. Equilibrium state of the system when both the IDP endcap self-attraction and attraction to the minority phase lipid headgroups are zero (solvent particles are invisible for clarity).

Increasing ε_{EE} while keeping the ε_{EM} zero led to bulk phase separation of the IDPs without influencing the membrane, which remained well mixed (Figure 5 and Supplementary Video S2). The value of $\varepsilon_{EE} = 0.68$ is not very strong, leading to the droplet being in equilibrium with the dilute phase of IDPs dispersed in the solvent.

Increasing $\varepsilon_{EM} = 0.8$ while keeping ε_{EE} zero caused the IDPs to wet the membrane, forming a thin layer as their endcaps bind to the lipid headgroups (Figure 6 and Supplementary Video S3). For the chosen fraction of minority phase lipid, the IDPs were able to saturate the domain leaving some excess IDPs in the solvent. It is interesting to note that the wetting domain formed as the result of the adsorption of the IDPs to the minority lipid headgroups because the IDPs have no tendency to phase separate. The short B_6 IDPs keep the lipids adsorbed to their endcaps in close proximity. When two or more such adsorbed IDPs approach by diffusion, the thermal fluctuations of the IDPs lead to their mixing between the lipids. This caused the IDPs to effectively connect all the minority lipids into a large domain instead of independent small clusters. However, the wetted domain was non-circular, indicating it has a low line tension, and small patches of lipid occasionally detach, taking a few IDPs with them before reattaching. We note that although the minor lipid species comprises only 16% of the total number of lipids, they are all in the upper monolayer and hence can occupy an area up to 32% of the monolayer's area.

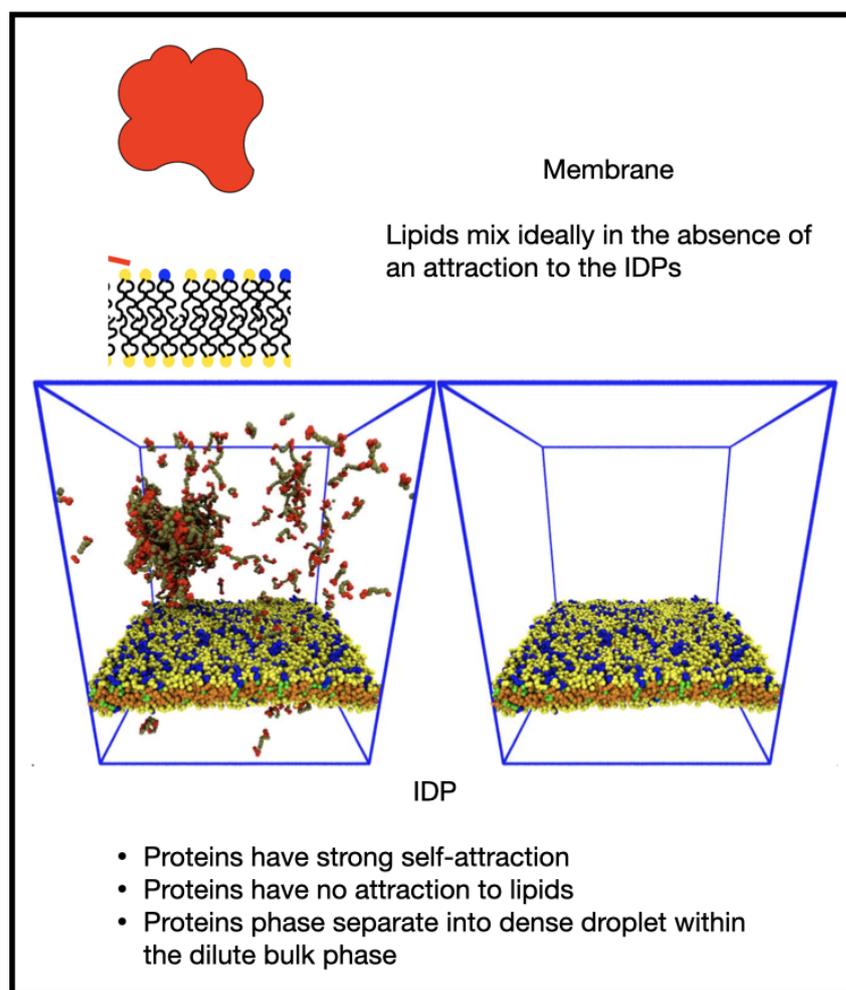


Figure 5. Increasing the IDP self-attraction with no attraction to the membrane leads to a dense protein droplet within a dilute phase, and leaves the membrane well mixed.

Finally, increasing both affinities to $\varepsilon_{EE} = \varepsilon_{EM} = 0.8$ led to the formation of a bulk droplet of IDP and wetting of a single domain in the membrane. After diffusing for some time, the droplet adsorbed to the domain, but at the selected concentration, it protruded from the membrane, forming a membrane-associated dense phase. (Figure 7 and Supplementary Video S4). The domain fluctuated around a circular shape, indicating the existence of a fairly high line tension. Small numbers of majority phase lipid sometimes diffused through the domain as seen in the movie Supplementary Video S4.

Our results for the equilibrium states of the system are summarized in Figure 8 as a two-dimensional morphology diagram with the endcap self-attraction ε_{EE} along the abscissa and the endcap–lipid headgroup ε_{EM} along the ordinate. We reemphasize that there was no attraction between the majority and minority phase lipids, and the observed domains were created only by the influence of the IDPs. At the origin, there was no attraction between the IDPs and the membrane nor between themselves, and the membrane was well mixed with the IDPs dispersed in bulk solvent (invisible for clarity). Increasing the IDP self-attraction ε_{EE} while keeping their attraction to the membrane ε_{EM} low (bottom right) led to their bulk phase separation while the membrane is well mixed. Increasing the IDP attraction for the membrane ε_{EM} while it has no self-attraction (top left) led to wetting of the membrane until the minority phase lipids are covered and excess IDPs remain in the bulk. When both interactions were high (top right), the IDPs drove domain formation in the membrane and adsorbed to it forming a droplet that projects into the solvent.

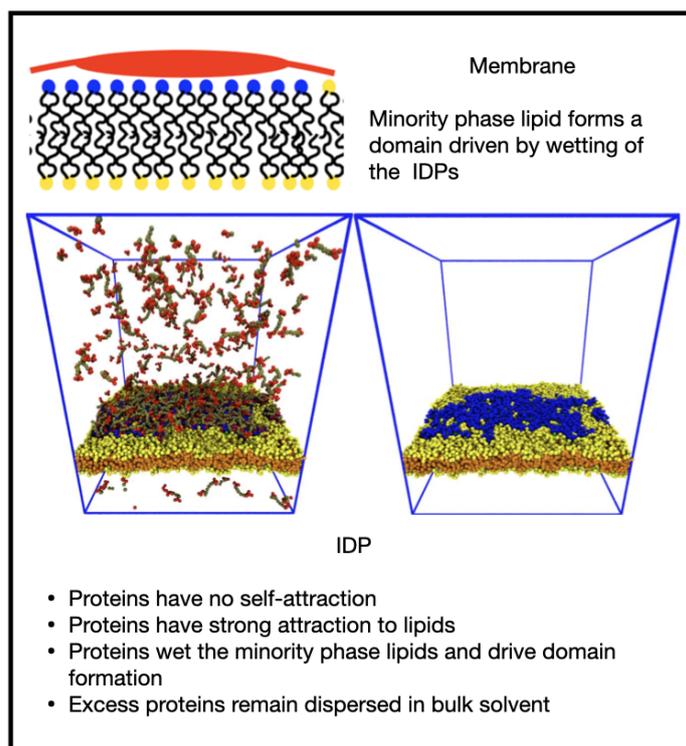


Figure 6. Increasing the IDP–membrane attraction in the absence of any IDP self-attraction leads to wetting of a domain formed of the minority phase lipids.

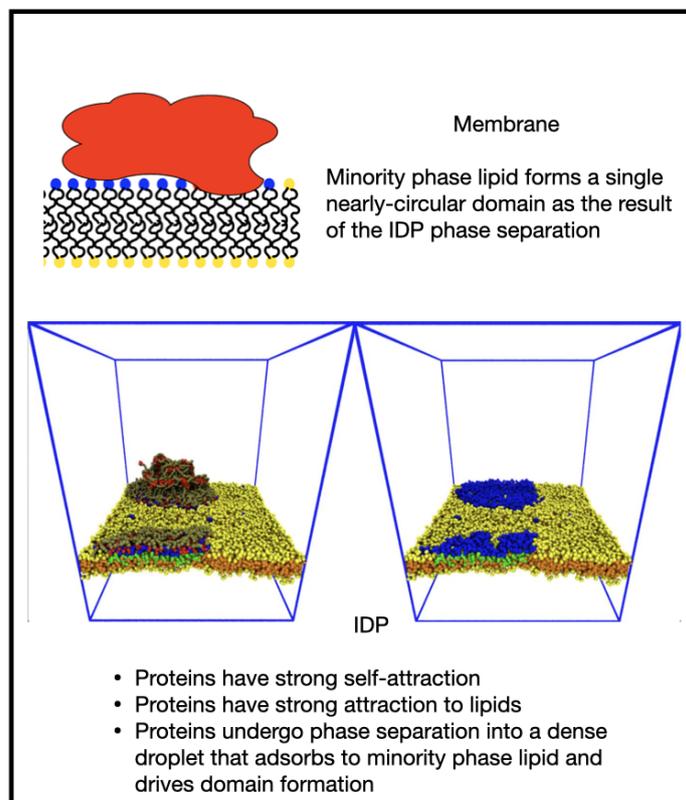


Figure 7. When both IDP self-attraction and the IDP–membrane attraction are strong, the proteins undergo phase separation into a dense droplet that adsorbs to the minority phase lipids driving domain formation but still protrudes into bulk solvent.

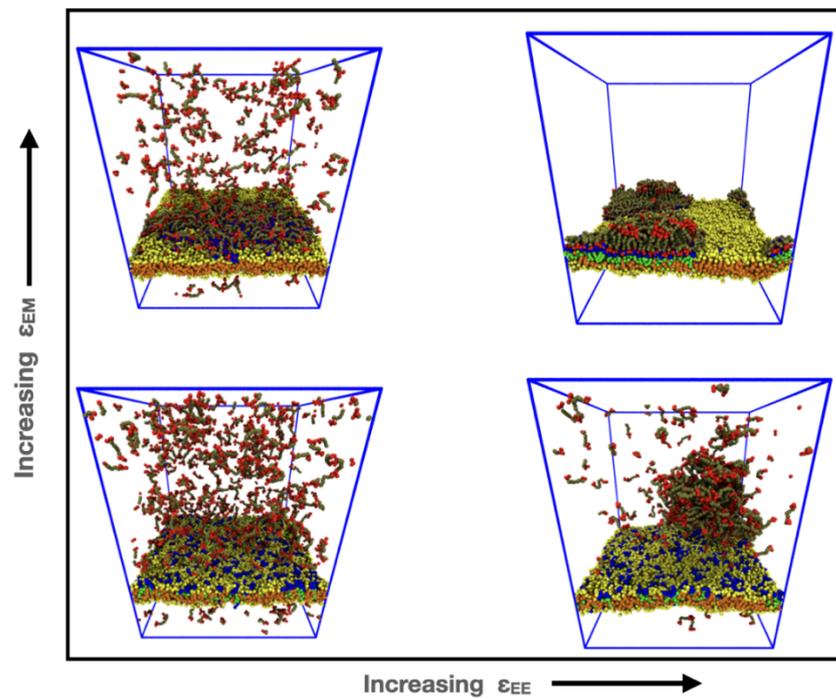


Figure 8. Morphology diagram in the $(\epsilon_{EE}, \epsilon_{EM})$ plane showing the equilibrium states of the IDPs and membrane, as their relative interactions are varied.

3.2. Reversible Coupling of Phase Separation and Domain Formation

Membrane domains frequently form transiently in response to cellular signals [32,35]. We observed a reversible transition between a phase separated droplet in the bulk solvent and membrane wetting that was induced by turning on and off the attraction of the IDPs to the minority lipids. Figure 9 shows snapshots from a simulation of 494 IDPs and 395/1978 minority and majority phase lipids, respectively. The timing of the snapshots is given as a fraction of the total simulation time of $T_{max} = 3 \times 10^6$ time steps. The IDPs were initially neutral towards the minority phase lipids ($\epsilon_{EM} = 0$), but had a self-attraction ($\epsilon_{EE} = 0.68$) that caused them to phase separate into a droplet in the bulk (see Supplementary Video S5). At time $0.2 T_{max}$, a strong attraction ($\epsilon_{EM} = 0.8$) of the IDP endcaps to the minority lipids was turned on. As the IDPs adsorbed to the minority lipids and drove them into a single domain, the droplet dissolved because the bulk IDP concentration fell below that at which it is in equilibrium with its dilute phase. The IDP/lipid attraction was removed at time $0.85 T_{max}$ and the IDPs left the membrane and reassembled in the bulk while the membrane rapidly returned to being well mixed. The time required for the membrane to return to a well-mixed state and the IDPs to reassemble in the bulk was much shorter than that for the wetting domain to form.

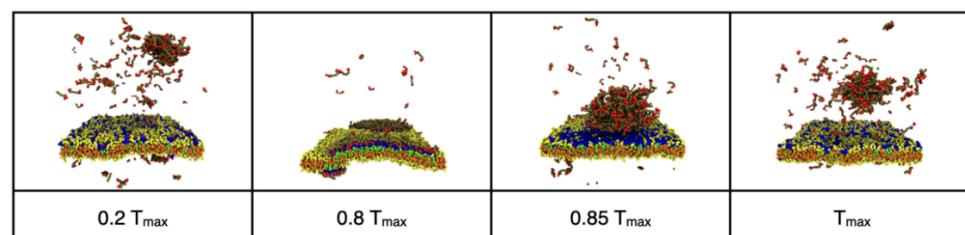


Figure 9. Reversible dissolution of the bulk IDP droplet and wetting of the membrane induced by toggling the interaction of the endcaps (red beads) with the minority phase lipid headgroups (blue beads.).

3.3. Increasing IDP Length Results in Patchy Domains

We next explored how the backbone length of the IDPs affected the domain morphology. We recall that IDPs with a short backbone B_6 and high affinities ($\epsilon_{EE} = \epsilon_{EM} = 0.8$) adsorb to the membrane forming a circular domain that shows small shape fluctuations, indicating a significant line tension (see Figure 7 and Supplementary Video S4). The upper panel of Figure 10 (and Supplementary Video S6) shows that the equilibrium state of a system containing 396 IDPs of the type B_8 and 396/1982 minority and majority phase lipids, respectively, also contained a single domain, but it contained small patches of the majority lipid and its non-circular shape indicated a lower line tension. On further increasing the IDP backbone length to B_{16} (lower panel of Figure 10 and Supplementary Video S7), we found that the conformationally-fluctuating IDPs bridged multiple, small lipid clusters, but no domain was formed. This simulation contained 296 IDPs and 395/1975 minority and majority phase lipids, respectively, to reduce the crowding effect of the longer IDPs. We hypothesize that the domain formed by short IDPs is broken up by the conformational entropy of the longer IDP backbones. Previous work has shown that B_{16} IDPs assemble into a structured network when they phase separate in the bulk solvent with a spatial structure whose length scale is selected by the length of the IDPs [55]. When these associated IDPs adsorb to the minority lipid headgroups, they maintain this structure which keeps the adsorbed minority lipids farther apart and unable to form a single domain.

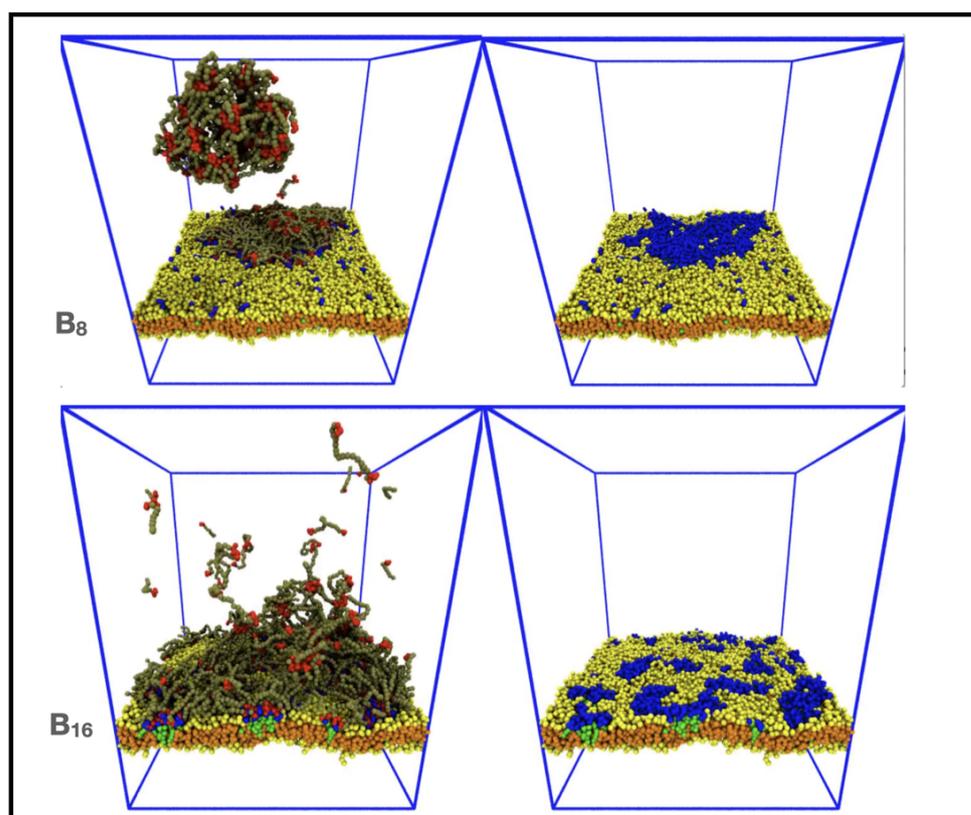


Figure 10. Domains formed from B_8 (upper panel) and B_{16} (lower panel) IDPs become increasingly patchy and disconnected for strong affinities $\epsilon_{EE} = \epsilon_{EM} = 0.8$ as the backbone length increases.

3.4. Clustering of Minority Lipids by IDP–Linker Attraction

IDPs are over-represented in cellular signaling and regulatory pathways [63], and are involved in clustering of transmembrane receptors [32]. The cytoplasmic domain of transmembrane receptors plays an important role in their phase separation. One example is the activation of T cell receptors in the immune system, which has been reconstituted in model membranes and leads to formation of a droplet of IDPs associating with the

membrane-bound receptors [34]. In addition, the phase separation of IDPs in neurons localizes receptors in the postsynaptic density and vesicles in the pre-synaptic zone [64,65].

To explore the ability of phase-separating IDPs to drive lipid domain formation via binding to a flexible linker, we attached a hydrophilic linker to the minority lipid headgroup and placed several sticky domains along it that are attracted to the IDP endcaps. The addition of the linkers introduces (at least) an additional three parameters: the number and affinity of the sticky domains, and their separation along the linker. For simplicity, we used a linear sequence of four sticky domains (of four S beads each) separated by inert regions (of four L beads each). The inert and sticky linker domains are hydrophilic and have no net attraction. We defined the dimensionless strength of the attraction between the IDP endcaps and the sticky domains (ϵ_{ES}) similarly to the endcaps, but used the linker–linker sticky domain conservative parameter as the baseline: $\epsilon_{ES} = (a_{SS} - a_{ES})/a_{SS}$. To avoid artifacts of the linkers prematurely clustering due to their initial configuration, we set the endcap–linker interaction to $\epsilon_{ES} = 0$ ($a_{SS} = a_{ES} = 30$), and equilibrated the system for 250,000 time steps allowing the minority lipids to diffuse freely across the membrane. Then the endcap self-attraction and endcap–linker attraction were set to a strong value, $\epsilon_{EE} = \epsilon_{ES} = 0.83$. Figure 11 shows the effects of 9 linker–lipids present in the membrane. As the IDPs phase separated, they bound to the linkers and slowly merged into a large droplet that drew the minority lipid/linkers into a domain (Figure 11, top panel, and Supplementary Video S8).

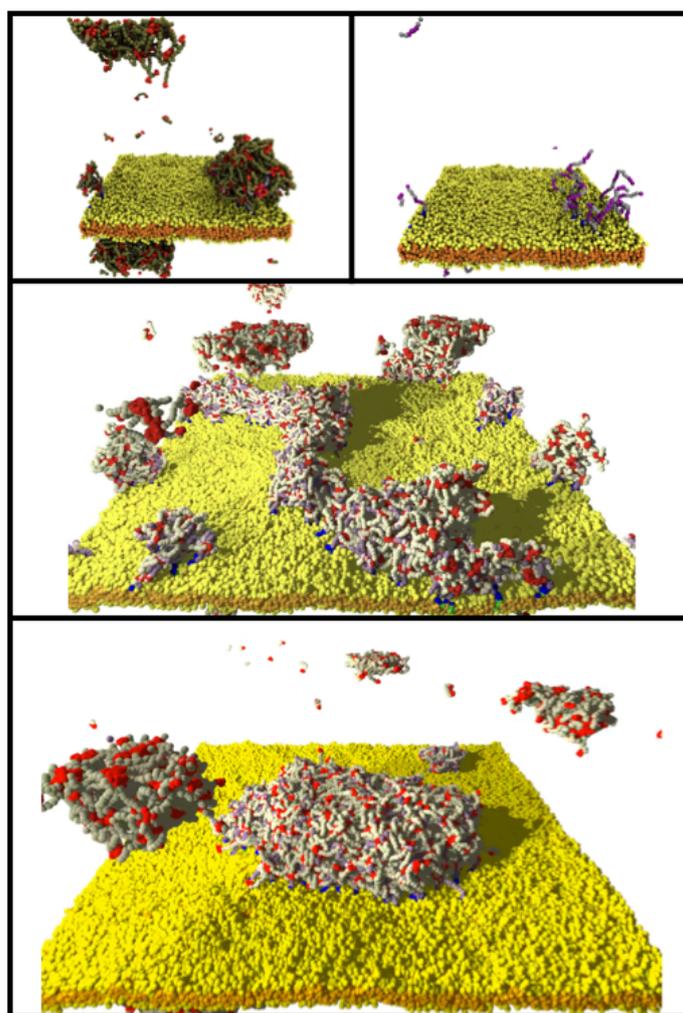


Figure 11. (Top) Clustering of linker–lipids driven by the phase separation of 395 IDPs (green with red endcaps) in the smaller system with OSPREY-DPD. (Middle, Bottom) Intermediate and final states of 1260 IDPs (white with red endcaps) in the larger system simulated by POETS-DPD.

The above results were generated using the OSPREY-DPD code in a simulation box of size $40 \times 40 \times 48 (d_0)^3$. In Figure 11 (and Supplementary Video S9), we compare the case of LLPS-driven domain formation from Figure 7 with those generated using the POETS-DPD code in the larger simulation box $100 \times 100 \times 48 (d_0)^3$. The upper panel shows the equilibrium state in the small simulation box. We found that as few as 7 linker–lipids can bind a droplet of 395 IDPs whose size is much larger than the clustered linkers. Two linkers flip-flopped from the upper leaflet as a consequence of the initial state and were able to bind sufficient IDPs to form a second domain visible in the lower left (and top left as it is connected across the periodic boundaries). The middle and lower panels of Figure 11 show two views of the equivalent simulation executed with POETS-DPD that contains a membrane that is 100 nm in linear dimension and contains 1260 IDPs and 2520 linker–lipids. The same clustering of the larger number of IDPs to the larger number of linker–lipids results in a similar droplet bound to the membrane (cp. Supplementary Videos S8 and S9).

4. Discussion

The increasing number of biomolecular condensates discovered in cells and the ubiquity of membrane-bound organelles suggests that interactions between them are likely to be important for cellular physiology. They also provide an interesting conceptual system for exploring how the two-dimensional nature of a membrane interacts with three-dimensional droplets. An example is provided by the phase separation of multivalent IDPs coupled to transmembrane signaling proteins in immune cell signaling [11,35]. We used coarse-grained molecular simulations to explore the morphologies adopted by a dilute phase of IDPs near a two-component lipid membrane. The high computational cost of such simulations drove this collaboration between biophysicists and the developers of novel computer hardware to explore how coarse-grained simulations on much larger scales can be accelerated and yet consume less electricity.

Our biological results are summarized in the morphology diagram in Figure 8, where we explored the relationship between the molecular properties of the constituent molecules and their equilibrium morphological phase behavior. Two non-trivial outcomes were observed. First, if the IDP endcaps were more strongly attracted to the minority phase lipid headgroups than to each other, they wet the membrane. If the length of the IDPs is short, this drove the minority lipids into a single large domain, while longer IDPs adsorbed via multiple, small, discrete clusters. When the IDP self-interaction and their interaction with the minority lipid headgroup (or with sticky linkers attached to the headgroup) were both sufficiently large, bulk LLPS occurred and drove formation of a single membrane domain.

Most of the preceding results were obtained from small systems with around 2400 lipids in the membrane and several hundred IDPs, comprising 230,000 particles in total. Each simulation required 14 cpu-days on a single core of a Ryzen Threadripper 3970X 4.5 GHz desktop machine. It is tedious and unfeasible to carry out hundreds of these simulations to explore exhaustively the parameter space of such complex models. In addition, the ~70 simulations we performed consumed an environmentally-unsustainable amount of electricity. We therefore performed some larger simulations in a box size $100 \times 100 \times 48 (d_0)^3$ on a novel computer hardware called POETS that drastically reduces the wall clock time of our DPD simulations. The larger system contains 14,600 lipids and 1260 IDPs, comprising 1,400,000 particles in total, but requires only 3 days to run the same number of time-steps as the smaller case.

The research goal of our collaboration of biophysicists and computer engineers is to ensure that the new POETS algorithms and hardware agree with the established DPD code-base (for which many published results exist); that they can handle real-world problems that practitioners want to solve; and that they are highly scalable. Initially, we focused on matching the performance of single-node systems such as shared-memory multi-core CPUs and individual GPUs.

Providing direct comparisons between disparate hardware/software systems is very difficult. For example, it is often unclear how much systems actually cost (buying 1 unit

is different from buying 1000), what their power consumption is (should cooling costs be included?), and whether computations are equivalent (is single-precision the same as double-precision?). Table 2 shows indicative metrics for how different systems behave in practice, based on published figures and (where possible) our own experiments. We characterized them in terms of one compute-node, where a node is a multi-core CPU, GPU plus support CPU, or an FPGA card for POETS, with power consumption quoted per node. The performance was measured in total bead-steps per second across all nodes, which is simply the total number of beads multiplied by the total number of simulation time steps, divided by the execution time. CPU figures are based on our own measurements, while GPU figures are courtesy of Seaton [66].

Table 2. Indicative performance of distinct compute platforms. Year: year hardware platform was introduced; Watts: power consumption of one node under load; Nodes: # compute nodes; Perf: performance in millions of bead-steps per second; Perf/Watt: performance per watt of electricity consumed.

System	Year	Software	Watts	Nodes	Perf/10 ⁶	Perf/Watt
CPUx1	2020	Osprey-DPD	250	1	0.5	2000
CPUx32	2020	LAMMPS	500	1	19.0	38,000
GPUx1	2016	DL-MESO	400	1	23.7	59,195
GPUx8	2016	DL-MESO	400	32	285.7	22,321
POETS Gen1	2011	POETS-DPD	200	48	24.0	2500
POETS Gen2	2022	POETS-DPD	250	48	360.0	30,000
Target parameters for next generation POETS system:						
POETS Gen3	2025	POETS-DPD	250	48	3600.0	300,000

The figures for “POETS Gen1” are the performance attained in this work, which is the measured performance of the hardware performing the large-scale simulations described in this paper. We clearly exceeded the performance of the single-core Osprey-DPD simulator used for the parameter exploration, which is our baseline comparison. It is also slightly faster than a high-performance 32-core machine in our HPC cluster when using multi-core LAMMPS, and about the same performance as a single high-performance GPU. However, we can see that the performance per watt was relatively weak—similar to a single-threaded implementation running on a CPU. This is a consequence of two effects: First, the hardware we used is the first generation, which is based on old FPGAs from 2011; second, we have so far focused on correctness and useability, and have spent little time on performance optimization.

Overall, we achieved the research goals of the collaboration in producing a new hardware system and algorithms that:

- Produce reliable DPD results (agree with published DPD implementation) [53];
- Are useful for practitioners;
- Are competitive in performance with existing single-node systems.

We are currently assembling the second generation of POETS hardware (POETS Gen2), which is substantially more powerful as it is built on more recent FPGAs. This increases the number of cores by a factor of 10, and the floating-point performance per core by a factor of 4, so we have included in Table 2 estimated figures for the POETS-DPD implementation running on this new hardware. At this point, the performance and efficiency start to become competitive with multi-GPU systems, even without additional tuning and algorithmic optimizations. For example, we have already found that dynamic load-balancing within the POETS hardware dramatically improves simulation rate.

Looking further ahead, we are actively exploring new algorithms and optimizations which substantially improve the performance per FPGA node. In particular, adding custom instructions designed to support DPD calculations will reduce the number of instructions

per bead interaction by a factor of 10. We included tentative estimates for the eventual performance we are targeting using the hardware currently being assembled, based on reasonable extrapolations. Clearly more research is needed, and this level of performance is an aspirational goal, but having demonstrated that the basic idea works and is useful we have confidence that this target is worth pursuing.

5. Conclusions

We live in interesting times: The power of computer systems is increasing year-by-year, bringing biological fidelity at increasing length scales and ever longer time periods (almost) within our grasp. Existing technologies such as multi-core, MPI, and GPUs provide one way of scaling to larger simulations, but it is increasingly important to reduce the financial and environmental costs of the simulations. Exotic architectures with custom hardware are another way, but it is unclear whether they work for real-world problems, or even if they give the correct results.

The principal goal of this paper was to identify the equilibrium morphologies of a model of phase separating proteins near a two-component membrane using coarse-grained simulations. But a secondary goal was to demonstrate how biophysical simulation needs can drive inter-disciplinary research that yields faster and cheaper simulations. By embedding the development of a new hardware simulation platform into the scientific process, we create a feedback loop between the computational modeling pull and the computer technology push that ensures the hardware advances work in practice—not just in theory. The results presented in this paper were generated on distinct compute platforms (Single core: Ryzen ThreadRipper 3970X; 4.5 GHz x86 32-core desktop; POETS machine: 49000 RISC-V threads): Each platform (running different codebases) provides consistent results, which is an important and often overlooked aspect of simulation-based research.

Our final aim was to illuminate the path towards using event-based computing to extend computational biophysics into more realistic arenas of membranes in health and disease. Extensions to our work could follow several interesting routes. The assumption of ideal mixing of the lipid species in the membrane could be relaxed to explore the converse effect of domain formation on the bulk phase separation. The model IDPs could also be more closely modeled on real proteins by adding multiple, punctate binding sites, or a mixture of IDPs could be included. However, increasing the biological fidelity creates more complex models with more parameters, making a systematic exploration of their behaviour challenging on conventional architectures. The POETS computer framework introduced here accelerates DPD simulations sufficiently that the exploration of high-dimensional parameter space models is feasible while simultaneously reducing the energy consumed. As simulations are increasingly used to augment wet-lab experiments, this will become an increasingly important route to sustainable science.

Supplementary Materials: The following are available online at <https://www.mdpi.com/article/10.3390/membranes12010017/s1>, Video S1: Dispersed B₆ IDPs and well-mixed membrane ($\epsilon_{EE} = \epsilon_{EM} = 0$), Video S2: Bulk LLPS of B₆ IDPs and well-mixed membrane ($\epsilon_{EE} = 0.68$, $\epsilon_{EM} = 0$), Video S3: Wetting of membrane by B₆ IDPs drives domain formation ($\epsilon_{EE} = 0$, $\epsilon_{EM} = 0.8$), Video S4: LLPS of B₆ IDPs drives domain formation ($\epsilon_{EE} = \epsilon_{EM} = 0.8$), Video S5: Reversible LLPS and domain formation of B₆ IDPs ($\epsilon_{EE} = 0.68$, $\epsilon_{EM} = 0.8$), Video S6: LLPS of B₈ IDPs creates patchy domains ($\epsilon_{EE} = \epsilon_{EM} = 0.8$), Video S7: LLPS of B₁₆ IDPs creates small lipid clusters ($\epsilon_{EE} = \epsilon_{EM} = 0.8$), Video S8: LLPS of B₈ IDPs and linker binding drives domain formation ($\epsilon_{EE} = \epsilon_{ES} = 0.83$), Video S9: LLPS of B₈ IDPs and linker binding drives domain formation in (100 nm)³ box ($\epsilon_{EE} = \epsilon_{ES} = 0.83$).

Author Contributions: Conceptualization, J.C.S., D.B.T. and A.D.B.; methodology, J.C.S., D.B.T., J.R.B., G.M.B. and M.L.V.; software, J.C.S., D.B.T., J.R.B., G.M.B. and M.L.V.; validation, J.C.S., D.B.T. and A.D.B.; formal analysis, J.C.S. and D.B.T.; resources, A.D.B.; writing—original draft preparation, J.C.S.; writing—review and editing, J.C.S., D.B.T., J.R.B., G.M.B., M.L.V. and A.D.B.; visualization, J.C.S. and D.B.T.; supervision D.B.T. and A.D.B.; project administration, D.B.T. and A.D.B.; funding acquisition, A.D.B. All authors have read and agreed to the published version of the manuscript.

Funding: J.C.S. was supported by funding to the Blue Brain Project, a research center of the École polytechnique fédérale de Lausanne (EPFL), from the Swiss government's ETH Board of the Swiss Federal Institutes of Technology. D.B.T., J.R.B., G.M.B., M.L.V., and A.D.B. were supported by the UK EPSRC Grant EP/N031768/1 (POETS).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Simulation datasets are available on reasonable request to the corresponding author.

Acknowledgments: The original DPD source code used in this work is available on GitHub (<https://github.com/Osprey-DPD/osprey-dpd>; accessed on 18 December 21). Snapshots and movies of the simulations were produced using the open-source VMD software from the University of Illinois Urbana Champaign (<https://www.ks.uiuc.edu/Research/vmd/>; accessed on 18 December 21) [67]. The authors are grateful to M. Seaton for sharing performance data on their GPU-based simulation hardware [66].

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Alberts, B.; Bray, D.; Lewis, J.; Raff, M.; Roberts, K.; Watson, J.D. *Molecular Biology of the Cell*, 2nd ed.; Garland Publishing, Inc.: New York, NY, USA, 1989.
2. Engelman, D.M. Membranes are more mosaic than fluid. *Nature* **2005**, *438*, 578–580. [[CrossRef](#)]
3. Zeno, W.F.; Day, K.J.; Gordon, V.D.; Stachowiak, J.C. Principles and Applications of Biological Membrane Organization. *Annu. Rev. Biophys.* **2020**, *49*, 19–39. [[CrossRef](#)] [[PubMed](#)]
4. Lipowsky, R.; Brinkmann, M.; Dimova, R.; Haluska, C.; Kierfeld, J.; Shillcock, J. Wetting, budding, and fusion—Morphological transitions of soft surfaces. *J. Phys. Condens. Matter* **2005**, *17*, S2885. [[CrossRef](#)]
5. Johannes, L.; Pezeshkian, W.; Ipsen, J.H.; Shillcock, J.C. Clustering on Membranes: Fluctuations and More. *Trends Cell Biol.* **2018**, *28*, 405–415. [[CrossRef](#)] [[PubMed](#)]
6. Simons, K.; Gerl, M.J. Revitalizing membrane rafts: New tools and insights. *Nat. Rev. Mol. Cell Biol.* **2010**, *11*, 688–699. [[CrossRef](#)]
7. Pezeshkian, W.; Gao, H.; Arumugam, S.; Becken, U.; Bassereau, P.; Florent, J.-C.; Ipsen, J.H.; Johannes, L.; Shillcock, J.C. Mechanism of Shiga Toxin Clustering on Membranes. *ACS Nano* **2017**, *11*, 314–324. [[CrossRef](#)]
8. Johannes, L.; Römer, W. Shiga toxins—From cell biology to biomedical applications. *Nat. Rev. Microbiol.* **2010**, *8*, 105–116. [[CrossRef](#)]
9. Banani, S.F.; Lee, H.O.; Hyman, A.A.; Rosen, M.K. Biomolecular Condensates: Organizers of Cellular Biochemistry. *Nature Rev. Mol. Cell Biol.* **2017**, *18*, 285–298. [[CrossRef](#)]
10. Boeynaems, S.; Alberti, S.; Fawzi, N.L.; Mittag, T.; Polymenidou, M.; Rousseau, F.; Schymkowitz, J.; Shorter, J.; Wolozin, B.; Bosch, L.V.D.; et al. Protein Phase Separation: A New Phase in Cell Biology. *Trends Cell Biol.* **2018**, *28*, 420–435. [[CrossRef](#)]
11. Zhang, J.Z.; Mehta, S.; Zhang, J. Liquid–liquid phase separation: A principal organizer of the cell's biochemical activity architecture. *Trends Pharmacol. Sci.* **2021**, *42*, 845–856. [[CrossRef](#)]
12. Shin, Y.; Brangwynne, C.P. Liquid phase condensation in cell physiology and disease. *Science* **2017**, *357*, eaaf4382, 1–11. [[CrossRef](#)] [[PubMed](#)]
13. Brangwynne, C.P.; Tompa, P.; Pappu, R.V. Polymer Physics of Intracellular Phase Transitions. *Nat. Phys.* **2015**, *11*, 899–904. [[CrossRef](#)]
14. Holehouse, A.S.; Pappu, R.V. Functional Implications of Intracellular Phase Transitions. *Biochemistry* **2018**, *57*, 2415–2423. [[CrossRef](#)]
15. Protter, D.S.; Parker, R. Principles and Properties of Stress Granules. *Trends Cell Biol.* **2016**, *26*, 668–679. [[CrossRef](#)] [[PubMed](#)]
16. Shin, Y.; Chang, Y.-C.; Lee, D.S.; Berry, J.; Sanders, D.W.; Ronceray, P.; Wingreen, N.S.; Haataja, M.; Brangwynne, C.P. Liquid Nuclear Condensates Mechanically Sense and Restructure the Genome. *Cell* **2018**, *175*, 1481–1491. [[CrossRef](#)]
17. Zeng, M.; Shang, Y.; Araki, Y.; Guo, T.; Hugarir, R.L.; Zhang, M. Phase Transition in Postsynaptic Densities Underlies Formation of Synaptic Complexes and Synaptic Plasticity. *Cell* **2016**, *166*, 1163–1175. [[CrossRef](#)] [[PubMed](#)]
18. Feng, Z.; Wu, X.; Zhang, M. Presynaptic bouton compartmentalization and postsynaptic density-mediated glutamate receptor clustering via phase separation. *Neuropharmacology* **2021**, *193*, 108622. [[CrossRef](#)]
19. Söding, J.; Zwicker, D.; Sohrabi-Jahromi, S.; Boehning, M.; Kirschbaum, J. Mechanisms for Active Regulation of Biomolecular Condensates. *Trends Cell Biol.* **2019**, *30*, 4–14. [[CrossRef](#)]
20. Patel, A.; Lee, H.O.; Jawerth, L.; Maharana, S.; Jahnel, M.; Hein, M.Y.; Stoynov, S.; Mahamid, J.; Saha, S.; Franzmann, T.M.; et al. A Liquid-to-Solid Phase Transition of the ALS Protein FUS Accelerated by Disease Mutation. *Cell* **2015**, *162*, 1066–1077. [[CrossRef](#)]
21. Peskett, T.R.; Rau, F.; O'Driscoll, J.; Patani, R.; Lowe, A.; Saibil, H.R. A Liquid to Solid Phase Transition Underlying Pathological Huntingtin Exon1 Aggregation. *Mol. Cell* **2018**, *70*, 588–601. [[CrossRef](#)] [[PubMed](#)]

22. Ray, S.; Singh, N.; Kumar, R.; Patel, K.; Pandey, S.; Datta, D.; Mahato, J.; Panigrahi, R.; Navalkar, A.; Mehra, S.; et al. α -Synuclein aggregation nucleates through liquid–liquid phase separation. *Nat. Chem.* **2020**, *12*, 705–716. [[CrossRef](#)]
23. Alberti, S.; Hyman, A.A. Are Aberrant Phase Transitions a Driver of Cellular Aging? *Bioessays* **2016**, *38*, 959–968. [[CrossRef](#)] [[PubMed](#)]
24. Boija, A.; Klein, I.A.; Young, R.A. Biomolecular Condensates and Cancer. *Cancer Cell.* **2021**, *39*, 174–192. [[CrossRef](#)]
25. Spanni, S.; Tereshchenko, M.; Mastromarco, G.J.; Ihn, S.J.; Lee, H.O. Biomolecular condensates in neurodegeneration and cancer. *Traffic* **2019**, *20*, 890–911. [[CrossRef](#)] [[PubMed](#)]
26. Taniue, K.; Akimitsu, N. Aberrant phase separation and cancer. *FEBS J.* **2021**. [[CrossRef](#)] [[PubMed](#)]
27. Zhao, X.; Bartolucci, G.; Honigmann, A.; Jülicher, F.; Weber, C.A. Wetting and Prewetting Phase Transitions facilitated by Surface Binding. *arXiv* **2021**, arXiv:2106.12565.
28. Botterbusch, S.; Baumgart, T. Interactions between Phase-Separated Liquids and Membrane Surfaces. *Appl. Sci.* **2021**, *11*, 1288. [[CrossRef](#)] [[PubMed](#)]
29. Kusumaatmaja, H.; May, A.I.; Knorr, R.L. Intracellular wetting mediates contacts between liquid compartments and membrane-bound organelles. *J. Cell Biol.* **2021**, *220*, e202103175. [[CrossRef](#)]
30. Snead, W.T.; Gladfelter, A.S. The Control Centers of Biomolecular Phase Separation: How Membrane Surfaces, PTMs, and Active Processes Regulate Condensation. *Mol. Cell* **2019**, *76*, 295–305. [[CrossRef](#)]
31. Banjade, S.; Rosen, M.K. Phase transitions of multivalent proteins can promote clustering of membrane receptors. *eLife* **2014**, *3*, e04123. [[CrossRef](#)]
32. Jaqaman, K.; Ditlev, J.A. Biomolecular condensates in membrane receptor signaling. *Curr. Opin. Cell Biol.* **2021**, *69*, 48–54. [[CrossRef](#)] [[PubMed](#)]
33. Dustin, M.L.; Muller, J. Liquidity in immune cell signaling. *Science* **2016**, *352*, 516–517. [[CrossRef](#)] [[PubMed](#)]
34. Su, X.; Ditlev, J.A.; Hui, E.; Xing, W.; Banjade, S.; Okrut, J.; King, D.S.; Taunton, J.; Rosen, M.K.; Vale, R.D. Phase separation of signaling molecules promotes T cell receptor signal transduction. *Science* **2016**, *352*, 595–599. [[CrossRef](#)]
35. Chong, P.A.; Forman-Kay, J.D. Liquid–liquid phase separation in cellular signaling systems. *Curr. Opin. Struct. Biol.* **2016**, *41*, 180–186. [[CrossRef](#)] [[PubMed](#)]
36. Chung, J.K.; Huang, W.Y.; Carbone, C.B.; Nocka, L.M.; Parikh, A.N.; Vale, R.D.; Groves, J.T. Coupled membrane lipid miscibility and phosphotyrosine-driven protein condensation phase transitions. *Biophys. J.* **2020**, *120*, 1257–1265. [[CrossRef](#)] [[PubMed](#)]
37. Mu, W.; Ji, Z.; Zhou, M.; Wu, J.; Lin, Y.; Qiao, Y. Membrane-confined liquid-liquid phase separation toward artificial organelles. *Sci. Adv.* **2021**, *7*, eabf9000. [[CrossRef](#)] [[PubMed](#)]
38. Hoogerbrugge, P.J.; Koelman, J.M.V.A. Simulating Microscopic Hydrodynamic Phenomena with Dissipative Particle Dynamics. *Europhys Lett.* **1992**, *19*, 155–160. [[CrossRef](#)]
39. Espagnol, P.; Warren, P.B. Statistical Mechanics of Dissipative Particle Dynamics. *Europhys. Lett.* **1995**, *30*, 191–196. [[CrossRef](#)]
40. Groot, R.D.; Warren, P.B. Dissipative Particle Dynamics: Bridging the Gap Between Atomistic and Mesoscopic Simulations. *J. Chem. Phys.* **1997**, *107*, 4423–4435. [[CrossRef](#)]
41. Shillcock, J.; Lipowsky, R. Equilibrium structure and lateral stress distribution of amphiphilic bilayers from dissipative particle dynamics simulations. *J. Chem. Phys.* **2002**, *117*, 5048–5061. [[CrossRef](#)]
42. Laradji, M.; Kumar, P.B.S. Dynamics of Domain Growth in Self-Assembled Fluid Vesicles. *Phys. Rev. Lett.* **2004**, *93*, 198105. [[CrossRef](#)]
43. Espagnol, P.; Warren, P.B. Perspective: Dissipative Particle Dynamics. *J. Chem. Phys.* **2017**, *146*, 150901. [[CrossRef](#)] [[PubMed](#)]
44. Zhang, Z.; Chen, Q.; Colby, R.H. Dynamics of associative polymers. *Soft Matter* **2018**, *14*, 2961–2977. [[CrossRef](#)]
45. Li, L.; Hu, J.; Shi, X.; Różycki, B.; Song, F. Interplay between cooperativity of intercellular receptor–ligand binding and coalescence of nanoscale lipid clusters in adhering membranes. *Soft Matter* **2021**, *17*, 1912–1920. [[CrossRef](#)] [[PubMed](#)]
46. Pezeshkian, W.; Marrink, S.J. Simulating realistic membrane shapes. *Curr. Opin. Cell Biol.* **2021**, *71*, 103–111. [[CrossRef](#)]
47. Enkavi, G.; Javanainen, M.; Kulig, W.; Róg, T.; Vattulainen, I. Multiscale Simulations of Biological Membranes: The Challenge To Understand Biological Phenomena in a Living Substance. *Chem. Rev.* **2019**, *119*, 5607–5774. [[CrossRef](#)] [[PubMed](#)]
48. Pezeshkian, W.; König, M.; Wassenaar, T.A.; Marrink, S.J. Backmapping triangulated surfaces to coarse-grained membrane models. *Nat. Commun.* **2020**, *11*, 2296. [[CrossRef](#)] [[PubMed](#)]
49. Yu, A.; Pak, A.J.; He, P.; Monje-Galvan, V.; Casalino, L.; Gaieb, Z.; Dommer, A.C.; Amaro, R.E.; Voth, G.A. A multiscale coarse-grained model of the SARS-CoV-2 virion. *Biophys. J.* **2021**, *120*, 1097–1104. [[CrossRef](#)]
50. Zimmerman, M.I.; Porter, J.R.; Ward, M.D.; Singh, S.; Vithani, N.; Meller, A.; Mallimadugula, U.L.; Kuhn, C.E.; Borowsky, J.H.; Wiewiora, R.P. SARS-CoV-2 simulations go exascale to predict dramatic spike opening and cryptic pockets across the proteome. *Nat. Chem.* **2021**, *13*, 651–659. [[CrossRef](#)]
51. Karniadakis, G.E.; Kirby, R.M.I. *Parallel Scientific Computing in C++ and MPI*; Cambridge University Press: Cambridge, UK, 2003.
52. Naylor, M.; Moore, S.W.; Thomas, D. Tinsel: A Manythread Overlay for FPGA Clusters. In Proceedings of the 29th International Conference on Field Programmable Logic and Applications (FPL), Barcelona, Spain, 8–12 September 2019; pp. 375–383.
53. Shillcock, J.C. OSPREY-DPD. Open Source Polymer Research Engine—Dissipative Particle Dynamics. 2020. Available online: <https://github.com/Osprey-DPD/osprey-dpd> (accessed on 18 December 2021).
54. Beaumont, J.R.; Brown, A.D.; Thomas, D.B.; Shillcock, J.C.; Naylor, M.F.; Bragg, G.M.; Vousden, M.L.; Moore, S.W.; Fleming, S.T. An event-driven approach to Dissipative Particle Dynamics. *ACM Trans. Parallel Comput.* (not published). **2021**.

55. Shillcock, J.C.; Brochut, M.; Chénais, E.; Ipsen, J.H. Phase behaviour and structure of a model biomolecular condensate. *Soft Matter*. **2020**, *16*, 6413–6423. [[CrossRef](#)]
56. Venturoli, M.; Sperotto, M.M.; Kranenburg, M.; Smit, B. Mesoscopic Models of Biological Membranes. *Phys. Rep.* **2006**, *437*, 1–54. [[CrossRef](#)]
57. Laradji, M.; Sunil Kumar, P.B. Domain growth, budding, and fission in phase-separating self-assembled fluid bilayers. *J. Chem. Phys.* **2005**, *123*, 224902. [[CrossRef](#)] [[PubMed](#)]
58. Illya, G.; Lipowsky, R.; Shillcock, J.C. Two-component membrane material properties and domain formation from dissipative particle dynamics. *J. Chem. Phys.* **2006**, *125*, 114710. [[CrossRef](#)] [[PubMed](#)]
59. Smith, K.A.; Jasnow, D.; Balazs, A.C. Designing synthetic vesicles that engulf nanoscopic particles. *J. Chem. Phys.* **2007**, *127*, 084703. [[CrossRef](#)] [[PubMed](#)]
60. Yang, K.; Ma, Y.-Q. Computer simulation of the translocation of nanoparticles with different shapes across a lipid bilayer. *Nat. Nanotechnol.* **2010**, *5*, 579–583. [[CrossRef](#)] [[PubMed](#)]
61. Grafmuller, A.; Shillcock, J.; Lipowsky, R. The fusion of membranes and vesicles: Pathway and energy barriers from dissipative particle dynamics. *Biophys. J.* **2009**, *96*, 2658–2675. [[CrossRef](#)]
62. Brown, A.; Vousden, M.; Rast, A.; Bragg, G.M.; Thomas, D.; Beauont, J.; Naylor, M.; Mokhov, A. POETS: Distributed event-based computing—Scaling behaviour. In Proceedings of the International Conference on Parallel Computing, Prague, Czech Republic, 10–13 September 2019.
63. Oldfield, C.J.; Dunker, A.K. Intrinsically Disordered Proteins and Intrinsically Disordered Protein Regions. *Annu. Rev. Biochem.* **2014**, *83*, 553–584. [[CrossRef](#)] [[PubMed](#)]
64. Zeng, M.; Chen, X.; Guan, D.; Xu, J.; Wu, H.; Tong, P.; Zhang, M. Reconstituted Postsynaptic Density as a Molecular Platform for Understanding Synapse Formation and Plasticity. *Cell* **2018**, *174*, 1172–1187. [[CrossRef](#)]
65. Wu, X.; Cai, Q.; Shen, Z.; Chen, X.; Zeng, M.; Du, S.; Zhang, M. RIM and RIM-BP Form Presynaptic Active-Zone-like Condensates via Phase Separation. *Mol. Cell* **2019**, *73*, 971–984. [[CrossRef](#)]
66. Castagna, J.; Guo, X.; Seaton, M.; O’Cais, A. Towards extreme scale dissipative particle dynamics simulations using multiple GPGPUs. *Comput. Phys. Commun.* **2020**, *251*, 107159. [[CrossRef](#)]
67. Humphrey, W.; Dalke, A.; Schulten, K. VMD—Visual Molecular Dynamics. *J. Mol. Graphics.* **1996**, *14*, 33–38. [[CrossRef](#)]