Politecnico di Torino

Thesis for
M.Sc. programme in Physics of Complex Systems

Molecular sorting on a fluctuating membrane

Supervisors:
Prof. Andrea Gamba
Prof. Luca Dall'Asta

Candidate:
Damiano Andreghetti

July 2023

Abstract

Molecular sorting is a fundamental ordering process taking place in eukaryotic cells, whereby proteins and other biomolecules are sorted and distilled into lipid vesicles. This process aims at countering the homogenizing effect of diffusion. Molecules attach to lipid membranes and laterally diffuse, then due to a variety of direct and indirect interactions, these molecules tend to aggregate in domains characterized by specific chemical compositions. In their turn, such may promote membrane bending and fission, and are ultimately packed into lipid vesicles, which are then delivered to appropriate intracellular destinations with the aid of molecular motors. This vesicle distillation phenomenon can be investigated from a physics perspective. The process of molecules attaching and sorting on the membrane and then being extracted as aggregates can be seen as a system driven out of its thermodynamic equilibrium which eventually reaches a stationary state. In previous studies, this phenomenon was investigated without accounting for the effect of membrane fluctuations and molecules coupling to the membrane. However the presence of biomolecular inclusions on the membrane perturbs locally some of its properties (e.g. bending rigidity or local spontaneous curvature) and gives rise to membrane-mediated interactions between inclusions. This kind of interactions has been widely studied since the 90s. Seems thus reasonable to ask what is the role of these fluctuation-induced interactions in the process of distillation taking place on lipid membranes. These long-range membrane-mediated interactions are not pairwise additive, thus their effect on the process of molecular sorting, where the membrane is rich of inclusions, is non-trivial. In this work this phenomenon is investigated numerically, with particular interest in the efficiency of the sorting process. We concluded that the role of fluctuation-induced forces is divided into two regimes: one in which they enhance the sorting process and one in which they impair it.

Contents

1 Sorting process

Inside cells each function is performed by specific proteins and other chemical factors, these undergo constant homogenization due to diffusion. To maintain their correct functioning eukaryotic cells developed a variety of self-organized compartmentalization processes, most of which are active and energy-consuming [38]. These phenomena are particularly charming from a physics perspective, indeed mixing and demixing transitions have been thoroughly studied in physics.

In this thesis the focus is in particular on the vesicle distillation process, a molecular sorting process that aims at producing lipid vesicles containing specialized chemical factors ready to be delivered to appropriate subcellular regions. This kind of process takes place on lipid membranes, in particular on plasma membrane, on inner membrane bodies (like endosomes) and in the Golgi membrane network. Here chemical factors attach to the lipid bilayer and laterally diffuse on it, then due to a variety of interactions they phase separate into specialized domains. These domains with the help of biomolecules promoting membrane bending and fission are then packed into lipid vesicles ready to be delivered to appropriate regions.

From a physics perspective it is fascinating to understand molecular sorting as a systemic process, beyond molecular detail. Here we describe a recently proposed phenomenological theory for this distillation process. Later in the thesis this model will be enriched taking into account the role of membrane fluctuations.

1.1 Phenomenological theory

Has recently been proposed that the self-organized distillation process emerges from two main ingredients: a) the tendency of biomolecules to phase-separate in localized submicrometric domains, and b) domain-induced vesicle nucleation [39]. A phenomenological theory based on this general physical picture was explored both theoretically and by means of numerical simulations [39, 41, 44]. The model described a situation where molecules arrive on a membrane region with a flux ϕ, diffuse and aggregate into localized enriched domains, and these domains are removed from the membrane, after reaching a characteristic size R_E. In this context the system is constantly forced out of its thermodynamic equilibrium however eventually reaches a stationary state.

In regimes of strong direct interactions, as predicted by diffusion-limited aggregation (DLA) theory fractal clusters may appear. It's important to notice that the existence of a threshold size for domains above which they're extracted ensures their shape remains approximately round. During the sorting process, once domains reach a critical size R_c they grow irreversibly absorbing free particles diffusing towards them, this picture is similar to that described by Lifshitz-Slezov (LS) theory [26]. In a situation where the average molecule density is $\bar{n}$ and for domains larger than R_c the density of molecules on the domain boundary n_0 is independent of its size. If inter-domain distance L is much larger than R_E (that is maximum domain size) the density difference Δn between regions far from domains and on their boundaries is approximately given by $\bar{n} - n_0 > 0$. In the statistically stationary state of the sorting process the Δn is kept finite by a constant flux of particles, differently from LS theory.

Quasi-static profile The quasi-static profile of freely diffusing molecules in the vicinity of a domain of size R can be evaluated. The quasi-static condition allows us to reduce the diffusion equation to a Laplace equation.

$$\frac{\partial n}{\partial t} = D\nabla^2 n \longrightarrow \nabla^2 n = 0 \tag{1}$$

We solve it by imposing Dirichlet boundary conditions, setting $n(R) = n_0$ and $n(L) = \bar{n}$. The result is

$$n(r) = n_0 + \frac{\ln(r/R)}{\ln(L/R)}\Delta n \tag{2}$$

Domain growth We can use this result to evaluate the flux of particles Φ_R coming towards a domain. This is done integrating the flux density $-D\nabla n$ over a circle of radius $r \gg R$:

$$\Phi_R = 2\pi R D \partial_r n(r)|_{r=R} = \frac{2\pi D \Delta n}{\ln(L/R)} \tag{3}$$

From this can easily be derived the dynamics equation for domain growth:

$$\dot{R} = \frac{A_0 D \Delta n}{R \ln(L/R)} \tag{4}$$

where A_0 is the area occupied by a molecule in the domain, assumed to be $A_0 \ll R_E$.

Domain size distribution If we call $N(t, R)$ the number of domains of size between R and $R + dR$, this quantity satisfies the Smoluchowski equation:

$$\frac{\partial N}{\partial t} + \frac{\partial}{\partial R}(\dot{R}N) = -\gamma(R)N \tag{5}$$

where $\gamma(R)$ is the rate by which domains of size R are removed from the system. This rate is a parameter encoding mesoscopic effects of vesicle extraction, thus abstracting from complicated molecular details. A stationary solution to Eq.5 is:

$$N_{st}(R) = \frac{JR\ln(L/R)}{D\Delta n} exp\left[-\int_0^R dr \frac{r\ln(L/r)\gamma(r)}{A_0 D\Delta n}\right] \tag{6}$$

Now we assume that $\gamma(R)$ is negligible for $R < R_E$ and strongly suppresses $N_{st}(R)$ for $R > R_E$. The factor J in Eq.6 can be evaluated by noticing that at stationary state the average flux $\int \Phi_R N_{st}(R) dR$ must equal the incoming flux of molecules per unit area ϕ, thus giving $J \sim \phi/R_E^2$. Being $\gamma(R)$ negligible for $R < R_E$ then Eq.6 is linear in that region (with logarithmic corrections).

Existence of an optimal sorting regime Now we can show that exists an optimal sorting regime in terms of sorting rate, given by the inverse of the average residence time $\bar{T}$. The latter can be divided into two parts $\bar{T} = \bar{T}_f + \bar{T}_d$, respectively the average time spent by a freely diffusing molecule to reach a domain and the average time spent inside the domain before extraction. For evenly distributed domains the first contribution is inversely proportional to the

average number of domains N_d per unit area, namely $\bar{T}_f \sim (1/DN_d)$. Being $N_d = \int N_{st}(R)dR \sim \phi/(D\Delta n)$ we get $\bar{T}_f \sim \Delta n/\phi$. For the second contribution instead, using Eq.3 we get $\bar{T}_d \sim R_E^2/(A_0\Phi_R) \sim R_E^2/(DA_0\Delta n)$.

The rate of formation of new domains can be estimated as

$$dN_d/dt = CD\bar{n}^2 \tag{7}$$

where C is a dimensionless quantity characterizing the efficiency of absorption of single molecules by the germ of a domain. At stationary state this rate Eq.7 is equal to $N_d/\bar{T}_d$. Thus we get

$$\bar{n} \sim \left(\frac{\phi A_0}{CDR_E^2}\right)^{-1/2} \tag{8}$$

If we assume $n_0 \lesssim \Delta n$ we have $\Delta n \sim \bar{n}$ and therefore

$$\begin{aligned} \bar{T}_f &\sim C^{-1/2}\frac{A_0^{1/2}}{(D\phi)^{1/2}R_E} \\ \bar{T}_d &\sim C^{1/2}\frac{R_E^3}{(D\phi)^{1/2}A_0^{3/2}} \end{aligned} \tag{9}$$

The average residence time $\bar{T} = \bar{T}_f + \bar{T}_d$ has a minimum w.r.t. C in $C \sim A_0^2/R_E^4 \ll 1$. Around this value the process is optimal, meaning that is more efficient in terms of time needed to sort molecules. It's possible to retrieve two scaling relations for this regime

$$\begin{aligned} \bar{T}_f \sim \bar{T}_d &\sim \frac{R_E}{(DA_0\phi)^{1/2}} \\ \bar{n} \sim \Delta n &\sim \frac{\phi^{1/2}R_E}{(DA_0^{1/2})} \end{aligned} \tag{10}$$

Density as a proxy for the sorting rate At stationarity it was proven that the average molecule density $\bar{\rho}$ on the membrane is proportional to the mean residence time $\bar{T}$ of molecules, and thus can be used to evaluate indirectly the sorting rate $\bar{T}^{-1}$. To prove this we let the residence time of a molecule on the membrane be a stochastic variable described by probability density $p(t)$. Moreover being ϕ the flux of incoming molecules, then on average each infinitesimal time interval dt a quantity ϕdt of molecules enters the system. Adding the condition that the residence time of such molecules has not elapsed yet we get:

$$\begin{aligned} \bar{\rho} &= \int_0^\infty Prob(T > t)\phi dt = \phi\int_0^\infty \left(\int_t^\infty p(\tau)d\tau\right)dt \\ &= \phi\int_0^\infty \tau p(\tau)d\tau = \phi\bar{T} \end{aligned} \tag{11}$$

Here we used the fact that the integral of tail distribution is equal to the expectation value. This means that the efficiency of the sorting process can be evaluated by measuring $\bar{\rho}$.

1.2 Numerical results

To further explore the behaviour of such a process for a wide range of parameters, numerical results were obtained with a minimal single occupancy lattice gas model [39]. The lattice represents the lipid membrane on which the distillation takes place, the system evolves according to the following three elementary mechanisms:

1. **Insertion:** molecules are inserted in empty sites at a rate k_I, leading to a flux $\phi = k_I(1-\rho)$ of molecules coming on average towards each lattice site.

2. **Diffusion:** molecules diffuse on the lattice at a rate $k_D/g^{n_{occ}}$, where n_{occ} is the number of occupied nearest neighbours and $g > 1$ plays the role of direct nearest neighbour interaction.

3. **Extraction:** once clusters of molecules of linear size $l \sim R_E/A_0^{1/2}$ are formed, they're removed at a rate k_E, which we will assume to be infinite.

This simple lattice gas model mimics the distillation process and gives insights into the behaviour of the system depending on parameters. In Fig.1 is represented this process schematically.

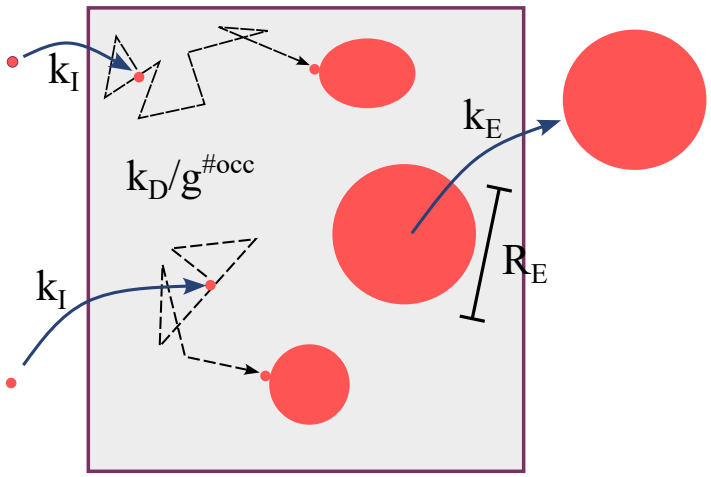

Figure 1: Schematic representation of the sorting process. The system is constantly force out of its thermodynamic equilibrium. Particles attach on the membrane at a rate k_I and they laterally diffuse on it at a rate k_D/g^{n_occ}. Parameter g controls nearest-neighbour interactions with n_{occ} being the number of occupied sites near the particle. Domains form and grow up to an extraction threshold R_E. Once they reach the extraction size they're removed form the system.

In particular it was found that the distillation process finds a minimum w.r.t. to parameter g, controlling direct interaction. This minimum becomes more and more appreciable as the ratio of k_I/k_D decreases. These results are summarized in Fig 2. Intuitively k_I/k_D represents the average time a molecule has to explore

the lattice (and possibly join an aggregate) before another molecule is inserted. As described above the average molecule density $\bar{\rho}$ can be used as an indirect measure of the mean residence time. This minimum corresponds thus to the same found before through the phenomenological approach. Indeed the dimensionless parameter C represents the efficiency of adsorption of single molecules which increases monotonically with g.

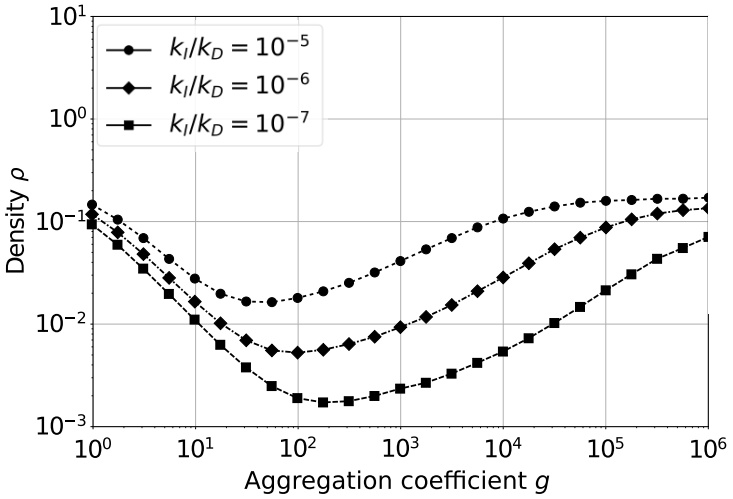

Figure 2: Image is from [39]. This plot shows that exists an optimal g value at which average density and thus the mean residence time of molecules is minimum. This minimum becomes more and more appreciable as k_I/k_D decreases.

These results were also compared with experimental data regarding low-density lipoproteins (LDL) and their receptors (LDLR) on the plasma membrane that laterally diffuse, aggregate and are internalized into endocytic vesicles. The comparison showed that this sorting process takes place in the vicinity of the previously described optimal regime; this may be related to evolutionary constraints that tuned the involved proteins to get the maximal sorting efficiency.

In Fig.3 a phase diagram of the process is shown. The optimal sorting regime is highlighted by the dashed line and the area where experimental data lie is the shaded one.

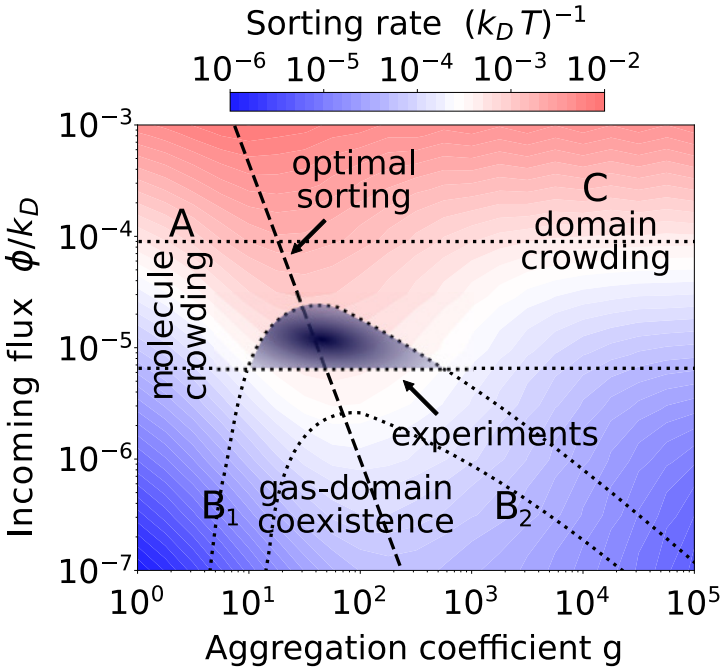

Figure 3: Image from [39]. This phase diagram depicts nondimensional sorting rate $(k_D\bar{T})^{-1}$ as a function of aggregation coefficient g and nondimensional flux ϕ/k_D. Phase areas A and C are high-density phases characterized respectively by molecule crowding and domain crowding. B is a low-density phase, divided into regions of less (B_1) and more (B_2) dilute gas. The shaded area pointed by the arrow is the area where experimental data lie.

2 Field-theoretic description of lipid bilayers

Since the goal of this thesis is to account for the effect of membrane fluctuation in the sorting process described above, in this section is reported a description of how in physics lipid membranes are usually studied. Particular attention will be given to the field-theoretic description since it enables us to evaluate fluctuation-induced forces through perturbative analysis.

2.1 Canham-Helfrich Hamiltonian

The simplest component of a biomembrane is the lipid bilayer which has a thickness that is much smaller than the membrane length scale. Therefore mathematically it is usually described as a two-dimensional surface.

The elastic energy usually associated with this surface is described by the following Hamiltonian (usually named Canham-Helfrich Hamiltonian after the physicists who first proposed it in the early 70s [1, 2][1])

$$\mathcal{H}_{el} = \int dS \left[\frac{\kappa}{2}(2M - C_0)^2 + \bar{\kappa}K + \sigma \right] \tag{12}$$

Where κ and $\bar{\kappa}$ are bending rigidities, $M = (c_1 + c_2)/2$ is the mean curvature, C_0 is the spontaneous curvature, K is the Gaussian curvature and σ is the surface tension. The squared term somehow measures the local deviation from the spontaneous curvature $C_0 = \frac{2}{R_0}$. The Gaussian curvature is defined as the product of the two principal curvatures $K = c_1 c_2$. At scales below $l_\sigma = \sqrt{\kappa/\sigma}$ the membrane tension has a weak influence on the membrane behaviour [43].

Moreover, often the term regarding Gaussian curvature is discarded when considering closed surfaces of fixed topology, since leads to a constant term. Indeed the Gauss-Bonnet theorem implies:

$$\int K dS = 2\pi\chi_E \tag{13}$$

where χ_E is Euler characteristic, a topological invariant. For polyhedra, it is given by $\chi_E = N_v - N_e + N_f$, where N_v is the number of vertices N_e is the number of edges and N_f is the number of faces.

2.2 Quasi-flat approximation: Monge gauge

Consider the following form of the Canham-Helfrich Hamiltonian, where spontaneous curvature is assumed to be zero:

$$H_0 = \int dS \left\{ \frac{\kappa}{2}(2M)^2 + \bar{\kappa}K + \frac{\sigma}{2} \right\} \tag{14}$$

A surface is a $d-1$ dimensional object embedded in a d dimensional space. Points of the surface are thus addressed by a d dimensional vector $\vec{r}(\vec{u})$, a function of the $d-1$ coordinates. Any set of $d-1$ independent coordinates can be used to parametrize the surface. If we consider the context of a quasi-flat surface, where the amplitude of membrane fluctuations is much smaller than its longitudinal

[1] Actually they proposed slightly different formulations of the free energy but they've been proven to be equivalent through the Gauss-Bonnet theorem.

size, it's convenient to reparametrize the surface in the so-called Monge gauge. In this gauge each surface point $\vec{r}$ is described as its height w.r.t. a reference plane, i.e. $\vec{r}(\vec{x}_\perp) = (\vec{x}_\perp, h(\vec{x}_\perp))$. The new coordinate $\vec{x_\perp}$ represent the position on the reference plane.

Surface element Our surface can be parametrized with two variables u_1 and u_2, and an infinitesimal displacement along the curve can be written as

$$d\vec{r} = \frac{\partial \vec{r}}{\partial u_1} du_1 + \frac{\partial \vec{r}}{\partial u_2} du_2 = \vec{t}_1 du_1 + \vec{t}_2 du_2 \tag{15}$$

where we called $\vec{t}_1$ and $\vec{t}_2$ the vectors tangent to the surface at position (u_1, u_2). It's important to notice that while u_1 and u_2 are orthogonal coordinates, the same is not true for $\vec{t}_1$ and $\vec{t}_2$. The infinitesimal surface element dS is then given by

$$dS = \left|\vec{t}_1 \times \vec{t}_2\right| du_1 du_2 \tag{16}$$

In Monge gauge, where $\vec{r} = (x, y, h(x, y))^T$ this reads

$$dS = \left|(1, 0, \partial_x h)^T \times (0, 1, \partial_y h)^T\right| dxdy = \sqrt{1 + (\nabla_\perp h)^2} dxdy \tag{17}$$

where $\nabla_\perp$ is the gradient operator in the coordinates of the reference plane.

Curvatures For each point of the surface curvature can be defined, in this paragraph we derive the expression for mean and Gaussian curvature in this gauge. We recall that mean and Gaussian curvatures are defined as

$$\begin{aligned} H &:= \frac{c_1 + c_2}{2} \\ K &:= c_1 c_2 \end{aligned} \tag{18}$$

where c_1 and c_2 are the principal curvatures. Exist different ways to derive these, here we'll use the shape operator $\mathcal{S}$. This operator has the principal curvatures as eigenvalues and thus the following relations can be used to determine mean and Gaussian curvature:

$$\begin{aligned} H &= \frac{1}{2} Tr(\mathcal{S}) \\ K &= det(\mathcal{S}) \end{aligned} \tag{19}$$

Shape operator can be computed as:

$$\mathcal{S} = b g^{-1} \tag{20}$$

where g is the *metric* or *first fundamental form*, while b is the *second fundamental form*. These are tensors defined as [18]:

$$\begin{aligned} g_{ij} &= \vec{t}_i \cdot \vec{t}_j \\ b_{ij} &= \vec{t}_{i,j} \cdot \vec{n} \end{aligned} \tag{21}$$

Here $\vec{t}_i$ are the vectors tangent to the surface defined above, $\vec{t}_{i,j} = \partial_i \partial_j \vec{r}$ and $\vec{n}$ is the unit vector normal to the surface, defined by

$$\vec{n} = \frac{\vec{t}_1 \times \vec{t}_2}{|\vec{t}_1 \times \vec{t}_2|} = \frac{1}{\sqrt{1 + \partial_x h^2 + \partial_y h^2}} \begin{pmatrix} -\partial_x h \\ -\partial_y h \\ 1 \end{pmatrix} \tag{22}$$

In our chosen parametrization, using h_a as a short-hand for $\partial h/\partial a$ we get:

$$\mathcal{S} = \frac{1}{(1+h_x^2+h_y^2)^{3/2}} \begin{pmatrix} h_{xx}(1+h_y^2) - h_{xy}h_x h_y & h_{xy}(1+h_x^2) - h_{xx}h_x h_y \\ h_{xy}(1+h_y^2) - h_{yy}h_x h_y & h_{yy}(1+h_x^2) - h_{xy}h_x h_y \end{pmatrix} \tag{23}$$

The expression we get for mean and Gaussian curvature is then:

$$H = \frac{h_{xx}(1+h_y^2) + h_{yy}(1+h_x^2) - 2h_{xy}h_x h_y}{2(1+h_x^2+h_y^2)^{3/2}} K = \frac{h_{xx}h_{yy} - (h_{xy})^2}{(1+h_x^2+h_y^2)^2} \tag{24}$$

We now have all the expressions needed to rewrite Eq.14 in Monge gauge. Usually this is done in a quasi-flat approximation i.e. supposing the gradient of h to be small. Keeping terms up to second order in gradient, the Hamiltonian reads:

$$H_0 = \int dxdy \left\{ \frac{\kappa}{2}[\nabla^2 h]^2 + \bar{\kappa}(\partial_{xx}^2 h \partial_{yy}^2 h - (\partial_{xy}^2 h)^2) + \frac{\sigma}{2}(\nabla h)^2 + \mathcal{O}((\nabla h)^2) \right\} \tag{25}$$

2.3 Overview of membrane-mediated interactions

Shortly after the introduction of models to study the dynamics and equilibrium of biological membranes, many studies have addressed several ways in which membrane properties may be influenced by its fluidity and elasticity. Depending on the model chosen and on the particular regime of the system many different membrane-mediated interactions may arise [34]. In this work, we're going to investigate the role of these interactions in the aggregation of membrane proteins. Here we summarize the main 4 mechanisms behind these forces:

1. Capillary forces: the free energy caused by a strong variation in composition across the interface is what generates the line tension on interfacial boundaries. The interfacial tension tends to reduce the domain perimeter length and can thus generate a force (Figure 4, panel B).

2. Lipid depletion forces: Lipid molecules close to the 'hard' interface of integral membrane proteins are conformationally restricted. When these lipids are mobile relative to the proteins, they may diffuse away so as to increase their entropy. The effective expulsion of lipids from between nearby proteins leads to an effective short-range attraction between the proteins (Figure 4, panel C).

3. Curvature-induced forces: membrane proteins can locally deform the membrane through different mechanisms. In response, the membrane changes the distribution of these proteins to minimize its overall elastic energy. This mechanism drives the proteins to associate or dissociate, depending on the lateral protein shape and the magnitude of the induced curvature (Figure 4, panel D).

4. Thermal Casimir-like forces: membranes undergo ceaseless thermal fluctuations. The presence of proteins that modify these fluctuations may generate forces that may drive them to cluster in order to maximize the entropy of the membrane's fluctuations. The mechanism behind these

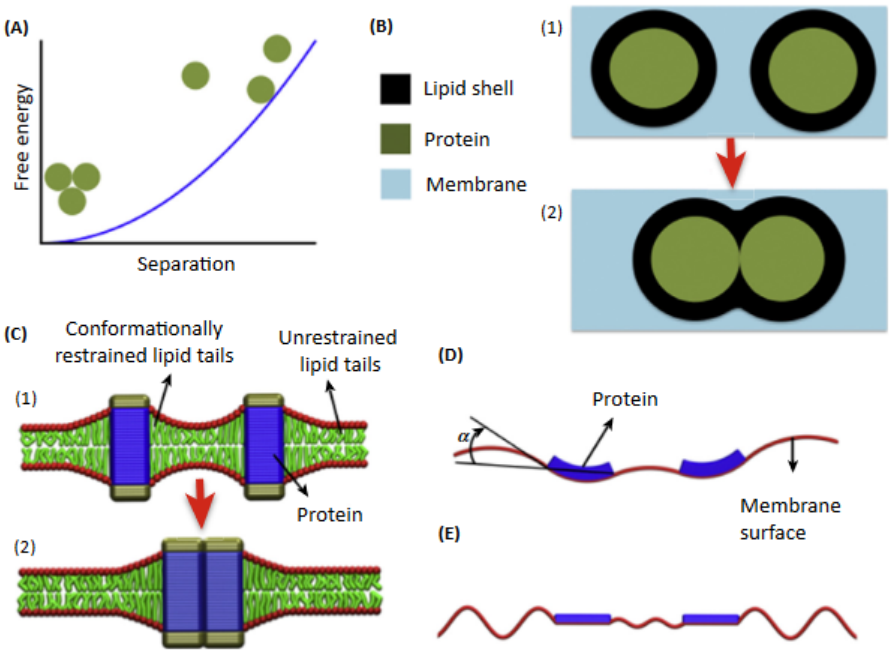

Figure 4: Figure from [34]: (A) molecules aggregate if the net change in free energy is favourable. (B) Capillary forces drive proteins to clusters in order to minimize free energy associated with lipid shells. (C) Lipid depletion interactions. (D) Curvature-induced forces. (E) Thermal Casimir-like forces.

forces is similar to the one proposed by Casimir in Quantum Field Theory to describe forces arising from the alteration of the vacuum expectation value of the energy due to the presence of interface materials. Recalling now that fluid membranes are elastic and free to fluctuate we can expect that in a similar way rigid membrane inclusions alter fluctuations and in response membrane rearranges them in order to minimize overall free energy (Figure 4, panel E). An explicit calculation of how rigid inclusions suppress some of the membrane fluctuation modes is carried out in [14]. This reduction of fluctuation modes result in an entropic force that acts on inclusions at equilibrium. This elegant calculation shows how these forces arise from purely entropic arguments.

However, this classification is mostly formal since depending on the model used to describe the membrane and inclusions more than one of these forces may arise and it's difficult to genuinely decouple the different interaction mechanisms. In general we'll observe a mixture of them.

2.4 Phenomenological theory for fluctuation-induced forces

The interactions between membrane inclusions imposing local perturbations on the membrane have been studied since the 90s [7, 8, 12, 11, 35, 43]. Here we report a phenomenological approach to this kind of interaction. Suppose to have a quasi-flat membrane described by the usual Canham-Helfrich Hamiltonian in Monge Gauge:

$$H_0 = \int d\vec{x_\perp} \left\{ \frac{\kappa}{2} [\nabla_\perp^2 h]^2 + \frac{\sigma}{2} (\nabla h)^2 \right\} \tag{26}$$

where the field $h(\vec{x})$ has Gaussian statistics and the Gaussian curvature term is discarded since we're focusing on surfaces of fixed topology.

2.4.1 Perturbative expansion

Suppose now the energy of the membrane with proteins on it can be described by

$$H = H_0 + H_{int} \tag{27}$$

where H_{int} is the coupling energy of the molecules with the membrane. Recalling now the definition for the partition function:

$$Z = e^{-\beta \mathcal{F}} = \int Dh (e^{-\beta H_0 - \beta H_{int}}) \tag{28}$$

Multiplying both sides by $exp(\beta \mathcal{F}_0)$ (where $\mathcal{F}_0$ is the free energy associated to H_0 only) we get

$$e^{-\beta(\mathcal{F}-\mathcal{F}_0)} = \frac{\int Dh (e^{-\beta H_0 - \beta H_{int}})}{\int Dh (e^{-\beta H_0})} = \left\langle e^{-\beta H_{int}} \right\rangle_0 \tag{29}$$

Using now the cumulant expansion (justified for small H_{int})

$$\left\langle e^V \right\rangle_0 = \left\langle 1 + V + \frac{1}{2} V^2 + ... \right\rangle_0 = exp[< V >_0 + \frac{1}{2} (< V^2 >_0 - < V >_0^2) + O(V^3)] \tag{30}$$

we retrieve

$$\Delta \mathcal{F} = \mathcal{F} - \mathcal{F}_0 = < H_{int} >_0 - \frac{1}{2} \beta (< H_{int}^2 >_0 - < H_{int} >_0^2) + ... \tag{31}$$

Assuming now that the energy H_{int} is given by the sum of the contribution of the different inclusions at different positions $\vec{r}_j$:

$$H_{int} = \sum_j H_{int}^j (\vec{r}_j) \tag{32}$$

Now terms in Eq.31 of the kind $< H_{int} >_0$ won't give rise to a free energy dependence on both the molecules' positions (which is what we're interested in to compute interactions) [11].
Thus we can evaluate the interaction between inclusions as

$$U = -\frac{1}{2 k_B T} \left\langle H_{int}^2 \right\rangle_0 \tag{33}$$

2.4.2 Linear curvature coupling

The first case we'll focus on is the one of linear coupling with curvature, which represents the case of inclusions that impose locally a spontaneous curvature to the membrane. Indeed, the energy per area contribution for a membrane patch covered by an inclusion imposing a local spontaneous curvature c_0 will be

$$\frac{E_{inclusion}}{A_{inclusion}} = \frac{\kappa}{2}(2M - c_0)^2 = \frac{\kappa}{2}(2M)^2 - \kappa(2M)c_0 + \frac{\kappa}{2}c_0^2 \tag{34}$$

So the energy differs from the unperturbed case H_0 for a term linear in the mean curvature ($2M$) and a term renormalizing surface tension (we will ignore this contribution here). This means that phenomenologically we're interested in

$$H_{int} = \sum_j A_j \nabla^2 h(\vec{r}_j) \tag{35}$$

Where the sum is over the inclusions and A_j is a phenomenological constant; from the above Eq.34 we can suppose this constant be proportional to the inclusion area, bending rigidity and the imposed spontaneous curvature, this is in agreement with non-phenomenological results [8]. For two identical inclusions, i.e. $A_1 = A_2 = A$, located at $\vec{r}_1$ and $\vec{r}_2$, the interaction energy will be (discarding again terms not depending on the distance of proteins):

$$U = -\frac{A^2}{k_B T} \left\langle \nabla^2 h(\vec{r}_1) \nabla^2 h(\vec{r}_2) \right\rangle_0 \tag{36}$$

We can now evaluate the correlation between curvature ($\nabla^2 h$) at different positions, switching to Fourier space:

$$\begin{aligned}
\left\langle \nabla^2 h(\vec{r}_1) \nabla^2 h(\vec{r}_2) \right\rangle_0 &= \left\langle \nabla^2 \left(\int \frac{d\vec{k}}{(2\pi)^2} \hat{h}(\vec{k}) e^{i\vec{k}\cdot\vec{r_1}} \right) \nabla^2 \left(\int \frac{d\vec{q}}{(2\pi)^2} \hat{h}(\vec{q}) e^{i\vec{q}\cdot\vec{r_2}} \right) \right\rangle_0 \\
&= \int \frac{d\vec{k} d\vec{q}}{(2\pi)^4} k^2 q^2 \left\langle \hat{h}(\vec{k}) \hat{h}(\vec{q}) \right\rangle_0 e^{i(\vec{k}\cdot\vec{r_1} + \vec{q}\cdot\vec{r_2})} \\
&= \int \frac{d\vec{q}}{(2\pi)^2} q^4 \hat{G}_0(\vec{q}) e^{i\vec{q}\cdot(\vec{r_1} - \vec{r_2})}
\end{aligned} \tag{37}$$

In the last line we used $\left\langle \hat{h}(\vec{k}) \hat{h}(\vec{k'}) \right\rangle_0 = (2\pi)^2 \delta(\vec{k} + \vec{k'}) \hat{G}_0(\vec{k})$ where $\hat{G}_0(\vec{k})$ is the Gaussian propagator in Fourier space.
Explicitly writing the latter and defining $\vec{r} = \vec{r}_1 - \vec{r}_2$ we get

$$\begin{aligned}
\left\langle \nabla^2 h(\vec{r}_1) \nabla^2 h(\vec{r}_2) \right\rangle_0 &= \int \frac{d\vec{q}}{(2\pi)^2} \frac{q^4 k_B T}{\kappa q^4 + \sigma q^2} e^{i\vec{q}\cdot\vec{r}} \\
&= \frac{k_B T}{\kappa} \int \frac{d\vec{q}}{(2\pi)^2} \frac{q^2}{q^2 + \sigma/\kappa} e^{i\vec{q}\cdot\vec{r}} \\
&= \frac{k_B T}{\kappa} \delta(\vec{r}) - \frac{\sigma k_B T}{\kappa^2} \int \frac{d\vec{q}}{(2\pi)^2} \frac{1}{q^2 + \sigma/\kappa} e^{i\vec{q}\cdot\vec{r}}
\end{aligned} \tag{38}$$

Switching now to polar coordinates and assuming (without loss of generality) that x axis lies along $\vec{r}$ direction:

$$\begin{aligned}\left\langle \nabla^2 h(\vec{r}_1)\nabla^2 h(\vec{r}_2)\right\rangle_0 &= \frac{k_B T}{\kappa}\delta(\vec{r}) - \frac{\sigma k_B T}{\kappa^2}\int \frac{dqd\theta}{(2\pi)^2}\frac{q}{q^2+\sigma/\kappa}e^{iqrcos\theta} \\ &= \frac{k_B T}{\kappa}\delta(\vec{r}) - \frac{\sigma k_B T}{\kappa^2}\int \frac{dq}{2\pi}\frac{q}{q^2+\sigma/\kappa}\int \frac{d\theta}{2\pi}e^{iqrcos\theta} \\ &= \frac{k_B T}{\kappa}\delta(\vec{r}) - \frac{\sigma k_B T}{2\pi\kappa^2}\int dq\frac{q}{q^2+\sigma/\kappa}\mathcal{J}_0(qr) \\ &= \frac{k_B T}{\kappa}\delta(\vec{r}) - \frac{\sigma k_B T}{2\pi\kappa^2}\mathcal{K}_0\left(\sqrt{\frac{\sigma}{\kappa}}r\right)\end{aligned} \tag{39}$$

For the last two equalities were used: $\int_0^{2\pi}\frac{d\theta}{2\pi}e^{ipcos\theta} = \mathcal{J}_0(p)$, the integral definition of the Bessel function of the first kind and $\int_0^\infty \frac{dk\ k}{k^2+m^2}\mathcal{J}_0(kr) = \mathcal{K}_0(mr)$, a known integral giving rise to modified Bessel function of the second kind $\mathcal{K}_0(x)$. Thus eventually the expression of the interaction energy between the two proteins, at a finite distance r, is

$$U = \frac{A^2\sigma}{2\pi\kappa^2}\mathcal{K}_0\left(\sqrt{\frac{\sigma}{\kappa}}r\right) \tag{40}$$

The interaction for two identical inclusions is thus repulsive, linear in the tension and for $r < \sqrt{\frac{\kappa}{\sigma}}$ we find

$$U \sim \frac{A^2\sigma}{2\pi\kappa^2} ln\left(\frac{2\sqrt{\kappa/\sigma}}{r}\right)$$

.

2.4.3 Quadratic curvature coupling: interaction arise without surface tension

We are now interested in another kind of interaction which arises even in absence of surface tension. Thus from now on we'll consider $\sigma = 0$. Suppose now that molecules interact with the membrane through a quadratic coupling in curvature. This can happen both through mean curvature:

$$H_{B,int} = \sum_j B_j[\nabla^2 h(\vec{r_j})]^2 \tag{41}$$

and Gaussian curvature:

$$H_{D,int} = \sum_j D_j[\partial_x^2 h(\vec{r_j})\partial_y^2 h(\vec{r_j}) - (\partial_x\partial_y h(\vec{r_j}))^2] \tag{42}$$

This kind of coupling corresponds to inclusions locally altering bending rigidities of the membrane. If only one of the above coupling is present the resulting interaction is zero. We verify this in the following calculation (supposing again to have identical inclusions):

$$\begin{aligned}\left\langle H^2_{B,int}\right\rangle &= B^2\left\langle[\nabla^2 h(\vec{r}_1)]^2[\nabla^2 h(\vec{r}_2)]^2\right\rangle_0 \\ &= B^2\int\frac{d\vec{k}d\vec{q}d\vec{k_1}d\vec{q_1}}{(2\pi)^8}k^2q^2k_1^2q_1^2\left\langle\hat{h}(\vec{k})\hat{h}(\vec{q})\hat{h}(\vec{k_1})\hat{h}(\vec{q_1})\right\rangle_0 e^{i(\vec{k}+\vec{k_1})\cdot\vec{r_1}+i(\vec{q}+\vec{q_1})\cdot\vec{r_2}} \\ &= B^2\int\frac{d\vec{k}d\vec{q}d\vec{k_1}d\vec{q_1}}{(2\pi)^8}k^2q^2k_1^2q_1^2\Big[\left\langle\hat{h}(\vec{k})\hat{h}(\vec{q})\right\rangle_0\left\langle\hat{h}(\vec{k_1})\hat{h}(\vec{q_1})\right\rangle_0+\left\langle\hat{h}(\vec{k})\hat{h}(\vec{k_1})\right\rangle_0 \\ &\left\langle\hat{h}(\vec{q})\hat{h}(\vec{q_1})\right\rangle_0+\left\langle\hat{h}(\vec{k})\hat{h}(\vec{q_1})\right\rangle_0\left\langle\hat{h}(\vec{q})\hat{h}(\vec{k_1})\right\rangle_0\Big]e^{i(\vec{k}+\vec{k_1})\cdot\vec{r_1}+i(\vec{q}+\vec{q_1})\cdot\vec{r_2}}\end{aligned} \tag{43}$$

Where for the last equality Wick's theorem for a Gaussian field has been exploited to resolve the four point correlation function. Inserting, as we did for the previous case, the expression for the Gaussian propagator (remember that in absence of surface tension $\hat{G}_0(\vec{q}) = k_BT/(\kappa q^4)$).

$$\begin{aligned}\left\langle H^2_{B,int}\right\rangle_0 &= B^2\int\frac{d\vec{k}d\vec{q}}{(2\pi)^4}k^4q^4\hat{G}_0(\vec{k})\hat{G}_0(\vec{q})+ \\ &+B^2\int\frac{d\vec{k}d\vec{k_1}}{(2\pi)^4}k^4k_1^4\hat{G}_0(\vec{k})\hat{G}_0(\vec{k_1})e^{i\vec{k}\cdot(\vec{r_1}-\vec{r_2})+i\vec{k_1}(\vec{r_1}-\vec{r_2})}+ \\ &+B^2\int\frac{d\vec{k}d\vec{q}}{(2\pi)^4}k^4q^4\hat{G}_0(\vec{k})\hat{G}_0(\vec{q})e^{i\vec{k}\cdot(\vec{r_1}-\vec{r_2})+i\vec{q}(\vec{r_1}-\vec{r_2})} \\ &= const.+\frac{2B^2k_BT}{\kappa^2}\int\frac{d\vec{k}d\vec{q}}{(2\pi)^4}e^{i\vec{k}\cdot(\vec{r_1}-\vec{r_2})+i\vec{q}(\vec{r_1}-\vec{r_2})} \\ &= const.+\frac{2B^2k_BT}{\kappa^2}\delta(\vec{r_1}-\vec{r_2})^2\end{aligned} \tag{44}$$

This clearly leads to no interaction energy, since it's just a constant term and a delta function (which is zero since inclusions aren't at the same position).
We verify the same for coupling with Gaussian curvature. In this case (and also the next one) it's useful to proceed in a different way: using Wick's theorem we will reduce the original expression to a combination of various terms like.

$$S_{ab,cd}(\vec{r}_1-\vec{r}_2) = \langle\partial_a\partial_b h(\vec{r}_1)\partial_c\partial_d h(\vec{r}_2)\rangle \tag{45}$$

To evaluate this we follow Park-Lubensky recipe [11]. First we evaluate the height correlation function in real space:

$$\begin{aligned}G_{hh}(\vec{x}-\vec{y}) &= \langle h(\vec{x})h(\vec{y})\rangle_0 \\ &= \int\frac{d\vec{p}d\vec{q}}{(2\pi)^4}\langle h(\vec{p})h(\vec{q})\rangle_0 e^{i(\vec{p}\cdot\vec{x}+\vec{q}\cdot\vec{y})} \\ &= \int\frac{d\vec{p}}{(2\pi)^2}\frac{k_BTe^{i\vec{p}\cdot(\vec{x}-\vec{y})}}{\kappa p^4} = \frac{k_BT}{16\pi\kappa}r^2lnr^2\end{aligned} \tag{46}$$

Where in the last line we defined $\vec{r} = \vec{x}-\vec{y}$. Now we use this result to evaluate $S_{ab,cd}(\vec{r})$ (here we do not report the straightforward but still tedious evaluation

of derivatives) :

$$\begin{aligned}\langle\partial_a\partial_b h(\vec{r}_1)\partial_c\partial_d h(\vec{r}_2)\rangle &= \partial_a\partial_b\partial_c\partial_d G_{hh}(\vec{r}_1-\vec{r}_2)\\ &= \frac{k_BT}{4\pi\kappa r^2}\Bigg\{\delta_{ab}\delta_{cd}+\delta_{ac}\delta_{bd}+\delta_{bc}\delta_{ad}-\frac{2}{r^2}\Big[r_ar_b\delta_{cd}+r_ar_c\delta_{bd}\\ &+ r_ar_d\delta_{bc}+r_dr_c\delta_{ab}+r_br_d\delta_{ac}+r_br_c\delta_{ad}\Big]+\frac{8r_ar_br_cr_d}{r^4}\Bigg\}\end{aligned} \tag{47}$$

Where again we defined $\vec{r}=\vec{r}_1-\vec{r}_2$ and r_i is the component of this vector along axis i. The terms we'll be interested in are:

$$\begin{aligned}\left\langle\partial_x^2h(\vec{r}_1)\partial_x^2h(\vec{r}_2)\right\rangle_0 &= \frac{k_BT}{4\pi\kappa r^6}\left\{-r_x^4-6r_x^2r_y^2+3r_y^4\right\}\\ \left\langle\partial_y^2h(\vec{r}_1)\partial_y^2h(\vec{r}_2)\right\rangle_0 &= \frac{k_BT}{4\pi\kappa r^6}\left\{3r_x^4-6r_x^2r_y^2-r_y^4\right\}\\ \left\langle\partial_x\partial_yh(\vec{r}_1)\partial_x\partial_yh(\vec{r}_2)\right\rangle_0 &= \frac{k_BT}{4\pi\kappa r^6}\left\{-r_x^4+6r_x^2r_y^2-r_y^4\right\}\\ \left\langle\partial_x^2h(\vec{r}_1)\partial_y^2h(\vec{r}_2)\right\rangle_0 &= \frac{k_BT}{4\pi\kappa r^6}\left\{-r_x^4+6r_x^2r_y^2-r_y^4\right\}\\ \left\langle\partial_x^2h(\vec{r}_1)\partial_x\partial_yh(\vec{r}_2)\right\rangle_0 &= \frac{k_BT}{4\pi\kappa r^6}\left\{2r_x^3r_y-6r_xr_y^3\right\}\\ \left\langle\partial_y^2h(\vec{r}_1)\partial_x\partial_yh(\vec{r}_2)\right\rangle_0 &= \frac{k_BT}{4\pi\kappa r^6}\left\{-6r_x^3r_y+2r_xr_y^3\right\}\end{aligned} \tag{48}$$

We can now use the retrieved expressions to evaluate the interaction in case of coupling with Gaussian curvature. We'll make use of Wick theorem to expand high-order correlations, as always discarding terms not involving distance between molecules.

$$\begin{aligned}\left\langle H_{D,int}^2\right\rangle_0 &= D^2\Big\langle[\partial_x^2h(\vec{r_1})\partial_y^2h(\vec{r_1})-(\partial_x\partial_yh(\vec{r_1}))^2]\times\\ &\qquad[\partial_x^2h(\vec{r_2})\partial_y^2h(\vec{r_2})-(\partial_x\partial_yh(\vec{r_2}))^2]\Big\rangle_0\\ &= D^2\Big\{\left\langle\partial_x^2h(\vec{r}_1)\partial_x^2h(\vec{r}_2)\right\rangle_0\left\langle\partial_y^2h(\vec{r}_1)\partial_y^2h(\vec{r}_2)\right\rangle_0\\ &+\left\langle\partial_x^2h(\vec{r}_1)\partial_y^2h(\vec{r}_2)\right\rangle_0\left\langle\partial_y^2h(\vec{r}_1)\partial_x^2h(\vec{r}_2)\right\rangle_0\\ &-4\left\langle\partial_x^2h(\vec{r}_1)\partial_x\partial_yh(\vec{r}_2)\right\rangle_0\left\langle\partial_y^2h(\vec{r}_1)\partial_x\partial_yh(\vec{r}_2)\right\rangle_0\\ &+2\left(\langle\partial_x\partial_yh(\vec{r}_1)\partial_x\partial_yh(\vec{r}_2)\rangle_0\right)^2\Big\}\end{aligned} \tag{49}$$

If now we insert expressions 48 we get (again we do not report all calculations):

$$\left\langle H_{D,int}^2\right\rangle_0 = 0 \tag{50}$$

However if molecules are coupled quadratically with mean and Gaussian curvature, i.e.

$$H_{BD,int} = \sum_j B_j[\nabla^2h(\vec{r_j})]^2 + D_j[\partial_x^2h(\vec{r_j})\partial_y^2h(\vec{r_j})-(\partial_x\partial_yh(\vec{r_j}))^2] \tag{51}$$

then an interaction arises. Consider the case of identical inclusions (i.e. $B_j = B$ and $D_j = D\ \forall j$), the interaction energy reads:

$$\begin{aligned}
U &= -\frac{1}{2k_BT}\left\langle H^2_{BD,int}\right\rangle_0 \\
&= -\frac{BD}{k_BT}\left\langle [\nabla^2 h(\vec{r}_1)]^2[\partial_x^2 h(\vec{r}_2)\partial_y^2 h(\vec{r}_2) - (\partial_x\partial_y h(\vec{r}_2))^2]\right\rangle_0 \\
&= -\frac{2BD}{k_BT}\left\langle \nabla^2 h(\vec{r}_1)\partial_x^2 h(\vec{r}_2)\right\rangle_0\left\langle \nabla^2 h(\vec{r}_1)\partial_y^2 h(\vec{r}_2)\right\rangle_0 + \frac{2BD}{k_BT}\left\langle \nabla^2 h(\vec{r}_1)\partial_x\partial_y h(\vec{r}_2)\right\rangle_0^2 \\
&= -\frac{2BD}{k_BT}\left(\left\langle \partial_x^2 h(\vec{r}_1)\partial_x^2 h(\vec{r}_2)\right\rangle_0 + \left\langle \partial_y^2 h(\vec{r}_1)\partial_x^2 h(\vec{r}_2)\right\rangle_0\right)\times \\
&\left(\left\langle \partial_x^2 h(\vec{r}_1)\partial_y^2 h(\vec{r}_2)\right\rangle_0 + \left\langle \partial_y^2 h(\vec{r}_1)\partial_y^2 h(\vec{r}_2)\right\rangle_0\right) \\
&+ \frac{2BD}{k_BT}\left(\left\langle \partial_x^2 h(\vec{r}_1)\partial_x\partial_y h(\vec{r}_2)\right\rangle_0 + \left\langle \partial_y^2 h(\vec{r}_1)\partial_x\partial_y h(\vec{r}_2)\right\rangle_0\right)^2
\end{aligned} \tag{52}$$

Inserting again expressions 48 we eventually get:

$$U = \frac{BDk_BT}{\pi^2\kappa^2 r^4} \tag{53}$$

This means that if $BD < 0$, i.e. have an opposite sign, the inclusions attract each other.

3 Simulations setup

Now that has been described the sorting process under investigation and how a fluctuating membrane can mediate interactions between molecules attaching on it, we describe the discrete model used to derive numerical results. This model describes a discretized fluctuating membrane on which molecules attach, laterally diffuse and phase separate into domains. Once these domains are large enough are removed from the membrane. This scheme aims at simulating the same process described in 1.1 with the extra ingredient of membrane fluctuations and molecules locally perturbing its properties. In Fig.5 is shown a schematic representation of the process. We chose to perform simulations through a Monte Carlo scheme.

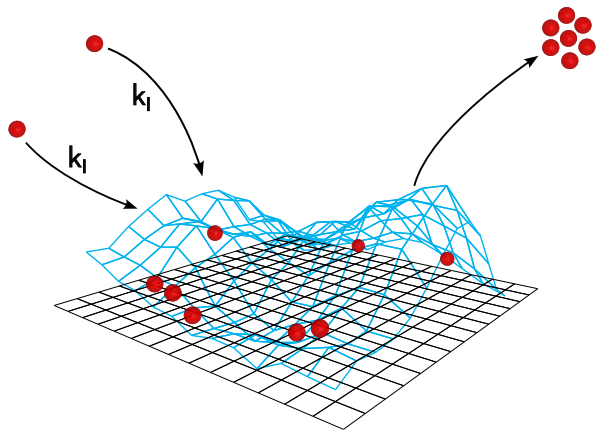

Figure 5: Schematic representation of the discrete model. Membrane is described as a fluctuating discretized surface on which particles attach at a rate k_I within empty sites. Then they laterally diffuse on it and aggregate into clusters that are extracted once big enough.

3.1 Discrete Hamiltonian

To make simulations of the above-described process the surface has to be discretized. Here we follow Weikl's recipe for surface discretization [15, 17].
The surface is described as in the Monge gauge as its height over a reference plane; the latter is discretized into a square lattice with constant a of $L \times L$ sites (we will adopt periodic boundary conditions). In the following, we will refer to the height of the surface w.r.t. the reference plane at lattice site i as $h_i = h_{x_i,y_i}$ (here with x_i and y_i we refer to the position of the i^{th} site on the lattice). It is important to recall that in order to describe consistently membrane fluctuations the lattice spacing must be comparable with the smallest deformations wavelength. For lipid bilayers of a thickness of about $4nm$ this means $a \approx 6nm$ [13, 17]. In this framework Canham–Helfrich energy of the whole unperturbed

surface can be written as:

$$H_{el} = \sum_i \frac{\kappa_0}{2} a^2 (c_{i,x} + c_{i,y})^2 \tag{54}$$

where κ_0 is the unperturbed membrane stiffness, while $c_{i,x}$ a $c_{i,y}$ are the two curvatures along x and y axis at site i. These can be evaluated as:

$$c_{i,x} = \frac{(h_{x_i+a,y_i} + h_{x_i-a,y_i} - 2h_{x_i,y_i})}{a^2} \tag{55}$$

$$c_{i,y} = \frac{(h_{x_i,y_i+a} + h_{x_i,y_i-a} - 2h_{x_i,y_i})}{a^2} \tag{56}$$

Now we perturb the membrane with molecules: we associate to each site an occupation number n_i which can be either 1 (a molecule is present) or 0 (no molecule). When a molecule is present on site i, the elastic energy associated to the site becomes:

$$\frac{K}{2} a^2 (c_{i,x}^2 + c_{i,y}^2) \tag{57}$$

We will consider $K > \kappa_0$, this means that the presence of a molecule perturbs locally the membrane making it stiffer. Notice that in this case minimizing the energy suppresses also saddle-like configurations ($c_{i,x} = -c_{i,y}$). Accounting for the presence of molecules the energy of the whole surface becomes:

$$H_{el} = \sum_i \left[(1 - n_i) \frac{\kappa_0}{2} a^2 (c_{i,x} + c_{i,y})^2 + n_i \frac{K}{2} a^2 (c_{i,x}^2 + c_{i,y}^2) \right] \tag{58}$$

The last ingredient that we add in this discrete model is a direct interaction between molecules, controlled by a parameter W. The resulting complete Hamiltonian for membrane+molecules system is:

$$H = \sum_i \left[(1 - n_i) \frac{\kappa_0}{2} a^2 (c_{i,x} + c_{i,y})^2 + n_i \frac{K}{2} a^2 (c_{i,x}^2 + c_{i,y}^2) - \frac{W}{2} \sum_{j \in \mathcal{N}_i} n_i n_j \right] \tag{59}$$

Here $\mathcal{N}_i$ is the set of nearest neighbours of site i in the lattice.

In our simulations we set $a = 1$ and $\beta = (k_B T)^{-1} = 1$, thus setting $6nm$ as our unit length and $k_B T$ as our energy unit. Moreover, we assume in all simulations that $\kappa_0 = 10 k_B T$ which is biologically realistic and ensures the system to be in a regime of small fluctuation, i.e. quasi-flat membrane configurations. This choice justifies the second-order approximation in the Monge gauge of the Helfrich Hamiltonian and the evaluation of derivatives through finite differences.

Link with phenomenological theory. We can try to transpose to the previous phenomenological description this perturbation, evaluating H_{int} corresponding to our discrete model. To this purpose, we evaluate the energy difference for a lattice site i between the case of presence and absence of a molecule:

$$\begin{aligned} H_{int} = H_{el,i}^{n_i=1} - H_{el,i}^{n_i=0} &= \frac{K}{2} a^2 (c_{i,x}^2 + c_{i,y}^2) - \frac{\kappa_0}{2} a^2 (c_{i,x} + c_{i,y})^2 \\ &= \underbrace{\frac{a^2 (K - \kappa_0)}{2}}_{\text{B}} (c_{i,x} + c_{i,y})^2 \underbrace{-a^2 K}_{\text{D}} c_{i,x} c_{i,y} \end{aligned} \tag{60}$$

From the above re-expression, we see that we are considering the last case described in the theoretical description: quadratic coupling in the curvature, both through mean and Gaussian curvature.
Thus the pair-wise interaction that will arise between two inclusions will be:

$$U \propto \frac{BD}{\kappa_0^2} \propto \frac{-(K-\kappa_0)K}{\kappa_0^2} = -\frac{K^2}{\kappa_0^2} + \frac{K}{\kappa_0} \tag{61}$$

From this we deduce that a parameter controlling this fluctuation-induced interaction is K/κ_0, which in fact is the one appearing in Weikl's phase diagram [15, 17].

3.2 Coupling membrane and molecules dynamics

In the previous studies of the sorting process membrane fluctuations weren't taken into account, now we wish to add this ingredient. Depending on the timescales involved, one can imagine two different regimes: one in which molecules are *slow* and the membrane has the time to relax to configurations influenced by the molecule disposition on the membrane itself, and an opposite regime of *fast* molecules. The characteristic time for lateral diffusion is

$$\tau_{\text{diff}} = \frac{\lambda^2}{D}$$

where λ is the size of the system.

The characteristic relaxation time for membrane dynamics, supposing that the viscosity of the cytosol is the main dissipation source, is [16, 40, 3]

$$\tau_{rel} \sim \frac{\eta\lambda^3}{\kappa}$$

where η is the viscosity of the cytosol, $\sim 5 \cdot 10^{-3}$ Pa·s [3]. The typical bending rigidity κ for example for red blood cells spans $5 - 25\ k_BT$ [27, 42]. Endosome diameters are of the order of $\lambda = 100 - 500$nm [32], while the lateral diffusivity D of proteins is in the range $1 - 10\ \mu\text{m}^2/\text{s}$ [25, 31], therefore:

$$\frac{\tau_{\text{diff}}}{\tau_{\text{rel}}} \sim \frac{\kappa}{D\eta\lambda}$$

which spans $1-10^2$, thus suggesting that the dynamics of membrane fluctuations is faster than lateral molecule diffusion. This result on characteristic timescales is in agreement with previous works[22, 24].

The performed simulations use a Monte Carlo scheme to relax the system towards its equilibrium, in this kind of simulations a comparison with realistic timescales is not feasible. Nevertheless, in the choice of the MC kinetics we have some degree of control on the speed at which membrane and molecules configurations relax towards equilibrium. The next section describes details of this kinetics and how we managed to control, at least qualitatively, the involved timescales.

3.3 Monte Carlo procedure

Given an initial configuration for both the membrane $\{h_i^0\}_{i=1}^{L^2}$ and molecules $\{n_i^0\}_{i=1}^{L^2}$ we let the system evolve toward equilibrium through Monte Carlo sampling. Our procedure for a Monte Carlo Sweep (MCS) is made up of four parts, described in the following.

1. **Membrane relaxation** Going through each lattice site i in random order, the corresponding height of the surface h_i over the reference plane is displaced in a new position chosen uniformly in an interval of measure $2l_0$ around its previous position:

   $$h_i^{(t+1)} \sim \mathcal{U}(h_i^{(t)} - l_0, h_i^{(t)} + l_0)$$

   The energy difference between the two configurations $\Delta H = H^{(t+1)} - H^{(t)}$ is evaluated locally and used to accept or reject the new configurations through the usual Metropolis criterion:

   $$p_{acc} = \min[1, \exp(-\beta \Delta H)]$$

   We choose l_0 in order to have $\sim 50\%$ acceptance of moves. This procedure is repeated N_0 times.

2. **Diffusion** Going through each lattice site i in random order if a molecule is present at that site, i.e. $n_i = 1$, one of its unoccupied neighbours, if any, is randomly chosen and the molecule is displaced at that site. Again the energy difference between the two configurations is used to evaluate acceptation probability through the Metropolis criterion.
   This kind of dynamics is called Kawasaki dynamics (from the first physicist who proposed it to study spin systems at fixed magnetization). The possibility for molecules to jump to nearest neighbours sites simulates their diffusion.

3. **Insertion**: choose in random order each site and if its free try to insert a molecule with probability k_I.

4. **Extraction**: if exists a connected component of occupied sites of size $\geq N_e$, then the molecules in the cluster are removed.

The last two kinds of moves drive the system out of its thermodynamic equilibrium, in order to mimic the distillation process. While performing a single sweep over the whole lattice for the diffusion moves, the N_0 repeated sweeps for membrane moves ensure to set the system in the above-described timescales regime. In particular we chose $N_0 = 10$, which seems to ensure membrane relaxation between two consecutive diffusion moves and is computationally feasible. The Fig.6 shows that for the values of K considered in this work setting $N_0 = 10$ should ensure membrane relaxation between consecutive diffusion moves. This procedure was implemented in Julia 1.8.3 using library *JLD* for file I/O.

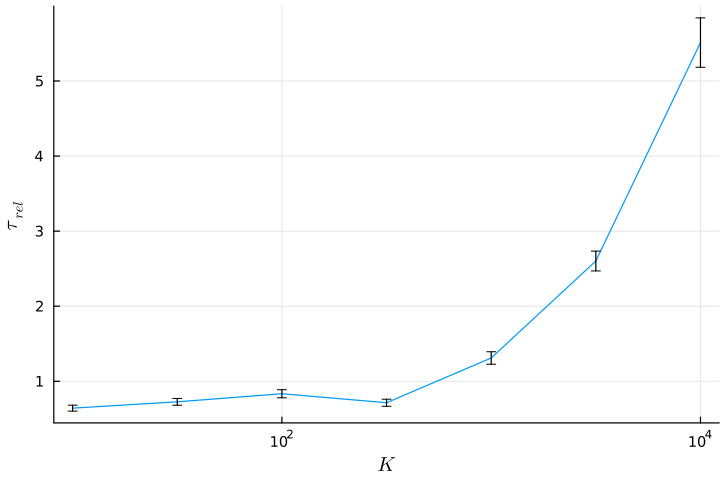

Figure 6: Relaxation time for membrane configuration between two consecutive diffusion moves. The unit for τ_{rel} is a complete sweep along all lattice sites while K is measured in k_BT units.

4 Numerical results

4.1 Entropy-driven phase separation

In this section are reported some preliminary results describing the phase separation of particles due to only entropy-driven interactions. We fix an initial concentration of molecules $\rho = 0.3$ and let the system relax towards equilibrium, in absence of direct interactions and without forcing molecules in or out of the system. This means we set $k_I = 0$, $N_e > \rho L^2$ and $W = 0$, in order to set our simulations in a context similar to that investigated by Weikl [17]. To tell the onset of phase separation we measure the observable

$$f_{nn} = \frac{\sum_{\langle ij \rangle} n_i n_j}{\sum_{\langle ij \rangle}} \tag{62}$$

where the summation runs over nearest neighbour pairs. This quantity is the fraction of edges connecting occupied sites. The higher is the value of f_{nn}, the more molecules are aggregating into domains. Running these simulations for various K/κ_0 values we get that above a threshold value of molecules rigidity, these phase separate even in absence of direct interaction (see Fig.7). This is in agreement with Weikl's results.

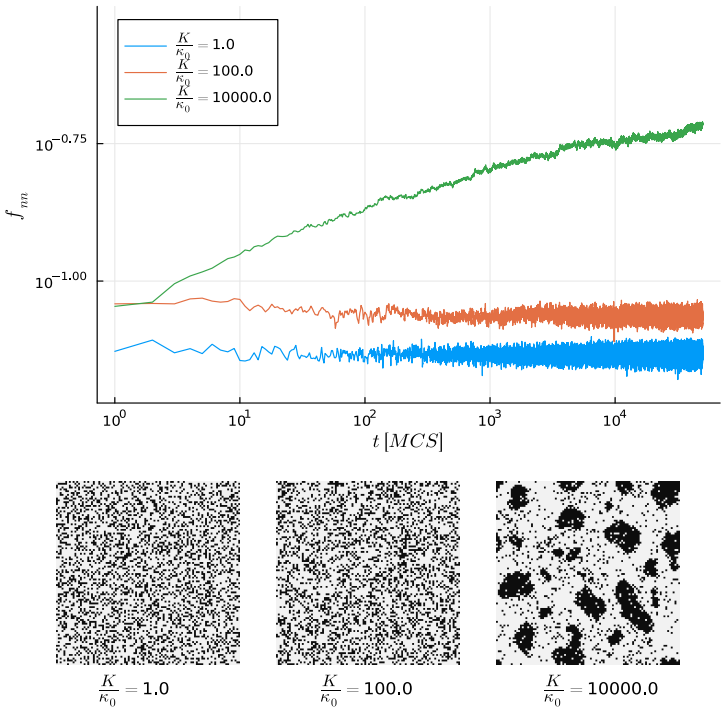

Figure 7: f_{nn} as a function of time (in MCS unit) for various values of molecules rigidity K/κ_0. Snapshots are taken after 10^4 MCS.

4.2 Reduced diffusivity

Another relevant effect of molecules rigidity is to reduce their lateral diffusivity. This effect has been described in previous analytical and numerical works [29, 24].

To check for the presence of this effect in our model we measured the ratio of acceptance of diffusion moves. If we consider as time unit an MCS this ratio corresponds to the microscopic diffusion rate k_D. In particular, we measured this ratio for molecules lacking occupied neighbours in order to consider only "free" molecules.

Measuring this quantity for different values of rigidity K imposed by molecules we get the results in Fig.8 Increasing the value of K, decreases the probability

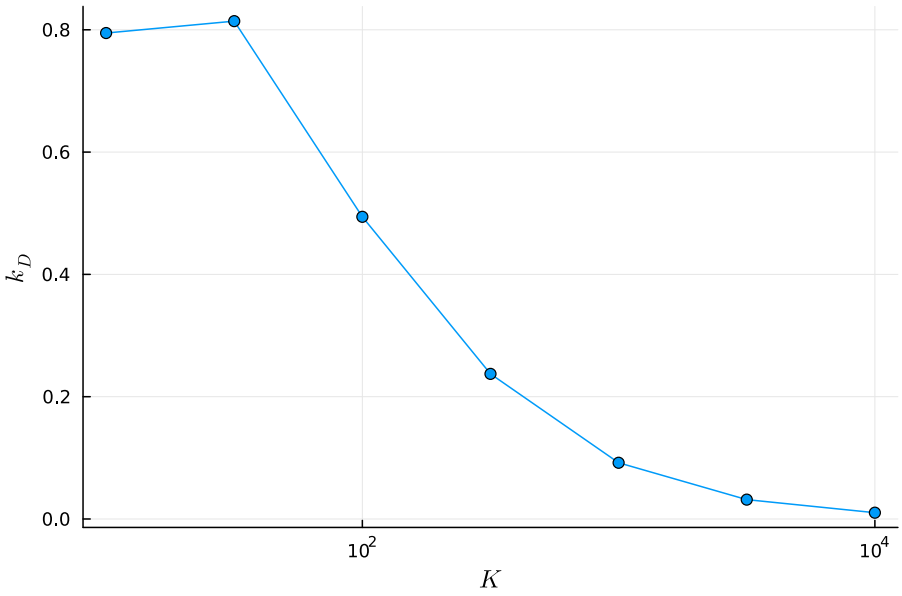

Figure 8: Microscopic diffusion rate k_D for free molecules, as a function of K. In these simulations $W = 0$, molecule density is fixed, $\rho = 0.1$.

to accept molecule jumps. We didn't see any particular dependence on W and k_I.

4.3 Out of equilibrium stationary state

Now we wish to evaluate the effect of the fluctuation-induced interactions on the distillation process taking place on the lattice, to do this we need to drive the system out of the equilibrium and then evaluate a stationary state. Thus now we consider finite k_I and W values and $N_e = 100$. We run simulations starting from initially flat and empty membrane configuration, i.e. setting $h_i = 0$ and $n_i = 0$ $\forall i$. As simulations go on they reach a stationary state, where molecules density fluctuates around a mean value. Discarding initial measures, since they're not relevant for the stationary state, and block averaging with consistent block sizes (see App.A for details) we get independent measures of density from which we can compute the average value at stationary state $\bar{\rho}$. Using this procedure for simulations where $k_I = 10^{-6}$, for various values of W and K/κ_0 we get results in Fig.9.

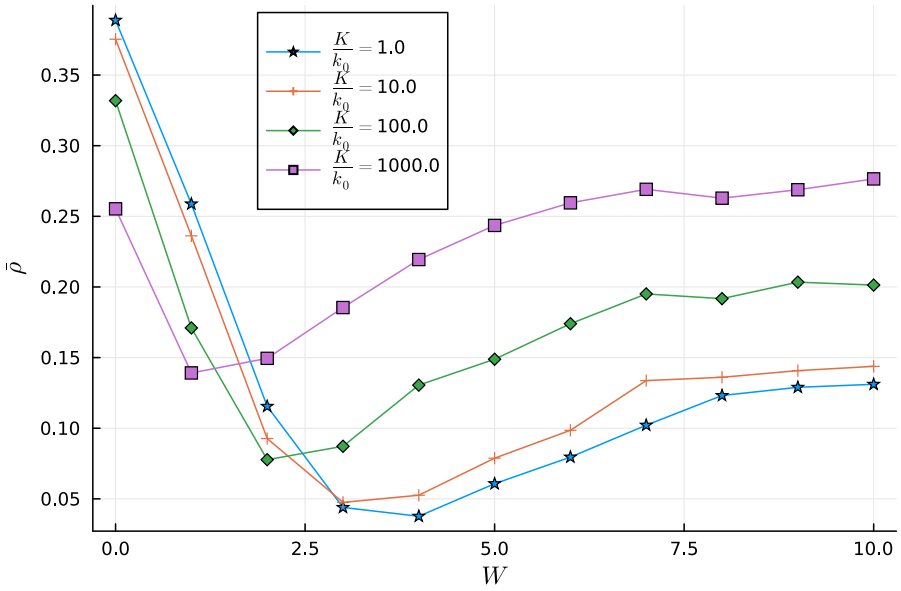

Figure 9: Average density of molecules at stationary state $\bar{\rho}$ as a function of W for various values of K/κ_0. For all points relative error is below 5% thus errorbars are not shown.

From these plots, we can notice that at small W values the rigidity imposed by molecules lowers the density at the stationary state, i.e. makes the process more efficient. Then this behaviour becomes the opposite (approximately for $W \geq 2$) and the process becomes more efficient with small K values. Moreover, the minimum of $\bar{\rho}$, corresponding to the optimum in the distillation gets shifted towards smaller values of W with increasing K.

However, if our aim is telling the effect of molecules rigidity on the efficiency of the distillation process we need to recall the effect described in 4.2. Since the different curves of Fig.9 correspond to different K values of molecules rigidity, then also their microscopic diffusivity rates k_D are different. This diffusivity rate along with the flux ϕ of incoming molecules plays an important role in this

process, as investigated theoretically and numerically in [39, 41] and reported in Fig.2. In the following we describe a procedure we adopted to produce curves where the ratio ϕ/k_D is fixed.

4.4 Estmating $\bar{\rho}$ for a given ϕ/k_D ratio

In order to compare consistently $\bar{\rho} - W$ curves for different K values we need them to be produced at the same ϕ/k_D ratio. Unfortunately, this isn't something we can control consistently in simulations. Here we propose a possible solution to the problem. For each value of the parameters W and K we perform simulations at different k_I insertion rate values. In these simulations, along with molecule density, we measure both the effective flux of molecules towards the membrane $\phi = k_I(1-\rho)$ and the effective molecule diffusion rate k_D. This way, for given values of W and K we can plot measured values of ρ as a function of ϕ/k_D. In Fig.10 is reported a scatter plot of this kind. Each color of the various cloud of points represents data obtained from a simulation with different k_I.

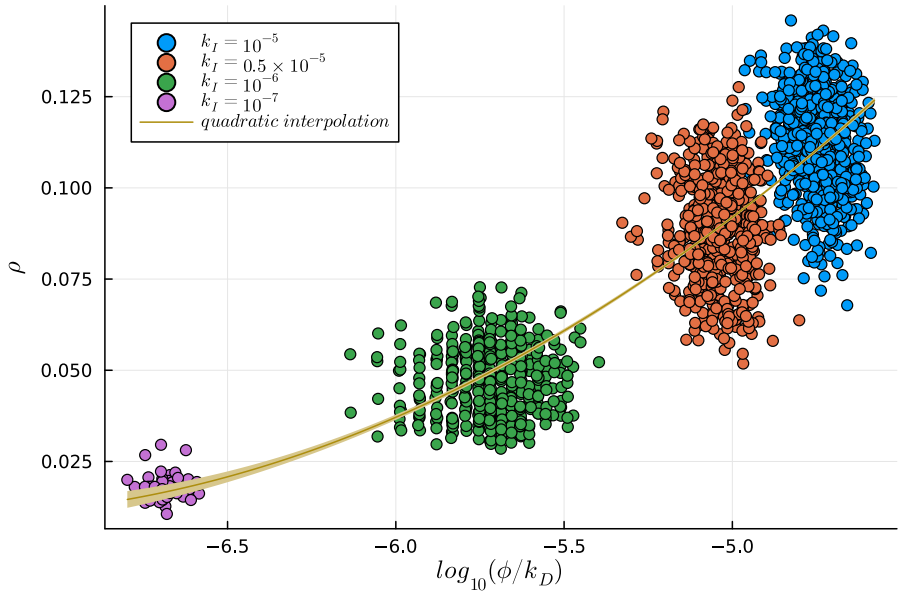

Figure 10: Various values of ρ, obtained at different ϕ/k_D values. Each colored cloud corresponds to data obtained for a different k_I insertion rate. The fitted quadratic curve, with error ribbon, is plotted too.

We fit these values with a quadratic curve and use it to get the estimated mean response at the desired ϕ/k_D value. To estimate an error on this mean response we use procedure described in App.B.

Using this procedure for each W and K value we're interested in we're able to estimate $\bar{\rho} - W$ curves for different K values that are consistently comparable since they are obtained for the same value of ϕ/k_D. The result of this operation is shown in Fig.11. From the plot the effect of fluctuation-induced forces seems to be divided into two regimes. For $W < W^* \approx 3$, a low affinity region the effect of these entropic interactions is to make the sorting more efficient. For

$W > W^*$ the effect is the opposite, with the rigidity of molecules impairing the sorting process.

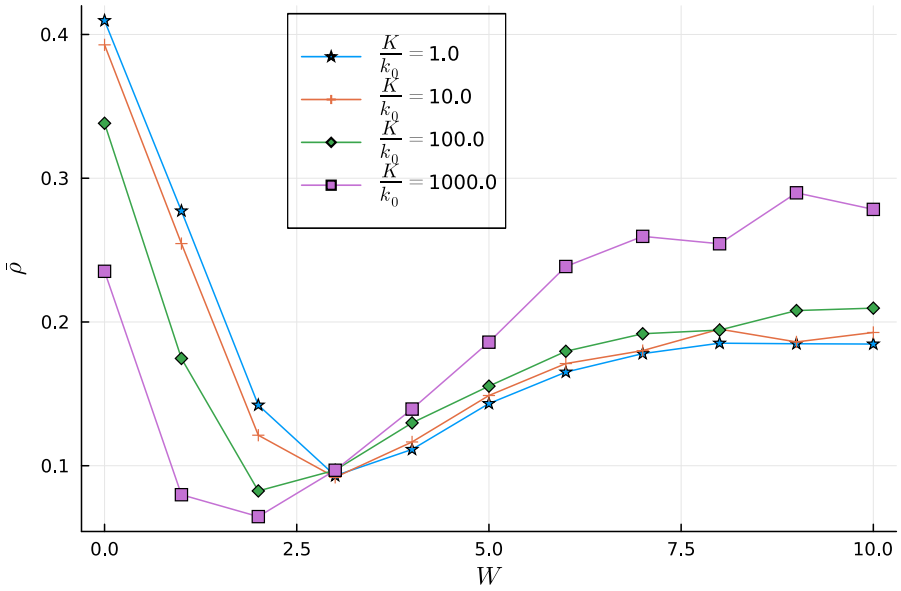

Figure 11: Estimated average density of molecules at stationary state $\bar{\rho}$ as a function of W for various values of K. These values are obtained for $\phi/k_D = 10^{-5}$. For all points relative error is below 5% thus errorbars are not shown.

4.5 Mapping with a biological problem

The result above suggests that these membrane-mediated forces play indeed a role in the sorting process, however, the range of parameters explored is not necessarily realistic. It is possible to map a biological problem on the model making some considerations. The value of membrane stiffness considered $\kappa_0 = 10k_BT$ is biologically realistic (see Sec.3.2). If we consider the case of Clathrin-coated vesicles (CCVs) the bending rigidity of the lipid membrane covered by a clathrin molecule is $K_{cla}^{physio} \sim 300k_BT$ [21]. At first sight, one may think that the ratio $K_{cla}^{physio}/\kappa_0 \sim 30$ isn't large enough to produce a relevant effect. However, we're not taking into account the area occupied by clathrin on the membrane: in our model molecules occupy a single lattice site of area $a^2 \sim 36nm^2$, but clathrin covers a much larger area of the membrane. Previous analytical [7, 8, 11] and numerical [17] studies have shown that the fluctuation-induced forces we're investigating have a dependence on the area of inclusions. Intuitively this dependence comes from the fact that a large inclusion with its stiffness suppresses more fluctuation modes for the membrane leading to stronger entropy-driven interactions. In particular, Weikl in his work [17] found that the critical bending stiffness K^* at which phase separation occurs due to only fluctuation-induced forces scales with the area occupied by inclusions Q (in lattice sites unit) as $K^* \sim Q^{-0.7}$. This means that if we have phase separation at a given K_1^* value for inclusions occupying a single lattice site,

we would have phase separation anyway with $K_Q^* < K_1^*$ for larger inclusions of size Q. Thus we have that the value of the bending rigidity of clathrin we should consider when mapping it to our simulations is $K_{cla}^{sim} \sim K_{cla}^{physio} Q^{0.7}$. Each clathrin molecule covers an area of approximately $(20nm)^2$, hence for this case $Q \approx 10$. The obtained ratio $K_{cla}^{sim}/\kappa_0 \sim 150$ is large enough to produce appreciable effects due to the membrane-mediated interactions, especially in the weak direct interactions region.

5 Quasi-spherical vesicles and dynamically triangulated Monte Carlo

In order to describe more realistic membranes another discrete model was implemented. It is based on previous works on equilibrium simulations for quasi-spherical vesicles [6, 28], where a closed surface is represented as a polyhedron made up of triangles (generally in computer graphics an object of this kind is named *mesh*). An important feature of these simulations is that in order to ensure a correct relaxation of the vesicle shape the triangulation is dynamical, i.e. changes during simulations. This effect aims also at mimicking the lateral diffusivity of the lipids of which membrane is made up. That's why previous works of this kind often refer to *fluid* membranes.

During thesis just Monte Carlo method for membrane relaxation was implemented, nevertheless as a future work would be reasonable to check for the presence of these entropic forces on this quasi-spherical membrane and investigate their role on the sorting process.

5.1 Discrete Hamiltonian

On a discrete surface there many ways to define mean curvature depending on the surface regularization. For a triangulated quasi-spherical surface usually Helfrich Hamiltonian takes the form [4, 6]

$$H = \lambda \sum_{<\alpha\beta>} [1 - \mathbf{n}_\alpha \cdot \mathbf{n}_\beta] \tag{63}$$

where λ is a parameter playing the role of the bending rigidity and the summation is over all edges. The unit vectors $\mathbf{n}_\alpha$ and $\mathbf{n}_\beta$ are normal respectively to the triangles α and β sharing the edge $< \alpha\beta >$. However it has been shown that the relation between λ and κ depends on the shape of the surface [10]. Moreover since the goal of this thesis is to study effects of local perturbation of paramaeters (bending rigidity, gaussian bending rigidity, spontaneous curvature ecc..) originated by molecules attaching to the surface, a clear local definition of curvatures is needed.

In this work will be used a different discretization, first proposed in 2010 by Ramakrishnan *et al.* [28], based on a previous work by Taubin [9]. The idea is to define shape operator along edges of the mesh and then project the properly to the vertices they connect to build up a vertex shape operator.

Some basic definitions A given configuration of the membrane is described by the tuple

$$\eta = (\{\vec{X}\}, \{\mathcal{T}\}) = (\{\vec{x}_i\}_{i=1}^{N_v}, \{f_i\}_{i=1}^{N_f})$$

made up of vertex positions and face triplets (i.e. triplet of indices of vertices constituting each face, ordered in counter-clockwise order). Given this we can define the set of faces and edges neighboring a vertex v: respectively $\{f\}_v$ and $\{e\}_v$.
To each face f we associate an area A_f and a normal $\hat{N}_f$. While to each edge we associate:

1. A vector: along an edge e connecting vertex v to its neighbour i, $\vec{r}_e = \vec{x}_i - \vec{x_v}$. The unit vector along its direction will be denoted as $\hat{r}_e = \frac{\vec{r}_e}{|\vec{r}_e|}$

2. A normal:
$$\hat{N}_e = \frac{\hat{N}_{e,1} + \hat{N}_{e,2}}{|\hat{N}_{e,1} + \hat{N}_{e,2}|}$$
Where $\hat{N}_{e,1}$ and $\hat{N}_{e,2}$ are the normals to faces adjacent to edge e.

3. A binormal: $\hat{b}_e = \hat{r}_e \times \hat{N}_e$

Also each vertex v is associated with a normal $\hat{N}_v$, but defining it is ambiguous. In a continuous picture, the normal to a vertex is defined as:

$$\mathcal{N}_v = \int_{\mathcal{C}} \mathcal{N}_S(\mathcal{C}) d\mathcal{C} \tag{64}$$

where $\mathcal{N}_S(\mathcal{C})$ is the normal to the surface S containing the vertex, evaluated along a closed contour $\mathcal{C}$ that encloses the vertex. If the surface is discretized into triangles, the normal to the surface changes only on the interface between faces, so we have to introduce proper weighting to account for this [33].
In the triangulated surface normal to vertex v reads:

$$\hat{N}_v = \frac{\sum_{\{f\}_v} \hat{N}_f A_f}{|\sum_{\{f\}_v} \hat{N}_f A_f|} \tag{65}$$

The schematic representation in Fig.12 may be useful for a better understanding.

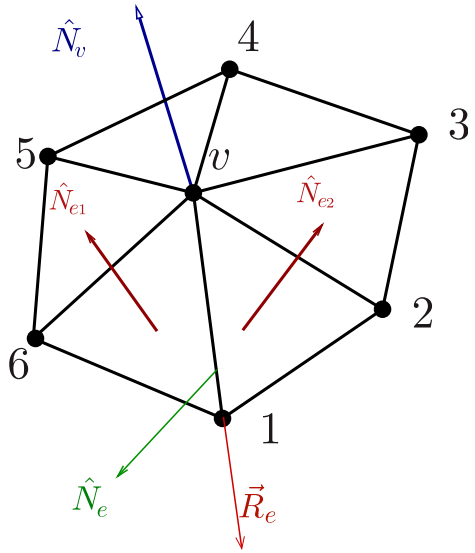

Figure 12: Figure from [30], shows some of the quantities defined above.

Edge shape operator The idea is to first evaluate edge shape operators S_e and then project them onto vertices with proper weighting. The edge shape operator is constructed as

$$S_e = h(e)[b_e \otimes b_e] \tag{66}$$

Here $\otimes$ denotes the tensor product between two vectors, while $h(e)$ represents a curvature, defined as the gradient of the area vector of the two triangles sharing e (evaluated at any point p along the edge).

$$h(e) = \nabla_p(area) \approx 2|\vec{r}_e|cos\left(\frac{\phi(e)}{2}\right) \tag{67}$$

Here $\phi(e)$ is the (signed) dihedral angle between the faces sharing e (see figure 13).

$$\phi(e) = sgn[(\hat{N}_{e,1} \times \hat{N}_{e,2}) \cdot \vec{r}_e] arccos(\hat{N}_{e,1} \cdot \hat{N}_{e,2}) + \pi \tag{68}$$

Actually $h(e)$ has the dimension of length but should be divided by an area to be a curvature, this operation is done afterwards.

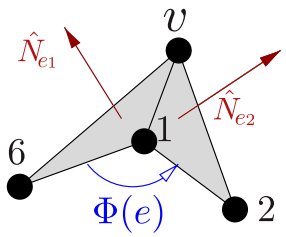

Figure 13: Figure from [30], shows the geometrical meaning of the dihedral angle $\phi(e)$

Vertex shape operator and Hamiltonian The shape operator at vertex v is defined as the superposition of all edge shape operators S_e around it:

$$S_v = \frac{1}{A_v} \sum_{\{e\}_v} W(e,v) P_v^\dagger S_e P_v \tag{69}$$

Here $A_v = \frac{1}{3}\sum_{\{f\}_v} A_f$ can be seen as the average area around vertex v. $W(e,v) = \hat{N}_v \cdot \hat{N}_e$ is a factor that weights the various contribution coming by various edge shape operators S_e. This way we've introduced a local definition for the shape operator on vertices and for the infinitesimal area element, with a much clearer interpretation than Eq.63. The former has as eigenvalues the principal curvatures $c_1(v)$ and $c_2(v)$ at vertex v; the third eigenvalue is zero since it's referred to the direction normal to the vertex $\hat{N}_v$. Using S_v we can evaluate both mean and Gaussian curvatures, respectively as the trace and the determinant of this matrix. It's important to remember that these two quantities are independent on the chosen basis. For this surface discretization [28, 30], the energy reads:

$$\mathcal{H} = \frac{\kappa}{2}\int (2M)^2 dS \longrightarrow H = \frac{\kappa}{2}\sum_v^{N_v}\left[\frac{c_1(v)+c_2(v)}{2}\right]^2 A_v \tag{70}$$

5.2 Sphere generation

Triangulated sphere generation starts from an icosahedron with $N_v = 12$ vertices, $N_f = 20$ faces and $N_e = 30$ edges. The icosahedron is chosen since its topology is spherical (i.e. $\chi_E = N_v - N_e + N_f = 2$). In order to increase the number of faces composing the mesh loop subdivision algorithm is used. In practice each triangular face is divided into four triangles adding a vertex at the midpoint of each edge, Fig.14 may be useful for understanding. The newly created vertices are then projected onto the unit sphere. All the vertices' positions are then normalized in order to have edges of the desired length. These last steps actually change the length of newly created edges so the new faces are no more equilateral triangles, however even after reiterating the procedure many times the introduced discrepancy in edge length is compatible with the constraint on edges to have a self-avoiding membrane (see next subsection for details).

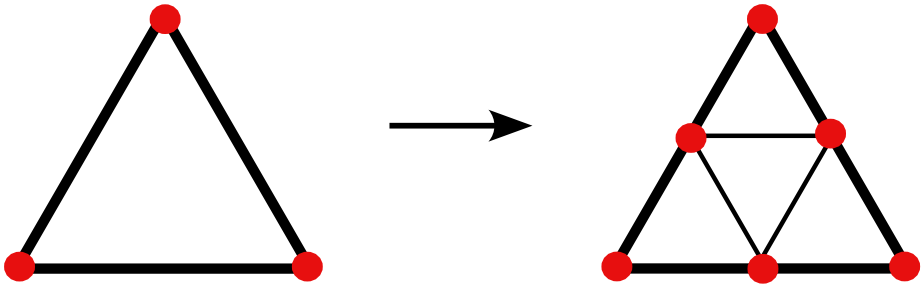

Figure 14: Loop subdivision algorithm: on the left a face before subdivision, on the right the same face after an iteration

Using this subdivision leads to $N_v' = N_v + N_e$, $N_e' = 2N_e + 3N_f$ and $N_f' = 4N_f$. Now since $\chi_E' = N_v' - N_e' + N_f' = 2$ we can say that this transformation doesn't change the topology of the mesh (it's still spherical). Performing n iterations of this algorithm result in

$$\begin{aligned} N_v &= 10(4^n) + 2 \\ N_e &= 30(4^n) \\ N_f &= 20(4^n) \end{aligned} \tag{71}$$

In the performed simulations we usually repeated this subdivision algorithm for $n = 3$ iterations, resulting in a quasi-spherical mesh composed of $N_v = 642$ vertices. This value is chosen as a balance to make simulations computationally feasible and to avoid finite-size effects.

5.3 Hard spheres potential and self-avoiding membrane

In order to simulate a realistic membrane we ant our surface to be self-avoiding, i.e. we want to ensure that vertices do not intersect faces. In these simulations this is usually done through two simple constraints [6]:

1. **Hard-sphere potential:** this consists in a potential that avoids intersection of spheres centered on vertices. Mathematically this is defined

as

$$\sum_{i,j} V_{HS}(|\mathbf{r_i} - \mathbf{r_j}|)$$

where $V_{HS}(d) = 0$ if $d > 2r$ and $V_{HS}(d) = +\infty$ if $d \leq 2r$, summation is over vertices of the surface and $\mathbf{r_i}$ denotes the position of vertex i. In our simulations we chose $r = 1/2$, this fixes as our unit length the diameter of spheres centered in vertices.

2. **Maximum tether length:** to avoid also that these spheres intersect triangular faces a maximum edge length l_{max} is constrained. Mathematically this is defined as

$$\sum_{<i,j>} V_{TL}(|\mathbf{r_i} - \mathbf{r_j}|)$$

where $V_{TL}(d) = 0$ if $d < l_{max}$ and $V_{TL}(d) = +\infty$ if $d \geq l_{max}$. In this case, the sum is over $< i, j >$ neighbour vertices.

Depending on the value of l_{max} we can have "weak" or "strong" self-avoidance [5], in this work we adopted $l_{max} = \sqrt{3}$, a condition that avoids the possibility for the center of an incident sphere to cross the triangular face. The limit case for this condition is shown in Fig.15; the face is an equilateral triangle, with incident vertex P equidistant from face vertices A, B, C, lying on the face itself and incident sphere tangent to the others. In this case, evaluating the side of the triangle we get $l = 2\overline{CH} = 2cos(30)\overline{PC} = 2 \cdot \frac{\sqrt{3}}{2} \cdot 2r$.

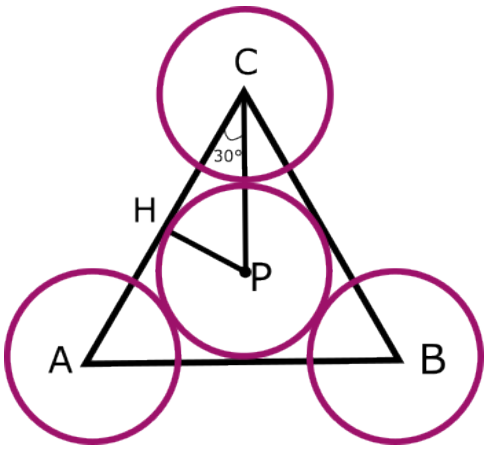

Figure 15: Limit case of an incident sphere lying on the surface of the triangle and tangent to spheres on the vertices

Note that these newly introduced potentials aren't actually taken into account during the evaluation of energy. Starting from a configuration in which the two conditions are satisfied it's simply checked at each MC step that the new membrane configuration doesn't violate these constraints. It is worth noticing that these next-to-nearest neighbour excluded volume effects induce a positive bending rigidity on the membrane [19].

5.4 Membrane evolution: vertices and links

In order to relax membrane configuration towards its thermodynamic equilibrium we perform simulations using a Monte Carlo scheme. We consider two main membrane evolution mechanisms: vertices displacements and link flips.

5.4.1 Vertex move

At each sweep, a predefined number of vertices is randomly chosen and an MC move is attempted on them through the Metropolis scheme. The vertex is moved in space within a cubic box around its original position of size l_0. Then the energy change ΔH of the membrane is evaluated and used to compute the probability of accepting the move:

$$p_{acc} = min[1, exp(-\beta \Delta H)]$$

In our simulations, without loss of generality, we set $\beta = \frac{1}{k_b T} = 1$. The value of l_0 can be tuned in order to get $\sim 50\%$ accepted moves.

5.4.2 Link-flips

This second kind of membrane evolution mechanism aims at changing the surface triangulation, that's why this class of Monte Carlo algorithm is called dynamically triangulated. At each sweep, a predefined number of edges is randomly chosen, this edge is common to two triangular faces f_1 and f_2. The idea is to try to flip the common edge and use it to connect previously unconnected edges, evaluate the energy difference between the two configurations and use it to accept or reject move through Metropolis criterion. Using a more formal description we redefine faces as:

$$f_1 = \{i, j, k\}\ f_2 = \{j, i, l\} \longrightarrow f_1' = \{k, l, j\}\ f_2' = \{l, k, i\} \tag{72}$$

The scheme in Fig.16 may be helpful to understand this procedure.

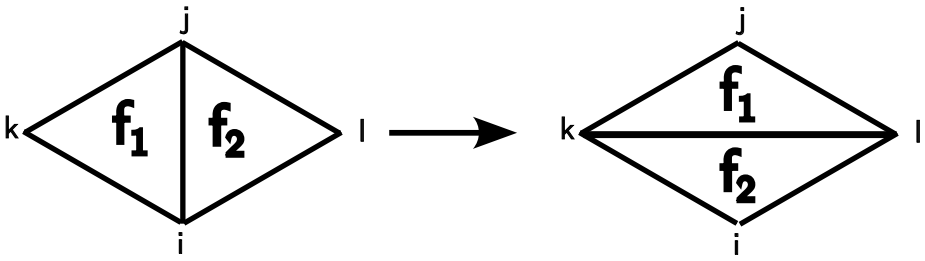

Figure 16: Schematic representation of the link-flip move.

The number of link-flips attempted at each Monte Carlo sweep can be related to the membrane viscosity, i.e. to the lateral diffusivity of lipids composing the membrane [20, 23, 37]. We decided to define a Monte Carlo Sweep (MCS) as:

1. N_v attempts to move a randomly chosen vertex.
2. N_e attempts to flip a randomly chosen edge.

5.5 Computational efficiency

The model was implemented by scratch in Julia 1.8.3 using libraries:

1. JLD: for file I/O
2. Meshes, MeshViz and GLMakie/CairoMakie backend: for mesh visualization
3. StaticArrays: to define statically sized arrays

This simulation procedure is much more computationally expensive than the one described in Sec.3. Indeed the hard-spheres constraints, the dynamical triangulation (and all related aspects like keeping the counter-clockwise order of vertices in a face) and the algebraic computations needed to evaluate principal curvatures at vertices are algorithmically expensive. In this section are briefly described some technical details about implementation that were needed to achieve computationally feasible simulations.

Hard-sphere potential Whenever is proposed a new membrane configuration, it must be checked that it satisfies the hard-sphere potential. If during a MCS a number $\mathcal{O}(N_v)$ of vertices are displaced, then if we have to evaluate for each of them the distance with all other vertices the algorithm would be $\mathcal{O}(N_v^2)$. However, for the majority of vertices, the hard-sphere potential will be satisfied since are far from the displaced one and won't overlap. To fasten this procedure the simulation space, assumed to be cubic, is divided into $L \times L \times L$ equally sized cells. By choosing the appropriate size for cells and keeping track with a list of the vertices lying in each cell it's possible to reduce the computation of inter-vertices distances to the cell containing the displaced vertex and the neighbouring ones. This way we retrieve a $\mathcal{O}(N_v)$ complexity for this procedure.

Local energy evaluation Whenever a new configuration is proposed, to accept or reject it through Metropolis criterion the energy difference has to be evaluated. Is clear that evaluation of energy on the whole surface is useless: for most vertices the energy contribution will be the same, since changes in configurations are local. Thus after a vertex displacement or a link-flip energy is evaluated locally, only on vertices involved by the change in configuration.

Profiling and Julia performance enhancements In order to address computationally expensive operations profiling procedure was adopted, using the library *ProfileView*. Another useful tool was Julia's built-in macro *@code_warntype*, which highlights type instabilities. Indeed a huge difference in Julia is made by defining type clear and type stable variables. This means respectively to use concrete types and making sure the type of a variable doesn't change during runtime. Moreover since in Julia by default arrays and vectors are defined dynamically the library *StaticArrays* was used to implement static arrays. These are mutable arrays but of a predefined length, thus the time needed to make operations on them is smaller.

Using the above-described performance improvements the implemented code is proven to be $\mathcal{O}(N_v)$ as can be seen in Fig.17

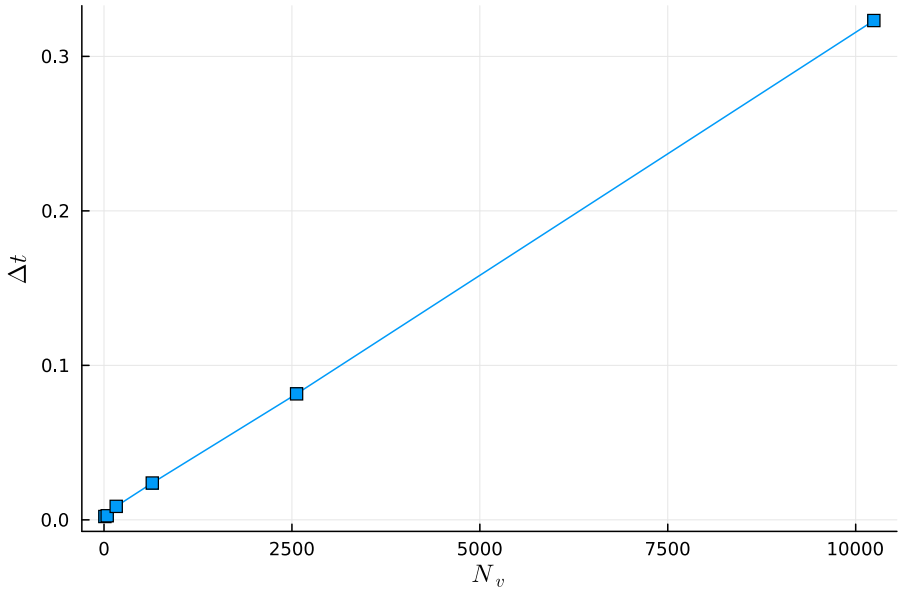

Figure 17: Average time performance for an MCS (N_v vertex steps, $3N_v$ edge flips) for various values of N_v. Δt is expressed in seconds.

5.6 Numerical results

In order to check for the correct implementation of this simulation procedure it is reasonable to reproduce the crossover between branched polymer configurations and quasi-spherical ones. This crossover is a known behaviour for closed vesicles and [19] and was also found numerically by Gompper and Kroll in their work [6] using energy discretization like Eq.63. We found this crossover as can be seen from the equilibrium configurations in Fig.18 with crossover point around $\kappa^* \approx 1$. At sufficiently low κ values the entropy term dominates the free energy and thus the vesicle rearranges itself with the only constraints given by self-avoidance. At large κ values instead the elastic energy term is dominant suppressing large fluctuations and resulting in quasi-spherical configurations.

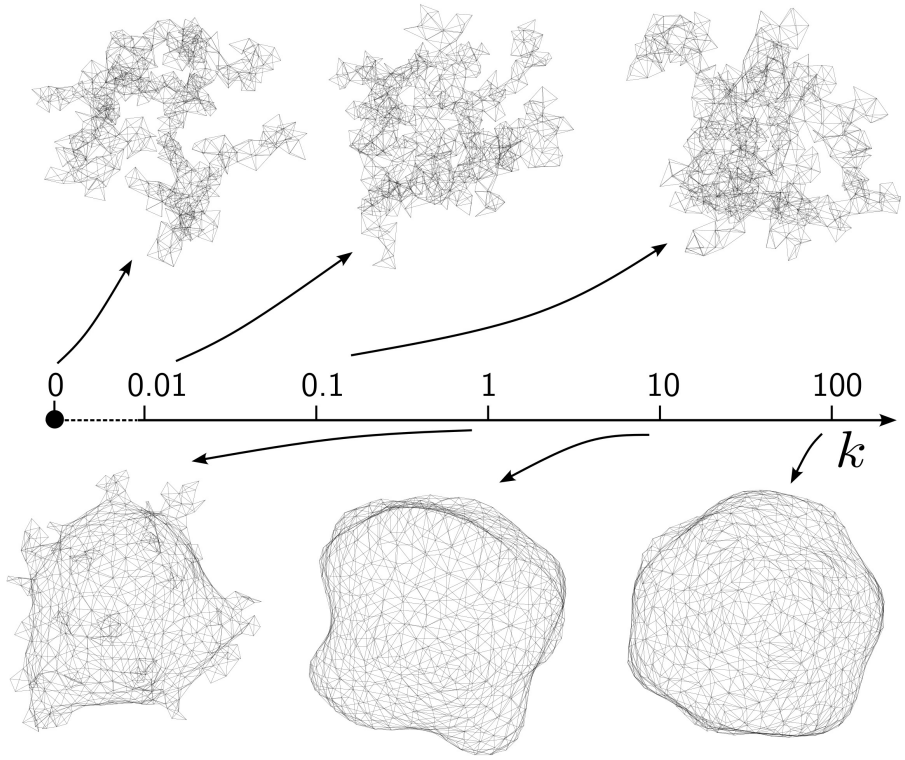

Figure 18: Crossover between branched polymer configurations and quasi-spherical ones. The shown configurations are equilibrium ones, obtained for the various κ values on the axis; the energy unit is again k_BT.

6 Conclusions

In this thesis, I investigated the role of fluctuation-induced interactions in the process of molecular sorting taking place on lipid membranes. In the first chapter, I introduced the sorting process and the phenomenological theory recently proposed to understand this phenomenon. In the second chapter, I introduced the analytical framework in which lipid membranes are usually studied. Furthermore I reproduced systematically the results from the literature for evaluating fluctuation-induced forces with a perturbative field-theoretic approach. Then after introducing the discrete model and the simulation procedure, I discussed the numerical results. The results show that these fluctuation-induced interactions can lead to entropy-driven phase separation. Moreover, in the context of the sorting process, the interplay between these membrane-mediated interactions and direct interactions is divided into two regimes. In the weak direct interactions regime these Casimir-like forces enhance the sorting process making it more efficient, while in the strong direct interactions regime these forces impair the sorting, decreasing its efficiency. A mapping with biologically realistic parameters has been estimated, suggesting that these forces may be biologically relevant, especially for the sorting of molecules with low affinity.

In the last part of the thesis I introduced another surface discretization technique and the Monte Carlo procedure used to evolve its shape towards equilibrium. In this case, the surface is closed, quasi-spherical and dynamically triangulated so may be a more realistic choice for simulating the equilibrium configurations of lipid bilayers.

This work highlights many fascinating aspects of the sorting process that can still be investigated, in particular the ones related to the variety of phenomena mediated by lipid membranes.

A Integrated autocorrelation time

To make reliable measures of observables it's important to know what the autocorrelation time is. The idea is that each MC sweep just makes small changes in the configuration of the membrane, thus to be sure to sample effectively uncorrelated measures is important to wait a certain amount of sweeps (i.e. $\Delta t \sim \tau_{int}$) between two consecutive measures to ensure that the values are uncorrelated. It's important to remember that the value of τ_{int} depends on the observable measured, on the system size and other parameters. However the procedure for its evaluation is always the same, here we report the same formulation as in [36].
Supposing to have already discarded the first part of the time serie (for thermalization/reaching of stationary state) and that our system is in a stationary state (thus statistically invariant under time shifts), the autocorrelation function of an observable O is:

$$\rho_{OO}(t) = \frac{\langle O_{t_0} O_{t_0+t} \rangle - \langle O \rangle^2}{\langle O^2 \rangle - \langle O \rangle^2} \tag{73}$$

The variance on individual measurements is

$$\sigma^2_{O_i} = \left\langle O_i^2 \right\rangle - \langle O_i \rangle^2 \tag{74}$$

If individual measurements were independent then the variance of the average value $\bar{O}$ would be $\frac{\sigma^2_{O_i}}{n}$, where n is the number of individual measurements. In general we have:

$$\begin{aligned}
\sigma^2_{\bar{O}} &= \left\langle \bar{O}^2 \right\rangle - \left\langle \bar{O} \right\rangle^2 \\
&= \frac{1}{n^2} \sum_{i,j=1}^{n} \langle O_i O_j \rangle \frac{1}{n^2} \sum_{i,j=1}^{n} \langle O_i \rangle \langle O_j \rangle \\
&= \frac{1}{n^2} \sum_{i=1}^{n} (\langle O_i^2 \rangle - \langle O_i \rangle^2) + 2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} (\langle O_i O_j \rangle - \langle O_i \rangle \langle O_j \rangle) \\
&= \frac{1}{n} \left[\sigma^2_{O_i} + 2 \sum_{t=1}^{n} (\langle O_1 O_{1+t} \rangle - \langle O_1 \rangle \langle O_{1+t} \rangle) \left(1 - \frac{t}{n}\right) \right] \\
&= \frac{\sigma^2_{O_i}}{n} \left[1 + 2 \sum_{t=1}^{n} \rho_{OO}(t) \left(1 - \frac{t}{n}\right) \right] \\
&= \frac{\sigma^2_{O_i}}{n} 2\tau_{int}(O)
\end{aligned} \tag{75}$$

Thus the number of effectively independent measurements is $n_{eff} = n/(2\tau_{int})$. To be sure to sample independent measures, we will block average our time serie in blocks of size $>> \tau_{int}$

B Estimated mean response error

To estimate $\bar{\rho}$ for a given ϕ/k_D ratio, we use a least square method to fit data with a quadratic model through Least Square Regression, i.e. we find

$$\theta^* = \arg\min_{\theta} \left\{ \sum_i [m(x_i;\theta) - y_i]^2 \right\}. \tag{76}$$

Here x_i represent $log_{10}(\phi_i/k_{Di})$ and $y_i = \rho_i$, while $m(x,\theta) = \theta_1 + \theta_2 x + \theta_3 x^2$ is the quadratic model we're trying to fit. We call residuals the quantities $r_i = m(x_i;\theta*) - y_i$. Once the above minimization has been done we're left with the model which best describes our data, and we can use it to estimate new mean response $\hat{y_{new}} = m(x_{new},\theta^*)$ at x_{new}. But how can we estimate the error on this prediction?

First of all is important to understand what error we're interested in calculating: confidence interval on a new prediction (also called prediction interval) or confidence interval on the estimated mean response (also called simply confidence interval). Basically we can make an analogy between these two and respectively variance of an observable and variance of the mean of an observable. Prediction interval indeed is the interval in which, assuming our data is well approximated by $m(x,\theta^*)$ a new observation will fall with a certain significance level. To evaluate this information about the distribution of residuals must be retrieved, since residuals give a good estimate of data variance, assuming the fitted model is correct. Usually this is done through a bootstrap procedure that can be improved through various techniques like *leave-one-out* and *.632+ rule* that aim at reducing overfitting.

However what we're really interested in is the confidence interval of the estimated mean response. This interval is smaller than prediction interval and represents the interval in which the mean of a serie of predicted new responses will fall with a certain significance level. In our case it can be evaluated expanding the error we get on fitted parameters θ. The variance of a mean new response is given by:

$$\begin{aligned} Var(\hat{y}_p) &= Var(m(x_p,\theta^*)) = Var(\theta_1^* + \theta_2^* x_p + \theta_3^* x_p^2) \\ &= \vec{x}(x_p)^T Cov(\theta^*)\vec{x}(x_p) \end{aligned} \tag{77}$$

Where $\vec{x}(x_p) = (1, x_p, x_p^2)^T$ and $Cov(\theta^*)$ is the covariance matrix for model parameters, easily obtained with any Least Square Regression tool. This way we can propagate the error on the estimate of parameters and get the variance of the mean response. Once we have variance on the mean response if we're interested in a particular confidence interval we can easily evaluate it through t-distribution.

References

[1] P.B. Canham. "The minimum energy of bending as a possible explanation of the biconcave shape of the human red blood cell". In: *Journal of Theoretical Biology* 26.1 (1970), pp. 61–81.

[2] W. Helfrich. In: *Zeitschrift für Naturforschung C* 28.11-12 (1973), pp. 693–703.

[3] Francoise Brochard-Wyart and J.F. Lennon. "Frequency spectrum of flicker phenomenon in erythrocytes. J Phys (France)". In: 36 (Jan. 1975).

[4] Yacov Kantor and David R. Nelson. "Crumpling transition in polymerized membranes". In: *Phys. Rev. Lett.* 58 (26 June 1987), pp. 2774–2777.

[5] C.F. Baillie and D.A. Johnston. "Crossover between weakly and strongly self-avoiding random surfaces". In: *Physics Letters B* 295.3-4 (1992), pp. 249–254.

[6] D. M. Kroll and G. Gompper. "The Conformation of Fluid Membranes: Monte Carlo Simulations". In: *Science* 255.5047 (1992), pp. 968–971.

[7] M. Goulian, R. Bruinsma, and P. Pincus. "Long-Range Forces in Heterogeneous Fluid Membranes". In: *Europhysics Letters* 22.2 (Apr. 1993), p. 145.

[8] Roland R. Netz and P. Pincus. "Inhomogeneous fluid membranes: Segregation, ordering, and effective rigidity". In: *Phys. Rev. E* 52 (4 Oct. 1995), pp. 4114–4128.

[9] G. Taubin. "Estimating the tensor of curvature of a surface from a polyhedral approximation". In: *Proceedings of IEEE International Conference on Computer Vision*. 1995, pp. 902–907.

[10] G. Gompper and D.M. Kroll. "Random Surface Discretizations and the Renormalization of the Bending Rigidity". In: *J. Phys. I France* 6.10 (1996), pp. 1305–1320.

[11] Jeong-Man Park and T. C. Lubensky. "Interactions between membrane Inclusions on Fluctuating Membranes". In: *Journal de Physique I* 6.9 (Sept. 1996), pp. 1217–1235.

[12] Roland R. Netz. "Inclusions in Fluctuating Membranes: Exact Results". In: *J. Phys. I France* 7.7 (1997), pp. 833–852.

[13] Rüdiger Goetz, Gerhard Gompper, and Reinhard Lipowsky. "Mobility and Elasticity of Self-Assembled Membranes". In: *Phys. Rev. Lett.* 82 (1 Jan. 1999), pp. 221–224.

[14] W. Helfrich and T.R. Weikl. "Two direct methods to calculate fluctuation forces between rigid objects embedded in fluid membranes". In: *The European Physical Journal E* 5 (2001), pp. 423–439.

[15] T. R. Weikl. "Fluctuation-induced aggregation of rigid membrane inclusions". In: *Europhysics Letters* 54.4 (May 2001), p. 547.

[16] F. Divet et al. "Fluctuations of a membrane interacting with a diffusion field". In: *Europhysics Letters* 60.5 (Dec. 2002), p. 795.

[17] Thomas R. Weikl. "Dynamic phase separation of fluid membranes with rigid inclusions". In: *Phys. Rev. E* 66 (6 Dec. 2002), p. 061915.

[18] M. Deserno. *Notes on differential geometry*. 2004.

[19] D. Nelson, T. Piran, and S. Weinberg. *Statistical Mechanics of Membranes and Surfaces*. 2nd. WORLD SCIENTIFIC, 2004.

[20] Hiroshi Noguchi and Gerhard Gompper. "Dynamics of fluid vesicles in shear flow: Effect of membrane viscosity and thermal fluctuations". In: *Phys. Rev. E* 72 (1 June 2005), p. 011901.

[21] Albert J Jin et al. "Measuring the elasticity of clathrin-coated vesicles via atomic force microscopy". In: *Biophysical journal* 90.9 (2006), pp. 3333–3344.

[22] Ali Naji and Frank L.H. Brown. "Diffusion on ruffled membrane surfaces". In: *The Journal of chemical physics* 126.23 (2007), 06B611.

[23] G. Gompper et al. "Multi-Particle Collision Dynamics: A Particle-Based Mesoscale Simulation Approach to the Hydrodynamics of Complex Fluids". In: *Advanced Computer Simulation Approaches for Soft Matter Sciences III*. Ed. by Christian Holm and Kurt Kremer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1–87.

[24] Ali Naji, Paul J. Atzberger, and Frank L.H. Brown. "Hybrid elastic and discrete-particle approach to biomembrane dynamics with application to the mobility of curved integral membrane proteins". In: *Physical review letters* 102.13 (2009), p. 138102.

[25] Sivaramakrishnan Ramadurai et al. "Lateral Diffusion of Membrane Proteins". In: *Journal of the American Chemical Society* 131.35 (2009), pp. 12650–12656.

[26] Vitaly V. Slezov. *Kinetics of first order phase transitions*. John Wiley & Sons, 2009.

[27] YongKeun Park et al. "Measurement of red blood cell mechanics during morphological changes". In: *Proceedings of the National Academy of Sciences* 107.15 (2010), pp. 6731–6736.

[28] N. Ramakrishnan, P. B. Sunil Kumar, and John H. Ipsen. "Monte Carlo simulations of fluid vesicles with in-plane orientational ordering". In: *Phys. Rev. E* 81 (4 Apr. 2010), p. 041922.

[29] Ellen Reister-Gottfried, Stefan M. Leitenberger, and Udo Seifert. "Diffusing proteins on a fluctuating membrane: Analytical theory and simulations". In: *Physical Review E* 81.3 (2010), p. 031903.

[30] N. Ramakrishnan, P. B. Sunil Kumar, and John H. Ipsen. "Modeling Anisotropic Elasticity of Fluid Membranes". In: *Macromolecular Theory and Simulations* 20.7 (2011), pp. 446–450.

[31] Kerstin Weiß et al. "Quantifying the Diffusion of Membrane Proteins and Peptides in Black Lipid Membranes with 2-Focus Fluorescence Correlation Spectroscopy". In: *Biophysical journal* 105 (July 2013), pp. 455–62.

[32] Judith Klumperman and Graça Raposo. "The complex ultrastructure of the endolysosomal system". In: *Cold Spring Harbor perspectives in biology* 6.10 (2014), a016857.

[33] Ramakrishnan Natesan, Sunil Kumar, and Ravi Radhakrishnan. "Mesoscale computational studies of membrane bilayer remodeling by curvature-inducing proteins". In: *Physics Reports* (Oct. 2014), pp. 1–60.

[34] Ludger Johannes et al. “Clustering on Membranes: Fluctuations and More”. In: *Trends in Cell Biology* 28.5 (2018), pp. 405–415.

[35] Samuel A. Safran. *Statistical thermodynamics of surfaces, interfaces, and membranes*. CRC Press, 2018.

[36] Everitz H.G. *Computer simulations - lecture notes 2020*. 2020.

[37] Mitja Drab et al. “A Monte Carlo study of giant vesicle morphologies in nonequilibrium environments”. In: *Biophysical Journal* 120.20 (2021), pp. 4418–4428.

[38] Elisa Floris et al. “Physics of compartmentalization: How phase separation and signaling shape membrane and organelle identity”. In: *Computational and Structural Biotechnology Journal* 19 (2021), pp. 3225–3233.

[39] Marco Zamparo et al. “Optimality in Self-Organized Molecular Sorting”. In: *Phys. Rev. Lett.* 126 (8 Feb. 2021), p. 088101.

[40] Hammad Faizi, Rony Granek, and Petia Vlahovska. “Membrane viscosity signature in thermal undulations of curved fluid bilayers”. In: *arXiv preprint arXiv:2208.07966* (Aug. 2022).

[41] Elisa Floris et al. “Phase separation and critical size in molecular sorting”. In: *Phys. Rev. E* 106 (4 Oct. 2022), p. 044412.

[42] Sebastian Himbert et al. “The bending rigidity of the red blood cell cytoplasmic membrane”. In: *PLOS ONE* 17.8 (Aug. 2022), pp. 1–20.

[43] E.S. Pikina et al. “Long-range interactions between membrane inclusions: Electric field induced giant amplification of the pairwise potential”. In: *Annals of Physics* 447 (2022), p. 168916.

[44] Andrea Piras et al. “Sorting of multiple molecular species on cell membranes”. In: *arXiv preprint arXiv:2301.05966* (2023).