

Rapport - Projet Aire Par Lipide (APL)

M1BI : Dounia BENYAKHLAF (22007439) et Louise LAM (21980795)

Référente : Maya ZYGADLO

Introduction

La simulation de dynamique moléculaire est un outil fondamental pour explorer les propriétés complexes des membranes biologiques. Dans cette étude, nous nous concentrons sur l'estimation de l'Aire Par Lipide (APL) à l'aide des tessellations de Voronoï.

Les membranes biologiques jouent un rôle crucial dans de nombreux processus biologiques en raison de leur implication dans l'organisation cellulaire, la signalisation et le transport de molécules. L'APL, qui représente la surface disponible pour chaque lipide dans la membrane, influence directement l'assemblage, la stabilité et les interactions avec d'autres composants cellulaires, tels que les protéines.

Les méthodes traditionnelles pour déterminer l'APL sont souvent coûteuses et chronophages. Ainsi, il est devenu impératif de développer des approches computationnelles pour estimer cette mesure de manière plus rentable et efficace. Notre projet vise précisément à combler cette lacune en proposant une méthode *in silico* pour l'estimation de l'APL.

Nous avons appliqué notre méthode à une membrane composée uniquement de phospholipides DMPC (phosphocholine), puis étendu notre analyse à une membrane mixte contenant à la fois des DMPC et des DMPG (phosphoglycérol). Les APL calculées pour chaque type de lipide *in silico* seront comparées aux données expérimentales disponibles. Enfin, le programme générera également trois graphiques : le diagramme de Voronoï de la première et de la dernière frame, ainsi que la distribution de l'APL en fonction des frames.

Méthodes

Afin d'atteindre notre objectif d'estimation de l'APL par le biais des tessellations de Voronoï, notre approche méthodologique repose sur une série d'étapes bien définies.

Données d'entrée

Notre projet prend en charge le calcul de l'Aire Par Lipide (APL) pour deux types de membranes : une contenant uniquement du DMPC et une autre composée à parts égales de DMPC et de DMPG [\[Annexe 1\]](#).

Le 1,2-dimyristoyl-sn-glycéro-3-phosphocholine (DMPC) est un phospholipide comprenant une chaîne carbonée saturée de 14 atomes de carbone. Il est couramment utilisé comme modèle en biophysique membranaire en raison de sa capacité à former des bicouches lipidiques bien ordonnées à des températures physiologiques. D'autre part, le 1,2-dimyristoyl-sn-glycéro-3-phosphoglycérol (DMPG) est un phospholipide doté d'une chaîne carbonée saturée de 14 atomes de carbone et d'un groupement glycérol estérifié par deux acides gras. Il est également largement utilisé comme modèle dans les études de membranes lipidiques en raison de ses propriétés amphiphiles et de sa capacité à former des structures membranaires similaires aux membranes biologiques.

Pour nos analyses, nous disposons de fichiers de topologie .gro [\[3\]](#) et de fichiers de trajectoire .xtc [\[4\]](#) provenant de simulations de dynamique moléculaire effectuées en laboratoire. Ces membranes ont été construites avec CHARMM-GUI, une plateforme dédiée à la construction de systèmes pour la simulation de dynamique moléculaire. Elles ont ensuite été soumises à une minimisation et à un équilibrage en conditions NVT (nombre d'atomes, volume et température constants) puis NPT (nombre d'atomes, pression et température constants). Il convient de noter que ces modèles membranaires, bien que représentatifs sur le plan structural, ne sont pas des systèmes biologiques, mais des structures générées pour les besoins de la simulation. La trajectoire de la membrane exclusivement composée de 128 DMPC se compose de 5001 frames, tandis que celle de la membrane mixte, contenant 64 DMPC et 64 DMPG, ne comprend que 1001 frames.

Ainsi, à partir de ces fichiers de topologie et de trajectoire, nous pouvons calculer une APL moyenne par type de lipides sur l'ensemble de la simulation.

Protocole

Ce projet a été réalisé avec la version 3.11.7 de Python. Bien qu'il ne soit pas nécessairement le langage le plus performant en termes de vitesse brute, il présente de nombreux avantages grâce à sa grande flexibilité et à la richesse de ses bibliothèques, que nous avons exploitées. De plus, le code produit est réputé pour sa lisibilité et sa clarté.

En complément, notre analyse s'est appuyée sur trois dépendances majeures :

- [MDAnalysis 2.4.3 \[5\]](#) : Cette bibliothèque est largement reconnue pour son utilité dans l'analyse et la manipulation de données de simulation de dynamique moléculaire.
- [Scipy 1.10.1 \[6\]](#) : En offrant une large gamme de fonctions mathématiques puissantes et efficaces, Scipy s'est avérée être une ressource précieuse pour notre projet.
- [Shapely 2.0.1 \[7\]](#) : Cette bibliothèque a été essentielle pour la création, la manipulation et la validation de formes géométriques telles que les points, les lignes et les polygones.

Grâce à ces trois outils, nous avons pu effectuer le calcul de l'Aire Par Lipide (APL) pour les deux types de lipides, selon la méthodologie suivante :

```
APL:
univers = lecture(.gro, .xtc)           - lecture des fichiers .gro et .xtc avec MDAnalysis.
pour chaque frame dans la trajectoire: - pour chaque frame :
    p_atoms = selection(P atoms)        - récupérer les positions des phosphates représentatifs des lipides
    diagram_voronoi = Voronoi(p_atoms)  - création d'un diagramme de Voronoï avec la bibliothèque Scipy
    pour chaque region dans diagram_voronoi:
        aires_regions += aire(region)   - pour chaque région du diagramme :
        - récupération des aires en les transformant en Polygon avec Shapely
    frames_apl += moyenne(aires_regions) - l'aire moyenne par frame calculée est stockée dans une liste
apl_trajectoire = moyenne(frames_apl)  - la moyenne de l'APL sur toutes les frames est calculée
```

Ce protocole est adapté aux membranes contenant un seul type de lipides. Pour les membranes mixtes, le même protocole s'applique ; il suffit simplement de calculer une APL pour chaque type de lipides.

Dans les sections suivantes, nous allons détailler l'utilisation de chaque bibliothèque mentionnée, en mettant en évidence son rôle spécifique dans notre analyse.

MDAnalysis (version 2.4.3)

L'utilisateur a la possibilité de choisir sur quelle membrane (simple ou mixte) il souhaite calculer l'APL. Une fois ce choix effectué, c'est le paquet MDAnalysis qui s'occupe du chargement des fichiers, du parcours des frames ainsi que de la sélection des lipides de la façon suivante :

- Chargement des fichiers : Dans le `main`, les fichiers de topologie `.gro` et de trajectoire `.xtc` sont chargés à l'aide de la méthode `universe()`, qui génère un univers, un objet Python contenant dans ses attributs les caractéristiques de la simulation.
- Parcours des frames : L'univers généré possède notamment un attribut `.trajectoire` sur lequel on peut itérer pour parcourir chaque frame.
- Sélection des lipides : Ainsi, pour chaque frame, nous avons sélectionné les lipides via leur atome de phosphore à l'aide de la méthode `select_atoms("name P")`. Il est également possible de les sélectionner selon leur type (DMPC ou DMPG) avec cette méthode `select_atoms("resname DMP_")`. Enfin, étant donné que nos membranes sont des bicouches, nous avons décidé de limiter nos calculs au feuillet supérieur. Pour ce faire, nous avons récupéré les coordonnées en Å via l'attribut `.positions` puis nous avons identifié le feuillet en utilisant un masque basé sur les coordonnées z : les lipides le z est supérieur à la moyenne sont considérés comme appartenant au feuillet supérieur.

Scipy (version 1.10.1)

Une fois que nous avons récupéré les coordonnées de nos atomes de phosphore, la prochaine étape consiste à construire un diagramme de Voronoï pour la frame en question. Pour ce faire, nous utilisons la méthode `voronoi()` du paquet `Scipy.spatial [6]`, qui construit un diagramme en se basant sur l'algorithme des tessellations de Voronoï.

Cet algorithme divise l'espace en cellules en utilisant un ensemble de points de contrôle, représentés ici par les phosphates des lipides membranaires. Pour chaque point, il identifie la région de l'espace la plus proche, délimitant ainsi les frontières des cellules de Voronoï. Ces frontières sont généralement calculées comme les lignes médianes entre le point en question et ses voisins les plus proches. Le résultat final est une tessellation de l'espace en cellules de forme

irrégulière, où chaque cellule représente la région de l'espace la plus proche d'un point de contrôle spécifique. Par souci de temps de calcul, nous avons décidé de ne conserver que la visualisation de la première et de la dernière frame, qui sont sauvegardées dans le répertoire fig/ grâce à la bibliothèque matplotlib.pyplot.

Ainsi, pour chaque frame, nous disposons d'un objet Voronoï qui possède à la fois un attribut `.vertices`, listant les coordonnées de tous les sommets des régions, ainsi qu'un attribut `.region` qui liste toutes les régions du diagramme représentées par des ensembles d'indices de sommets. En les combinant, nous pouvons alors obtenir la liste des régions du diagramme représentées par des ensembles de coordonnées.

Shapely (version 2.0.1)

En réalité, la construction d'un objet Voronoï a permis de délimiter mathématiquement les régions entourant les lipides, facilitant ainsi le calcul de leur aire. L'objectif est alors de recueillir l'APL de chaque frame afin d'en déduire une APL moyenne sur l'ensemble de la trajectoire.

Pour ce faire, nous transformons chaque région des diagrammes de Voronoï, représentés par des ensembles de coordonnées, en objets géométriques Polygon grâce à la méthode éponyme du paquet Shapely [8]. Ces objets ont en particulier l'attribut `.area` contenant l'aire du polygone correspondant. Nous sauvegardons alors la liste des aires de tous les polygones du diagramme de Voronoï afin de déduire une valeur moyenne d'APL pour chaque frame, dont la distribution est tracée puis sauvegardée dans le répertoire fig/. Enfin, la valeur moyenne de l'APL sur toute la trajectoire est donc la moyenne des APL de toutes les frames, à laquelle on applique un facteur de conversion pour obtenir des résultats des Å² en nm².

Application sur les données fournies

Nous appliquons le protocole précédemment détaillé à la fois sur la membrane simple composée de 100% de DMPC et sur la membrane mixte composée de 50% DMPC - 50% DMPG. Ensuite, nous étudions les résultats obtenus (diagrammes de Voronoï, distribution de l'APL à travers la simulation, valeur d'APL calculée) et les comparons aux valeurs expérimentales.

Étude de la membrane 100% DMPC

Avant toute chose, visualisons le système membranaire à l'aide de VMD :

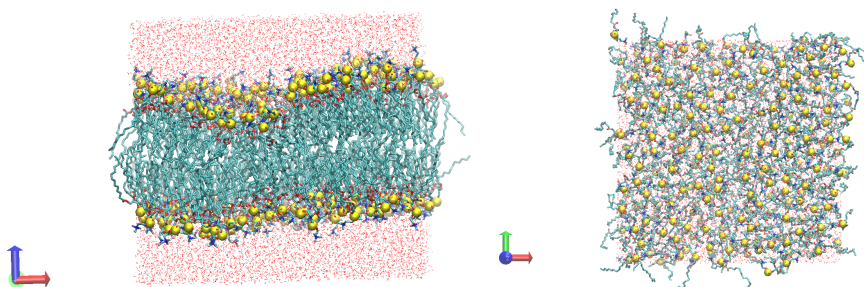


Figure 1 : Visualisations latérale et en surface de la membrane simple 100% DMPC avec VMD. Les molécules d'eau sont en lines (O en rouge, H en blanc), les queues hydrophobes des DMPC sont en sticks (C en cyan, O en rouge, N en bleu) et les atomes de phosphore des têtes hydrophiles sont en sphères jaune.

Nous pouvons maintenant étudier les diagrammes de Voronoï générés pour les première et dernière frame.

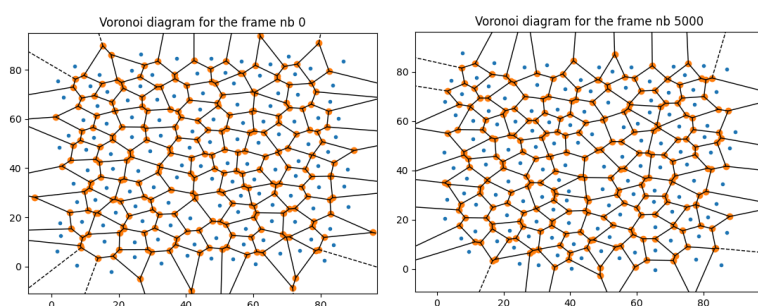


Figure 2 : Diagrammes de Voronoï de la première (0) et de la dernière (5000) frame pour la membrane simple. Les points bleus représentent les points de contrôle, tandis que les points oranges correspondent aux *vertices*. Les axes x et y représentent les positions de chaque point en Å.

Cette visualisation met en évidence les régions occupées par chaque lipide dans la couche supérieure de la membrane DMPC, tant au début qu'à la fin de la trajectoire de simulation [Figure 2]. Cette stabilité peut être attribuée à la nature ordonnée des bicouches lipidiques formées par les lipides DMPC, connus pour leur capacité à maintenir une structure cohérente. De plus, l'absence de changements significatifs dans les régions occupées par les lipides suggère une stabilité spatiale, indiquant une faible dynamique de mouvement des lipides dans la membrane à travers la simulation. Cette stabilité de la membrane peut être cruciale pour de nombreuses fonctions biologiques.

Observons comment ces résultats sont confirmés par la distribution de l'APL à travers la trajectoire.

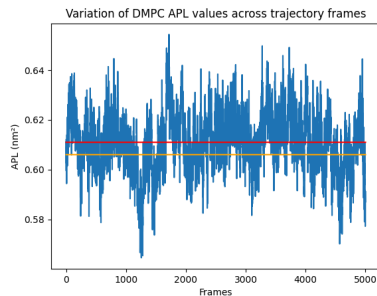


Figure 3 : Évolution de l'APL en nm² selon les frames de la trajectoire de simulation de la membrane simple. La ligne rouge illustre la moyenne globale de l'APL estimée in silico et la ligne jaune la valeur expérimentale.

Les données présentées dans la figure 3 illustrent une variation modérée de l'APL dans la trajectoire de simulation, oscillant entre 0,58 et 0,64 nm². Cette constance suggère une stabilité dans la structure et la densité de la membrane lipidique simulée. La valeur moyenne de l'APL pour une membrane composée exclusivement de DMPC est estimée à 0,611 nm² par notre programme, ce qui est très proche de la valeur trouvée expérimentalement de 0,606 ± 0,005 nm² [1]. L'écart entre ces deux valeurs est minime ($\approx 0,005$ nm²) et il est compris dans la marge d'erreur. Ces résultats indiquent que cette simulation de dynamique moléculaire (128 DMPC sur 5001 frames) reproduit fidèlement les caractéristiques de la membrane lipidique biologique, soulignant ainsi la robustesse de notre approche in silico pour la représentation des propriétés de cette membrane.

Étudions maintenant les performances de notre estimateur d'APL sur la membrane mixte.

Étude de la membrane 50%DMPC-50%DMPG

Avant toute chose, visualisons le système membranaire à l'aide de VMD :

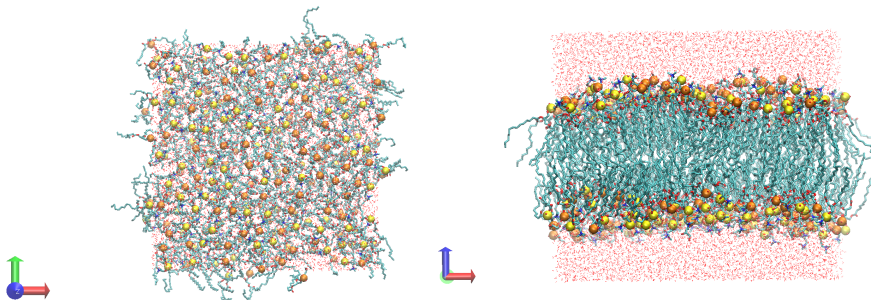


Figure 4 : Visualisations latérale et en surface de la membrane mixte 50% DMPC-50% DMPG avec VMD. Les molécules d'eau sont en lines (O en rouge, H en blanc), les queues hydrophobes des lipides sont en sticks (C en cyan, O en rouge, N en bleu) et les atomes de phosphore des têtes hydrophiles sont en spheres, **jaune pour les DMPC et orange pour les DMPG**.

Nous pouvons maintenant étudier les diagrammes de Voronoï générés pour les première et dernière frame.

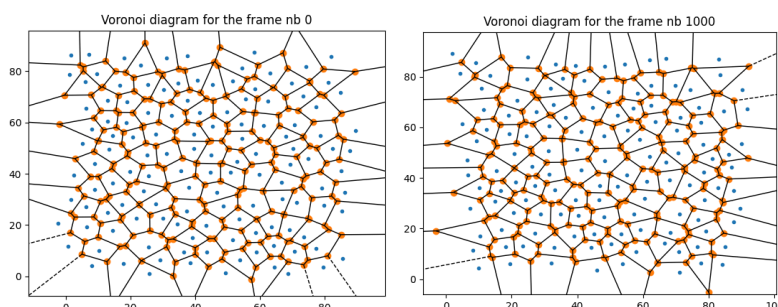


Figure 5 : Diagrammes de Voronoï de la première (0) et de la dernière (5000) frame pour la membrane mixte. Les points bleus représentent les points de contrôle, tandis que les points oranges correspondent aux *vertices*. Les axes x et y représentent les positions de chaque point en Å.

De manière similaire aux diagrammes de Voronoï de la membrane simple, nous remarquons que les régions occupées par les lipides ne changent pas beaucoup, ce qui témoigne d'une certaine stabilité et d'une faible dynamique. Cependant, un examen plus approfondi révèle une stabilité légèrement moindre par rapport à la membrane simple

composée exclusivement de DMPC. Cette observation suggère que l'incorporation de DMPG dans la membrane pourrait avoir un effet sur la dynamique de la membrane, bien que celle-ci reste globalement stable. Il est possible que les interactions entre les lipides DMPC et DMPG influencent les arrangements lipidiques et les mouvements moléculaires dans la membrane mixte, conduisant à des fluctuations plus importantes par rapport à la membrane simple. Ces résultats mettent en lumière l'importance de considérer la composition lipidique lors de l'étude des propriétés membranaires.

Comparons à présent la distribution de l'APL à travers la trajectoire pour les DMPC et les DMPG.

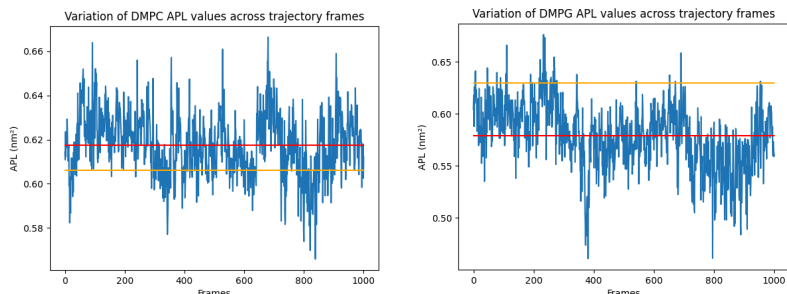


Figure 6 : Variation de l'APL (en nm²) des DMPC d'un côté et des DMPG de l'autre en fonction des frames de la trajectoire de simulation de la membrane mixte. La ligne rouge représente la valeur moyenne de l'APL globale et la ligne jaune la valeur expérimentale.

L'APL des DMPC dans la membrane DMPC-DMPG varie de 0,50 à 0,66 nm² au cours de la simulation, avec une APL globale estimée à 0,618 nm² [Figure 5]. Cependant, l'écart entre cette estimation et la valeur expérimentale (0,606 ± 0,005 nm²) [1] est de 0,01 nm², dépassant ainsi la marge d'erreur expérimentale de 0,005 nm².

De façon analogue, pour les DMPG, l'APL fluctue entre 0,48 et 0,68 nm², avec une APL globale de 0,579 nm² [Figure 6]. L'écart entre l'APL estimée et la valeur expérimentale (0,630 ± 0,005 nm²) [2] est de 0,05 nm², dépassant également la marge d'erreur de 0,005 nm².

Ces divergences peuvent être attribuées au nombre réduit de lipides de chaque type, étant deux fois moins nombreux (64 de chaque) que dans la simulation précédente, où la membrane était exclusivement composée de DMPC (128). De plus, la trajectoire de simulation comporte seulement 1001 frames, soit cinq fois moins que dans la simulation antérieure (5001 frames). Ainsi, l'échantillonnage plus faible conduit à une représentation moins précise du système étudié. Avec moins de données disponibles, les estimations peuvent être plus sensibles aux fluctuations aléatoires, ce qui peut conduire à des résultats moins fiables et moins proches de la réalité. Notons tout de même que malgré cet échantillonnage limité, la marge d'erreur observée de 0,05 nm² est tout à fait acceptable, ce qui confirme que notre programme d'estimation demeure relativement précis.

Conclusion

Dans cette étude, nous avons appliqué avec succès une méthode *in silico* basée sur les tessellations de Voronoï pour estimer l'Aire par Lipide (APL) dans des membranes lipidiques générées par dynamique moléculaire. Notre approche s'est appuyée sur l'utilisation de plusieurs packages Python, notamment MDAnalysis, Scipy et Shapely, qui ont joué un rôle crucial dans le chargement des données, la construction des diagrammes de Voronoï et le calcul des aires des régions. Nous avons observé une cohérence relative entre nos estimations et les valeurs expérimentales, malgré des écarts dans certains cas, attribuables en partie à un échantillonnage limité et à une durée de simulation courte.

Pour améliorer notre méthode, plusieurs pistes peuvent être envisagées. Tout d'abord, il serait intéressant de générer des fichiers de trajectoire plus longs avec un nombre plus élevé de lipides pour la membrane mixte, permettant ainsi une meilleure représentation statistique du système. Ensuite, étant donné le temps de calcul moyen d'environ 5 minutes, une optimisation des performances en reprogrammant certaines fonctions des packages utilisés pourrait être envisagée pour exploiter plus efficacement les ressources informatiques. De plus, une autre amélioration consisterait à considérer les deux feuillet de la membrane plutôt que de se limiter arbitrairement au feuillet supérieur, ce qui pourrait offrir une vision plus complète et précise des propriétés membranaires. Enfin, une généralisation du code pour calculer l'APL pour tous les types de lipides présents dans les fichiers de simulation serait une extension intéressante, permettant une utilisation plus large de notre méthode dans différents contextes biologiques.

Bien que des améliorations soient possibles pour optimiser notre méthode, nos résultats offrent des perspectives encourageantes pour réduire le recours aux analyses expérimentales coûteuses et chronophages. Cela ouvre ainsi la voie à des études plus rentables et plus efficaces des membranes biologiques.

Annexe

Répartition du travail

Nous avons adopté une méthode de travail collaborative et efficace pour mener à bien ce projet. Utilisant GitHub comme plateforme principale pour la gestion des versions et la collaboration, nous avons pu synchroniser constamment notre travail. Nous avons également opté pour une approche en *pair-programming* en alternant les rôles de conducteur. Cela nous a permis de partager équitablement la responsabilité de la rédaction du code tout en permettant une correction rapide des erreurs et un partage des idées pour parvenir à un code optimal et de qualité. Nous étions également constamment à jour sur le code produit, évitant ainsi le besoin de rattraper le travail de l'autre. Ainsi, cette approche nous a permis de progresser efficacement en exploitant nos compétences complémentaires et en favorisant une collaboration fluide.

Difficultés rencontrées

Nous avons fait face à plusieurs difficultés tout au long du projet.

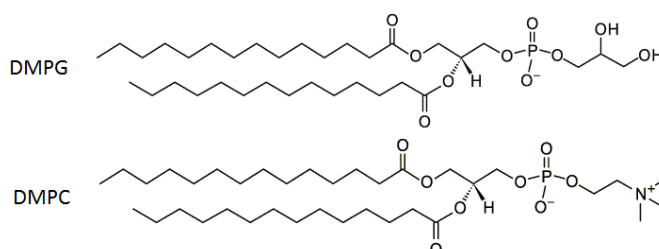
Tout d'abord, l'utilisation de GitHub avec des fichiers de simulation de grande taille s'est avérée problématique. Nous avons dû recourir à *Git LFS (Large File Storage)* pour gérer efficacement ces fichiers volumineux, ce qui a demandé un temps d'apprentissage supplémentaire.

Ensuite, la définition des régions ouvrantes pour le calcul des aires par molécule a posé un défi technique. Ces régions ouvrantes se réfèrent aux parties du diagramme de Voronoï qui dépassent de la boîte de simulation, altérant ainsi nos calculs. Dans un premier temps, nous avons choisi de les ignorer, mais cela a affecté la précision de nos résultats. Par la suite, nous avons opté pour le calcul de l'intersection entre chaque polygone représentant une région et la boîte représentée par l'objet box. Cette approche nous a permis de corriger les erreurs liées aux régions ouvrantes, améliorant ainsi la fiabilité de nos calculs pour les APL.

De plus, le calcul des APL pour la membrane mixte s'est révélé plus complexe que prévu. Nous avons réalisé qu'il ne suffisait pas simplement d'appliquer exactement le même protocole que pour la membrane simple, nécessitant ainsi une adaptation plus poussée de notre méthode.

Bien que ces difficultés aient ralenti notre progression, nous avons pu les surmonter en travaillant de manière collaborative et en recherchant des solutions créatives.

Figures facultatives



Annexe 1 : Formules topologiques des lipides DMPG et DMPC.

Références

- [1] D. J. Bonthuis, et al. (2006). "Dynamics in atomistic simulations of phospholipid membranes: nuclear magnetic resonance relaxation rates and lateral diffusion". *The Journal of Chemical Physics*, 125(20), 204703.
- [2] N. Kucerka, Y. Liu, N. Chu, H.I. Petrache, S. Tristram-Nagle, J.F. Nagle. (2005). "Structure of fully hydrated fluid phase DMPC and DLPC lipid bilayers using X-ray scattering from oriented multilamellar arrays and from unilamellar vesicles". *Biophysical Journal*, 88(4), 2626-2637. doi: 10.1529/biophysj.104.056606.
- [3] GROMACS. (Consulté en mars 2024). Format de fichier GRO. [En ligne]
Disponible sur : <https://manual.gromacs.org/archive/5.0.3/online/gro.html>
- [4] GROMACS. (Consulté en mars 2024). Format de fichier XTC. [En ligne]
Disponible sur : <https://manual.gromacs.org/archive/5.0.4/online/xtc.html>
- [5] MDAnalysis. (Consulté en mars 2024). Guide de l'utilisateur - Démarrage rapide. [En ligne]
Disponible sur : <https://userguide.mdanalysis.org/stable/examples/quickstart.html>
- [6] SciPy. (Consulté en mars 2024). Classe Voronoi. [En ligne]
Disponible sur : <https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.Voronoi.html>
- [7] Shapely. (Consulté en mars 2024). Documentation. [En ligne]
Disponible sur : <https://shapely.readthedocs.io/en/stable/manual.html>
- [8] Shapely. (Consulté en mars 2024). Classe Polygon. [En ligne]
Disponible sur : <https://shapely.readthedocs.io/en/stable/reference/shapely.Polygon.html>