Curtin HIVE (Hub for Immersive Visualisation and eResearch)
and School of Pharmacy and Biomedical Sciences

What happens when you rub on that cream? Understanding how drug molecules penetrate our skin by molecular visualization

Michael T. Greif

Supervisors: Prof. Ricardo L. Mancera, Lanie Ruiz-Perez
(School of Pharmacy and Biomedical Sciences)

Curtin University, GPO Box U1987, Perth WA 6845

February 2020

A research project supported by the Curtin HIVE Summer Internship Program (2019-20)

"What happens when you rub on that cream? Understanding how drug molecules penetrate our skin by molecular visualization"

Michael T. Greif, Prof. Ricardo L. Mancera, Lanie Ruiz Perez.

Curtin HIVE internship report (2020).

Abstract

The outermost layer of the skin, the *stratum corneum* (SC), is highly impermeable and blocks foreign molecules from entering the body because it is made of dead cells (corneocytes) and multiple lipid layers. Delivery of drugs through the skin is advantageous as their release is slow and this method is the least invasive. Attachment of short lipid tails can increase the permeability of peptide drugs through the SC. Molecular dynamics (MD) simulations have been used to study the molecular mechanism of permeation of a model peptide through a lipid bilayer representative of the SC. Although some of the computational methods used can better simulate the process of permeation, they also pose difficulties for their visualization in a 3D stereoscopic display, such as in the Unity-based Molecular Dynamics Visualization (MDV) program. This project aimed to overcome the intrinsic challenges of visualizing MD simulations in MDV through two novel approaches. The first approach removed most components of the lipid bilayer, leaving only certain key atoms and a number of representative lipid molecules. The second approach was based on a pull simulation of the peptide across the lipid bilayer, and used previously designed post-processing workflows to produce a visually appealing visualization. Both approaches resulted in visualizations that were successfully displayed using MDV on the Curtin HIVE Cylinder display.

Table of Contents

1. Introduction

Drug delivery through the skin is unique as the slow permeation of drugs results in a long, steady dose. Topical applications are less invasive than intravenous or intramuscular injection and more convenient than oral administration; in particular for cosmetic purposes or for treatment of skin conditions. The stratum corneum (SC), the outermost layer of the skin, is made of dead cells (corneocytes) and multiple lipid layers. This makes the SC exceptionally good at preventing water and many other molecules from entering the body. Consequently, drugs are often unable to penetrate the skin.

Molecular dynamics (MD) simulations have been used to describe in atomistic detail the interactions and mechanism of permeation of drug molecules across a lipid bilayer representative of the SC. For this project the molecular mechanism of permeation of a model dipeptide molecule across a lipid bilayer was studied using a combination of two so-called enhanced sampling techniques. In the first, the peptide is placed and restrained at different positions, or 'windows', outside and inside the bilayer, in what is referred to as umbrella sampling (US). (1) In the second, simulation replicas corresponding to varying conditions are conducted in parallel and are allowed to swap, in what is referred to as replica exchange with solute tempering (REST). (2) The combined US+REST approach is effective at sampling the interactions of the peptide with the bilayer at different depth. However, it results in difficulties in the visualization of the simulation; largely jumps in molecular positions that occur between concatenated windows and jumps after replica exchange. The aim of this project was to develop a suitable approach to creating an appealing stereoscopic visualization of the peptide crossing a lipid bilayer using the Molecular Dynamics Visualization (MDV) program. (3)

2. Background

Molecular Dynamics Visualization (MDV)

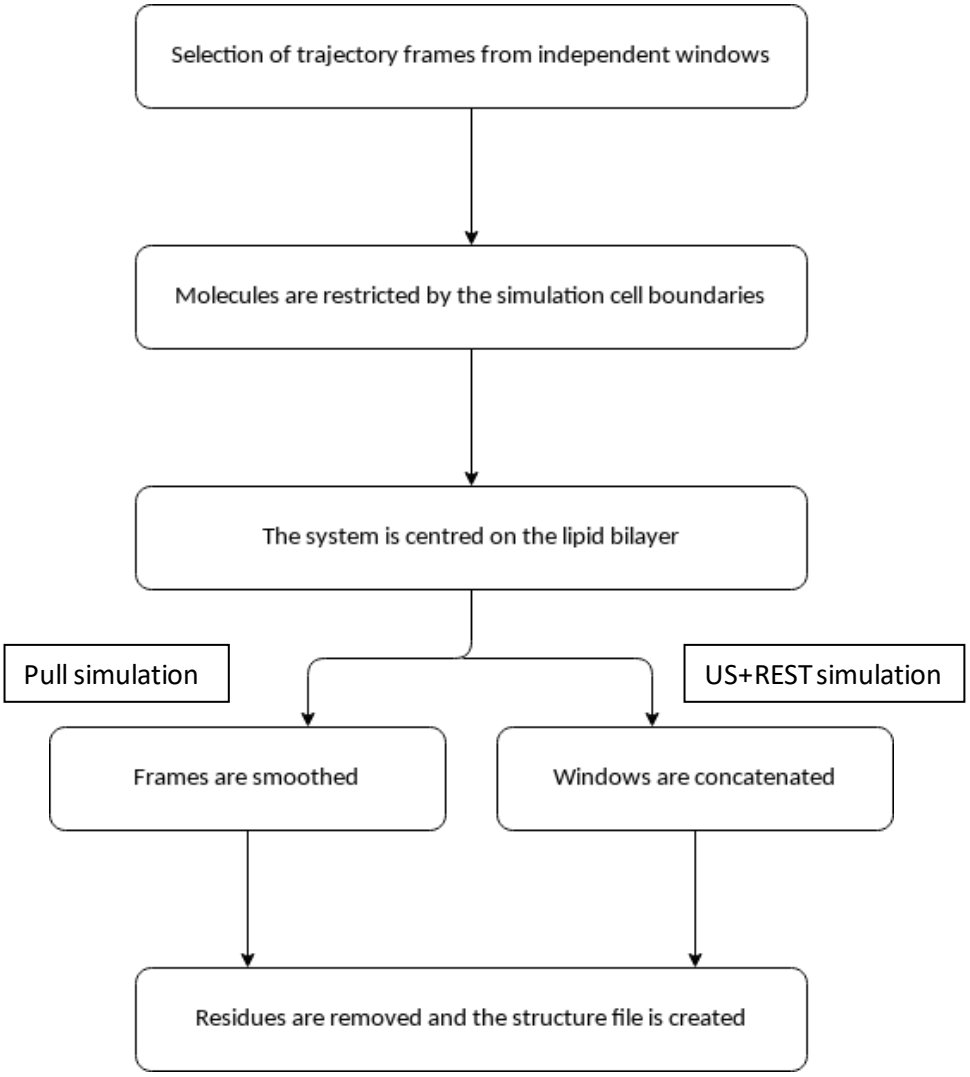
MDV is a Unity-based molecular visualization suite that was developed specifically for use at the Curtin HIVE (Hub for Immersive Visualization and eResearch) on a 3 x 8 m, 180 ° cylindrical visualization system that displays content in high-resolution stereoscopic 3D. (3, 4)

3. Methodology

3.0 Overview of methodology

MD simulations tested for visualization were conducted using two different computational approaches: a pull simulation and a US+REST simulation.

The workflow for the post-processing of raw trajectories was:



3.1 MD pulling simulation

Pre-processing

The pre-processing commands to conduct this type of simulation were executed using Gromacs 4.6.7. (5)

A parameter file (.mdp) was created for the system with a pull command specifying constant force and periodic direction (Appendix 1a). The Gromacs command *grompp* was used to create a portable binary run input (.tpr) file using, as input, an index (.ndx) file and previously prepared system topology (.top) and structure (.gro) files (Figure 1). The index file was created using the Gromacs command *make_ndx* command using the structure file as the input.

```
make_ndx -f struct.gro -o index.ndx
grompp -f pull.mdp -c struct.gro -p pull.top -o pull.tpr -n index.ndx
```

The simulation was run on Pawsey's supercomputer Magnus for 192 core hours with the Gromacs command *mdrun* using the portable binary run input file.

```
srun -mpi=pmi2 -N 2 -n 48 mdrun_mpi -v -maxh 4 -deffnm pull -px pull.xvg
```

Post-processing

The trajectory was post-processed with Gromacs 2018.1 broadly following the method described by Malajczuk C.J. (4)

Visual Molecular Dynamics (VMD) is one of the most widely used visualisation software for biomolecules and was used to simply and quickly view raw trajectories. (3) VMD was used to confirm that the simulation had run as expected, and a range of frames were chosen for processing. By default Gromacs uses picoseconds as time units and VMD uses frames. The Gromacs command *check* can be used to determine the time range and thus the relationship between frames and time in the trajectory (.xtc) file. Gromacs command *trjconv* was then used to crop a section of frames between the flags -b and –e for *begin* and *end*, respectively.

```
gmx check -f pull.xtc
gmx trjconv -f pull.xtc -o pull_b1000e2000.xtc -s pull.tpr -b 1000 -e 2000
```

The trajectories were further processed to prevent molecules from crossing the boundaries of the simulation area with the flag -pcb nojump and molecules were kept whole with the flag -pcb mol.

```
gmx trjconv -f pull_b1000e2000.xtc -o pull_b1000e2000_nj.xtc -s pull.tpr -pbc nojump
gmx trjconv -f pull_b1000e2000_nj.xtc -o pull_b1000e2000_njm.xtc -s pull.tpr –pbc
mol
```

The lipid bilayer was centred in the middle of the simulation cell using the flag -center.

```
gmx trjconv -f pull_b1000e2000_njm.xtc -o pull_b1000e2000_njm_c19.xtc -s pull.tpr -
n index.ndx -center
```

The trajectory was filtered using the low pass filter (-ol) fitted on the lipid structure (-fit), writing and smoothing every five frames (-nf 5) and writing out all filtered frames (-all), as -all overrides -nf 5.

```
gmx filter -f pull_b1000e2000_njm_c19.xtc -ol pull_b1000e2000_njm_c19_ol5all.xtc -s pull.tpr -nf 5 -fit -n index.ndx -all
```

To remove molecules from the trajectory, a custom index file entry was made with Gromacs command *make_ndx* by grouping desired residues. The system was then filtered with Gromacs *trjconv* using the custom index file.

```
gmx make_ndx -n index.ndx -o myindex.ndx -f struct.gro
```

```
gmx trjconv -f pull_b1000e2000_njm_c19_ol5all.xtc –o pull_b1000e2000_njm_c19_ol5all_rm.xtc -n myindex.ndx
```

Similarly, the structure file was filtered for the desired residues; for this a custom Python program was developed (Appendix 1b). A more traditional way to obtain a structure file is to dump a frame from a trajectory using an index file with *trjconv* and the -dump flag.

```
gmx trjconv -f pull_b1000e2000_njm_c19_ol5all_rm.xtc -s pull.tpr -n myindex.ndx -o struct_rm.gro -dump 10
```

3.2 MD simulation of US+REST

Pre-processing

Accurately simulating the permeation of a solute across a SC bilayer requires enhanced sampling methods. To visually recreate the path of the peptide across the bilayer data must be collected from independent simulations where the peptide is restrained at several depths of the bilayer (US windows). The raw trajectories (.xtc) were obtained from US+REST simulations of an AlaTrp dipeptide permeating across one SC lipid bilayer using an in-house code based on the Gromacs 4.6.7. This produced raw trajectory, topology (.top), and structure files; an index file was created using Gromacs *make_ndx*.

```
gmx make_ndx -f struct.gro -o index.ndx
```

Post-processing

VMD was used to confirm the simulation had run as expected and a range of frames were chosen from each window for processing. After deciding the targe frames and corresponding time in the trajectory file, Gromacs *trjconv* was used to obtain a trajectory with only selected frames from each US window. The number of frames selected for each US window was calculated considering the 1000 frame limitation of the current implementation of MDV and the number of US windows available.

```
gmx check -f traj0.2.xtc
```

```
gmx trjconv -f traj0.2.xtc -o traj0.2_b1000e1080.xtc -s pull.tpr -b 1000 -e 2000
```

The trajectories were processed to keep molecules with the flag *-pcb mol*.

```
gmx trjconv -f traj0.2_b1000e1080.xtc -o traj0.2_b1000e1080_m.xtc -s pull.tpr –pbc
mol
```

US trajectories were concatenated into a single file that followed the translocation of the peptide across the bilayer. For this Gromacs *trjcat* was and concatenation times were given using the -settime flag.

```
gmx trjcat -f traj?.?_b1000e1080_m.xtc -o trajcomb_0.0_3.8_22_
b1000e1080_njm_c19.xtc -settime
```

Our approach to address the inherent challenges of visualising US-REST simulations was to remove the bulk of the lipid layer. The rationale is to create a clear visualization where the most relevant features of the simulations are retained: the two water-bilayer interfaces, the centre of the bilayer and the orientation changes of the peptide as it moves towards the bilayer centre. A custom index file entry that included only selected atoms and molecules was created using Gromacs *make_ndx*.

The oxygen and nitrogen atoms of lipids are located at the water-bilayer interface and gave a visual representation of the boundary. They were selected using the atom type, *t*, option and atom types *O* and *OA* for oxygen, N for nitrogen. Representative lipids residues from each leaflet and lipid type were included with the residue option, *r*, and the residue number given to them in the structure file. Finally, the atoms marking the centre of the bilayer were selected by a combination of their molecule type, *r*, and the atom designation, *a*, of the terminal carbon. All groups were then combined to create an actionable index group.

```
gmx make_ndx -n index.ndx -o myindex.ndx -f struct.gro

19 & t N O OA
r CHOL & a C29
r CR224 & a C49
r FA24 & a C24
r 10 | r 30 | r 50 | r 66 | r 67 | r 90 | r 110 | r 130 | r 131 | r 145 | r 155 | r 162
8 | 23 | 24 | 25 | 26 | 28 | 29
q
```

The index group was then used to edit the trajectory.

```
gmx trjconv -f trajcomb_0.0_3.8_22_ b1000e1080_njm_c19.xtc -o
trajcomb_0.0_3.8_22_ b1000e1080_njm_c19_float.xtc -n myindex.ndx
```

The structure file needs to match the atoms and residues in the filtered trajectory. To efficiently remove unwanted atoms from the structure file a custom Python program was developed (Appendix 1b). The program deletes lines (atoms) based on user input in the form of exclusionary or inclusionary text strings; acting as a python object or an interactive terminal script. The program searches each line (atom) of the structure file and removes it if all inclusionary, and no exclusionary,

strings are found. An alternative way to create a structure file is to use a trajectory file, customised index file, trjconv and the -dump flag.

4. Results

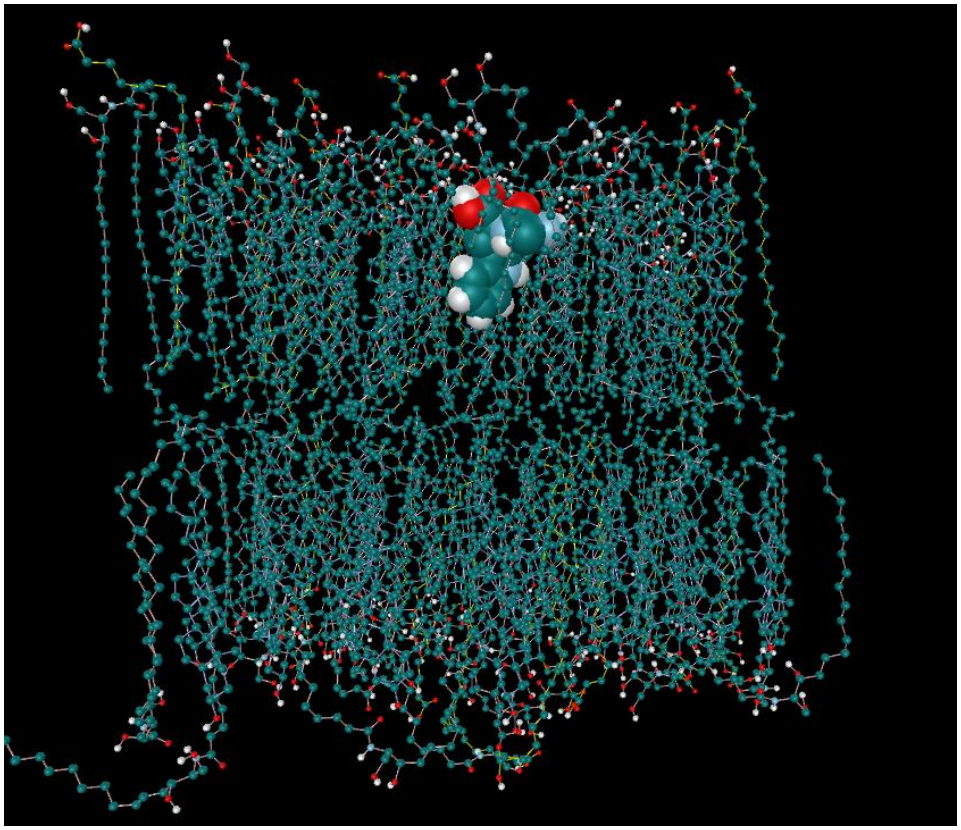


Figure 1. The dipeptide AlaTrp interacting with a lipid bilayer; created with a Gromacs pull simulation with a force of 250 and visualised with MDV.

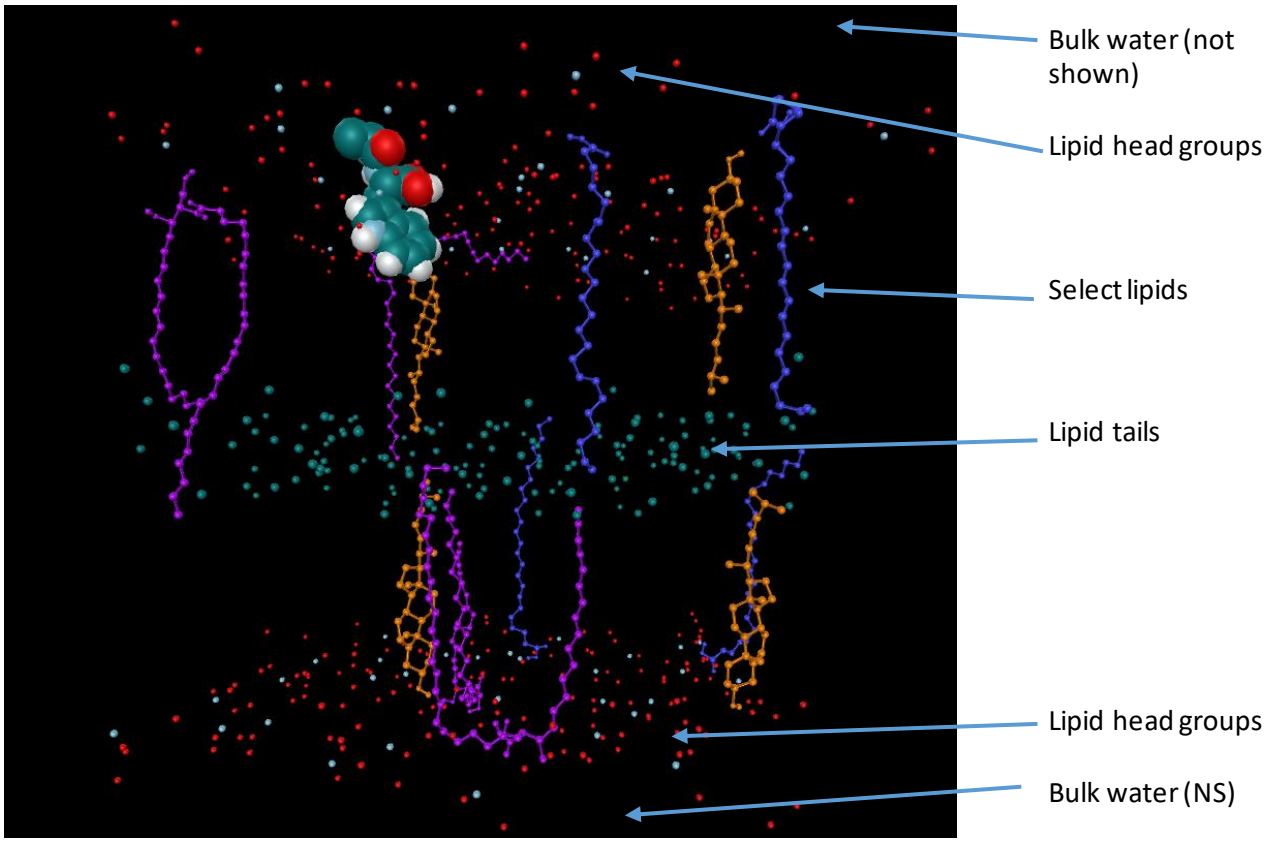


Figure 2. The dipeptide AlaTrp interacting with a lipid bilayer; created with a Gromacs US+REST. (Cholesterol: orange, ceramide: purple, free fatty acid: blue, lipid oxygen: red, lipid nitrogen: light blue, carbon: aqua blue, peptide: Van der Waals, water: not shown)
For videos see appendix 1e.

5. Discussion

Visualization of a US+REST simulation was significantly different to other MD simulations that have previously been presented at the HIVE. We were unable to replicate the smoothing described by Malajczuk C.J. (4) and various commands and modules in Gromacs were tested in order to find a way to adequately filter US+REST simulation trajectories. The most successful adjustment to the Malajczuk method was to skip the periodic boundary condition treatment and use the lipid bilayer as a reference structure (-fit) and filter each window individually. However, this did not work for all windows, resulting in a lipid jumps, separation of the bilayer, molecules moving outside of the simulation cell, and molecules forming unnatural bonds (Appendix 1e). All combinations of filtering options resulted in stretching of the structure of the peptide and misrepresented conformations of the peptide (Appendix 1e).

This smoothing issue was approached from two directions. The first was to run a pull simulation in order to represent the path of the peptide through the lipid bilayer in a visually appealing manner. This resulted in a trajectory that could be treated by the Malajczuk method without issues. The second approach was to remove large sections of the bilayer in order to reduce movement, and thus nausea, when visualizing in stereo. This had the added benefit of drawing the viewer's eye to the configurations of the peptide at different depths of the bilayer, which was a major point of interest for the project. With the visualizations produced by these two approaches, both expert and non-expert audiences can appreciate the scientific and entertainment value of MD simulations.

6. Conclusion

Two appealing stereoscopic visualisations with MDV of a peptide permeating across a SC lipid bilayer were produced. MD simulation trajectories came from two sources: a pull simulation post-processed with Gromacs using the Malajczuk method, and a more accurate US+REST simulation that required the refinement of a custom post processing workflow.

Further work should aim at creating general workflows that enable visualization of MD simulation trajectories using MDV in stereo at the HIVE Cylinder. While smoothing was possible for some windows it caused unsettling movement and, in some circumstances, molecules stretching or moving outside of the simulation cell. Further investigation of the cause of these issues could result in the ability to visualise individual windows directly. The concatenation of windows was not focused on in this report as the inability to smooth windows prevented further investigation. There may be issues with smoothing the transition of the concatenated US windows as the lipids had time to diffuse laterally within their independent simulations. These solutions will be likely found by creating or altering software outside of the Gromacs suite as the abilities of its commands and modules were largely exhausted during the course of this project.

7. Acknowledgements

This project was made possible by technical support from the Curtin HIVE Summer Internship Program (2019-20) and financial support from the Curtin Institute for Computation. The Curtin HIVE Summer Internship Program allows Curtin students to undertake a 10 week full-time internship to undertake a research project investigating the application of visualization technologies to a particular discipline area. Interns have regular access to the Curtin HIVE, where they are supported by the HIVE staff, and are supervised by a scientific discipline leader. The outcomes of the HIVE Summer Intern projects were presented at the HIVE Summer Intern Showcase held at the Curtin HIVE on Friday, 14 February 2020, and also presented in a written report (this report).

All MD simulations were undertaken using resources provided by the Pawsey Supercomputing Centre. The author would also like to acknowledge the support and guidance of Prof. Ricardo L. Mancera, Lanie Ruiz-Pérez, Chris Malajczuk and everyone associated with the Biomolecular Modelling Group at Curtin University.

Finally, the author would like to extend thanks to Dr Andrew Woods, Daniel Adams, Wesley Lamont, Carley Tillett, and all staff associated with the Curtin HIVE for facilitating the Curtin HIVE Summer Internship Program and providing generous support and encouragement over the course of the program.

8. References

1. Hansen HS, Hünenberger PH. Using the local elevation method to construct optimized umbrella sampling potentials: calculation of the relative free energies and interconversion barriers of glucopyranose ring conformers in water. Journal of computational chemistry. 2010 Jan 15;31(1):1-23.

2. Liu P, Kim B, Friesner RA, Berne BJ. Replica exchange with solute tempering: A method for sampling biological systems in explicit water. Proceedings of the National Academy of Sciences. 2005 Sep 27;102(39):13749-54.

3. Wiebrands M, Malajczuk CJ, Woods AJ, Rohl AL, Mancera RL. Molecular dynamics visualization (MDV): stereoscopic 3D display of biomolecular structure and interactions using the Unity game engine. Journal of integrative bioinformatics. 2018 Jun 1;15(2).

4. Malajczuk CJ. Visualisation of biomolecular and chemical phenomena in the Unity-based molecular graphics suite 'Molecular Dynamics Viewer.' Curtin HIVE. 2017

5. Humphrey W, Dalke A, Schulten K. VMD: visual molecular dynamics. Journal of molecular graphics. 1996 Feb 1;14(1):33-8.

6. Hess B, Kutzner C, Van Der Spoel D, Lindahl E. GROMACS 4: algorithms for highly efficient, load-balanced, and scalable molecular simulation. Journal of chemical theory and computation. 2008 Mar 11;4(3):435-47.

9. Appendices

Appendix 1.

A Github repository was created to host the appendices https://github.com/BMMG-Curtin/MDV_SCmembrane-AlaTrp_forHIVE. Alternatively these files can be sourced from the Curtin HIVE upon request. a) The optimized parameters for the pull simulation can be found on the Github repository as pull.mdp. b) The Python class made for editing structure files can be found as StructureFile.py, with a test file testfilein.gro. c) The final visualisation files for MDV can be found under the file 'Files'. d) The bash scripts that automate testing and creation of the final files can be found as pullautomation.sh and replexautomation.sh. e) Finally, videos made for the presentation can be found under 'videos'.