AMSI VACATIONRESEARCH SCHOLARSHIPS 2020–21

Get a Thirst for Research this Summer

Accuracy and Limits of Nodal Surface Approximations of the Lonsdaleite Minimal Surface

Monica Seeber

Supervised by Dr Gerd Schroeder-Turk

urdoch University

Vacation Research Scholarships are funded jointly by the Department of Education, Skills and Employment and the Australian Mathematical Sciences Institute.

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

Australian Government
Department of Education, Skills and Employment

Abstract

Bicontinuous minimal surfaces are a naturally occurring nanostructure across all kingdoms of life. They are smooth, infinite surfaces that partition space into two intertwined labyrinthine domains or channels. The Lonsdaleite surface, also known as hexagonal diamond or wurtzite, is believed to model some membranes in plant chloroplasts. In order to understand the formation and functioning of these membranes we must first identify and measure the exact shape and structure of the associated minimal surface. Nodal mesh and soap film models of the P surface were generated using Houdini and Surface Evolver, respectively. Measurements of the P surface taken from three different models and numerical data published by Jung, et al. (2006) and were analysed for convergence. While measured values for the channel volume fraction and interfacial area of the P surface converged well with published data, the convergence of measured curvature values was inconsistent across models. Using the process developed for the P surface, two different models of the Lonsdaleite surface were generated and their measurements were analysed for convergence. Measurements from the nodal mesh model resemble the general trend of numerical data calculated for other surfaces, however the nodal mesh and soap film models do not converge when the volume fractions of the channels are not equal. Time constraints prevented a full investigation into why the models do not converge, but it is hypothesised that the topological complexity of the Lonsdaleite surface increases the sensitivity of soap film models to computational processes. It is recommended that models of the Lonsdaleite surface are applied with caution unless multiple models validate results or until their accuracy can be confirmed using numerical data.

Table of Contents

Introduction

Bicontinuous minimal surfaces are smooth infinite surfaces that partition space into two intertwined labyrinthine domains, and a naturally occurring nanostructure across all kingdoms of life. They are useful for modelling a range of soft matter materials and are particularly successful in describing the structure of membranes in biological organisms (Han & Che, 2018). One such surface is the Lonsdaleite, believed to model some membranes in plant chloroplasts. Identifying the structure of membranes requires accurate models to generate 2D projections and scattering functions that can be compared to electron microscopy and x-ray diffraction data.

A commonly used model for bicontinuous and constant-mean-curvature (CMC) surfaces is the so-called nodal surface representation, an approximation to the exact geometry of these surfaces derived from crystallographic lattices. Surface families include a minimal surface and all CMC surfaces exhibiting the same symmetry and topology but with differing volume fractions. While measurements from the nodal surface approximation yield good results where the surface divides space into two channels of equal or nearly equal volume fraction (i.e. the minimal surface), models with non-equal channel volumes are not well studied and the convergence of these models with numerically calculated data has not been quantified.

The aim of this project is to investigate the accuracy of nodal surface models of the Lonsdaleite surface. A review of the literature indicates the analysis of nodal surface models of minimal surfaces is dominated by the P, D and G surfaces and numerically computed data is limited in general (Lord & Mackay, 2003; Jung, et al., 2006). Identifying research specific to the Lonsdaleite surface is complicated by inconsistent nomenclature across publications; it is variably referred to as Schwarz' hexagonal surface, Schwarz' H surface, H surface, hexagonal diamond, Schoen's GW surface, wurtzite, graphite-wurtzite and other names derived from minerals exhibiting the same symmetry as the Lonsdaleite. These names are not strictly interchangeable and depending on the context of the research, different names can indicate very different structures. As such the scope of the literature review was limited to publications of an explicitly geometric nature. Use of the name Lonsdaleite in this report refers to the geometric structure belonging to the $P6_3/mmc$ space group and observed as an allotrope of carbon.

The parametrisation of the Lonsdaleite structure was published in 1992 by Fogden and Hyde, and the related Minkowski functionals were published in 2012 by Michel, et al. However, numerically computed data specific to the Lonsdaleite was not identified in the broader literature. Experimental data related to the Lonsdaleite and wurtzite crystal lattices is available in the fields of biology and materials technology, however their relevance to the geometry of the minimal surface and its mathematical properties is limited (Gunning, 2001; McCulloch, et al., 2020). Unsurprisingly, the most relevant research in the literature included the project supervisor, Schröeder-Turk, as an author. Additional material such as lecture slides and workshop notes were provided by the supervisor. Numerically calculated data is available for families of P, D, G, I-WP, F-RD and C(P) surfaces, and is useful for establishing general trends in relationships between channel volumes, interfacial area and mean curvature of minimal and associated CMC surfaces.

Hence this project first analysed the degree to which measurements of channel volumes, area and curvature taken from nodal surface representations converged with numerically calculated data for minimal P surface family published by Jung, et al. (2006). Two types of nodal surface models were created: a level set mesh modal (created using Houdini) and a soap film model (created using Surface Evolver). Two versions of the soap film model were created and employed different constraints to generate model surfaces, one using prescribed channel volume fractions (the proportion of volume contained within a designated channel) and the other prescribed mean curvature. Having established the degree to which these models converged with the published data, equivalent models were created for the Lonsdaleite surface albeit only a single soap film model employing prescribed volume fraction was used in the final comparison of measurements from Lonsdaleite models.

Statement of authorship

The research topic was suggested by Schröeder-Turk with background research and literature review conducted by Seeber. Methodology was designed by Seeber in consultation with Schröeder-Turk. Fourier series approximations and initial Houdini files were provided by Hain; final Houdini models created by Seeber. Script code files for Surface Evolver were provided by Schröeder-Turk, Hain and the Surface Evolver webpage; final script code files were adapted by Seeber with assistance

troubleshooting from Schröeder-Turk and Hain. Measurement and analysis of models was conducted by Seeber and included data published by Jung, et al. (2006). This report was written by Seeber and the project was supervised by Schröeder-Turk.

Mathematical background

A minimal surface is an object in $\mathbb{R}^3$ with zero mean curvature at every point. The mean curvature at a given point on the surface is the average of the two principal normal curvatures (maximal and minimal) or, alternatively, the trace of the gradient of the unit normal vector at that point. In other words, a minimal surface is a three-dimensional object that is equally convex and concave at every point. Crystallographic space groups are used to describe the symmetric properties of minimal surfaces and corresponding families of CMC surfaces. Fourier transforms of crystallographic lattices can be used to generate implicit functions that closely approximate the exact parametrisation of a surface.

Triply periodic minimal surfaces (TPMS) are minimal surfaces that are infinite, embedded (non-self-intersecting), and symmetric in three principal directions. Both the P surface and Lonsdaleite are TPMS corresponding with the space groups P4/*mmm* and P6$_3$/*mmc* respectively. The P surface can be parametrised analytically using the Enneper-Weierstrass representation and complex integration, however its exact computation has only been publicly available since 2000 and the computational effort required to calculate the coordinates of multiple P family surfaces is prohibitive (Gandy & Klinowski, 2000). At the time of writing, the analytic parametrisation of the Lonsdaleite surface is not available. In lieu of analytic parameters, Fourier series approximations (also known as periodic nodal surface expansions) are implicit level set functions that generate minimal surfaces to an acceptable degree of accuracy (Jung, et al., 2006). Truncated implicit level set functions for the P and Lonsdaleite surfaces are given by:

P surface: $$\phi(x,y,z) = \cos(x) + \cos(y) + \cos(z)$$

Lonsdaleite: $$\begin{aligned}\phi(x,y,z) = {} & -0.36\cos(2z) + 0.22\cos(2y) + 0.23\sin(2y)\sin(z) \\ & + 0.45\cos(x)\cos(y) - 0.45\cos(x)\sin(y)\sin(z) \\ & + 0.21\cos(x)\cos(3y)\cos(2z)\end{aligned}$$

It has been established that the relationship between channel volume fraction and interfacial area for each family of surfaces visually resembles a negative parabola within boundary values of the volume fraction and with the maximum turning point occurring at or near 0.5. Boundary values are identified as the volume fraction at which the surface is no longer continuous or the exhibits symmetry and topology that is inconsistent with the minimal surface (Anderson, et al., 1990; Grosse-Brauckmannm, 1997; Jung, et al., 2006). Similarly, the relationship between channel volume fraction and mean curvature appears almost linear within boundary values. Beyond these values the curvature reverses direction and there appears to be multiple volume fractions corresponding to distinct mean curvature values.

Computer modelling

While Fourier series approximations may be easier to work with that analytic parametrisations of surfaces, numerical calculation of nodal representations is computationally intensive and time consuming. Jung, et al (2006) reported computational times of approximately 29 hours per surface, for a 200 × 200 × 200 mesh grid, calculated using custom algorithms (which they derived) deployed by specialist software run on Linux clusters with processing capacity equivalent to several personal computers. By comparison, modelling software can generate nodal surface models on a standard computer in a matter of minutes.

Modelling software usually employs one of two methods to construct surfaces: level set meshes defined by implicit functions, or soap films defined by skeletal graphs and space group symmetry.

Level set mesh models are created via an undirected process and are refined by adjusting the coarse grid of the mesh. They are called undirected because the software generates the model on a point-by-point basis rather than shaping it algorithmically to conform to predefined parameters. The mesh surface has three structural levels:

1. Points: the x, y, and z co-ordinates corresponding to intersections of mesh divisions at values calculated using the implicit function.
2. Primitives: discrete areas bounded by the lines of division and 3 or 4 corresponding points.

3. Details: global properties of the model, i.e. mean curvature.

The coarse grid refers to the number of divisions in a mesh, with more divisions creating more points on the surface model. The finer the grid, the more divisions in a mesh, the more points on the surface and the more precise the model. Level set mesh models are sensitive to small changes to the mesh such as changing the frequency of divisions or increasing/decreasing the total area of the mesh, and points generated on the boundaries of a mesh can have unpredictable effects on global surface properties such as mean curvature. Due to the nature of level set meshes, surface properties such as channel volume fractions are identified after the fact and often via trial and error, meaning measurements rarely correspond exactly to predetermined values of interest. However, meshing software tools are often straightforward to use and require only the implicit formula of a surface to begin modelling.

Soap film models are created by iteratively refining a virtual soap film that is attached to a virtual wire frame constructed from pre-determined vertices, edges and facets. The vertices and edges form the skeletal graph of the surface and their positions are determined from crystallographic data. Soap film models employ directed processes when refining the surface, the most common being the method of steepest descent. The method of steepest descent is analogous to descending a mountain slope:

> Imagine standing at the summit of a mountain and trying to determine the shortest route to the base. The steeper the path of descent, the shorter the distance to be traversed. The shortest path would be along a vertical drop from summit to base, unfortunately this is also likely to be the most deadly. Fortunately, a less deadly path leads down the side of the mountain, however the precise route is unknown. So to begin, a random path descending from the summit is chosen that may or may not be of the steepest descent. Along this path there are frequent junctions that can be used to change direction and follow a different path of steeper descent. After traversing several junctions, the base of the mountain is reached (and nobody died).

Typical soap film models employ algorithms that calculate the gradient of each edge present on the surface as a function of the positions of its bounding vertices. The vertices are then shifted to increase the gradient of attached edges, thus reducing

the length of those edges and the total area of the surface. After several iterations, the surface achieves the optimal state of minimised surface area. For a minimal surface, minimised surface area corresponds with zero mean curvature.

Unlike level set mesh models, the evolution of the soap film model is controlled entirely by the user who can perform operations that alter the surface between iterations, such as sub-dividing facets, switching edges and averaging vertices. The models can be very sensitive to the order in which these operations are performed, such that incorrect application of these operations can prevent the surface from reaching the optimal state or cause the model to fail entirely. The number of iterations required to reach the optimal state of the surface can number in the thousands, such that the process can resemble a “black box” and making it very difficult to diagnose faults in the process if the model fails.

Unfortunately, what is gained in efficiency by using computer models is lost in precision. While soap film and level set mesh models are generally assumed to be “good enough”, the degree to which they differ from numerically calculated nodal surfaces is largely unknown. Furthermore, while soap films and level set meshes may generate surface models that visually resemble each other, the properties of these models have not been systematically compared to determine the degree to which the models are mathematically similar.

For this project level set mesh models were generated using Houdini Apprentice, a 3D visualisation and animation package developed by SideFX. Soap film models were created using Surface Evolver, a gradient descent optimisation tool developed by Kenneth A. Brakke. Both programs are widely used for mathematical modelling and are supported by large collections of user-generated documentation such as the “Houdini tutorials tailored for mathematicians” (Anon., n.d.) and “The Surface Evolver” (Brakke, 2013) webpages. Both programs are available for free from their respective developers.

Methodology

Level set mesh models of minimal and CMC P surfaces were procedurally generated using built-in operators and networks for Houdini Apprentice:

IsoSurface	• define mplicit equation and level set • set boundary coordinates • set mesh divisions
Primitives	• stabilise surface primitives
Clean	• “tidy” the mesh
Polyfill / Bounding	• enclose the surface for volume measurement
Measure	• measure property of interest

Level set values corresponding to channel volume fractions specified in Jung et all were identified by encoding frame number as a variable within the implicit formula for the P surface, and processing the model as an animation over 500 frames, while monitoring channel volumes via the geometry spreadsheet pane. When the level set values were identified, the corresponding channel volume, interfacial area and mean curvature was measured and saved to an excel spreadsheet [see Appendix I: Houdini procedural network for P family surfaces].

Two soap film models were created using Surface Evolver: one constrained my channel volume and the other constrained by mean curvature. Parameters for volume fraction and mean curvature were set before loading the script code for each model. The surfaces were evolved through a series of built in operations and the core iteration function. When the surfaces reached the optimal state or could not be further evolved (due to computer processing limitations), measurements corresponding to channel volume,

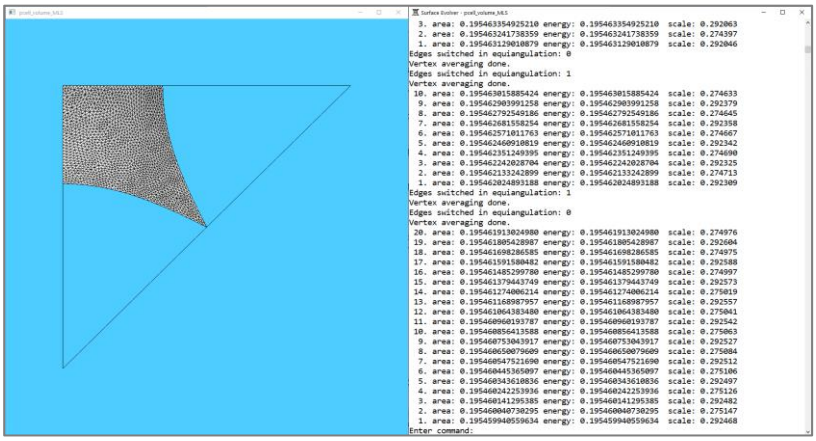


Figure 1: screen capture of Surface Evolver including the graphical display (left) and command line interface (right) for the P surface model of prescribed channel volume.

interfacial area and mean curvature were extracted to a text file and then saved to an excel spreadsheet. The measurements were tabulated and compared graphically using scatter plots. The difference between the measured values for each model and the data published by Jung et al was calculated. All models were subsequently adapted to improve the accuracy of interfacial area and mean curvature measurements [see Appendix II: Surface Evolver commands and scripts].

Channel volume, interfacial area and mean curvature were remeasured for each of the adapted models and saved to an excel spreadsheet. The new measurements were tabulated and scaled as appropriate, then compared graphically on new scatter plots to the published data. The absolute mean percentage difference (AMPD) between measurements from each of the models and the published data was calculated using the formula:

$$AMPD = |(model\ value - published\ value)\ \div published\ value|$$

The processes developed for the P surface models were applied to create level set mesh and a volume-constrained soap film models of the Lonsdaleite surface family.

As for the P surface, level set values for the Lonsdaleite surface and the corresponding volume fractions were identified by encoding the frame number as a variable in the implicit function [see Appendix I: Houdini procedural network for Lonsdaleite family surfaces].

The level sets corresponding to the boundary values for channel volume fraction were identified by observing when the animated model became discontinuous or the orientation of primitives spontaneously inverted. Channel volume, interfacial area and mean curvature were measured at regular intervals between these level sets and saved to an excel spreadsheet.

A single soap film model constrained by channel volume was created using Surface Evolver. Several processes for evolving the surface were trialed and the most consistent process applied in the final model. When the surfaces reached the optimal state or could not be further evolved (due to computer processing limitations), measurements corresponding to channel volume, interfacial area and mean curvature were extracted to a text file and then saved to an excel spreadsheet [see

Appendix II: Surface Evolver datafile for Wurtzite 6-Sided Mirror Patch with volume constraint].

The measurements were tabulated and compared graphically using scatter plots. The absolute mean percentage difference between measurements from each model was calculated. Visual trends in relationships between volume fraction and interfacial area, and channel volume fraction and mean curvature were compared with trends identified in other surfaces.

Findings

The P family

Measurements for interfacial area from the initial level set mesh and the prescribed volume soap film models converged well with the data published by Jung, et al. (2006), while those from the prescribed curvature soap film model followed the general trend at consistently higher values then the other models. The level set mesh models had an AMPD of 0.87% for both channel volume and interfacial area when compared to published data. The prescribed volume soap film model had an AMPD of 0.16% for channel volume and 0.26% for interfacial area. The prescribed curvature soap film model had AMPDs of 5.76% and 7.99% for channel volume and interfacial area respectively.

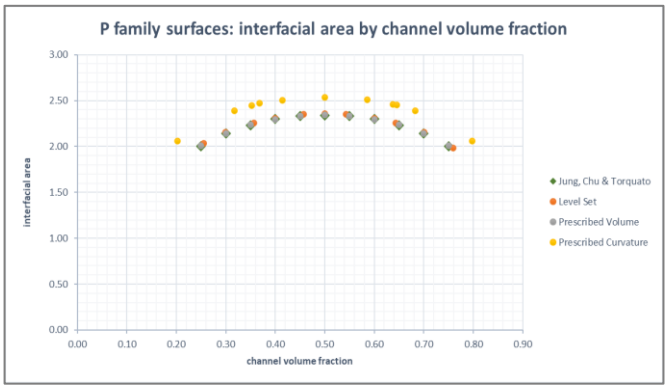


Figure 2: scatterplot of channel volume and interfacial area measurements from initial models and published data.

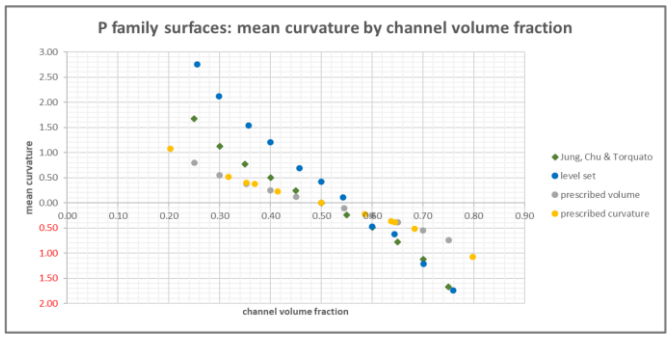


Figure 3: scatterplot of channel volume and mean curvature measurements from initial models and published data.

Measurements for mean curvature from the initial prescribed volume and prescribed curvature soap film models converged well together however convergence with published data was poor for all models, with AMPD values for mean curvature equal

to 37.75% for the level set mesh model, 46.74% for the prescribed volume soap film model and 30.83% for the prescribed curvature soap film model.

The models were adapted to improve convergence with the published data and remeasured. Mean curvature values were scaled by a factor of 0.5 for level set mesh models and a factor of 2 for soap film models. The second set of measurements converged much more closely with published data, however the convergence of interfacial area measurements for the prescribed curvature soap film model with published data is inconsistent. The AMPD values for the final models were calculated as follows:

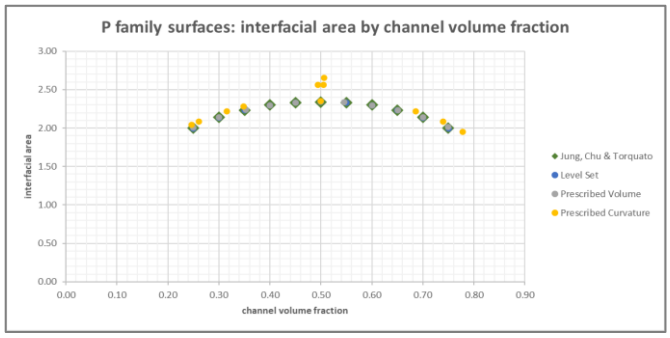


Figure 4: scatterplot of channel volume and interfacial area measurements from adapted models and published data.

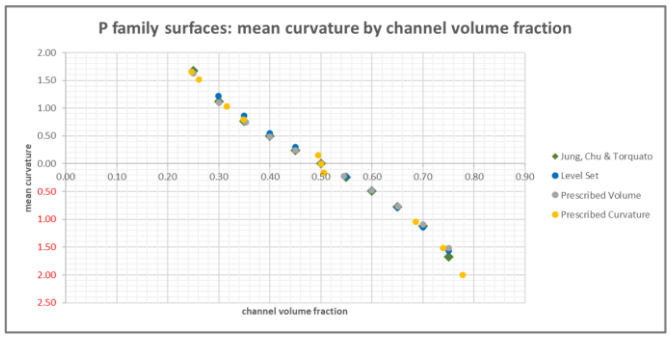


Figure 5: scatterplot of channel volume and mean curvature measurements from adapted models and published data.

Table 1: AMPD between published data and computer models:

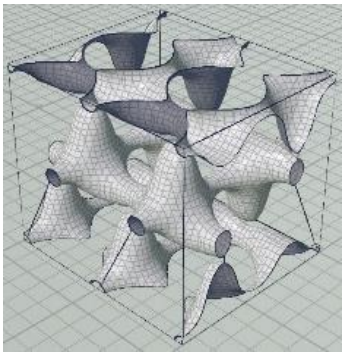
	Channel volume	Interfacial area	Mean curvature
Level set mesh	0.04133922	0.15778787	6.4266327
Prescribed volume soap film	0.15995098	0.27141608	3.2563407
Prescribed curvature soap film	7.75848841	4.32757731	32.2045943

The Lonsdaleite family

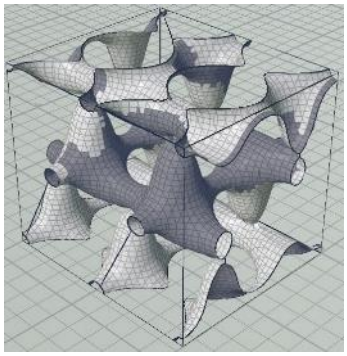
Measurements collected from the level set mesh and prescribed volume soap film models of the Lonsdaleite family surfaces do not converge, and the measurements of the prescribed volume soap film model do not appear to approximate the general trends exhibited by other surface families [see Appendix III: Measured values for the prescribed channel volume fraction soap film model for Lonsdaleite family surfaces]. The boundary values for Lonsdaleite channel volume fraction were identified by

observing the spontaneous inversion of surface primitives of the level set mesh model.

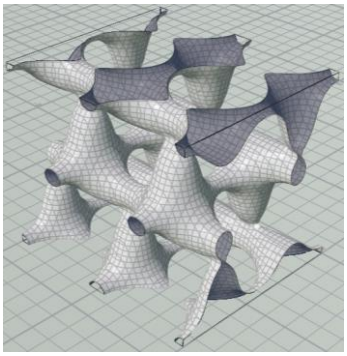
Inversion of primitives occurred at level sets +0.4 and -0.4, corresponding with measured channel volume fractions of 0.78 for the upper boundary and 0.23 for the lower boundary. At level sets beyond these values the surface is no longer continuous and the measured channel volume fractions are highly unstable, with some level set values returning channel volume fractions greater than 1. For values between the volume fraction boundaries, the data from the level set mesh model exhibits similar trends in relationship between channel volume fractions, interfacial area and mean curvature that is exhibited by other minimal and CMC surface families.



Frame no. 68:
white indicates the exterior of surface primitives; vertical edges of the bounding box are visible.



Frame no. 69:
groups of primitives have inverted.



Frame no. 70:
orientation of primitives have stabilised; vertical edges of the bounding box are no longer visible.

Figure 2: identification of channel volume boundary values by observing spontaneous inversion of surface primitives

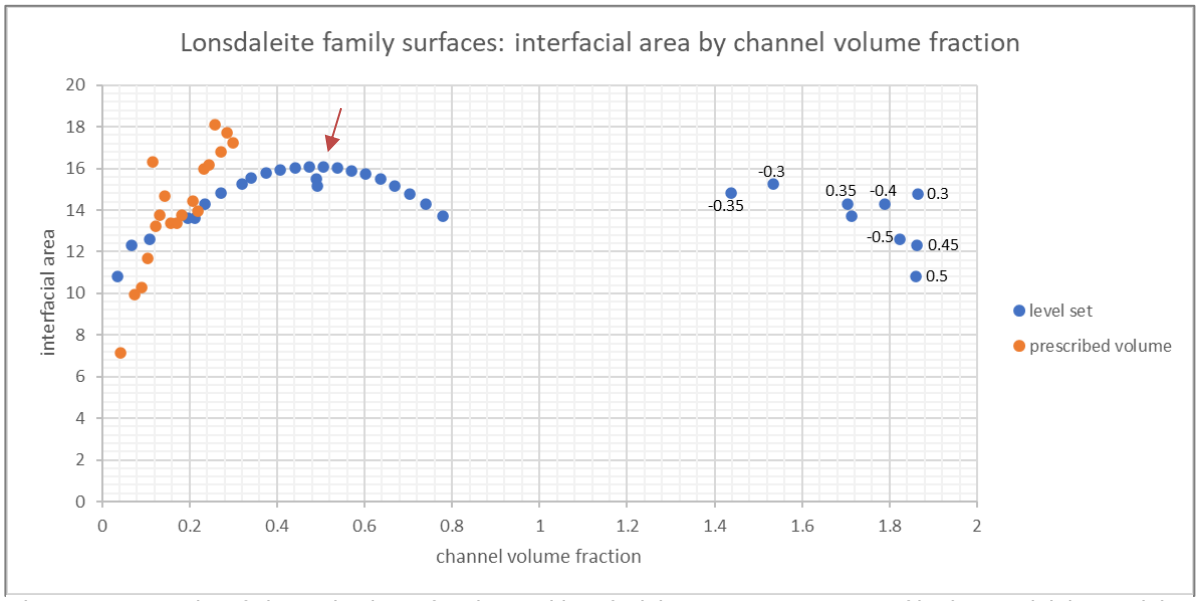


Figure 7: scatterplot of channel volume fraction and interfacial area measurements of both Lonsdaleite models. Points with volume fractions greater than 1 are labeled with the corresponding level set value. The level set measurements can be seen forming a parabolic arc between the volume fraction boundaries with the 0.5 channel volume fraction acting as the maximal turning point (red arrow).

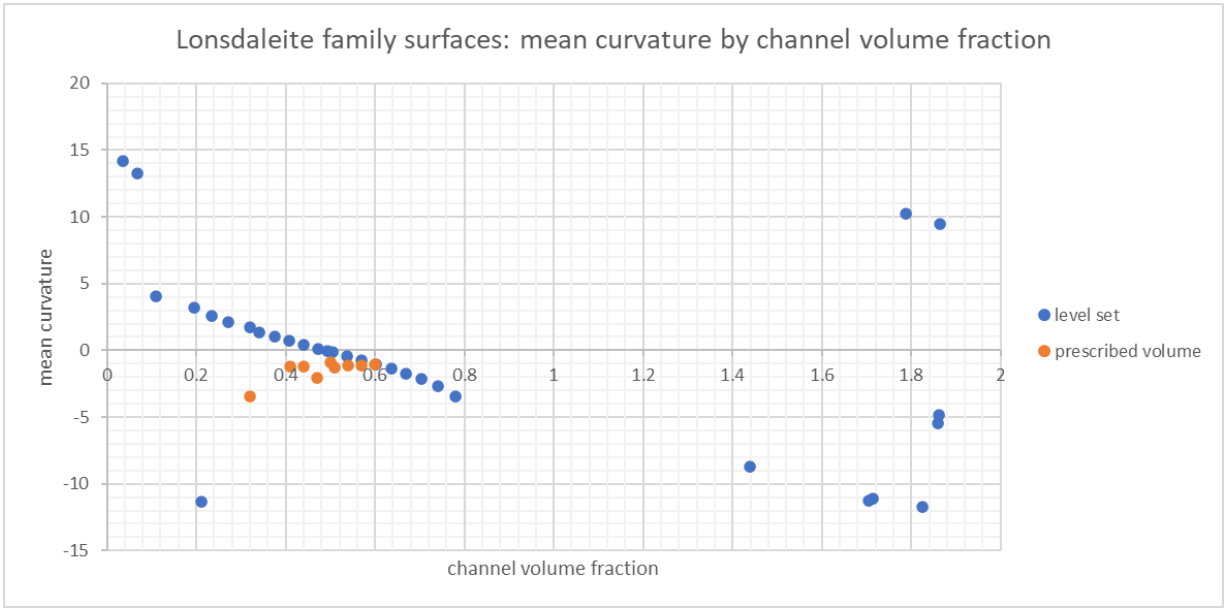


Figure 8: scatterplot of channel volume fraction and mean curvature measurements for both Lonsdaleite models. Measurements from the prescribed volume fraction soap film model outside the values of 0.41 and 0.6 have been excluded due to extreme values [See Appendix III: Measured values for the prescribed channel volume fraction soap film model for Lonsdaleite family surfaces].

Discussion

Computer models are powerful resources and can provide a wealth of information that would be otherwise unattainable. However, models are only as good as the data and methods used to create them. While this project was conceived as opportunity to validate the methods and models used in mathematical research, it transformed into an exercise of brute force code debugging as model after model failed to produce the results for which it was created. It was a frustrating and exhausting process, much of which could have been avoided if not for two problems:

1. Most papers don't include the code scripts or specific details of the model constructions that generated the data used in the research; as such it is extremely difficult to replicate methods or results.
2. When code scripts or model constructions are published with the research, they are often lacking annotation and thus while the "what" of a modelling process made be clear, the "how" and "why" often is not. The code lacks context and as such it also lacks meaning.

Broken models and misbehaving code aside, this project did achieve what it set out to do: analyse the accuracy of models of the Lonsdaleite family of surfaces and investigate the limitations of those models. Of the models created for the Lonsdaleite surfaces, only the level set mesh model implemented using Houdini appeared to produce reliable data. The accuracy of the volume constrained soap film model is severely compromised and is a prime example of the most significant limitation of computer modelling of this nature: it is extremely difficult to create models that are stable and produce data of consistent quality.

Soap film models created using Surface Evolver are an excellent example of this problem: soap film models can be extremely accurate and demonstrate high degrees of convergence with known quantities (Lord & Mackay, 2003; Mancini, et al., 2015). However they are also extremely sensitive to the order in which evolution operations and iterations are conducted. Moreover, processes that generate good results for one model, may generate disasterous results for another.

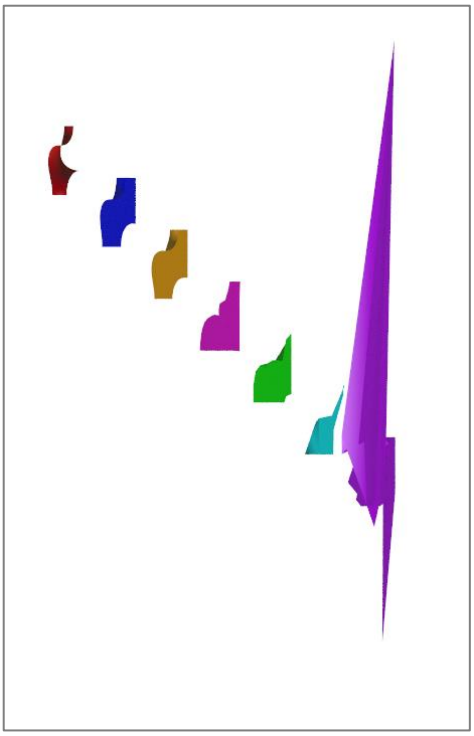


Figure 3: comparison of degenerate models of the Lonsdaleite surface. The red surface in the top left is the ideal case with each model degenerating to a greater degree until the extreme case on the far right (purple).

Figure 9 displays six failed soap film models for the Lonsdaleite surface family alongside the final model used for this project (top left, red). The measurements for the functional model did not converge with either the measurements from the level set mesh model or general trends in geometries observed in other minimal and CMC surface families. This is in part because limited time was available to work with the model and optimise the evolution process. However, time was limited because the procedures developed for the P family soap film models did not work for the Lonsdaleite family surfaces and several models were abandoned after considerable effort was spent trying to identify why the evolution process was producing degenerate surfaces.

In contrast, the level set mesh model produced consistent results to a high degree of accuracy for the P family surfaces, and at least an acceptable degree of accuracy for the Lonsdaleite family surfaces given the model data appears to exhibit the same trends in surface measurements that are observed in more closely studied surface families. The identification of the upper and lower boundary values for channel volume fraction is encouraging, even if the generation of impossible channel volume fractions outside those boundaries is not. It is suspected that the impossible values are the unexpected consequence of the procedural nature of Houdini itself. Likewise, it is thought that the lower degree of accuracy measuring mean curvature compared to interfacial area may be related to the mathematical process employed by the software. Unfortunately, the mathematics of Houdini is not explicitly defined in the user documentation and reverse-engineering the software is beyond the scope of this project.

Conclusion

Computer models are not a sufficient replacement for numerical approximations of minimal or CMC periodic surfaces when a significant degree of accuracy required. They can provide useful “starting point” however measurements and other data should be applied with caution unless it can be validated by additional models or calculated data.

Acknowledgements

This project could not have been completed without the assistance of many generous people supporting me personally, academically and with copious amounts of coffee.

I would like to thank Tobias Hain, PhD student, for his assistance with programming and script codes and troubleshooting Houdini when it wasn’t doing what I wanted.

I am extremely grateful to Dr Gerd Schröeder-Turk for supervising this project and generally being a fantastic teacher and mentor for the duration of my undergraduate degree. It has been a genuine pleasure working with you and learning more about your research. I hope the future brings you many enthusiastic students and well-funded research grants.

Finally, I would like to acknowledge that I live and study on Wadjuk Noongar Boodjar and it is my privilege to call it home. I acknowledge the Wadjak people as the traditional owners of this land and pay my respects to their Elders, past, present and emerging.

Sovereignty was never ceded.

References

Anderson, D. M., Davis, H. T., Scriven, L. E. & Nitsche, J. C. C., 1990. Periodic Surfaces of Prescribed Mean Curvature. In: S. A. Rice & I. Prigogine, eds. *Advances in Chemical Physics.* s.l.:John Wiley & Sons, Inc., pp. 337-396.

Anon., n.d.. *Houdini Tutorials Tailored for Mathematicians.* [Online] Available at: http://wordpress.discretization.de/houdini/ [Accessed 2021].

Bitsche, R., 2005. *Space-filling polyhedra as mechanical models for solidified dry foams.* s.l.:Institut für Leichtbau und Struktur-Biomechanik.

Brakke, K., 2013. *The Surface Evolver.* [Online] Available at: http://facstaff.susqu.edu/brakke/evolver/html/evolver.htm [Accessed 2021].

Brakke, K., n.d.. *Triply Periodic Minimal Surfaces.* [Online] Available at: http://facstaff.susqu.edu/brakke/evolver/examples/periodic/periodic.html [Accessed 2021].

Gandy, P. J. F., Bardhan, S., Mackay, A. L. & Klinowski, J., 2001. Nodal surface approximations to the P, G, D and I-WP triply periodic minimal surfaces. *Chemical Physics Letters,* March, 336(3-4), pp. 187-195.

Gandy, P. J. F. & Klinowski, J., 2000. Exact computation of the triply periodic Schwarz P minimal surface. *Chemical Physics Letters,* June, 322(6), pp. 579-586.

Grosse-Brauckmannm, K., 1997. On gyroid interfaces. *Journal of Colloid and Interface Science,* March, 187(2), pp. 418-428.

Gunning, B. E. S., 1965. The greening process in plastids. *Protoplasma,* Volume 60, pp. 111-130.

Gunning, B. E. S., 2001. Membrane geometry of "open" prolamellar bodies. *Protoplasma,* Volume 215, pp. 4-15.

Han, L. & Che, S., 2018. An Overview of Materials with Triply Periodic Minimal Surfaces and Related Geometry: From Biological Structures to Self-Assembled Systems.. *Advanced Materials,* 30(17).

Hyde, S. T. & Schröder-Turk, G. E., 2012. Geometry of interfaces: topological complexity in biology and materials. *Interface Focus,* August, Volume 2, pp. 529-538.

Jung, Y., Chu, K. T. & Torquato, S., 2006. A Variational Level Set Approach for Surface Area Minimization of Triply Periodic Surfaces. *Journal of Computational Physics,* May, 233(2), pp. 711-730.

Lord, E. A. & Mackay, A. L., 2003. Periodic minimal surfaces of cubic symmetry. *Current Science,* 85(3), pp. 346-362.

Mancini, M., Guene, E. M., Lambert, J. & Delannay, R., 2015. Using Surface Evolver to measure pressure and energies of real 2D foams submitted to quasi-static deformations. *Coloids and Surfaces A: Physiochemical and Engineering Aspects,* Volume 468, pp. 193-200.

McCulloch, D. G. et al., 2020. Investigation of Room Temperature Formation of the ultra-hard nanocarbons diamond and Lonsdaleite. *Small,* Volume 16.

Mickel, W., Schröder-Turk, G. E. & Mecke, K., 2012. Tensorial Minkowski functionals of triply periodic minimal surfaces. *Interface Focus,* June, Volume 2, pp. 623-633.

Schröder-Turk, G., 2017. *Representations of triply-periodic minimal surfaces.* Copenhagen: The Niels Bohr Institute.

Shewchuk, J. R., 1994. *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain,* Schenley Park Pittsburgh: Carnegie Mellon University.

Appendix I: Houdini procedural networks

Houdini procedural network for P family surfaces

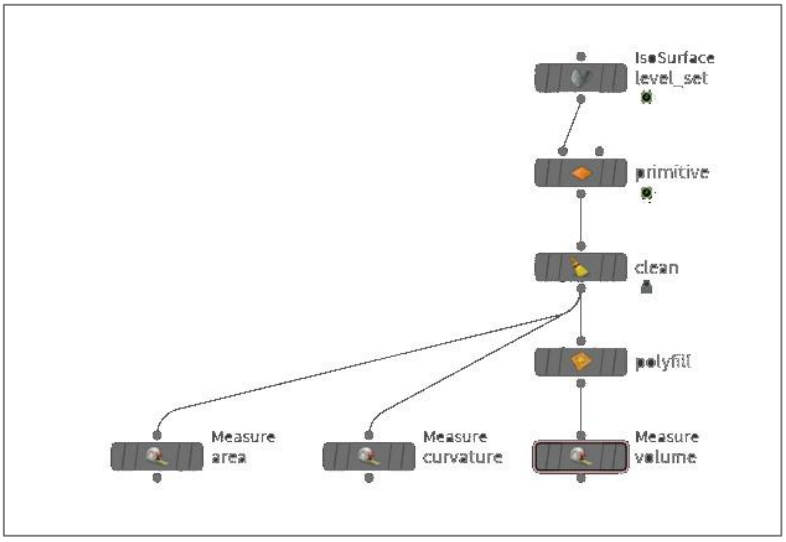


Figure 4: screen capture of Houdini procedural network for P family surfaces.

Houdini procedural network for Lonsdaleite family surfaces

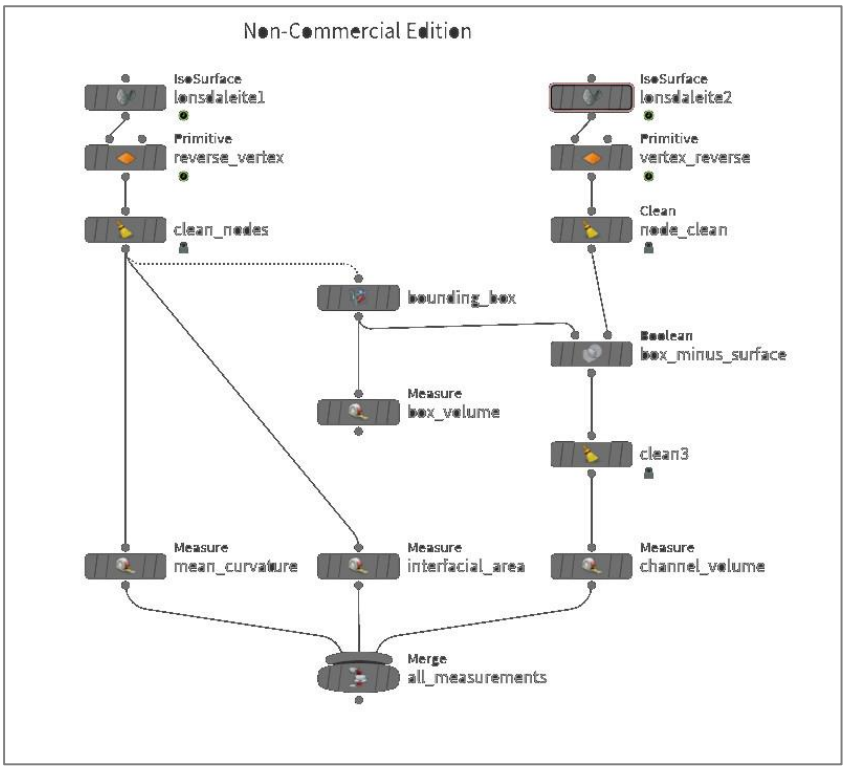


Figure 5: screen capture of Houdini procedural network for Lonsdaleite family surfaces.

Appendix II: Surface Evolver commands and scripts

Common Surface Evolver commands

(excerpt from "Surface Evolver Manual" available from http://facstaff.susqu.edu/brakke/evolver/evolver.html)

g	Go one iteration step. Often followed by a repetition count.	Each iteration calculates the force on each vertex due to its contribution to the total energy and moves the vertex by a multiple of the force. There is a global scale factor that multiplies the force to give the displacement. If area normaliztion is turned on, the force at each vertex is also divided by the total area of facets adjacent to the vertex to better approximate motion by mean curvature (but this seems to be often numerically badly behaved due to long skinny facets). If any bodies have prescribed volumes, the vertices are also displaced to bring the volumes back to near the prescribed values
r	Refine triangulation	Edges are divided in two, and SOAPFILM facets are divided into four facets with inherited attributes. Edges and facets with the *no_refine* attribute set are not refined. Reports the number of structures and amount of memory used.
u	Equiangulate	In the soapfilm model, each edge that has two neighboring facets (and hence is the diagonal of a quadrilateral) is tested to see if switching the quadrilateral diagonal would make the triangles more equiangular.
V	Vertex averaging	For each vertex, computes new position as area-weighted average of centroids of adjacent facets. Only adjacent facets with the same constraints and boundaries are used. Preserves volumes, at least to first order. See the *rawv* command below for vertex averaging without volume preservation, and *rawestv* for ignoring likeness of constraints. Does not move vertices on triple edges or fixed vertices. Vertices on constraints are projected back onto their constraints. All new positions are calculated before moving.

Surface Evolver datafile for fundamental cell for Schwartz' P minimal surface with prescribed volume fraction

```
//===========================================================================
// pcell_volume_MLS.fe adapted from pcell.fe
// Fundamental region is a tetrahedron 1/48 of a cube
// Programmer: Ken Brakke, brakke@susqu.edu, http://www.susqu.edu
// Adapted by Monica Seeber, monica@seeber.id.au, 2021
//===========================================================================

// ********************************************************************
// P surface contraints

constraint 1            // top
function:  z = 1
content:   c1: 0.5*y
           c2: -0.5*x
           c3: 0

constraint 2            // right side
function:  y - x = 0

constraint 3            // left  side
function:  -y = 0

constraint 4            // front bottom
function:  x - z = 0
content:   c1: x*y /3
           c2: -x^2/3
           c3: 0

// ********************************************************************
// symmetry defining P surface

vertices

1  0.5 0   1    constraints 1 3
2  0.5 0.5 1    constraints 1 2
3  0.5 0.5 0.5  constraints 2 4
4  0.5 0   0.5  constraints 3 4
5  0   0   0    fixed  bare  // for outlining tetrahedron
6  0   0   1    fixed  bare
7  1   0   1    fixed bare
8  1   1   1    fixed  bare

edges

1   2 1 constraint 1
2   3 2 constraint 2
3   4 3 constraint 4
4   1 4 constraint 3
5   5 6 fixed bare // for tetrahedron display
6   5 7 fixed bare
7   5 8 fixed bare
8   6 7 fixed bare
9   6 8 fixed bare
10  7 8 fixed bare

facets

1  4 3 2 1 tension 1

bodies

1  1 volume 1/12 // specify volume for prescribed volume fraction, 1/12 = minimal surface

read

// ********************************************************************
// script routines

// general settings

hessian_normal on    /* only normal motion for hessian */
band_flag : = 0      /* whether banding has been done */

// read file for explicit calculation of curvature data
read "CurvatureRoutines_MLS.fe";

// print data to screen

print_stats := {
            mean_curv_min_max_int_av_std;
            printf "vol   = %5.10e \n", body[1].volume;
```

```
        printf "area  = %5.10e \n", sum(facet, area);
        printf "curve = %5.10e \n", mcurv_int/mcurv_area;
        printf "vert  = %5.10e \n", avg(vertex where not fixed, mean_curvature);
        printf "press = %5.10e \n", body[1].pressure;
        };
// ****************************************************************
// evolution scripts
// Mesh grooming. Use in evolution scripts to keep a nice mesh.
groom_length := 0.08;
groom := {
        refine edge where length > groom_length;
        u;
        V;
        delete edge where length < groom_length/5;
        u;
        V;
        };
// Make sure "r" cuts the groom_length
r :::= { 'r'; groom_length /= 2};


// evolution script - short
gg : = {g 10; groom;};
// evolution script - standard
gogo : = {
        u;
        u;
        gg;
        r;
        gg 5;
        r;
        gg 10;
        g 20;  // final evolution without doing vertex averaging, to let the contact lines settle down.
        };
//================================================================================
```

Surface Evolver datafile for fundamental cell for Schwartz' P minimal surface with prescribed mean curvature

```
//===========================================================================
// pcell_curve_MLS.fe adapted from pcell.fe
// Fundamental region is a tetrahedron, 1/48 of a cube.
// Programmer: Ken Brakke, brakke@susqu.edu, http://www.susqu.edu
// Adapted by Monica Seeber, monica@seeber.id.au, 2021
//===========================================================================

// ******************************************************************
// P surface contraints

constraint 1            // top
function:  z = 1
content:   c1: 0.5*y
           c2: -0.5*x
           c3: 0

constraint 2            // right side
function:  y - x = 0

constraint 3            // left  side
function:  -y = 0

constraint 4            // front bottom
function:  x - z = 0
content:   c1: x*y /3
           c2: -x^2/3
           c3: 0

// method required to constrain mean curvature
// set to pi scalar of prescribed curvature - larger scalar forces faster convergence
// set to "energy" to constrain curvature; set to "info_only" for measurement only
// see “Surface Evolver exercises” created by Gerd Schröeder-Turk for the RMIT Summer School 12-16 August 2014

PARAMETER h_zero:= -2*PI*1.67 // 0 = minimal surface

quantity sqmean energy method star_perp_sq_mean_curvature global

// ******************************************************************
// symmetry defining P surface

vertices

1  0.5 0   1    constraints 1 3
2  0.5 0.5 1    constraints 1 2
3  0.5 0.5 0.5  constraints 2 4
4  0.5 0   0.5  constraints 3 4
5  0   0   0    fixed  bare  // for outlining tetrahedron
6  0   0   1    fixed  bare
8  1   1   1    fixed  bare

edges

1   2 1 constraint 1
2   3 2 constraint 2
3   4 3 constraint 4
4   1 4 constraint 3
5   5 6 fixed bare // for tetrahedron display
6   5 7 fixed bare
7   5 8 fixed bare
8   6 7 fixed bare
9   6 8 fixed bare
10  7 8 fixed bare

facets

1  4 3 2 1

bodies

1  1

read

conj_grad on

hessian_normal on              // only normal motion for hessian

normal_curvature on            // how mean curvature is defined , see section 7.3.9 Surface Evolver Manual

check_increase on              // switch off at times !

default_area_modulus : = 1     // area is switched on , but tension 0
```

```
set facet tension 0                    // means area is info quantity only
unset body target                      // volume not a constraint, info only
ysmp off                               // to avoid crashes while computing hessian
// ********************************************************************
// script routines
// Mesh grooming.  Use in evolution scripts to keep a nice mesh.
groom_length := 0.08;
groom := {
        refine edge where length > groom_length;
        u; V;
        delete edge where length < groom_length/5;
        u; V;
        };
// Make sure "r" cuts the groom_length
r :::= { 'r'; groom_length /= 2}
// evolution script - short
gg := {groom 5; g 50; groom};
// evolution script - standard
gogo := {
        gg;
        g 750;
        gg;
        r;
        g 500;
        groom 5;
        r;
        gg;
        g 100;      // final evolution without doing vertex averaging, to let the contact lines settle down.
        print_stats;
        };
// automated process to identify best refine points when curvature is constrained – prone to maxing out processing capacity
refine_point := {
quantity farea energy method facet_area global        // method required to constrain facet area
set facet_area          // set quantity to surface area corresponding with expected volume fraction
unset body target       // volume not a constraint, info only
set facet tension 1     // converts area to named quantity
target_area := 1        // create new quantity for surface area; set quantity to surface area corresponding with prescribed curvature
IF (sum(facet, area) > target_area) THEN {
        r; g; WHILE (sum(facet, area) > target_area) DO {
                g10; groom; IF (sum(facet, area) > target_area) THEN {
                        r;
                        }
                }
        }
ELSE print_stats;
};
//====================================================================================
```

Surface Evolver datafile for Wurtzite 6-Sided Mirror Patch with volume constraint

```
//================================================================================
// Wurtzite_6SidedMirrorPatch_VolConstrAreaMin2_MLS.fe adapted from Wurtzite_6SidedMirrorPatch_VolConstrAreaMin2.fe
// Programmer: Gerd Schroder-Turk, http://gerdschroeder-turk.org
// adapted by Monica Seeber, monica@seeber.id.au, 2021
//================================================================================
// ********************************************************************
// crystallographic lattice parameter a of space group of the Wurtzite structure, P6(3)/mmc
PARAMETER crysta =  1.6331;
// crystallographic lattice parameter c of space group of the Wurtzite structure, P6(3)/mmc
PARAMETER crystc =  2.6667;
// target volume fraction
// UPDATE BEFORE LOADING FILE specify prescribed volume fraction
PARAMETER target_phi = 0.32 // 0.5 = minimal surface
// total volume of bounding prism
PARAMETER prismvolume = sqrt(3)/4*(crysta/sqrt(3))**2;
// target volume
PARAMETER target_volume = target_phi*prismvolume; // 1/48/2 = 0.010416666666666666 = minimal surface
// ********************************************************************
//wurtzite/lonsdaleite surface constraints
constraint 1            // green - horizontal mirror xz plane
function:  X2  = 0
constraint 2            // yellow - horizontal mirror
function: X2 = sqrt(3)*X1   // rotated through origin
constraint 3            // red - horizontal mirror not through origin
function: X2 = (crysta/sqrt(3)-X1)*sqrt(3)
constraint 4            // gray – bottom
function: X3 = -1/4*crystc
constraint 5            // black – top
function: X3 = 1/4*crystc
// default curvature method results in deformed surface - must specify alternate method to ensure surface integrity
// set to "energy" to constrain curvature; set to "info_only" for measurement only
// see “Surface Evolver exercises” created by Gerd Schröeder-Turk for the RMIT Summer School 12-16 August 2014
quantity sqmean info_only method star_perp_sq_mean_curvature global
// method required to compute the Gauss curvature distribution
quantity total_deficit info_only method star_gauss_curvature global
// ********************************************************************
// symmetry defining wurtzite/lonsdaleite surface
vertices
// vertices of patch to be minimised
1 0.23570 0.40825  crystc/4 constraint 2 5
2 0.70711 0.40825  crystc/4 constraint 3 5
3 0.94281 0.00000  crystc/8 constraint 3 1
4 0.47141 0.00000 -crystc/4 constraint 1 4
5 0.70711 0.40825 -crystc/4 constraint 4 3
6 0.47141 0.81650 -crystc/8 constraint 2 3
//vertices for  volume computation
7  0.94281 0.00000 -crystc/4 fixed
8  0.47141 0.81650  crystc/4 fixed
// vertices for asymmetric frame - display only
9  0.00000 0.00000 -crystc/4 fixed
10 0.94281 0.00000 -crystc/4 fixed
11 0.47141 0.81650 -crystc/4 fixed
12 0.00000 0.00000  crystc/4 fixed
13 0.94281 0.00000  crystc/4 fixed
14 0.47141 0.81650  crystc/4 fixed
// vertices for graph - display only
```

```
15 0.94281 0.00000 -0.16667 fixed
16 0.94281 0.00000 -crystc/4 fixed
17 0.47141 0.81650  0.16667 fixed
18 0.47141 0.81650  crystc/4 fixed
19 0.70711 0.40825  0.00000 fixed

edges

// edges defining the patch

1 1 2 constraint 5 color blue
2 2 3 constraint 3 color blue
3 3 4 constraint 1 color blue
4 4 5 constraint 4 color blue
5 5 6 constraint 3 color blue
6 6 1 constraint 2 color blue

// edges for asymmetric frame - display only

9  9  10 bare color black fixed
10 10 11 bare color black fixed
11 11  9 bare color black fixed
12 12 13 bare color black fixed
13 13 14 bare color black fixed
14 14 12 bare color black fixed
15  9 12 bare color black fixed
16 10 13 bare color black fixed
17 11 14 bare color black fixed

// graph edges - display only

18 16 15 bare color green fixed
19 15 19 bare color green fixed
20 19 17 bare color green fixed
21 17 18 bare color green fixed

// edges for body facets with tension 0

22 1 8 constraint 5 2 color gray no_refine
23 8 2 constraint 5 3 color gray no_refine
24 8 6 constraint 3 2 color gray no_refine
25 5 7 constraint 4 3 color gray no_refine
26 7 3 constraint 1 3 color gray no_refine
27 4 7 constraint 4 1 color gray no_refine

facets

1 1 2 3 4 5 6 color blue tension 1
2 22 23 -1 constraint 5 color black tension 0.
3 -23 24 -5 25 26 -2 constraint 3 color red tension 0.
4 -4 27 -25 constraint 4 color gray tension 0.
5 -26 -27 -3 constraint 1 color green tension 0.
6 -24 -22 -6 constraint 2 color yellow tension 0.

bodies

1 -1 -2 -3 -4 -5 -6

read

//**********************************************************************
// script routines */

set body target target_volume;  // set volume constraint

// remove body to measure surface quantities

removeVolume := {
                        currentVolume := body[1].volume;
                        dissolve facets where color != blue;
                        dissolve vertices;
                        dissolve body;
                        dissolve edges;
                        dissolve vertices;
                        }

// create empty histogram for curvature data

gauss_curv_histogram_nbins := 30;

// read file for explicit calculation of curvature data – available from Schroder-Turk

read "./Wurtzite_6SidedMirrorPatch_CurvatureRoutines.fe";
```

```
// ********************************************************************
// evolution scripts
// Mesh grooming.  Use in evolution scripts to keep a nice mesh.
groom_length := 0.08;
groom := {
        refine edge where length > groom_length;
        u;
        V;
        delete edge where length < groom_length/5;
        u;
        V;
        }
// Make sure "r" cuts the groom_length
r :::= { 'r'; groom_length /= 2}
// RUN FIRST forces facets to divide away from central vertex
gogoinit := {
        foreach facet ff where color == blue DO {
                refine ff;
                };
        g 100;
        foreach edge ee where color == blue DO {
                IF (ee on_constraint 1 or ee on_constraint 2 or ee on_constraint 3 or  ee on_constraint 4 or  ee on_constraint 5 ) THEN {
                refine ee;
                        };
                }
        g 100;
        foreach edge ee where color == blue DO {
                IF (ee on_constraint 1 or ee on_constraint 2 or ee on_constraint 3 or  ee on_constraint 4 or  ee on_constraint 5 ) THEN {
                        refine ee;
                        };
                }
        g 100;
        u;
        g 100;
        foreach edge ee where color == blue DO {
                IF (ee on_constraint 1 or ee on_constraint 2 or ee on_constraint 3 or  ee on_constraint 4 or ee on_constraint 5) THEN {
                        refine ee;
                        };
                }
        g 100;
        foreach facet ff where color == blue DO {
                refine ff;
                };
        g 500;
        u;
        g 250;
        }
// ********************************************************************
// routines to export measurements to .dat file
// create empty file to print data to
make_gauss_histo_filename := {
                        mean_curv_min_max_int_av_std;
                        gauss_curv_histogram_filename := sprintf
                        "Wurtzite_6SidedMirrorPatch_VolConstrAreaMin_GaussCurvHistogram_h%5.3f_a%5.3f_v%5.4f_phi%5.4f_nt%d_ca%3.2f_cc%3.2f_dh%4.2e.dat",
                        curv_av, total_area, currentVolume, (currentVolume/prismvolume), facet_count, crysta, crystc, mcurv_std;
                        }
// print data to file
print_brief_curv_stats:= {
                        mean_curv_min_max_int_av_std;
                        lfilename:= sprintf "Wurtzite_VolConstrAreaMin_curvatureStats_MLS_VF%5.4f_.dat", target_phi;
                        logfile off;
                        logfile lfilename;
                        printf "# Column headers \n";
                        printf "VolumeFraction ActualVolume TotalArea MeanCurvature GaussianCurvature \n";
                        printf "%7.5e %7.5e %7.5e %7.5e %7.5e \n",
                        target_phi, currentVolume, mcurv_area, mcurv_av, gcurv_av;
```

```
                    logfile off;
                    }
// ********************************************************************
// subroutines for evolution scripts – advanced
cut := {
    equiangulate edge;
    equiangulate edge;
    equiangulate edge;
    l 1.5*avg(edge, length);
    equiangulate edge;
    t 0.1*avg(edge, length);
    equiangulate edge;
    foreach facet ff DO {
        IF ff.area > 2*avg(facet, area) THEN {
                refine ff;
                };
        };
    t 0.3*avg(edge, length);
    t 0.4*avg(edge, length);
    equiangulate edge;
    };
calcAveragePatchTriangleArea := {
                    sumAreaPatch:=0;
                    nTriangles:=0;
                    foreach facet ff where color == blue DO {
                        sumAreaPatch:=sumAreaPatch+ff.area;
                        nTriangles:=nTriangles+1;
                        };
                    averagePatchTriangleArea:=sumAreaPatch/nTriangles;
                    nominalTriangleEdgeLength:=sqrt(4/sqrt(3)*averagePatchTriangleArea);
                    print averagePatchTriangleArea;
                    };
subdividePatchTriangles := {
                calcAveragePatchTriangleArea;
                foreach facet ff where color == blue DO {
                    IF ff.area > 2*averagePatchTriangleArea THEN {
                        refine ff;
                        };
                    };
                u;
                u;
                t 0.2*nominalTriangleEdgeLength;
                w 0.3*averagePatchTriangleArea;
                u;
                u;
                g 10;
                };
subdividePatchBorder := {
                foreach edge ee where color == blue DO {
                    IF ( ee on_constraint 1 or ee on_constraint 2 or ee on_constraint 3 or ee on_constraint 4 or ee on_constraint 5)
                    THEN {
                        IF ee.length > 1.6*nominalTriangleEdgeLength THEN {
                            refine ee;
                            };
                        };
                    };
                };
refinePatch := {
        foreach facet ff where color == blue DO {
            refine ff;
            };
        g 5;
        calcAveragePatchTriangleArea;
        subdividePatchTriangles;
        subdividePatchBorder;
        u;
        u;
        u;
        g 10;
        };
```

```
subdividetriangles := {
                        equiangulate edge;
                        g 5;
                        equiangulate edge;
                        g 5;
                        equiangulate edge;
                        g 5;
                        l 1.5*avg(edge, length);
                        g 5;
                        equiangulate edge;
                        g 5;
                        t 0.1*avg(edge, length);
                        g 5;
                        equiangulate edge;
                        g 5;
                        r;
                        g 5;
                        equiangulate edge;
                        g 5;
                        equiangulate edge;
                        g 5;
                        cut;
                        };
// ********************************************************************
// evolution scripts - advanced
gogo2 := {
        g 2;
        refinePatch;
        g 5;
        set body target target_volume;
        refinePatch;
        g 20;
        refinePatch;
        g 200;
        refinePatch;
        g 200;
        refinePatch;
        g 10;
        print_brief_curv_stats;
        };
evolve:= {
        gogoinit;
        set body target target_volume;
        g250;
        r;
        u;
        u;
        V;
        g500;
        u;
        u;
        V;
        g1000;
        removeVolume;
        groom;
        u;
        u;
        V;
        u;
        u;
        V;
        u;
        u;
        V;
        g500;
        u;
        u;
        V;
        groom;
        r;
        g250;
```

```
        print_brief_curv_stats;
        };
//=============================================================================
```

Appendix III: Lonsdaleite family surface data

Visualisation of discontinuous Lonsdaleite family surfaces

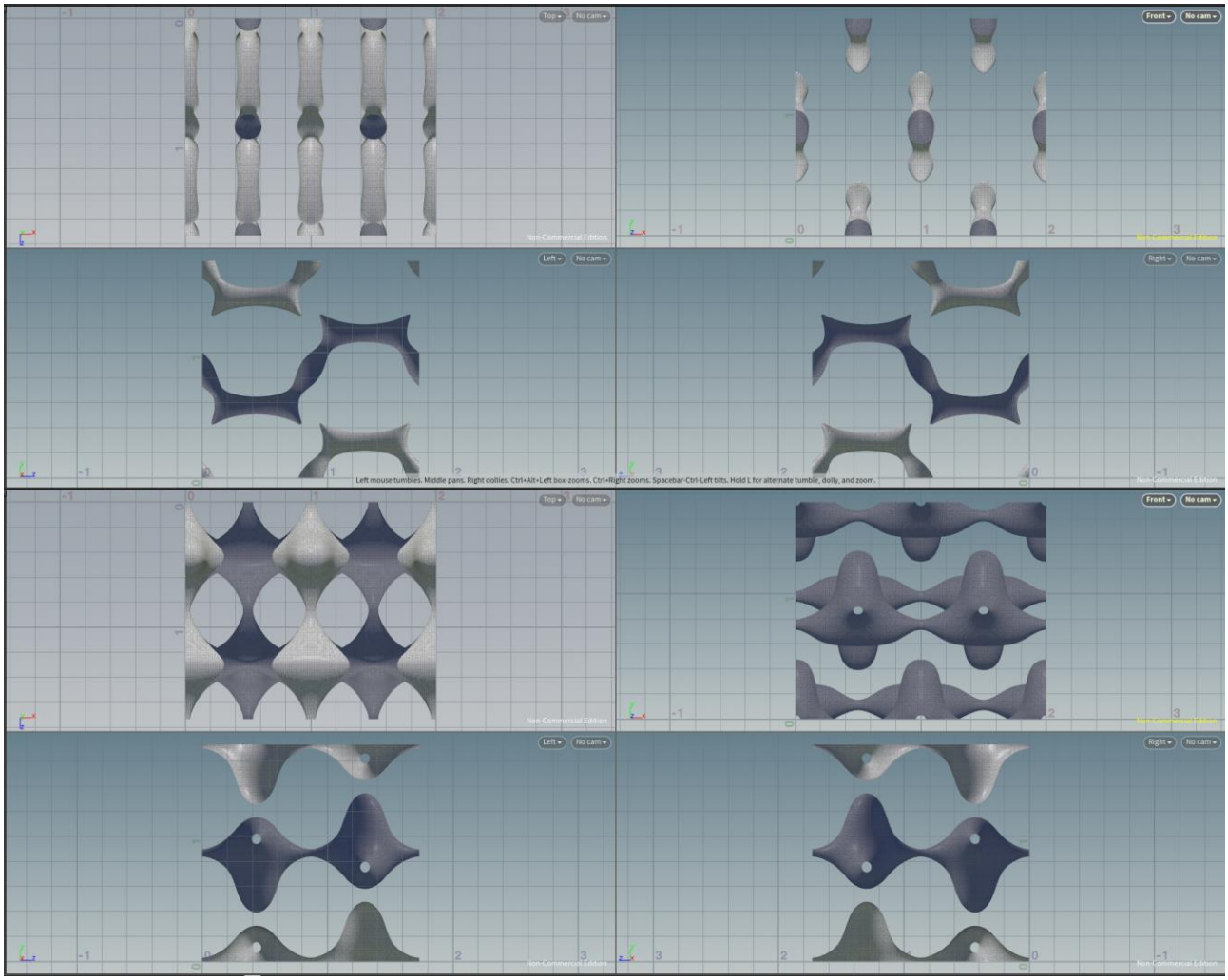


Figure 6: screen capture of Houdini visualisation of of Lonsdaleite family surfaces at level sets equal to 0.5 (upper grid) and -0.65 (lower grid).

Measured values for the prescribed channel volume fraction soap film model for Lonsdaleite family surfaces

level set VF	*prescribed volume*	*measured volume*	*measured VF*	*measured area*	*measured curvature*
0.10948	0.04214	0.04234	0.11000	7.11942	-9132030.00000
0.19610	0.07549	0.07699	0.20000	9.93178	8701370.00000
0.23461	0.09032	0.08854	0.23000	10.28268	-11080700.00000
0.27144	0.10449	0.10394	0.27000	11.66452	-113011.00000
0.32083	0.12350	0.12318	0.32000	13.20612	-3.46469
0.34139	0.13142	0.13088	0.34000	13.77022	166665000.00000
0.37509	0.14439	0.14628	0.38000	14.65928	-3660380.00000
0.40823	0.15715	0.15783	0.41000	13.35594	-1.17892
0.44093	0.16973	0.16938	0.44000	13.35260	-1.17835
0.47336	0.18222	0.18093	0.47000	13.76348	-2.07064
0.50000	0.11605	0.19248	0.50000	16.30229	-0.92196
0.53789	0.20706	0.19633	0.51000	14.42278	-1.29233
0.57029	0.21953	0.20787	0.54000	13.97038	-1.10568
0.60293	0.23210	0.21942	0.57000	15.96288	-1.12738
0.63594	0.24481	0.23097	0.60000	16.18762	-1.01352
0.66966	0.25779	0.24637	0.64000	18.09634	-52670500.00000
0.70423	0.27109	0.25792	0.67000	16.78522	1889820.00000
0.74029	0.28497	0.26947	0.70000	17.71650	376399000.00000
0.77914	0.29993	0.28486	0.74000	17.21356	-10572100.00000