Membrane Vesicle Dynamics Simulation Code Documentation
Chloe M. Funkhouser, Ph.D.

This document provides a brief introduction to the Fortran 90 code written to simulate the dynamics of multicomponent lipid membrane vesicles called Membrane_Vesicle.F90. This code was developed from 2009 to 2011 in the research group of Prof. Katsuyo Thornton in the Materials Science and Engineering Department at the University of Michigan, Ann Arbor. The principal developer was Dr. Chloe Funkhouser with theoretical support provided by Prof. Francisco Solis at Arizona State University. This is a research-quality code that was utilized in Refs. 1-3; additional information can be obtained from these documents. The code is unsupported and the owners assume no responsibility for any usage of the code performed outside of the Thornton Research Group at the University of Michigan.

The code utilizes a phase-field method to represent a two-phase system, with composition defined by phase-field variable 'phi.' This phase-field variable is defined spatially on the vesicle surface, which is initialized as a spherical shell. This two-dimensional surface is defined in and permitted to deform in three-dimensional space. The location of each surface element is defined by its radial distance ('r') from the origin in spherical coordinates and the shape of each element is defined by the metric tensor ('g'). The Yin-Yang grid system (Ref. 4) is employed for the surface discretization, using two grids identical in size and shape to form a spherical surface with minimal overlap similar to the regions on a tennis ball. In the code, all quantities that are defined for each surface element are specified as two arrays: one for the Yin grid and one for the Yang grid. The variable names for the two grids are differentiated by a capital 'E' or 'A;' for example, phase-field variable 'phi' is represented by arrays 'phiE' and 'phiA.' An interpolation scheme is used to connect the two grids in each iteration of the code. The code is written using message passing interface (MPI) commands such that it runs on multiple processors, typically 4 or 8.

The composition of the membrane (each of the two phases) is coupled to the membrane mechanical properties by specifying a spontaneous curvature for each phase; bending rigidity is also included but is constant across phases. Temporal evolution of the composition is accomplished using Cahn-Hilliard dynamics.

The code is designed to be run as a stand-alone code; no configuration or parameter input files are used. Instead the user must manually alter parameters at the top of the source code. These parameters include physical quantities such as the phase area fraction and initial vesicle radius, as well as simulation parameters such as total loop iterations and frequency of data output. Specifying these input parameters in the source code is the only input required, however, the user has the option to provide files specifying the initial phase-field and radius values (this feature was included with the intent that the user could continue a previous simulation run from where it left off). The outputs of the code are space-delimited text files of the phase-field variable 'phi' for each grid element and the radial distance 'r' for each grid element. Separate text files are generated for the Yin and Yang grids, such that a total of four text files are generated each time data are written to file.

Basic Structure of the Code:

- Initialization of parameters, requiring user input
- Initialization of arrays and other non-user-specified code parameters
- Initial calculations of physical parameters and code parameters derived from user input values
- Initialization of the phase-field and radius arrays, either by generating compositional random noise on a perfect sphere or reading in files (typically from a previously run simulation)
- Initial setup for interpolation between Yin and Yang grids, establishing parameters to be used for interpolations performed within the main loop (Ref. 2 provides further explanation)
- Initial setup for surface element weighting, in relation to the Yin and Yang grids, needed for surface area calculations (Ref. 2 provides further explanation)
- Yin-Yang interpolation and surface area calculation for initial membrane state are performed
- Initialization of MPI setup based on grid sizes defined by user
- Main code loop simulating temporal evolution
    - Spatial grid sections are distributed to multiple processors (MPI)
    - Calculations necessary for compositional and spatial evolution are performed
        - These include surface area calculations used to determine current value of surface tension sigma
    - The phase-field variable phi and surface element radial location variable r are updated for current time step
    - The spatial grid sections are collected from multiple processors and sent back to the master processor
    - The interpolation of Yin and Yang grids is performed on master processor only
    - Data are exported to files

Subroutines:

- Derivative: Finite-difference method employed for calculating spatial derivatives
- Derivative_all: Special version of the Derivative subroutine used only for surface area calculations in initialization (since this is performed on the master processor only rather than on pieces of the grid on all processors)
- expdata: Used to export data into text files
- function r4_uniform_01: Used to perform random noise compositional initialization

References:

1. C. M. Funkhouser, F. J. Solis, and K. Thornton, "Dynamics of two-phase lipid vesicles: effects of mechanical properties on morphology evolution," Soft Matter, 6, 3462 (2010).

2. C. M. Funkhouser, Phase-Field Simulations of Multicomponent Lipid Membranes Coupling Composition with Deformation, Ph.D. thesis, University of Michigan (2011).

3. C. M. Funkhouser, F. J. Solis, and K. Thornton, "Dynamics of coarsening in multicomponent lipid vesicles with non-uniform mechanical properties," Journal of Chemical Physics, 140, 144908 (2014).

4. A. Kageyama and T. Sato, ""Yin-Yang grid": An overset grid in spherical geometry," Geochem. Geophy. Geosy., 5, Q09005 (2004).