Simulating a Germinal Center Notes

July 7, 2018

Contents

1 Resources.

Research Paper: Robert, P., Rastogi, A., Binder, S. and Meyer-Hermann, M. (2017). How to Simulate a Germinal Center. Methods in Molecular Biology, pp.303-334.

2 Model Assumptions.

In this section, we highlight the assumptions made in the model we are following. This is primarily to refer to in future when making our own adjustments to the model.

- Only one antigen per Germinal Center.

3 Implementation Ideas

Generate $N \times N \times N$ spatial grid to place discrete sphere with radius $N/2$ within. Find and store the valid spatial points within this grid. At each time step, record where the discrete grid spot is free or contains a cell. Use dictionary, return None if empty, otherwise cell ID? If we want to find the position of a cell, we would need to work the dictionary backwards, likely to be computationally slow. Could have two dictionaries - one that gives position from cell ID and another that gives Cell ID / None from position. Is this likely to cause update issues?

The model given assigns cells an ID based on their position of each cell in its respective type list. This would allow for multiple cells to have the same ID and would require ID and type to be uniquely identified. Rather give each cell a unique ID regardless of type so always identifiable from ID alone.

Don't need to have dictionary storing whether a point is within the sphere, can just re-apply the same calculation used to find original sphere co-ordinates.

Let the points on the surface of the sphere be the points which are missing a neighbour. Should be easy to check.

Algorithms state to randomly iterate through list. Currently doing it by shuffling list and iterating over. Since elements sometimes have to be deleted from these lists, would be better to randomly shuffle a list of indices so the element known to be deleted can be removed straight away. Can make the removal operation quicker by popping end of list and placing it in position of removed element.

Algorithm 4, progress_cycle has been implemented poorly. Instead of having a lot of if statements, could store the states in a list in order of how they transition. Then only have to store the index for each cell and when they progress, add one to that index opposed to using lots of if statements.

The following was followed to generate random unit vectors for initial polarity: https://codereview.stackexchange.com/ques
random-unit-vectors-around-circle

The germinal center has a maximum size. We generate ID values from 0 to the maximum number of cells in the germinal center and store/track them in a list. We assign each new cell an ID number (from the end of the list, using pop() to ensure O(1) operation) and when a cell dies, we append its ID value back into the list. In a numpy array of fixed size, we will store the properties of each cell as a named tuple. To access the correct properties, the associated ID to a cell will be used as the index in the numpy array. Unassigned IDs will contain None values in numpy array and when a cell gets removed from the simulation, the numpy array index is converted back to being a None value.

In main function, we have to iterate over lists randomly and occasionally remove elements from this list, can do so using enumerate function to track indices of what needs to be removed.

4 Algorithms.

4.1 Parameters for each Cell

This table is heavily outdated. Will be updated to talk about the types of cells and their properties.

Property	Domain	Storage	Description?
Assigned ID	Integer starting at Zero	List?	
Location	(x, y, z) co-ordinates	List	
Type/Agent	String	List	
State	String	List	
BCR	4-digit integer	List	
pMutation	Float [0,1]	List	
responsiveToSignal[CXCL12]	Boolean	List	
responsiveToSignal[CXCL13]	Boolean	List	
Polarity	$(\hat{i}, \hat{j}, \hat{k})$ 3D Vector	List	
cycleStateTime	Float	List	
endOfThisPhase	Float	List	
numDivisionsToDo	Integer	List	
IAmHighAg	Boolean	List	
retainedAg	Float	List	
numFDCContacts	Integer	List	
selectedClock	Float?	List	
Selectable	Boolean	List	
Clock	Float	List	
tcClock	Float	List	
tcSignalDuration	Float	List	
individualDifDelay	Float	List	

4.2 General Parameters & Tracked Lists

- We keep two dictionaries, Grid_id and Grid_type. These dictionaries have a key of a tuple location within the sphere and return the cell and cell type located at that location, respectively. If that position is free, both dictionaries will contain 'None'.
- Dictionaries CXCL12 and CXCL12 will stores the amount of each located at a given location.
- We will use StormaList, FDCList, CBList, and TCList to store the IDs of cells in each of these respective states.

4.3 Algorithm 1 (Mutation)

Function: mutate()
Notes: I believe all cells have an affinity? Need to examine reference 13 to determine what is meant by +/-1, do we do both or randomly choose from possibilities?
Input: Cell, C; Probability of Mutation, pMutation.
Accesses: BCR.
Alterations: BCR.

4.4 Algorithm 2 (Dynamic Updating of Chemotaxis)

Function: initiateChemokineReceptors()
Notes:
Input: Cell, C (ID); Type of cell, T.
Accesses: responsiveToSignal[CXCL12], responsiveToSignal[CXCL13].
Alterations: responsiveToSignal[CXCL12], responsiveToSignal[CXCL13].

Function: updateChemokinesReceptors()
Notes:
Input: Cell, C (ID).

Accesses: responsiveToSignal[CXCL12], responsiveToSignal[CXCL13].
Alterations: responsiveToSignal[CXCL12], responsiveToSignal[CXCL13].

4.5 Algorithm 3 (Updating Position and Polarity of cells at each time-point)

Function: move()
Notes: With probability persistentLengthTime(Cell Type) do ..., looking at table of values, persistentLengthTime is greater than 1 for all but one type of cell.
Input: Cell, C (ID).
Accesses: Polarity, responsiveToSignal[CXCL12], responsiveToSignal[CXCL13].
Alterations: Cell Location; Cell Polarity
Function: updateChemokinesReceptors()
Notes: The inputs Θ and Φ are randomly generated in the function move(). The instructions for this function seem to be a bit confusing. Will need to draw this out to understand it better.
Input: Cell Polarity; Θ; Φ.
Accesses:
Alterations: Cell Polarity.

4.6 Algorithm 4 (Updating events at the Centroblast Stage)

Function: initiateCycle()
Notes: getDuration(x) implies sampling guassian with mean x, standard deviation 1.
Input: Cell, C (ID); number of divisions to occur.
Accesses:
Alterations: C.state, C.cycleStartTime, C.endOfThisPhase, C.numDivisionsToDo.
Function: progressCycle()
Notes: 'Progress: ...' Implies that you move the cell to the next state available.
Input: Cell, C (ID).
Accesses: C.cycleStateTime, C.state.
Alterations: C.state, C.cycleStartTime, C.endOfThisPhase, C.numDivisionsToDo.
Function: divideAndMutate()
Notes: This function can create a new cell.
Input: Cell, C (ID).
Accesses: C.retainedAg
Alterations: New cell (Location and ID), IAmHighAg, retainedAg

4.7 Algorithm 5 (Antigen Collection from FDCs)

Function: progressFDCSelection()
Notes: Clock++ means increase clock by 1 right? If so, is it really an integer? F is an FDC, f is a fragment neighbouring C.
Input: Cell, C (ID).
Accesses:
Alterations: selectedClock, clock, selectable, numFDCContacts, F.antigenAmount[f].

4.8 Algorithm 6 (Screening for T cell help at the Centrocyte Stage)

Function: progressTCellSelection()
Notes: Calls functions in Algorithm individualDifDelay is Guassian.
Input: Cell, C (ID).
Accesses:
Alterations: state, tcClock, tcSignalDuration, individualDifDelay.

4.9 Algorithm 7 (Updating the T cells according to B cells Interactions)

Note: For the following two function, we introduce a list denoted 'ListInteractingB' to track the IDs of the B cells currently in interaction.
Function: updateTCell()
Notes:
Input: Cell, TC (ID); Cell, B (ID).
Accesses:
Alterations: TC.state, ListInteractingB
Function: liberateTCell()
Notes:
Input: Cell, TC (ID); Cell, B (ID).
Accesses:
Alterations: TC.state, ListInteractingB

4.10 Algorithm 8 (Transition between Centroblasts, centrocyctes, and output Cells)

Function: differToOut()
Notes: We track Out cells using list 'outList', and number of using 'numOutCells'.
Input: Cell, C (ID);
Accesses:
Alterations: numOutCells, outList
Function: differToCB()
Notes: We track Centroblasts using list 'CBlist'.
Input: Cell, C (ID).
Accesses:
Alterations: CBlist, pMutation, retainedAg, IAmHighAg
Function: differToCC()
Notes: We track Centrocytes using list 'CClist'.
Input: Cell, C (ID).
Accesses: C.retainedAg
Alterations: State, numFCDContacts.

4.11 Algorithm 9 (Initialisation)

Function: initialiseCells()
Notes:
Input: Various inputs such as number of Stromal cells.
Accesses:
Alterations: Generates cells, FCD Fragments, etc.

4.12 Algorithm 10 (Hyphasma: Simulation of Germinal Center)

Function: initialiseCells()
Notes: Driver function.
Input: Various inputs regarding simulation parameters.
Accesses:
Alterations:

5 Definitions & Terms.

- **Affinity Maturation**: The process by which Tfh cell-activated B cells produce antibodies with increase affinity for antigen during the course of an immune response.

- **Stromal Cells**: Connective tissue cells of any organ.
- **B Cells**: Type of white blood cell.
- **Centroblasts**: B cell that is enlarged and proliferating in the germinal center.
- **Clonal Expansion**: A large increase in the number of B cells and T cells in the presence of an infection.
- **Somatic Hypermutation (SHM)**: A cellular mechanism by which the immune system adapts to the new foreign elements that confront it. Allows B cells to mutate the genes that they use to produce antibodies.
- **B Cell Receptor (BCR)**:
- **VDJ Recombination Pattern**:
- **in vivo**: A study where the effects of various biological entities are tested on whole, living organisms or cells.
- **Centrocytes**: Nondividing B cells that endure a high apoptosis rate.
- **Follicular Dendritic Cells (FDCs)**: A type of cell in the immune system.
- **Fc Receptors**: A protein found on the surface of certain cells that contributes to the protective functions of the immune system.
- **MHC Class II**: A class of molecules normally found only on antigen presenting cells. Important in initiating immune responses.
- **Antigen Presenting Cell (APC)**: A cell that displays antigen complexed with major histocompatibility complexes (MHCs) on their surfaces.
- **T Helper Cells**: Cells that help the activity of other immune cells by releasing T cell cytokines (small proteins).
- **T Follicular Helper (Tfh)**: Within a Germinal Center they mediate the selection and survival of B cells that differentiate either into special plasma cells capable of producing high affinity antibodies against foreign antigen, or memory B cells capable of quick immune re-activation in the future if the same antigen is ever accounted again.

5.1 Questions for Julian

- Currently no question.

5.2 To do list

- Configure ssh key on github account and on computer
- Go through learn git branching try.github.io
- Book: Chapters 1, 3, 9, 11, 15, 16, 18, 19