Assignment 3

Making a Cellular Potts Model of Collective Migration

Deadline March 18, 2022

Handout for the *Natural Computing* lecture, March 8, 2021

Johannes Textor, Inge Wortel

In this assignment, you are going to use the cellular Potts modeling (CPM) framework to create cells *in silico*. You will first learn how to use the model's many parameters to create cells of realistic shapes and motility patterns. Then you will go a step further and model several migrating cells that interact with each other.

This assignment considers a slightly more advanced CPM than we discussed in the lecture. The assignment therefore consists of two parts. First, there are three self-study exercises that walk you through the CPM using an interactive online implementation. *There is no need to write answers to these questions, these are simply meant to introduce you to the model.* Some questions are marked "Optional" which means that we don't consider these to be totally essential for completing this assignment, but you can use these to test how far your understanding of the model goes.

Second, we ask you to implement your own CPM simulation that extends the model from this exercise. This is the assignment that you will write a (brief) report about.

Objectives of This Exercise

1. Understand how the various CMPM parameters interact with each other to govern cell behavior.
2. Realize that it can be difficult to tune the parameters of a CPM.
3. Apply this knowledge to create different modes of cell migration in the CPM, and explain how this helps understand those migration modes.
4. Apply the CPM to investigate the dynamics of collective cell migration *in silico*.

Please go to computational-immunology.org/cpm/collective.html. This is an implementation of a special version of the Cellular Potts Model in which cells can migrate [**?**]. Note that *this web page does not work properly in Internet Explorer*. It does work in either Firefox, Chrome, or Safari – so we recommend using any of those for this exercise.

Recap: the CPM

The following section recaps the information on the CPM given in the lecture; you can use this as a reference while making the exercises.

Let us recall the basic CPM as defined in the lecture. We have a spatial structure $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ with grid nodes $\mathbf{V}$ and neighbourhood relationship $\mathbf{E} \subseteq \{\{i, j\} \mid i, j \in \mathbf{V}\}$. In this exercise, we use a square 2D grid with 8-neighbourhoods, so each grid point is connected to its north, northeast, east, southeast, south, southwest, west, and northwest neighbours. At the borders, we wrap the lattice such that, for example, the west neighbour of the westernmost pixel is the easternmost pixel. We have a function

$$\sigma : \mathbf{V} \to \{0, 1, \ldots\}$$

that assigns a *cell identity* to each grid node. By convention, the cell identity 0 stands for "no cell", also called "background" or "matrix"[1]. Positive cell identities stand for different cells (foreground).

The state of the CPM changes dynamically over time because of local "copy attempts", in which nodes try to copy their identity σ into (randomly selected) neighbor nodes. Remember that such copy attempts have a higher chance of succeeding if they lower the total energy $\mathcal{H}$ of the system. In other words, $\Delta\mathcal{H} = \mathcal{H}_{\text{after copy}} - \mathcal{H}_{\text{before copy}}$

[1] This word refers to the concept of the "extracellular matrix" in Biology, which basically refers to structures outside of cells in the body.

should be negative to guarantee that the copy attempt will work. Copy attempts with positive $\Delta\mathcal{H}$ can still succeed, but do so with a lower probability that depends on the temperature of the system.
In general, the formula for $\Delta\mathcal{H}$ of a CPM looks something like this[2]:

$$\Delta\mathcal{H} = \Delta\mathcal{H}_{\text{adhesion}} + \lambda_{\text{V}}\Delta\mathcal{H}_{\text{volume}} + \lambda_{...}\Delta\mathcal{H}_{...} + ... \tag{1}$$

Thus, we build up $\Delta\mathcal{H}$ from different energetic factors (terms) we want the cell to consider – and we can always add more terms to make the model more complex. Often, we will use "importance parameters" $\lambda_{...}$ [3] to set the relative strengths of the energy terms with respect to the baseline adhesion term.
The adhesion term in the Hamiltonian is defined as

$$\mathcal{H}_{\text{adhesion}} = J(\sigma_i, \sigma_j) \sum_{\{i,j\}\in\mathbf{E}} \delta(\sigma_i, \sigma_j)$$

where J is a parameter that determines the adhesion between cell σ_i and σ_j. Often, we group the cells into different "cell types" (for example, by assigning the same type to all non-background cells), and then we will only let J depend on the type of each cell rather than the identity of each cell.
The volume term is defined as

$$\mathcal{H}_{\text{volume}} = \sum_{\tau>0} \left(\#\left\{i \mid \sigma_i = \tau\right\} - V(\tau)\right)^2$$

In words, $\mathcal{H}_{\text{volume}}$ measures the squared difference between each cell's current volume and its desired target volume. $V(\tau)$ is a parameter that determines the desired size of cell τ. Again, often several cells are grouped to one cell type, and the desired size is only determined per cell type. Summing over $\tau > 0$ means that the background has no volume constraint.
We will also use a further term called a *perimeter constraint*, which is defined as follows:

$$\mathcal{H}_{\text{perimeter}} = \sum_{\tau>0} \left(\#\left\{\{i,j\} \in \mathbf{E} \mid \sigma_i = \tau, \sigma_j \neq \tau\right\} - P(\tau)\right)^2$$

In words, $\mathcal{H}_{\text{perimeter}}$ measures the squared difference between each cell's current perimeter and its desired target perimeter. The perimeter of a cell is measured as the number of neighbouring pixel pairs where one of the pixels belongs to the cell and the other does not. Similarly to the volume constraint, $P(\tau)$ is a parameter that determines the desired perimeter of cell τ, which, again, often depends only on the "type" of cell τ. Summing over $\tau > 0$ means that the background has no perimeter constraint.
You can see some of these constraints in action here: https://artistoo.net/explorables/Explorable-CPM.html.

How do we need to set our parameters such that we get realistic cells? In the following exercises, you will get a feel for how you can model cell behavior with a CPM by tuning the parameters that control the energy $\Delta\mathcal{H}$.

[2] The first term in this equation is the general *adhesion term* included in all Potts models, and it is described in slide 5ff in the lecture. There, the adhesion parameter is called "J" in accordance with the literature on the cellular Potts model.
[3] The λ actually stands for "Lagrange multiplier"

Self-study exercises

Exercise 3.1 A very basic CPM

In this exercise, we will first examine a very basic CPM in which cells do not (yet) migrate. All of these cells are considered to have the same type, so they are all affected by the same parameters. Make sure that the field is empty (hit refresh or "remove all cells"), and that the parameters have the following values:

$\text{Adhesion}_{\text{cell-matrix}}$	$\text{Adhesion}_{\text{cell-cell}}$	Volume	λ_V	Perimeter	λ_P	Max_{Act}	λ_{Act}	T	Framerate
20	0	500	50	340	0	0	0	20	1

We will now investigate how the basic CPM parameters – controlling adhesion, cell volume, and cell perimeter (circumference) – influence behavior (this means you can ignore the λ_{Act} and Max_{Act} parameters for now). This exercise is meant mostly to give you an idea of what the CPM parameters do, and the questions are to guide your thinking – so you don't have to write everything down. Try to spend no longer than 30-40 minutes on this exercise before continuing to the next.

1. Make sure all the parameters are set as in the table above, click "seed cell" and then "start". What do you see? What kind of motion does this cell have?
2. Now set the $\text{Adhesion}_{\text{cell-matrix}}$ to 0. What happens to the cell? Why do you think that happens? (Hint: look back to the description of the CPM and adhesion energy in the lecture...) Also try a negative value for $\text{Adhesion}_{\text{cell-matrix}}$. What is the meaning of positive or negative adhesion values here?
3. *(Optional)* Instead of setting the $\text{Adhesion}_{\text{cell-matrix}}$ back to 20, try setting the $\text{Adhesion}_{\text{cell-cell}}$ to -20 while having the Adhesion $_{\text{cell-matrix}}$ still at 0. Does that have the effect that you expected? Why/why not do you think that is? Hint: try drawing a grid like you saw in the lecture for a copy attempt you are interested in. Do the two adhesion energies change in the same way for that copy attempt?
4. Return to the parameters in the table above. With these parameters, the cell is given an ideal volume (500 pixels), and a "level of importance" of this volume for the energy (λ_V). Try making the cell bigger or smaller (what parameter should you change?). How can you make the volume unimportant for the energy – and what happens then? What happens when you make λ_V really large (say, 1000)?
5. So far, we have considered $\Delta\mathcal{H}$ with only terms for adhesion and cell volume. We will now investigate the effect of the cell perimeter (circumference). The cell already has a target perimeter (340), but this is currently not taken into account in the calculation of $\Delta\mathcal{H}$. For that, we need to make λ_P non-zero. Try setting it to 2. What happens to the cell? Try making the cell "membrane" more or less ruffled. How would you do that?
6. Set the perimeter to 340 and λ_P to 2. Now change the $\text{Adhesion}_{\text{cell-matrix}}$ to 0 again. Does this have the same effect as it did in question 2? Why do you think that is?
7. *(Optional)* Play around with the volume and perimeter parameters for a while (using the adhesion parameters from the table, or try your own). How can you change the cell? Can you change the parameters independently of each other? And what happens if you change the temperature?

Exercise 3.2 Cell migration: the Act model

We will now investigate the Act model, an extension of the CPM that allows the cells to migrate **[?]**. This model adds an extra "act" term to the system energy, so that:

$$\Delta\mathcal{H} = \Delta\mathcal{H}_{\text{adhesion}} + \lambda_V\Delta\mathcal{H}_{\text{volume}} + \lambda_P\Delta\mathcal{H}_{\text{perimeter}} + \lambda_{\text{Act}}\Delta\mathcal{H}_{\text{act}} \tag{2}$$

In this model, pixels that were recently added to the cell remember their recent "protrusive activity". This makes them more likely to protrude again. This positive feedback is controlled by the energy term $\Delta\mathcal{H}_{\text{Act}}$, which is negative (favourable!) when a recently active pixel tries to copy itself into a less active pixel. The Act model has two extra parameters: λ_{Act}, which controls how important the positive feedback is relative to the other $\Delta\mathcal{H}$ energies, and Max_{Act}, which determines how long pixels "remember" that they were active. More formally, we define an integer-valued "activity function" on the grid

$$A : \mathbf{V} \to \{0, 1, \ldots\}$$

We initially set $A(v) = 0$ for all $v \in \mathbf{V}$. After each Monte Carlo step, we decrease each positive $A(v)$ by 1. Whenever a non-background cell successfully manages to copy itself into a target pixel v, we set $A(v) = \text{Max}_{\text{Act}}$.

For a copy attempt where pixel i tries to copy its identity into pixel j, we define the contribution of the Act model to the energy gradient as follows:

$$\Delta\mathcal{H}_{\text{Act}} = \frac{\text{GM}_A(j) - \text{GM}_A(i)}{\text{Max}_{\text{Act}}}$$

Here $\text{GM}_A(i)$ stands for the geometric mean of the activity function near pixel i within the same cell. Formally, if

$$N_\tau(i) = \{i\} \cup \{k \mid \{i,k\} \in \mathbf{E}, \sigma_i = \sigma_k\}$$

then

$$\text{GM}_A(i) = \left(\prod_{k \in N_\tau(i)} A(k)\right)^{1/\#N_\tau(i)}.$$

This definition has the following effect. If there is a lot of activity near the source pixel i and little activity near the target pixel j, then $\Delta\mathcal{H}_{\text{Act}}$ is negative and the copy attempt becomes more likely to succeed. If there is little activity at i and lots at j, the copy attempt becomes less likely to succeed. Because activity is created from successful copy attempts, this basically means that we "reward" pixels that have successfully managed to reproduce themselves and make it more likely for them to reproduce again in future copy attempts. In other words, a positive feedback is generated.

In this exercise, we will see what happens when we vary the parameters Max_{Act} and λ_{Act} in the model. In particular, we will see that we can reproduce two very different "modes" of migration: amoeboid and keratocyte-like (see lecture). Before you start, please refresh the page and set the CPM parameters as follows:

$\text{Adhesion}_{\text{cell-matrix}}$	$\text{Adhesion}_{\text{cell-cell}}$	Volume	λ_{Volume}	Perimeter	λ_{P}	Max_{Act}	λ_{Act}	T	Framerate
20	0	500	50	340	2	20	0	20	1

1. Seed a cell and click "start". You should now see colored pixels at the border of the cell, which indicate the "activity" that pixels remember (because we have set Max_{Act} to 20). Other than the color of the pixels, does the cell behave in a different way than with $\text{Max}_{\text{Act}} = 0$? Why/why not?
2. Set λ_{Act} to 100. Would you describe this movement as random or persistent?
3. *(Optional)* What happens when you set λ_{P} to 0 now? Why do you think that is? (Reset it to 2 before going to the next question)
4. What happens when you increase λ_{Act} further? (Try steps of 100).
5. *(Optional)* If you increase λ_{Act} to very high values (eg 1000), the cell is prone to breaking in pieces. Can you fix that by altering some other CPM parameter again? (Note: you may have to increase λ_{Act} further when you have done this... Does that make sense to you?)
6. Reset λ_{Act} to 0, change Max_{Act} to 80, and repeat questions 1,2, and 4 above. What do you see?
7. *(Optional)* If you have time, play around with different combinations of λ_{Act} and Max_{Act}. Can you get a clue of what they are doing – beyond the mathematical description given above?
8. *(Optional)* Try halving or doubling the cell's target volume. That won't work. What do you need to change to get the same behavior as before? What does that mean for your model (in other words: to what extent are your choices of parameters important for the behaviour you see? How worried should you be about getting parameters "wrong" and drawing the wrong conclusions?)?

Exercise 3.3 Migration in a multicellular system

In this last exercise, we will investigate what happens when there are many cells. Before starting, please refresh the page to clear the grid, and then ensure that the parameters have the following values:

$\text{Adhesion}_{\text{cell-matrix}}$	$\text{Adhesion}_{\text{cell-cell}}$	Volume	λ_{Volume}	Perimeter	λ_{P}	Max_{Act}	λ_{Act}	T	Framerate
20	0	200	50	180	2	20	200	20	5

(The frame rate is not a parameter of the model, but specifies how often the updated grid is drawn (eg framerate = 5 means: draw only 1 in every 5 "frames"). Setting it to 5 may speed up the animation.

1. To seed many cells at once, click "+10 cells" a few times (this will take a while...). What do you see? How does the behavior change as the grid becomes more densely packed with cells?
2. Now, refresh the page, reset the parameters, increase Max_{Act} to 80, and repeat the previous exercise. What happens? How is this different from what you saw with $\text{Max}_{\text{Act}} = 20$? How full can you make the grid before the behavior really changes (if it ever does...)? If you're in doubt, change Max_{Act} back to 20 and compare what happens with a full grid versus a single cell at this parameter value...

Assignment

Now that you have completed the self-study part of this assignment, you have all the necessary background information to implement your own model to investigate the "sensitivity" of your observation to changes in the model. Specifically, we will look what happens when obstacles are present.

For this part of the exercise, we recommend that you use our framework Artistoo, written in JavaScript. (Of course, feel free to try implementing a CPM yourself in any programming language you wish if you prefer to do so – you should just realize that this could take you quite some time...)

You can find Artistoo at https://github.com/ingewortel/artistoo. It contains many methods allowing you to build and visualize CPMs, for which you can find documentation and some tutorials at https://artistoo.net (see "Manual" and "Documentation"). We also recommend having a look at the examples (https://artistoo.net/examples.html), for which you can find the code in the `examples/html/` folder of the code repository.

In the first part of the exercise, we have seen that "crowding" (having a lot of cells close together) can impact movement under some conditions, but not others. We will now investigate how obstacles change the motion of cells on a densely packed grid.

1. How could you model an obstacle in the CPM? Hint: can you make a round obstacle by changing the CPM parameters in a certain way?
2. From exercise 3.3, start from the simulation where cells kept moving even at large densities. Then adapt it to investigate the effect of obstacles. Hint: you can do this by building a simulation with two kinds of cells, one of which is for the obstacles – see e.g. `Cellsorting.html`, `ManyCellsPrefDir.html`, or `examples/html/EpidermisWithTCells.html` in the `examples/html/` folder.
3. What happens when you place obstacles between the cells? Do they still move? What happens if you increase the number of obstacles? Please use obstacles with half the size of the cells for this exercise, and place them on the grid with regular spacing between them. Think about which obstacle densities you consider and why (Hint: You will have to modify the `initializeGrid` method of your simulation. Have a look at the `seedCellAt` method and the `examples/html/CancerInvasion.html` example).
4. Do check your simulations carefully; getting rid of obvious bugs/artefacts is also part of simulation research...

If you encounter any problems or have any questions, please let us know and we will try to get back to you as soon as possible. From the above, it should be obvious what you are supposed to do – if it is not or is taking you an unreasonable amount of time, do reach out in time!

Report

- Write a brief report (at most ~ 4-5 pages) on this assignment. In particular, you should discuss how you have implemented the obstacles in the CPM, and what effects these obstacles have on collective cell migration. Observe closely. What do you see? Did you expect that? What are your main conclusions?
- Where necessary, you can add screenshots to illustrate the behavior you see, and one or two screen recordings of the simulations that you based your conclusions on. Think about visualizations – how can you best show your point? For example, you can make screenshots of simulations, you can attempt to measure the motion of the cells. Think about how colors etc can help bring your message across. What do you need to show to convince me of your conclusions?
- You can also make videos and upload them with your assignment, but if you do: (1) please ensure that you use some generic video format that can be opened from any computer. Please don't embed them in the pdf file as this can be tricky with portability. And (2) please don't rely on videos only; some (non-moving) visualization of your main results should be clearly visible in the report itself.
- Please specify precisely how you built your simulation, for example, by reporting parameters in a table. We should be able to reproduce your work from the report alone – without having to look at any code!
- The deadline for submission is March 18th (see Brightspace).

References

[1] I. Niculescu, J. Textor, and R. J. de Boer. Crawling and Gliding: A Computational Model for Shape-Driven Cell Migration. *PLOS Computational Biology*, 11(10):1–22, 2015.