Annales Mathematicae et Informaticae
***51** (2020) pp. 41–52*
DOI: 10.33039/ami.2020.07.010
https://ami.uni-eszterhazy.hu

Evaluation of colony formation dataset of simulated cell cultures

Dániel Kiss[a], Gábor Kertész[a], Máté Jaskó[a], Sándor Szénási[ac], Anna Lovrics[b], Zoltán Vámossy[a]

[a]John von Neumann Faculty of Informatics, Óbuda University, Hungary

[b]Membrane Protein Research Group, Research Centre for Natural Sciences, Hungarian Academy of Sciences, Hungary

[c]Faculty of Economics and Informatics, J. Selye University, Slovakia

Submitted: February 4, 2020
Accepted: July 1, 2020
Published online: July 23, 2020

Abstract

In vitro biological experiments and *in silico* individual-based computational models are widely used to understand the low-level behavior of cells and cellular functions. Many of these functions can not be directly observed, however, may be deduced from other properties that can be well measured and modeled. In this paper, we present a procedure to evaluate synthetic cell colony formation generated by an off-lattice individual-based model. The calculated shape features of the artificial cell aggregates can be related to the parameter values of the simulated agents, therefore this data can be used to quantify properties of real-life cells such as motility or binding affinity that can not be easily determined otherwise. Our experiments showed that only a few of these parameters are responsible for the difference in shape features of the colonies.

Keywords: modeling in vitro cells; colony formation; agent-based simulation; parameter inference; machine learning

1. Introduction

1.1. Background and motivation

Biological experiments frequently carried out on cell cultures, also known as *in vitro* cultures. These cells are usually kept alive in culture media added into different sized culture dishes. At ideal conditions, the cells start to proliferate and reproduce, which leads to an increased size of the initial populations. Depending on individual cellular-level properties, these cells may form tight, regularly-shaped colonies, loosely-connected aggregates with irregular edges or no distinguishable clusters at all.

An experienced researcher can recognize a change in some cellular functions or properties only by looking at the culture under a microscope and see the differences in the pattern of cell aggregates, their size, shape, etc. Analyzing microscopy images by specific image processing applications can also also provide useful information, such as the percentage area covered by cells, the number of cell aggregates detected or the statistical features of the shape descriptors of colonies. On the other hand, it is usually impossible to objectively and precisely define the changes in cellular-level functions only by evaluating microscopy photos.

1.2. Aims of the research

Our objective was to propose a method, which demonstrates how it is possible to relate some pre-selected cellular properties to the measured shape features of multi-cellular aggregates. To do so, we first created an individual-based model that captures the selected properties of a single cell *in vitro*. Then, a large number of input parameters were generated and multiple simulations were executed. The resulting dataset was processed by a shape feature extraction algorithm. Finally, we used a multi-layered neural network to relate the extracted shape features of the artificial cell-aggregates to the input parameters of each simulation.

2. Related work

In the last few decades, the so-called individual-based modeling technique became more and more popular in this field, partially because it can provide useful insights into cellular level features based on the emergent behavior of a large population of individuals, also called as agents. For instance, such techniques can be utilizes to study collective cell migration [14], to model the calcium dependent behaviour of epithelial cells [16] or malignant tumor growth [12], just to mention some.

Previous researches showed, that even a relatively simple individual-based computational model of individual cells is able to produce different colony morphologies when some of the input parameters were altered [6–8]. The growth dynamics and morphology features of multi-cellular aggregates can be also captured by statistical physics models [1, 9]. Such models suggest that the key mechanism in monolayer

colony formation is the surface diffusion of cells at the boundary of the aggregate [3], however, this problem is still not entirely solved [4].

3. Methods

3.1. Agent-based modeling

We used a simplified version of the model introduced by Drasdo et al. [8] and previously presented in [11]. This model uses the position, mean diameter, core ratio, adhesive ratio, adhesion factor and velocity data of each individual. Therefore, cells can be interpreted as partially overlapping, sticky disks with diameter d on a two-dimensional circular and bounded flat surface (the bottom of the culture dish). The position of agent i is stored in its coordinate vector $\vec{x_i}$. Core ratio $r^c < 1$ defines the diameter of an embedded disk (core). For two or more agents cores should not overlap, so values of $r^c \approx 0$ belong to highly elastic cells, while $r^c \approx 1$ to highly rigid ones. Adhesive ratio $r^a > 1$ defines the distance in which two agents form adhesive bonds. Interaction properties of agents i and j being in distance x are incorporated by an interaction potential function defined as

$$V_{ij}(x) = \begin{cases} \infty & \text{if } x \leq d_{ij}^c \\ -\varepsilon & \text{if } d_{ij}^c < x \leq d_{ij}^a \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

where

$$d_{ij}^c = \frac{r_i^c d_i + r_j^c d_j}{2} \quad \text{and} \quad d_{ij}^a = \frac{r_i^a d_i + r_j^a d_j}{2}$$

are the core and adhesion distances of the agents, respectively. Value ε describes the agent-agent adhesion energy and can be directly linked to quantities such as cell membrane adhesion receptor density [2, 7]. To simulate this model, a Monte Carlo rejection sampling process [13] is executed, during which agent i is displaced by the vector $\delta\vec{u}$ with acceptance probability

$$\min\left\{1, \quad \exp^{-1}\left(\sum_{i\neq j} V_{ij}^{t+\Delta t} - \sum_{i\neq j} V_{ij}^{t}\right)\right\},$$

where $\vec{u}$ is a randomly directed unit vector, δ is a gamma distributed distance with shape parameter k and scale parameter θ and Δt is the unit of simulation time scale.

Duplication of agent i is based on its cell cycle time τ_i (the time duration between two division events) and time counter (internal clock) state t_i. When

$$t_i \geq \frac{\tau_i}{\Delta t}$$

cell duplication is performed by creating a copy i' of agent i and assigning new coordinates

$$\vec{x_i}^{\text{new}} = \vec{x_i}^{\text{old}} + \frac{1}{2} r_i^c d_i \vec{u}$$
$$\vec{x'_i}^{\text{new}} = \vec{x_i}^{\text{old}} - \frac{1}{2} r_i^c d_i \vec{u}$$

to the agents, where $\vec{u}$ is a uniformly distributed two-dimensional unit vector. If there is no sufficient space to locate both agents (that is, when the interaction energy defined by (3.1) is infinity), the duplication trial is rejected. Otherwise, both agents are set to their initial cell cycle state.

3.2. Input data generation and simulation

To simulate the model, we generated all possible input parameter combination to better explore the structure of the feature space. However, to decrease the number of individual combination, some model parameters were fixed by analyzing and evaluating real microscopy images, therefore we set $d = 25$ µm, $r^c = 0.8$, $r^a = 1.2$, $\tau = 24$ h and $\theta = 0.1$. All other input values were picked one by one from its possible range (see Table 2 for details). To minimize stochastic effects, threefold replication were used with all parameter combinations.

When a simulation is started, a given number of agents (approximately 250) are randomly placed into the simulation space which is a circular surface with a diameter of 6.4 mm (this is approximately equivalent to the diameter of a standard 96-well culture dish). The locations of each agent are saved periodically and a pseudo-microscopy image is rendered from the simulation data, where black disks represent the simulated cell on a white background (somewhat similar to in Fig. 1). This photo is later loaded into the image processing software for further evaluation.

3.3. Shape feature extraction

To evaluate the rendered pseudo-microscopy images, we created a batch feature extraction pipeline in CellProfiler [5]. This pipeline first smooths the image using a Gaussian filter with kernel size $\sigma = 1.0$ to make the artificial image more realistic. Then, an automated binarization using minimum cross-entropy thresholding separates the foreground (cell aggregates) from the background. To remove small artifacts and holes at the boundary of the colonies, we performed a closing operation with a disk structuring element. After that, all distinct foreground objects are marked and processed one by one. When shape features of all detected objects are determined, the data is saved as an output file, along with the corresponding simulation parameter values and identifiers such as object label, frame number, etc.

The most significant attributes along with their description and key statistical properties of the produced dataset is summarized on Table 1.

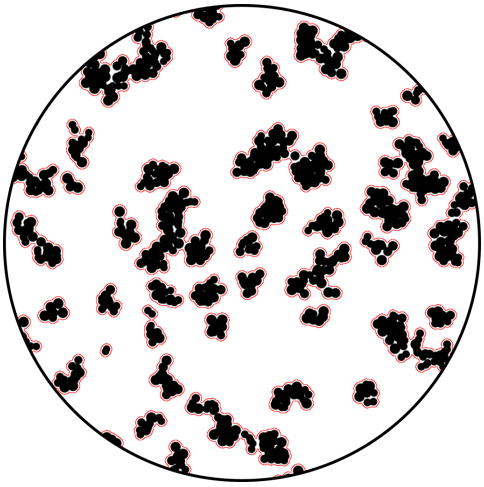

Figure 1: Representative image of a pseudo-microscopy image of simulated cells (black dots) in a small circular vessel. The boundaries of detected cell aggregate objects are shown red. Note, that cell diameters and vessel diameter are not realistic on this image.

Attribute	Unit	Mean	SD	Min	Max
Area: the number of pixels covered by the given object	pixel	1549.77	1995.13	259.0	228649.0
Perimeter: the total number of pixels around the boundary of each region in the image	pixel	157.11	119.8	64.42	9792.12
MeanRadius: the mean distance of any pixel in the object to the closest pixel outside of the object	pixel	5.74	1.61	3.0	26.65
Compactness: the mean squared distance of the object's pixels from the centroid divided by the area	dimensionless	1.15	0.21	1.0	5.29
Solidity: the proportion of the pixels in the convex hull that are also in the object	dimensionless	0.92	0.05	0.36	1.0
FormFactor: calculated as $4\pi \text{Area}/\text{Perimeter}^2$	dimensionless	0.77	0.16	0.03	0.98

Table 1: Most significant shape attributes of simulated cell aggregates along with their short description and basic statistical properties

3.3.1. Data filtering

Since the set contains time series data, measured values should be reviewed. An interesting phenomenon can be observed when multiple distinct cell aggregates merged into one large aggregate. In these cases, the segmentation method is not able to correctly distinguish these objects, therefore produce invalid shape measurements.

On the other hand, individual cell movements can result in breaking up these

Parameter	Unit	Value
ε: adhesion energy parameter ("stickyness")	dimensionless	$\varepsilon \in \{1, 3, 5\}$
k: distribution shape parameter of the mean displacement step size ("velocity")	dimensionless	$k \in \{2, 3, 4\}$
Δt: time resolution of the simulation	minute	$\Delta t \in \{1, 2, 4, 8, 16, 32, 64\}$
N_{MC}: number of repeated Monte Carlo displacement trials of an agent in a given time step.	dimensionless	$N_{MC} \in \{1, 2, 4, 8, 16, 32, 64\}$

Table 2: Input parameter data of the simulated agents (See 3.1 for a detailed description)

clusters into separate objects. These combined features cause outliers, resulting significant noise in the observations. Affected objects can be removed from the dataset based on the area sizes, using a simple algorithm. At time step 0, all objects are removed from the set where the area size is considered large, according to the input values. At every other time step i, the observed area size A_i^j of object number j is compared to the last observation A_{i-1}^j, and if $A_i^j > mA_{i-1}^j$, where m is a factor defined as $\sqrt{2}$, 2 or 3, the observation is considered as an outlier; therefore, it is removed.

The application of this method will result in serious imbalance, or – if all observations of an object are removed on detection – cropping of the dataset. In our experiments, approximately 99% of objects are removed because of a size mismatch at some point of their lifetime. It is important to point out, that object numbers are not unique identifiers: the numbering in each frame restarts. As a consequence, some objects which themselves are not affected by clustering error are removed by the dataset because of incorrect labeling caused by a nearby error.

However, cell aggregate object identification is possible based on the coordinates of their midpoint: an assumption can be made that object o_1 at observed time point t_k and object o_2 of observation t_{k+1} are the same for some k if the Euclidean distance $d(o_1, o_2)$ between the center points $c_{t_k}^{o_1}$ and $c_{t_{k+1}}^{o_2}$ are minimal:

$$\min_i d(c_{t_k}^{o_1}, c_{t_{k+1}}^{o_n}).$$

We would like to note, that other methods, such as the iterative closest point (ICP) method [15] could also be applied to extend or replace the described matching procedure. After identifying the cells through frames, the previously described outlier filtering method can be applied to filter the data, using the calculated object IDs.

3.4. Statistical analysis

To analyze the behavior of the model for different inputs, a machine learning-based model for regression was used. As a proof of concept, a classical multi-layered neural

network was trained to predict the area size of a cell at a given time for a given set of input parameters.

The input layer receives the normalized time and input features, a total of 11 features. The shallow network architecture (visualized on Fig. 2) consists of 7 hidden layers with parameter numbers between 100 and 25, followed by an output layer with one single neuron with a "leaky" Rectified Linear Unit activation function to predict the area. For training, the state-of-the-art Adam optimizer [10] was used to minimize the mean squared error. Results showed a mean absolute error of 13.5%.

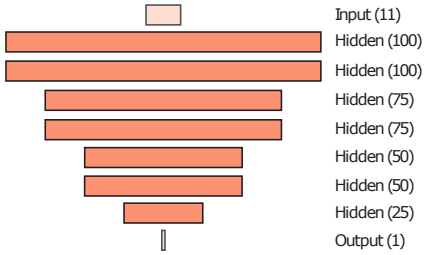

Figure 2: The structure of the fully connected neural network. The 11 input parameters are followed by a total of 7 hidden layers, and one single output is calculated. The activation functions are common ReLU activations, while the output neuron has a "leaky" ReLU activation to support negative values.

The relatively high error rate can be explained by the dependency of the adjacent cell objects, and the relatively small size of training data. We also would like to point out, that our future plans include the analysis of convolutional neural networks (CNN) to incorporate the features of adjacent cells, as well as recurrent neural networks (RNN) to take the past states of the objects into account. This paper shows a proof of concept, and the base idea of the procedure.

3.5. Input inference

The presented prediction technique of the output values is used as a basis of an input inference method. The pre-trained network is extended with a first layer with random trainable weights, while all other layers, including the trained parameters, are unchanged. As it is shown in Fig. 3, the neuron number of the inserted layer equals the number of input parameters, while other parts of the network – including the weights – remain unchanged.

For a given output value, all input values are set to constant 1, and using these values a few steps of back-propagation is done. The given values are flown through the network to get a prediction, the error is calculated from the expected value, this loss is then back-propagated to the first layer, and the weights are changed accordingly. A few of these training steps are done until the error rate descends to

a fixed value.

Afterwards, the final step is to calculate the so-called expected input parameters from the trained parameters. Layer weights can be represented as a matrix W with a size of 11×11, as all the 11 neurons of the layer are connected with all the 11 neurons of the input, resulting in 11 rows of weight vectors of length 11. The layer also has bias values for each neuron, resulting in a total of 11 values represented by vector b.

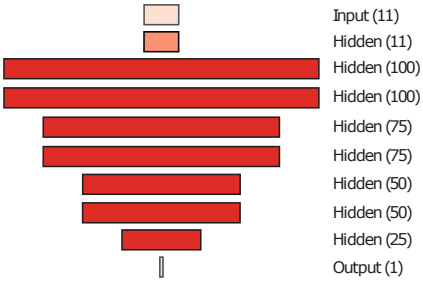

Figure 3: The structure of the input inference model. The hidden layers colored with red are "frozen", non-trainable, the weights are pre-trained. The fully connected layer inserted as the first layer is the only trainable layer in the network.

Based on the classic formula of activation, the sum of each row i is calculated as

$$\sum_{j=1}^{N} w_{i,j},$$

resulting in a vector of 11 values. This will be multiplied by the input values, which are constant ones, therefore, vector w is unchanged and the bias is added:

$$z_i = \sum_{j=1}^{N} w_{i,j} + b_i.$$

In case the expected input values are in the domain of $[0, 1]$, a sigmoid activation function could be applied as

$$a_i = \mathrm{sigm}(z_i)$$

to get the expected input values for a given output.

It is notable, that the function of the neural network is non-injective, the inputs can not be inverted, the inputs can only be inferred, while a set of multiple solutions might exist, the method defined here only results in the input set with the lowest error rate.

4. Results

Since the large number of possible features, we inspected the resulting dataset by performing a hierarchical clustering on the attributes and visualizing their dependence on a correlation heatmap (see Fig. 4). To build the dendrogram of the attributes, an agglomerative clustering process was used with Ward's minimum variance method and Euclidean distance function as a measure of dissimilarity. As it was revealed, the measured shape attributes are highly interconnected, therefore it is possible to reduce the number of shape features to a much smaller subset. We selected Area, Compactness and FormFactor as they fall into separate classes based on the hierarchical clustering.

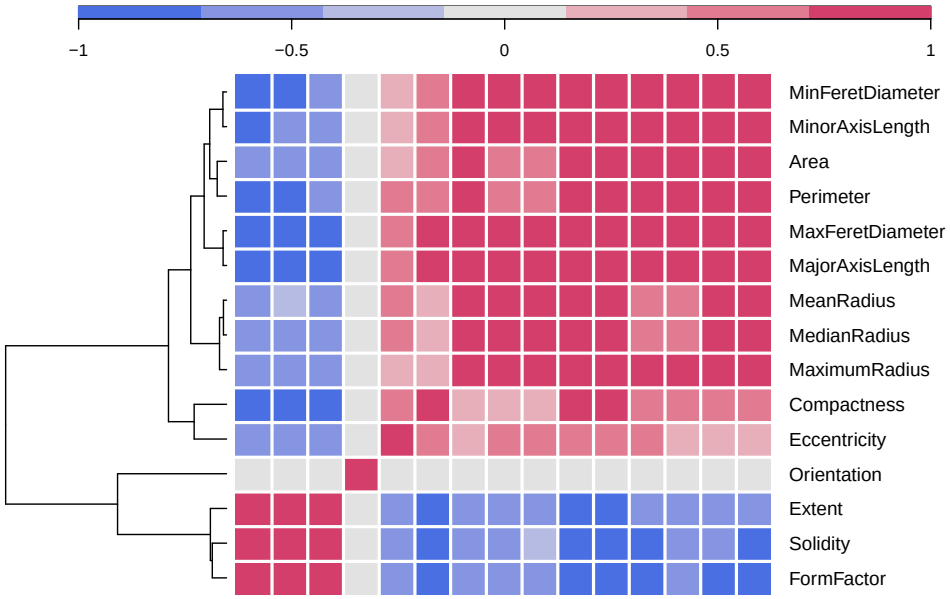

Figure 4: Hierarchical clustering and correlation heatmap of the dataset. As attributes are highly interconnected, a well-chosen subset is able to catch most of the differences in the shape features of the cell aggregates.

The final dataset contained approximately 250,000 records. The multi-layered neural network is trained on these data to predict a selected shape feature. For demonstration purposes, we chose Area, as it is easy to interpret and it is a good representative element of the feature set. During training, 70% of the original dataset was used for training, and the remaining 30% for validation, to detect overfitting.

The training of this baseline model was evaluated with a separated test set of 1522 synthetic test records: the measured average difference between the expected

and the predicted area size was 12.6%. After the model was trained, the parameters were used to infer the input variable, now based on the outcome area size of the cell aggregate, using the method defined in section 3.5. After freezing the original weights, training affects only the parameters of the appended first layer. Training concludes when the measured loss stops descending; during our experiments this happened with a loss value near zero. During our experiments, the inferred inputs were fed back to the original model, and the predicted area size is compared with the expected.

During the experiments, we created a novel embedding structure, where a selected input parameter can be predefined, and are not affected by training. This defined method is easily extendable, allowing the researchers to predict some input values for fixed input parameters. Future plans include the extension of the model to examine the behavior of multiple cells and time-series based on the previously mentioned CNN or RNN structures.

Our initial aim was to demonstrate the possibility of relating simulation input parameters to measured shape features. We inspected this relation on our simulation data. Using principal component analysis, we concluded that the two most significant input parameters are ε and k, i.e. the interaction potential well depth ("stickyness") and the shape parameter of the step size distribution ("velocity").

As depicted in Fig. 5, stronger attractive forces between the agents (larger ε values) results in smaller but more circular cell aggregates. This observation was confirmed statistically by performing a two-way ANOVA which showed that ε has a clear effect on those features ($p < 0.001$). On the other hand, we found that the velocity has statistically significant effect only on the FormFactor feature ($p < 0.001$) but not on the Area.

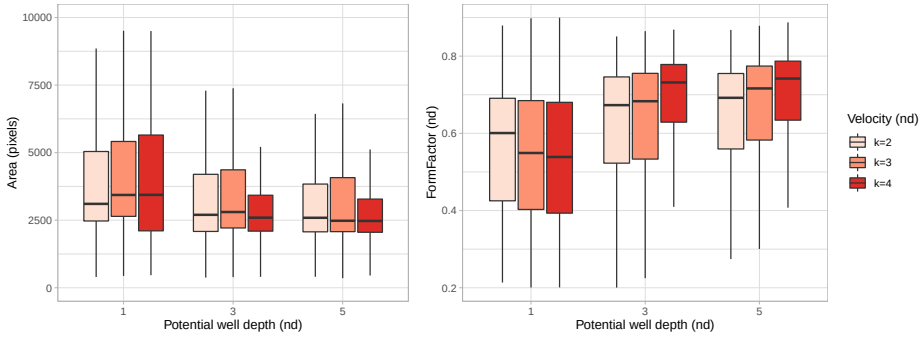

Figure 5: The effect of altered input parameters on shape features. There is a relation between the distribution of Area (left) and FormFactor (right) and the input parameters ε and k.

5. Conclusion

In this paper, we proposed a procedure to generate a dataset of colony formation process of simulated cell cultures. Using a simplified version of an existing agent-based model, multiple simulations were executed parallelly and the results were analyzed using an image processing pipeline. The details of the agent-based model, the feature extraction method as well as the data filtering technique were also discussed. Following the statistical analysis of the results, a proof-of-concept regression method was presented to predict an output of the simulation, based on input parameter data. Built on the regression model, an input inference method is introduced to produce possible input parameters from the received output.

As a preliminary result, we found that the measured shape features of the artificial cell aggregates (such as the area or the circularity) can be predicted by only a few input parameters, namely the simulated time t which is simple to understand, but also the adhesion energy ε and velocity distribution shape parameter k which both belong to the motility of a living cell. This observation is consistent with other published results. Our experiments show, that the proposed method – extended by sensitivity analysis and a precisely defined search – could be promising in case of parameter search for simulated environments.

We believe that the proposed procedure could serve as a useful base for creating and testing more accurate prediction models based on machine learning or developing advanced statistical methods that reveal some non-trivial patterns of *in vitro* cell pattern formation. This concept could also contribute to researches aiming to predict some hard-to-measure properties of living cells by creating and fitting a model with known parameters to the real phenomenon.

Acknowledgements. Dániel Kiss was supported by UNKP-18-3-I-OE-94 New National Excellence Program of the Ministry of Human Capacities. Anna Lovrics was supported by OTKA PD124467 grant. The authors acknowledge the financial support by the Hungarian State and the European Union under the EFOP-3.6.1-16-2016-00010 project. The authors thankfully acknowledge the support of the Doctoral School of Applied Informatics and Applied Mathematics of Óbuda University.

References

[1] M. Baraldi, A. Alemi, J. Sethna, et al.: *Growth and form of melanoma cell colonies*, J. Stat. Mech. (2013), doi: https://doi.org/10.1088/1742-5468/2013/02/P02032.

[2] D. A. Beysens, G. Forgacs, J. A. Glazier: *Cell sorting is analogous to phase ordering in fluids*, Proceedings of the National Academy of Sciences 97.17 (2000), pp. 9467–9471, doi: https://doi.org/10.1073/pnas.97.17.9467.

[3] A. Brú, S. Albertos, J. L. Subiza, J. L. García-Asenjo, I. Brú: *The Universal Dynamics of Tumor Growth*, Biophysical Journal 85 (2003), pp. 2948–2961, doi: https://doi.org/10.1016/S0006-3495(03)74715-8.

[4] J. Buceta, J. Galeano: *Comments on the Article "The Universal Dynamics of Tumor Growt" by A. Brú et al.* Biophysical Journal 85 (5 2005), pp. 3734–3736, doi: https://doi.org/10.1529/biophysj.104.043463.

[5] A. Carpenter, T. Jones, M. Lamprecht, et al.: *CellProfiler: image analysis software for identifying and quantifying cell phenotypes*, Genome Biology 7.100 (2006).

[6] D. Drasdo, S. Hoehme: *Individual-based approaches to birth and death in avascular tumors*, Mathematical and Computer Modelling 37 (2003), pp. 1163–1175, doi: https://doi.org/10.1016/S0895-7177(03)00128-6.

[7] D. Drasdo, S. Hoehme, M. Block: *On the Role of Physics in the Growth and Pattern Formation of Multi-Cellular Systems: What can we Learn from Individual-Cell Based Models?*, Journal of Statistical Physics 128.287 (2007), doi: https://doi.org/10.1007/s10955-007-9289-x.

[8] D. Drasdo, R. Kree, J. McCaskill: *Monte Carlo approach to tissue-cell populations*, Physical Review E. 52.6 (1995), pp. 6635–6657, doi: https://doi.org/10.1103/PhysRevE.52.6635.

[9] M. A. C. Huergo, M. A. Pasquale, P. H. González, A. E. Bolzán, A. J. Arvia: *Dynamics and morphology characteristics of cell colonies with radially spreading growth fronts*, Phys. Rev. E. 84 (2011), doi: https://doi.org/10.1103/PhysRevE.84.021917.

[10] D. Kingma, J. Ba: *Adam: A method for stochastic optimization*, arXiv preprint arXiv: 1412.6980 (2014).

[11] D. Kiss, A. Lovrics: *Performance analysis of a computational off-lattice tumor growth model*, in: IEEE 30th Jubilee Neumann Colloquium, 2017, pp. 141–146.

[12] P. Macklin, M. Edgerton, A. Thompson, V. Cristini: *Patient-calibrated agent-based modelling of ductal carcinoma in situ (DCIS): From microscopic measurements to macroscopic predictions of clinical progression*, Journal of Theoretical Biology 301 (2012), pp. 122–140, doi: https://doi.org/10.1016/j.jtbi.2012.02.002.

[13] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, E. Teller: *Equations of State Calculations by Fast Computing Machines*, Journal of Chemical Physics 21.6 (1953), pp. 1087–1092, doi: https://doi.org/10.1063/1.1699114.

[14] M. J. Plank, M. J. Simpson: *Models of collective cell behaviour with crowding effects: comparing lattice-based and lattice-free approaches*, J. R. Soc. Interface 9 (2012), pp. 2983–2996, doi: https://doi.org/10.1098/rsif.2012.0319.

[15] D. Stojcsics, Z. Domozi, A. Molnár: *Iterative Closest Point Based Volume Analysis on UAV Made Timeseries Large-scale Point Clouds*, in: IEEE 16th International Symposium on Intelligent Systems and Informatics: SISY 2018, 2018, pp. 69–74, doi: https://doi.org/10.1109/SISY.2018.8524645.

[16] D. Walker, J. Southgate, G. Hill, et al.: *The epitheliome: agent-based modelling of the social behaviour of cells*, Biosystems 76.1–3 (2004), pp. 89–100, doi: https://doi.org/10.1016/j.biosystems.2004.05.025.