# Emulating neuronal activity of in-vitro cortical cultures using two complex models based on evolutionary principles

**Kommentert [SL1]:** Decide for a fitting title in the end.

Elsevier[1]

*Radarweg 29, Amsterdam*

*Elsevier Inc[a,b], Global Customer Service[b,2*]*

*[a] 1600 John F Kennedy Boulevard, Philadelphia*
*[b] 360 Park Avenue South, New York*

## Abstract

Place abstract here.

*Keywords:* Emulation, Neuronal activity

## 1. Introduction

Neuronal activity has been the inspiration of several technologies and concept in the modern society. Its complex behaviour and dynamical structure have been researched over several decades and is still a hot

[1] Since 1880.

[*] This is to indicate the corresponding author.
Email address: support@elsevier.com (Global Customer Service)
URL: www.elsevier.com (Elsevier Inc)

topic for researchers within several fields [1]. Advancements in technology and science are continuously contributing to increasing our understanding within the field, but there are still many challenges faced by researchers when trying to understand it [2].

The microscopic and sensitive nature of neurons makes it hard to observe and measure neuronal cultures without interfering their environment. A way of researching their behaviour is by investigating such cultures inside controlled environments to be able to both regulate its surroundings and observe its dynamic. This approach makes it possible to keep the neurons alive for a given amount of time but can also be challenging to work with [3].

Advancements in computer science has led to possibilities of developing models of complex systems with functionalities that can simulate the properties of such cultures. By creating such complex models to mimic the features of the desired systems, interesting observations can be seen, and the challenges connected to the cultures sensitive nature are removed.

In this work, two complex models are developed to emulate the behaviour of an in-vitro cortical culture. The first model is a Cellular Automata (CA) model that with a simplistic set of rules can generate complex patterns with similarities to the observed culture. The second model bases on a network structure with differences from the first model. Based on principles from evolutionary computation, both models are evaluated and evolved based on their abilities to emulate the existing data. By using two different approaches for emulation, information regarding model design and performance can be observed and discussed.

**Kommentert [HD2]:** Which might resemble?

**Kommentert [HD3]:** This should be rephrased

**Kommentert [SL4]:** Rephrase

The dataset that this work is built on comes from the 2006 study conducted by Wagenaar et al. [3] and their observation of in-vitro neuronal cultures. The data is recorded using a micro-electrode array (MEA) and preprocessed such that it includes the time of each detected voltage spike related to their electrode number.

**Kommentert [HD5]:** This part fits very well with the paragraph about how to research neuronal behaviour

## 2. Method

### *2.1 Development process*

The process of developing models for emulation of neural activity has included interesting challenges which has formed the path and the outcomes of this work. To understand the workings of evolutionary algorithms combined with complex systems, different approaches have been explored. In the decision of model selection, the data provided by Wagenaar et al. [3] were plotted to better understand the problem domain. This plotting was conducted using an 8-by-8 grid of cells from the PyCX framework developed by Sayama [4], as well as a standard raster plot. First a model based on Cellular Automata (CA) were developed to both investigate how the evolutionary algorithm should be implemented in such a system and whether it would be a possible model worth further investigation. This approach gave interesting results and challenges, and led to a desire to further explore how it could be developed.

While working with the CA model, limitations regarding parameter implementation led to an interest of exploring a different model in parallel to be able to compare different types of results. With the option of attaching parameters to nodes and edges, a model based on the NetworkX python package was developed.

## *2.2 Model design*

The two models presented in this work were designed to emulate the activity in the cortical culture. Each model aims to generate data which has similarities to the reference dataset [3]. With different structures and functionalities they are both working to generate this data and evaluated continuously.

### *The CA model*

**Kommentert [SL6]:** Wait with this until Elias have placed what he has written.

The CA model, a relatively simple approach to model complex systems, while the other was a slightly more advanced network model.

**Kommentert [HD7]:** Find synonym or drop

Since CA models are following a strict set of rules, and in this case are designed with binary states, the objective of emulating the neuronal behavior

very well in its objective to emulate the neuronal behaviour. However, it was a good place to start to get a better insight into the programming and the relation between the model and the EA. d to work and the other is a network model. two models are different approaches where the subsequent state of the cells in the CA model is strictly decided by the states of the cells in the

The network model is slightly more advanced in the way that it is not uniform in its connections and can therefore be more difficult to analyse [5].

**Kommentert [HD8]:** This part seems rather “malplassert ut” after the new introduction to the chapter was written and the header was formed

### *Network model*

The network model design is replicating the structure of the MEA and the spatial positioning of its electrodes. This means that the graph consists of sixty nodes that are connected in a regular structure, according to graph theory presented by Poli et al. [6], however, their definition of a regular network structure is not as rigid as Sayama[5]. He states that all nodes must have the same degree, and in our case, the lateral nodes have a lower degree than the centre nodes, which means that the graph is only relatable to a regular structure. The structure is visualised in Figure 1.

**Kommentert [HD9]:** Smaller?

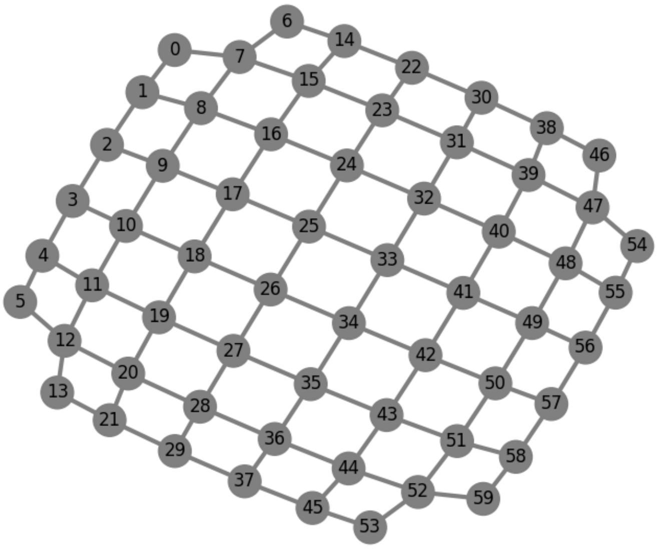


**Figure 1 Layout of the network model. It is designed in the same manner as the MEA with fixed connections between the closest neighbours.**

The interaction between the nodes is following the principle of structural connectivity, also called anatomical connectivity. This type of connectivity indicates that neurons are physically connected and interact with each other, such as chemical or electric synapses. The nodes in the network model are linked together in such a fashion and simplify how the nodes can interact with each other. In addition, we assume that every node in the model is connected to the same network. [6]

**Kommentert [HD10]:** Is this part correct when we have used undirected edges?

**Kommentert [HD11R10]:** Write about undirected edges

The functionality of the nodes is based on a simple representation of how an action potential is generated in a neuron, called the leaky integrate-and-fire model. Integrate-and-fire models work in such a way that the neuron unleashes an action potential when the potential inside the neuron reaches the threshold value, and then the potential is reset to its resting potential. In addition, there are mechanisms in a neuron that seek equilibrium and if the neuron has increased potential inside the cell it will try to regain its resting potential, and this is why the model is called "leaky".[7]

**Kommentert [HD12]:** Other word?

The leaky integrate-and-fire model is implemented in the nodes by programming five attributes: three constants, a voltage threshold, a leaking constant and a refraction period, and a voltage potential variable, as well as a spiking output. The relationship between the attributes can be seen in Figure 2. The refraction period implemented in the nodes is a simplified version where it is just a time constant instead of a voltage potential that will increase from negative potential and back to resting potential in the same amount of time. There is one additional attribute programmed into the node, which is a probability of self spiking. Without this parameter, the neuronal activity would eventually die out.

**Kommentert [HD13]:** This was long and kronglete

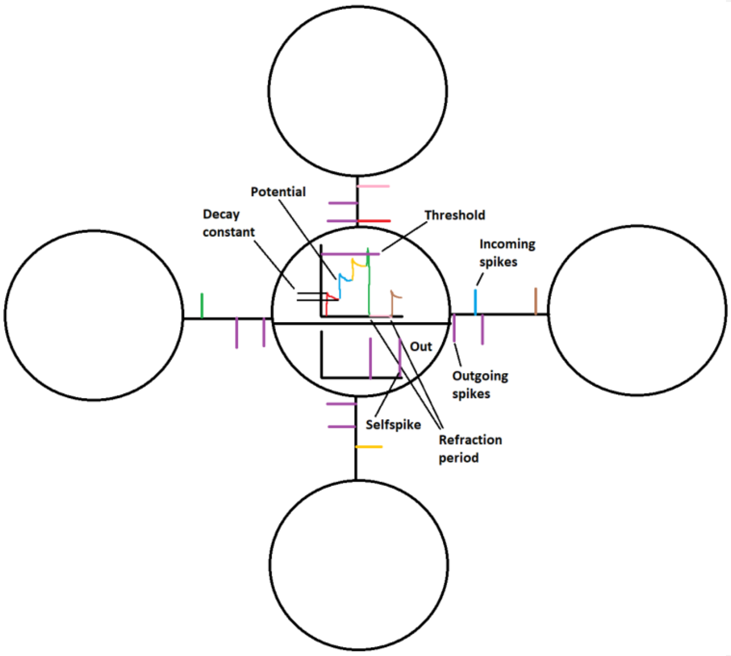


**Figure 2 Illustration of how the leaky integrate-and-fire model is working.**

The aim of using two different methods is motivated by the potential different results they can provide and to make it possible to analyse the outcome in the light of one another.

**Kommentert [HD14]:** Should this be implemented in the introduction to the method chapter?

The network has been developed both with and without weighted edges, where the weights are either updated in a simple way that reminds of Hebbian plasticity(Hebb, D.O. (1949) The Organization of Behavior) or as a part of the genotype in the evolutionary algorithm (EA). The network was developed with the help of the networkx framework.

**Kommentert [HD15]:** First abbreviation?

Comment whether the model is rule-based or descriptive.

### *2.3 Evolutionary algorithm*

The optimization of both of the models is provided using principles from evolutionary computation. With an asexual, elitistic evolution [8] both models evolve by choosing the individual with the best fitness score from each generation to be the parent of the following generation. The fitness score is based on the lowest difference between the total number of spikes in the original data and the data generated by each individual. Fitness is represented in the same way in both models, but the internal representation is different. From the CA model, the genotype is … In the network model on the other hand, the genotype includes four genes: a decay constant, threshold value, refraction period, and a probability of self-spike. For each mutation, these parameters are changed with a probability of 0.01. The population size of each generation is determined based on the number of CPUs within the device used for data generation.

**Kommentert [HD16]:** Initialise:
-Number of individuals within a population has been equal to the number of CPU we can run, because of the immense amount of data
-Genotype: Decay constant, threshold, refraction period, probability of selfspiking
Mutation:
-Probability 0.01
Fitness function:
-Diff = abs(Len(original data) – Len(generated data))
Phenotype: generated spikes

**Kommentert [SL17]:** Is the genotype in the CA the rules that are implemented or what? But these rules have nothing to do with the ordinary CA rules right?

**Kommentert [HD18R17]:** The genotype is a totally (?) random binary string from 0-255. It is the state of the 8*8 neighbourhood that decides which of the 0-255 positions it should take as its next state

## 3. Results

The models designed in this project was developed to emulate the behaviour of a real neuronal cortical sample. A way of evaluating the performance of the networks is to look at how well they perform according to the fitness function in the EA. However, the fitness score

**Kommentert [HD19]:** Rephrase

does not tell the whole story on its own and it is necessary to use other tools to compare the phenotype, i.e. the spikes generated by the complex system model, with the original data.

The network model has iterated through several versions where different approaches towards the edges' weights have been investigated. In the beginning, the edges were unweighted, but the EA struggled to find the correct parameters and the model did not manage to acquire a good fitness score. Consequently, weights were added as a property to the edges and included as a part of the genotype for the EA to alter through mutation. This approach was a better fit between the model and its parameter range and the defined fitness function. In the final version, the weights were added as a variable responding to the number of times two neighbouring nodes interact with each other and make the other spike. This approach was also successful concerning the fitness score. The fitness scores of the different approaches are displayed in Figure 3.

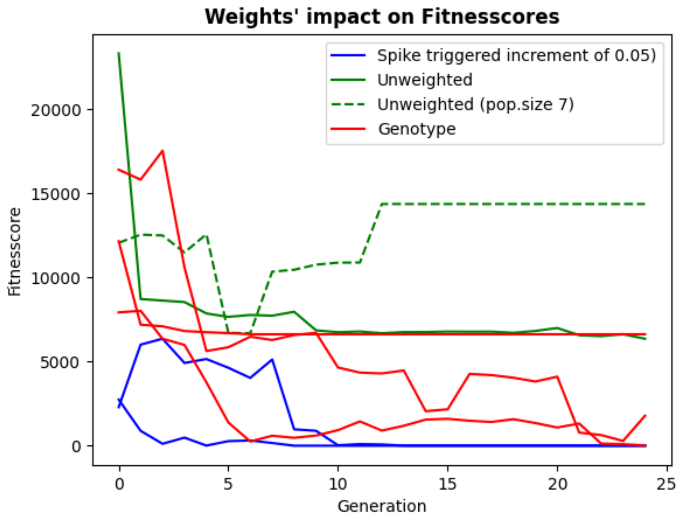


**Figure 3 Overview of how the fitness score is affected if the edges' weight configuration is changed. The blue lines represent a small increment to the weights each time two neighbours spike together. The weights are reset to their initial state in each generation.**

**Kommentert [HD20]:** -Add another GenWeight graph
-Check if there are other phenotypes that has more burst pattern

The behaviour of the model can be seen in Figure 4, where the spikes generated by each node is plotted as a function against time. In the same figure, the neuronal activity in the reference culture is plotted as a function of spikes produced by each electrode against time. This setup gives a good impression of how similar the artificial neuronal activity is to the authentic neuronal activity.

**Kommentert [HD21]:** Should this be changed?

When investigating the behaviour of the network model in the bottom diagram in Figure 4 it appears relatively uniform or repetitive, especially compared to the cortical neuronal behaviour, as seen in the top diagram in the same figure. It can be observed a burst pattern in the reference data which was not recreated by the designed model. The difference is more clearly expressed when comparing the ASDR between the data, as seen in Figure 5.

Although the network model was not able to emulate the general behaviour of the neuronal data, there are some similar features. Some electrodes seem to be much more active than others and appear to be spiking "all the time" in the diagram indicated by what looks like solid lines. These are just many spikes spawning closely together to form a line, and these lines can be seen in the generated data as well. Although, there are several overactive nodes than electrodes.

**Kommentert [HD22]:** rephrase

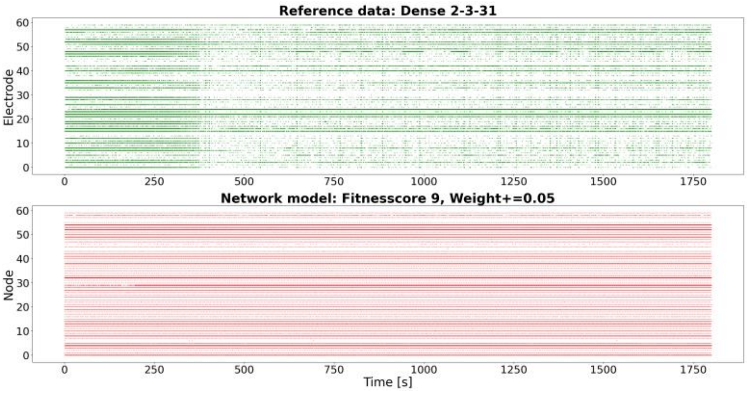


**Figure 4 Comparison of behaviour between the cortical neuronal activity and the artificial activity generated by the EA and the network model. The generated data had a fitness score of 9.**

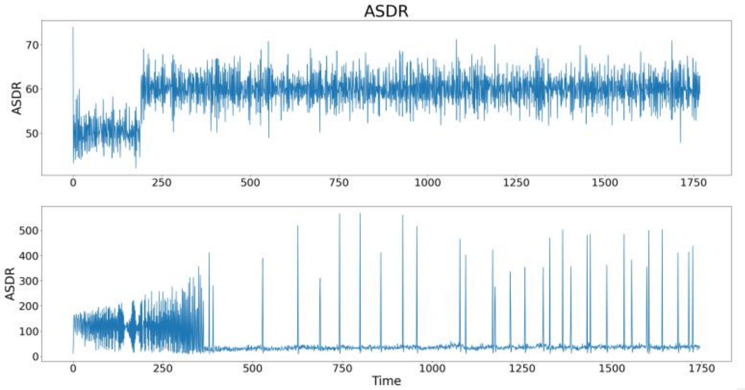


**Figure 5 Array-wide Spike Detection Rate (ASDR) comparison of the network model and the reference data. The reference data represented in the bottom diagram has a clear burst pattern which is not present in the generated spike data. The diagrams are created with time bins of one second, which means that it is the number of spikes/second that is displayed.**

**Kommentert [HD23]:** Make title to both diagrams to make it clear what is displayed. Switch the order so that it is consequent which is reference data and generated data

## 4. Discussion

**Kommentert [HD24]:** Possible discussions:

- -Why didn't our models perform better?
  - ○Bad/simple fitness function (could look at time bins)
  - ○Simple network structure
  - ○Wrong parameter range
  - ○Simple EA (no cross over, elitistic?)
  - ○Several ways of performing well in fitness: e.g. only self spikes and no volt potential spikes, only one node spiking really fast
  - ○Bad programming? E.g. I don't think I have a backup plan if it gets stuck in a local minimum
- -Observed performance difference according to the number of individuals in the populations. (Good, better, best: 4,7,11).
- -Is the network model really a network model or is it a CA with real inputs?
  - ○Pro CA:
    - ▪Fixed connections to the closest neighbours (just like Moore)
    - ▪The simulation is running through each node looking at it's state and updating it according to the neighbours previous state.
  - ○Against CA:
    - ▪Have a simplistic version of Spiking neuron (leaky and integrate-spike) which means that the output is not strictly determined by the rules
- -Discuss what impact weights have on the performance.
- -Discuss why we chose to do 2 models?
- -Discuss the model design?
- -Discuss our EA
  - ○Why did we choose an elitistic algorithm?
  - ○Why have we designed the fitness function that way?

The models developed in this work have the ability of generating spike data that emulates the activity in the cortical cultures to some extent. However, neuronal activity is complex and the modelling of its behaviour requires models with the ability of matching this level of complexity [7]. The simplistic workings of the two models limits the complexity of the outcome which leads to possibilities in further development.

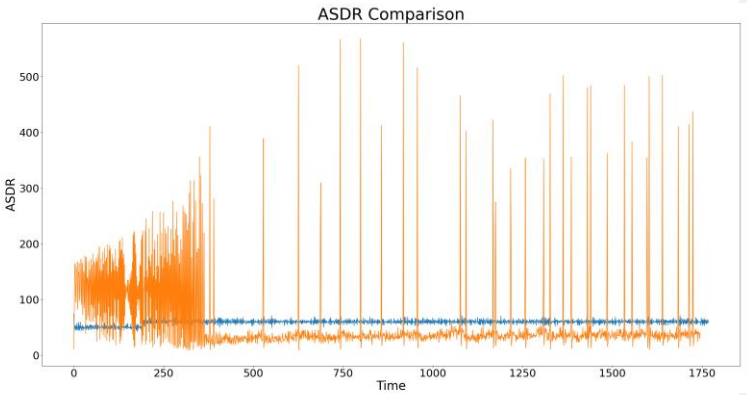


**Figure 6 ASDR of the generated data from the Network mode and the reference data.**

**Kommentert [HD25]:** This figure with ASDR of generated data and original data shows how the model managed to get so good fitnesscore without producing a nice bursty pattern.

### *4.1 The complex models*

The two models designed in this work have different ways of emulating the data from the dataset. They both generate results that include the number of spikes over a given time which is a relevant parameter to observe in neuronal culture analysis as it is a foundation for the ASDR measurement [7]. These generated spikes are distributed over a time and illustrates the total behavior of the system. However, this parameter is not evaluated in this context due to the low level of pattern complexity in the generated data. By implementing more features related to the determination of the state of the nodes in the models, the generated spike patterns could provide other interesting information and provide results of higher complexity. Such features could be derived from other well-known spiking neuron models like the integrate-and-fire models [4, p.162-166]. The network model presented here include some of these features as the genotype includes a refraction period, an inter-spike probability, leakage voltage and a threshold value. By combining these parameters, the model provides some of the complexity from the integrate-and-fire models, but still have the potential of more complexity.

*Inhibitory vs excitatory*

One relevant parameter to add would be the difference between inhibitory and excitatory nodes. This would emulate the operation of a neurotransmitter with the capability of increasing or decreasing the action potential in a neuron [4, p.162-166]. This could influence the spike patterns as they would expand the spike rate in some of the nodes and prevent others from spiking.

*Connectivity*

Another interesting parameter to add would be the distribution of connected nodes. The CA and the network model have similarities as they are both connected to all of their neighboring cells and nodes in a *regular* way [6]. This is also known as a grid and leads to a spatially oriented distribution of connectivity. By facilitating for connectivity across the grid, using e.g. random or scale-free connectivity patterns between the nodes [6], other spike patterns might evolve from the models.

**Kommentert [SL26]:** Move the discussion about the regular connections here. Think it is in the result chapter or in the model design chapter.

*Weighted edges*

The connectivity can also be influenced by different weighting of the neural paths between the nodes. This would emulate the variations in synaptic connections in neuron communication [1]. In the last stage of improvement of the network model in this work, weights were added to the edges to investigate the potential influence on the patterns. The reason for adding these weights based on the quote by Donald Hebb from 1949 regarding the increased connectivity between neurons of repeated and persistent firing [6, p. 62]. Implementation of the weights had clear impact on the outcome as the fitness score of this model reached a higher value than the ones without the weights. However,

**Kommentert [SL27]:** Comment the negative aspect of implementing the weights in the edges.

**Kommentert [HD28R27]:** We don’t know what the weights actually did. Complexity is increased significantly giving the EA a harder time.

, or by even more realistic models such as the Hudgkin-Huxley model [7]. or single-compartment type of models

**Kommentert [SL29]:** Change word. Mean “more like in real life” without using realistic.

By implementing features from the commonly used integrate-and-fire approach, features such as

**Kommentert [HD30]:** We have used a leaky integrate-and-fire approach, but it other models are more advanced.

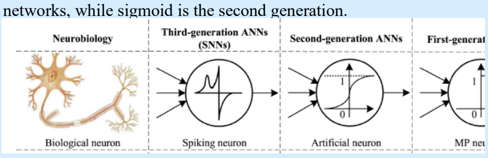
For improvement of the CA model changes in

Is spiking neural network something to include?

Sigmoid implementation as Activation function is worth mentioning.

.

**Kommentert [HD31]:** I think this is outdated compared to leaky integrate-and-fire model. If I’m right integrate-and-fire is a Spiking neural network which is the third generation of artificial neural networks, while sigmoid is the second generation.



CA has a more MP cell-model. Network have a more perceptron approach.

## 4.2 *Evolutionary Algorithm*

Evo algo

### *4.3 Fitness function*

he fitness function

By

taking the ASDR into consideration

**Our model**

Type: Asexual without cross-over. Phenotypic. Elitistic – We choose the offspring with the best fitness (a truncation).

**Basic information:**

Phenotype: A candidate solution for the considered problem.

A program in a given programming language.

A set of numerical Parameters.

Genotype: Internally represented.

Both models are of an asexual type without any cross-over from parent generation.

Considerations when choosing a model:

- How to select which individuals to survive
- How to select which parents to reproduce

- Type of selection:
- Uniform (neutral) selection
- Fitness proportional selection
- Rank-proportional selection
- Truncation selection
- Tournament selection

Is the population diverse enough?
“no” – Too much exploitation -> local minimum.
“yes” – no drawbacks.

Finding the right level of abstraction for a problem is an acquired skill and the results are highly problem dependent.

From [10]:

Many authors have argued that diversity is beneficial in evolutionary optimization processes. Comment: This could be mentioned when criticising our models. They are very elitistic and lack diversity.

But optimization algorithms use a fitness function that evaluates the goodness of each candidate solution with respect to a given task—that is, the whole ecosystem is indirectly modeled through its effects, and only few very specific facets are taken into consideration. The general inability to exploit environmental niches noted by Holland could be explained with the absence of such natural spots to survive in

## 5. Conclusion

## 6. Acknowledgements

Acknowledgements and Reference heading should be left justified, bold, with the first letter capitalized but have no numbers. Text below continues as normal.

## 7. References

[1] C. C. Aggarwal, *Neural Networks and Deep Learning*. Cham: Springer Nature, 2018.

[2] W. Gerstner and W. M. Kistler, “Introduction,” in *Spiking Neuron Models*, Cambridge University Press, 2002, pp. 1–28.

[3] D. A. Wagenaar, J. Pine, and S. M. Potter, “An extremely rich repertoire of bursting patterns during the development of cortical cultures,” *BMC Neurosci.*, vol. 7, pp. 1–18, 2006, doi: 10.1186/1471-2202-7-11.

[4] H. Sayama, “PyCX: a Python-based simulation code repository for complex systems education,” *Complex Adapt. Syst. Model.*, vol. 1, no. 1, pp. 1–10, 2013, doi: 10.1186/2194-3206-1-2.

[5] H. Sayama, *Introduction to the Modeling and Analysis of Complex Systems*, vol. 41, no. SUPPL. 1. 2015.

[6] D. Poli, V. P. Pastore, and P. Massobrio, “Functional connectivity in in vitro neuronal assemblies,” *Front. Neural Circuits*, vol. 9, no. OCT, pp. 1–14, 2015, doi: 10.3389/fncir.2015.00057.

[7] P. Dayan and L. F. Abbott, *Theoretical Neuroscience*. Cambridge, Massachusetts. London, England: The MIT Press, 2005.

[8] K. A. De Jong, *Evolutionary Computation*. Cambridge: The MIT Press, 2006.

[9] F. Attneave, M. B., and D. O. Hebb, “The Organization of Behavior; A Neuropsychological Theory,” *Am. J. Psychol.*, vol. 63, no. 4, p. 633, 1950, doi: 10.2307/1418888.

[10] G. Squillero and A. Tonda, “Divergence of character and premature convergence: A survey of methodologies for promoting diversity in evolutionary optimization,” *Inf. Sci. (Ny).*, vol. 329, pp. 782–799, 2016, doi: 10.1016/j.ins.2015.09.056.