Pannon Egyetem

Műszaki Informatikai Kar

Rendszer- és Számítástudományi Tanszék

Programtervező informatikus MSc

**DIPLOMADOLGOZAT**

**Mikrobiális növekedés adatainak elemzését támogató szoftver fejlesztése**

**Pillér Attila**

Témavezető: Dr. Fogarassyné dr. Vathy Ágnes

Table of contents

[1. Introduction 5](#_Toc120441082)

[2. Scientific background 7](#_Toc120441083)

[2.1. Experimental setup 7](#_Toc120441084)

[2.2. About the *Saccharomyces cerevisiae* 8](#_Toc120441085)

[2.3. Growth of Multicomponent yeast populations 9](#_Toc120441086)

[2.4. Interactions between yeast cells 10](#_Toc120441087)

[3. Objective of the Software 11](#_Toc120441088)

[4. Used technologies 12](#_Toc120441089)

[4.1. PyCharm, Python 12](#_Toc120441090)

[4.1.1. Keras 13](#_Toc120441091)

[4.2. Visual Studio, C# 14](#_Toc120441092)

[4.2.1. LiveCharts2 14](#_Toc120441093)

[5. Current progress 15](#_Toc120441094)

[6. Plans for next semester 16](#_Toc120441095)

[7. References 17](#_Toc120441096)

# Introduction

Nowadays cell-cell communication has been the topic of a numerous ongoing research since the understanding of interactions between cells contributes to solving complex molecular processes and functions. Studying these interactions with the help of yeast cells gives us a clearer picture not just about the behavior of multicomponent microbial populations in nature, but also about how these correlates with functions in higher organisms. The local interactions in these environments can greatly influence the future of the different populations.

During the experiments, which focus on investigating the behavior of the strains of *Saccharomyces cerevisiae* in mixed cultures a lot of data is generated that must be properly processed and analyzed. Doing this manually takes up a big amount of time, and it’s also much easier to make mistakes when working with this amount of data.

This is where a software can greatly increase the efficiency of this process. With an easy-to-handle user interface the evaluation of the results does not require any programming skills from the user, and with the quick data processing and precise data analyzing the amount of time spent will decrease immensely.

The processing of the experimental data with this software can be divided into two parts. The first part of the application which is implemented in Python is used to read the provided Excel files containing the data, check for inconsistency and if possible, correct them, then perform the analysis and at the end transform them into a format that can be easily accessed later as well.

During these steps when I check for errors in the data, I will be able to make good use of Artificial Intelligence and LSTM networks, since from previous measurements it is possible to predict a range in which the data should fall in. For the analysis there are certain statistical calculations that must be done, and also some characteristic growth parameters have to be calculated.

The second part is the visualization of the processed data. For this I have used Visual Studio, C# WPF application. The main focus of the UI was to be user friendly and easy to configure. The growth data will be shown on diagrams with the help of LiveCharts2 extension. It will be possible for the user to select the strains that they want to see on the diagram and to check the corresponding statistics calculated previously. There will also be a way to save some personal settings such as: colors for each strain, default file locations, which strain was selected the most often.

Another goal of the software would be to find a solution to predict future interactions in multicomponent populations by using the previously calculated growth parameters of individual strains. This would also be a huge advantage and help to design the following laboratory experiments.

# Scientific background

Research about intracellular processes has always been popular, and our knowledge of them is also steadily increasing, but even after all these years there is still new information that can be discovered. That is one of the reasons why there are more and more researches done about the interactions between cells.

## Experimental setup

The software that I am working on is also made for the purpose of supporting the analysis of the data produced during laboratory experiments. These experiments are done using strains of *Saccharomyces cerevisiae.* During these experiments all the strains are geneticin resistant to avoid contamination and also fluorescently labeled with green (EGFP) and red (mCherry) fluorescent proteins[1]. This is what makes it possible to differentiate them when they are put into mixed cultures. With this labeling we have three variations of the same strain, named as the following:

* Strain A GFP: a strain labeled with green fluorescent proteins
* Strain A mCh: a strain labeled with red fluorescent proteins
* Strain A KanMX: unlabeled (not fluorescent) strain only geneticin.

The measurement itself is done in 96 well plates using a plate reader called Tecan Spark 20M Multimode Microplate Reader[1]. One experiment takes around twenty-four hours, taking measurements every 30 or 10 minutes. The current growth of each strain is given by measuring the optical density and the fluorescence of the culture in a well every cycle. After the experiment the results are saved in an excel file that has corresponding tables for the optical density and also for both of the fluorescent measurements. For the individual strains we can rely on the optical density but in the mixed cultures the analysis is based on the fluorescence intensity coming from the different strains[1]. This data makes it possible to characterize each strain according to their level of growth throughout the experiment, and in a later phase observing their behavior and the interactions between them when they are put into co-colonies.

## About the *Saccharomyces cerevisiae*

The *Saccharomyces cerevisiae* commonly known as baker’s yeast is a single-celled fungus microorganism[2]. It has been used for winemaking, brewing and baking for a very long time. It is one of the mostly studied eukaryotic organism and is also recognized as a model system[2]. Its genetic complexity is almost identical to a bacterium, and its genome can also be easily manipulated[3]. It has seventeen chromosomes, and in 1996 its whole genome sequence has been discovered which consists of 12 052 kilobases, and as of now there are around 1500 strains that can be differentiated[3].

One of the reasons that it is an important model organism and often used by researchers, is that it is possible to introduce different mutations into the yeast genome, which makes it easy to study what happens when the DNA sequence is modified. With this it can be observed how will the functions and properties of the cells change when mutation happens[3]. The other reasons it is suitable for biological research are for example its rapid growth, well-defined genetic system, and easy access.

There are also several other interesting research where Saccharomyces Cerevisiae is being used, such as understanding mammalian genes that can affect aging by observing the number of times a cell can divide, and their lifespan in non-dividing stasis state[4]. It is also used for medical purposes like probiotics, and recent studies also show that it can be an important part of cancer research more precisely for testing different drugs which can lead to new discoveries about anti-cancer drugs[5].

Overall, these are some of the interesting but also important facts about *Saccharomyces cerevisiae*, that describes why is it used as a model organism in molecular and cell biology, commercial applications and also plays a part in medical research.

Figure 2.1.1 Importance of Saccharomyces Cerevisiae in the industry

Figure 1.1.2   
Saccharomyces Cerevisiae in research

## Growth of Multicomponent yeast populations

Studying the growth of yeast cells is an important part of microbiology which focuses on observing how the number of cells in a culture change over time. Fortunately, with the currently available modern plate readers it is not difficult to track the growth of colonies in liquid medium[6]. The software that I am working on will also be specialized for analyzing these data, and to evaluate the behavior of the cells during the time they grow.

There are several possibilities to measure the growth, but in this case, it is done by measuring optical density with the help of Spectrophotometers. The basic concept behind it is to calculate the cell density at given time points from the opacity of the culture. The Spectrophotometer itself is responsible for measuring the amount of light that travels through the cells[6]. This will depend on how much of the light scatters or get absorbed.

The ideal growth of bacterial cultures is at the beginning exponential and can be determined by the following differential equation[6] (1):

(1)

N equals the number of the cells, t marks the time and α is growth rate. When this equation is integrated from t = 0 to t = t, the below equation will be the result[6]:

(2)

There are four phases of the growth of yeast colonies, which can be seen on Figure 2.2.1. First there will be a lag time where the samples do not grow, then it is followed by an acceleration phase where the growth rate increases. After a given time this growth rate will become constant, which will still happen inside the exponential phase[6]. Then the deceleration phase starts where growth rate show decline and at last saturation phase begins when available resources for the cells has been exhausted.

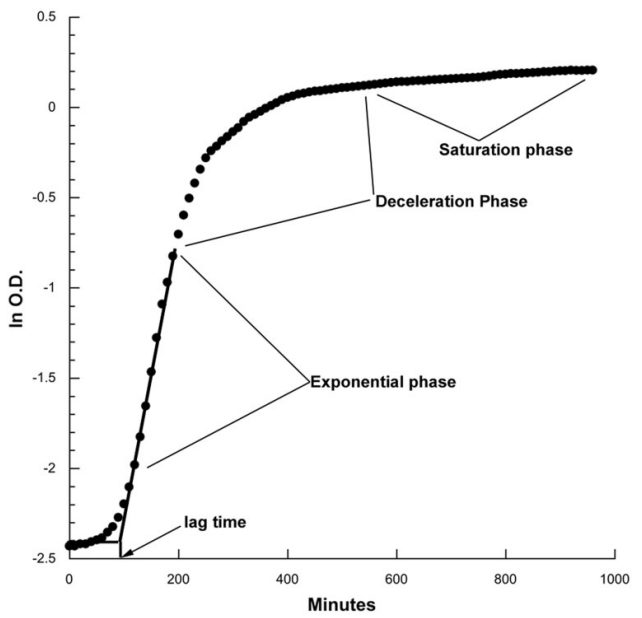


Figure 2.2.1 Tipical yeast cell growth curve and its four phases[6]

## Interactions between yeast cells

There are two forms of the interactions between strains. One is cooperation and the other is competition. It is often assumed that cooperation is the favored since it provides more benefits. However, this cannot work all the time since there will always be a chance for selfish individuals that want to invade a population, who will not cooperate but still obtain all the benefits from the work of the others[7].

Cooperation can occur for several reasons such as increasing the survival rate of the colony. There are also two types of cooperation. One is altruistic and the other is mutually beneficial. Altruistic cooperation happens when effect on the actor is mostly negative, while the effects on the recipient will be positive, meaning that while the actions taken is beneficial for the whole colony, it will not be for the strain itself[7]. When there is mutually beneficial cooperation both the strain and the colony will receive the benefits from the behavior of one strain.

Competition on the other hand is the opposite behavior. This usually occurs over the resources like nutrients. It can show up in several forms, for example toxin production. In this case a strain in the colony is capable of producing toxins which can hinder the other strains[8].

# Objective of the Software

All the different experiments done in order to analyze and evaluate the behavior of several yeast strains generates a big amount of data. This would normally have to be processed manually which is really difficult to do on this scale, because it takes a lot of time, and there is also the risk of making a mistake. There are some programs on the market that deal with some parts of processing of these kinds of microbial data but neither of them is able to do a full evaluation of the results. Also, many of these are not compatible with our experimental data and most of them require programming skills from the user.

My goal was to create a software that is specifically designed to give a full evaluation of our data and also incorporates mathematical calculations connected to microbial growth. Furthermore, the user-friendly environment was also of importance, this way the results can be evaluated with the help of a graphical user interface, no need for any programming skills or making any changes in the code which could lead to several mistakes.

The first problem that had to be solved was the visualization of the data on diagrams, since there are about three-hundred columns in one Excel containing the data that must be shown. But after further discussions I have got to know that besides making the diagrams there are a lot of other things that is done with the generated data like statistical calculations. That is why it was necessary to make a software not just for visualization but also to create a framework that is dedicated to creating a complete report of the strains behavior used in the experiments where the user only has to give the Excel file to the program they want to evaluate. This also makes it possible to compare strains from a different batch of experiments, and it also makes it easier to store the received results in a way that it can be reused any time by anyone.

However, in order to create such a software, I had to make sure that even with its several functions it stays user friendly with a clean UI, generates reliable data but at the same time the processing and analyzing of the data remains quick and efficient. With these objectives in mind, I have started working on the desired software.

# Used technologies

In the next chapter I have listed some of the technologies that I have used for the implementation of the software. Furthermore, I have written about their essential features and why I have decided to use these.

## PyCharm, Python

Python is a high-level, general-purpose programming language. One of the most important reasons that I have chosen Python is because I have experience working in it. However, the other reason is its versatility when it comes to data processing and string operations.

There are also a lot of additional Libraries available that can be integrated during the implementation such as NumPy, LMfit, Matplotlib and so on. This makes Python a really effective programming language for me, since there are several biology related statistical calculations that I have to perform in my software. There is also the fact that I want to use Artificial Intelligence in some parts of my thesis, and for that purpose I will utilize functions taken from Keras which I will write about a bit more in chapter 4.1.1.

As for runtime environment during the implementation and testing phase I have used PyCharm from JetBrains, since it offers great debug capabilities, error highlighting, code completion and makes it easier to maintain a clean and readable code.

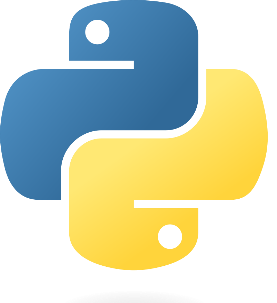
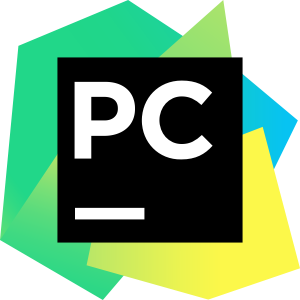
Its nature that allows me to use the written codes as a script that I can call from outside sources also makes it a great choice for me, since I wanted to create the UI itself in a C# WPF application.

Figure 4.1.1. Python icon

Figure 4.1.2. PyCharm icon

### Keras

Keras is a deep learning API written in Python. It runs using TensorFlow 2 which is an open-source machine learning platform. It was created in order to make experimentations faster and easier for developers. It is a high-level API and its speed is slower compared to other similar Libraries like PyTorch or directly using TensorFlow, because of its user-friendly environment, simple and flexible structure its still one of the best choice especially for people like me who is still inexperienced in developing software using machine learning. Its building blocks are layers and models. There are some built in models, like the Sequential model, but there is also the possibility to write the models entirely from scratch if a more complex architecture is needed.

In my software I will be using Keras to build LSTM networks in order to predict different values from the growth data of yeast cells in past experiments. Then I can use these values for example filtering the data that is about to be processed by checking if there are bigger then allowed differences compared to the values that were predicted.

Figure 4.1.1.1. Keras icon

## Visual Studio, C#

Visual Studio is an integrated development environment made by Microsoft. It can be used to develop various programs, like web or desktop applications, different web services, or even mobile programs. It supports a wide range of programming languages. There are a few amongst them then is already built-in but the other can be used through plug-ins. For my program I will be using C# as the programming language and Windows Presentation Foundation as the platform to create the user interface. I have used WPF as the foundation of my applications several times that is why I have decided to develop the UI part of my program using it this time too. It offers a lot of useful tools for creating a clean and user-friendly interface. The other reason is that LiveCharts2 what I am going to use for the data visualization on diagrams supports WPF application.

C# as a programming language is similar to Python in the area that it is also a general-purpose, object-oriented language. The biggest difference might be that C# is statistically typed and offers a smaller range of support for external Libraries.

Figure 4.2.1 Visual Studio icon

### LiveCharts2

LiveCharts is an open-source cross platform data visualization library for .Net. It can be used to create a variety of diagrams such as basic line or bar charts, scatter plots, pie charts and several more. It allows to create flexible and well customizable diagrams while still being simple to use and interactive. In my project I will be using this Library to visualize the growth data of the yeast cells.

Figure 4.2.1.1 LiveCharts icon

# Current progress

In the first half of the year, I have been mostly working on the UI, the processing of the data and the visualization using diagrams.

On the data processing part, I have finished the transformation of the input Excel file, so I can separate the header part of the file from the actual data. I have managed to create the necessary classes and data structure for the JSON files where I want to store the properties of the experiment extracted from the header and the processed data. I have also implemented some parts of the data analyzation, like calculating the average and standard deviation.

For the user interface part, I was able to create a fully dynamic workflow so that the diagrams can update on runtime without any unnecessary reloads or button presses when a new strain is selected, or additional data is loaded from other experiments. I have added the possibility for the user to configure some personal settings like assigning custom colors to selected strains and configuring default directories where the program should look for input files. There are also settings affecting the diagram, for example only showing average values, displaying two strains in two separate Y axes, or using the previously set custom colors.

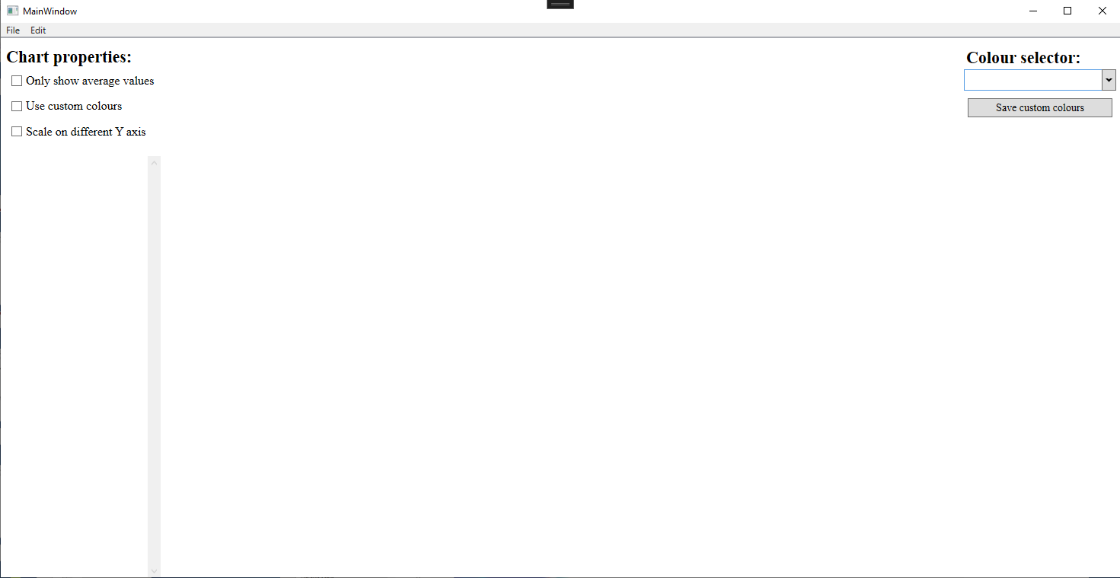
All in all, I feel like I was able to make great progress in this semester especially on the UI part because it feels straightforward and clean, but still offering the necessary configurations. By implementing a part of the data processing and analyzing I have developed about half of the application, making sure that I can finish everything in time.

Figure 5.1 User interface at start

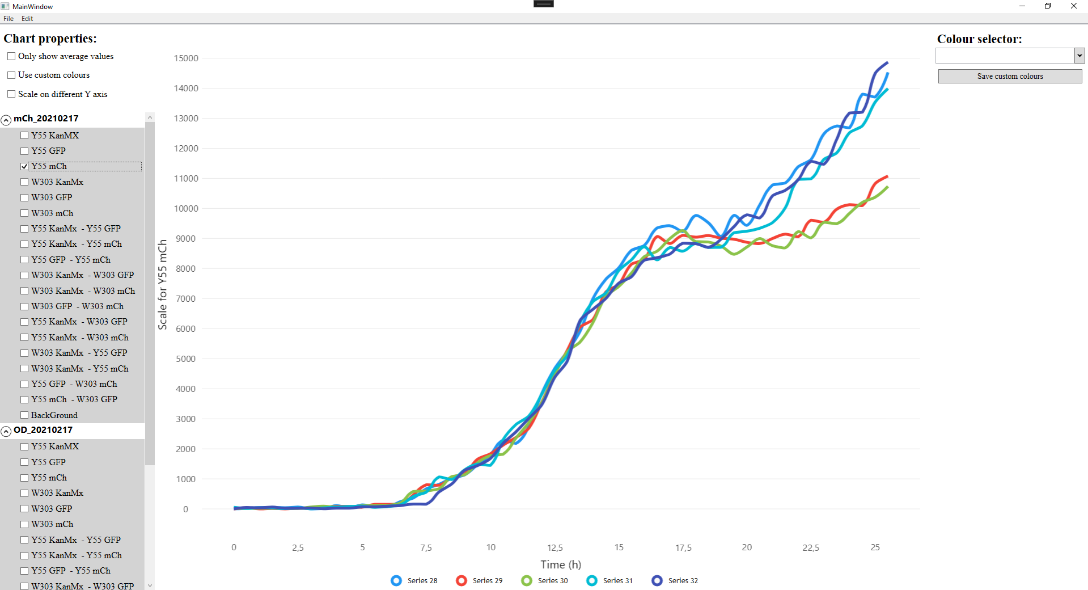
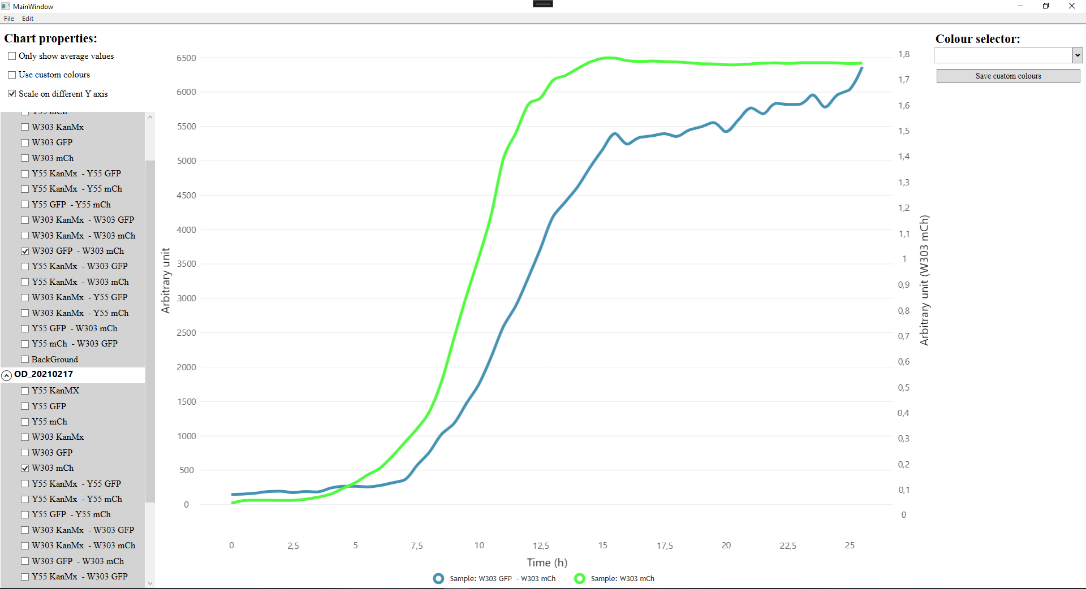
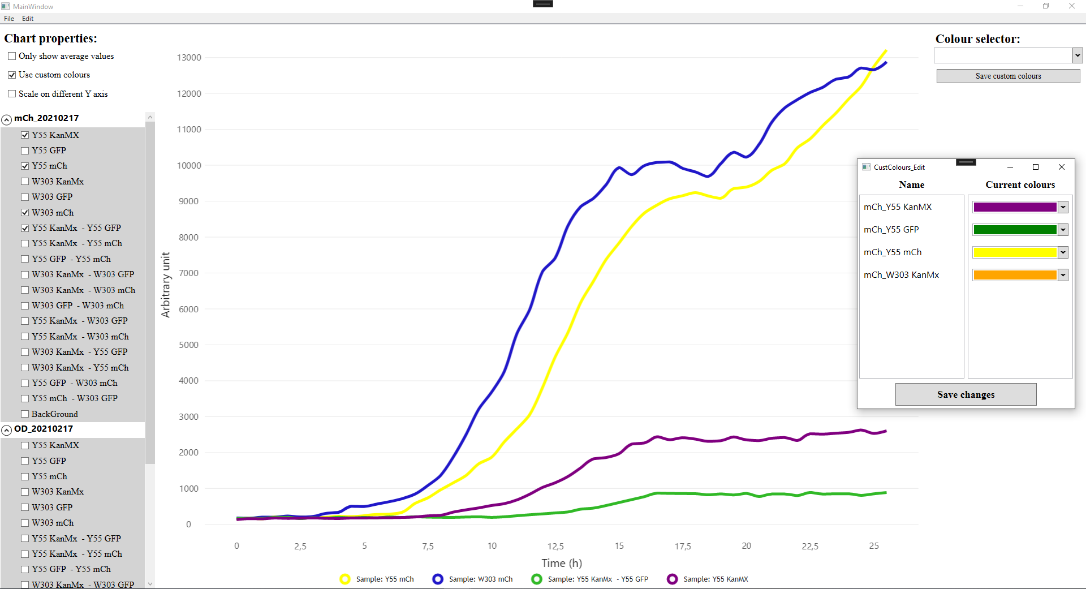


Figure 5.3 Multiple strains with custom colors

Figure 5.2 Only 1 selected strain

Figure 5.4 Double Y axes

# Plans for next semester

During the next semester I am going to work on the data filtering, the remaining data analysis, and the checking of possible interactions between strains.

For the filtering I will be using Artificial Intelligence, LSTM networks by predicting a range for every measurement taken at every given point of time from previous experiments and check if the data in the current experiment falls into this range. If it does not, then I will warn the user about the possible data corruption and offer the possibility, to opt out that specific strain. This way that data from that one strain will not influence the average and growth rate calculations during the analysis.

One of the bigger topics in the analysis step is calculating the microbial growth rate. For this I will be using some model fitting algorithms like Gompertz Model and Rolling Regression.

The last part is the checking for interactions between strains, but we are still discussing this part in the labor to find the best solution and create the appropriate equation and model for it. During these discussions there is also the possibility that other ideas will come up, that should also be implemented in the software.

# References

[1] B. Pillér „Interactions of cells in multicomponent yeast populations”.

[2] F. Sherman, “[1] Getting started with yeast,” *Methods Enzymol*, vol. 194, no. C, pp. 3–21, Jan. 1991, doi: 10.1016/0076-6879(91)94004-V.

[3] D. Botstein and G. R. Fink, “Yeast: An Experimental Organism for 21st Century Biology,” 2011, doi: 10.1534/genetics.111.130765.

[4] V. D. Longo, G. S. Shadel, M. Kaeberlein, and B. Kennedy, “Replicative and Chronological Aging in Saccharomyces cerevisiae,” *Cell Metab*, vol. 16, no. 1, p. 18, Jul. 2012, doi: 10.1016/J.CMET.2012.06.002.

[5] J. A. Simon and A. Bedalov, “Yeast as a model system for anticancer drug discovery,” *Nat Rev Cancer*, vol. 4, no. 6, pp. 481–488, 2004, doi: 10.1038/NRC1372.

[6] B. G. Hall, H. Acar, A. Nandipati, and M. Barlow, “Growth rates made easy,” *Mol Biol Evol*, vol. 31, no. 1, pp. 232–238, Jan. 2014, doi: 10.1093/MOLBEV/MST187.

[7] S. A. West, A. S. Griffin, A. Gardner, and S. P. Diggle, “Social evolution theory for microorganisms,” *Nature Reviews Microbiology 2006 4:8*, vol. 4, no. 8, pp. 597–607, Aug. 2006, doi: 10.1038/nrmicro1461.

[8] J. Friedman and J. Gore, “Ecological systems biology: The dynamics of interacting populations,” *Curr Opin Syst Biol*, vol. 1, pp. 114–121, Feb. 2017, doi: 10.1016/J.COISB.2016.12.001.