5-2018

PID Algorithm Design for Automation of Eukaryote Fermentation

Madison Mann
University of Arkansas, Fayetteville

Abstract

Single celled eukaryotic organisms are advantageous in the production of proteins for use in biopharmaceuticals due to their ease of cultivation and manipulation. Because of high cell density during fermentation and an ability to allow post-translational protein modifications, yeast is particularly well-suited. To maximize the specific formation rate of the desired proteins, an optimum specific growth rate of the yeast should be found and maintained. This rate is dependent upon pH, temperature, dissolved oxygen content, and carbon feed rate. Fermentation can be done in a fed batch reactor system, and it is possible to use a Buglab (online biomss) sensor to monitor the conditions in the reactor. The focus of this project was to create a Proportional Integral Derivative Controller algorithm to optimize the specific growth rate of a yeast automatically. Using real time data provided by the online biomass sensor, the algorithm will be able to calculate an output to correct any deviations in the optimum growth rate by controlling the feed rate. Maintaining this specific growth rate will in turn optimize the production of heterologous proteins.

Background

Eukaryotes are known to be a standard recombinant protein expression host, allowing for a relatively inexpensive and highly successful system for production of many different heterologous proteins. Despite being single celled, yeast is capable of post-translational modifications including: protein folding, formation of disulfide bridges, proteolytic processing, and glycosylation. Thus, yeast can express proteins containing complicated structures and functions. These modifications also contribute to the production of biologically active molecules that are often found as inactive in bacteria protein expression.[3] *Kluyveromyces lactis* was chosen

as the main organism for this project, and its preference for respiratory growth creates the ability to culture at high cell densities. This fermentation quality allows for high cell density without the production of large amounts of ethanol that would limit growth.[1]

K. lactis can be cultured in a fed batch system with relatively inexpensive medium components: carbon source (glucose), biotin, and various salts to achieve high cell density. To maximize the growth rate an exponential feeding rate would have to be performed.[7] A balance must be struck between providing glucose at rates fast enough to continue the fermentation, and slow enough to not poison the culture with too much carbon source. This research will investigate the following hypothesis:

Online biomass monitoring can form the basis for a control strategy that provides glucose at a rate necessary for high growth and protein expression without overfeeding the fermentation.

The fed batch operational strategy is designed to optimize conditions for growth and maximum product formation.[5] Optimum productivity for the system is dependent on the pH, temperature, and nutrient supply. Productivity, or the specific formation rate of the target protein, shares a distinct relationship with the specific growth rate of biomass in the reactor. An optimum specific growth rate can be determined based on the target protein, as well as the operating conditions and nutrient Therefore, a robust system capable of controlling the organisms' specific growth rate is desired.

A novel strategy using sensor measurements and a PID control algorithm is proposed to provide continuous control to maintain a desired specific growth rate in fed batch systems. Very few strategies that employ fed batch (if any) permit changes in specific growth rate. Indeed, the

growth rate (μ). The noninvasive BugEye sensor can provide real time measurements of biomass that can then be used in by a PID algorithm to maintain feed at an optimal level. The optical density readings given in arbitrary Bug Units (BU) from the BugEye sensor along with the rate of change are used to calculated the specific growth rate at a specific time (t) in the fermentation using the equation below:

$$\mu = \frac{1}{BU} * \frac{dBU}{dt}$$

This project had three objectives: to determine the hardware requirements that would pass information from the BugEye to the fermentation control software, to provide written code in BioExpert (control software) that would calculate MU and the feed rate, and finally to test the control structure. Ultimately, the project centers around rapid and consistent response to changes in cell mass during fermentation. After the hardware elements were determined, the first fermentation experiment was used to confirm the system and optical density probe setup was appropriate. The next two experiments explored the possibility of growth rate calculations in the software, and the ability for exponential feeding. Next, an exponential feeding profile was written that could control on-line variables of a standard eukaryotic fermentation. This feed profile is a modified version of a code used in the Ph.D. dissertation of Dr. McKinzie Fruchtl.[8] In this program, the feed rate is determined by the following equation:

$$F(t) = P * \frac{\mu}{Yxs * Sin} * X_0 * V_0 * e^{\mu t}$$

In this equation, the feed rate at any given time is a function of the specific growth rate, μ, substrate yield coefficient, $Y_{xs,}$ substrate concentration in the feed, S_{in}, initial cell mass concentration, X_0, and initial volume, V_0. The general yield coefficient used in this experiment is 0.5 (g/g) and was determined based on E. coli, a common eukaryote used in protein expression.[8]

Finally, the proportional steady state gain constant, P, was to be found using a guess and check method in the final experiment of the project.

Materials and Methods

Materials:

Before the actual fermentation, the hardware was set-up and connected appropriately. It was necessary to convert the BugEye output of 4-20 milliamps to 0-2 volts. This was done by adding a resistor across the output and ground. This voltage was then read by the analog to digital controller by the control system.

All cell cultures were purchased from New England Biolabs and initially grown in shake flasks in small volumes of either Lysogeny broth or YPGlu media prior to inoculation of the fermentation vessel. The dehydrated LB concentrate used was purchased from Sigma Aldrich as well as the Bacto Peptone and Yeast Extract used for the YPGlu media. All media was mixed the day before the start of fermentation and sterilized at 121℃. Table 1 shows a summary of information for each fermentation.

Experimental Data

Run	*Experiment*	*Organism*	*Media*	*Temp (℃)*	*Agitation (rpm)*
1	Set Up Confirmation	*Saccharomyces cerevisiae*	LB	30	250
2	Feed Pump Set Up	*Kluyveromyces lactis*	LB	30	350
3	Rate Functions	*Escherichia coli*	LB	37	350
4	Code Run	*Kluyveromyces lactis*	YPGlu	30	450

All optical density data was taken using a DU 800 spectrophotometer (Beckman) 600 nanometers. When E. coli was cultured, ampicillin purchased from VWR was added at a concentration of 50 ug/ml.

Initial Set-Up confirmation:

A 500 ml LB media solution was inoculated with 10 ml of *Saccharomyces cerevisiae* (baker's yeast) in a 1L fermenter. Inoculation proceeded after the fermenter contents were heated to 30ºC at time equal to 0.5 hours. Other growth parameters included: agitation rate of 250 rpm, air pressure at 5 psi, and an initial pH of 6.5 The grow was just shy of 60 hours, and data was taken intermittently. The absorbance of 0.5 ml samples was measured at 600nm, and corresponding optical density readings from the BugEye sensor and the software were recorded. Also, the voltage signal from the BugEye to the analog data recorder was measured for comparison using a voltmeter.

Feed Pump Set-Up:

The next growth was performed with a similar set up as previously mentioned with 500ml of LB media in a 1L fermenter. However, the yeast strain used was *K. lactis*, and the agitation rate was increased to 350 rpm. To contest the low dissolved oxygen levels in the first run, an oxygen tank was set up. This allowed bursts of pure oxygen to enter the fermenter when levels were detected to be too low (less than 35%). Inoculation occurred at time equal to 6 hours. Once steady growth was established and optical density began to decline, a rectangular feed pulse was introduced to stimulate growth. At time equal to 21 hours, an arbitrary feed profile was started that added approximately 100 ml feed of a 40 % glucose solution over a period of 6 hours.

Determination of Rate Functions:

To explore the BioXpert software, a fermentation was done using 2 ml of *E. coli* and 500 ml of LB media in a 1L vessel. The temperature and agitation were set to a constant 37ºC and 350 rpm respectively. Ampicillin was also included at the time of inoculation (time equal to 4.9 hours). The rate of the BugEye's measure OD changes was calculated in the BioXpert software and plotted as well as the specific growth rate. To cut down on the noise and create a graph that is easier to interpret, the average of the change in optical density as well as the specific growth rate taken over 120 minutes was plotted.

Code Check:

After the control code was finalized, another trial was performed to tune the proportional control via a guess and check method on the steady state gain constant. To see the code used, refer to the appendix. This run was done completely sterile in a 3L vessel using the *K. lactis* yeast. The larger vessel provided a longer cultivation time to optimize the control settings. Similarly, to the previous runs, the yeast was grown previously in a shake flask and transferred to 200 ml of the YPGLu media. 1.5 liters of the same media was used and heated to a temperature of 30°C while a 40% glucose feed as well as a 45% ammonium hydroxide solution were hooked up to the fermenter. The ammonium hydroxide base was used to regulate pH during the fermentation. The constants, V_0=1.5L, and X_0=0.15 gcells/L were send at time equal to 1 hour, the vessel was inoculated with 60ml of the *K. lactis* solution. This gave an initial optical density of 0.5 as determined by the spectrophotometer. The yeast grew for 96 hours and a final optical density reading of 22.1 was obtained.

Results and Discussion

The initial set up confirmation fermentation was done to ensure the system was running correctly and the software was reading the online variables. This included the correct interpretation of the optical density probe into units for the software to use. To test this, the data for optical density from the BugEye probe, and the measured absorbance were plotted as well as the optical density data over time for both the software, and hand taken data points. Figure 1 below shows straight line correlations between the Bug Units and the absorbance measured at 600nm as well as the output voltage from the probe that would be read by the control system.

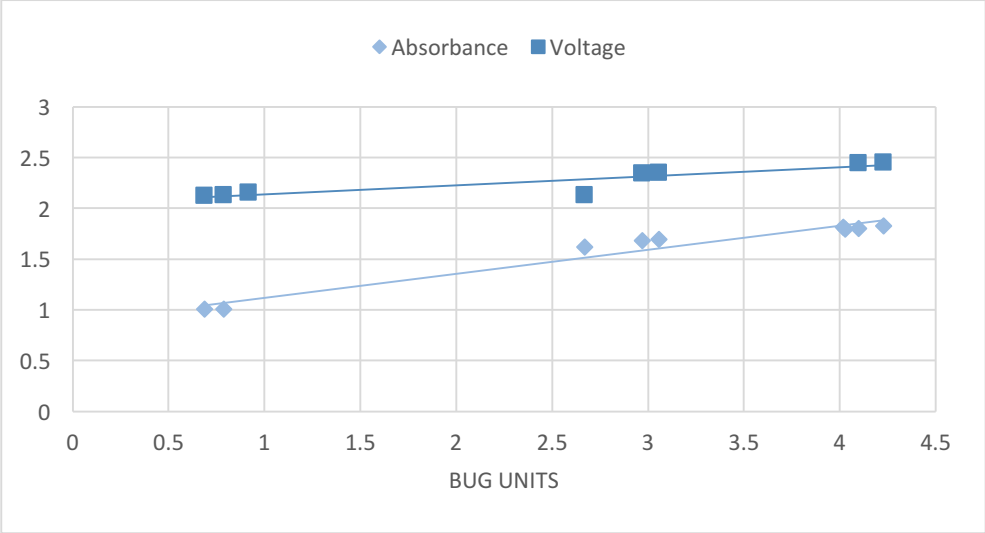


Figure 1: Absorbance and Voltage dependence on Bug Units

Figures 2a and 2b plot the correlation between the optical density data over time between the system software, and measured data points. These graphs show a distinct and similar pattern meaning the software could correctly take the signal from the BugEye sensor and use it to plot growth over time.

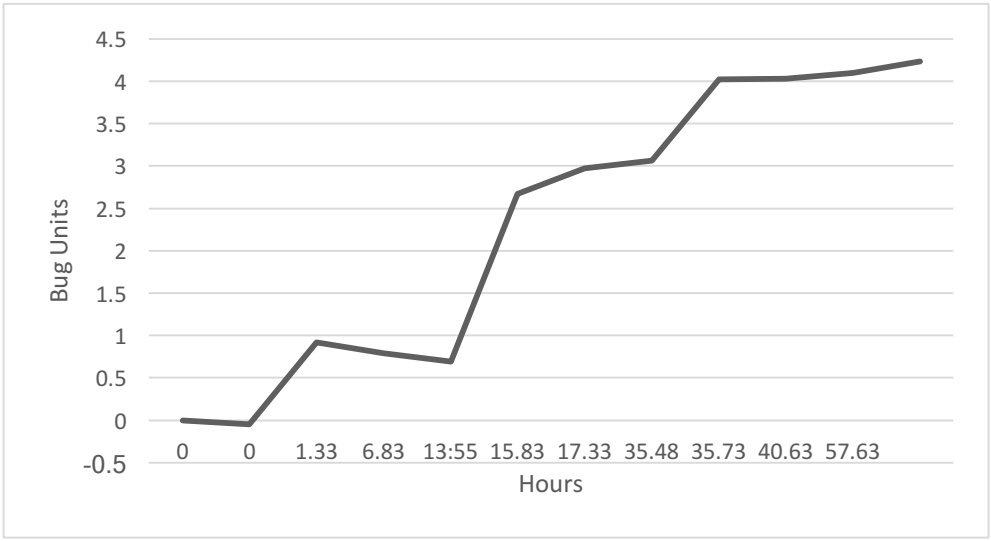


Figure 2a: Hand Taken Bug Unit Data Over Time

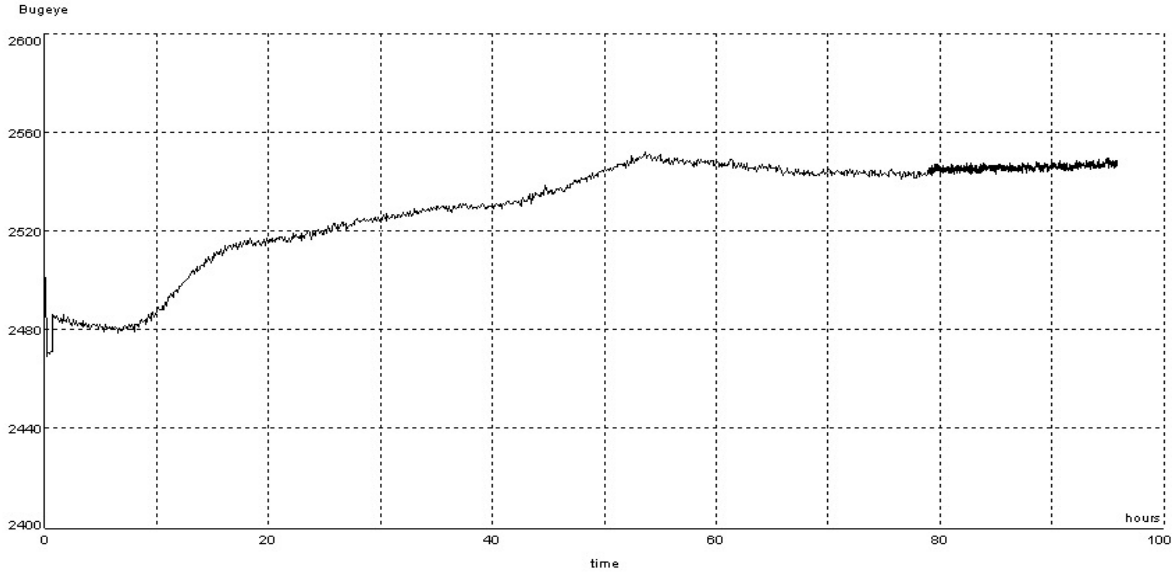


Figure 2b: Software Developed Bug Unit Data Over Time

The second fermentation was performed after the set-up of the feed pump to ensure it was in working order and to delve into the software and explore the profile options for feed. Figure 3 below shows the growth profile measured by the BugEye over the 27-hour fermentation plotted along with the dissolved oxygen in the fermenter and the feed pump profile.

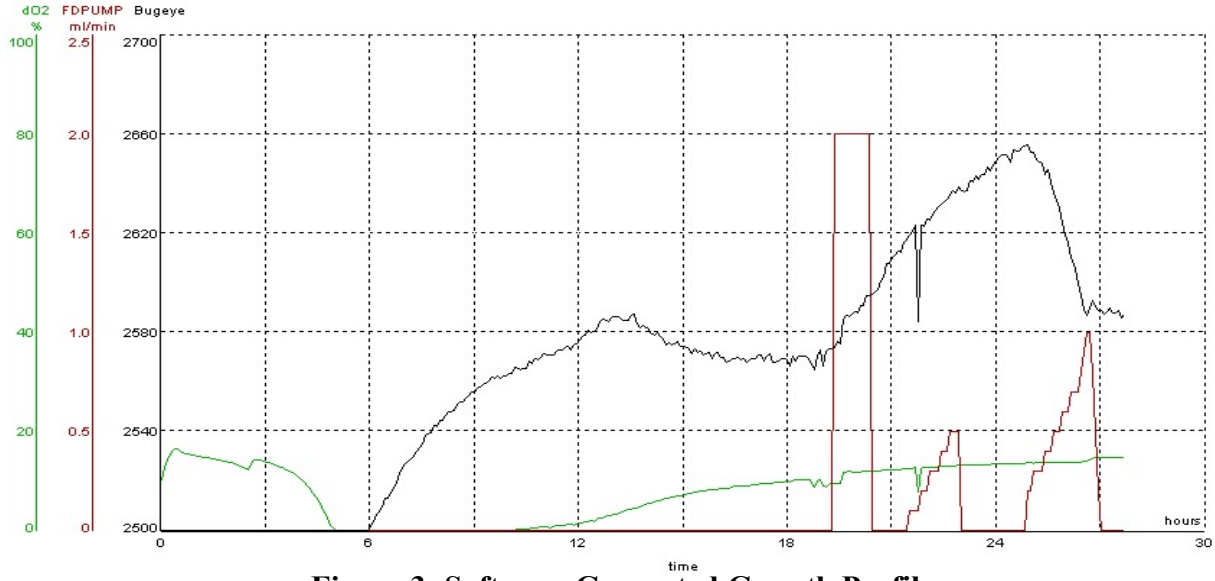


Figure 3: Software Generated Growth Profile

This graph, generated by the BioXpert software, shows the growth of the *K. lactis* yeast as well as the decline in dissolved oxygen around six hours after inoculation. This six hour period represents the characteristic lag phase of exponential growth. As expected, a decline in dissolved oxygen is observed when growth begins. After this, the software corrects the lack of dissolved oxygen using the pure oxygen provided. The feed pump was started after a declining growth trend was observed. A rectangular pulse of 2 ml per minute was administered to reestablish growth in the yeast. Then an arbitrary feed profile was programed that would feed 94 milliliters of the 40% glucose solution over the next 6 hours. This profile was executed perfectly by the software as shown in the triangular patterns in feed rate towards the end of the fermentation. Eventually the cells became exhausted, evident by a sharp decline in the optical

density even with the feed still being introduced to the reactor. Other small abnormalities in the growth and dissolved oxygen levels at time equal to 21.5 and 26.5 hours occur due to a pause in agitation for samples to be taken.

The purpose of the fermentation using the E. coli was to test if the software would measure not just the optical density and overall growth, but the rate at which the growth occurs. The following formulas were added to the list of on-line variables before inoculation:

$$Rbug = rate(Bugeye) \qquad Mu = \left(\frac{1}{Bugeye}\right) * (rate(Bugeye))$$

To reduce the noise in the graph during the fermentation, the average of the rate of change as well as the average of the specific growth rate were plotted in Figure 4.

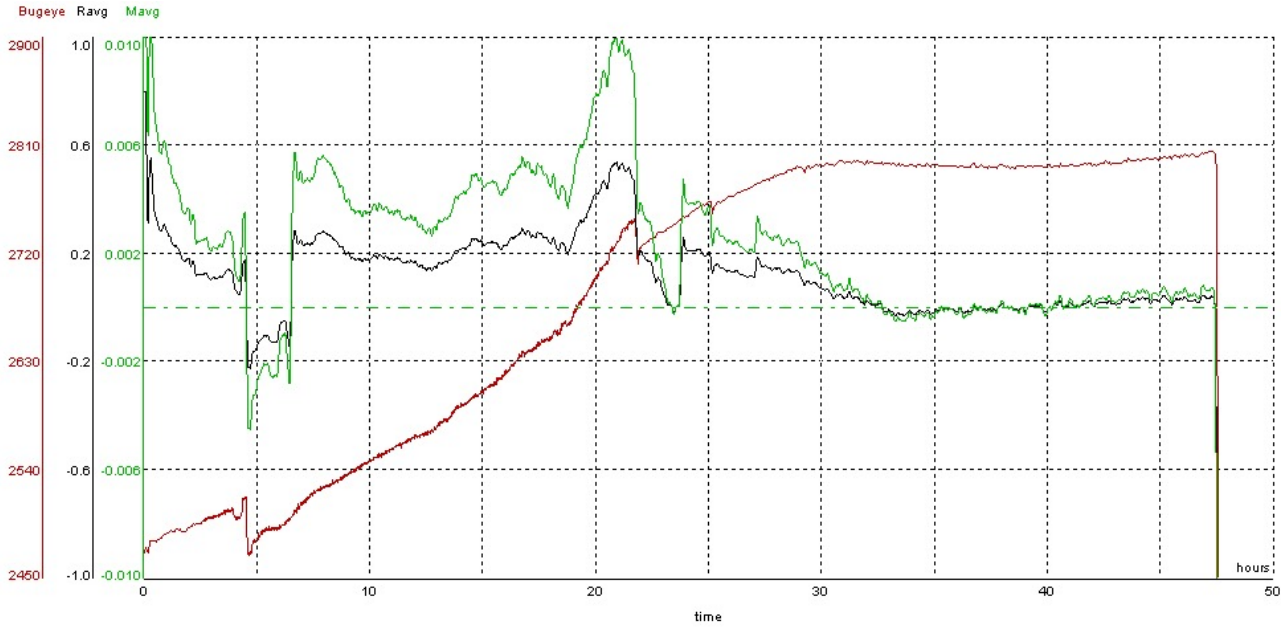


Figure 4: Rate and Growth Averages

Data averages were taken over two-hour time intervals to produce a graph that is easily read and interpreted. The formulas added were successful in calculating the specific growth rate over the 47-hour fermentation. Not only was the deviation kept within a range of 0.01, but the

trend shows the expected decrease to zero when the cells stop multiplying (around time equal to 30 hours), and a plateau is reached in the BugEye reading.

The final growth done with the full code showed some room for improvement. Figure 5 below shows the growth profile highlighting the specific growth rate, BugEye reading, feed pump, and pH reading.

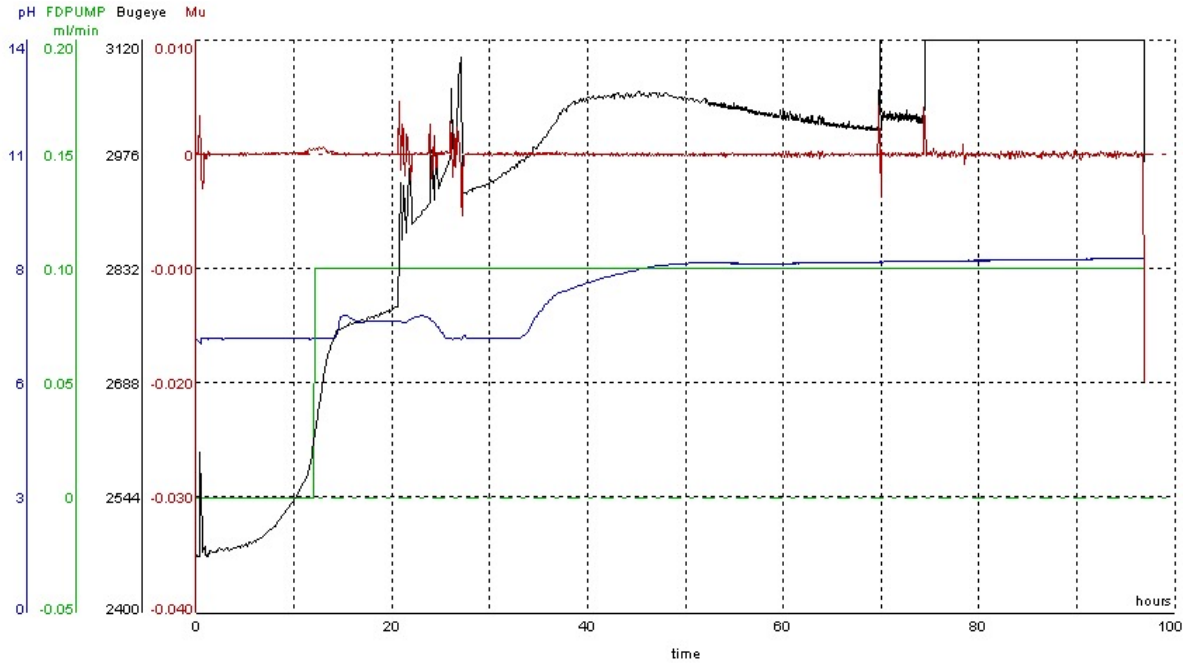


Figure 5: Code Check Growth Profile

The optical density curve itself (black) shows sigmoidal curves signature to exponential growth as expected. The optical density sample taken at the beginning of growth was 0.5 at 600nm while the final sample taken measures 22.1 at 600nm. This level of growth was satisfactory. The feed pump (in green) triggers on when there is a small spike in the specific growth (red) and a rise in optical density. However, the feed remained consistent at 0.1 milliliter per minute. The radical changes in BugEye reading and specific growth rate at around 20 to 25 hours are from changes

in the speed of agitation. The changes occur because of the dependence of optical density to agitation rate. The speed was adjusted, however, to try to account for loss in dissolved oxygen.

Even when the gain constant, P, from the feed equation was changed to 10, the feed pump remained at 0.1 milliliter per minute. This means after an order of magnitude change, the feed remained the same. There could be many reasons for this including the tubing size used for the feed, the rise in pH shown in Figure 5, or the coding itself.

Future Work

From this project, it is clear that creating an automated feed system based on non-invasive optical density readings using a BugEye probe is feasible. The next step would be to trouble shoot the written code and software calculations to ensure they are working properly. Some investigation was done at the end of this project on the rate function used to calculate specific growth rate in the software. The change in BugEye reading over time for hours 72 through 82 were plotted in Figure 6. The slope of this line should correspond to the average rate of growth calculation by the software. However, the average rate of growth of -0.24 was not nearly close to the average slope of growth of 32.3. Therefore, the next step of this project would be to find the error in the software's calculations and decide on a way to correct this.

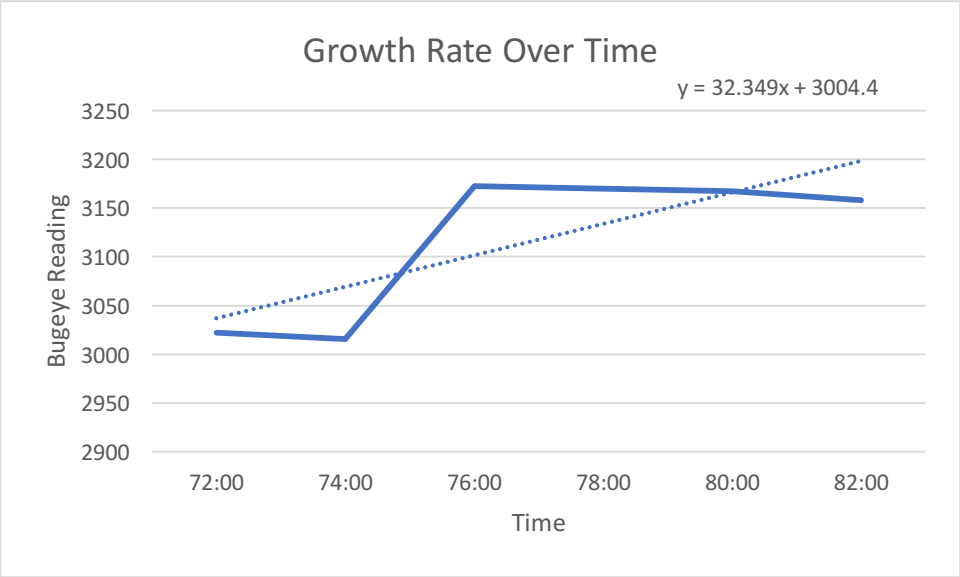


Figure 6: Growth Rate over Time

References

1. Cereghino, J. L., & Cregg, J. M. (2000). Heterologous protein expression in the methylotrophic yeast Pichia pastoris. *FEMS Microbiology Reviews*, *24*(1), 45–66.

2. Cino, J. (1999). High-yield protein production from Pichia pastoris yeast : A protocol for benchtop fermentation. *American Biotechnology Laboratory*, *17*(6), 10–12.

3. Cregg, J. (1993). The Pichia System. *Keck Graduate Institute, Claremont, Calif*, 1–8

4. Invitrogen Corporation. (2002). Pichia Fermentation Process Guidelines Overview Overview , continued. *Progress in Botany*, *67*, 1–11.

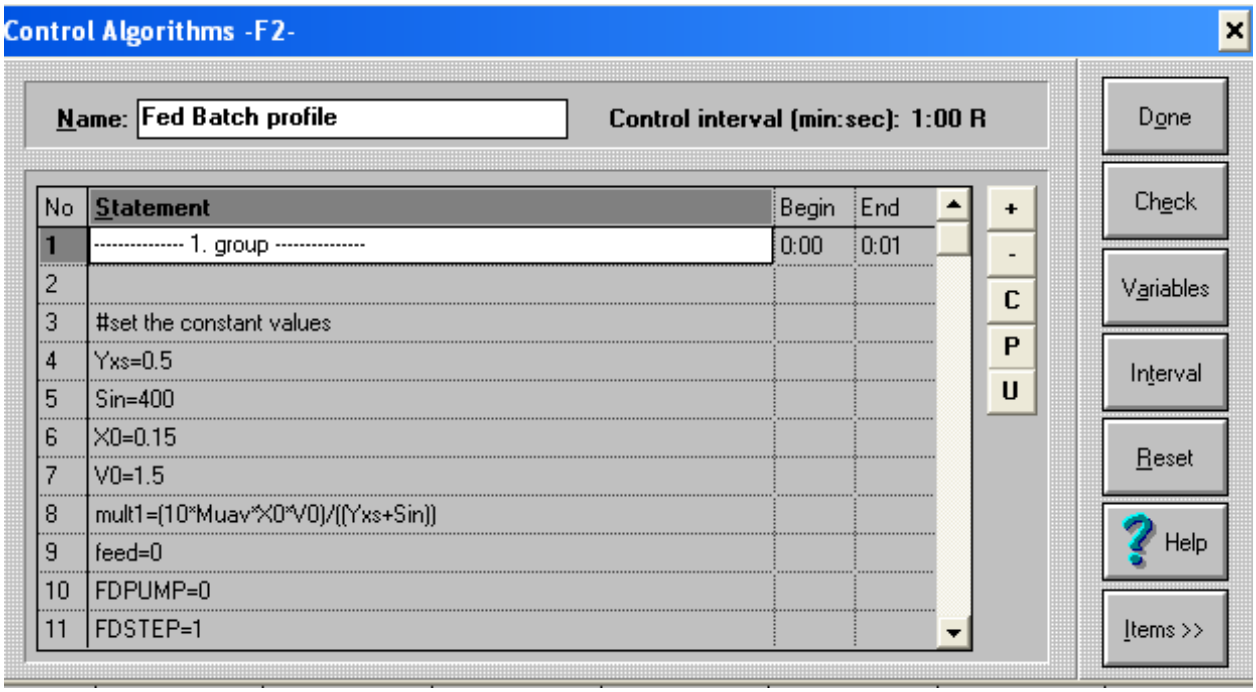
5. Looser, V., Bruhlmann, B., Bumbak, F., Stenger, C., Costa, M., Camattari, A., … Kovar, K. (2014). Cultivation strategies to enhance productivity of Pichia pastoris: A review. *Biotechnology Advances*.

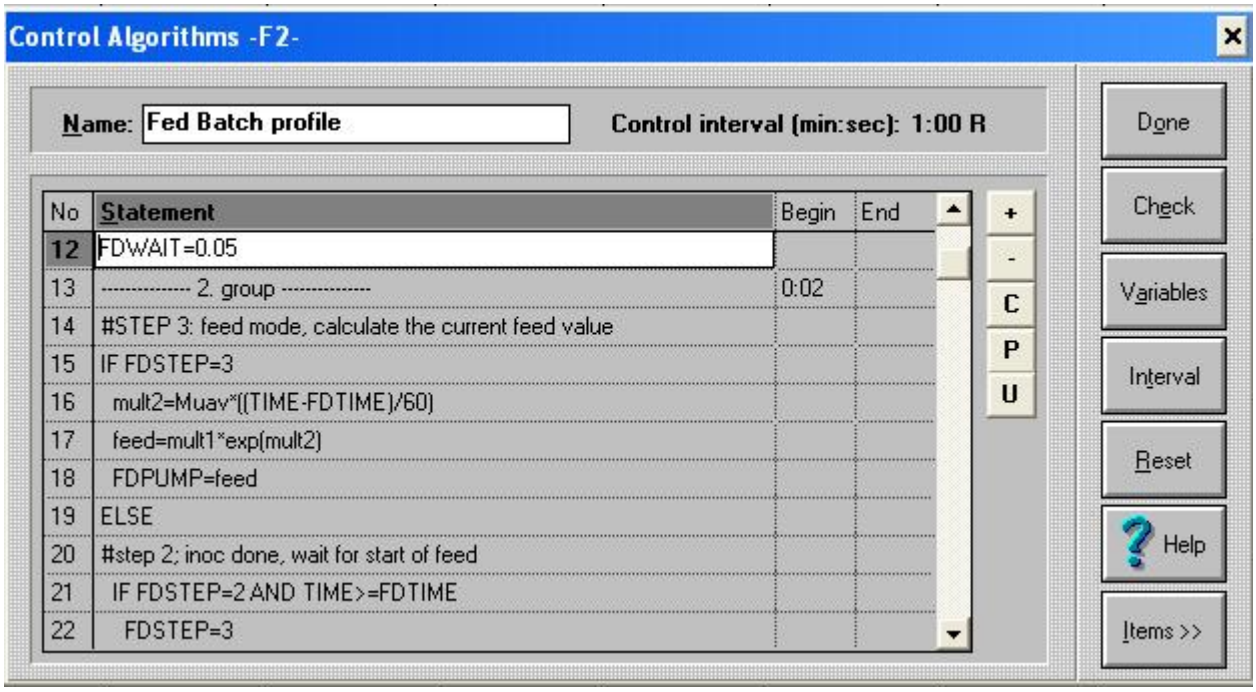
6. Tehrani, K. A., & Mpanda, A. (2012). PID Control Theory, (1).

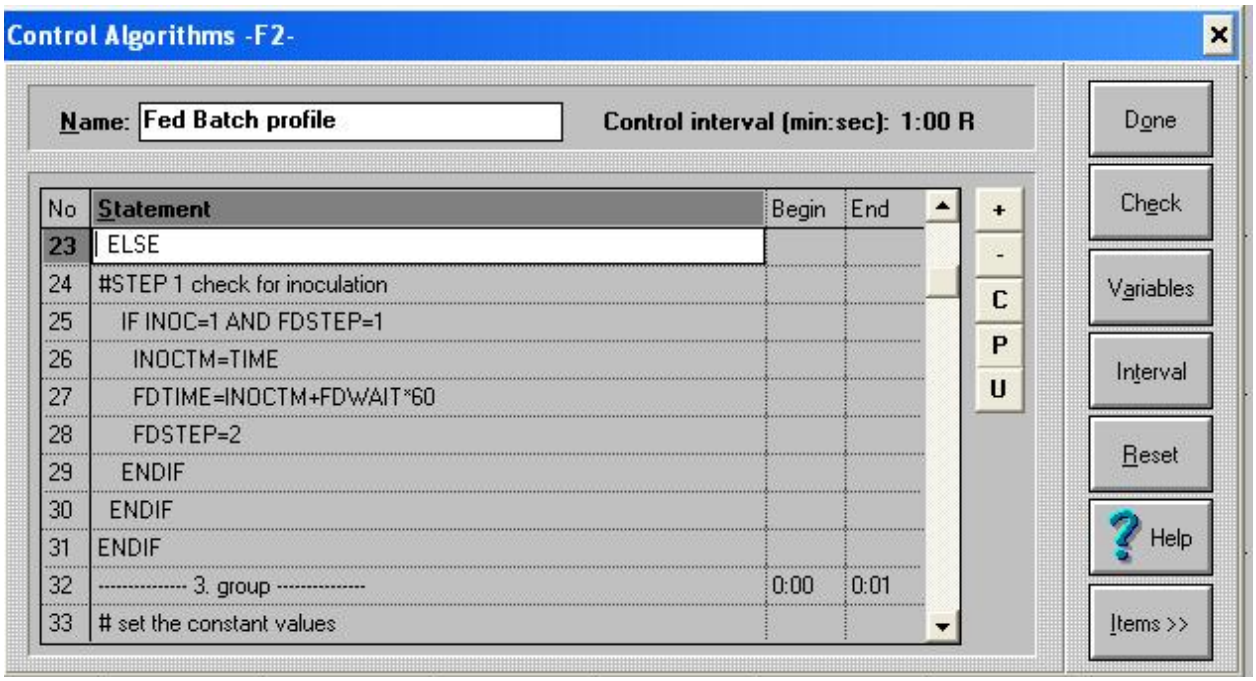
7. Ooyen, Albert J.J. Van, et al. "Heterologous Protein Production in the Yeast Kluyveromyces Lactis." *FEMS Yeast Research*, Wiley/Blackwell (10.1111), 21 Mar. 2006, onlinelibrary.wiley.com/doi/10.1111/j.1567-1364.2006.00049.x/full.

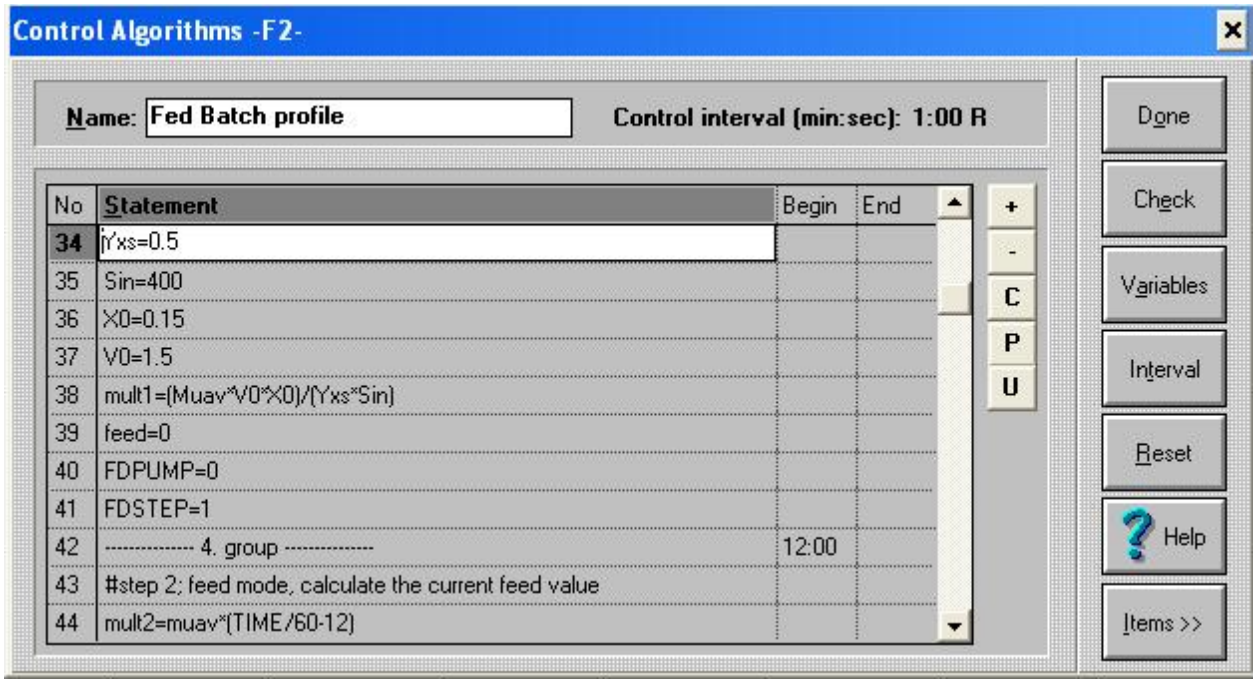
8. Fruchtl, McKinzie S. *Expression, Production, and Purification of Novel Therapeutic Proteins*. 2013.

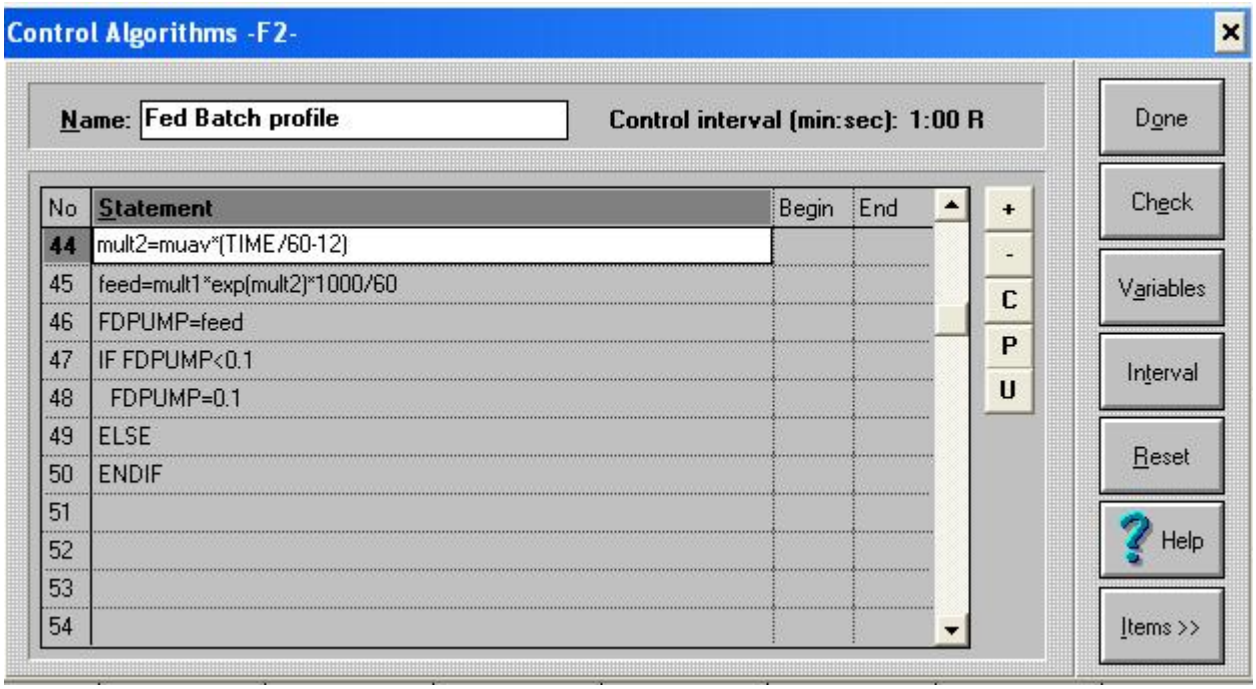
Appendix


Control Algorithms -F2-
Name: Fed Batch profile
Control interval (min:sec): 1:00 R
No Statement Begin End
1 ---------- 1. group ---------- 0:00 0:01
2
3 #set the constant values
4 Yxs=0.5
5 Sin=400
6 X0=0.15
7 V0=1.5
8 mult1=(10*Muav*X0*V0)/((Yxs+Sin))
9 feed=0
10 FDPUMP=0
11 FDSTEP=1
+
-
C
P
U
Done
Check
Variables
Interval
Reset
Help
Items >>


Control Algorithms -F2-
Name: Fed Batch profile
Control interval (min:sec): 1:00 R
No Statement Begin End
12 FDWAIT=0.05
13 ---------- 2. group ---------- 0:02
14 #STEP 3: feed mode, calculate the current feed value
15 IF FDSTEP=3
16 mult2=Muav*((TIME-FDTIME)/60)
17 feed=mult1*exp(mult2)
18 FDPUMP=feed
19 ELSE
20 #step 2; inoc done, wait for start of feed
21 IF FDSTEP=2 AND TIME>=FDTIME
22 FDSTEP=3
+
-
C
P
U
Done
Check
Variables
Interval
Reset
Help
Items >>


Control Algorithms -F2-
Name: Fed Batch profile
Control interval (min:sec): 1:00 R
No
Statement
Begin
End
23 ELSE
24 #STEP 1 check for inoculation
25 IF INOC=1 AND FDSTEP=1
26 INOCTM=TIME
27 FDTIME=INOCTM+FDWAIT*60
28 FDSTEP=2
29 ENDIF
30 ENDIF
31 ENDIF
32 ---------- 3. group ---------- 0:00 0:01
33 # set the constant values
+
-
C
P
U
Done
Check
Variables
Interval
Reset
Help
Items >>


Control Algorithms -F2-
Name: Fed Batch profile
Control interval (min:sec): 1:00 R
No
Statement
Begin
End
34 Yxs=0.5
35 Sin=400
36 X0=0.15
37 V0=1.5
38 mult1=(Muav*V0*X0)/(Yxs*Sin)
39 feed=0
40 FDPUMP=0
41 FDSTEP=1
42 ---------- 4. group ---------- 12:00
43 #step 2; feed mode, calculate the current feed value
44 mult2=muav*(TIME/60-12)
+
-
C
P
U
Done
Check
Variables
Interval
Reset
Help
Items >>


Control Algorithms -F2-
Name: Fed Batch profile
Control interval (min:sec): 1:00 R
No
Statement
Begin
End
44 mult2=muav*(TIME/60-12)
45 feed=mult1*exp(mult2)*1000/60
46 FDPUMP=feed
47 IF FDPUMP<0.1
48 FDPUMP=0.1
49 ELSE
50 ENDIF
51
52
53
54
+
-
C
P
U
Done
Check
Variables
Interval
Reset
Help
Items >>