Summary of Summer Activities

Christian L. Omidiran, Jason Ou

September 2, 2003

1 Introduction

For this summer's VIGRE work, we were given the assignment of learning as much as possible about DNA microarrays (also known as genechips) for the purpose of gathering data on E. Coli for further analysis. As there was a lot of background material to be learned, our investigation spanned many different areas of mathematics, biology, and chemistry. We worked with bioengineering researchers, learning from them the necessary lab techniques needed for conducting experiments with the microarrays. In the following, we describe some of the different directions our studies extended to, including the basics of DNA microarrays, our experience in the bioengineering laboratory, the technology of scanner hardware, and metabolic engineering. Furthermore, we discuss some of the future directions that our summer experience may lead.

2 DNA Microarrays

DNA microarrays, also called genechips, are a tool for the analysis of genetic samples. Although they have many applications, our application is for the reverse engineering of the genetic networks of *E. Coli*. This is done by providing a snapshot of all the genes that are turned on in a cell at any one time. The seemingly complex process actually uses basic biological principles, and is easy to understand.

The transcription of an enzyme is the result of a gene being turned on, and can serve as an indicator of that gene's activity. If a gene is *on*, then RNA polymerase is actively transcribing the gene into mRNA. If it is *off*, then no corresponding mRNA is produced. A single genechip is designed to have all the genes encoded into an array on the slide. This process is done by artificially synthesizing and attaching nucleotides to the glass (or other material) slide. Different techniques exist for this process, two of which include photo-coupling and printing. After obtaining such a chip for a certain organism, i.e. *E. Coli*, the researcher must then obtain all the relevant mRNA being produced while under the conditions of a particular experiment. The steps involved in this process include lysing the cell, density-gradient centrifugation, and purification. At this point, the researcher has the single-stranded DNA encoding all genes of interest, each attached to a specific spot on the chip array. He also has mRNA from the cell under study. To determine which genes are turned on, the researcher uses RT-PCR (reverse-transcriptase polymerase chain reaction) to amplify and produce many strands of fluorescently-dyed cDNA from the RNA. These strands of DNA essentially represent one-half of the gene that it came from and will hybridize strongly to its complement on the genechip. The next step basically involves flooding the slide with the cDNA, and complementary strands will hybridize with each other. Once this process is complete, a fluoresecence excitation energy source strikes the slide, and another instrument detects the light emitted from the fluorophore (dye). It is important to note that microarrays are always performed on **two** samples – a wild-type and an experiment. Thus, the **relative intensities** of

light emitted from corresponding genes of both samples can be compared to deduce whether or not the gene has been turned on. We nteracted with the GenePixPro program from Acuity, which seems to have many regression algorithms built in to filter noise from the emitted light for more accurate and representative data.

3 Experience at a Bioengineering Lab

In order to gain experience with the lab techniques necessary for using the genechips, we observed Dr. Sagit Levanon, Karen Chao, and Jeff Reitsema from Dr. San's lab over the span of four weeks. During this time, we were able to observe an *E. Coli* fermentor/bioreactor, the process of PCR, and the use of High Performance Liquid Chromatography. The *E. Coli* bioreactor served the purpose of growing cell cultures in a controlled environment. Using the bioreactor, Dr. Levanon was able to control the temperature, pH, dissolved oxygen, and glucose feed, as well as other factors. Much care had to be exercised to prevent contamination of the glucose feed which, if contaminated, would cause the cell culture to begin growing in the glucose, ruining the experiment. Karen Chao and Jeff Reitsema, both undergraduates, worked on genetically engineering *E. Coli* to produce more succinate. In the citric acid cycle, both the succinate and ethanol pathways compete for available NADH. To yield more succinate, the ethanol pathway had to be eliminated. Since this pathway relies on alcohol dehydrogenase (ADH), knocking out the Adh gene would theoretically eliminate the ethanol pathway. Observing their work on this project exposed us to the technique of genetic recombination – using plasmids and vectors to insert or delete certain base sequences, and testing if such a transformation was successful by coupling the genes with an antibiotic-resistant gene. In this case, Karen and Jeff used Kenamyacin as the antibiotic. After all the transformations were complete, its success was tested by growing a colony in a medium with Kenamyacin. Colonies that survived indicated a successful deletion of the Adh gene.

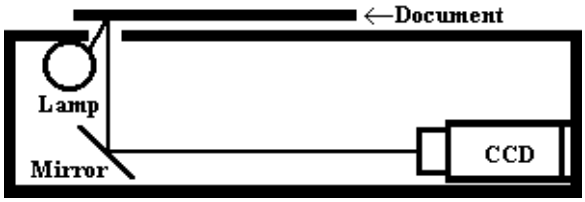
4 Scanners and Imaging

4.1 Motivation

I'll basically be talking about how scanners work from the placement of the document to be scanned, to it being viewed on a computer. The Axon GenePix 4200A works in a similar fashion as most scanners. Instead of scanning papers, pictures, and the like, however, it scans gene arrays. However, the principles discussed here still apply.

4.2 Scanner Components

The basic purpose of a scanner is to obtain an image and process it for replication or storage. A scanner is composed of many components, each of which assists with this goal. The core component of a scanner is a CCD, a Charged Coupled Device. A CCD contains an array of photo-diodes supplemented by an optical lens. The optical lens magnifies incoming light, which the photo-diodes use to generate an electrical signal which corresponds to the strength of the signal. Another important component of the scanner is the lamp. The light from the lamp illuminates the sample, and reflects onto mirrors, which are directed towards the CCD. All of the aformentioned components are components of the "scan head", and is controlled to scan lines by a motor.

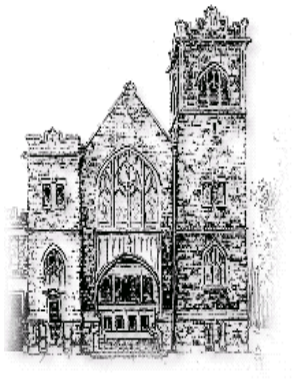


4.3 Scan Modes

Images can be scanned in four different methods, each of which retains varying amounts of information.

4.3.1 Line Art

A Line art scanned image only stores the black and white information of an image. Black pixels are represented using 1 bit of memory, while white pixels use no bits. This mode uses the least amount of memory and storage for a picture.



4.3.2 Halftone

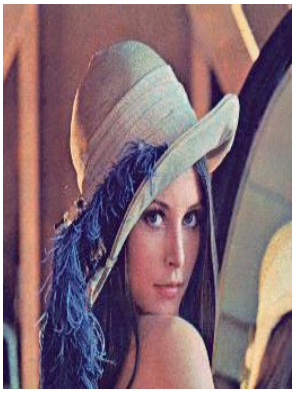
Because some printers cannot print out continuous shades of gray, the halftone method is used. The basic methodology is to create a matrix of white pixels, and have a number of them black. Newspapers are often printed using this type of image.

4.3.3 Grayscale

A grayscale image is a black and white picture in which a number from 0-255 is assigned to each image. 0 corresponds to white, 255 to black, and numbers in between are shades of gray. Each pixel takes up 8 bits of memory.

4.3.4 Color

A color image is composed of the 3 basic colors, red, blue, and green. For each pixel, the intensities of each color is assigned a value from 0 to 255, with zero corresponding to black (off), and 255 to the brightest color. Each pixel can thus display up to 16.77 million unique colors.



4.4 Display

Although some scanners use more than 24 bits per pixel, for display purposes, only 24 are used. For example, the Axon GenePix micro-array scanner uses 16 bits per pixel for each color while scanning. However, for display purposes, only 8 of those 16 bits are used. There are two options to reduce the size from 16 bits to 8 bits, truncation or a square root transform. The high, middle, or low bits can be selected, and the rest discarded. Alternatively, one can apply a square root transform, which simply computes the square root of the decimal value of the image intensity.

5 Metabolic Flux Balance and Succinate Yield Optimization

5.1 Introduction

We use to use the principles of metabolic flux balance in order to represent The Embden-Meyerhof metabolic pathway, its inputs, and its flow rates, as a system of linear equations $Sv = b$. Specifically, we use this model to maximize the production of succinate, and to predict the effects on succinate production caused by the introduction of new reactions.

We begin with the reactions describing the Embden-Meyerhof pathway:

$$
\begin{aligned}
Glucose + PEP &\rightarrow G_6P + Pyruvate && (1)\\
G_6P + ATP &\rightarrow 2G_3P + ADP && (2)\\
G_3P + NAD + ADP &\rightarrow PEP + NADH + ATP && (3)\\
PEP + ADP &\rightarrow Pyruvate + ATP && (4)\\
PEP + CO_2 &\rightarrow OAA && (5)\\
Pyruvate + CoA &\rightarrow Formate + AcetylCoA && (6)\\
AcetylCoA + ADP &\rightarrow Acetate + ATP + CoA && (7)\\
AcetylCoA + 2NADH &\rightarrow Ethanol + CoA + 2NAD^+ && (8)\\
Isocitrate &\rightarrow Glyoxalate + Succinate && (9)\\
AcetylCoA + Glyoxalate &\rightarrow Malate + CoA && (10)\\
OAA + NADH &\rightarrow Malate + NAD^+ && (11)\\
Malate &\rightarrow Fumarate && (12)\\
Fumarate + NADH &\rightarrow Succinate + NAD^+ && (13)\\
AceytlCoA + OAA &\rightarrow Isocitrate + CoA && (14)
\end{aligned}
$$

We generate our 12×14 stiochiometric matrix $\mathtt{S}$ by encoding the coefficients of each reaction into a column of $\mathtt{S}$:

$$
S = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
-1 & 0 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & -1 & 0 & 0 & 0 & -1\\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0\\
0 & 0 & 1 & 0 & 0 & 1 & 0 & -2 & 0 & 0 & -1 & 0 & -1 & 0
\end{pmatrix}
$$

The inputs to our system are encoded in the 12×1 vector $\mathtt{b}$:

$$
b = \begin{pmatrix} 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^T
$$

As glucose is the sole input, b_1 is 100, and b_n for $n > 1$ is 0. Succinate is produced in reactions 10 and 14, so we desire to solve:

$$max_{Sv=b,\ v\geq 0} f^T v$$

where:

$$f^T = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}^T$$

As $Sv = b$ is an underdetermined system, $v = S^+ b + Ey$, where E is the nullspace of S, and y is an arbitrary vector of appropriate dimensions. The aforementioned maximization problem can be rewritten as:

$$max_{Ey \geq -S^+ b} f^T Ey$$

Which is

$$max_{Ey+S^+ b \geq 0} y_1 + y_2$$

A standard linear programming problem. We can find a solutions in many ways, by using a searching algorithm such as the simplex algorithm, or, in simple cases, by solving the inequalities by hand. The simplex algorithm, however, produces only one solution, when many may exist. If many solutions exist, then there are many possibly flow rates `v` that maximize production of succinate. This is a useful property, at it gives experimentalists more freedom in producing optimal yields of succinate.

Even more interesting results can be obtained by adding new pathways to our original system, often increasing the possibile outputs.

6 Automation of Data Extraction

As the above process can become tedious when dealing with complex reactions, we attempted to automate the process of generating the stoichiometric matrix. The program was written in Matlab and required the user to input the reactions of interest into a plain text file. The syntax for the reactions is 'a + b + ... `-->` c + d + ...'. After parsing through the text file line by line and turning each line into a vector of strings, the program makes use of essentially two functions – QueryMolecules() and StripCoeff(). QueryMolecules() simply takes in one reaction, compares each reactant or product with that of a 'molecule vector,' and inserts a certain molecule if it doesn't already exist. This makes use of StripCoeff(), which takes in one molecule, i.e. 2NADH, and returns the coefficient and base, i.e. coeff=2 base=NADH. Also, there exists a 'multiplying factor' that determines whether a molecule is a reactant (negative value) or product (positive value) in the reaction. The multiplying factor is initially negative, and if the function parses the `'-->'` string, then the multiplying factor simply becomes positive along with the coefficient. With these tools, it is quite straightforward to form our stoichiometric matrix.

7 Future Directions

From the basic project of genechips, we have already gone to explore many tangents. This leaves only more tangents to research in the future, including database extraction, reaction drawing, and regression analysis.

We would like to automate the process of data extraction even further by interfacing our program with an enzyme database, namely Brenda (www.brenda.org). As speculation, such a feat would most likely require a modification of the syntax used to identify reactions, and this is assuming that reactions from the database have a consistent structure to them. Along the same lines of automating further, we would also like to devise a way to actually draw the reactions in a logical fashion to serve as a visual aid. As reaction coupling and interaction becomes more complex, visualization becomes more important. We have already touched the surface of such a project, experimenting with tree structures and tree-drawing functions in Matlab. So far the hurdle to overcome seems to lie in finding a tree structure that will suit our needs – one that can handle multiple parents as well as multiple children. Most likely we will have to write our own tree structure as objects, with each object able to hold information pertinent to our purpose.