Report of Term Project
Using relational databases to analyze microarray probes and single nucleotide polymorphisms

Report dated July 21, 2005

Author:
Abhijit Phatak
Graduate Student, CECS Dept., University of Louisville, KY

Advisor:
Dr. Eric C. Rouchka
Director, Bioinformatics Laboratory, University of Louisville, KY

(Project undertaken for CECS 696 course of the Master's Degree in Computer Science)

Using relational databases to analyze microarray probes and single nucleotide polymorphisms

Phatak, Abhijit*
(Graduate Student, CECS department, University of Louisville, KY)

Submitted: July 21, 2005

* Address for communication: awphat01@louisville.edu, **abhi2766@yahoo.com**

Abstract

Microarrays such as those from the Affymetrix Inc[1]provide a very useful means of studying thousands of genes for DNA analysis and expression levels and are also valuable in the study of single nucleotide polymorphisms (**SNP**s). While the physical use of gene expression microarrays involving the assessment of expression levels by 'washing' the arrays with extracted mRNA is their primary purpose, the information on these microarrays can itself be used in various research efforts without conducting actual physical tests on the product.

This project focuses on creating a relational database of sequence alignment searches of probe data from an Affymetrix microarray against sequence data from the publicly available **dbSNP**[2] and human genome databases as well as setting the process of analyzing the results of these searches into motion. The objective is primarily to study information from microarray experiments that is typically discarded during analyses of the results from such experiments for potentially useful answers to various genetic research questions.

Introduction

Single nucleotide polymorphisms (SNPs, often pronounced as *snips*) are the focus of an increasing number of research efforts due to the potential they have in providing clues to various diseases and abnormalities, including cystic fibrosis [1], sickle cell anemia [2], muscular dystrophy [3], Type II diabetes [4] and migraine headaches[5]. In sickle cell anemia, for example, the change of an adenosine base in a healthy individual's hemoglobin to a thymine base causes the inserted protein to become valine instead of glutamine (Figure 1.1 and 1.2). This results in the diseased individual's blood containing cells that look like sickles, thus giving the disease its name.

```
>gi|28302128|ref|NM_000518.4| Homo sapiens hemoglobin, beta (HBB), mRNA
ACATTTGCTTCTGACACAACTGTGTTCACTAGCAACCTCAAACAGACACCATGGTGCATCTGACTCCTGA
GGAGAAGTCTGCCGTTACTGCCCTGTGGGGCAAGGTGAACGTGGATGAAGTTGGTGGTGAGGCCCTGGGC
AGGCTGCTGGTGGTCTACCCTTGGACCCAGAGGTTCTTTGAGTCCTTTGGGGATCTGTCCACTCCTGATG
CTGTTATGGGCAACCCTAAGGTGAAGGCTCATGGCAAGAAAGTGCTCGGTGCCTTTAGTGATGGCCTGGC
TCACCTGGACAACCTCAAGGGCACCTTTGCCACACTGAGTGAGCTGCACTGTGACAAGCTGCACGTGGAT
CCTGAGAACTTCAGGCTCCTGGGCAACGTGCTGGTCTGTGTGCTGGCCCATCACTTTGGCAAAGAATTCA
CCCCACCAGTGCAGGCTGCCTATCAGAAAGTGGTGGCTGGTGTGGCTAATGCCCTGGCCCACAAGTATCA
CTAAGCTCGCTTTCTTGCTGTCCAATTTCTATTAAAGGTTCCTTTGTTCCCTAAGTCCAACTACTAAACT
GGGGGATATTATGAAGGGCCTTGAGCATCTGGATTCTGCCTAATAAAAAACATTTATTTTCATTGC
```

Figure 1.1
Sequence showing Adenosine in hemoglobin sequence of healthy individual

```
>gi|28302128|ref|NM_000518.4| Homo sapiens hemoglobin, beta (HBB), mRNA
ACATTTGCTTCTGACACAACTGTGTTCACTAGCAACCTCAAACAGACACCATGGTGCATCTGACTCCTGA
GGTGAAGTCTGCCGTTACTGCCCTGTGGGGCAAGGTGAACGTGGATGAAGTTGGTGGTGAGGCCCTGGGC
AGGCTGCTGGTGGTCTACCCTTGGACCCAGAGGTTCTTTGAGTCCTTTGGGGATCTGTCCACTCCTGATG
CTGTTATGGGCAACCCTAAGGTGAAGGCTCATGGCAAGAAAGTGCTCGGTGCCTTTAGTGATGGCCTGGC
TCACCTGGACAACCTCAAGGGCACCTTTGCCACACTGAGTGAGCTGCACTGTGACAAGCTGCACGTGGAT
CCTGAGAACTTCAGGCTCCTGGGCAACGTGCTGGTCTGTGTGCTGGCCCATCACTTTGGCAAAGAATTCA
CCCCACCAGTGCAGGCTGCCTATCAGAAAGTGGTGGCTGGTGTGGCTAATGCCCTGGCCCACAAGTATCA
CTAAGCTCGCTTTCTTGCTGTCCAATTTCTATTAAAGGTTCCTTTGTTCCCTAAGTCCAACTACTAAACT
GGGGGATATTATGAAGGGCCTTGAGCATCTGGATTCTGCCTAATAAAAAACATTTATTTTCATTGC
```

Figure 1.2
Sequence showing Thymine in place of Adenosine at marked location for diseased individual

Microarray technology[1] has proven to be a potent tool in furthering a wide range of genetic research including SNPs. In particular, microarrays produced by Affymetrix provide a very useful way of conducting research in SNPs. The typical Affymetrix microarray consists of hundreds of thousands of probes representing thousands of genes. The probes are each just twenty-five bases long. Since a SNP relates to a variation in a single base within a segment of DNA, the relatively small size of the probe allows the researcher to focus his/her attention close to the locus of a known SNP.

The current project utilizes this specificity of probes from an Affymetrix microarray to find highly similar sequence segments in the most authoritative public SNP database currently available, namely the dbSNP[2] database maintained by the National Center for Biotechnology Information (NCBI)[3]. The project focuses on searching the human SNPs section of dbSNP as available under the name Build 124 from the NCBI website[2].

The project, in many aspects, continues the groundwork laid down by an earlier project carried out as part of the requirement for the Introduction to Bioinformatics course (CECS 660) during the Spring 2005 term. This project expands the scope of the earlier one by searching the data for the entire human genome available in the dbSNP database as against the single chromosome data used in the previous project. Additionally, a parallel search of the same Affymetrix probes against the entire human genome itself, as it exists in the hg17 build of July 2004 available from the UCSC 'goldenpath' Genome Database[6], was conducted.

This project and the earlier one are remarkable in view of the following facts. There is much growing interest in the separate fields of SNP research and microarray research. It is also well understood that microarray technology provides a very useful means of studying SNPs. There has already been a considerable amount of research conducted in the area of the study of SNPs through the physical use of microarrays. A few examples of such research are available in the references section[7-9]. However, analyzing data across the two sources of SNP research, namely the SNP databases and the microarrays that provide one of the sources for creating these databases is not yet a widely used research technique. A quick search through the PubMed Central database available through the NCBI website[4] [10;11] for the key terms, microarray, SNP and database yielded some interesting references[12-14]. However, these are still not as numerous as are reported in many other areas and techniques of genetic research. There is, therefore, scope for such projects, not because it is uncommon but because there is potential in this approach for answering some pertinent genetic research questions. This project attempts to lay the foundation for such studies by turning the comparative data generated through the use of tools such as BLAST[5] into a relational database.

This relational database will, it is hoped, be helpful in furthering the study of both sources in a better manner. Indeed, time constraints on this project have meant that this generated database is limited to a search of data only from the human genome. However, the methods and the knowledge gained from conducting this project will certainly help facilitate similar and more ambitious studies of data from other genomes as well.

It must noted here that this project would not have been possible without the active guidance and support of Dr. Eric Rouchka, director of the Bioinformatics Lab at the University of Louisville. There are a number of other people whose help and support made this project possible. The Acknowledgments section makes note of these contributions.

Methods

Source Data

HG-U133A microarray

Since this project is an extension of a project undertaken as part of the CECS 660 term project during Spring 2005, the preliminary data for the current project was the same as that in the earlier project. Specifically, data from the Affymetrix HG-U133A microarray[6] was made available for the project by the advisor. HG-U133A is a gene expression microarray representing nearly 20,000 well-documented genes from the human genome. More specifically, both projects were conducted using 247,965 oligomers of 25 bases each representing the 'match' half of the microarray.

Microarrays from Affymetrix typically consist of probes that are twenty-five bases long that are expected to hybridize with the complementary mRNA samples that are washed over the array to study the expression levels of the genes on the array. However, these microarrays also contain a paired set of 25-base oligomers that are identical to the first set except for the middle (thirteenth) base, which is a complement of the base in the first set. This paired set is intended as a control measure to trace possible cross-hybridization during the testing process. The actual sequence segment is loosely referred to as a 'match' and the other half of the pair with the complement of thirteenth base as a 'mismatch'[7]. Since these pairs of probes differ only in a known specific location, it is possible to use only one half of each pair of probes in non-physical uses of the microarray data as is done in this project. The mismatch half of each pair can be inferred from the match data. If required at any stage, the mismatch data can be either used directly from the microarray or generated by substituting the middle base in each match probes with its complement.

Being a measure of control, results of experiments showing hybridizations with the mismatches are usually discarded as being the outcome of testing errors and random cross hybridizations. An important purpose of creating the relational database is to store even this regularly ignored information along with the information from 'matched' hybridizations to check if even a small amount of the discarded data contains any clues to genuine mutations and variations.

The data for the microarray probes was originally received in tab-delimited plain text file format (Figure 2). This was converted to the FASTA format[8] accepted by BLAST.

Probe Set Name	Probe X	Probe Y	Probe Interrogation Position	Probe Sequence	Target Strandedness
1007_s_at	467	181	3330	CACCCAGCTGGTCCTGTGGATGGGA	Antisense
1007_s_at	531	299	3443	GCCCCACTGGACAACACTGATTCCT	Antisense
1007_s_at	86	557	3512	TGGACCCCACTGGCTGAGAATCTGG	Antisense
1007_s_at	365	115	3563	AAATGTTTCCTTGTGCCTGCTCCTG	Antisense
1007_s_at	207	605	3570	TCCTTGTGCCTGCTCCTGTACTTGT	Antisense
1007_s_at	593	599	3576	TGCCTGCTCCTGTACTTGTCCTCAG	Antisense
1007_s_at	425	607	3583	TCCTGTACTTGTCCTCAGCTTGGGC	Antisense
1007_s_at	552	101	3589	ACTTGTCCTCAGCTTGGGCTTCTTC	Antisense
1007_s_at	680	607	3615	TCCTCCATCACCTGAAACACTGGAC	Antisense
1007_s_at	532	139	3713	AAGCCTATACGTTTCTGTGGAGTAA	Antisense
1007_s_at	143	709	3786	TTGGACATCTCTAGTGTAGCTGCCA	Antisense
1007_s_at	285	623	3793	TCTCTAGTGTAGCTGCCACATTGAT	Antisense
1007_s_at	383	479	3799	GTGTAGCTGCCACATTGATTTTTCT	Antisense

Figure 2
Sample source data from HG-U133A microarray in tab-delimited format

dbSNP

The data for the SNPs consists of the compressed FASTA format [8](Figure 3) files available for download from the NCBI dbSNP site[9]. As noted earlier, the data used corresponds to Build 124 of the dbSNP database, which is itself based on Build 33 of the NCBI Human genome BUILD 33.The data consists of referenced sequences containing SNPs (*refSNPs*) that have been verified and consolidated from multiple submission sources. Thus, each sequence in this database is annotated as 'rsNNNNNN' where N represents the sequence number. This data is available, unlike for other organisms, in files comprising SNPs found on each of the 23 human chromosomes. Chromosome 23 is appropriately broken into two parts, X and Y. Additionally, a file consisting of sequences that are likely to occur across multiple chromosomes and a file containing sequences that have not yet been mapped to any specific chromosome. These dbSNP files contain a total of just over 10 million sequences containing an important allele each[10].

```
>gnl|dbSNP|rs17105379_allelePos=101totallen=201|taxid=9606|snpClass=1|a
lleles='C/T'|mol=genomic|build=123
AAAGGTCACA ATTTAAGCAC TAATTGCATA TAGTTTTTCT TGACTTGGCA TTCAAGGGAT
GGGAAAACTC AATAGAAGAC TCTTGCAATA GCCCAGATAA
Y
GTGTAGATAG TTAGCAGAGG GAATGAACAG TAGTGAACAA AACCCAAAGA CACATCACAG
GCAAAAATCA ATTGGGTCTG GAAATACATT TAAGTTATGG
```

Figure 3
Sample sequence from dbSNP database in FASTA format (highlighted in box)

Human genome data

The other part of the project used data from the entire human genome for comparison against the Affymetrix microarray probes. This data is also available in compressed FASTA format[8] on the Kybrin Bioinformatics cluster[11] at the University of Louisville[11]with one file for each of the first twenty-two chromosomes and separate files for the X and Y chromosomes. The latest available version of the data dated July 2004, named *hg17,* was used.

Tools

WUBLAST 2.0

The tool most prominently used to carry out this project was the Washington University version of BLAST (referred to as WUBLAST)[5]. Over the course of many years, BLAST (Basic Local Alignment Search Tool) has proven to be the most popular sequence-matching tool available on account of its robustness and efficiency. BLAST was originally developed by the NCBI [15]and remains one of the most frequently used tools for sequence alignments. However, the version of BLAST developed at the Washington University at St. Louis, MO is equally popular and offers many enhancements over the NCBI version. In fact, version 2.0 of BLASTN contains many improvements over its own previous versions[12].

NCBI BLAST and WUBLAST have multiple variants geared towards finding alignments for different purposes. The entire set of BLAST tools from the Washington University is referred to as BLASTA and includes several utilities besides the main algorithms for making the searches more efficient. BLASTN compares nucleotide sequences; BLASTP performs protein-to-protein matching, while the other programs in BLASTA such as BLASTX, TBLASTN and TBLASTX work with translated queries and/or databases. The appropriate program for this project was BLASTN.

These tools are available both from NCBI and from Washington University in online form where researchers can submit query and target sequences in specified formats and choose to alter multiple default parameters depending on the amount of sensitivity desired. However, these online programs are suited to aligning either short sequences and/or a limited number of query/target sequences. For finding alignments where the sequences lengths are long and/or the number of searches is high, it is advisable to use the standalone versions of the programs that can be downloaded from the respective sites. This is what the project required and therefore the standalone BLASTN program available on the Kybin Bioinformatics cluster was used to find the desired alignments.

One other tool tested in the course of the previous project in Spring 2005 was SSAHA[16]from the Sanger Institute[13]. This is another tool for rapidly searching for exact or near-exact alignments and uses an efficient hashing algorithm to achieve them. However, SSAHA could not be used for the current project mainly on account of the restricted time schedule. Although the algorithm is known to be quite fast, there is a

learning curve involved in applying it for a project of this scale. In comparison, the author had greater experience with BLAST. It may be worthwhile, in follow-up projects, to explore the possibility of using tools such as SSAHA in addition to BLAST.

MPBLAST

mpblast is a Perl program developed by Ian Korf and Warren Gish[17]for multiplexing query sequences. The challenge with many bioinformatics research efforts is managing large amounts of query and target data efficiently. There are many logical and computational adjustments researchers can and should make to improve the efficiency of their research processes although there are, most often, trade-offs in using operational or programmatic 'tricks' of this kind. *mpblast* is one such tool for speeding up BLAST alignment searches and making better use of available computing resources.

Specifically, the program combines several short query sequences into chunks of concatenated queries. These chunks typically contain thousands of sequences each (the default is 100,000) with each sequence being separated by a delimiter that ensures that alignments with the target do not cross the boundary of each individual query sequence. The process allows this whole chunk of queries to remain in memory during computation and thus reduces the number of disk I/Os that would be required in fetching a database sequence separately for each query sequence.

As the authors note, *mpblast* offers up to a 10-fold improvement in efficiency in large datasets. This can be a significant percentage in projects that involve millions of sequences. There are, of course, some limitations to the amount of improvement *mpblast* provides in different circumstances and the author contends that the program tends to give better results for NCBI BLAST in comparison with WUBLAST. Even so, with a query set of nearly 250,000 small sequences to be matched against several hundred thousand sequences in each chromosome of the dbSNP database, *mpblast* did provide some improvement in computation during this project.

PERL SCRIPTS

Although the project calls for creating a relational database from the microarray, dbSNP and human genome data, the task was greatly facilitated by the use of some simple scripts written in the Perl[14] language. These scripts were needed to transform the tab-delimited data from the microarray into the FASTA format[8] accepted by WUBLAST, to truncate the dbSNP sequences to the forty-nine base segments mentioned in the sub-section on BLAST, to parse the output from the BLAST searches into a simpler format and also to turn this parsed data into tables in a MySQL[15]database. Perl is eminently suited to these tasks because of its powerful file-handling and string manipulation functions and because the Perl interpreter works smoothly in the Linux-based environment in which most of project was carried out. Although, the MySQL database is currently maintained on a desktop running the Windows™ XP operating system, the Windows™ version of Perl, ActivePerl (version 5.6)[16], developed by Active State[16] and installed on the desktop machine was useful in populating the database tables.

Although the author had almost no experience with programming or using Perl prior to this project, he was able to educate himself enough to write and compile the programs that were required. In this effort, once again, acknowledgment is due for the active guidance from Dr. Rouchka who directing the author to several useful resources for learning the language and applying it for the purposes of the project.

MYSQL®

The object of this project was to create a relational database of exact or near exact alignments of pairs of the probes from the Affymetrix microarray with, on the one hand, the referenced sequences from the dbSNP database and on the other, with the sequences from the human genome. This is a fundamental step towards using the generated data for analysis of the expression levels of the genes represented on the microarray. At this stage, any standard relational database software would have sufficed. As such, the latest version, version 5.0, of the open-source MySQL database server[17] was installed on the Windows™ desktop computer, along with the very useful graphical versions of the MySQL Administrator[17] and MySQL Query Browser[17]. Version 5.0 of MySQL provides several enhancements over the previous versions, including support for running stored procedures. Although there are some experimental aspects in this current version, they do not come into play when storing the results of the current project. It is hoped and expected that by the time any of these advanced features of the software are required in future research, these issues will have stabilized and the software can easily be upgraded to those newer versions as and when required. In the event that the current version appears to have significant issues that affect such future research, the developers allow the downgrading of the product to the earlier and well-tested version 4.0.

The advantages of using MySQL at this stage lie in the open-source nature of the software as well as the relatively lower level of complexity involved in installing and using MySQL as opposed to the popular commercial systems such as Microsoft® SQL Sever 2000 or Oracle® 9i. MySQL, especially the current version 5.0, provides strong support for all the regular database operations as the expensive commercial versions do. It is the preferred database system of millions of small and medium businesses around the world and has proven to be extremely stable, as time has gone on. It works very well with the open-source PHP scripting language as well as with Perl, the programming language used in this project.

Apart from these advantages, the MySQL data as well as the database schemas are maintained in easily portable formats. Queries can be stored in multiple formats including the *sql* format understood by SQL Server 2000. Result sets can, similarly, be exported into multiple formats such as *csv, html* and *xml*. Thus, if at a later stage, the data and/or the database schema need to be exported for use on other popular database systems, they can be so exported quite easily.

COMPUTING ENVIRONMENT

The bulk of the project was conducted using the nodes of the Kybrin Bioinformatics cluster at the University of Louisville[11]. The cluster consists of sixteen nodes, each running on dual AMD® 2400 processing units, using a memory size of 2 GB and running on the Red Hat™ Linux operating system. The cluster has a combined storage capacity of 2 terabytes.

Access was also provided to a desktop workstation on the Bioinformatics Laboratory intranet, which is directly connected to the Kybrin cluster through a gigabit switch. The workstation also has dual AMD® 2400 processors and 2 GB memory and an 80 GB hard drive and can run either the Microsoft® Windows™ XP (SP2) operating system or the open-source Fedora version 9.0 from Red Hat Corporation.

The large volume of the query and target data used in the BLAST searches as well as the substantial size of the result sets required the use of such high-storage capacity systems as the cluster and the workstation. Equally important, on account of the size of the source data and the desired level of sensitivity from BLAST, the powerful, multiple nodes of the cluster proved to be very useful in spreading the load of computation and in reducing the overall computation time. A maximum of ten nodes were simultaneously used in conducting the searches although more commonly a smaller number was used, typically four to five, so as not to tie up resources that may have been required by other researchers and users of the cluster.

The source data and results were all stored on the cluster and compressed using the popular *gzip* data compression utility. Wherever possible, this compressed data was extracted to the standard input and piped to the appropriate program without actually decompressing the zipped source files. Backup copies of the source data and results were stored on the desktop workstation as well as transferred to removable storage in the form of CDs and DVDs.

The process

In essence, the steps involved in conducting the project consisted of the following:

- Acquiring and downloading the source data
- Preprocessing source data
- Running the BLAST alignment searches on processed source data
- Parsing and storing search results in formats appropriate for the project
- Translating the stored results into MySQL database tables
- Analyzing the results

The details of the above are as follows:

Data Acquisition

As noted earlier the sources of the data were the Affymetrix HG-U133A microarray, the referenced sequences for known SNPs in the human genome available in twenty-seven compressed FASTA files from the NCBI dbSNP database and twenty-five compressed FASTA files for the human genome from the UCSC.

For the microarray, the data used were the 247,965 twenty-five base oligomers representing the exact 'match' probes. The SNP data was from Build 124 of the **dbSNP** database. The human genome data was from the latest build *hg17* from the UCSC Genome Database as available in July 2004. Dr. Rouchka made all the data from the three sources available for this project.

Preprocessing

The source data mentioned above required to be preprocessed for running the focused alignment searches using **WUBLAST**. Although WUBLAST accepts data in more than one format, the FASTA format is most popularly used. The microarray data was in a tab-delimited text file and had to be turned into the FASTA format using a simple Perl script. The dbSNP and UCSC data was already in FASTA format but still needed some transformation for the purpose of the project.

The sequences in the **dbSNP** database contain a single instance of an important SNP each, which is denoted by the standard IUPAC[18] code in most cases. The sequences typically range from a few hundred to a few thousand bases in length. However, the location of interest for the project was the segment surrounding the SNP for alignment with the twenty-five base oligomers from the microarray. As such, another Perl script was used to create a subset of forty-nine bases from each dnSNP sequence spanning twenty-four bases to the left and to the right of the locus of the allele in the majority of the cases (Figure 4). This meant that all possible alignments containing the SNP could be found when matched against the microarray probes. In cases where the allele position or the length of the original sequence did not leave twenty-four bases on either side of the allele, the resulting segment became less than forty-nine bases long. So, for instance, if a dbSNP sequence was 101 bases long and the allele was noted to occur at base 95 in the sequence, the truncated sequence of interest was taken at twenty-four bases before the SNP and the remaining 6 bases in the sequence after the SNP position. The truncated sequence would, therefore comprise the 31 base segment starting from base 71 and ending in base 101.

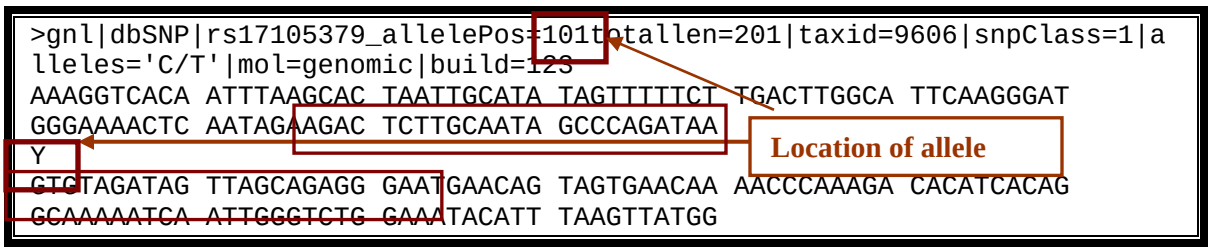

Figure 4
SNP of interest is assigned separate line in sequence. Boxes show the segments typically concatenated and used from such source sequences for the actual searches.

Additionally, the original dbSNP sequences are soft-masked for low-complexity regions and tandem repeats that are of low significance, by using lowercase symbols for the nucleotides. Although masking of this sort is often advisable for removing known non-informative regions and focusing on regions with potentially more information, it was not a conducive approach for this project. At the risk of producing several alignments of little significance, it was necessary to unmask the source data since, as noted before, the main objective was precisely to sift through data that is normally discarded forthwith for possible clues to the causes of various diseases and harmful mutations. Masked data is one such source of routinely bypassed or discarded information that just may be useful after all. Hence, the truncated segments from the original dbSNP data were restored to uppercase symbols so they would not be ignored in the BLAST searches.

Verification of this shortened dataset was essential to ensure that the process of extracting the smaller segments had no errors in it. This was again done using Perl scripts to compare the original and the truncated data.

Running BLAST

After creation of the reduced dbSNP dataset it was formatted into a BLAST database using the *xdformat* utility, which is part of the BLASTA package. This step is necessary since both NCBI BLAST (which uses *formatdb* in place of *xdformat*) and WUBLAST create their own formats for the target database for improved efficiency.

As for the query probes from the microarray, the *mpblast* program originally written by Ian Korf and Warren Gish[17] was used for greater efficiency.

The next important issue was deciding on the parameters to use for running the WUBLAST searches. Since the objective was to create a database of alignments between the microarray probes and the two other databases for analysis of the number of matches and mismatches involved in the possible hybridization of typical mRNA samples with the probes, the need was to find matches that were identical to the probes or very nearly so. These alignments would ideally include gapped as well as ungapped alignments. For the purpose of the current project, ungapped alignments were considered. Typically, other

factors remaining the same, ungapped alignments are expected to take less time to locate than those with gaps, since BLAST would need more computations to locate the best alignments with gaps satisfying the search parameters. Indeed, even when gapped alignments are allowed, BLAST first looks for ungapped alignments and then for the ungapped ones. As it turned out, the time available for the project was just enough to complete the ungapped alignment searches, leaving the expanded search for gapped alignments for a future effort.

As stated earlier, the appropriate program from the BLAST suite of tools for matching nucleotide sequences is ***BLASTN***. The default parameters for this program include a word size of 11 for nucleotide sequences and a scoring scheme where matching base pairs get 5 points, mismatches -4 and where gap opening and extending penalties are both set at 10 points. Thus, a perfectly matching alignment of 25 base pairs would score 125, an alignment of 25 base pairs with two mismatches would score 107 and an alignment of 25 base pairs with two gaps would score 95. By default, the program looks for gapped as well as ungapped alignments.

The word size is used as a seed to look for matches against the database. When a match is found the algorithm works in a greedy fashion by trying to extend the matched word on either side until the overall score of the alignment drops below a threshold value. A smaller word size is expected to increase the sensitivity of the algorithm, yielding potentially more and closer alignments than the default while a larger word size is expected to reduce the sensitivity. However, the gains in sensitivity with a smaller word size are most likely to be achieved at a cost of longer search time in databases of any considerable length.

There are a number of other parameters that can be set to fine tune the search process. A complete list is available at the WUSTL website[18]. After some trials with different word sizes and score thresholds, a word size of 8 and a raw score threshold of 98 proved to be appropriate for the desired output. The word size of 8 would allow for alignments with up to two mismatches on the maximum of twenty-five base-pair alignments that would be obtained by running the searches of the microarray probes against the dbSNP and the genome databases. The raw score threshold of 98 implied alignments looking for alignments of twenty-five base pairs containing a maximum of three mismatches. As such, a cut-off limit where ungapped alignments of at least 22 base pairs would be saved in the eventual database tables was set.

Another possible measure to improve the speed and efficiency of the BLAST searches that can potentially be used is to reverse the searches whereby the query becomes that target and the target becomes the query. This is a useful enhancement when the data that would normally be the target has fewer sequences than the query itself. For example, most chromosomes of the human genome have relatively few sequences although they are large ones. If however, the number of query sequences far outweighs the genomic sequences it might yield a considerable benefit in terms of time and computational efficiency to turn the query set into the database to be searched against. An attempt at this reversal was made when matching the microarray probes against the human genome

sequences. Although the search took less time, as expected, the quality of the results was not as expected. In essence, the sensitivity of the reverse search was not very encouraging. Therefore, the process was reverted to the regular situation of searching the probes against the genomic database. It must be noted that this situation arose at a time when the masked version of the genome data was still being considered for the searches. Time constraints did not permit extensive testing of the reverse search with the unmasked data to check for performance gains in comparison with the regular search method. The sample test of the masked and unmasked data for two chromosome files by reversing the query and search order did yield much faster results lasting from a couple of hours to about four hours per file. However, as noted before, the sensitivity of those searches was not optimal compared with the slower but normal method of using the microarray probes as queries against the other genomic database.

In sum, there are a number of options available to the researcher for achieving the closest possible results when using a tool such as BLAST. In view of the short duration of the current project, sufficient time had to be allocated for the actual search process to be sufficiently complete. This meant that more sophisticated search parameters had to be put off for future projects.

On an average, a BLAST search using multiplexing took from 18 to 24 hours for each of the chromosome files to complete. Of course, some files had fewer sequences; others had fewer matching data while some others had a lot of low-complexity regions, which slow down the search process. These differences meant that there was no easy way to predict how long each search would take.

Each search used two processor threads on a node, when available and also a considerable amount of the 2 GB memory on a node in the multiplexed form. As such, it was the norm to use one node per file when running the searches.

Multiplexing with *mpblast* was not suitable for searches against the genomic data since the number of sequences in the genome data files per chromosome is much less than in the case of the dbSNP files. However, the larger length of genome sequences compared to the short 49-base segments from the dbSNP data meant that even without multiplexing a considerable amount of data remained in memory when a search was on. Therefore, here too, one node was assigned one file during the search process.

One other factor that took up considerable time and effort was monitoring the progress of the BLAST searches. In view of the factors mentioned above as well as vagaries of the operating system, major searches were likely to become dormant after some time, logging out of the system while a search was running could have unpredictable consequences in some cases and sometimes the pipes from the BLAST output to the Perl filter script got broken. On account of such possibilities, the searches had to be constantly monitored while they were running and thoroughly verified on completion. Logs of the input parameters, the job numbers and the node number had to be maintained to track the progress of the process. Although remote access to the computing nodes was available through secure shell connections, it had to be used sparingly and carefully since

disconnection midway through a search, especially if the search was not running in the background, could terminate a job without warning.

However, these issues in conducting the project can and should serve as guidelines for future projects. Extensive logs of the process as well as this report should serve that purpose well. In a sense, navigating through such issues is a capital investment for a relatively untried venture such as this project.

Parsing and storing the results

The BLAST searches were conducted chromosome-wise, keeping the structure of the source data intact. However, the output from BLAST searches contains much more information than was practically required for creating the database tables. As such, the normal BLAST output was piped to Perl scripts to filter out the basic statistical information required for a database table (Figure 5). This included the following:

- Reference numbers of the query and target sequences,
- Their locations on the microarray and on the dbSNP and genomic databases,
- The location of the SNP within a dbSNP segment,
- The lengths of the query and target sequences,
- The start and end positions of the alignments found,
- The aligned segment pairs
- The alignment string itself, which is a means of depicting the matched and mismatched base pairs within a sequence. Matching pairs have a '|' between them, mismatches remain blank and where a base in the query sequence matches any one of the possible variations of the SNP, a plus sign ('+') is used to show this 'partial' match. A sample of this output is shown in Figure 5
- The length of the alignment found
- The number of matches within the sequence
- The percent identity of the matches (number of matches divided by the alignment length)
- The raw score of the alignment from standard scoring scheme of WUBLAST

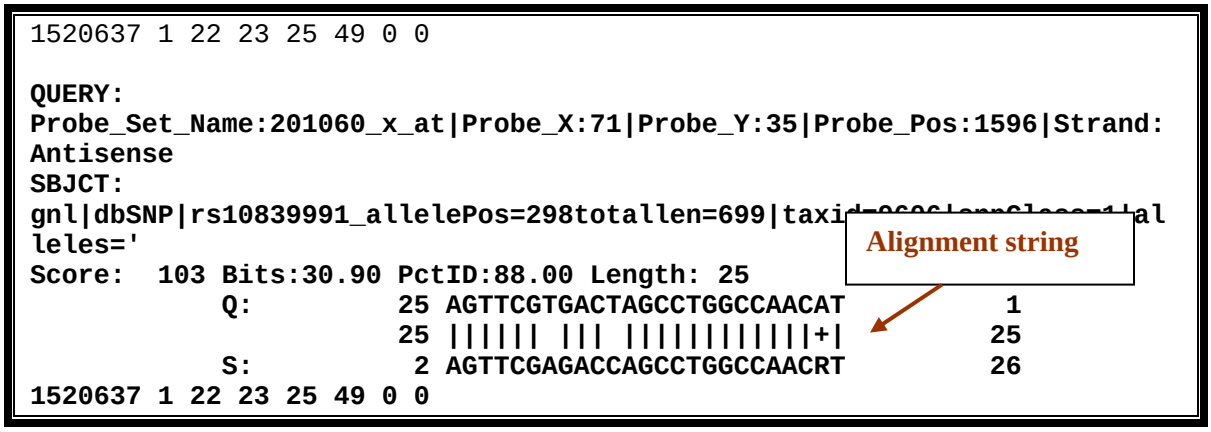

Figure 5
Sample of the parsed output from BLAST for a dbSNP sequence with a '+' at SNP position

For the results of the dbSNP alignments, since the query sequences were multiplexed with *mpblast*, a slightly modified version of *mpblast* itself was used to conduct the filtering of the output while in the case of the genome sequences, a Perl script that used the *BPlite* [19]Perl module (also originally developed by Ian Korf) was written to parse the output. In fact, mpblast also uses *BPlite* to capture the BLAST results in a structure of which the annotation and statistical information noted above are fields than can be easily referenced for parsing the output.

The parsed output was maintained in plain text files on the Kybrin cluster hard drives with one file corresponding to each input file. Copies of these were stored on the Windows™ desktop as a backup as well as for use in turning the text files into the My SQL database tables. Additional copies of these results were also burned to CDs and DVDs.

An important part of the process of storing the preprocessed source data as well as the results was to ensure the integrity of the data through the various transformations they went through. In fact, a considerable amount of time and effort was spent in carrying out these checks through various means such as running scripts, using the *grep* utility available on most UNIX and Linux systems, and cross-checking the number of input records to the output.

Creating the MySQL database tables

From the parsed output it was a fairly straightforward, if tedious task to further split the lines of each alignment into formats suitable for the MySQL database and conduct the transfer of data from the text files to the database. In this effort the Active State version of Perl and the DBI and DBD modules that help establish the connection between the files and the tables were used.

A new database was created in MySQL with a schema that had master tables for the source of the database sequences such as the dbSNP database and the UCSC genome database. Another master table was made to store the name of the organism involved in the searches. These measures were done in view of the likely extension of the scope of the research to other databases and organisms in the future.

The identification numbers for the source databases and organisms were used as foreign keys in the tables that were created to hold the results from the BLAST searches.

The database schema is as shown in Figure 6 below:

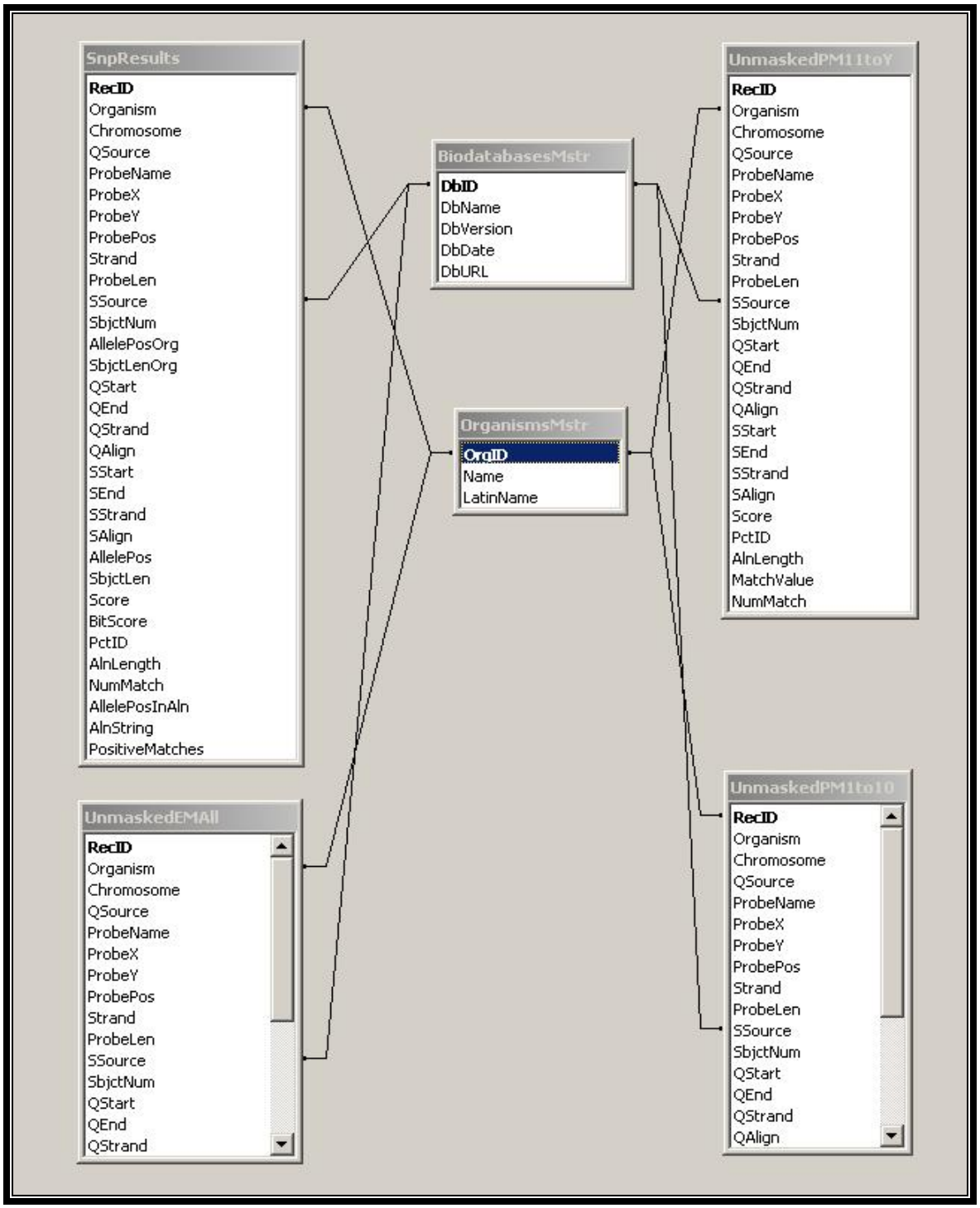

Figure 6
Database schema for the search results. Foreign keys for source database and organism reference the master tables for the values in those fields.

The database tables' schema followed the information obtained from the alignments. Some additional fields were added by calculating values from other fields (Figures 7.1-

7.3). So, for instance, a field was added to store the position of an SNP in the alignment itself from the knowledge of its position on the original string. The alignment string itself was also stored besides the aligned sequence segments for easy searching. The empty space that exists in the original string from the BLAST output representing a mismatch between a base pair was replaced with the character '*m*' for easier viewing and searching Figure 7.3).

One table holds the entire data for the alignments obtained from matching the microarray probes to the dbSNP sequences. The fields in this table are as shown in Figures 7.1-7.3.

RecID	Organism	Chromosome	QSource	ProbeName	ProbeX	ProbeY	ProbePos	Strand	ProbeLen
1	1	1	Affy_HG-U133A	117_at	510	685	1691	Antisense	25
2	1	1	Affy_HG-U133A	117_at	222	111	1697	Antisense	25
3	1	1	Affy_HG-U133A	117_at	370	699	1703	Antisense	25
4	1	1	Affy_HG-U133A	117_at	185	581	1727	Antisense	25
5	1	1	Affy_HG-U133A	117_at	185	581	1727	Antisense	25
6	1	1	Affy_HG-U133A	117_at	539	111	1733	Antisense	25
7	1	1	Affy_HG-U133A	117_at	452	459	1739	Antisense	25
8	1	1	Affy_HG-U133A	117_at	341	501	1871	Antisense	25
9	1	1	Affy_HG-U133A	117_at	341	501	1871	Antisense	25
10	1	1	Affy_HG-U133A	117_at	341	501	1871	Antisense	25
11	1	1	Affy_HG-U133A	117_at	278	325	1877	Antisense	25
12	1	1	Affy_HG-U133A	117_at	278	325	1877	Antisense	25
13	1	1	Affy_HG-U133A	117_at	278	325	1877	Antisense	25

Figure 7.1
Snapshot of records in the table containing alignment data for the dbSNP sequences

ProbeLen	SSource	SbjctNum	AllelePosOrg	SbjctLenOrg	QStart	QEnd	QStrand	QAlign
25	2	rs753856	73	777	25	1	minus	TCCTCTTCGGGAATCTTGTCCCTAA
25	2	rs753856	73	777	25	1	minus	CGCCTGTCCTCTTCGGGAATCTTGT
25	2	rs452004	201	401	25	1	minus	ATTTTGCGCCTGTCCTCTTCGGGAA
25	2	rs452004	201	401	24	1	minus	GGACTTCCCGACACTTGTCTTGCA
25	2	rs368844	29	610	25	1	minus	AGGACTTCCCGACACTTGTCTTGCA
25	2	rs368844	29	610	25	1	minus	CAGGCAAGGACTTCCCGACACTTGT
25	2	rs368844	29	610	25	1	minus	TCCAGCCAGGCAAGGACTTCCCGAC
25	2	rs439078	201	401	25	1	minus	GCTTGAGTGCCACAACTGCTGCCCC
25	2	rs394965	201	401	4	25	plus	GCAGCAGTTGTGGCACTCAAGC
25	2	rs391125	201	401	1	25	plus	GGGGCAGCAGTTGTGGCACTCAAGC
25	2	rs394965	201	401	1	25	plus	GCAGTTGTGGCACTCAAGCCCGCCA
25	2	rs439078	201	401	25	1	minus	TGGCGGGCTTGAGTGCCACAACTGC
25	2	rs391125	201	401	1	25	plus	GCAGTTGTGGCACTCAAGCCCGCCA
25	2	rs394965	201	401	1	25	plus	GTGGCACTCAAGCCCGCCAGGGGGA

Figure 7.2
More fields of same table as Fig. 7.1 showing database and query information per alignment

Score	BitScore	PctID	AlnLength	NumMatch	AllelePosInAln	AlnString
121	35.9	96	25	24	18	||||||||||||+||||||
121	35.9	96	25	24	24	||||||||||||||||||+|
121	35.9	96	25	24	2	|+||||||||||||||||||||
120	35.6	100	24	24	25	||||||||||||||||||||
121	35.9	96	25	24	10	|||||||+||||||||||||
121	35.9	96	25	24	16	||||||||||||+|||||||
121	35.9	96	25	24	22	||||||||||||||||||+|||
121	35.9	96	25	24	11	|||||||+||||||||||||
92	27.9	90.9	22	20	25	|||||||||m||m|||||||
112	33.4	92	25	23	18	||||||||||||m||+||||||
103	30.9	88	25	22	22	||||||m||m|||||||+|||
121	35.9	96	25	24	17	||||||||||||+|||||||
103	30.9	88	25	22	12	|||||||m||+|||||||m|||
103	30.9	88	25	22	16	||m||m|||||||+||||||||
121	35.9	96	25	24	23	|||||||||||||||||||+||

Figure 7.3
Remaining fields of dbSNP results table showing the statistics and alignment string for each alignment

The results from the searches involving the probes and the genome database sequences were stored in three tables with an identical schema mainly for convenience. The schema was similar to that for the dbSNP results except that specific fields for the allele (SNP) positions and the subject lengths were not relevant for these tables. One table holds all alignments of exact matches between the query and target sequences. The other two tables hold the less than exact matches. One table has data from chromosome 1 to 10 and the other from chromosomes 11 through X and Y. The reason for this division is that a combined table for all chromosomes would have required storage space of over 2 GB. Tables larger than that size need to be treated differently by the database program and require slightly different handling than regular tables. So, the entire dataset was divided into two parts to avoid the complexity that would have been introduced by one table of size larger than 2 GB. When required, using the appropriate join syntax in queries and stored procedures can accumulate data from both tables.

All tables have an auto generated primary key field since it is possible that the identifiers for the query and the target sequences, may be repeated in the table due to multiple alignments found for the same sequence.

The table for the dbSNP search results occupies 1.3 GB for its over 6 million records. If required, this can be broken down into small tables as was done for the results from the genomic sequence searches if handling the entire data in one table proves to be unwieldy or overly time-consuming.

Analysis of results

The next step in the process was to begin analyzing the results and that is the subject of the following section of this report.

Results and Analysis

Since this is a first effort at searching such large volumes of input data, reliable estimates of the amount of time and resources that would be required for these searches were not available. As such the search for ungapped alignments was taken up as a starting point. With estimates of time, computing resources and storage requirements from this process some basis for projecting the same could be estimated when looking for the expanded result sets that would include alignments with gaps as well as those without them. Of course, it must be borne in mind that being a new research process, some amount of time and effort that was spent in coming to terms with the various issues involved in the implementation of the research will, hopefully serve as a capital investment for future projects that build upon this one. Too, the over six million and over fifteen million ungapped alignments found between the microarray probes and the SNP segments and between the probes and the genomic sequences respectively should provide a strong base for beginning the analysis even as the search for a wider result set including gapped alignments is undertaken in future projects.

The major objective of this exercise in creating databases of alignments between microarray probe data and other databases is to provide the basis for conducting deeper analyses of microarray assays than those that focus mainly on gathering information from the intensity of the hybridizations. In cases such as that of Affymetrix microarrays, with their short, 25-mer match/mismatch structure, when physical experiments are analyzed, the entire result sets showing hybridization with the mismatch probes is completely discarded. A major hypothesis of this project is that it is likely that in at least a few cases, this approach amounts to throwing the baby out with the bathwater, so to speak. So, a prominent question a project of this nature seeks to answer is ‘what if there is valuable information in what is thrown away?’ Since a SNP is just that, a variation in a single base among different genotypes, it is worth mining the habitually discarded information and studying it more closely to find out if any of that is not just a case of random cross hybridization but is actually a site that contains a real SNP.

One possible offshoot of such research is gaining the detailed information necessary to make possible projections about the genotype of the test subjects. If samples from a particular individual or race show a sustained tendency to hybridize differently than the expected norm, it would be worth investigating whether the hybridization information is more than a testing issue and whether it actually reveals characteristics about that individual, that race or perhaps a disease that individual or race may be prone to. For example, individuals can differ in their zygosity on the same genes. A person may be homozygous for the gene but that may indicate presence of a disease while another person with the other variation of the allele may have no signs of the disease.

Another outcome of analysis of microarray data using the relational databases that may help to potentially map patterns of unusual hybridizations to certain diseases or genotypes is the design of more specialized arrays. These arrays would contain probes that hybridize

with certain sequences known to be involved in diseases or abnormal mutations and thus furthering the possibility of what is referred to as 'personalized medicine' – the ability to find a remedy for an individual problem instead of a more generalized solution.

As much as the search for SNPs that may be the cause of particular diseases and abnormalities is like searching for a needle in a haystack, it is still a worthwhile effort when even one such variation is finally identified positively.

With these possibilities in mind, a result set that consisted of a little over six million ungapped alignments of twenty-two base pairs or more between the microarray and the dbSNP database was found. Some other interesting figures are shown in Table 1. In particular, the 1,656 alignments with a mismatch only at the thirteenth base are of interest. Although it is not a large number, that is in line with expectations. It must be noted that several thousand sequences from the human repeat regions of the genome having annotations starting with 'Affx-hum' were not considered in these totals since they are known to be of little value for the purpose of this particular analysis.

SNP alignments	**Over 6,000,000**
Perfect matches	**45,984**
Mismatch in 13th base	**1,656**
Mismatch with allele position	**58,505**
Genome sequences hits	**Over 15,000,000**

Table 1: Summary statistics from database tables

An analysis of a little over 1000 alignments covering the entire length of the 25-mer probe in each case where the only mismatch is at the thirteenth base showed some interesting results. Bearing in mind that Affymetrix microarrays have pairs of probes where one half of the pair has the complement of the other's middle (thirteenth) base, a reason to study such a set of results is to find out the relative hybridizations between the match and mismatch pairs. If a higher percentage of hybridizations with the mismatch probes compared to the match probes appears to consistently occur that may raise questions as to whether those results are merely caused by random testing error or if they are indicative of possible variations or markers involved in a disease or harmful mutation.

Accordingly the test results of 14 experiments with 3 replicates each involving 1114 of these alignment probes with a mismatch at base 13 were plotted by taking the log-odds ratio of the matches to the mismatches (Figure 8.1). The data used was actual sample data for past experiments involving those probes made available by Affymetrix Inc, the producer of the microarray. The scatter diagram for these experiments shows a fairly high percentage of values at or below 0, indicating the region where the mismatches outweigh the matches. Another plot of these probes (Figure 8.2) representing the frequency of matches greater than mismatches and vice versa also shows an interesting pattern. Both graphs intersect smoothly and 263 of 1114 probes consistently have the mismatches higher than the matches. Although not high, it is still significantly more than expected. Also, a greater number of values would have been expected in the middle and with a greater scatter than is seen. As a first step in beginning to use the information from the

database tables it is certainly encouraging, in a sense, to come across such results that seem to very from the norm. Although far from being conclusive, they do evoke curiosity and the desire to dig deeper into the maze for more answers.

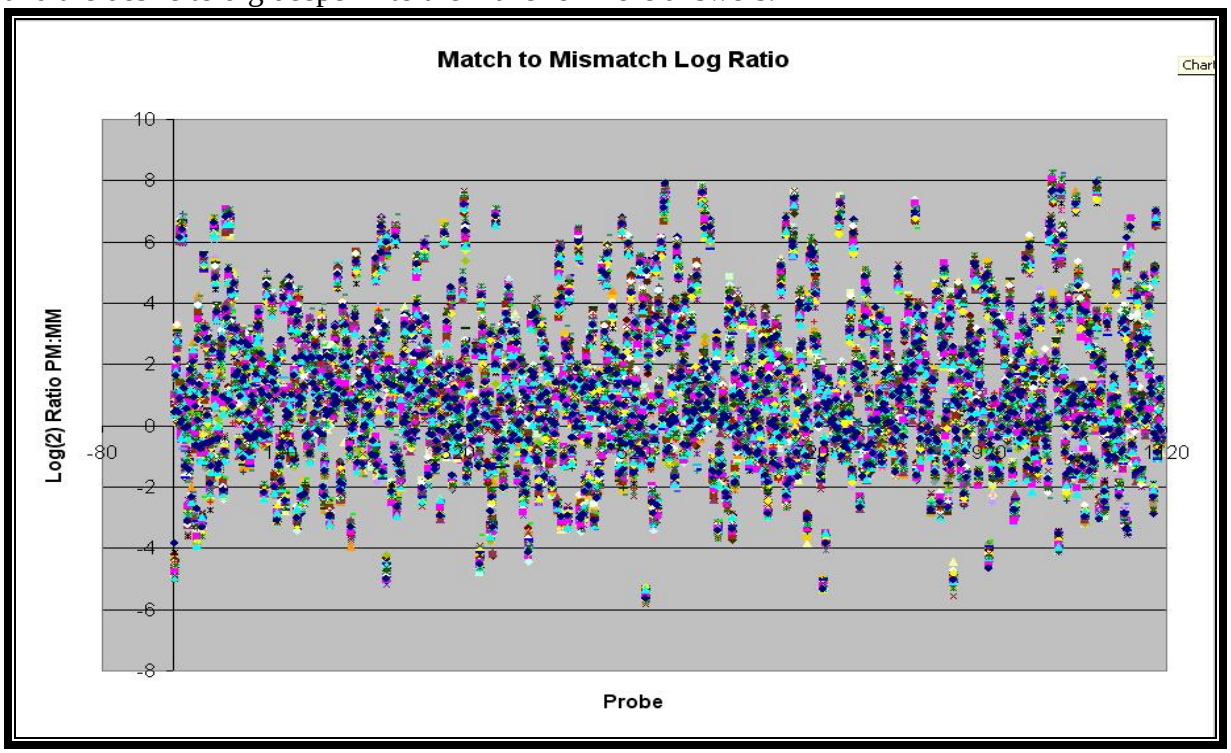

Figure 8.1:*Chart showing plot of log-odds ratio of perfect matches to mismatches for1114 query probes involved in alignments with 1 mismatch at 13th base (14 exp., 3 replicates each)*

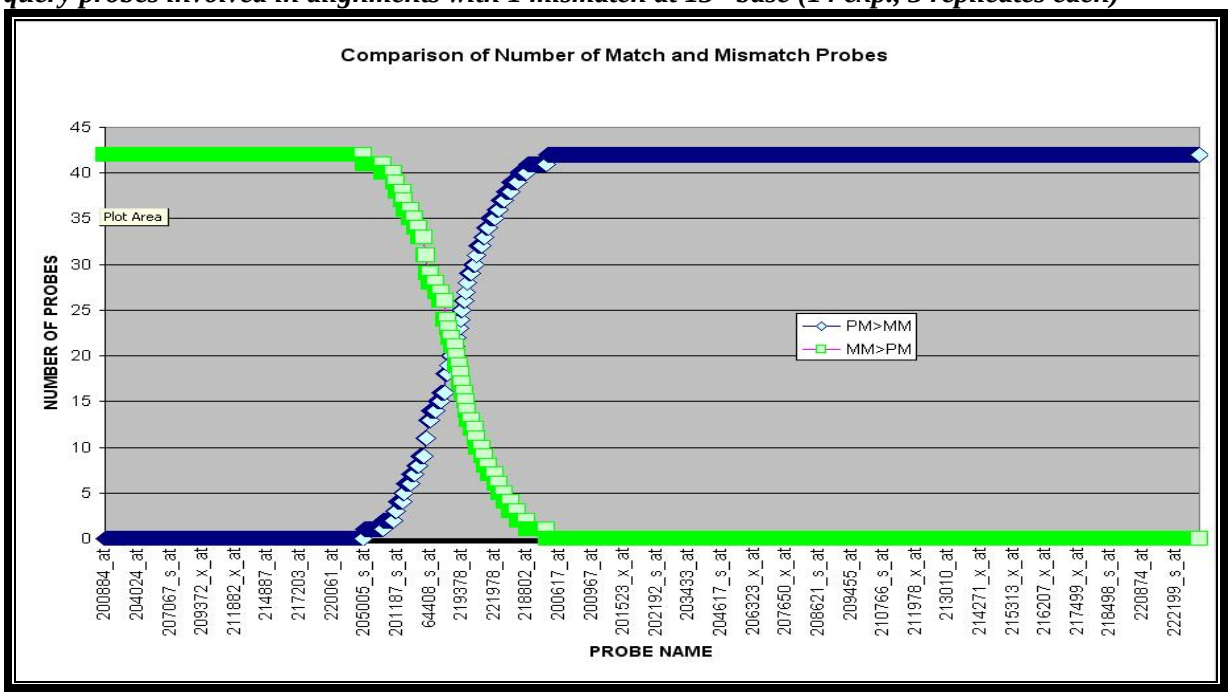

Figure 8.2:*Line chart showing number of times matches greater than mismatches and vice-versa for same experiments as for Fig. 8.1*

As a test of these seemingly unusual results, a test of 765 alignments where the alignments were still 25 bases long and containing only one mismatch but where the mismatch was noted to be at positions other than the thirteenth base was also run. Figures 9.1 and 9.2 below show the plots for this second set of tests. This time, the scatter of the log-odds values on the mismatch side i.e. below zero, is much less than in the previous set. However, the second plot shows a very even distribution between matches and mismatches, when the matches would have been expected to outnumber the mismatches more definitely.

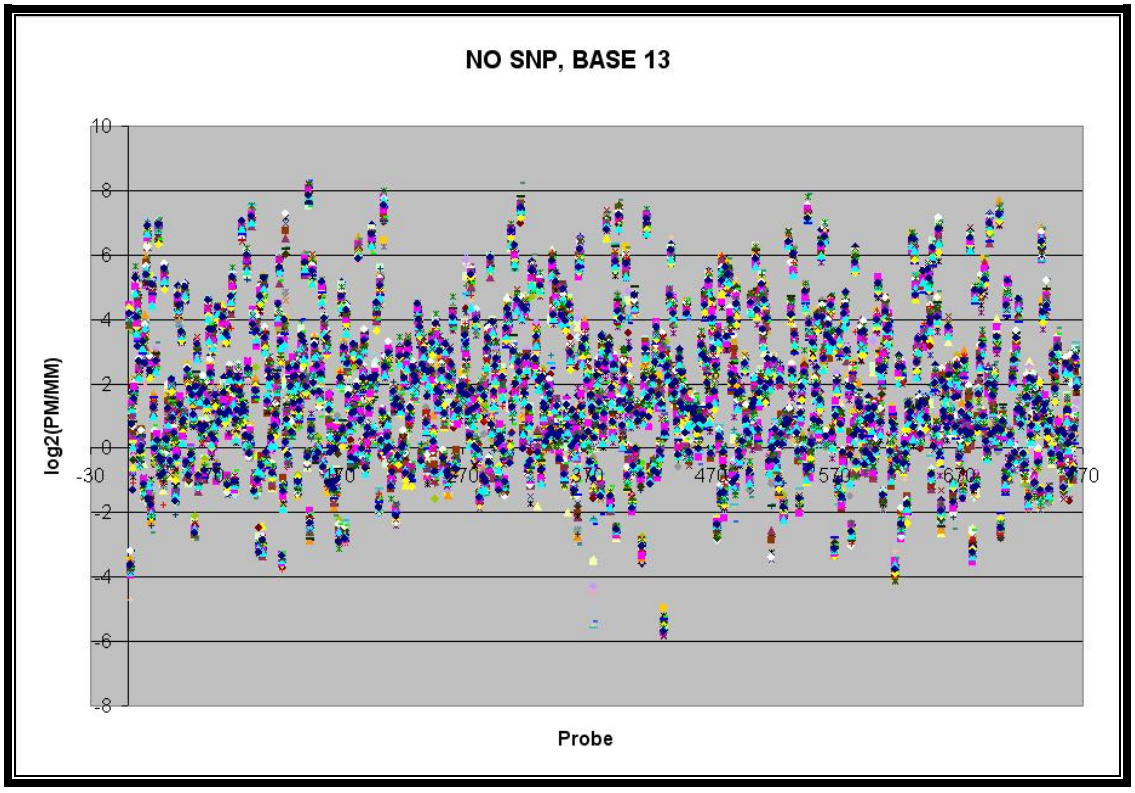

Figure 9.1
Plot similar to Fig. 8.1, of values for 765 probes with alignments having 1 mismatch, but not at 13th base

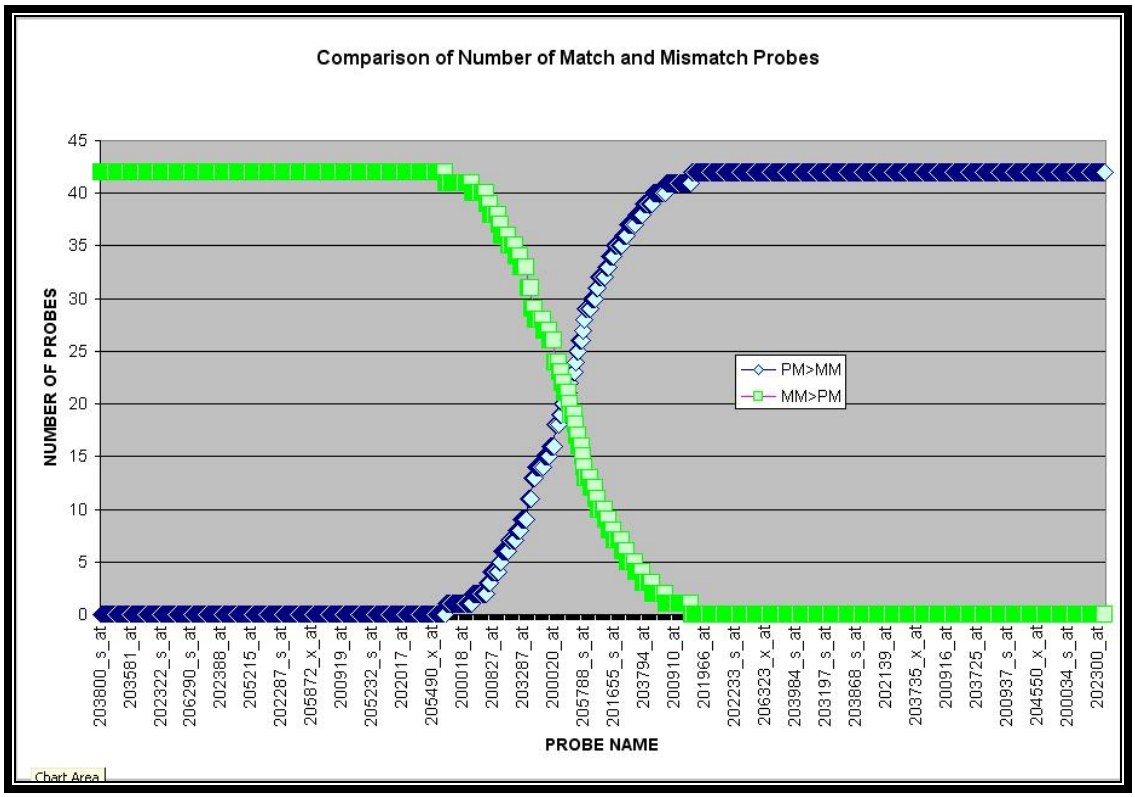

Figure 9.2
Line graph similar to Fig. 8.2 for second dataset of 765 probes

As noted before, although such results are very far from being conclusive of any kind of significant deviation from the normal they do highlight the value of generating the kind of information this project was intended for.

Discussion

The project provided the author with an opportunity to delve a little deeper into the areas of microarray analysis and SNPs. Although the time frame for the project was fairly small, the author did get to understand a great many of the issues involved in efforts on those aspects of genetic research.

Perhaps the greatest concern in such a project is the handling of large volumes of data in as efficient a manner as possible. This is not a new realization in terms of genetic research but is a significant issue for this kind of a project. With a longer timeframe, there will be better scope to test alternate tools and methods for better computational performance as well as for finding more focused results.

The section on running the BLAST searches notes some of the important obstacles that had to be overcome in the conduct of the project. The recording of those hurdles and the

measures taken to overcome them will, it is hoped, prove useful in making future efforts less laborious, less time-consuming and easier to manage.

Some future objectives include testing data from microarrays from sources other than Affymetrix such as from Agilent Technologies[20], another major producer of microarrays. Agilent's microarrays typically contain sixty-base probes as opposed to the shorter, twenty-five base ones from Affymetrix.

Also, expanding the current database to include alignments with gaps is a logical step in future projects. SNPs can be related to insertions/deletions as well as alternate bases at a locus. Alignments with gaps will help to study such variations better.

Other future goals, as suggested earlier include creating such databases for data from other species besides humans. This will allow comparisons across organisms for specific mutations and gene expression levels.

Despite the limited scope of the project though, the creation of the database is an important first step towards conducting larger and more in-depth research in future.

Acknowledgments

- Dr. Eric Rouchka – Director, Bioinformatics Laboratory, University of Louisville, KY
- Support from NIH: NCRR grant P20 RR 16481 (Nigel Cooper, PI) is gratefully acknowledged for providing resources for the Kybrin cluster.
- Nathan Johnson (System Administrator, Dahlem Supercomputer Lab, University of Louisville), Elizabeth Cha and Tim Hardin (post graduate students at the Bioinformatics Lab, University of Louisville) for their gracious assistance in making the author familiar with the working environment and in making the conduct of the project easier.

References and Bibliography

Bibliography

[1] E. Mateu, F. Calafell, O. Lao, B. Bonne-Tamir, J. R. Kidd, A. Pakstis, K. K. Kidd, and J. Bertranpetit, "Worldwide genetic analysis of the CFTR region," *Am. J. Hum. Genet.*, vol. 68, no. 1, pp. 103-117, Jan.2001.

[2] K. Y. Chang JC, "Antenatal diagnosis of sickle cell anaemia by direct analysis of the sickle mutation," *Lancet*, 2005.

[3] M. Koenig, A. H. Beggs, M. Moyer, S. Scherpf, K. Heindrich, T. Bettecken, G. Meng, C. R. Muller, M. Lindlof, H. Kaariainen, A. Delachapelle, A. Kiuru, M. L. Savontaus, H. Gilgenkrantz, D. Recan, J. Chelly, J. C. Kaplan, A. E. Covone, N.

Archidiacono, G. Romeo, S. Liechtigallati, V. Schneider, S. Braga, H. Moser, B. T. Darras, P. Murphy, U. Francke, J. D. Chen, G. Morgan, M. Denton, C. R. Greenberg, K. Wrogemann, L. A. J. Blonden, H. M. B. Vanpaassen, G. J. B. Vanommen, and L. M. Kunkel, "The Molecular-Basis for Duchenne Versus Becker Muscular-Dystrophy - Correlation of Severity with Type of Deletion," *American Journal of Human Genetics*, vol. 45, no. 4, pp. 498-506, Oct.1989.

[4] N. Vionnet, M. Stoffel, J. Takeda, K. Yasuda, G. I. Bell, H. Zouali, S. Lesage, G. Velho, F. Iris, P. Passa, P. Froguel, and D. Cohen, "Nonsense Mutation in the Glucokinase Gene Causes Early-Onset Non-Insulin-Dependent Diabetes-Mellitus," *Nature*, vol. 356, no. 6371, pp. 721-722, Apr.1992.

[5] M. Wessman, M. Kallela, M. A. Kaunisto, P. Marttila, E. Sobel, J. Hartiala, G. Oswell, S. M. Leal, J. C. Papp, E. Hamalainen, P. Broas, G. Joslyn, I. Hovatta, T. Hiekkalinna, J. Kaprio, J. Ott, R. M. Cantor, J. A. Zwart, M. Ilmavirta, H. Havanka, M. Farkkila, L. Peltonen, and A. Palotie, "A susceptibility locus for migraine with aura, on chromosome 4q24," *American Journal of Human Genetics*, vol. 70, no. 3, pp. 652-662, Mar.2002.

[6] D. Karolchik, R. Baertsch, M. Diekhans, T. S. Furey, A. Hinrichs, Y. T. Lu, K. M. Roskin, M. Schwartz, C. W. Sugnet, D. J. Thomas, R. J. Weber, D. Haussler, and W. J. Kent, "The UCSC Genome Browser Database," *Nucleic Acids Res.*, vol. 31, no. 1, pp. 51-54, Jan.2003.

[7] M. Wirtenberger, K. Hemminki, B. Chen, and B. Burwinkel, "SNP microarray analysis for genome-wide detection of crossover regions," *Hum. Genet.*, June2005.

[8] Ann-Christine Syvänen, "Toward genome-wide SNP genotyping,", 37 ed 2005, p. S5-S10.

[9] Liu S, Li Y, and et al, "Analysis of the factors affecting the accuracy of detection for single base alterations by oligonucleotide microarray,", 37 ed 2005, pp. 71-77.

[10] R. J. Roberts, "PubMed Central: The GenBank of the published literature," *Proc. Natl. Acad. Sci. U. S. A*, vol. 98, no. 2, pp. 381-382, Jan.2001.

[11] "PubMed Central,".

[12] S. J. Tebbutt, I. V. Opushnyev, B. W. Tripp, A. M. Kassamali, W. L. Alexander, and M. I. Andersen, "SNP Chart: an integrated platform for visualization and interpretation of microarray genotyping data," *Bioinformatics.*, vol. 21, no. 1, pp. 124-127, Jan.2005.

[13] Pérez-Encisoa M, "In silico study of transcriptome genetic variation in outbred populations,", 166 ed 2004, p. 554.

[14] Itoshi Nikaido and et al, "EICO (Expression-based Imprint Candidate Organizer): finding disease-related imprinted genes,", 32 ed 2004, p. D548-D551.

[15] W. G. W. M. E. W. M. a. D. J. L. Stephen F.Altschul, "Basic Local Alignment Search Tool," 2005, pp. 403-410.

[16] Z. Ning, A. J. Cox, and J. C. Mullikin, "SSAHA: a fast search method for large DNA databases," *Genome Res.*, vol. 11, no. 10, pp. 1725-1729, Oct.2001.

[17] I. Korf and W. Gish, "MPBLAST : improved BLAST performance with multiplexed queries," *Bioinformatics.*, vol. 16, no. 11, pp. 1052-1053, Nov.2000.

[18] [Anon], "Iupac-Iub Commission on Biochemical Nomenclature (Cbn) - Abbreviations and Symbols for Nucleic Acids, Polynucleotides and Their Constituents," *Virology*, vol. 45, no. 1, p. 326-&, 1971.

Links:

1. Affymetrix Inc.: http://www.affymetrix.com
2. dbSnp Homepage: http://www.ncbi.nlm.nih.gov/projects/SNP/
3. NCBI Homepage: http://www.ncbi.nlm.nih.gov/
4. PubMed Central: http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Pmc

5. W. Gish, "WUBLAST," 2005. BLASTN 2.0MP-WashU http://blast.wustl.edu/

6.HG-U133A microarray:
http://www.affymetrix.com/support/technical/technotes/hgu133_p2_technote.pdf

7. Match/Mismatch:
http://www.affymetrix.com/support/technical/technotes/25mer_technote.pdf
8. FASTA format: http://www.ncbi.nlm.nih.gov/BLAST/fasta.shtml
9.dbSNP Download page:
http://www.ncbi.nlm.nih.gov/About/outreach/gettingstarted/snpftp/index.html
http://www.ncbi.nlm.nih.gov/About/outreach/gettingstarted/snpftp/human_rsfasta.html
10. dbSNP Summary: http://www.ncbi.nlm.nih.gov/projects/SNP/snp_summary.cgi
11. Kybrin Bioinformatics cluster: http://kybrin.spd.louisville.edu
12. BLAST comparison: http://blast.wustl.edu/blast/cparms.html
13 Sanger Institute: http://www.sanger.ac.uk/
14 Perl Homepage: http://www.perl.org
15 MySQL Homepage: http://www.mysql.com
16 Active State & ActivePerl 5.6: http://www.activestate.com/
17 MySQL program source: http://www.mysql.com/products/
18 WUBLAST parameters: http://blast.wustl.edu/blast/parameters.html
19 Bplite: http://homepage.mac.com/iankorf/BPlite.pm
20 Agilent Technologies: http://www.agilent.com