DNA Hash Pooling and its Applications

Dennis Shasha
Courant Institute of Mathematical Sciences
New York University
shasha@cs.nyu.edu
Martyn Amos
Department of Computing and Mathematics
Manchester Metropolitan University
M.Amos@mmu.ac.uk

May 24, 2007

Abstract

In this paper we describe a new technique for the characterisation of populations of DNA strands. Such tools are vital to the study of ecological systems, at both the micro (e.g., individual humans) and macro (e.g., lakes) scales. Existing methods make extensive use of DNA sequencing and cloning, which can prove costly and time consuming. The overall objective is to address questions such as: (i) (Genome detection) Is a known genome sequence present at least in part in an environmental sample? (ii) (Sequence query) Is a specific fragment sequence present in a sample? (iii) (Similarity Discovery) How similar in terms of sequence content are two unsequenced samples?

We propose a method involving multiple filtering criteria that result in "pools" of DNA of high or very high purity. Because our method is similar in spirit to hashing in computer science, we call the method *DNA hash pooling*. To illustrate this method, we describe examples using pairs of restriction enzymes. The *in silico* empirical results we present reflect a sensitivity to experimental error. The method requires minimal DNA sequencing and, when sequencing is required, little or no cloning.

1 Introduction

Biologists often examine large and diverse populations of organisms (for example, molecules, microbes or plants). This is particularly the case in fields such as microbial ecology, which studies the interactions between living microorganisms (such as algae, or bacteria) and their environment. One of the most significant and challenging problems in these areas of biology is to quantify the overall *diversity* of a given population. This task is often made even more difficult by the fact that many "wild" organisms resist laboratory cultivation (and, thus, have unknown phenotypes and their genomes are unknown), or may be present in a population in relatively low numbers.

The study of *metagenomics* has emerged in recent years [8, 10, 16, 18] to perform what has been described as "environmental forensics," including the

quantification of relative abundances of known species, and the estimation of the number of "unknown" species in a given environment [10]. The potential impact of this new field is huge, with applications ranging from medicine to agriculture and biotechnology. Further the insights gained will be of significant assistance in furthering our understanding of biodiversity in both new and familiar environments, such as frozen Antarctic lakes and the human gut [8].

Metagenomic analysis currently involves the extraction of DNA from an environmental sample, cloning of the DNA into a suitable "vector", insertion of the vector into a host bacterium and then screening the resulting transformed bacteria [8]. Screening may occur on the basis of gene expression using microarrays [18] or some other trait, such as antibiotic production [15], or the bacteria may simply be sequenced at random. We now briefly describe the first (microarray) and third (sequence-based) methods.

1.1 Microarrays

Microarrays [14] permit the study of gene expression and the detection of mutations. "Similar to the situation in which microprocessors have increased the speed of computation, microarray-based genomic technologies have revolutionized genetic analysis of biological systems" [18]. However, they tend to be used for relatively pure, or homogenous samples, and their applicability at the community level is less well-understood. In order to identify unknown bacterial strains, the microarray requires a "probe" taken from a related strain, which may or may not be available. In addition, the cost of microarray equipment is, in the short to medium term, high for non-trivial studies.

1.2 Sequencing

Sequence-based approaches, on the other hand are guided by existing genomic knowledge-bases, and we briefly describe two of these here.

1.2.1 Environmental Gene Tags

The assembly of genomes from complex communities currently "demands enormous sequencing expenditure for the assembly of even the most predominant members" [16]. Because of this difficulty, borne out by initial studies by Tringe *et al.* [16], the authors decided to employ an alternative, "gene-centric" approach that does not attempt to attribute genes obtained to any particular genome. They obtained their initial dataset by taking four sets of samples, one from agricultural soil, and three from whale carcasses. Samples were then partitioned into bacteria, archea or eukaryotes using PCR-amplified rRNA libraries. Genomic small-insert libraries were then sequenced from each sample (100 million base pairs from the soil and 25 million base pairs from each whale sample). These sequences, derived from different population members, were termed "Environmental Gene Tags" (EGTs), since they may encode regions of functional genes that are necessary for survival in a particular environment. Different environment types will exhibit unique EGT "fingerprints", containing genes derived from many different genomes. The study showed that two whale carcasses, located 8000km apart, nontheless had very similar EGT patterns. Thus, one may determine the type of environment from this fingerprinting technique.

1.2.2 MEGAN

In a recent study, Huson *et al* present an approach [10] to the problem of genomic assembly in which the authors compare sequenced data to existing databases. Specifically, the set of DNA sequences obtained by random shotgun sequencing from the environmental sample is run against known sequences using BLAST. The resulting meta-data is then provided as input to the MEGAN package, which estimates and explores the taxonomical content of the data set. This may be a good technique to obtain the most abundant species in a sample, but will have difficulty locating rare sequences of interest. One of the themes of our approach is to hunt systematically for signs of a genome of interest.

1.3 Implications

The existing approaches require significant sequencing effort. In 2007, DNA sequencing costs approximately 1 (U.S.) cent per base [5]. Even for a relatively simple community study on the drainage region of an acid mine, roughly 15 million bases were sequenced in order to obtain the required metagenomic data [16], at a cost (today) of approximately $150,000. A soil study, requiring at least 50 million bases, might then cost half a million US dollars.

Fortunately, sequencing is not always necessary as a first step. Molecular techniques that work at the whole sequence level may be used to reduce the initial complexity of a sample population. One tool commonly employed is "GC fractionation" [9], which works along the principles of a molecular "sieve", sorting strands according to their relative GC content (guanine and cytosine being heavier than their counterparts adenine and thymine). This may be effective when trying to partition a sample into *eukaryotic* and bacterial sets, since eukaryotic DNA tends to have a much lower GC content (e.g, we selected two complete bacterial genome sequences, A (*Escherichia coli* K12) and B (*Shigella boydii* Sb227) for early studies; each of these had a GC content of roughly 51%, while the human genome is made up of around 45% GC and that of the mouse roughly 44%). However, such a relatively crude tool rapidly proves ineffective when dealing with shorter sequences, where we may only possess genomic fragments within our sample. For bacterial sequence A, when taking 200 random consecutive sequences of length 50,000, we obtained a GC content ranging from 46.7% to 53.3% with the 90% confidence interval ranging from 47.3% to 52.8%.

Preliminary work on estimating the complexity of a heterogenous population of DNA strands (without using sequencing) is reported in [7]. This paper, motivated in part by the authors' earlier work on DNA-based computing [1], reports initial experimental investigations into the use of basic laboratory methods (combined with probability theory) to estimate the complexity of a tube of strands. Faulhammer *et al.* digested their initial tube with a set of restriction enzymes with recognition sites differing in sequence and of length $4 \leq k \leq 8$. The contents of the tube were then visualised in a gel, and the *number* of distinct bands observed used to obtain an estimate of the number of different strands.

Our proposal uses some of the same basic laboratory methods, but it differs from that of [7] in several important ways:

1. Rather than simply *counting* the number of different restriction fragments obtained, we use the lengths *themselves* to obtain a partition of segments.

2. We use the resulting data structure both *in silico* and *in vitro* to compare different DNA "tubes".

3. We use multiple levels of restriction enzyme digestion.

This basic approach (the use of restriction enzymes to digest a population sample, followed by analysis of the fragment size) also underpins an early variant of the well-known technique of *DNA fingerprinting* [11]. *Restriction fragment length polymorphisms* (RFLPs) [6] provide a technique by which organisms may be differentiated by comparing the patterns obtained by digesting a certain portion of their DNA. If two organisms differ in the distance between restriction sites, the length of the fragments produced will differ (i.e., be polymorphic) when the DNA is digested. However, this method is generally only useful when the population sample is relatively homogeneous (e.g., one wishes to distinguish between members of the same species).

In the rest of the paper, we present and evaluate a simple and powerful technique called *DNA hash pooling*. We conclude with a discussion of plans for future theoretical and experimental work.

2 DNA Hash Pooling

In computer science, *hashing* [12] maps a relatively small set from a large domain (e.g., 10,000 integers ranging in value from 0 to one billion) to a small domain (e.g,. the set of integers from 1 to 5000) through a mathematical *hash function*. Applications of hashing include cryptography, error correction, authentication and identification. A typical hash function is *modulus* (i.e., remainder). For example, 7 mod 5 = 2 because 2 is the remainder after dividing 7 by 5. For the same reason, 28 mod 5 = 3. A hash data structure based on "mod 5" will map 28 to bucket (or *pool*) 3, 7 to pool 2, 12 to pool 2, 59 to pool 4, and so on. There are many variants of hashing, some of which entail hashing each pool resulting from the first hash function in order to get "purer" pools, and then using the combined hash results to generate an item "label". For example, using a second hash function, based on "mod 7", 28 would map to 0 and 53 to 5. Thus the full "label" of 28 would be (3, 0) because 28 mod 5 = 3 and 28 mod 7 = 0. By contrast, the label of 53 would be (3, 4) because 53 mod 5 = 3 and 53 mod 7 is 4. Associated with each unique label is a pool having a relatively small number of distinct values.

DNA hash pooling or *hash pooling* for short is the analogous operation on DNA. The "hash functions" in this scenario correspond to biological operations that give rise to distinctive and quantifiable "fingerprints" (e.g., measurement of GC content followed by digestion by a set of restriction enzymes). The label components correspond to the "values" obtained by application of the hash functions (e.g. GC content and fragment length).

In silico, our method involves simulating these operations on known sequences (typically though not necessarily of entire genomes) and characterizing different portions of those sequences from the result(s). *In vitro*, our method involves performing the bench-based operations and sequencing only those pools that are likely to be pure (this is, unique to one genome), or otherwise of interest.

For concreteness, this paper focusses on hash pooling based solely on *restriction enzymes*. The basic operations are

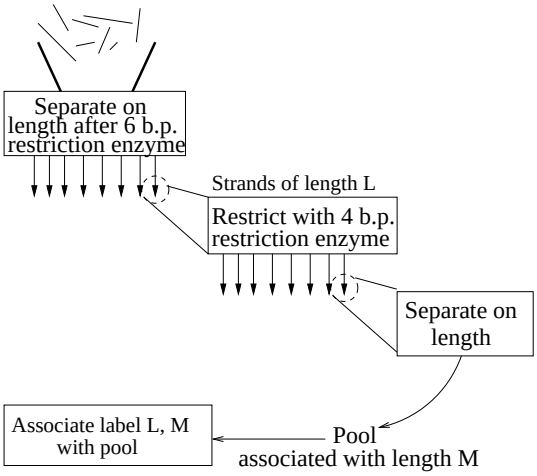

Figure 1: Two-stage hash pooling

1. Apply a six base-pair (base pair) restriction enzyme to a sequence, yielding a set of fragments.

2. Partition those fragments based on length (perhaps approximately), using a technique such as gel electrophoresis.

3. Apply a four base pair restriction enzyme to a selected subset of partitions and separate on length again.

4. Sequence selected lengths.

Each resulting pool is therefore associated with a label consisting of two lengths, the first based on a six base pair restriction enzyme and the second based on a four base pair restriction enzyme (Figure 1).

The procedure may be described for K stages in pseudo-code as follows:

```
hash(stage j, sample s, label L, K)
  r: = restriction enzyme for stage j
  frags := apply r to s
  mysamps := partition frags on length
  for each t in mysamps
       tlabel := L concat (r, length(t))
       if (j < K)
            hash(j+1, t, tlabel, K)
       else (t, tlabel) is a member of the final pool
  end for
```

```
Pseudo-code for K stage hash pooling based on restriction enzymes.
The initial call on an initial  sequence orig would be
hash(1, orig, null, K), where null is the empty label.
```

For example, consider the genomic sequence of bacterium A (*E. coli* K12). If we cut A using the enzyme SmaI (recognising CCCGGG), take the pool corresponding to length 264, cut that pool with RsaI (GTAC) and take the pool of length 31, we get a pool having label (264, 31). It happens to have a single member with the sequence CTATCCGCTCAATGAGTCGGTCGCCATTGCC. By contrast, the pool with label (770, 207) has three different sequences. For some applications, we will want pools having singletons (i.e., a set with only a single element) in order to obtain a pure sequence without the need for cloning.

One may object that separating fragments by length entails a certain inaccuracy imposed by the laboratory technique; a reasonable estimate of this error may be plus or minus 10 base pairs[3]. In this case, in order to obtain a pure sample, we may be interested in finding a pool whose label has no "10 base pair-neighbors." The labels L and L' are 10 base pair-neighbors if (i) the first component of L and the first component of L' are different but differ by 10 or less ($0 <| L[0] - L'[0] |\leq 10$); or (ii) the first component of L and L' are the same but the second components differ by 10 or less ($0 <| L[1] - L'[1] |\leq 10$). For *E. coli* K12, the labels (188, 59) and (188, 106), for example, have no 10 base pair-neighbors.

3 Experiments

Having presented our formal framework, we can now present several applications and our *in silico* empirical results.

3.1 Genome Detection

The first question we ask is the following: given a tube, T, of unknown DNA (perhaps from an environmental sample) and a genome whose sequence is known, are "reasonably sized" portions of that genome present in T, even if in small concentrations? (Figure 2)

A "reasonably sized" portion is a sequence of length at least 200,000 base pairs (or roughly 5% of the length of a bacterial genome.) This might be used for the detection of bacterial pathogens in food, for example. In what follows, we

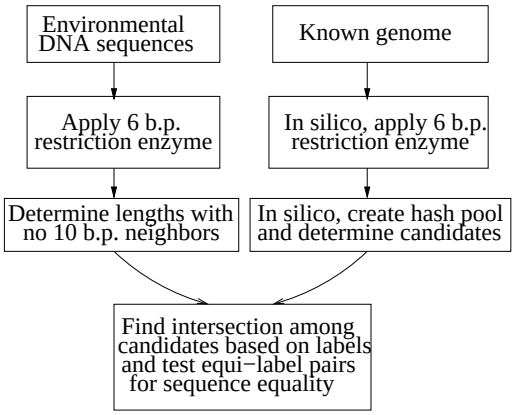

Figure 2: Genome sequence detection

used bacterium A, *E. coli* K12, which is often used as an indicator organism in the detection of faecal contamination. Our *in silico* result involved the following steps:

1. Compute the *candidate set* of A, consisting of possibly non-singleton pools having no 10 base pair neighbors. There were 3,567 candidates. This gives us a "comparison library" of pools. It is important to note that this step is purely computational and can be computed just once for any combination of known genome and restriction enzyme set.

2. We simulate the unknown sample T by taking a 200,000 consecutive base pair subsequence of A (with the start position taken uniformly at random) and combining it with a sequence of length four times that of A (generated pseudo-randomly to have the same GC content as A).

3. We then compute the resulting candidate set of hash pools, based on no 10 base pair neighbors.

Having tried this 20 times, we found, on average, 2,000 pools in the second candidate set. On average the two sets had an intersection of cardinality 71 based on their labels. When labels were equal, 99.8% of the time there was a match of the sequence *and* the sequence came from that 200,000 consecutive base pair subsequence, giving a precision of 99.8%. This implies that one may be able to avoid sequencing if one finds enough common candidates. Further the recall was 100% in that we always found a matching label.

When applying this in a laboratory setting, there is the significant question of whether this operation many separate DNA extractions and applications of a restriction enzyme. Fortunately, the answer is no. For each of the 20 tests, first the six base pair restriction enzyme was used. This gave a collection of fragment lengths. On average only 5.8 of those lengths had no 10 base pair neighbors and had lengths similar to the lengths of the candidates from A. [1] So on average only 5.8 fragment lengths required extraction. Of those, 4.7 (on average) yielded matching sequences. So, if this were done *in vitro*, approximately

[1] Typical lengths were between 7,000 base pairs and 39,000 base pairs

70 common strands would be found using one application of SmaI and under six applications of RsaI. Virtually all (99.8tested strands would be shown to be equal, so sequencing would not even be necessary.

This experiment shows that *in silico* hash pooling on a known genome can identify pools to look for in a sample, such that those pools have a strong likelihood of containing a subsequence of the known genome. Thus, we can see this method as an improvement over random sampling, and can be used even if the bacterium of interest is relatively rare in the sample.

3.2 Sequence Query

Here we address the question: given a query sequence, is that sequence present, at least *in part*, in the tube? This might be used to look for the presence of a pathogen, for example. This question is clearly related to the previous one. In fact, an experiment similar to that used to address the first question serves as an illustration: suppose the sample under scrutiny contains A plus a lot of other assorted DNA (e.g. the full genome of A amongst a pseudo-random sequence four times the length of the A sequence and having the same GC content). Then, 20 times, we take a random query subsequence of length 200,000 from A and see if we can find matching parts in the sample tube.

The sample tube (A sequence plus a *random* sequence four times A in length with no 10 base pair neighbors) has 3,516 candidate pools. The average 200,000 base pair subsequence of A has about 200 candidates. In our 20 experiments, whenever two labels are equal, the corresponding sequences matched 100% of the time (precision of sequence matching given label match of 100%). This is not guaranteed to hold always of course, but again shows that even without sequencing one can be quite sure that sequences will match if labels match. Recall was not as good, as we found matching labels in 17/20 or 85

As in the first experiment, the six base pair restriction enzyme would cut the fragments into certain lengths, but, on the average, only 2.3 of those lengths (ranging from 10,000 base pairs to 30,000 base pairs) would have the properties that (i) they had no 10 base pair neighbors and (ii) they matched the candidates from the 200,000 base pair query sequence. Thus, on the average, under three extractions need to be taken and then digested by the four base pair restriction enzyme.

We also tried the negative case when the query sequence was nowhere present in the sample. In that case, on the average, after cutting with the six base pair restriction enzyme, on the average, under one of those lengths had the properties that (i) they had no 10 base pair neighbors and (ii) they matched the candidates from the 200,000 base pair query sequence. When extracted and digested by the four base pair restriction enzyme, there were no matching labels (other than a single label whose final fragment length was only 4). So this technique does not throw up false positives.

3.3 Similarity Discovery

Here we consider the problem: given two tubes of DNA, do they contain strands that are the same or very similar? This might be useful when comparing samples of unsequenced genomes. In this case, we cannot compute candidate pools that have no 10 base pair neighbors using known genomes. Instead, we have to

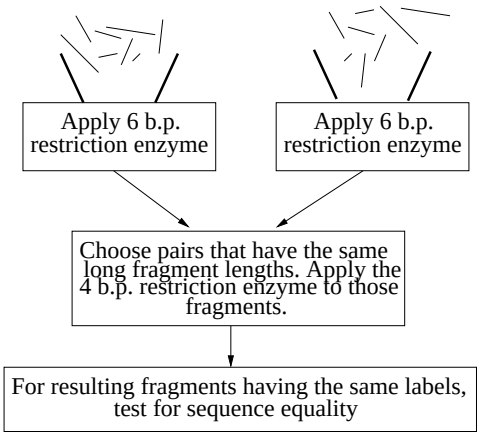

Figure 3: Sample comparison

measure them. Sometimes we may not know whether the sample contains known sequences. If it does, we can use the techniques in the Subsequence Detection subsection above to find out which known genomes each sample contains and then see which are the same.

Let us assume however that the sample contains no known genomes (or that we want to detect commonalities besides those among known genomes). Our strategy will be to choose the most *likely* pairs to study by focussing on "unusual" labels (Figure 3). We therefore performed the following *in silico* experiment:

1. Take a 200,000 base pair sequence, *target*, with the same GC content as A, plus a random sequence four times the size of A ($4 \times 4.7Mb =\approx 20Mb$), with the same average GC content as A.

2. For the second sample, we use the same 200,000 base pair sequence *target* plus *another* random sequence four times the size of A, with the same average GC content. Thus the target in each sample is 200,000 base pairs long, just 1% of the roughly roughly 20 million for the entire sequence present in each tube.

Now the question is this: in which pools should we look for common strands? That is, is it better to look at pools where the six base pair restriction enzyme has cut strands of length approximately 4,000 (the expected value) or much longer? The *in silico* answer is obvious in retrospect: to find common strands, the best pools to look at are ones corresponding to *long* lengths when cut by the first restriction enzyme. Thus the procedure is this:

1. Cut each sample with the six base pair restriction enzyme, then find all lengths that are the same (within an accuracy of 10 base pairs).

2. On the upper quartile of those lengths (approximately 235 of them), apply the four base pair restriction enzyme.

Of those fragments that have the same lengths (within 10 base pairs) for both the first and second restriction enzymes, between 4% and 7% are the same

sequence over the 20 experiments that we tried. Other quartiles are about a factor of 10 less good. On the other hand, if we look at dectiles (1/10), then the upper 1/10 of the lengths (from 10,210 to 24,550) gives a hit ratio of about 14%, and only 94 lengths from the six base pair restriction enzyme require an application of the second restriction enzyme.

If we have already identified known genomes that the two tubes share in common, then we should avoid labels that correspond to those.

4 Implementation Issues

In this section we give a brief overview of the two main laboratory tools from which our basic operations are built: restriction enzymes (for chopping DNA into sections) and gel electrophoresis (for sorting fragments according to length). The descriptions are taken from [2].

4.1 Restriction Enzymes

Restriction endonucleases [17] (often referred to as *restriction enzymes*) recognize a specific sequence of DNA known as a *restriction site*. Any DNA that contains the restriction site within its sequence is cut by the enzyme at that point.

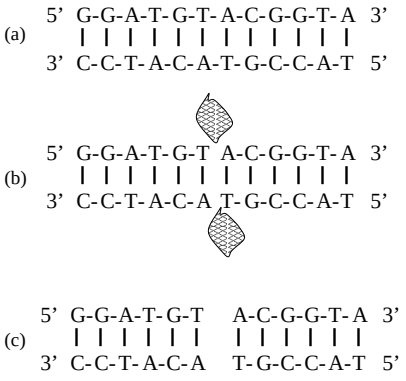

Figure 4: (a)Double-stranded DNA. (b) DNA being cut by *Rsa*AI. (c) The resulting blunt ends

For example, the double-stranded DNA in Fig. 4a is cut by restriction enzyme *Rsa*I, which recognizes the restriction site $GTAC$. The enzyme breaks (or "cleaves") the DNA in the middle of the restriction site (Fig. 4b). The exact nature of the break produced by a restriction enzyme is of great importance. Some enzymes like *Rsa*I (mentioned earlier) leave "blunt" ended DNA (Fig. 4c).

4.2 Gel Electrophoresis

Gel electrophoresis is an important technique for sorting DNA strands by size [4]. Electrophoresis is the movement of charged molecules in an electric field. Since DNA molecules carry a negative charge, when placed in an electric field they tend to migrate toward the positive pole. The rate of migration of a molecule

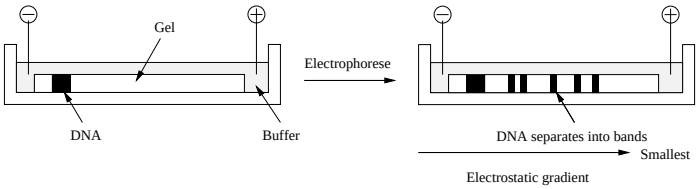

Figure 5: Gel electrophoresis process

in an *aqueous* solution depends on its shape and electric charge. Since DNA molecules have the same charge per unit length, they all migrate at the same speed in an aqueous solution. However, if electrophoresis is carried out in a *gel* (usually made of agarose, polyacrylamide, or a combination of the two), the migration rate of a molecule is also affected by its *size*.[2] This is due to the fact that the gel is a dense network of pores through which the molecules must travel. Smaller molecules therefore migrate faster through the gel, thus sorting them according to size.

A simplified representation of gel electrophoresis is depicted in Fig. 5. The DNA is placed in a well cut out of the gel, and a charge applied.

Once the gel has been run (usually overnight), it is necessary to visualize the results. This is achieved by staining the DNA with the fluorescent dye ethidium bromide and then viewing the gel under ultraviolet light. At this stage the gel is usually photographed.

One such photograph is depicted in Fig. 6. Gels are interpreted as follows; each *lane* (1–7 in our example) corresponds to one particular sample of DNA (we use the term *tube* in our abstract model). We can therefore run several tubes on the same gel for the purposes of comparison. Lane 7 is known as the *marker lane*; this contains various DNA fragments of known length, for the purpose of calibration. DNA fragments of the same length cluster to form visible horizontal *bands*, the longest fragments forming bands at the top of the picture, and the shortest ones at the bottom. The brightness of a particular band depends on the amount of DNA of the corresponding length present in the sample. Larger concentrations of DNA absorb more dye, and therefore appear brighter. One advantage of this technique is its sensitivity – as little as 0.05 μg of DNA in one band can be detected as visible fluorescence.

The size of fragments at various bands is shown to the right of the marker lane, and is measured in *base pairs*. In the photograph, the largest band resolvable by the gel is 2,036 base pairs long, and the shortest one is 134 base pairs long. Moving right to left (tracks 6–1) is a series of PCR reactions which were set up with progressively diluted target DNA (134 base pairs) to establish the sensitivity of a reaction. The dilution of each tube is evident from the fading of the bands, which eventually disappears in lane 1.

[2]Migration rate of a strand is inversely proportional to the logarithm of its molecular weight [13].

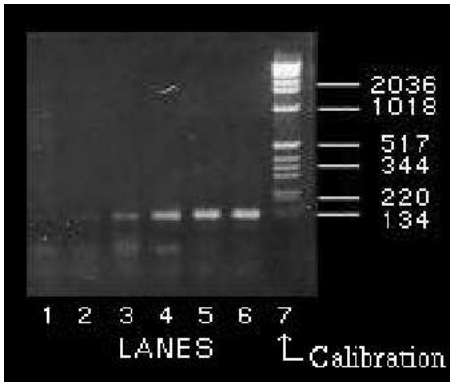

Figure 6: Gelelectrophoresis photograph

5 Conclusions

DNA Hash pooling is a method to simplify many problems in metagenomics. It gives the experimenter the ability to query for known sequences and genomes in a sample or to find common sequences from unknown genomes in two or more samples *even if the identified sequences are rare*. The version of the technique described in this paper involves a small number of steps of the form: extract DNA of a certain length, apply a restriction enzyme to it, and measure the lengths of the results. In most cases, sequencing is unnecessary and, where it is, cloning is not. The main technical challenge is to get reasonably accurate measurements of length.

The main future work we anticipate is to validate the technique and then extend the method as new application scenarios present themselves.

6 Acknowledgments

Shasha's work has been partly supported by the U.S. National Science Foundation under grants IIS-0414763, DBI-0445666, N2010 IOB-0519985, N2010 DBI-0519984, DBI-0421604, and MCB-0209754. This support is greatly appreciated.

References

[1] Leonard M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266:1021–1024, 1994.

[2] Martyn Amos. *Theoretical and Experimental DNA Computation*. Springer, 2005.

[3] Annelise E. Barron, Harvey W. Blanch, and David S. Soane. A transient entanglement coupling mechanism for dna separation by capillary electrophoresis in ultradilute polymer solutions. *Electrophoresis*, 15:597–615, 1994.

[4] T.A. Brown. *Genetics: A Molecular Approach*. Chapman and Hall, 1993.

[5] Francis S. Collins and Anna D. Barker. Mapping the cancer genome. *Scientific American*, February 2007.

[6] T.E. Dowling, C. Moritz, and J.D. Palmer. Nucleic acids II: restriction site analysis. In D.M. Hillis and C. Moritz, editors, *Molecular systematics*, pages 250–317. Sinauer Associates, 1990.

[7] Dirk Faulhammer, Richard J. Lipton, and Laura F. Landweber. Counting DNA: estimating the complexity of a test tube of DNA. *BioSystems*, 52(1–3):193–196, 1999.

[8] Jo Handelsman. Metagenomics: Application of genomics to uncultured microorganisms. *Microbiology and Molecular Biology Reviews*, 68(4):669–685, 2004.

[9] W.E. Holben and D. Harris. DNA-based monitoring of total bacterial community structure in environmental samples. *Molecular Ecology*, 4(5):27–31, 1995.

[10] Daniel H. Huson, Alexander F. Auch, Ji Qi, and Stephan C. Schuster. MEGAN analysis of metagenomic data. *Genome Research*, 17:377–386, 2007.

[11] A.J. Jeffreys, V. Wilson, and S.L. Thein. Hypervariable 'minisatellite' regions in human DNA. *Nature*, 314:67–73, 1985.

[12] Donald Knuth. *The Art of Computer Programming, Volume 3: Sorting and Searching*. Addison-Wesley, second edition, 1998.

[13] R. Old and S. Primrose. *Principles of Gene Manipulation, an Introduction to Genetic Engineering*. Blackwell, fifth edition, 1994.

[14] M. Schena, D. Shalon, R.W. Davis, and P.O. Brown. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*, 270(5235):467–470, 1995.

[15] P.D. Schloss and J. Handelsman. Biotechnological prospects from metagenomics. *Current Opinion in Biotechnology*, 14:303–310, 2003.

[16] Susannah Green Tringe et al. Comparative metagenomics of microbial communities. *Science*, 308:554–557, 2005.

[17] J. Williams, A. Ceccarelli, and N. Spurr. *Genetic Engineering*. βios Scientific Publishers, 1993.

[18] Jizhong Zhou. Microarrays for bacterial detection and microbial community analysis. *Current Opinion in Microbiology*, 6:288–294, 2003.