Fluctuation and Noise Letters
Vol. 5, No. 4 (2005) R13–R35

World Scientific
www.worldscientific.com

REVIEW OF SIGNAL PROCESSING IN GENETICS

M. J. BERRYMAN* and A. ALLISON†
Centre for Biomedical Engineering (CBME) and
School of Electrical & Electronic Engineering
The University of Adelaide, SA 5005, Australia
**mattjb@eleceng.adelaide.edu.au*
†*aallison@eleceng.adelaide.edu.au*

C. R. WILKINSON
Microarray Analysis Group
School of Mathematical Sciences
The University of Adelaide, SA 5005, Australia
and
Child Health Research Institute,
72 King William Road, North Adelaide, SA 5006, Australia
christopher.wilkinson@adelaide.edu.au

D. ABBOTT
Centre for Biomedical Engineering (CBME) and
School of Electrical & Electronic Engineering
The University of Adelaide, SA 5005, Australia
dabbott@eleceng.adelaide.edu.au

Received 7 July 2005
Revised 25 September 2005
Accepted 7 October 2005
Communicated by Shura Neiman

This paper reviews applications of signal processing techniques to a number of areas in the field of genetics. We focus on techniques for analyzing DNA sequences, and briefly discuss applications of signal processing to DNA sequencing, and other related areas in genetics that can provide biologically significant information to assist with sequence analysis.

Keywords: DNA sequences; time series analysis; hidden Markov models; mutual information; spectral analysis; autocorrelation.

1. Introduction

Genetics is concerned with the physical characteristics of organisms that are passed on from one organism to another through the use of deoxyribonucleic acid (DNA), consisting of a sequence of nucleotides. The nucleotides are the chemical bases

adenosine, thymine, cytosine and guanine that are denoted using the alphabet $\{A, T, C, G\}$. Those on one strand are paired in a complementary fashion with those on the other strand, where adenosine matches with thymine, and guanine with cytosine. Groups of three bases are called codons, and these encode the twenty amino acids that combine to form proteins, the building blocks of life. In a nutshell, the central dogma of molecular biology states that "DNA makes RNA makes protein". This is encapsulated in Fig. 1. The DNA is transcribed into complementary messenger ribonucleic acid (mRNA). In RNAs, the alphabet is $\{A, T, U, G\}$ where uracil plays the same role that thymine does in DNA, as it pairs with guanine. Sections of the mRNA that do not code for proteins are removed, and a "poly-A tail"—a sequence composed entirely of adenosine bases—is added to (chemically) stabilise the sequence. The mRNA then acts as a template for protein synthesis. Transfer RNAs (tRNAs) bind to an amino acid on one end, and a complimentary set of three bases on the mRNA template. A 1D sequence of amino acids forms and is then detached from the tRNAs and folds into a 3D structure. This sometimes occurs by itself and sometimes with the aid of other proteins, either immediately or at a later date in the life of the cell. DNA that binds to an mRNA sequence is complimentary to this sequence and is explicitly called cDNA. This principle is used in microarray technologies as described later.

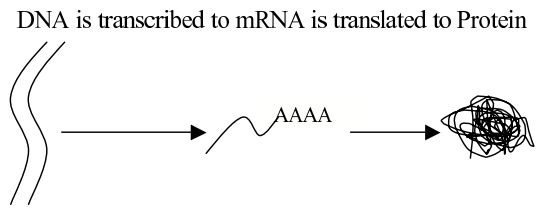


Fig. 1. The central dogma of molecular biology states that "DNA is transcribed into messenger RNA, which is then translated into protein." This diagram also shows DNA replication, which is done with the aid of a number of proteins. At the mRNA stage, introns are spliced out from the sequence, leaving only the protein coding exons. This dogma is of course vastly simplified, for example there is added complexity through splicing, RNA-only genes, RNA-RNA interactions, prions, and other details [1, 2]. But in its essential form this does describe the flow of information in a cell.

Not all regions of DNA code for proteins—some of these non-protein-coding regions have known functions, such as the *Xist* gene [3], which codes for an ribonucleic acid (or RNA) molecule that deactivates one of the two X chromosomes in female mammals. These RNAs may play an important role in the complexity of organisms such as humans [4]. There are also promoter regions around genes that act as targets for gene activation or deactivation [5]. Other non-coding regions appear to only be "junk" DNA left over from the biological past, with little or no use—or perhaps have a yet undiscovered function. Biologists have suggested that "junk" regions may act as a form of isolation between coding regions and may also act as error-robust locations for sexual recombination—this is described further in Harmer *et al.* [6], where it is conjectured that these effects could be modeled in game-theoretic terms.

Signal processing is the use of mathematical techniques to analyze any data

signal. This data could be an image, a sound, or any other sequence of data, such a sequence of nucleotides. The sequences of interest could be protein coding regions, repeating elements that may be associated with various diseases (such as Huntington's disease[7]) or regions rich in some set of complementary bases, such as A and T, which can give information on evolutionary history including lateral gene transfer in bacteria [8].

An area where signal processing techniques have enjoyed wide usage is in microarray processing [9]. In microarray analysis, effects on gene expression (as ascertained through mRNA levels) can be tested, for example the effect of a drug. Two-color microarrays are a colored grid of spots (typically one color for the control, the other for the cells under test) with spot intensity and color showing the expression levels for the gene associated with that spot. Affymetrix microarrays only consider one gene and a gene control in a paired-spot arrangement. The control spot controls for non-specific hybridization and background signals. The use of only one flourescent dye removes bias caused by differences in fluorescent dye tagging.

The analysis of the sequences produced has come under intense focus as an area where signal processing could be used to solve a number of important problems such as the nature of non-coding DNA and distinguishing coding DNA from non-coding DNA. Methods such as the discrete Fourier transform [10, 11] and multifractal analysis [12] have been applied to the problem, complementing more traditional techniques that often use hidden Markov models [13, 14]; these are detailed later. A good overview of Fourier transform methods and wavelet transforms, not discussed in this paper, and a more in-depth discussion of cellular neural networks can be found in Zhang *et al.* [15]. Here we focus on other applications of Fourier methods, and also explore the use of hidden Markov models and other mathematics to general problems in genetics.

Signal processing is not just a human enterprise—even individual cells process signals in the form of mRNA, protein, and more general chemical levels (for example sugars in the environment) [16, 17, 18, 19]. As with conventional computers, cells can be genetically programmed to process signals [20, 21, 22]. As in electrical circuits, switching elements can be built in, and positive and negative feedback loops are present, enabling a range of behaviours to be "programmed", such as chemical oscillations of a predetermined frequency. Such engineered "gene circuits" could have important applications in gene therapies where we wish to modify the existing protein and cellular interactions in an organism.

2. Sequence Analysis

Once a DNA sequence has been obtained, one can then ask questions about the DNA sequence by carrying out biological analysis *in silico*.[a] Some of the characteristics of interest that can be determined about the sequence are:

1. where the genes are located [23]
2. prediction of the three dimensional protein structures [24]

[a] *in silico* refers to a biological "experiment" done in computer simulation.

3. the relationships between genes in different organisms [25, 26]
4. searching sequences for genes related to known ones [27]
5. examining lateral gene transfer (where genes are transfered between existing species) [8]
6. correlations between regions of DNA [28].

Current techniques mainly use statistical and probabalistic techniques, especially hidden Markov models [13]. Recently, others have considered applying signal processing techniques [10] and fractal techniques [12] to these problems.

2.1. *Current techniques*

Many of the existing techniques for solving problems like finding the position of genes and determining protein folding are based around hidden Markov models. Hidden Markov models are statistical models for describing events in a given state-space, and act as a mathematical profile of the sequence, capturing important details. Hidden Markov models are trained on a set of data, with some assumptions about the data built in to the algorithm. Once trained, the model can then take a new sequence and find genes in it, or determine the way the encoded protein folds, or look for similar sequences in a larger new sequence in a computationally efficient way. Details of the training of hidden Markov models are given in Appendix A. Hidden Markov models have also been combined with support vector machines for determining the final base pair in DNA hairpin sequences, which can be difficult to sequence [29].

Essentially Markov models use a state-based approach to examine sequences, with a set of probabilities giving the probability of the system changing from one state to another. For example, a simple Markov model might treat a base as a state, and determine the probability that a T occurs after a G in the sequence. A *hidden* Markov model considers sets of states that are not directly observable in a sequence, for example the *GeneMark.hmm* software [14] has separate sets of states for coding and non-coding regions of DNA, so an A in a coding region is a different state to an A in a non-coding region. Note that the coding and non-coding regions are not observable in the sequence by itself, which makes this a hidden Markov model.

One application of hidden Markov models is in gene finding [14]. Here one considers a DNA sequence, just a long string of letters from $\{\mathrm{A}, \mathrm{T}, \mathrm{C}, \mathrm{G}\}$. Then with no information other than knowledge of the start and stop codons, one can predict genes with a missed gene rate (when the predicted genes are compared to known genes) of around 5%. Hidden Markov models have also been used to predict protein folding for the proteins encoded by known genes [30], with prediction of various structures within proteins having an accuracy around 55-70% after being trained on known protein structures. Other related statistical modelling techniques can give accuracy rates up to 77% [31].

It is often of interest to build up profiles of biological sequences (both sequences of nucleotides and sequences of amino acids), to enable comparisons of sequences between species, within species and comparisons between related sequences within an individual genome. Software is available that lets the user build a database of

profiles, which can then be used for the above mentioned purposes [32]. Using this software, for example, one can build a hidden Markov model profile of the *thrA* gene in the *Escherichia coli* strains *E. coli* K12 and *E. coli* O157:H7 EDL933 [33], and then use this to find and align the same gene in the CFT073 strain [34]. Searching and aligning can be done with other algorithms, such as the Smith-Waterman algorithm [13, 35]. Both HMMER and Smith-Waterman have a time complexity of $O\left(ln^2\right)$ and space complexity of $O\left(ln^2\right)$ in aligning l sequences of length n; the HMMER algorithm does more general sequence searching than dynamic programming algorithms such as the Smith-Waterman algorithm, identifying more loosely related sequences. It does this by taking a direct probabilistic approach to the sequences directly, rather than using probabilities of base substitutions, deletions, and insertions, and using these in a dynamic programming algorithm.

Here we show the match of part of the gene sequence found in *E. coli* CFT073. The first line gives a part of the sequence in the trained hidden Markov model. The third line, in upper case, gives part of the query sequence which matches the model; the matches of individual bases are shown in the second line along with the differences as indicated by gaps. The query sequence is usually shown in uppercase to distinguish it from the model and match sequences.

```
model     ccacctggtggcg
matches   cca ctggt gcg
query     CCATCTGGTAGCG
```

The match was found by using a hidden Markov model, which finds match states that are not directly observable in the sequences.

2.2. *Spectra and correlations*

The discrete Fourier transform (see Eq. B.7 in Appendix Appendix B), as given in Sussillo [36] (a slight variation of the work by Anastassiou [10]), is used to generate color spectrograms of DNA sequences. These enable the visual identification of regions in DNA where sequences are repeated, and what the repeat length is. Fourier transforms can also be used to find genes [37, 38]. The fast Fourier transform operates in $O\left(n \log n\right)$ time, thus this technique is faster than other algorithms for identifying genes, which typically operate in $O\left(n^2\right)$ time.

To illustrate the usefulness of the color spectrogram technique in identifying regions of DNA, Fig. 2 shows the color spectrogram for the DNA sequence of *Bacillus anthracis Ames* [39], indicating the genome is almost entirely coding, with some AT rich regions in the first half of the genome. Spectrograms also show promise in identifying coding regions and repeat sequences [36]. Another approach uses wavelet transforms to provide profiles of DNA sequences [40].

To obtain the color spectrograms, the DNA sequence is converted to sequences of numbers, as described further in Anastassiou [10], and the methods described in Appendix B are applied to this sequence. The presence of codons (length $T = 3$) shows up as a bright band, as in Fig. 2, at discrete frequency $k = N/T$ where N is the sequence length, or in digital frequency at $f_d = 1/T$.

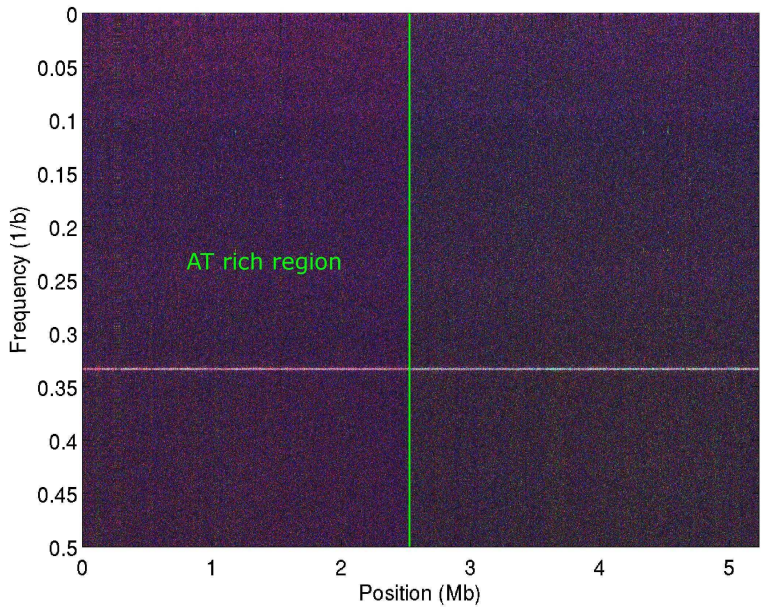


Fig. 2. Here we show the color spectrogram for *Bacillus anthracis Ames*, with each point representing the RGB components from the power in the Fourier transforms of the mappings of the sequence. The horizontal axis represents the position in the genome (in megabases), and the vertical axes in digital frequency—for genomes this has units of (1/bases). The bright band at frequency $f_d = 1/3$ highlights the coding regions which have periodicity $T = 1/f_d = 3$ bases. Since A and T map onto blue and red respectively, we expect regions that are AT rich to appear a brighter shade of purple than those that are not. Here, the spectrogram shows the first half of the genome to have a higher AT content than the second half. The transition point is marked with a bright green vertical line.

2.3. *Generalized correlation detection*

The power spectrum is related to the autocorrelation of a sequence by the Weiner-Khintchine relationship

$$P(f) = \frac{1}{N} \sum_{i=1}^{N} R_{ss}(i) e^{-j2\pi if}, \tag{1}$$

where N is the data length, $j = \sqrt{-1}$, and $R_{ss}(i)$ is the autocorrelation of the sequence for sequence distance i. This is only exactly true if the DNA sequence is a stationary process. Drifts in GC content and other variations throughout a genome mean this is not true in general across a genome [41]. Therefore we need to consider a variety of other methods for analyzing correlations in DNA, from mutual information [42] to correlation functions [43], wavelets [40], fractal techniques such as the Higuchi method [44] and those discussed below. The mutual information measure, as given in Appendix C, can be used to measure the mutual information (and hence correlations) across an entire genome [45, 46]. An example of the use of this on the *Escherichia coli* K12 genome [33] is shown in Fig. 3. An excellent overview of these and other techniques for studying correlations in DNA, and the

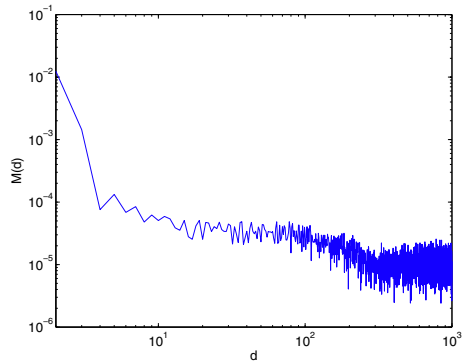


Fig. 3. This figure shows the mutual information plot of *E. coli* K12. The values of mutual information over the sets of bases separated by distance d were computed using Eq. C.12 (see Appendix C), for d up to 1000. Note that significant correlations exist only up to a few hundred bases.

implications of the results obtained using these methods can be found in a paper by Li [47].

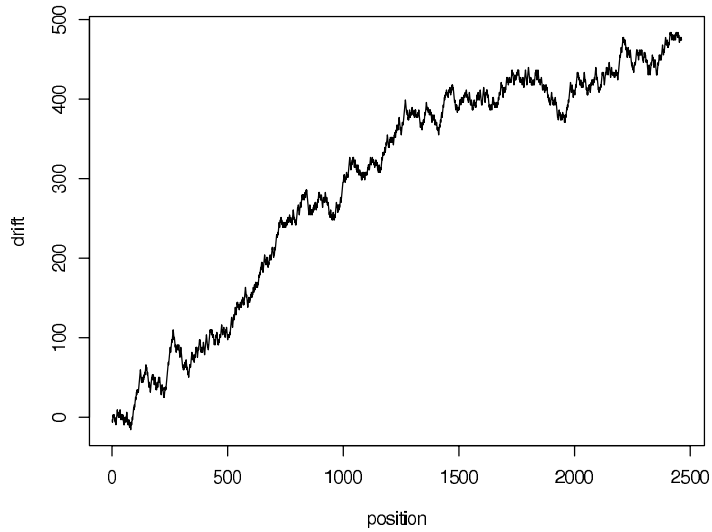
Correlations can arise as a result of genetic processes such as gene duplication and insertions [48], and these techniques can provide indication of such events, as well as other structures present in DNA sequences. Another technique used as part of correlation and structure detection is the DNA walk technique [49, 50]. Figure 4(a) shows the result of doing a "walk", where a step downwards is taken if a G or C is encountered in the sequence. A related technique is to map the bases onto complex numbers, and plot the cumulative phase [51]. We use the mapping for sequence element $s(i)$ given in Eq. 2,

$$\phi(i) = \begin{cases} \pi/4, & s(i) = A, \\ 3\pi/4, & s(i) = T, \\ -\pi/4, & s(i) = C, \\ -3\pi/4, & s(i) = G. \end{cases} \tag{2}$$

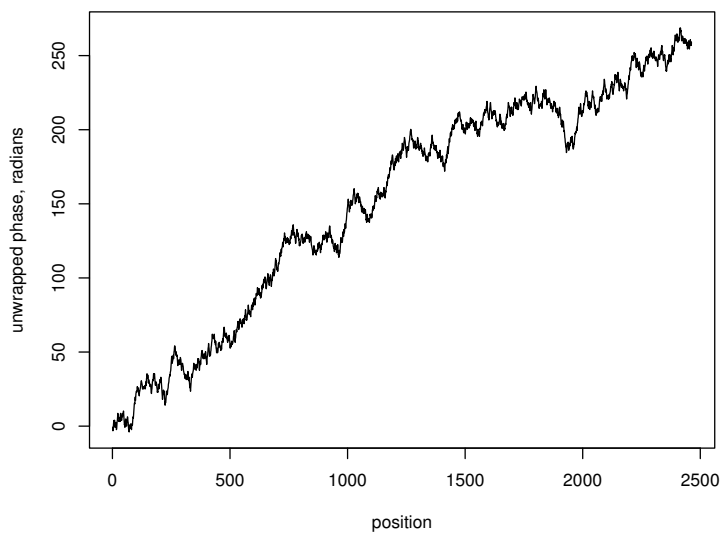
If the bases are evenly distributed, this gives an average phase of 0, and the mapping is designed to highlight the GC content, similar to the DNA walk. A phase plot is shown Fig. 4(b).

2.4. *Linguistics*

Since DNA and amino acid sequences can be thought of as a type of language, there is interest in the use of techniques from computational linguistics to analyze genetic sequences. This theory of grammar in a computational sense was first developed by Chomsky [52, 53]. It has been applied to a wide range of applications in sequence analysis from determining gene structures [54] to RNA (ribonucleic acid) secondary structure [55]. Mantegna *et al.* have taken methods from statistical linguistics, along with information theory approaches, to consider differences between non-coding and coding DNA [56, 57, 58]. This reveals the presence of hidden information and extra redundancy in non-coding regions, perhaps due to lengthy



(a) This plot shows the GC/AT content, with a step down if a G or C is encountered at a position in the sequence, else a step up is made.



(b) This plot shows the cumulative phase, with the phase added at each position determined by Eq. 2. Note the similarity to Fig. 4(a) due to the mapping used, but note that extra information is evident in the region from position 800 to position 1000.

Fig. 4. Two techniques have been used to show the structure in the DNA sequence of the *thrA* gene in *E. coli*. This sequence is clearly AT-rich, indicated by the upward trend of both graphs

promoter regions [59], or due to information left from now defunct coding regions. A good overview of linguistic techniques used can be found in Durbin *et al.* [13].

The PROSITE database contains a large number of protein families (related sequences), and their patterns, or "motifs" [60, 61]. This database can be searched using PROSITE patterns; an example of a pattern is

```
[ACFI]-[QC]-G-[AF]
```

where the capital letters denote amino acids. Square brackets denote that any one of the enclosed amino acids can occur in that position in the matched sequence, curly brackets (not used here) denote that none of the enclosed amino acids can occur in that position in the matched sequence. This can be written as a set of regular grammar rules, starting with position S and with W_i the positions of the sequence,

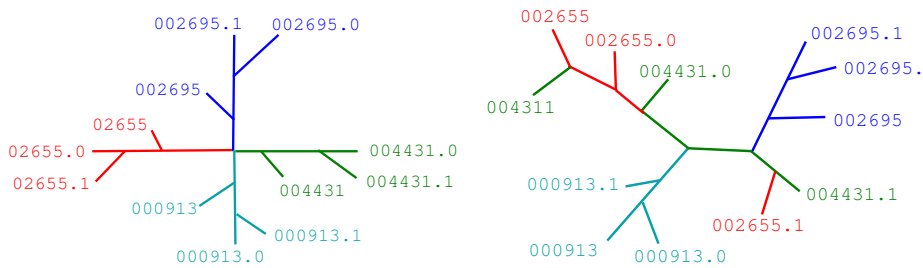
$$\begin{aligned} S &\rightarrow AW_1|CW_1|FW_1|IW_1 \\ W_1 &\rightarrow QW_2|CW_2 \\ W_2 &\rightarrow GW_3 \\ W_3 &\rightarrow A|F, \end{aligned} \tag{3}$$

where $\rightarrow$ means "rewrite as" and | means "or". Two example sequences which match this pattern are $AQGA$ and $FCGF$. The fact that PROSITE patterns can be written as regular grammars means the searching for sequences that contain the motif is highly efficient [13].

A new approach to feature detection in language is based upon inter-word spacing [62], or better referred to in a language context as word recurrence interval (WRI) [63], which is the number of words between each occurrence of a particular "word". In DNA one would consider inter-oligomer spacing. This has potential applications to classifying organisms [63], however to use this method on DNA and protein sequences one would have to define what a "word" is—for example, is it a gene or an exon? A method based on large scale structures like WRI, called gene order conservation, has shown some promise [64].

2.5. *Information theory and fractals*

Other techniques with possible applications in the area of sequence analysis include information-theoretic approaches [65], and related fractal approaches [26]. Multifractal approaches can be used to classify bacteria by a few numbers derived from the whole genome DNA sequence. Obviously this has limited use because of the large number of places in the genome sequence that even closely related species differ by, however the multifractal technique has shown some promise in general categorization of bacteria [26]. Other approaches based on information theory have been used in areas such as binding site recognition [65]. Phylogenetic trees are constructed from genetic data, and show the relationship between organisms based on their genetic data. Figure 5 shows two phylogenetic trees, one a known tree (left), and the other a tree constructed using a multifractal distance measure (right).



(a) Actual tree, of real *E. coli* and virtual descendants. The original ancestor, no longer in existence, is the central node of the tree, and the living descendants of this are bacteria indicated by the numerical parts of their accession numbers (NC_000913, NC_002655, NC_002695, and NC_004431). Descendants of those four descendants as generated by *in silico* stochastic mutations are indicated by suffixes .0 and .1.

(b) This is the tree of the same bacteria, generated using a multifractal measure of distance [26], and the neighbor joining algorithm [13], where a minimum distance means the two species are neighbors on the tree. The multifractal measure clearly has trouble distinguishing between such closely related species, but has some potential, and could be combined with other phylogenetic measures.

Fig. 5. These phylogenetic trees show the relationship between different organisms. Descendants of one organism are shown as branches from that organism, thus a family of related organisms with a common ancestor will all be on subtrees branching from the point at which that organism is shown.

3. Microarray Processing

3.1. *Introduction to microarray technology*

Microarrays, also known as gene or DNA chips, provide a relatively rapid way of analyzing gene expression patterns in an organism. Genes are expressed at different levels according to cell function, which may be altered in response to changes in its environment or may simply vary with time. The uses of microarray technology are numerous, and include identification of complex genetic diseases, drug discovery, pathogen detection and analysis, and detecting different expression of genes over time. Further details on microarray technology and potential uses may be found in the online Nature Genetics Chipping forecasts through `http://www.nature.com/ng/`

A microarray is an array of probes for detecting the expression levels of tens of thousands of genes simultaneously. In a typical two-color microarray experiment the relative expression levels between two target samples (cells) is measured for each probe. For each target, the mRNA is used to form complementary DNA (cDNA), labeled with a particular color dye. Typically green and red dyes are used. The two target cDNA samples are then passed over the probes, and the target cDNA fragments bind (hybridize) to probes according to matching probe sequences. The microarray is then imaged using a laser scanner that measures the fluorescence intensities of each dye. The ratio of intensities for each probe is a measure of the relative abundance, and hence gene expression level, of the corresponding DNA sequence in the two samples. So if a colored spot is bright yellow (bright green plus

bright red) it indicates both target samples have the gene corresponding to that spot highly expressed in equal amounts. See Fig. 6 for an example of a microarray image from a two-color system [66], and Yang *et al.* for a more detailed description on the hybridisation and scanning procedure [67].

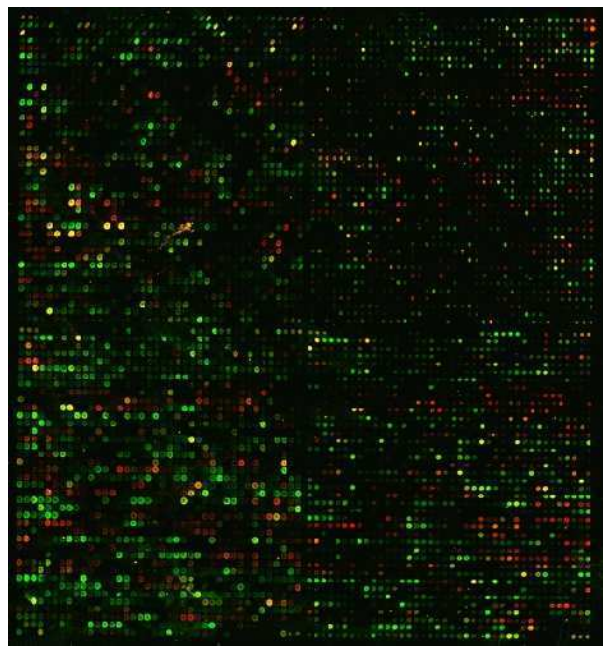


Fig. 6. A microarray showing gene expression for all 6000 yeast genes, from [66], with the two sets of data coming from various points in the *diauxic shift*, where the yeast go from fermentation (anaerobic respiration) to respiration (aerobic respiration). Each spot represents the expression level, as determined from the amount of mRNA washed over the microarray and bound to cDNA, of a pair of genes. Refer to the online version of this paper to see the full range of colors. The color of the spot shows which gene(s) is (are) being expressed: red only when the first gene is being expressed but not the second, and *vice versa* for green only. Other shades (mostly yellows, from an even mix of red of green) appear when both genes are being expressed. Bias in the spot size can be seen as one moves from the bottom-left to top-right of the image. A problem with the die can be seen in the orange streak approximately a quarter of the way down and across from the top-left corner.

Several different technologies exist for the printing of microarray slides. Two-color microarrays are printed using robotic arrayers that deposit probe material from cDNA or oligonucleotides libraries onto the microarray slide, or a modified inkjet printer, which builds up oligonucleotide probes base-pair by base-pair. Single color microarrays are produced using propriety technologies based on photolithography or the digital light processor and feature significantly higher spot density than their two-color relatives. The two different classes of microarray platform require different image analysis approaches. With a trend in minaturising the gene chips [68] combined with rapid image analysis of the results, it becomes possible to perform field tests for pathogens.

3.2. *Current applications of signal processing to microarrays*

The goal of processing microarrays is to turn the image of the array into a set of values giving the level of expression for each gene under analysis. The main tasks in processing this data are:

1. Clear the background noise from the image. The microarray will contain background noise that needs to be removed. For discussions of this noise, and techniques to remove it, refer to Haaland *et al.* [69].

2. Spot detection: a grid is overlayed over the array of spots, and those spots not occuring clearly within the grid cells are deleted (as sometimes a blotch will occur over several spots). Other irregular spots may be deleted, though if there is a reasonable number of pixels contained within the spot boundary it should still be usable regardless of shape. The edges of the annulus-shaped spots are often then detected, this helps with establishing the expression level of the spot (refer to Yang *et al.* [67]).

3. Normalization: the intensity of the spots represents the abundance of mRNA being expressed. Normalization is a process that removes any non-biological biases present, for example spatial or dye biases to allow the comparison of spots (genes) both within and between arrays. For most microarrays the majority of genes are not differentially expressed and normalisation approaches such as intensity-dependent robust local regression typically perform quite well. Control spots may be used if this is not the case. For more details on different normalisation approaches refer to Smyth and Speed [70]. Some new approaches to normalization based on non-linear methods are presented by Wilson *et al.* [71].

4. Identification of differentially expressed gene sets: Analysis methods to identify differentially expressed genes are actively being developed. Linear modelling and empirical Bayesian approaches are used to rank genes taking into account multiple testing issues, clustering and discrimination (unsupervised and supervised learning) techniques can be used to distinguish between different classes of treatment or diseases, and time series analysis can be performed to identify differences in gene regulation between samples [72, 73, 74, 75].

Biases may be caused by interactions between the cDNA sequences and dyes, dyes and arrays, and variations between arrays, among others. A variety of techniques have been developed to try and remove these biases, both by smarter design of experiments [76, 77] and also by intelligent choice of signal processing techniques [69, 78, 77]. Due to the costs involved in producing microarrays, it is important to have a cost-effective microarray experiment that maximises the amount of information produced. These design issues, and a way of designing cost-effective microarrays are given by Glonek and Solomon *et al.* [79].

One interesting technique for processing of microarray data is to use a CNNUM (cellular neural network universal machine) [80]. The main components of this electrical hardware are:

1. an array of analog processors, each one connected to all the surrounding processors,

2. a means of storing locally the intermediate computation results for each pixel, and

3. stored, programmable parameters.

The CNNUM is then programmed to implement the following steps:

1. To clear the image from the background noise, a sequence of thresholding and diffusion templates (sets of weights) are applied. This has the effect of quickly and accurately removing the background, even if the background luminosity varies across the microarray.

2. A set of operations are then applied, which first determine the grid in which the spots should lie, and then deletes those spots not in correct positions.

3. Four operations that remove small spots in any of the four directions (up, down, left, right) are then applied to remove those small irregular shaped spots, which are too small to be used accurately.

4. Another four operations which remove all unusable large irregular shaped spots are applied that operate similarly to those that remove the small irregular shaped spots.

5. A set of threshold operations are then performed, which classify the remaining well-defined spots into a set of expression levels.

Since the CNN algorithm is run in parallel on the spots, the overall time complexity is $O(n)$ in n, the number of spots (gene expression levels), as compared with more traditional techniques which operate in $O(n)$ time. As the gene expression levels can be obtained quickly with a high degree of accuracy in a CNNUM chip laid directly on top of a gene chip, it should be possible to build cheaper and faster microarray technologies for real-time analysis.

3.3. *Time series analysis of gene expression data*

Of interest to geneticists is not only what happens in the expression levels of two different samples at a fixed point in time, but how the expression levels vary over a number of different points in time. These experiments must be designed properly to ensure statistically significant information can be derived [79]. A number of different signal processing techniques have been developed to analyze such data, as well as "gene clusters" (sets of related genes) in microarrays. An overview of gene clustering algorithms can be found in Moreau *et al.* [81], and below we discuss some time series approaches.

One approach taken to analysis of time series microarray data, as well as other microarray data, is to take a standard statistical approach to determining factors affecting the output [72]. Bayesian network models have also been used to analyze time series microarray data [74]. With a Bayesian network model, one can efficiently analyze the relationships between the expression levels at different points in time, and between different genes. All the types of analysis that are used for microarray time series data have the property that they can make accurate predictions about

gene expression levels based on models with a limited amount of input data. The input data is limited due to the time and cost involved in preparing the microarray data.

One method for analysis of gene expression time series data is that developed by Bar-Joseph *et al.* [75]. Their method uses mathematical techniques similar to those developed by James and Hastie [82], however it deals properly with gene clusters—groups of related genes, which have correlated expression levels in a microarray analysis. The method fits curves to the limited number of data points available, which are limited due to the cost and time involved in preparing microarrays, and takes into account underlying biological processes and variability. The main steps of the method are outlined in Appendix D. The result of applying the above spline fitting and warping algorithms to gene expression levels in yeast are shown in Fig. 7.

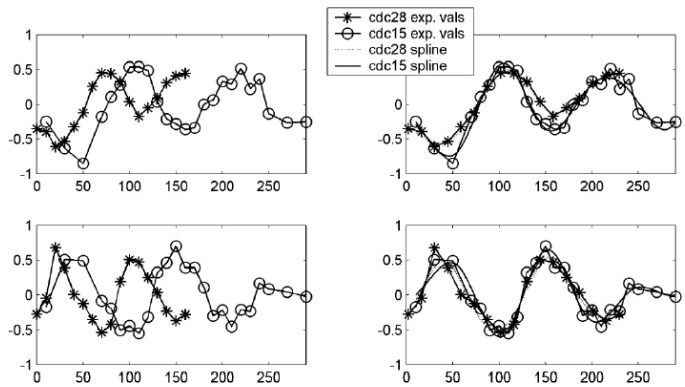


Fig. 7. These graphs, from [75], show the result of fitting spline curves to two sets of time data. These two sets are the expression levels of the yeast genes *cdc28* and *cdc15* as shown in the two left graphs. The results of applying the warping procedure is shown in the graphs on the right. The warping algorithm has changed the time scales so that the time series are aligned as intended.

3.4. *Future directions*

As microarray technology evolves, new applications, cheaper platforms, specialised microarrays, and increasing complex experimental designs are likely to be developed [68, 80]. Whilst existing techniques are still likely to be effective, such as the Bayesian network method and spline curve method for analysis of time series data, the generation of new or larger volumes of data will require the development of more sensitive and robust techniques to identify the biological information signal present amongst the noise.

As microarray technology becomes faster and cheaper [68, 80], the analysis of time series microarray data will become more commonplace. The existing techniques used, especially the Bayesian network method and spline curve method, will still be effective, however other techniques that work with a larger set of data will be able to be used, since it will be easier to generate larger sets of data. As more time series data is analyzed, it may be possible to build better models for clustering and for predicting the time responses of expression levels in response to a number of factors.

4. Conclusions

Vast amounts of data are generated by a wide variety of techniques in genetics. Signal processing methods, which have already made a great impact on a number of other areas, are part of a revolution in genetics as they are able to quickly and effectively process the large amount of data. In particular, signal procesing techniques can be used to rapidly process microarray data, making microarrays a much more powerful tool for genetic testing, drug development, and more.

Sequence analysis is an exponentially growing industry, as we explore more of the organisms we share our environment with, even the environment inside and on our own bodies. These sequences allow us to produce more effective drugs, better foods, biological solutions to pollution, and to gain valuable insights into the functioning of our own bodies.

Signal processing techniques show promise in being able to complement current techniques in analysis of genetic sequences. The genomes of over ten plants and animals and eighty bacteria are now available, and much of the data would benefit from further exploration. As advances in aerospace technology allowed us to reach out to the stars, so advances in genetic processing allow us to reach out to our own destinies.

Acknowledgements

Useful discussions with Doug Gray, The University of Adelaide, and Wentian Li, The Robert S. Boas Center for Genomics and Human Genetics at the North Shore LIJ Research Institute, are gratefully acknowledged.

References

[1] Nature Genetics editorial team, Wag the dogma, *Nature Genetics* **30** (2002) 343–344.

[2] L. H. Caporale, *Darwin in the Genome: Molecular Strategies in Biological Evolution* (McGraw-Hill, 2003).

[3] P. Clerc and P. Avner, Role of the region 3' to *xist* exon 6 in the counting process of X-chromosome inactivation, *Nature Genetics* **19** (1998) 249–253.

[4] J. S. Mattick, Non-coding RNAs: The architects of eukaryotic complexity, *EMBO Reports* **2** (2001) 986–991.

[5] D. C. Boyd, A. Pombo and S. Murphy, Interaction of proteins with promoter elements the human U2 snRNA genes *in vivo*, *Gene* **315** (2003) 103–112.

[6] G. P. Harmer, D. Abbott, P. G. Taylor and J. M. R. Parrondo, Brownian ratchets and Parrondo's games, *Chaos* **11** (2001) 705–714.

[7] D. C. Rubinsztein, W. Amos, J. Leggo, S. Goodburn, R. S. Ramesar, J. Old, R. Bontrop, R. McMahon, D. E. Barton and M. A. Ferguson-Smith, Mutational bias provides a model for the evolution of Huntington's disease and predicts a general increase in disease prevalance, *Nature Genetics* **7** (1994) 525–530.

[8] P. Worning, L. J. Jensen, K. E. Nelson, S. Brunak and D. W. Ussery, Structural analysis of DNA sequence: Evidence for lateral gene transfer, *Thermotoga maritima*, *Nucleic Acids Res.* **28** (2000) 706–709.

[9] J. P. Fitch and B. Sokhansanj, Genomic engineering: Moving beyond DNA sequence to function, *Proc. IEEE* **88** (2000) 1949–1971.

[10] D. Anastassiou, Genomic signal processing, *IEEE Signal Processing Magazine* **18** (2001) 8–20.

[11] D. Anastassiou, Frequency-domain analysis of biomolecular sequences, *Bioinformatics* **16** (2000) 1073–1081.

[12] Z. Yu, V. Anh and K. Lau, Measure representation and multifractal analysis of complete genomes, *Physical Review E* **64** (2001) 031903.

[13] R. Durbin, S. Eddy, A. Krogh and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids* (Cambridge University Press, 1998).

[14] A. V. Lukashin and M. Borodovsky, GeneMark.hmm: New solutions for gene finding, *Nucleic Acids Res.* **26** (1998) 1107–1115.

[15] X.-Y. Zhang, F. Chen, Y.-T. Zhang, S. C. Agner, M. Akay, Z.-H. Lu, M. M. Y. Waye and S.K.-W. Tsui, Signal processing techniques in genomic engineering, *Proc. IEEE* **90** (2002) 1822–1833.

[16] B. N. Kholodenko, A. Kiyatkin, F. J. Bruggeman, E. Sontag, H. V. Westerhoff and J. B. Hoek, Untangling the wires: A strategy to trace functional interactions in signaling and gene networks, *Proc. Natl. Acad. Sci. U.S.A.* **99** (2002) 12841–12846.

[17] J. T. Tyson, K. C. Chen and B. Novak, Sniffers, buzzers, toggles and blinkers: Dynamics of regulatory and signaling pathways in the cell, *Current Opinion in Cell Biology* **15** (2003) 221–231.

[18] M. Thattai and A. van Oudenaarden, Intrinsic noise in gene regulatory networks, *Proc. Natl. Acad. Sci. U.S.A.* **98** (2001) 8614–8619.

[19] A.-L. Barabási and Z. N. Oltvai, Network biology: Understanding the cell's functional organization, *Nature Reviews Genetics* **5** (2004) 101–113.

[20] H. Kobayashi, M. Kærn, M. Araki, K. Chung, T. S. Gardner, C. R. Cantor and J. J. Collins, Programmable cells: Interfacing natural and engineered gene networks, *Proc. Natl. Acad. Sci. U.S.A.* **101** (2004) 8414–8419.

[21] J. Hasty, D. McMillen and J. J. Collins, Engineered gene circuits, *Nature* **420** (2002) 224–230.

[22] E. M. Ozbudak, M. Thattai, I. Kurster, A. D. Grossman and A. van Oudenaarden, Regulation of noise in the expression of a single gene, *Nature Genetics* **31** (2002) 69–73.

[23] W. H. Majoros, M. Pertea and S. L. Salzberg, Efficient implementation of a generalized pair hidden Markov model for comparative gene finding, *Bioinformatics* **21** (2005) 1782–1788.

[24] N. von Öhsen, I. Sommer, R. Zimmer and T. Lengauer, Arby: Automatic protein structure prediction using profile-profile alignment and confidence measures, *Bioinformatics* **20** (2004) 2228–2235.

[25] M. J. Berryman, A. Allison, P. Carpena and D. Abbott, Signal processing and statistical methods in analysis of text and DNA, in *Proc. SPIE: Biomedical Applications of Micro- and Nanoengineering*, ed. D. V. Nicolau, Vol. 4937 (2002), pp. 231–240.

[26] M. J. Berryman, A. Allison and D. Abbott, Stochastic evolution and multifractal classification of prokaryotes, in *Proc. of the SPIE: Fluctuations and Noise in Biological, Biophysical and Biomedical Systems*, ed. S. M. Bezrukov, Vol. 5110 (2003), pp. 192–200.

[27] M. Itoh, S. Goto, T. Akutsu and M. Kanehisa, Fast and accurate database homology search using upper bounds of local alignment scores, *Bioinformatics* **21** (2005) 912–921.

[28] M. J. Berryman, A. Allison and D. Abbott, Mutual information for examining correlations in DNA, *Fluctuation and Noise Letters* **4** (2004) L237–L246.

[29] S. Winters-Hilt, W. Vercoutere, V. S. DeGuzman, D. Deamer, M. Akeson and D. Haussler, Highly accurate classification of Watson-Crick basepairs on termini of single DNA molecules, *Biophysical Journal* **84** (2003) 967–976.

[30] C. Bystroff, V. Thorsson and D. Baker, HMMSTR: A hidden Markov model for local sequence-structure correlations in proteins, *Journal of Molecular Biology* **301** (2000) 173–190.

[31] B. Rost, Better 1D predictions by experts with machines, *Proteins: Structure, Function and Genetics* **S1** (1997) 192–197.

[32] S. R. Eddy, HMMER: Profile hidden Markov models for biological sequence analysis (2001).

[33] T. Hayashi, K. Makino, M. Ohnishi, K. Kurokawa, K. Ishii, K. Yokoyama *et al.*, Complete genome sequence of enterohemorrhagic *escherichia coli* O157:H7 and genomic comparison with a laboratory strain K-12, *DNA Research* **8** (2001) 11–22.

[34] R. A. Welch, V. Burland, G. D. Plunkett, R. Redford, P. Roesch, D. Rasko *et al.*, Extensive mosaic structure revealed by the complete genome sequence of uropathogenic *Escherichia coli*, *Proc. Natl. Acad. Sci. U.S.A.* **99** (2002) 17020–17024.

[35] K. Putegowda, W. Worek, N. Pappas, A. Dandapani, P. Athanas and A. Dickerman, A run-time reconfigurable system for gene-sequence searching, in *Proc. 16th International Conference on VLSI Design*, IEEE Computer Society (2003), pp. 561–566.

[36] D. Sussillo, A. Kundaje and D. Anastassiou, Spectrogram analysis of genomes, *EURASIP Journal on Applied Signal Processing* **2004** (2004) 29–42.

[37] S. Tiwari, S. Ramachandran, A. Bhattacharya, S. Bhattacharya and R. Ramaswamy, Prediction of probable genes by Fourier analysis of genomic sequences, *Computer Applications in Biosciences* **13** (1997) 263–270.

[38] P. P. Vaidyanathan and B.-J. Yoon, Gene and exon prediction using allpass-based filters, in *Workshop on Genomic Signal Processing and Statistics (GENSIPS)* (2002).

[39] T. D. Read, S. N. Peterson, N. Tourasse, L. W. Baillie, I. T. Paulsen and K. E. Nelson *et al.*, The genome sequence of *Bacillus anthracis Ames* and comparison to closely related bacteria, *Nature* **423** (2003) 81–86.

[40] B. Audit, C. Vaillant, A. Arnéodo, Y. d'Aubenton Carafa and C. Thermes, Wavelet analysis of DNA bending profiles reveals structural constraints on the evolution of genomic sequences, *Journal of Biological Physics* **30** (2004) 33–81.

[41] International Human Genome Sequencing Consortium, Initial sequencing and analysis of the human genome, *Nature* **409** (2001) 860–921.

[42] W. Li, Mutual information functions versus correlation functions, *Journal of Statistical Physics* **60** (1990) 823–837.

[43] P. Bernaolo-Galván, P. Carpena, R. Román-Roldán and J. L. Oliver, Study of statistical correlations in DNA sequences, *Gene* **300** (2002) 105–115.

[44] T. Higuchi, Approach to an irregular time series on the basis of the fractal theory, *Physica D* **31** (1988) 277–283.

[45] D. Holste, I. Grosse and H. Herzel, Statistical analysis of the DNA sequence of human chromosome 22, *Physical Review E* **64** (2001) 041917.

[46] D. Holste, I. Grosse, S. Beirer, P. Schieg and H. Herzel, Repeats and correlations in human DNA sequences, *Physical Review E* **67** (2003) 061913.

[47] W. Li, The study of correlation structures of DNA sequences: A critical review, *Computers and Chemistry* **21** (1997) 257–271.

[48] W. Li, Generating nontrivial long-range correlations and 1/f spectra by replication and mutation, *International Journal of Bifurcation and Chaos* **2** (1992) 137–154.

[49] C.-K. Peng, S. V. Buldyrev, A. L. Goldberger, S. Havlin, F. Sciortino, M. Simon and H. E. Stanley, Long-range correlations in nucleotide sequences, *Nature* **356** (1992) 168–170.

[50] A. C. Frank and J. R. Lobry, Oriloc: Prediction of replication boundaries in unannotated bacterial chromosomes, *Bioinformatics* **16** (2000) 560–561.

[51] P. D. Cristea, Large scale features in DNA genomic signals, *Signal Processing* **83** (2003) 871–888.

[52] N. Chomsky, Three models for the description of language, *IRE Transactions on Information Theory* **2** (1956) 113–124.

[53] N. Chomsky, On certain formal properties of grammars, *Information and Control* **2** (1959) 137–167.

[54] S. Dong and D. B. Searls, Gene structure prediction by linguistic methods, *Genomics* **23** (1994) 540–551.

[55] S. R. Eddy and R. Durbin, RNA sequence analysis using covariance models, *Nucleic Acids Research* **22** (1994) 2079–2088.

[56] R. N. Mantegna, S. V. Buldyrev, A. L. Goldberger, S. Havlin, C.-K. Peng, M. Simons and H. E. Stanley, Linguistic features of non-coding DNA sequences, *Physical Review Letters* **73** (1994) 3169–3172.

[57] R. N. Mantegna, S. V. Buldyrev, A. L. Goldberger, S. Havlin, C.-K. Peng, M. Simons and H. E. Stanley, Systematic analysis of coding and noncoding DNA sequences using methods of statistical linguistics, *Physical Review E* **52** (1995) 2939–2950.

[58] R. N. Mantegna, S. V. Buldyrev, A. L. Goldberger, S. Havlin, C.-K. Peng, M. Simons and H. E. Stanley, Reply to comments on linguistic features of non-coding DNA sequences, *Physical Review Letters* **76** (1996) 1979–1981.

[59] S. Small, A. Blair and M. Levine, Regulation of even-skipped stripe 2 in the *Drosophila* embryo, *The EMBO Journal* **11** (1992) 4047–4057.

[60] P. Bucher and A. Bairoch, A generalized profile syntax for biomolecular sequences motifs and its function in automatic sequence interpretation, in *Proc. 2nd International Conference on Intelligent Systems for Molecular Biology*, eds. R. Altman, D. Brutlag, P. Karp, R. Lathrop and D. Searls (1994), pp. 53–61.

[61] L. Falquet, M. Pagni, P. Bucher, N. Hulo, C. J. A. Sigrist, K. Hofmann and A. Bairoch, The PROSITE database, its status in 2002, *Nucleic Acids Res.* **30** (2002) 235–238.

[62] M. Ortuño, P. Carpena, P. Bernaola-Galván, E. Muñoz and A. M. Somoza, Keyword detection in natural languages and DNA, *Europhysics Letters* **57** (2002) 759–764.

[63] M. J. Berryman, A. Allison and D. Abbott, Statistical techniques for text classification based on word recurrence intervals, *Fluctuations and Noise Letters* **3** (2003) L1–L10.

[64] B. M. E. Moret, J. Tang, L.-S. Wang and T. Warnow, Steps toward accurate reconstructions of phylogenies from gene-order data, *Journal of Computer and Systems Sciences* **65** (2002) 508–525.

[65] R. Mutihac, A. Cicuttin and R. C. Mutihac, Entropic approach to information coding in DNA molecules, *Materials Science and Engineering C* **18** (2001) 51–60.

[66] J. L. de Risi, V. R. Iyer and P. O. Brown, Exploring the metabolic and genetic control of gene expression on a genomic scale, *Science* **278** (1997) 680–686.

[67] Y. H. Yang, M. J. Buckley, S. Dudoit and T. P. Speed, Comparison of methods for image analysis on cDNA microarray data, *Journal of Computational and Graphical Statistics* **11** (2002) 108–136.

[68] C. H. Mastrangelo, M. A. Burns and D. T. Burke, Microfabricated devices for genetic diagnostics, *Proc. IEEE* **86** 1769–1787.

[69] M. J. Martinez, A. D. Aragon, A. L. Rodiguez, J. M. Weber, J. A. Timlin, M. B. Sinclair, D. M. Haaland and M. Werner-Washburne, Identification and removal of contaminating fluorescence from commercial and in-house printed DNA microarrays, *Nucleic Acids Res.* **31** (2003) e18:1–8.

[70] G. K. Smyth and T. P. Speed, Normalization of cDNA microarray data, in *Methods: Selecting Candidate Genes from DNA Array Screens: Application to Neuroscience Methods* **31** (2003) 265–273.

[71] D. L. Wilson, M. K. Buckley, C. A. Helliwell and I. W. Wilson, New normalization methods for cDNA microarray data, *Bioinformatics* **19** (2003) 1325–1332.

[72] L. P. Zhao, R. Prentice and L. Breeden, Statistical modeling of large microarray data sets to identify stimulus-response profiles, *Proc. Natl. Acad. Sci. U.S.A.* **98** (2001) 5631–5636.

[73] S. C. Madeira and A. L. Oliveira, Biclustering algorithms for biological data analysis: A survey, *IEEE Transactions on Computational Biology and Bioinformatics* **1** (2004) 24–45.

[74] N. Friedman, M. Linial, I. Nachman and D. Pe'er, Using Bayesian networks to analyze expression data, *Journal of Computation Biology* **7** (2000) 601–620.

[75] Z. Bar-Joseph, G. Gerber, D. Gifford, T. Jaakkola and I. Simon, A new approach to analyzing gene expression time series data, in *Proc. Sixth Annual International Conference on Research in Computational Molecular Biology* (2002), pp. 39–48.

[76] M. K. Kerr, M. Martin and G. A. Churchill, Analysis of variance for gene expression microarray data, *Journal of Computational Biology* **7** (2000) 819–837.

[77] G. Balázsi, K. A. Kay, A. L. Barabási and Z. N. Oltvai, Spurious spatial periodicity of co-expression in microarray data due to printing design, *Nucleic Acids Res.* **31** (2003) 4425–4433.

[78] J. A. Timlin, D. M. Haaland, M. B. Sinclair, A. D. Aragon, M. J. Martinez and M. Werner-Washburne, Hyperspectral microarray scanning: Impact on the accuracy and reliability of gene expression data, *BMC Genomics* **6** (2005).

[79] G. F. V. Glonek and P. J. Solomon, Factorial and time course designs for cDNA microarray experiments, *Biostatistics* **5** (2004) 89–111.

[80] P. Arena, L. Fortuna and L. Occhipinti, A CNN algorithm for real time analysis of DNA microarrays, *IEEE Transactions on Circuits and Systems – I: Fundamental Theory and Applications* **49** (2002) 335–340.

[81] Y. Moreau, F. de Smet, G. Thijs, K. Marchal and B. de Moor, Functional bioinformatics of microarray data: From expression to regulation, *Proc. IEEE* **90** (2002) 1722–1743.

[82] G. M. James and T. J. Hastie, Functional linear discriminant analysis for irregularly sampled curves, *Journal of the Royal Statistical Society Series B* **63** (2001) 533–550.

[83] S. Kullback, *Information Theory and Statistics* (Dover Publications, 1968).

[84] B. D. Silverman and R. Linsker, A measure of DNA periodicity, *Journal of Theoretical Biology* **118** (1986) 295–300.

[85] S. Tavaré and B. W. Giddings, *Mathematical Methods for DNA Sequences* (CRC Press, 1989), pp. 117–131.

[86] E. Coward, Equivalence of two Fourier methods for biological sequences, *Journal of Mathematical Biology* **36** (1997) 64–70.

Appendix A. Hidden Markov Models

Subsection 2.1 details the use of Markov models in sequence analysis. A Markov model of order k has a set of states S, with the probability of being in state $s \in S$

being dependent only on the previous k states. So for a discrete time sequence $s_1, \ldots, s_n$,

$$P\left(s_n = i_n | \, s_{n-1} = i_{n-1}, \ldots, s_1 = i_1\right) = P\left(s_n = i_n | s_{n-1} = i_{n-1}, \ldots, s_{n-k} = i_{n-k}\right), \tag{A.1}$$

except where $k = 0$ and the probability does not depend on the previous states. In a hidden Markov model, the states are unknown and must be inferred from the data. We find a model that maximizes the log likelihood (and thus the likelihood),

$$\log P\left(x|\theta\right) = \sum_y P\left(x, y|\theta\right), \tag{A.2}$$

for x the observed sequence, the y's are the possible sequences, and θ is the set of observed parameters. Then assume there exists a model θ^t, and we wish to see if there is a better model θ^{t+1}. Using Bayes' theorem, we can rewrite $\log P\left(x|\theta\right)$ as

$$\log P\left(x|\theta\right) = \log P\left(x, y|\theta\right) - \log P\left(y|x, \theta\right). \tag{A.3}$$

Using results from information theory [83], one can show that

$$\log P\left(x|\theta\right) - \log P\left(x|\theta^t\right) \geq Q\left(\theta|\theta^t\right) - Q\left(\theta^t|\theta^t\right), \tag{A.4}$$

where

$$Q\left(\theta|\theta^t\right) = \sum_y P\left(y|x, \theta^t\right) \log P\left(x, y|\theta\right). \tag{A.5}$$

Setting

$$\theta^{t+1} = \arg\max_\theta Q\left(\theta|\theta^t\right), \tag{A.6}$$

will always make the difference positive, and thus the log likelihood of the new model will be greater than the old one, unless $\theta^{t+1} = \theta^t$ in which case it stays the same. The above method is the expectation maximization algorithm, and forms the basis of the Baum-Welch algorithm used in hidden Markov model construction [13].

Appendix B. Fourier Transforms

Fourier transforms are used in a wide range applications such as voice prints for evidence in criminal cases, compressing images, removing noise from music, and of course in DNA sequence analysis (Subsec. 2.2) The discrete Fourier transform is given by

$$X(k) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-2\pi j n k / N}, \tag{B.7}$$

where $x(n)$ is the sequence of data ($n = 0, \ldots, N-1$), $j = \sqrt{-1}$, k is the discrete frequency, and $X(k)$ is the discrete Fourier transform at frequency k. For DNA sequences, we must transform the DNA sequence $s(n)$ into a numerical sequence $x(n)$, or in some cases several numerical sequences $x_i(n)$. One such transformation

is that used by Silverman and Linsker [84]. To a sequence of bases, denoted by $\mathbf{s} = s(1)s(2)\ldots s(N)$, a vector is assigned to each base $s(i)$ as per Eq. B.8,

$$\mathbf{x}(i) = \begin{cases} (1,0,0), & s(i) = A, \\ (-1/3, 0, 2\sqrt{2}/3), & s(i) = C, \\ (-1/3, -\sqrt{6}/3, -\sqrt{2}/3), & s(i) = G, \\ (-1/3, \sqrt{6}/3, -\sqrt{2}/3), & s(i) = T. \end{cases} \tag{B.8}$$

So for example the sequence ATG is represented by the sequence of vectors $(1,0,0)$, $(-1/3, \sqrt(6)/3, -\sqrt(2)/3), (-1/3, -\sqrt(6)/3, -\sqrt(2)/3)$. We then compute the power spectrum

$$P(f) = \sum_{c=1}^{3} \left| \frac{1}{N} \sum_{i=1}^{N} x(i)_c e^{-j2\pi if} \right|^2, \tag{B.9}$$

where $x(i)_c$ is the c-th component of $\mathbf{x}(i)$, and $j = \sqrt{-1}$. Here, N is the length of the sequence (number of bases). A simpler method is to use indicator functions

$$x(i) = \begin{cases} 1, s(i) = \alpha, \\ 0, \text{otherwise}, \end{cases} \tag{B.10}$$

for some $\alpha \in \{A, T, C, G\}$ [85]. The power spectra of these two methods are related through Eq. B.11 [86],

$$|Y(k)|^2 = \begin{cases} \dfrac{N}{N-1}|X(k)|^2, k \neq 0, \\ \dfrac{N}{N-1}|X(k)|^2 - \dfrac{c}{N-1}, \end{cases} \tag{B.11}$$

where N is the length of the sequences, c is a constant that varies with N, $X(k)$ is the Fourier transform of the indicator sequence, $Y(k)$ is the average of the Fourier transforms of the sequences of components of the vector sequence as given in Eq. B.9.

Appendix C. Mutual Information

The mutual information function, introduced in Subsec. 2.3, for symbols at distance d apart is given in Eq. C.12,

$$M(d) = \sum_{\alpha \in \mathcal{A}} \sum_{\beta \in \mathcal{A}} P_{\alpha\beta}(d) \log_2 \frac{P_{\alpha\beta}(d)}{P_\alpha P_\beta}, \tag{C.12}$$

for symbols $\alpha, \beta \in \mathcal{A}$, and in the case of DNA, $\mathcal{A} = \{\mathrm{A}, \mathrm{T}, \mathrm{C}, \mathrm{G}\}$. Here, $P_{\alpha\beta}(d)$ is the probability that symbols α and β are found a distance d apart. This is related to the correlation function in Eq. C.13 [42]:

$$\Gamma(d) = \sum_{\alpha \in \mathcal{A}} sum_{\beta \in \mathcal{A}} a_\alpha a_\beta P_{\alpha\beta}(d) - \left(\sum_{\alpha \in \mathcal{A}} a_\alpha P_\alpha \right)^2, \tag{C.13}$$

where a_α and a_β are numerical representations of symbols α and β.

Appendix D. Spline Curves

The use of spline curves was introduced in Subsec. 3.3. With a spline curve, one approximates a curve using a set of basic functions (often polynomials) that are fitted to the function at a set of points where the function used to approximate the curve can change, but must meet certain specifications (often ones designed to make the spline curve look smooth), and conditions are also specified on the ends of the spline curve. The main steps of the method used by Bar-Joseph *et al.* [75] are:

1. A spline curve model is developed that takes into account the gene cluster information, and fits a curve to the set of data points.
2. If the gene cluster information is not already available, an EM (Expectation and Maximization) algorithm is used to give estimates of the gene cluster information.
3. The curves are then scaled on the time axis, so that different realizations of biological processes can be compared.

In step one, we develop a spline curve using the model

$$Y_i(t) = s(t)(\mu_j + \gamma_i) + \epsilon_i, \tag{D.14}$$

where $Y_i(t)$ is the observed expression level for gene i at time t, $s(t)$ is a vector containing spline functions, μ_j is the average value of the spline coefficients for genes in cluster (or class) j, γ_i is the gene specific coefficients for gene i, and ϵ_i is a random noise term. If the μ_j or clusters are unknown, they are estimated using the following algorithm (note that MAP stands for Maximum *A Posteriori*):

```
TimeFit(Y, S, c, n)
    For all classes j {
        choose a random gene i
        initialize class center with a random gene
        calculate an initial value of μ_j
    }
    Initialize the other variables
    Repeat until the variables converge {
        E step:
            for all genes i and classes j
            compute the conditional probability p(j|i)
        M step:
            for all genes i and classes j
                find the MAP estimate of γ_{i,j}
            Maximize the other variables with respect to p(j, i)
            for all classes j, p_j ← (1/n) Σ_{i=1}^{n} p(j|i)
    }
}
```

The spline curves are then aligned using the following method. First denote a reference spline curve (that is, the one we are aligning to) as $g_i^{(1)}(s)$, where $s_{\min} \leq s \leq s_{\max}$, $s_{\min}$ and $s_{\max}$ are the start and end times. The splines to be aligned are denoted $g_i^{(2)}(t)$ for $t_{\min} \leq t \leq t_{\max}$. Then define a mapping for the time as $T(s) = t = (s - b)/a$. The alignment error e_i^2 for each gene is

$$e_i^2 = \frac{\int_\alpha^\beta \left[g_i^{(2)}(T(s)) - g_i^{(1)}(s) \right]^2}{\beta - \alpha}. \tag{D.15}$$

The error for a set of genes S of size n is then

$$E_S = \sum_{i=1}^{n} w_i e_i^2, \tag{D.16}$$

where $w_i = E_S/n$. Minimising E_S numerically then gives the alignment factors α and β.