SECTION SIX

Genomics and Proteomics

CHAPTER SEVENTEEN

Genome Mapping, Assembly, and Comparison

Genomics is the study of genomes. Genomic studies are characterized by simultaneous analysis of a large number of genes using automated data gathering tools. The topics of genomics range from genome mapping, sequencing, and functional genomic analysis to comparative genomic analysis. The advent of genomics and the ensuing explosion of sequence information are the main driving force behind the rapid development of bioinformatics today.

Genomic study can be tentatively divided into structural genomics and functional genomics. *Structural genomics* refers to the initial phase of genome analysis, which includes construction of genetic and physical maps of a genome, identification of genes, annotation of gene features, and comparison of genome structures. This is the major theme of discussion of this chapter. However, it should to be mentioned that the term *structural genomics* has already been used by a structural biology group for an initiative to determine three-dimensional structures of all proteins in a cell. Strictly speaking, the initiative of structural determination of proteins falls within the realm of *structural proteomics* and should not be confused as a subdiscipline of genomics. The structure genomics discussed herein mainly deals with structures of genome sequences. *Functional genomics* refers to the analysis of global gene expression and gene functions in a genome, which is discussed in Chapter 18.

GENOME MAPPING

The first step to understanding a genome structure is through genome mapping, which is a process of identifying relative locations of genes, mutations or traits on a chromosome. A low-resolution approach to mapping genomes is to describe the order and relative distances of genetic markers on a chromosome. *Genetic markers* are identifiable portions of a chromosome whose inheritance patterns can be followed. For many eukaryotes, genetic markers represent morphologic phenotypes. In addition to genetic linkage maps, there are also other types of genome maps such as physical maps and cytologic maps, which describe genomes at different levels of resolution. Their relations relative to the DNA sequence on a chromosome are illustrated in Figure 17.1. More details of each type of genome maps are discussed next.

Genetic linkage maps, also called *genetic maps*, identify the relative positions of genetic markers on a chromosome and are based on how frequent the markers are inherited together. The rationale behind genetic mapping is that the closer the two

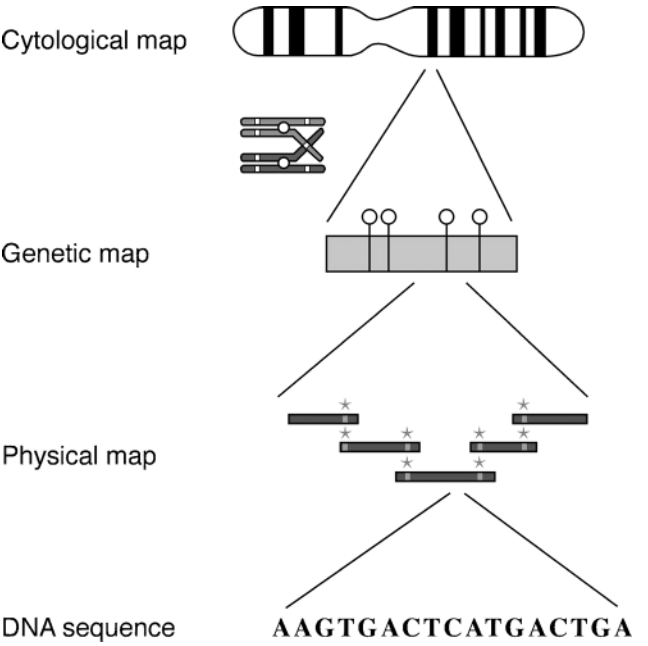


Figure 17.1: Overview of various genome maps relative to the genomic DNA sequence. The maps represent different levels of resolution to describe a genome using genetic markers. Cytologic maps are obtained microscopically. Genetic maps (*grey bar*) are obtained through genetic crossing experiments in which chromosome recombinations are analyzed. Physical maps are obtained from overlapping clones identified by hybridizing the clone fragments (grey bars) with common probes (grey asterisks).

genetic markers are, the more likely it is that they are inherited together and are not separated in a genetic crossing event. The distance between the two genetic markers is measured in centiMorgans (cM), which is the frequency of recombination of genetic markers. One centiMorgan is defined as one percentage of the total recombination events when separation of the two genetic markers is observed in a genetic crossing experiment. One centiMorgan is approximately 1 Mb in humans and 0.5 Mb in *Drosophila*.

Physical maps are maps of locations of identifiable landmarks on a genomic DNA regardless of inheritance patterns. The distance between genetic markers is measured directly as kilobases (Kb) or megabases (Mb). Because the distance is expressed in physical units, it is more accurate and reliable than centiMorgans used in genetic maps. Physical maps are constructed by using a chromosome walking technique, which uses a number of radiolabeled probes to hybridize to a library of DNA clone fragments. By identifying overlapping clones probed by common probes, a relative order of the cloned fragments can be established.

Cytologic maps refer to banding patterns seen on stained chromosomes, which can be directly observed under a microscope. The observable light and dark bands are the visually distinct markers on a chromosome. A genetic marker can be associated with a specific chromosomal band or region. The banding patterns, however, are not always constant and are subject to change depending on the extent

of chromosomal contraction. Thus, cytologic maps can be considered to be of very low resolution and hence somewhat inaccurate physical maps. The distance between two bands is expressed in relative units (Dustin units).

GENOME SEQUENCING

The highest resolution genome map is the genomic DNA sequence that can be considered as a type of physical map describing a genome at the single base-pair level. DNA sequencing is now routinely carried out using the Sanger method. This involves the use of DNA polymerases to synthesize DNA chains of varying lengths. The DNA synthesis is stopped by adding dideoxynucleotides. The dideoxynucleotides are labeled with fluorescent dyes, which terminate the DNA synthesis at positions containing all four bases, resulting in nested fragments that vary in length by a single base. When the labeled DNA is subjected to electrophoresis, the banding patterns in the gel reveal the DNA sequence.

The fluorescent traces of the DNA sequences are read by a computer program that assigns bases for each peak in a chromatogram. This process is called *base calling*. Automated base calling may generate errors and human intervention is often required to correct the sequence calls.

There are two major strategies for whole genome sequencing: the shotgun approach and the hierarchical approach. The *shotgun approach* randomly sequences clones from both ends of cloned DNA. This approach generates a large number of sequenced DNA fragments. The number of random fragments has to be very large, so large that the DNA fragments overlap sufficiently to cover the entire genome. This approach does not require knowledge of physical mapping of the clone fragments, but rather a robust computer assembly program to join the pieces of random fragments into a single, whole-genome sequence. Generally, the genome has to be redundantly sequenced in such a way that the overall length of the fragments covers the entire genome multiple times. This is designed to minimize sequencing errors and ensure correct assembly of a contiguous sequence. Overlapping sequences with an overall length of six to ten times the genome size are normally obtained for this purpose.

Despite the multiple coverage, sometimes certain genomic regions remain unsequenced, mainly owing to cloning difficulties. In such cases, the remainder gap sequences can be obtained through extending sequences from regions of known genomic sequences using a more traditional PCR technique, which requires the use of custom primers and performs genome walking in a stepwise fashion. This step of genome sequencing is also known as *finishing*, which is followed by computational assembly of all the sequence data into a final complete genome.

The hierarchical genome sequencing approach is similar to the shotgun approach, but on a smaller scale. The chromosomes are initially mapped using the physical mapping strategy. Longer fragments of genomic DNA (100 to 300 kB) are obtained

and cloned into a high-capacity bacterial vector called bacterial artificial chromosome (BAC). Based on the results of physical mapping, the locations and orders of the BAC clones on a chromosome can be determined. By successively sequencing adjacent BAC clone fragments, the entire genome can be covered. The complete sequence of each individual BAC clone can be obtained using the shotgun approach. Overlapping BAC clones are subsequently assembled into an entire genome sequence. Major differences between the hierarchical and the full shotgun approaches are shown in Figure 17.2.

During the era of human genome sequencing, there was a heated debate on the merits of each of the two strategies. In fact, there are advantages and disadvantages in either. The hierarchical approach is slower and more costly than the shotgun approach because it involves an initial clone-based physical mapping step. However, once the map is generated, assembly of the whole genome becomes relatively easy and less error prone. In contrast, the whole genome shotgun approach can produce a draft sequence very rapidly because it is based on the direct sequencing approach. However, it is computationally very demanding to assemble the short random fragments. Although the approach has been successfully employed in sequencing small microbial genomes, for a complex eukaryotic genome that contains high levels of repetitive sequences, such as the human genome, the full shotgun approach becomes less accurate and tends to leave more "holes" in the final assembled sequence than the hierarchical approach. Current genome sequencing of large organisms often uses a combination of both approaches.

GENOME SEQUENCE ASSEMBLY

As described, initial DNA sequencing reactions generate short sequence reads from DNA clones. The average length of the reads is about 500 bases. To assemble a whole genome sequence, these short fragments are joined to form larger fragments after removing overlaps. These longer, merged sequences are termed *contigs*, which are usually 5,000 to 10,000 bases long. A number of overlapping contigs can be further merged to form scaffolds (30,000–50,000 bases, also called *supercontigs*), which are unidirectionally oriented along a physical map of a chromosome (Fig. 17.3). Overlapping scaffolds are then connected to create the final highest resolution map of the genome.

Correct identification of overlaps and assembly of the sequence reads into contigs are like joining jigsaw puzzles, which can be very computationally intensive when dealing with data at the whole-genome level. The major challenges in genome assembly are sequence errors, contamination by bacterial vectors, and repetitive sequence regions. Sequence errors can often be corrected by drawing a consensus from an alignment of multiple overlapped sequences. Bacterial vector sequences can be removed using filtering programs prior to assembly. To overcome the problem of sequence repeats, programs such as RepeatMasker (see Chapter 4) can be used to detect and

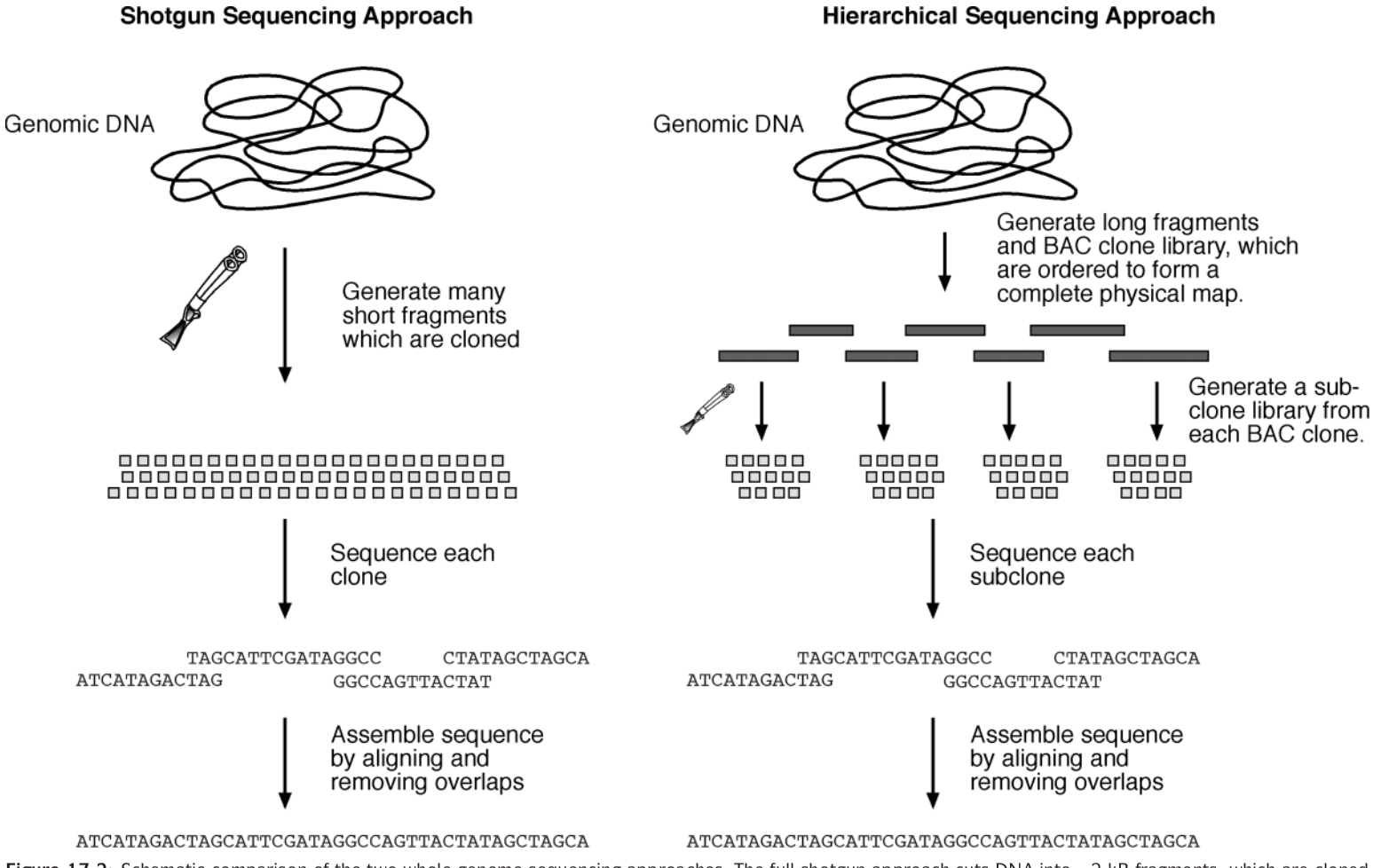


Figure 17.2: Schematic comparison of the two whole genome sequencing approaches. The full shotgun approach cuts DNA into ~2 kB fragments, which are cloned into small vectors and sequenced individually. The sequenced fragments are then put together into a final sequence in one step. The hierarchical approach cuts DNA into intermediate size fragments (~150 kB). The DNA fragments are cloned into BACs. A physical map has to be built based on the BAC clones. Each BAC clone is then subject to the shotgun approach.

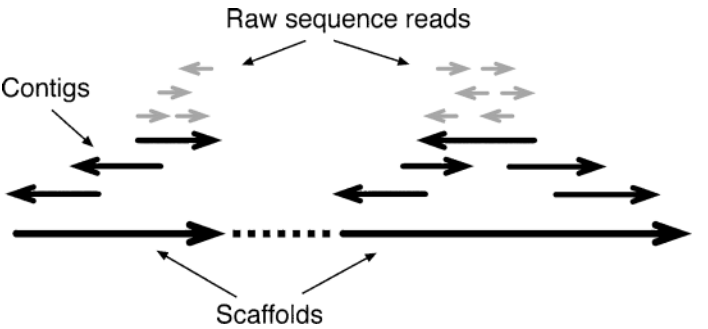


Figure 17.3: Schematic diagram showing three different levels of sequence assembly. Contigs are formed by combining raw sequence reads of various orientations after removing overlaps. Scaffolds are assembled from contigs and oriented unidirectionally on a chromosome. Because sequence fragments generated can be in either of the DNA strands, arrows are used to represent directionality of the sequences written in 5′ → 3′ orientation.

mask repeats. Additional constraints on the sequence reads can be applied to avoid misasembly caused by repeat sequences.

A commonly used constraint to avoid errors caused by sequence repeats is the so-called forward–reverse constraint. When a sequence is generated from both ends of a single clone, the distance between the two opposing fragments of a clone is fixed to a certain range, meaning that they are always separated by a distance defined by a clone length (normally 1,000 to 9,000 bases). When the constraint is applied, even when one of the fragments has a perfect match with a repetitive element outside the range, it is not able to be moved to that location to cause missassembly. An example of assembly with or without applying the forward–reverse constraints is shown in Figure 17.4.

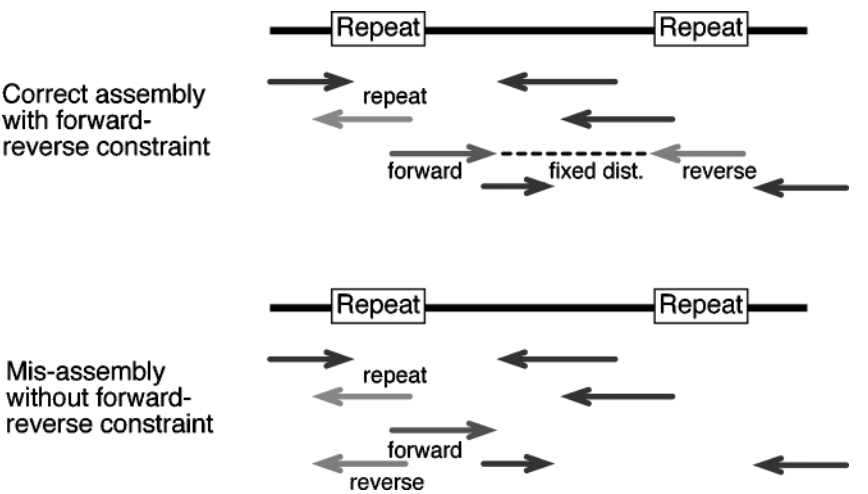


Figure 17.4: Example of sequence assembly with or without applying forward–reverse constraint, which fixes the sequence distance from both ends of a subclone. Without the restraint, the red fragment is misassembled due to matches of repetitive element in the middle of a fragment (see color plate section).

Base Calling and Assembly Programs

The first step toward genome assembly is to derive base calls and assign associated quality scores. The next step is to assemble the sequence reads into contiguous sequences. This step includes identifying overlaps between sequence fragments, assigning the order of the fragments and deriving a consensus of an overall sequence. Assembling all shotgun fragments into a full genome is a computationally very challenging step. There are a variety of programs available for processing the raw sequence data. The following is a selection of base calling and assembly programs commonly used in genome sequencing projects.

Phred (www.phrap.org/) is a UNIX program for base calling. It uses a Fourier analysis to resolve fluorescence traces and predict actual peak locations of bases. It also gives a probability score for each base call that may be attributable to error. The commonly accepted score threshold is twenty, which corresponds to a 1% chance of error. The higher the score, the better the quality of the sequence reads. If the score value falls below the threshold, human intervention is required.

Phrap (www.phrap.org/) is a UNIX program for sequence assembly. It takes Phred base-call files with quality scores as input and aligns individual fragments in a pairwise fashion using the Smith–Waterman algorithm. The base quality information is taken into account during the pairwise alignment. After all the pairwise sequence similarity is identified, the program performs assembly by progressively merging sequence pairs with decreasing similarity scores while removing overlapped regions. Consensus contigs are derived after joining all possible overlapped reads.

VecScreen (www.ncbi.nlm.nih.gov/VecScreen/VecScreen.html) is a web-based program that helps detect contaminating bacterial vector sequences. It scans an input nucleotide sequence and compares it with a database of known vector sequences by using the BLAST program.

TIGR Assembler (www.tigr.org/) is a UNIX program from TIGR for assembly of large shotgun sequence fragments. It treats the sequence input as clean reads without consideration of the sequence quality. A main feature of the program is the application of the forward–reverse constraints to avoid misassembly caused by sequence repeats. The sequence alignment in the assembly stage is performed using the Smith–Waterman algorithm.

ARACHNE (www-genome.wi.mit.edu/wga/) is a free UNIX program for the assembly of whole-genome shotgun reads. Its unique features include using a heuristic approach similar to FASTA to align overlapping fragments, evaluating alignments using statistical scores, correcting sequencing errors based on multiple sequence alignment, and using forward–reverse constraints. It accepts base calls with associated quality scores assigned by Phred as input and produces scaffolds or a fully assembled genome.

EULER (http://nbcr.sdsc.edu/euler/) is an assembly algorithm that uses a Eulerian Superpath approach, which is a polynomial algorithm for solving puzzles such as the famous “traveling salesman problem”: finding the shortest path of visiting a given

number of cities exactly once and returning to the starting point. In this approach, a sequence fragment is broken down to tuples of twenty nucleotides. The tuples are distributed in a diagram with numerous nodes that are all interconnected. The tuples are converted to binary vectors in the nodes. By using a Viterbi algorithm (see Chapter 6), the shortest path among the vectors can be found, which is the best way to connect the tuples into a full sequence. Because this approach does not directly rely on detecting overlaps, it may be advantageous in assembling sequences with repeat motifs.

GENOME ANNOTATION

Before the assembled sequence is deposited into a database, it has to be analyzed for useful biological features. The genome annotation process provides comments for the features. This involves two steps: gene prediction and functional assignment. Some examples of finished gene annotations in GenBank have been described in the Biological Database section (see Chapter 2). The following example illustrates the overall process employed in annotating the human genome.

As a real-world example, gene annotation of the human genome employs a combination of theoretical prediction and experimental verification. Gene structures are first predicted by ab initio exon prediction programs such as GenScan or FgenesH (see Chapter 8). The predictions are verified by BLAST searches against a sequence database. The predicted genes are further compared with experimentally determined cDNA and EST sequences using the pairwise alignment programs such as GeneWise, Spidey, SIM4, and EST2Genome. All predictions are manually checked by human curators. Once open reading frames are determined, functional assignment of the encoded proteins is carried out by homology searching using BLAST searches against a protein database. Further functional descriptions are added by searching protein motif and domain databases such as Pfam and InterPro (see Chapter 7) as well as by relying on published literature.

Gene Ontology

A problem arises when using existing literature because the description of a gene function uses natural language, which is often ambiguous and imprecise. Researchers working on different organisms tend to apply different terms to the same type of genes or proteins. Alternatively, the same terminology used in different organisms may actually refer to different genes or proteins. Therefore, there is a need to standardize protein functional descriptions. This demand has spurred the development of the gene ontology (GO) project, which uses a limited vocabulary to describe molecular functions, biological processes, and cellular components. The controlled vocabulary is organized such that a protein function is linked to the cellular function through a hierarchy of descriptions with increasing specificity. The top of the hierarchy provides an overall picture of the functional class, whereas the lower level

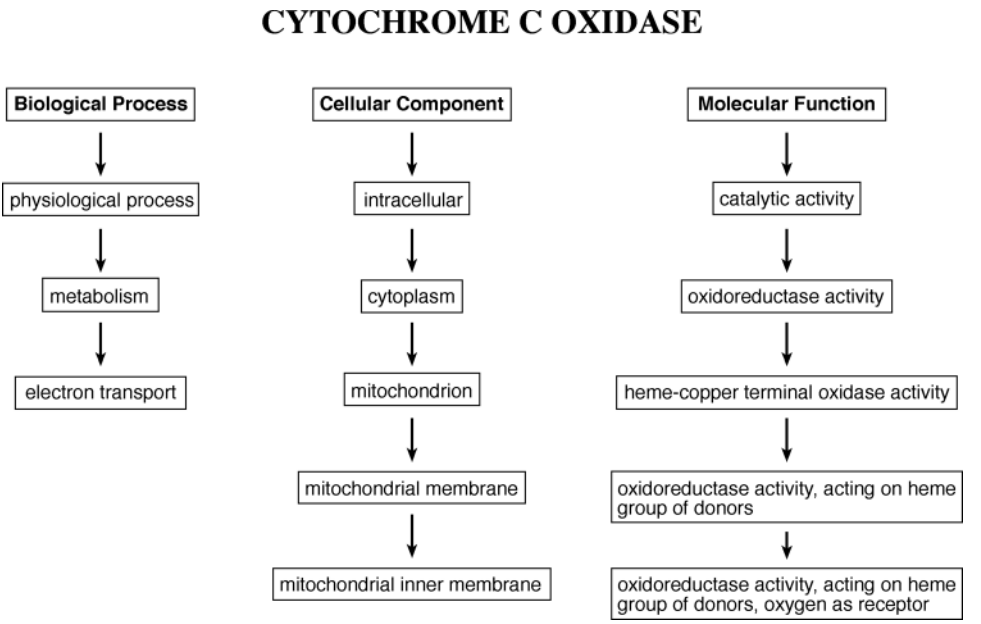


Figure 17.5: Example of GO annotation for cytochrome *c* oxidase. The functional and structural terms are arranged in three categories with a number of hierarchies indicating the levels of conceptual associations of protein functions.

in the hierarchy specifies more precisely the functional role. This way, protein functionality can be defined in a standardized and unambiguous way.

A GO description of a protein provides three sets of information: *biological process*, *cellular component*, and *molecular function*, each of which uses a unique set of nonoverlapping vocabularies. The standardization of the names, activities, and associated pathways provides consistency in describing overall protein functions and facilitates grouping of proteins of related functions. A database searching using GO for a particular protein can easily bring up other proteins of related functions in much the same way as using a thesaurus. Using GO, a genome annotator can assign functional properties of a gene product at different hierarchical levels, depending on how much is known about the gene product.

At present, the GO databases have been developed for a number of model organisms by an international consortium, in which each gene is associated with a hierarchy of GO terms. These have greatly facilitated genome annotation efforts. A good introduction of gene ontology can be found at www.geneontology.org. An example of GO annotation for cytochrome *c* oxidase is shown in Figure 17.5.

Automated Genome Annotation

With the genome sequence data being generated at an exponential rate, there is a need to develop fast and automated methods to annotate the genomic sequences. The automated approach relies on homology detection, which is essentially heuristic sequence similarity searching. If a newly sequenced gene or its gene product has

significant matches with a database sequence beyond a certain threshold, a transfer of functional assignment is taking place. In addition to sequence matching at the full length, detection of conserved motifs often offers additional functional clues.

Because using a single database searching method is often incomplete and error prone, automated methods have to mimic the manual process, which takes into consideration multiple lines of evidence in assigning a gene function, to minimize errors. The following algorithm is an example that goes a step beyond examining sequence similarity and provides functional annotations based on multiple protein characteristics.

GeneQuiz (http://jura.ebi.ac.uk:8765/ext-genequiz/) is a web server for protein sequence annotation. The program compares a query sequence against databases using BLAST and FASTA to identify homologs with high similarities. In addition, it performs domain analysis using the PROSITE and Blocks databases (see Chapter 7) as well as analysis of secondary structures and supersecondary structures that includes prediction of coiled coils and transmembrane helices. Multiple search and analysis results are compiled to produce a summary of protein function with an assigned confidence level (clear, tentative, marginal, and negligible).

Annotation of Hypothetical Proteins

Although a large number of genes and proteins can be assigned functions by the sequence similarity based approach, about 40% of the genes from newly sequenced genomes have no known functions and can only be annotated as genes encoding "hypothetical proteins." Experimental discovery of the functions of these genes and proteins is often time consuming and difficult because of lack of hypotheses to design experiments. In this situation, more advanced tools can be used for functional predictions by searching for remote homologs.

One way to obtain functional hints of genes encoding hypothetical proteins is by searching for remote homologs in databases. Detecting remote homologs typically involves combined searches of protein motifs and domains and prediction for secondary and tertiary structures. Conserved functional sites can be identified by profile and hidden Markov model–based motif and domain search tools such as SMART and InterPro (see Chapter 7). The prediction can also be performed using structure-based approaches such as threading and fold recognition (see Chapter 15). If the distant homologs detected using the structural approach are linked with well-defined functions, a broad functional class of the query protein if not the precise function of the protein can be inferred. In addition, prediction results for subcellular localization, protein–protein interactions can provide further functional hints (see Chapter 19).

These suggestions do not guarantee to provide correct annotations for the "hypothetical proteins," but they may provide critical hypotheses of the protein function that can be tested in the laboratory. The remote homology detection helps to shed light on the possible functions of the proteins that previously have no functional information at all. Thus, the bioinformatic analysis can spur an important advance in knowledge in many cases. Some hypothetical proteins, because of their novel

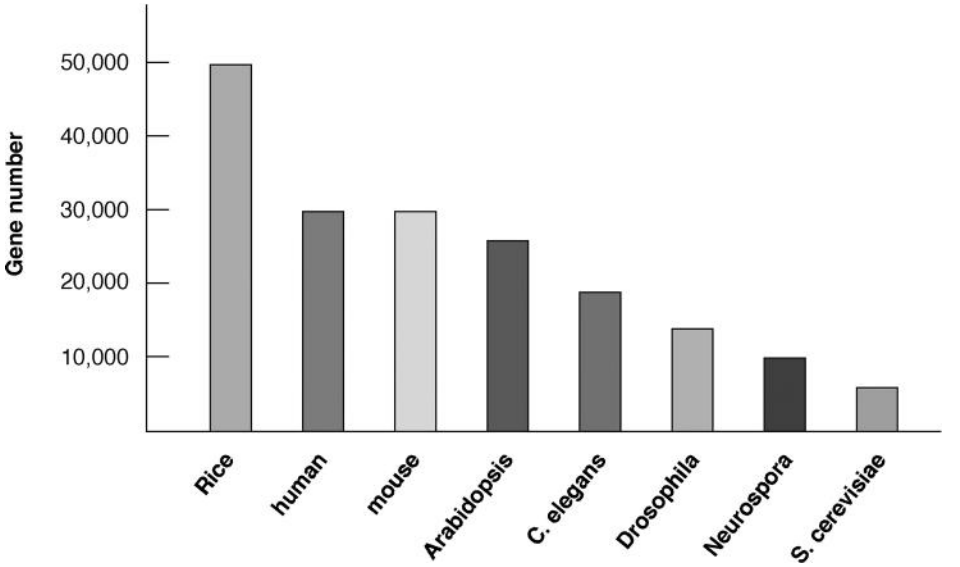


Figure 17.6: Gene numbers estimated from several sequenced eukaryotic genomes. (Data from Integrated Genomics Online Database http://ergo.integratedgenomics.com/GOLD/.)

structural folds, still cannot be predicted even with the advanced bioinformatics approaches and remain challenges for both experimental and computational work.

How Many Genes in a Genome?

One of the main tasks of genome annotation is to try to give a precise account of the total number of genes in a genome. This may be more feasible for prokaryotes as their gene structures are relatively simple. However, the number of genes in eukaryotic genomes, in particular the human genome, has been a subject of debate. This is mainly because of the complex structures of these genomes, which obscure gene prediction. Before the human genome sequencing was completed, the estimated gene numbers ranged from 20,000 to 120,000. Since the completion of the sequencing of the human genome, with the use of more sophisticated gene finding programs, the total number of human genes now dropped to close to 25,000 to 30,000. Although no exact number is agreed upon by all researchers, it is now widely believed that the total number of human genes will be no more than 30,000. This compares to estimates of 50,000 in rice, 30,000 in mouse, 26,000 in *Arabidopsis*, 18,400 in *C. elegans*, and 6,200 in yeast (Fig. 17.6).

The discovery of the low gene count in humans may be ego defeating to some as they realize that humans are only five times more complex than baker's yeast and apparently equally as complex as the mouse. What is worse, the food in their rice bowls has twice as many genes. The finding seriously challenges the view that humans are a superior species on Earth. As in many discoveries in scientific history, such as Darwin's evolutionary theory suggesting that humans arose from a "subhuman" ancestor, recent genomic discoveries have moved humans further away from this

exalted status. However, before we are overwhelmed by the humble realization, we should also realize that the complexity of an organism simply cannot be represented by gene numbers. As will soon become clear, gene expression and regulation, protein expression, modification, and interactions all contribute to the overall complexity of an organism.

Genome Economy

One level of genetic complexity is manifested at the protein expression level in which there are often more expressed proteins than genes available to code for them. For example, in humans, there are more than 100,000 proteins expressed based on EST analysis (see Chapter 18) compared to no more than 30,000 genes. If the "one gene, one protein" paradigm holds true, how could this discrepancy exist? Where does the extra coding power come from?

The answer lies in "genome economy," a phenomenon of synthesizing more proteins from fewer genes. This is a major strategy that eukaryotic organisms use to achieve a myriad of phenotypic diversities. There are many underlying genetic mechanisms to help account for genome economy. A major mechanism responsible for the protein diversity is *alternative splicing*, which refers to the splicing event that joins different exons from a single gene to form different transcripts. A related mechanism, known as *exon shuffling*, which joins exons from different genes to generate more transcripts, is also common in eukaryotes. It is known that, in humans, about two thirds of the genes exhibit alternative splicing and exon shuffling during expression, generating 90% of the total proteins. In *Drosophila*, the *DSCAM* gene contains 115 exons that can be alternatively spliced to produce 38,000 different proteins. This remarkable ability to generate protein diversity and new functions highlights the true complexity of a genome. It also illustrates the evolutionary significance of introns in eukaryotic genes, which serve as spacers that make the molecular recombination possible.

There are more surprising mechanisms responsible for genome economy. For example, trans-splicing can occur between RNAs produced from both DNA strands. In the *Drosophila mdg4* mutant, RNA transcribed from four exons in the sense strand and two exons in the antisense strand are joined to form a single mRNA. With different exon combinations, four different proteins can be produced. In some circumstances, one mRNA transcript can lead to the translation of more than one protein. For example, human dentin phosphoprotein and dentin sialoprotein are proteins involved in tooth formation. An mRNA transcript that includes coding regions from both proteins is translated into a precursor protein that is cleaved to produce two different mature proteins. Another situation, called "gene within gene," can be found in a gene for human prostate-specific antigen (PSA). In addition to regular PSA, humans can produce a similar protein, called PSA-LM, that functions antagonistically to PSA and is important for prostate cancer diagnosis. PSA-LM turns out to be encoded by the fourth intron of the PSA gene.

These are just a few known mechanisms of condensing the coding potential of genomic DNA to achieve increased protein diversity. From a bioinformatics point of

view, this makes gene prediction based on computational approaches all the more complicated. It also highlights one of the challenges that faces software program developers today. A number of databases have recently been established to archive alternatively spliced forms of eukaryotic genes. The following is one such example for human genes.

ProSplicer (http://prosplicer.mbc.nctu.edu.tw/) is a web-based database of human alternative spliced transcripts. The spliced variants are identified by aligning each known human protein, mRNA, and EST sequence against the genomic sequence using the SIM4 and TBLASTN program. The three sets of alignment are compiled to derive alternative splice forms. The database organizes data by tissue types and can be searched using keywords.

COMPARATIVE GENOMICS

Comparison of whole genomes from different organisms is comparative genomics, which includes comparison of gene number, gene location, and gene content from these genomes. The comparison helps to reveal the extent of conservation among genomes, which will provide insights into the mechanism of genome evolution and gene transfer among genomes. It helps to understand the pattern of acquisition of foreign genes through lateral gene transfer. It also helps to reveal the core set of genes common among different genomes, which should correspond to the genes that are crucial for survival. This knowledge can be potentially useful in future metabolic pathway engineering.

As alluded to previously, the main themes of comparative genomics include whole genome alignment, comparing gene order between genomes, constructing minimal genomes, and lateral gene transfer among genomes, each of which is discussed in more detail.

Whole Genome Alignment

With an ever-increasing number of genome sequences available, it becomes imperative to understand sequence conservation between genomes, which often helps to reveal the presence of conserved functional elements. This can be accomplished through direct genome comparison or genome alignment. The alignment at the genome level is fundamentally no different from the basic sequence alignment described in Chapters 3, 4, and 5. However, alignment of extremely large sequences presents new complexities owing to the sheer size of the sequences. Regular alignment programs tend to be error prone and inefficient when dealing with long stretches of DNA containing hundreds or thousands of genes. Another challenge of genome alignment is effective visualization of alignment results. Because it is obviously difficult to sift through and make sense of the extremely large alignments, a graphical representation is a must for interpretation of the result. Therefore, specific alignment algorithms are needed to deal with the unique challenges of whole genome alignment. A number of alignment programs for “super-long” DNA sequences are described next.

MUMmer (Maximal Unique Match, www.tigr.org/tigr-scripts/CMR2/webmum/mumplot) is a free UNIX program from TIGR for alignment of two entire genome sequences and comparison of the locations of orthologs. The program is essentially a modified BLAST, which, in the seeding step (see Chapter 4), finds the longest approximate matches that include mismatches instead of finding exact *k*-mer matches as in regular BLAST. The result of the alignment of whole genomes is shown as a dot plot with lines of connected dots to indicate collinearity of genes. It is optimized for pairwise comparison of closely related microbial genomes.

BLASTZ (http://bio.cse.psu.edu/) is a UNIX program modified from BLAST to do pairwise alignment of very large genomic DNA sequences. The modified BLAST program first masks repetitive sequences and searches for closely matched "words," which are defined as twelve identical matches within a stretch of nineteen nucleotides. The words serve as seeds for extension of alignment in both directions until the scores drop below a certain threshold. Nearby aligned regions are joined by using a weighted scheme that employs a unique gap penalty scheme that tolerates minor variations such as transitions in the seeding step of the alignment construction to increase its sensitivity.

LAGAN (Limited Area Global Alignment of Nucleotides; http://lagan.stanford.edu/) is a web-based program designed for pairwise alignment of large genomes. It first finds anchors between two genomic sequences using an algorithm that identifies short, exactly matching words. Regions that have high density of words are selected as anchors. The alignments around the anchors are built using the Needleman–Wunsch global alignment algorithm. Nearby aligned regions are further connected using the same algorithm. The unique feature of this program is that it is able to take into account degeneracy of the genetic codes and is therefore able to handle more distantly related genomes. Multi-LAGAN, an extension of LAGAN, available from the same website, performs multiple alignment of genomes using a progressive approach similar to that used in Clustal (see Chapter 5).

PipMaker (http://bio.cse.psu.edu/cgi-bin/pipmaker?basic) is a web server using the BLASTZ heuristic method to find similar regions in two DNA sequences. It produces a textual output of the alignment result and also a graphical output that presents the alignment as a percent identity plot as well as a dot plot. For comparing multiple genomes, MultiPipMaker is available from the same site.

MAVID (http://baboon.math.berkeley.edu/mavid/) is a web-based program for aligning multiple large DNA sequences. MAVID is based on a progressive alignment algorithm similar to Clustal. It produces an NJ tree as a guide tree. The sequences are aligned recursively using a heuristic pairwise alignment program called AVID. AVID works by first selecting anchors using the Smith–Waterman algorithm and then building alignments for the sequences between nearby anchors. Connected alignments are treated as new anchors for building longer alignments. The process is repeated iteratively until the entire sequence pair including weakly conserved regions are aligned.

GenomeVista (http://pipeline.lbl.gov/cgi-bin/GenomeVista) is a database searching program that searches against the human, mouse, rat, or *Drosophila* genomes using a large piece of DNA as query. It uses a program called BLAT to find anchors and

extends the alignment from the anchors using AVID. (BLAT is a fast local alignment algorithm that aligns short sequences of forty bases with more than 95% similarity.) It produces a graphical output that shows the sequence percent identity.

Finding a Minimal Genome

One of the goals of genome comparison is to understand what constitutes a minimal genome, which is a minimal set of genes required for maintaining a free-living cellular organism. Finding minimal genomes helps provide an understanding of genes constituting key metabolic pathways, which are critical for a cell's survival. This analysis involves identification of orthologous genes shared between a number of divergent genomes.

Coregenes (http://pasteur.atcc.org:8050/CoreGenes1.0//) is a web-based program that determines a core set of genes based on comparison of four small genomes. The user supplies NCBI accession numbers for the genomes of interest. The program performs an iterative BLAST comparison to find orthologous genes by using one genome as a reference and another as a query. This pairwise comparison is performed for all four genomes. As a result, the common genes are compiled as a core set of genes from the genomes.

Lateral Gene Transfer

Lateral gene transfer (or *horizontal gene transfer*) is defined as the exchange of genetic materials between species in a way that is incongruent with commonly accepted vertical evolutionary pathway. Lateral gene transfer mainly occurs among prokaryotic organisms when foreign genes are acquired through mechanisms such as transformation (direct uptake of foreign DNA from environment), conjugation (gene uptake through mating behavior), and transduction (gene uptake mediated by infecting viruses). The transmission of genes between organisms can occur relatively recently or as a more ancient event.

If lateral transfer events occurred relatively recently, one would expect to discover traces of the transfer by detecting regions of genomic sequence with unusual properties compared to surrounding regions. The unusual characteristics to be examined include nucleotide composition, codon usage, and amino acid composition. This can be considered a "within-genome" approach. Another way to discern lateral gene transfer is through phylogenetic analysis (see Chapters 10 and 11), referred to as an "among-genome" approach, which can be used to discover both recent and ancient lateral gene transfer events. Abnormal groupings in phylogenetic trees are often interpreted as the possibility of lateral gene transfer events. Because phylogenetic analyses have been described in detail in previous chapters, the following introduces basic tools for identifying genomic regions that may be a result of lateral gene transfer events using the within-genome approach.

Within-Genome Approach

This approach is to identify regions within a genome with unusual compositions. Single or oligonucleotide statistics, such as G–C composition, codon bias, and

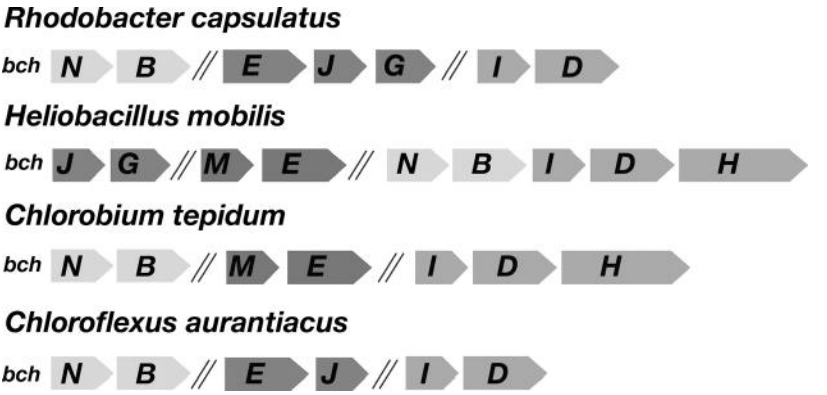


Figure 17.7: Schematic diagram showing a conserved linkage pattern of photosynthesis genes among four divergent photosynthetic bacterial groups. The synteny reveals potential physical interactions of encoded proteins, some of which have been experimentally verified. All the genes shown (*bch*) are involved in the pathway of bacteriochlorophyll biosynthesis. Intergenic regions of unspecified lengths are indicated by forward slashes (//). (*Source:* from Xiong et al., 2000; reproduced with permission from *Science*).

oligonucleotide frequencies are used. Unusual nucleotide statistics in certain genomic regions versus the rest of the genome may help to identify "foreign" genes in a genome. A commonly used parameter is GC skew $((G - C)/(G + C))$, which is compositional bias for G in a DNA sequence and is a commonly used indicator for newly acquired genetic elements.

ACT (Artemis Comparison Tool; www.sanger.ac.uk/Software/ACT) is a pairwise genomic DNA sequence comparison program (written in Java and run on UNIX, Macintosh, and Windows) for detecting gene insertions and deletions among related genomes. The pairwise sequence alignment is conducted using BLAST. The display feature includes showing collinear as well as noncollinear (rearrangement) regions between two genomes. It also calculates GC biases to indicate nucleotide patterns. However, it is up to the genome annotators to determine whether the observations constitute evidence for lateral gene transfer, as this requires combining evidence from multiple approaches.

Swaap (http://www.bacteriamuseum.org/SWAAP/SwaapPage.htm) is a Windows program that is able to distinguish coding versus noncoding regions and measure GC skews, oligonucleotide frequencies in a genomic sequence.

Gene Order Comparison

Another aspect of comparative genomics is the comparison of gene order. When the order of a number of linked genes is conserved between genomes, it is called *synteny*. Generally speaking, gene order is much less conserved compared with gene sequences. Gene order conservation is in fact rarely observed among divergent species. Therefore, comparison of syntenic relationships is normally carried out between relatively close lineages. However, if syntenic relationships for certain genes are indeed observed among divergent prokaryotes, they often provide important clues to functional relationships of the genes of interest. For example, genes involved in the

same metabolic pathway tend to be clustered among phylogenetically diverse organisms. The preservation of the gene order is a result of the selective pressure to allow the genes to be coregulated and function as an operon. Furthermore, the synteny of genes from divergent groups often associates with physical interactions of the encoded gene products. The use of conserved gene neighbors as predictors of protein interactions is discussed in Chapter 18. An example of synteny of bacterial photosynthesis genes coupled with protein interactions is illustrated in Figure 17.7.

GeneOrder (http://pumpkins.ib3.gmu.edu:8080/geneorder/) is a web-based program that allows direct comparison of a pair of genomic sequences of less than 2 Mb. It displays a dot plot with diagonal lines denoting collinearity of genes and lines off the diagonal indicating inversions or rearrangements in the genomes.

SUMMARY

Genome mapping using relative positions of genetic markers without knowledge of sequence data is a low-resolution approach to describing genome structures. A genome can be described at the highest resolution by a complete genome sequence. Whole-genome sequencing can be carried out using full shotgun or hierarchical approaches. The former requires more extensive computational power in the assembly step, and the latter is inefficient because of the physical mapping process required. Among the genome sequence assembly programs, ARACHNE and EULER are the best performers. Genome annotation includes gene finding and assignment of function to these genes. Functional assignment depends on homology searching and literature information. GO projects aim to facilitate automated annotation by standardizing the descriptions used for gene functions. The exact number of genes in the human genome is unknown, but is likely to be in the same range as most other eukaryotes. The gene number, however, does not dictate complexities of a genome. One example is exhibited in protein expression in which a larger number of proteins are produced than genes available to code for them. This is the so-called genome economy. The main mechanisms responsible for genome economy are alternative splicing and exon shuffling. Genomes can be compared on the basis of their gene content and gene order. Many specialized genome comparison programs for cross-genome alignment have been developed. Among them, BLASTZ and LAGAN may be the best in terms of speed and accuracy. Gene order comparison across genomes often helps to discover potential operons and assign putative functions. Conserved gene order among prokaryotes is often indicative of protein physical interactions.

FURTHER READING

Bennetzen, J. 2002. Opening the door to comparative plant biology. *Science* 296:60–3.

Chain, P., Kurtz, S., Ohlebusch, E., and Slezak, T. 2003. An applications-focused review of comparative genomics tools: Capabilities, limitations and future challenges. *Brief. Bioinform.* 4:105–23.

Dandekar, T, Snel, B., Huynen, M., and Bork, P. 1998. Conservation of gene order: A fingerprint of proteins that physically interact. *Trends Biochem. Sci*. 23:324–8.

Frazer, K. A., Elnitski, L., Church, D. M., Dubchak, I., and Hardison, R. C. 2003. Cross-species sequence comparisons: A review of methods and available resources. *Genome Res*. 13:1–12.

Gibson, G., and Muse, S. V. 2002. *A Primer of Genome Science*. Sunderland, MA: Sinauer Associates.

Karlin, S., Mrazek, J., and Gentles, A. J. 2003. Genome comparisons and analysis. *Curr. Opin. Struct. Biol.* 13:344–52.

Lewis, R., and Palevitz, B. A. 2001. Genome economy. *The Scientist* 15:21.

Lio, P. 2003. Statistical bioinformatic methods in microbial genome analysis. *BioEssays* 25:266–73.

Michalovich, D., Overington, J., and Fagan, R. 2002. Protein sequence analysis in silico: Application of structure-based bioinformatics to genomic initiatives. *Curr. Opin. Pharmacol.* 2:574–80.

Pennacchio, L. A., and Rubin, E. M. 2003. Comparative genomic tools and databases: Providing insights into the human genome. *J. Clin. Invest*. 111:1099–106.

Primrose, S. B., and Twyman, R. M. 2003. *Principles of Genome Analysis and Genomics*, 3rd ed. Oxford, UK: Blackwell.

Stein, L. 2001. Genome annotation: From sequence to biology. *Nat. Rev. Genetics* 2:493–503.

Sterky, F., and Lundeberg, J. 2000. Sequence analysis of gene and genomes. *J. Biotechnol.* 76:1–31.

Ureta-Vidal, A., Ettwiller, L., and Birney, E. 2003. Comparative genomics: Genome-wide analysis of metazoan eukaryotes. *Nature Rev. Genetics* 4:251–62.

Waterston, R. H., Lindblad-Toh, K., Birney, E., Rogers, J., Abril, J. F., Agarwal, P., Agarwala, R., et al. 2002. Initial sequencing and comparative analysis of the mouse genome. *Nature* 420:520–62.

Wei, L., Liu, Y., Dubchak, I., Shon, J., and Park, J. 2002. Comparative genomics approaches to study organism similarities and differences. *J. Biomed. Inform*. 35:142–50.

Xiong, J., Fischer, W. M., Inoue, K., Nakahara, M., and Bauer, C. E. 2000. Molecular evidence for the early evolution of photosynthesis. *Science* 289:1724–30.

CHAPTER EIGHTEEN

Functional Genomics

The field of genomics encompasses two main areas, structural genomics and functional genomics (see Chapter 17). The former mainly deals with genome structures with a focus on the study of genome mapping and assembly as well as genome annotation and comparison; the latter is largely experiment based with a focus on gene functions at the whole genome level using high throughput approaches. The emphasis here is on "high throughput," which is simultaneous analysis of all genes in a genome. This feature is in fact what separates genomics from traditional molecular biology, which studies only one gene at a time.

The high throughput analysis of all expressed genes is also termed *transcriptome analysis*, which is the expression analysis of the full set of RNA molecules produced by a cell under a given set of conditions. In practice, messenger RNA (mRNA) is the only RNA species being studied. Transcriptome analysis facilitates our understanding of how sets of genes work together to form metabolic, regulatory, and signaling pathways within the cell. It reveals patterns of coexpressed and coregulated genes and allows determination of the functions of genes that were previously uncharacterized. In short, functional genomics provides insight into the biological functions of the whole genome through automated high throughput expression analysis. This chapter mainly discusses the bioinformatics aspect of the transcriptome analysis that can be conducted using either sequence- or microarray-based approaches.

SEQUENCE-BASED APPROACHES

Expressed Sequence Tags

One of the high throughput approaches to genome-wide profiling of gene expression is sequencing expressed sequence tags (ESTs). ESTs are short sequences obtained from cDNA clones and serve as short identifiers of full-length genes. ESTs are typically in the range of 200 to 400 nucleotides in length obtained from either the 5′ end or 3′ end of cDNA inserts. Libraries of cDNA clones are prepared through reverse transcription of isolated mRNA populations by using oligo(dT) primers that hybridize with the poly(A) tail of mRNAs and ligation of the cDNAs to cloning vectors. To generate EST data, clones in the cDNA library are randomly selected for sequencing from either end of the inserts.

The EST data are able to provide a rough estimate of genes that are actively expressed in a genome under a particular physiological condition. This is because

the frequencies for particular ESTs reflect the abundance of the corresponding mRNA in a cell, which corresponds to the levels of gene expression at that condition. Another potential benefit of EST sampling is that, by randomly sequencing cDNA clones, it is possible to discover new genes.

However, there are also many drawbacks of using ESTs for expression profile analysis. EST sequences are often of low quality because they are automatically generated without verification and thus contain high error rates. Many bases are ambiguously determined, represented by *N*'s. Common errors also include frameshift errors and artifactual stop codons, resulting in failures of translating the sequences. In addition, there is often contamination by vector sequence, introns (from unspliced RNAs), ribosomal RNA (rRNA), mitochondrial RNA, among others. ESTs represent only partial sequences of genes. Gene sequences at the 3′ end tend to be more heavily represented than those at the 5′ end because reverse transcription is primed with oligo(dT) primers. Unfortunately, the sequences from the 3′ end are also most error prone because of the low base-call quality at the start of sequence reads. Another problem of ESTs is the presence of chimeric clones owing to cloning artifacts in library construction, in which more than one transcript is ligated in a clone resulting in the 5′ end of a sequence representing one gene and the 3′ end another gene. It has been estimated that up to 11% of cDNA clones may be chimeric. Another fundamental problem with EST profiling is that it predominantly represents highly expressed, abundant transcripts. Weakly expressed genes are hardly found in a EST sequencing survey.

Despite these limitations, EST technology is still widely used. This is because EST libraries can be easily generated from various cell lines, tissues, organs, and at various developmental stages. ESTs can also facilitate the unique identification of a gene from a cDNA library; a short tag can lead to a cDNA clone. Although individual ESTs are prone to error, an entire collection of ESTs contains valuable information. Often, after consolidation of multiple EST sequences, a full-length cDNA can be derived. By searching a nonredundant EST collection, one can identify potential genes of interest.

The rapid accumulation of EST sequences has prompted the establishment of public and private databases to archive the data. For example, GenBank has a special EST database, dbEST (www.ncbi.nlm.nih.gov/dbEST/) that contains EST collections for a large number of organisms (>250). The database is regularly updated to reflect the progress of various EST sequencing projects. Each newly submitted EST sequence is subject to a database search. If a strong similarity to a known gene is found, it is annotated accordingly.

EST Index Construction

One of the goals of the EST databases is to organize and consolidate the largely redundant EST data to improve the quality of the sequence information so the data can be used to extract full-length cDNAs. The process includes a preprocessing step that removes vector contaminants and masks repeats. Vecscreen, introduced in

Chapter 17, can be used to screen out bacterial vector sequences. This is followed by a clustering step that associates EST sequences with unique genes. The next step is to derive consensus sequences by fusing redundant, overlapping ESTs and to correct errors, especially frameshift errors. This step results in longer EST contigs. The procedure is somewhat similar to the genome assembly of shotgun sequence reads (see Chapter 17) . Finally, the coding regions are defined through the use of HMM-based gene-finding algorithms (see Chapter 8). This helps to exclude the potential intron and 3′-untranslated sequences. Once the coding sequence is identified, it can be annotated by translating it into protein sequences for database similarity searching. To go another step further, compiled ESTs can be used to align with the genomic sequence if available to identify the genome locus of the expressed gene as well as intron–exon boundaries of the gene. This is usually performed using the program SIM4 (http://pbil.univ-lyon1.fr/sim4.php).

The clustering process that reduces the EST redundancy and produces a collection of nonredundant and annotated EST sequences is known as *gene index construction*. The following lists a couple of major databases that index EST sequences.

UniGene (www.ncbi.nlm.nih.gov/UniGene/) is an NCBI EST cluster database. Each cluster is a set of overlapping EST sequences that are computationally processed to represent a single expressed gene. The database is constructed based on combined information from dbEST, GenBank mRNA database, and "electronically spliced" genomic DNA. Only ESTs with 3′ poly-A ends are clustered to minimize the the problem of chimerism. The resulting 3′ EST sequences provide more unique representation of the transcripts. The next step is to remove contaminant sequences that include bacterial vectors and linker sequences. The cleaned ESTs are used to search against a database of known unique genes (EGAD database) with the BLAST program. The compiling step identifies sequence overlaps and derives sequence consensus using the CAP3 program. During this step, errors in individual ESTs are corrected; the sequences are then partitioned into clusters and assembled into contigs. The final result is a set of nonredundant, gene-oriented clusters known as *UniGene clusters*. Each UniGene cluster represents a unique gene and is further annotated for putative function and its gene locus information, as well as information related to the tissue type where the gene has been expressed. The entire clustering procedure is outlined in Figure 18.1.

TIGR Gene Indices (www.tigr.org/tdb/tgi.shtml) is an EST database that uses a different clustering method from UniGene (Fig. 18.2). It compiles data from dbEST, GenBank mRNA and genomic DNA data, and TIGR's own sequence database. Sequences are only clustered if they are more than 95% identical for over a forty-nucleotide region in pairwise comparisons. BLAST and FASTA are used to identify sequence overlaps. In the sequence assembly stage, both TIGR Assembler (see Chapter 17) and CAP3 are used to construct contigs, producing a so-called tentative consensus (TC). To prevent chimerism, transcripts are clustered only if they match fully with known genes. Functional assignment is then given to the TC that relies most heavily on BLAST searches against protein databases. The TIGR gene indices serve as an

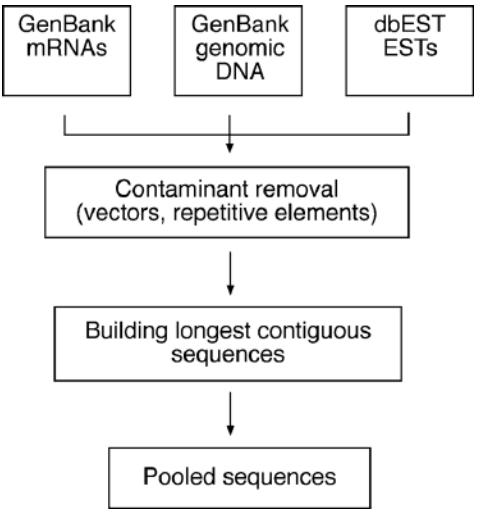


Figure 18.1: Outline of steps to process EST sequences for construction of the UniGene database.

alternative to the UniGene clusters with the resulting gene indices showing compiled EST sequences, functional annotation, and database similarity search results.

SAGE

Serial analysis of gene expression (SAGE) is another high throughput, sequence-based approach for global gene expression profile analysis. Unlike EST sampling, SAGE is more quantitative in determining mRNA expression in a cell. In this method, short fragments of DNA (usually 15 base pairs [bp]) are excised from cDNA sequences and used as unique markers of the gene transcripts. The sequence fragments are termed *tags*. They are subsequently concatenated (linked together), cloned, and sequenced. The transcript analysis is carried out computationally in a serial manner. Once gene tags are unambiguously identified, their frequency indicates the level

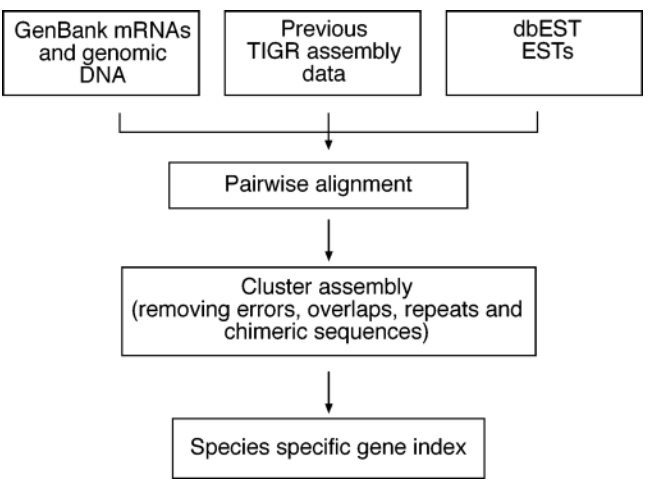


Figure 18.2: Outline of construction for TIGR gene indices.

of gene expression. This approach is much more efficient than the EST analysis in that it uses a short nucleotide tag to define a gene transcript and allows sequencing of multiple tags in a single clone. If an average clone has a size of 700 bp, it can contain up to 50 sequence tags (15 bp each), which means that the SAGE method can be at least fifty times more efficient than the brute force EST sequencing and counting. Therefore, the SAGE analysis has a better chance of detecting weakly expressed genes.

The detailed SAGE procedure (Fig. 18.3) involves the generation of short unique sequence tags (15 bp in length) by cleaving cDNA with a restriction enzyme (e.g., *Nla* III with a restriction site ↑CATG) that has a relatively high cutting frequency (*Nla* III cuts every 256 bp on average (4^4)). The *Nla* III restriction digestion produces a 4-bp overhang, which is complementary to that of a premade linker. The cleaved cDNA is divided into two pools that are ligated to different linkers, which have complementary 4-bp overhangs. The unique linker contains a restriction site for a "reach and grab" type of enzyme that cuts outside its recognition site by a specific number of base pairs downstream. For example, *Bsm*F I has a restriction site GGGAC(N_{10})↑ for the forward strand and ↑(N_{14})GTCCC for the reverse strand. When the linker with *Nla* III sticky ends is allowed to ligate with *Nla* III–treated cDNA, this creates the fusion product of linker and cDNA. This is then subject to *Bsm*F I digestion, which generates a digested product with a staggered end. The product is "blunt ended" by T4 DNA polymerase, which fills in the overhang to produce the 11-bp sequence downstream of the *Nla* III site (labeled with *X*s or *Y*s in Fig. 18.3). This sample is then allowed to ligate to the other pool of cDNA ligated to a different linker to produce a linked sequence "ditag." The linkers and the ditag are amplified using polymerase chain reaction (PCR) with primers specific to each linker. The linker sequences are then removed using *Nla* III. The ditag with sticky ends is then allowed to be concatenated with more ditags to form long serial molecules that can be cloned and sequenced. When a large number of clones with linked tags are sequenced, the frequency of occurrence of each tag is counted to obtain an accurate picture of gene expression patterns.

In a SAGE experiment, sequencing is the most costly and time-consuming step. It is difficult to know how many tags need to be sequenced to get a good coverage of the entire transcriptome. It is generally determined on a case-by-case basis. As a rule of thumb, 10,000 clones representing approximately 500,000 tags from each sample are sequenced. The scale and cost of the sequencing required for SAGE analysis are prohibitive for most laboratories. Only large sequencing centers can afford to carry out SAGE analysis routinely.

Another obvious drawback with this approach is the sensitivity to sequencing errors owing to the small size of oligonucleotide tags for transcript representation. One or two sequencing errors in the tag sequence can lead to ambiguous or erroneous tag identification. Another fundamental problem with SAGE is that a correctly sequenced SAGE tag sometimes may correspond to several genes or no gene at all. To improve the sensitivity and specificity of SAGE detection, the lengths of the tags need to be increased for the technique. The following list contains some comprehensive software tools for SAGE analysis.

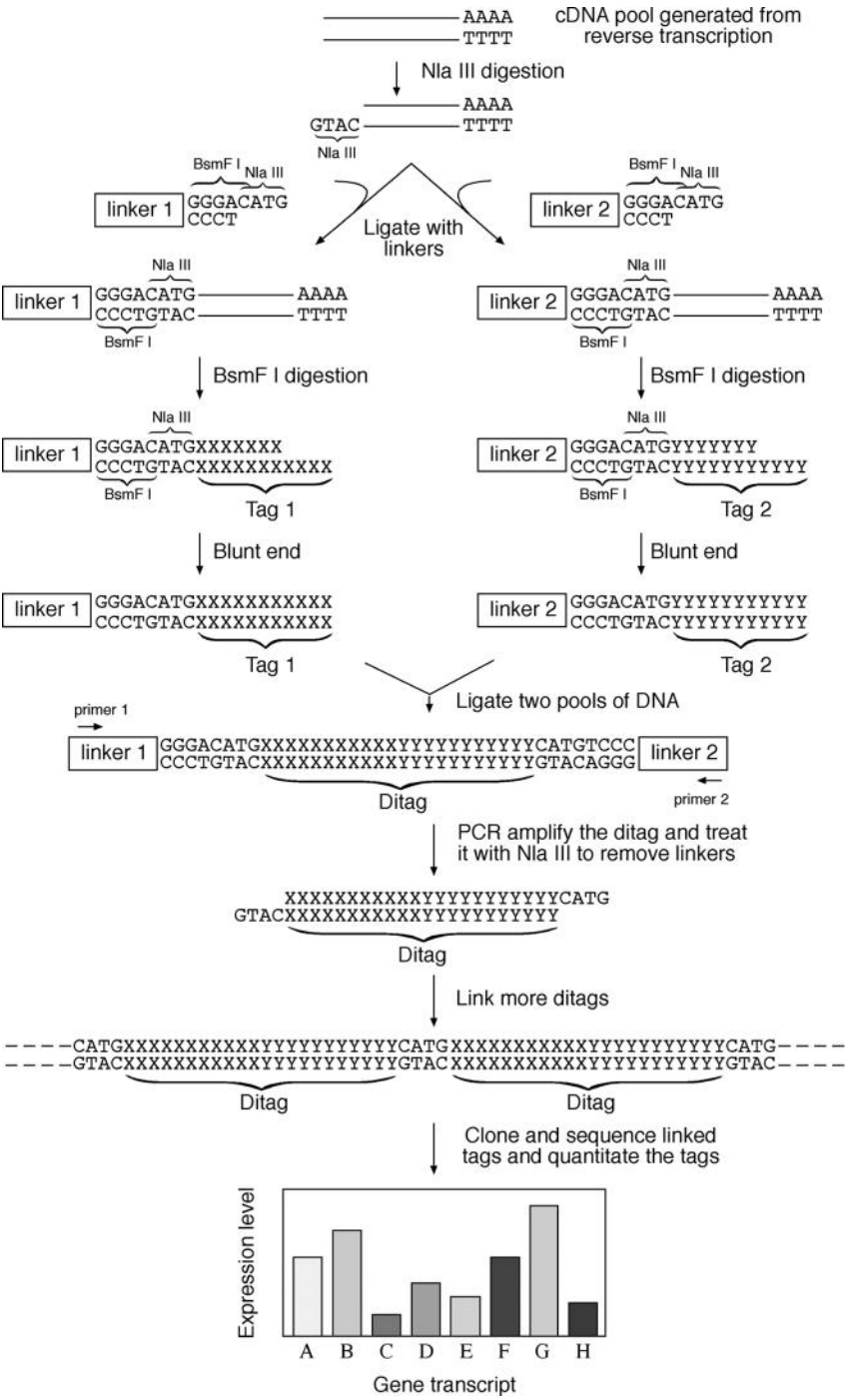


Figure 18.3: Outline of the SAGE experimental procedure.

SAGEmap (www.ncbi.nlm.nih.gov/SAGE/) is a SAGE database created by NCBI. Given a cDNA sequence, one can search SAGE libraries for possible SAGE tags and perform "virtual" Northern blots that indicate the relative abundance of a tag in a SAGE library. Each output is hyperlinked to a particular UniGene entry with sequence annotation.

SAGE xProfiler (www.ncbi.nlm.nih.gov/SAGE/sagexpsetup.cgi) is a web-based program that allows a "virtual subtraction" of an expression profile of one library (e.g., normal tissue) from another (e.g., diseased tissue). Comparison of the two libraries can provide information about overexpressed or silenced genes in normal versus diseased tissues.

SAGE Genie (http://cgap.nci.nih.gov/SAGE) is another NCBI web-based program that allows matching of experimentally obtained SAGE tags to known genes. It provides an interface for visualizing human gene expression. It has a filtering function that filters out linker sequences from experimentally obtained SAGE tags and allows expression pattern comparison between normal and diseased human tissues. The data output can be presented using subprograms such as the Anatomic Viewer, Digital Northern, and Digital Gene Expression Display.

MICROARRAY-BASED APPROACHES

The most commonly used global gene expression profiling method in current genomics research is the DNA microarray-based approach. A microarray (or gene chip) is a slide attached with a high-density array of immobilized DNA oligomers (sometimes cDNAs) representing the entire genome of the species under study. Each oligomer is spotted on the slide and serves as a probe for binding to a unique, complementary cDNA. The entire cDNA population, labeled with fluorescent dyes or radioisotopes, is allowed to hybridize with the oligo probes on the chip. The amount of fluorescent or radiolabels at each spot position reflects the amount of corresponding mRNA in the cell. Using this analysis, patterns of global gene expression in a cell can be examined. Sets of genes involved in the same regulatory or metabolic pathways can potentially be identified.

A typical DNA microarray experiment involves a multistep procedure: fabrication of microarrays by fixing properly designed oligonucleotides representing specific genes; hybridization of cDNA populations onto the microarray; scanning hybridization signals and image analysis; transformation and normalization of data; and analyzing data to identify differentially expressed genes as well as sets of genes that are coregulated (Fig. 18.4).

Oligonucleotide Design

DNA microarrays are generated by fixing oligonucleotides onto a solid support such as a glass slide using a robotic device. The oligonucleotide array slide represents thousands of preselected genes from an organism. The length of oligonucleotides is typically in the range of twenty-five to seventy bases long. The oligonucleotides are

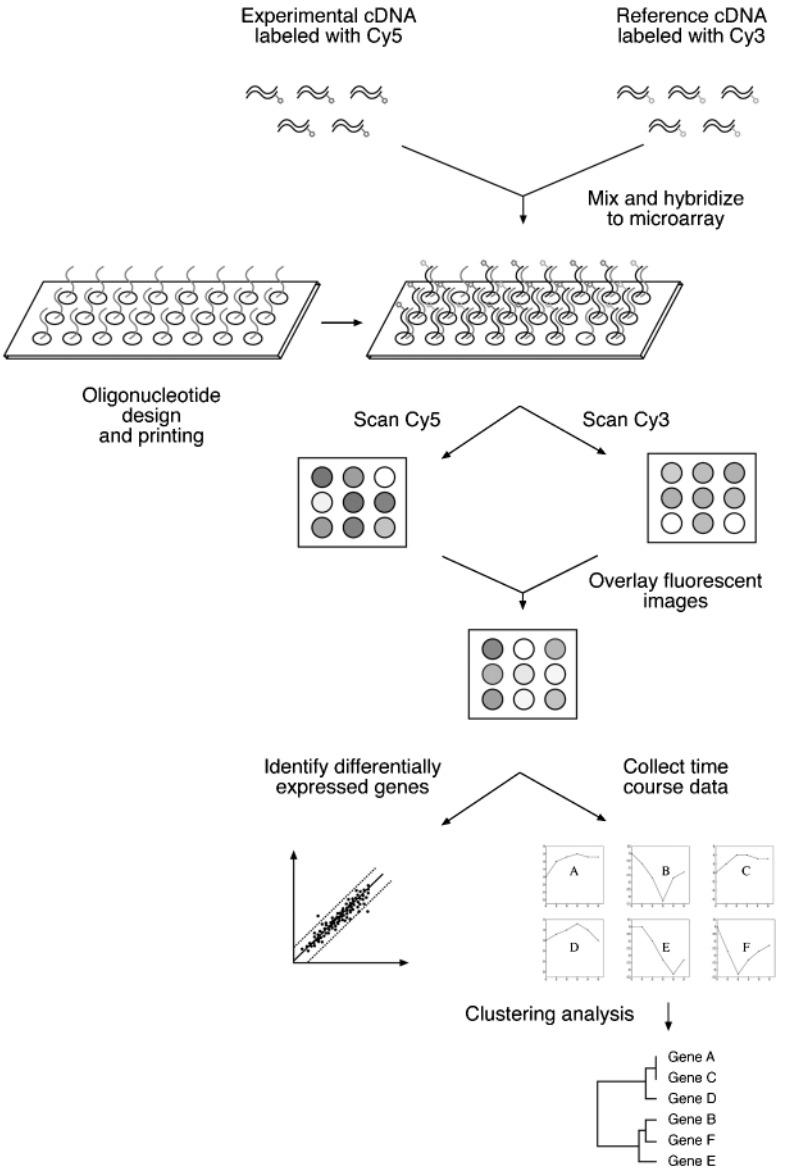


Figure 18.4: Schematic of a multistep procedure of a DNA microarray assay experiment and subsequent data analysis (see color plate section).

called probes that hybridize to labeled cDNA samples. Shorter oligo probes tend to be more specific in hybridization because they are better at discriminating perfect complementary sequences from sequences containing mismatches. However, longer oligos can be more sensitive in binding cDNAs. Sometimes, multiple distinct oligonucleotide probes hybridizing different regions of the same transcript can be used to increase the signal-to-noise ratio. To design optimal oligonucleotide sequences for microarrays, the following criteria are used.

The probes should be specific enough to minimize cross-hybridization with nonspecific genes. This requires BLAST searches against genome databases to find sequence regions with least sequence similarity with nontarget genes. The probes should be sensitive and devoid of low-complexity regions (a string of identical nucleotides; see Chapter 4). The filtering program RepeatMasker (see Chapter 4) is often used in the BLAST search. The oligonucleotide sequences should not form stable internal secondary structures, such as a hairpin structure, which could interfere with the hybridization reaction. DNA/RNA folding programs such as Mfold can help to detect secondary structures. The oligo design should be close to the 3′ end of the gene because the cDNA collection is often biased to the 3′ end. In addition, for operational convenience, all the probes should have an approximately equal melting temperature (T_m) and a GC content of 45% to 65%. A number of programs have been developed that use these rules in designing probe sequences for microarrays spotting.

OligoWiz (www.cbs.dtu.dk/services/OligoWiz/) is a Java program that runs locally but allows the user to connect to the server to perform analysis via a graphic user interface. It designs oligonucleotides by incorporating multiple criteria including homology, T_m, low complexity, and relative position within a transcript.

OligoArray (http://berry.engin.umich.edu/oligoarray2/) is also a Java client-server program that computes oligonucleotides for microarray construction. It uses the normal criteria with an emphasis on gene specificity and secondary structure for oligonucleotides. The secondary structures and related thermodynamic parameters are calculated using Mfold.

Data Collection

The expression of genes is measured via the signals from cDNAs hybridizing with the specific oligonucleotide probes on the microarray. The cDNAs are obtained by extracting total RNA or mRNA from tissues or cells and incorporating fluorescent dyes in the DNA strands during the cDNA biosynthesis. The most common type of microarray protocol is the two-color microarray, which involves labeling one set of cDNA from an experimental condition with one dye (Cy5, red fluorescence) and another set of cDNA from a reference condition (the controls) with another dye (Cy3, green fluorescence). When the two differently labeled cDNA samples are mixed in equal quantity and allowed to hybridize with the DNA probes on the chips, gene expression patterns of both samples can be measured simultaneously.

The image of the hybridized array is captured using a laser scanner that scans every spot on the microarray. Two wavelengths of the laser beam are used to excite the red and green fluorescent dyes to produce red and green fluorescence, which is detected using a photomultiplier tube. Thus, for each spot on the microarray, red and green fluorescence signals are recorded. The two fluorescence images from the scanner are then overlaid to create a composite image, which indicates the relative expression levels of each gene. Thus, the measurement from the composite image reflects the ratio of the two color intensities. If a gene is expressed at a higher level in the experimental condition (red) than in the control (green), the spot displays

a reddish color. If the gene is expressed at a lower level than the control, the spot appears greenish. Unchanged gene expression, having equal amount of green and red fluorescence, results in a yellow spot. The colored image is stored as a computer file (in TIFF format) for further processing.

Image Processing

Image processing is to locate and quantitate hybridization spots and to separate true hybridization signals from background noise. The background noise and artifacts produced in this step include nonspecific hybridization, unevenness of the slide surface, and the presence of contaminants such as dust on the surface of the slide. In addition, there are also geometric variations of hybridization spots resulting in some spots being of irregular shapes. Computer programs are used to correctly locate the boundaries of the spots and measure the intensities of the spot images after subtracting the background pixels.

After subtracting the background noise, the array signals are converted into numbers and reported as ratios between Cy5 and Cy3 for each spot. This ratio represents relative expression changes and reflects the fold change in mRNA quantity in experimental versus control conditions. The data are often presented as false colors of different intensities of red and green colors depending on whether the ratios are above 1 or below 1, respectively. Where there is an equal quantity of experimental and control mRNA (yellow in raw data), black is shown. The false color images are presented in squares in a matrix of genes versus conditions so that differentially expressed genes can be more easily analyzed (Box 18.1).

Manufacturers of microarray scanners normally provide software programs to specifically perform microarray image analysis. There are, however, also a small number of free image-processing software programs available on the Internet.

ArrayDB (http://genome.nhgri.nih.gov/arraydb/) is a web interface program that allows the user to upload data for graphical viewing. The user can present histograms, select actual microarray slide images, and display detailed information of each spot which is linked to functional annotation of the corresponding gene in the UniGene, Entrez, dbEST, and KEGG databases. This can help to provide a synopsis of gene function when interpreting the microarray data.

ScanAlyze (http://rana.lbl.gov/EisenSoftware.htm) is a Windows program for microarray fluorescent image analysis. It features semiautomatic spot definition and multichannel pixel and spot analyses.

TIGR Spotfinder (http://www.tigr.org/softlab/) is another Windows program for microarray image processing using the TIFF image format. It uses an adaptive threshold algorithm, which resolves the boundaries of spots according to their shapes. The algorithm determines the intensity of irregular spots more accurately than most other similar programs. It also interfaces with a gene expression database.

Data Transformation and Normalization

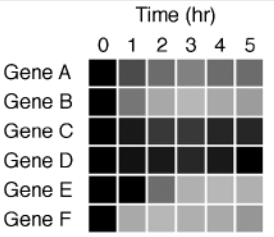
Following image processing, the digitized gene expression data need to be further processed before differentially expressed genes can be identified. This processing is

Box 18.1 Outline of the Procedure for Microarray Data Analysis

	0 hr	1 hr	2 hr	3 hr	4 hr	5 hr
Gene A	1	4	6	8	6	6
Gene B	1	0.6	0.3	0.1	0.3	0.4
Gene C	1	2	4	4	3	3
Gene D	1	1.5	2	3	2	1
Gene E	1	1	0.5	0.2	0.1	0.2
Gene F	1	0.3	0.1	0.2	0.3	0.4

convert to false colors

$\log_2$ conversion



	Gene B	Gene C	Gene D	Gene E	Gene F
Gene A	-0.82	0.96	0.65	-0.68	-0.79
Gene B		-0.85	-0.86	0.66	0.67
Gene C			0.70	-0.65	-0.87
Gene D				-0.41	-0.72
Gene E					0.26

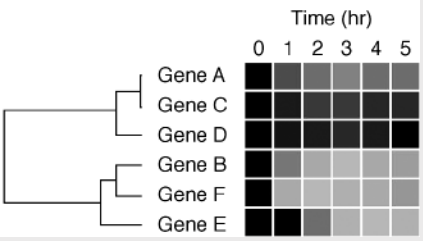
calculating Pearson correlation coefficients between genes

	0 hr	1 hr	2 hr	3 hr	4 hr	5 hr
Gene A	0	2	2.6	3	2.6	2.6
Gene B	0	-0.7	-1.7	-3.3	-1.7	-1.3
Gene C	0	1	2	2	1.6	1.6
Gene D	0	0.6	1	1.6	1	0
Gene E	0	0	-1	-2.3	-3.3	-2.3
Gene F	0	-1.7	-3.3	-2.3	-1.7	-1.3

conversion of coefficients to positive distance values

	Gene B	Gene C	Gene D	Gene E	Gene F
Gene A	1.82	0.04	0.35	1.68	1.79
Gene B		1.85	1.86	0.34	0.33
Gene C			0.30	1.65	1.87
Gene D				1.41	1.72
Gene E					0.74

hierarchical clustering



The example involves the use of six hypothetic genes whose expression is measured over a time course of 5 hours. The microarray raw data in the form of Cy5/Cy3 ratios are converted to false colors image in red, green and black. The data matrix is subjected to logarithmic transformation. The distances between genes are calculated using Pearson correlation coefficients. After conversion to a positive distance matrix, further classification analysis using the hierarchical clustering approach produces a tree showing the relationships of coexpressed genes (see color plate section).

referred to as *data normalization* and is designed to correct bias owing to variations in microarray data collection rather than intrinsic biological differences.

When the raw fluorescence intensity Cy5 is plotted against Cy3, most of the data are clustered near the bottom left of the plot, showing a non-normal distribution of the raw data (Fig. 18.5A). This is thought to be a result of the imbalance of red and green intensities during spot sampling, resulting in ineffective discrimination of differentially expressed genes. One way to improve the data discrimination is to transform

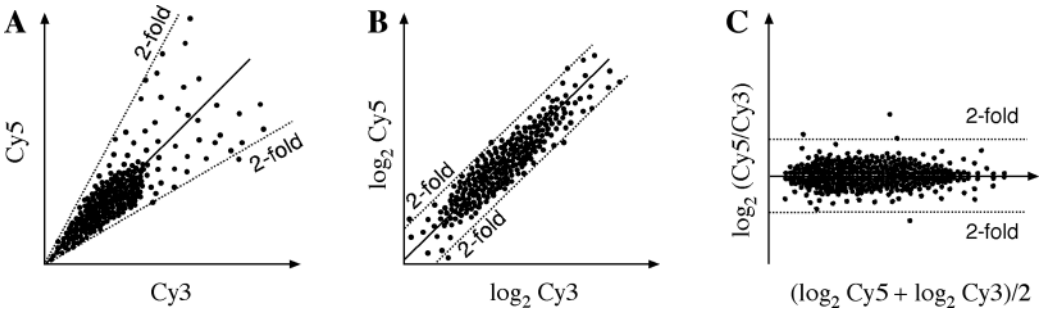


Figure 18.5: Scatter plot of gene expression analysis showing the process of data normalization. The solid line indicates linear regression of the data points; dashed lines show the cutoff for a twofold change in expression. **(A)** Plot of raw fluorescence signal intensities of Cy5 versus Cy3. **(B)** Plot of the same data after log transformation to the base of 2. **(C)** Plot of mean log intensity versus log ratio of the two fluorescence intensities, which shifts the data points to around the horizontal axis, making them easier to visualize.

raw Cy5 and Cy3 values by taking the logarithm to the base of 2. The transformation produces a more uniform distribution of data and has the advantage to display upregulated and downregulated genes more symetrically. As shown in Figure 18.5B, the data become more evenly distributed within a certain range, and assume a normal distribution pattern. By taking this transformation, the data for up-regulation and down-regulation can be more comparable.

There are many ways to further normalize the data. One way is to plot the data points horizontally. This requires plotting the log ratios (Cy5/Cy3) against the average log intensities (Fig. 18.5C). In this representation, the data are roughly symmetrically distributed about the horizontal axis. The differentially expressed genes can then be more easily visualized. This form of representation is also called *intensity-ratio plot*. In all these instances, linear regression is used.

Sometimes, the data do not conform to a linear relationship owing to systematic sampling errors. In this case, a nonlinear regression may produce a better fitting and help to eliminate the bias. The most frequently used regression type is known as Lowess (*locally weighted scatter plot smoother*) regression. This method performs a locally weighted linear fitting of the intensity-ratio data and calculates the differences between the curve-fitted values and experimental values. The algorithm further "corrects" the experimental data points by depressing large difference values more than small ones with respect to a reference. As a result, a new distribution of intensity-ratio data that conforms a linear relationship can be produced. After normalization of the data, the true outliers, which represent genes that are significantly up-regulated or down-regulated, can be more easily identified. The following two software programs that are freely available are specialized in image analysis and data normalization.

Arrayplot (www.biologie.ens.fr/fr/genetiqu/puces/publications/arrayplot/index.html) is a Windows program that allows visualization, filtering, and normalization of raw microarray data. It has an interface to view significantly up-regulated or down-regulated genes. It calculates normalization factors based on the overall median signal intensity.

SNOMAD (http://pevsnerlab.kennedykrieger.org/snomadinput.html) is a web server for microarray data normalization. It provides scatter plots based on raw signal intensities and performs log-transformation and linear regression as well as Lowess regression analysis of the data.

Statistical Analysis to Identify Differentially Expressed Genes

To separate genes that are differentially expressed, many published studies use a normalization cutoff of twofold as a criterion. However, this is an arbitrary cutoff value, which could be considered to be either too high or too low depending on the data variability. In addition, the inherent data variability is not taken into account. A data point above or below the cutoff line could simply be there by chance or because of error. The only way to ensure that a gene that appears to be differentially expressed is truly differentially expressed is to perform multiple replicate experiments and to perform statistical testing. The repeat experiments provide replicate data points that offer information about the variability of the expression data at a particular condition. The information on the distribution for the data points under particular conditions can help answer the question whether a given fold difference is significant. The main hindrance to obtaining multiple replicate datasets is often the cost: microarray experiments are extremely expensive for regular research laboratories.

If replicated datasets are available, rigorous statistical tests such as *t*-test and analysis of variance (ANOVA) can be performed to test the null hypothesis that a given data point is not significantly different from the mean of the data distribution. For such tests, it is common to use a *P*-value cutoff of .05, which means a confidence level of 95% to distinguish the data groups. This level also corresponds to a gene expression level with two standard deviations from the mean of distribution. It is noticeable that the number of standard deviations is only meaningful if the data are approximately normally distributed, which makes the previous normalization step more valuable.

MA-ANOVA (www.jax.org/staff/churchill/labsite/software/anova/) is a statistical program for Windows and UNIX that uses ANOVA to analyze microarray data. It calculates log ratios, displays ratio-intensity plots, and performs permutation tests and bootstrapping of confidence values.

Cyber-T (http://visitor.ics.uci.edu/genex/cybert/) is a web server that performs *t*-tests on observed changes of replicate gene expression measurements to identify significantly differentially expressed genes. It also contains a computational method for estimating false-positive and false-negative levels in experimental data based on modeling of *P*-value distributions.

Microarray Data Classification

One of the key features of DNA microarray analysis is to study the expression of many genes in parallel and identify groups of genes that exhibit similar expression patterns. The similar expression patterns are often a result of the fact that the genes involved

are in the same metabolic pathway and have similar functions. The genetic basis of the coregulation could be the result of common promoters and regulatory regions.

To discover genes with similar gene expression patterns based on the microarray data requires partitioning the data into subsets according to similarity. To achieve this goal, hybridization signals from microarray images are organized into matrices where rows represent genes and columns represent experimental sampling conditions (such as time points or drug concentrations). Each matrix value is the Cy5/Cy3 intensity ratio representing the relative expression of a gene under a specific condition (see Box 18.1). Various classification tools are subsequently used to classify the values in the matrices for gene expression comparison.

Distance Measure

The first step towards gene classification is to define a measure of the distance or dissimilarity between genes. This requires converting a gene expression matrix in a distance matrix. The distance can be expressed as Euclidean distance or Pearson correlation coefficient. *Euclidean distance* is the square root of the sum of squared distances between expression data points. When comparing X gene expression with Y gene expression at time point i (assuming there are n time points in total), the distance score (d) can be calculated by the following formula:

$$d = \sqrt{\sum_{\mathrm{i}=1}^{\mathrm{n}} (x_{\mathrm{i}} - y_{\mathrm{i}})^2} \qquad \text{(Eq. 18.1)}$$

Euclidean distances are widely used but suffer from the problem that when variations between genes are very small, the gene profiles can be very difficult to differentiate.

Alternatively, a Pearson correlation coefficient between two groups of data points can be used. This measures the overall similarity between the trends or shapes of the two sets of data. In this measure, a perfect positive correlation is $+1$ and a perfect negative correlation is -1. The distance score (d) between gene X and gene Y can be calculated using the following formula:

$$d = \frac{1}{n} \sum_{\mathrm{i}=1}^{\mathrm{n}} \left(\frac{x_{\mathrm{i}} - \bar{x}}{sd_{\mathrm{i}}} \right) \left(\frac{y_{\mathrm{i}} - \bar{y}_{\mathrm{i}}}{sd_{\mathrm{i}}} \right) \qquad \text{(Eq. 18.2)}$$

where n is the total number of time points; $\bar{\mathrm{x}}$ and $\bar{\mathrm{y}}$ are average values for the X gene and Y gene data, respectively; and sd are standard deviation values.

The choice of the distance measures can sometimes make a big difference in the final result. Sometimes, a small change in expression data can cause a significant change in an Euclidean distance matrix. Pearson correlation coefficients are more robust than Euclidean distances in guarding against small variations and noise in the experimental data. One notable feature of the Pearson correlation coefficients is that, when the genes to be compared have exactly the same expression patterns, their gene expression profiles have identical shapes. The correlation coefficient of the gene profiles equals to $+1$, in which case, the relative distance between the genes

is zero. When the concerned genes have absolute opposite expression patterns, the correlation coefficient becomes −1. That means that, when one gene is up-regulated, the other is down-regulated, and vice versa. In such case, the distance is converted to +2 (the absolute value of $|(-1) - 1|$), the maximum distance value in the matrix (see Box 18.1). The conversion to a positive distance value makes data classification more convenient.

Supervised and Unsupervised Classification

Based on the computed distances between genes in an expression profile, genes with similar expression patterns can be grouped. The classification analysis can be either supervised or unsupervised. A *supervised analysis* refers to classification of data into a set of predefined categories. For example, depending on the purpose of the experiment, the data can be classified into predefined "diseased" or "normal" categories. An *unsupervised analysis* does not assume predefined categories, but identifies data categories according to actual similarity patterns. The unsupervised analysis is also called *clustering*, which is to group patterns into clusters of genes with correlated profiles.

For microarray data, clustering analysis identifies coexpressed and coregulated genes. Genes within a category have more similarity in expression than genes from different categories. When genes are coregulated, they normally reflect related functionality. Through gene clustering, functions of previously uncharacterized genes may be discovered. Clustering methods include hierarchical clustering and partitioning clustering (e.g., k-means, self-organizing maps [SOMs]). The following discussion focuses on several of the most frequently used clustering methods.

The clustering algorithms can be further divided into two types, agglomerative and divisive (Fig. 18.6). An agglomerative method begins by clustering the two most similar data points and repeats the process to successively merge groups of data according to similarity until all groups of data are merged. This is also known as the *bottom-up approach*. A divisive method works the other way around by lumping all data points in a single cluster and successively dividing the data into smaller groups according to dissimilarity until all the hierarchical levels are resolved. This is also called the *top-down approach*.

Hierarchical Clustering. A hierarchical clustering method is in principle similar to the distance phylogenetic tree-building method (see Chapter 11). It produces a treelike structure that represents a hierarchy or relative relatedness of data groups. In the tree leaves, similar gene expression profiles are placed more closely together than dissimilar gene expression profiles. The tree-branching pattern illustrates a higher degree of relationship between related gene groups. When genes with similar expression profiles are grouped in such a way, functions for unknown genes can often be inferred.

Hierarchical clustering uses the agglomerative approach that works in much the same way as the UPGMA method (see Chapter 11), in which the most similar data pairs are joined first to form a cluster. The new cluster is treated as a single entity

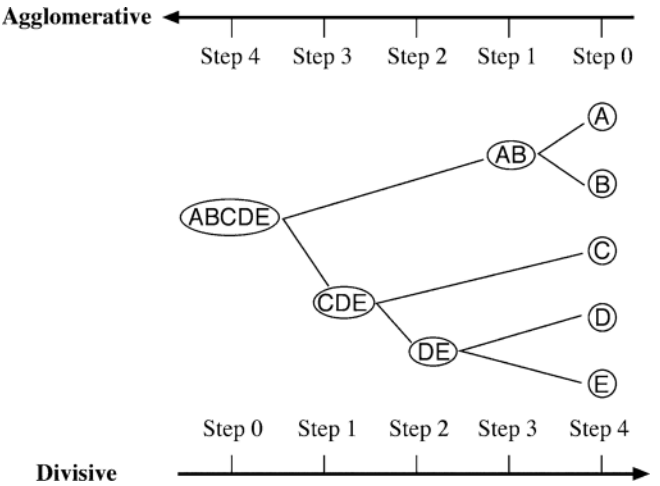


Figure 18.6: Schematic representation showing differences between agglomerative and divisive clustering methods.

creating a reduced matrix. The reduced matrix allows the next closest data point to be added to the previous cluster leading to the formation of a new cluster. By repeating the process, a dendrogram showing the clustering pattern of all data points is built.

The hierarchical clustering algorithms can be further divided in three subtypes known as single linkage, complete linkage, and average linkage. The single linkage method chooses the minimum value of a pair of distances as the cluster distance. The complete linkage method chooses the maximum value of a pair of distances, and the average linkage method chooses the mean of the two distances, which is the same as the UPGMA tree building approach. The UPGMA-based method is considered to be the most robust in discriminating expression clusters. It is important to point out that although a tree structure is produced as the final result, the resulting tree has no evolutionary meaning, but merely represents groupings of similarity patterns in gene expression.

In a tree produced by hierarchical clustering, the user has the flexibility to define a threshold for determining the boundaries of data clusters. The flexibility, however, sometimes can be a disadvantage in that it lacks objective criteria to distinguish clusters. Another potential drawback is that the hierarchical relationships of gene expression represented by the tree may not in fact exist. Some of the drawbacks can be alleviated by using alternative clustering approaches such as the k-means or self-organizing maps.

k-Means Clustering. In contrast to hierarchical clustering algorithms, k-means clustering does not produce a dendrogram, but instead classifies data through a single step partition. Thus, it is a divisive approach. In this method, data are partitioned into k-clusters, which are prespecified at the outset. The value of k is normally randomly set but can be adjusted if results are found to be unsatisfactory. In the first step, data

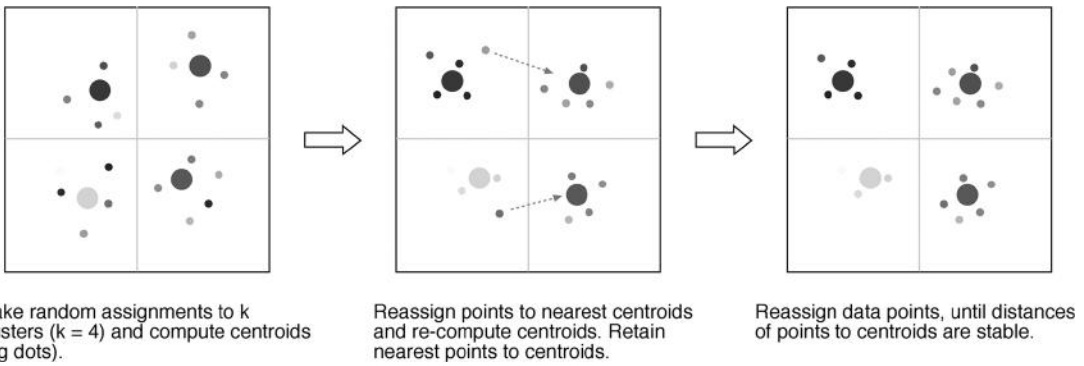


Figure 18.7: Example of k-means clustering using four partitions. Closeness of data points is indicated by resemblance of colors (see color plate section).

points are randomly assigned to each cluster. The average of the data in a group (*centroid value*) is calculated. The distance of each data point to the centroid is also calculated. The second step is to have all the data points randomly reassigned among the k-clusters. The centroid of each cluster and distances of data points to the centroid are recomputed. Then each data point is reassigned to a different cluster. If a data point is found to be closer to the centroid of a particular cluster than to any other cluster, that data point is retained in the partition. Otherwise, it is subject to reassignment in the next iteration. This process is repeated many times, until the distances between the data points and the new centroids no longer decrease. At this point, a final clustering pattern is reached (Fig. 18.7).

As described, the number of k-clusters is specified by the user at the outset, which is either chosen randomly or determined using external information. The cluster number can be adjustable, increased or decreased to get finer or coarser data distinctions. The k-means method may not be as accurate as hierarchical clustering because it has an inherent problem of being sensitive to the selection of the initial arbitrary number of clusters. Depending on the initial position of centroids, this may lead to a different partitioning solution each time when k-means is run for the same datasets. Without searching all possible initial partitions, a suboptimal solution may be reached. However, computationally speaking, it is faster than hierarchical clustering and is still widely used.

Self-Organizing Maps. Clustering by SOMs is in principle similar to the k-means method. This pattern recognition algorithm employs neural networks. It starts by defining a number of nodes. The data points are initially assigned to the nodes at random. The distance between the input data points and the centroids are calculated. The data points are successively adjusted among the nodes, and their distances to the centroids are recalculated. After many iterations, a stabilized clustering pattern are reached with the minimum distances of the data points to the centroids.

The differences between SOM and k-means are that, in SOM, the nodes are not treated as isolated entities, but as connected to other nodes. The calculation of the

centroid values in SOM takes into account not only information from within each cluster, but also information from adjacent clusters. This allows the analysis to be better at handling noisy data. Another difference is that, in SOM, some nodes are allowed to contain no data at all. Thus, at the completion of the clustering, the final number of clusters may be smaller than the initial nodes. This feature renders SOM less subjective than k-means. However, this type of algorithm is also much slower than the k-means method.

Clustering Programs. Cluster (http://rana.lbl.gov/EisenSoftware.htm) is a Windows program capable of hierarchical clustering, SOM, and k-means clustering. Outputs from hierarchical clustering are visualized with the Treeview program.

EPCLUST (www.ebi.ac.uk/EP/EPCLIST) is a web-based server that allows data to be uploaded and clustered with hierarchical clustering or k-means methods. In addition, the user can perform data selection, normalization, and database similarity searches with this program.

TIGR TM4 (www.tigr.org/tm4) is a suite of multiplatform programs for analyzing microarray data. This comprehensive package includes four interlinked programs, TIGR spot finder (for image analysis), MIDAS (for data normalization), MeV (for clustering analysis and visualization), and MADAM (for data management). The package provides different data normalization schemes and clustering options.

SOTA (Self-Organizing Tree Algorithm; www.almabioinfo.com/sota/) is a web server that uses a hybrid approach of SOM and hierarchical clustering. It builds a tree based on the divisive approach starting from the root node containing all data patterns. Instead of using the distance-based criteria to resolve a tree, the algorithm using the neural network based SOM algorithm to separate clusters of genes at each node. The homogeneity of gene clusters at each node is analyzed using SOM. The tree building stops at any point if desired homogeneity level is reached.

COMPARISON OF SAGE AND DNA MICROARRAYS

SAGE and DNA microarrays are both high throughput techniques that determine global mRNA expression levels. A number of comparative studies have indicated that the gene expression measurements from these methods are largely consistent with each other. However, the two techniques have important differences. First, SAGE does not require prior knowledge of the transcript sequence, whereas DNA microarray experiments can only detect the genes spotted on the microarray. Because SAGE is able to measure all the mRNA expressed in a sample, it has the potential to allow discovery of new, yet unknown gene transcripts. Second, SAGE measures "absolute" mRNA expression levels without arbitrary reference standards, whereas DNA microarrays indicate the relative expression levels. Therefore, SAGE expression data are more comparable across experimental conditions and platforms. This makes public SAGE databases more informative by allowing comparison of data from reference conditions with various experimental treatments. Third, the PCR amplification step involved in

the SAGE procedure means that it requires only a minute quantity of sample mRNA. This compares favorably to the requirement for a much larger quantity of mRNA for microarray experiments, which may be impossible to obtain under certain circumstances. Fourth, collecting a SAGE library is very labor intensive and expensive compared with carrying out a DNA microarray experiment, however. Therefore, SAGE is not suitable for rapid screening of cells whereas the microarray analysis is. Fifth, Gene identification from SAGE data is also more cumbersome because the mRNA tags have to be extracted, compiled, and identified computationally, whereas in DNA microarrays, the identities of the probes are already known. In SAGE, comparison of gene expression profiles to discover differentially expressed genes and coexpressed genes is performed manually, whereas for microarrays, there are a large number of software algorithms to automate the process.

SUMMARY

Transcriptome analysis using ESTs, SAGE, and DNA microarrays forms the core of functional genomics and is key to understanding the interactions of genes and their regulation at the whole-genome level. EST sampling, although widely used, has a number of drawbacks in terms of error rates, efficiency, and cost. The high throughput SAGE and DNA microarray approaches provide a more quantitative measure of global gene expression. SAGE measures the "absolute" mRNA expression levels, whereas microarrays indicate relative mRNA expression levels. DNA microarrays currently enjoy greater popularity because of the relative ease of experimentation. It is also a more suitable method to probe differential gene expression between different tissue and cell samples. This requires comparing gene profiles using statistical approaches. Another goal of microarray analysis is to identify coordinated gene expression patterns, which requires clustering analysis of microarray data.

The most popular microarray data clustering techniques include hierarchical clustering, SOM, and k-means. The hierarchical approach is very similar to the phylogenetic distance tree building method. SOM and k-means normally do not generate a treelike structure as a result of clustering. Once coregulated genes are identified, upstream sequences belonging to a cluster can be retrieved and analyzed for common regulatory sequences.

In conclusion, among the three techniques for studying global gene expression, the most popular one is DNA microarrays, which has the capability to provide information that is not possible with traditional techniques. However, one should also be aware of its limitations. This technique is a multistep procedure in which errors and biases can be introduced in each step (scanning, image processing, normalization, and choice of classification method). Thus, it is a rather crude assay and may contain considerable levels of false positives and false negatives. The results from microarray analysis only provide hypotheses for gene functions based on classification of expression data. To verify the hypotheses, one has to rely on traditional biochemical and molecular biological approaches. The fundamental limitation of this method lies in the use of

transcription as the sole indicator of gene expression, which may or may not correlate with expression at the protein level. The expression of proteins is what dictates the phenotypes. The last limitation is addressed in Chapter 19.

FURTHER READING

Causton, H. C., Quackenbush, J., and Brazma, A. 2003. *Microarray Gene Expression Data Analysis: A Beginner's Guide*. Malden, MA: Blackwell.

Forster, T., Roy, D., and Ghazal, P. 2003. Experiments using microarray technology: Limitations and standard operating procedure. *J. Endocrinol.* 178:195–204.

Gill, R. W., and Sanseau, P. 2000. Rapid in silico cloning of genes using expressed sequence tags (ESTs). *Biotechnol. Annu. Rev*. 5:25–44.

Quackenbush, J. 2002. Microarray data normalization and transformation. *Nat Genet.* 32(Suppl): 496–501.

Scott, H. S., and Chrast, R. 2001. Global transcript expression profiling by Serial Analysis of Gene Expression (SAGE). *Genet. Eng.* 23:201–19.

Slonim, D. K. 2002. From patterns to pathways: Gene expression data analysis comes of age. *Nat. Genet.* 32(Suppl):502–8.

Stanton, L. W. 2001. Methods to profile gene expression. *Trends Cardiovasc. Med.* 11:49–54.

Stekel, D. 2003. *Microarray Bioinformatics*. Cambridge, UK: Cambridge University Press.

Ye, S. Q., Usher, D. C., and Zhang, L. Q. 2002. Gene expression profiling of human diseases by serial analysis of gene expression. *J. Biomed. Sci.* 9:384–94.

CHAPTER NINETEEN

Proteomics

Proteome refers to the entire set of expressed proteins in a cell. In other words, it is the full complement of translated product of a genome. *Proteomics* is simply the study of the proteome. More specifically, it involves simultaneous analyses of all translated proteins in a cell. It encompasses a range of activities including large-scale identification and quantification of proteins and determination of their localization, modifications, interactions, and functions. This chapter covers the major topics in proteomics such as analysis of protein expression, posttranslational modifications, protein sorting, and protein–protein interaction with an emphasis on bioinformatics applications.

Compared to transcriptional profiling in functional genomics, proteomics has clear advantages in elucidating gene functions. It provides a more direct approach to understanding cellular functions because most of the gene functions are realized by proteins. Transcriptome analysis alone does not provide clear answers to cellular functions because there is generally not a one-to-one correlation between messenger RNAs (mRNAs) and proteins in the cells. In addition, a gene in an eukaryotic genome may produce more varied translational products owing to alternative splicing, RNA editing, and so on. This means that multiple and distinct proteins may be produced from one single gene. Further complexities of protein functions can be found in posttranslational modifications, protein targeting, and protein–protein interactions. Therefore, the noncorrelation of mRNA with proteins means that studying protein expression can provide more insight on understanding of gene functions.

TECHNOLOGY OF PROTEIN EXPRESSION ANALYSIS

Characterization of protein expression at the whole proteome level involves quantitative measurement of proteins in a cell at a particular metabolic state. Unlike in DNA microarray analysis, in which the identities of the probes are known beforehand, the identities of the expressed proteins in a proteome have to determined by performing protein separation, identification, quantification, and identification procedures. The classic protein separation methods involve two-dimensional gel electrophoresis followed by gel image analysis. Further characterization involves determination of amino acid composition, peptide mass fingerprints, and sequences using mass spectrometry (MS). Finally, database searching is needed for protein identification. The outline of the procedure is shown in Figure 19.1.

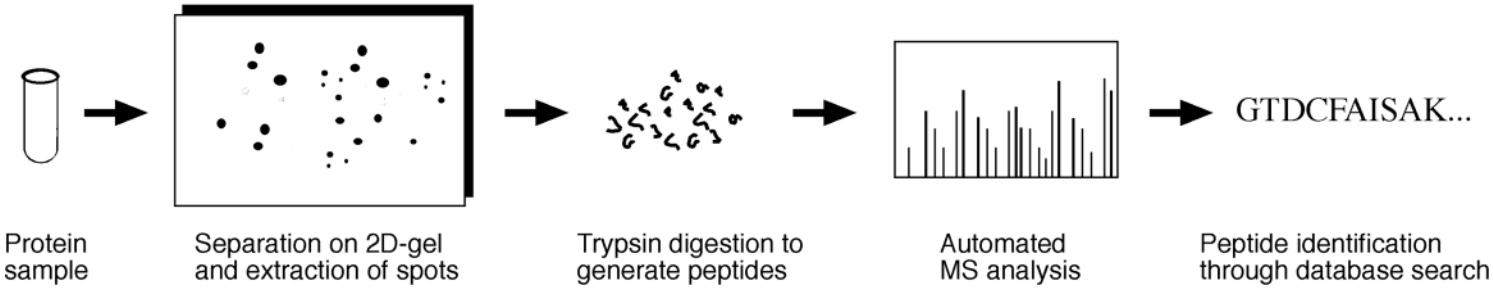


Figure 19.1: Overview of the procedure for proteome characterization using two-dimensional gel and MS.

2D-Page

Two-dimensional polyacrylamide gel electrophoresis (2D-PAGE) is a high-resolution technique that separates proteins by charge and mass. The gel is run in one direction in a pH gradient under a nondenaturing condition to separate proteins by isoelectric points (pI) and then in an orthogonal dimension under a denaturing condition to separate proteins by molecular weights (MW). This is followed by staining, usually silver staining, which is very sensitive, to reveal the position of all proteins. The result is a two-dimensional gel map; each spot on the map corresponds to a single protein being expressed. The stained gel can be further scanned and digitized for image analysis.

However, not all proteins can be separated by this method or stained properly. One of the challenges of this technique is the separation of membrane proteins, which are largely hydrophobic and not readily solublized. They tend to aggregate in the aqueous medium of a two-dimensional gel. To overcome this problem, membrane proteins can be fractionated using specialized protocols and then electrophoresed using optimized buffers containing zwitterionic detergents. Subfractionation can be carried out to separate nuclear, cytosol, cytoskeletal, and other subcellular fractions to boost the concentrations of rare proteins and to reveal subcellular localizations of the proteins.

Gel image analysis is the next step that helps to reveal differential global protein expression patterns. This analysis includes spot determination, quantitation, and normalization. Image analysis software is used to measure the center, edges, and densities of the spots. Comparing two-dimensional gel images from various experiments can sometimes pose a challenge because the gels, unlike DNA microarrays, may shrink or warp. This requires the software programs to be able to stretch or maneuver one of the gels relative to the other to find a common geometry. When the reference spots are aligned properly, the rest of the spots can be subsequently compared automatically. There are a number of web-based tools available for this type of image analysis.

Melanie (http://us.expasy.org/melanie/) is a commercially available comprehensive software package for Windows. It carries out background subtraction, spot detection, quantitation, annotation, image manipulation and merging, and linking to 2D-PAGE databases as well as image comparison through statistical tests.

CAROL (http://gelmatching.inf.fu-berlin.de/Carol.html) is a free Java program for two-dimensional gel matching, which takes into account geometrical distortions of gel spots.

Comp2Dgel (www2.imtech.res.in/raghava/comp2dgel/) is a web server that allows the user to compare two-dimensional gel images with a two-dimensional gel database or with other gels that the user inputs. A percentage deviation of the images is obtained through superimposition of the images.

SWISS-2DPAGE (www.expasy.ch/) is a database of two-dimensional gel maps of cells of many organisms at metabolic resting conditions (control conditions), which can be used for comparison with experimental or diseased conditions. It can be searched by a spot identifier or keyword.

Mass Spectrometry Protein Identification

Once the proteins are separated on a two-dimensional gel, they can be further identified and characterized using MS. In this procedure, the proteins from a two-dimensional gel system are first digested in situ with a protease (e.g., trypsin). Protein spots of interest are excised from the two-dimensional gel. The proteolysis generates a unique pattern of peptide fragments of various MWs, which is termed a peptide fingerprint. The fragments can be analyzed with MS, a high-resolution technique for determining molecular masses. Currently, electrospray ionization MS and matrix-assisted laser desorption ionization (MALDI) MS are commonly used. These two approaches only differ in the ionization procedure used. In MALDI-MS, for example, the peptides are charged with positive ions and forced through an analyzing tube with a magnetic field. Peptides are analyzed in the gas phase. Because smaller peptides are deflected more than larger ones in a magnetic field, the peptide fragments can be separated according to molecular mass and charges. A detector generates a spectrum that displays ion intensity as a function of the mass-to-charge ratio.

As a step toward further identification, the peptides can be sequenced with successive phases of fragmentation and mass analysis. This is the technique of tandem mass spectrometry (MS/MS), in which a peptide has to pass through two analyzers for sequence determination. In the first analyzer, the peptide is fragmented by physical means generating fragments with nested sizes differing by only one amino acid. The molecular masses of these fragments are more precisely determined in the second analyzer yielding the sequence of the fragment.

Protein Identification through Database Searching

MS characterization of proteins is highly dependent on bioinformatic analysis. Once the peptide mass fingerprints or peptide sequences are determined, bioinformatics programs can be used to search for the identity of a protein in a database of theoretically digested proteins. The purpose of the database search is to find exact or nearly exact matches. However, in reality, protease digestion is rarely perfect, often generating partially digested products as a result of missed cuts at expected cutting sites. Peptides resulting from MALDI-MS are also charged, which increases their mass slightly. To increase the discriminatory ability of the database search, the search engine must allow some leeway in matching molecular masses of peptides in the cases of missed cuts and charge modifications. The user is required to provide as much information as possible as input. For example, molecular masses of peptide fingerprints, peptide sequence, MW, and pI of the intact protein, even the species names are important in obtaining unique identification of a particular protein. A basic requirement for peptide identification through database matching is the availability of all the protein sequences from an organism. Thus, this method only works well with model organisms that have completely sequenced and well-annotated genomes, but has much limitation to be applied in nonmodel organisms.

ExPASY (www.expasy.ch/tools/) is a comprehensive proteomics web server with a suite of programs for searching peptide information from the SWISS-PROT and TrEMBL databases. There are twelve database search tools in this server dedicated to protein identification based on MS data. For example, the AACompIdent program identifies proteins based on pI, MW, and amino acid composition and compares these values with theoretical compositions of all proteins in SWISS-PROT/TrEMBL. The number of candidate proteins can be further narrowed down by using species names and keywords. The TagIdent program can narrow down the candidate list by peptide sequences because of the high specificity of short sequence matches. The PeptIdent program incorporates mass fingerprinting information with information such as pI, MW, and species name. Candidate proteins are ranked by the number of matching peptides. The CombSearch tool takes advantage of the strength of multiple parameters by using combined composition, sequence tags, and peptide fingerprinting information to perform combined searches against the databases.

ProFound (http://prowl.rockefeller.edu/profound_bin/WebProFound.exe) is a web server with a set of interconnected programs. It searches a protein sequence database using MS fingerprinting information. A Bayesian algorithm ranks the database matches according to the probability of database sequences producing the peptide mass fingerprints.

Mascot (www.matrixscience.com/search_form_select.html) is another web server that identifies proteins based on peptide mass fingerprints, sequence entries, or raw MS/MS data from one or more peptides.

Differential In-Gel Electrophoresis

Differences in protein expression patterns can be detected in a similar way as in fluorescent-labeled DNA microarrays, using a technique called *differential in-gel electrophoresis* (DIGE) (Fig. 19.2). Proteins from experimental and control samples are labeled with differently colored fluorescent dyes. They are mixed together before electrophoresis on a two-dimensional gel. Differentially expressed proteins in both conditions can be coseparated and visualized in the same gel. Compared to regular 2D-PAGE, the process reduces the noise and improves the reproducibility and sensitivity of detection. In principle, it resembles the two-color DNA microarray analysis. The drawbacks of this approach are that different proteins take up fluorescent tags to different extents and that some proteins labeled with the fluorophores may become less soluble and precipitate before electrophoresis.

Protein Microarrays

Protein microarray chips are conceptually similar to DNA microarray chips (see chapter 17) and can be built to contain high-density grids with immobilized proteins for high throughput analysis. The chips contain entire immobilized proteome. However, they are not meant to be used to bind and quantitate complementary molecules as in DNA microarrays. Instead, they are used for studying protein function by

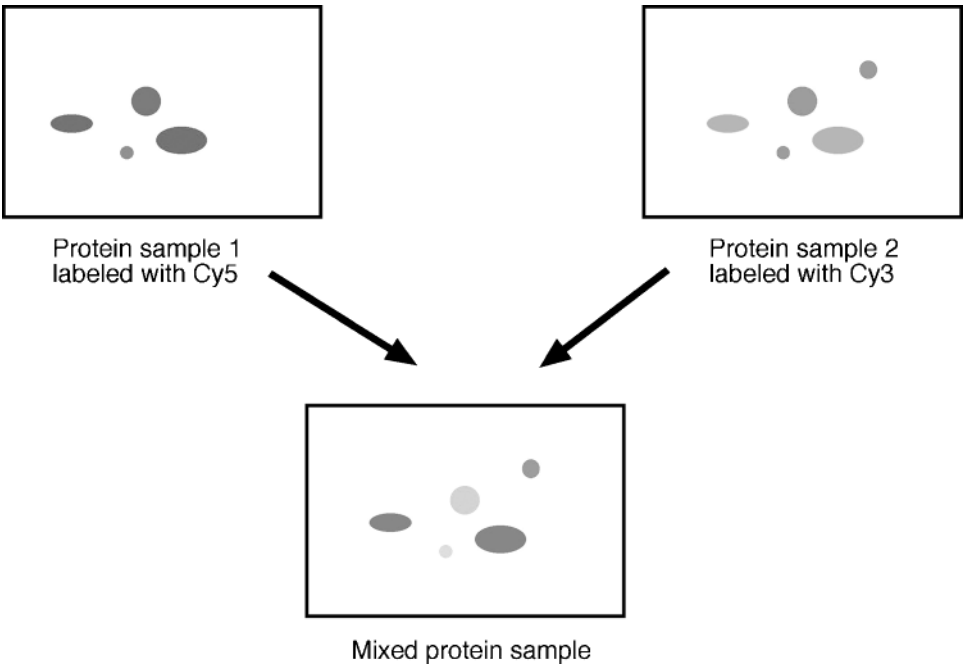


Figure 19.2: Schematic diagram showing protein differential detection using DIGE. Protein sample 1 (representing the experimental condition) is labeled with a red fluorescent dye (Cy5). Protein sample 2 (representing the control condition) is labeled with a green fluorescent dye (Cy3). The two samples are mixed together before running on a two-dimensional gel to obtain a total protein differential display map (see color plate section).

providing a solid support for assaying enzyme activity, or protein–protein interactions, protein–DNA/RNA interactions or protein–ligand interactions in an all-against-all format.

To make protein chips truly analogous to DNA chips, the solid support has to contain specific proteins or ligands that capture protein molecules by complementarity. A classical approach to this problem is to perform an immunoassay by using a spectrum of antibodies against the whole proteome. The antibodies can be fixed on a solid support for assaying thousands of proteins simultaneously. However, a major drawback of this approach is that natural antibodies are easily denatured and have a high tendency to cross-react with nonspecific antigens. In addition, producing antibodies for every single protein from an organism is prohibitively expensive.

To overcome this hurdle, a new technique is being developed that uses "protein scaffolds" to capture target molecules. The scaffolds are similar to antibodies but smaller, more stable and more specific in their binding of target proteins. They can be made in a cell-free system and attached with two fluorescence tags. This technique uses the principle of fluorescence resonance energy transfer, which is an excitation energy transfer between two fluorescent dye molecules whose excitation and

absorption spectra overlap. The efficiency of the energy transfer depends on the distance of the two dyes. If one portion of the tagged protein is involved in binding to a target protein, the protein conformational changes cause the two fluorescent tags to move apart, disrupting the excitation energy transfer between the dyes such that it can be monitored on fluorescence spectra.

A technology called *Protein-Print* is in early development, which is essentially a molecular imprinting method. Chemical monomers are used to coat target proteins, which are then allow to polymerize. When polymerization is complete and the target molecules removed, a mould is formed that resembles the shape of the target protein. The moulds can then be used to capture like molecules with high specificity.

These are some of the promising technologies currently under development. Their high throughput nature means that they may eventually succeed the two-dimensional gel-based method. When the proteome chips become available, data analysis for identifying coregulated proteins should be relatively easy because it will be similar to that used for DNA microarrays. Similar image analysis and clustering algorithms can be applied to identify coregulated proteins.

POSTTRANSLATIONAL MODIFICATIONS

Another important aspect of the proteome analysis concerns posttranslational modifications. To assume biological activity, many nascent polypeptides have to be covalently modified before or after the folding process. This is especially true in eukaryotic cells where most modifications take place in the endoplasmic reticulum and the Golgi apparatus. The modifications include proteolytic cleavage; formation of disulfide bonds; addition of phosphoryl, methyl, acetyl, or other groups onto certain amino acid residues; or attachment of oligosaccharides or prosthetic groups to create mature proteins. Posttranslational modifications have a great impact on protein function by altering the size, hydrophobicity and overall conformation of the proteins. The modifications can directly influence protein–protein interactions and distribution of proteins to different subcellular locations.

It is therefore important to use bioinformatics tools to predict sites for posttranslational modifications based on specific protein sequences. However, prediction of such modifications can often be difficult because the short lengths of the sequence motifs associated with certain modifications. This often leads to many false-positive identifications. One such example is the known consensus motif for protein phosphorylation, [ST]-x-[RK]. Such a short motif can be found multiple times in almost every protein sequence. Most of the predictions based on this sequence motif alone are likely to be wrong, producing very high rates of false-positives. Similar situations can be found in other predicted modification sites. One of the reasons for the false predictions is that neighboring environment of the modification sites is not considered.

To minimize false-positive results, a statistical learning process called *support vector machine* (SVM) can be used to increase the specificity of prediction. This is

a data classification method similar to the linear or quadratic discriminant analysis (see Chapter 8). In this method, the data are projected in a three-dimensional space or even a multidimensional space. A *hyperplane* – a linear or nonlinear mathematical function – is used to best separate true signals from noise. The algorithm has more environmental variables included that may be required for the enzyme modification. After training the algorithm with sufficient structural features, it is able to correctly recognize many posttranslational modification patterns.

AutoMotif (http://automotif.bioinfo.pl/) is a web server predicting protein sequence motifs using the SVM approach. In this process, the query sequence is chopped up into a number of overlapping fragments, which are fed into different kernels (similar to nodes). A hyperplane, which has been trained to recognize known protein sequence motifs, separates the kernels into different classes. Each separation is compared with known motif classes, most of which are related to posttranslational modification. The best match with a known class defines the functional motif.

Prediction of Disulfide Bridges

A disulfide bridge is a unique type of posttranslational modification in which covalent bonds are formed between cysteine residues. Disulfide bonds are important for maintaining the stability of certain types of proteins.

The disulfide prediction is the prediction of paring potential or bonding states of cysteines in a protein. Accurate prediction of disulfide bonds may also help to predict the three-dimensional structure of the protein of interest. This problem can be tackled by using either profiles constructed from multiple sequence alignment or residue contact potentials calculated based on the local sequence environment. Advanced neural networks or SVM or hidden Markov model (HMM) algorithms are often used to discern long-distance pairwise interactions among cysteine residues. The following program is one of the publicly available programs specialized in disulfide prediction.

Cysteine (http://cassandra.dsi.unifi.it/cysteines/) is a web server that predicts the disulfide bonding states of cysteine residues in a protein sequence by building profiles based on multiple sequence alignment information. A recursive neural network (see Chapter 14) ranks the candidate residues for disulfide formation.

Identification of Posttranslational Modifications in Proteomic Analysis

Posttranslational modifications can be experimentally identified based on MS fingerprinting data. Certain peptide identification tools are able to search for known posttranslational modification sites in a sequence and incorporate extra mass based on the type of modifications during database fragment matching. There are two subprograms in the ExPASY proteomics server and an independent RESID database that are related to predicting posttranslational modifications.

ExPASY (www.expasy.ch/tools) contains a number of programs to determine posttranslational modifications based on MS molecular mass data. FindMod is a subprogram that uses experimentally determined peptide fingerprint information to

compare the masses of the peptide fragments with those of theoretical peptides. If a difference is found, it predicts a particular type of modification based on a set of predefined rules. It can predict twenty-eight types of modifications, including methylation, phosphorylation, lipidation, and sulfation. GlyMod is a subprogram that specializes in glycosylation determination based on the difference in mass between experimentally determined peptides and theoretical ones.

RESID (http://pir.georgetown.edu/pirwww/search/textresid.html) is an independent posttranslational modification database listing 283 types of known modifications. It can search by text or MWs.

PROTEIN SORTING

Subcellular localization is an integral part of protein functionality. Many proteins exhibit functions only after being transported to certain compartments of the cell. The study of the mechanism of protein trafficking and subcellular localization is the field of protein sorting (also known as protein targeting), which has become one of the central themes in modern cell biology. Identifying protein subcellular localization is an important aspect of functional annotation, because knowing the cellular localization of a protein often helps to narrow down its putative functions.

For many eukaryotic proteins, newly synthesized protein precursors have to be transported to specific membrane-bound compartments and be proteolytically processed to become functional. These compartments include chloroplasts, mitochondria, the nucleus, and peroxisomes. To carry out protein translocation, unique peptide signals have to be present in the nascent proteins, which function as "zip codes" that direct the proteins to each of these compartments. Once the proteins are translocated within the organelles, protease cleavage takes place to remove the signal sequences and generate mature proteins (another example of posttranslational modification). Even in prokaryotes, proteins can be targeted to the inner or outer membranes, the periplasmic space between these membranes, or the extracellular space. The sorting of these proteins is similar to that in eukaryotes and relies on the presence of signal peptides.

The signal sequences have a weak consensus but contain some specific features. They all have a hydrophobic core region preceded by one or more positively charged residues. However, the length and sequence of the signal sequences vary tremendously. Peptides targeting mitochondria, for example, are located in the *N*-terminal region. The sequences are typically twenty to eighty residues long, rich in positively charged residues such as arginines as well as hydroxyl residues such as serines and threonines, but devoid of negatively charged residues, and have the tendency to form amphiphilic α-helices. These targeting sequences are cleaved once the precursor proteins are inside the mitochondria. Chloroplast localization signals (also called transit peptides) are also located in the *N*-terminus and are about 25 to 100 residues in length, containing very few negatively charged residues but many hydroxylated residues such as serine. An interesting feature of the proteins targeted for the chloroplasts is that the

transit signals are bipartite. That is, they consist of two adjacent signal peptides, one for targeting the proteins to the stroma portion of the chloroplast before being cleaved and the other for targeting the remaining portion of the proteins to the thylakoids. Localization signals targeting to the nucleus are variable in length (seven to forty-one residues) and are found in the internal region of the proteins. They typically consist of one or two stretches of basic residues with a consensus motif K(K/R)X(K/R). Nuclear signal sequences are not cleaved after protein transport.

Considerable variations in length and sequence make accurate prediction of signal peptides using computational approaches difficult. Nonetheless, various computational methods have been developed to predict the subcellular localization signals. In general, they fall within three categories. Some algorithms are signal based, depending on the knowledge of charge, hydrophobicity, or consensus motifs. Some are content based, depending on the sequence statistics such as amino acid composition. The third group of algorithm combines the virtue of both signals and content and appears to be more successful in prediction. Neural network- and HMM-based algorithms are examples of the combined approach. Here are some of the most frequently used programs for the prediction of subcellular localization and protein sorting signals with reasonable accuracy (65% to 70%).

SignalP (www.cbs.dtu.dk/services/SignalP-2.0/#submission) is a web-based program that predicts subcellular localization signals by using both neural networks and HMMs. The neural network algorithm combines two different scores, one for recognizing signal peptides and the other for protease cleavage sites. The HMM-based analysis discriminates between signal peptides and the *N*-terminal transmembrane anchor segments required for insertion of the protein into the membrane. The program is trained by three different training sets, namely, eukaryotes, Gram-negative bacteria and Gram-positive bacteria. This distinction is necessary because there are significant differences in the characteristics of the signal peptides from these organisms. Therefore, appropriate datasets need to be selected before analyzing the sequence. The program predicts both the signal peptides and the protease cleavage sites of the query sequence.

TargetP (www.cbs.dtu.dk/services/TargetP/) is a neural network-based program, similar to SignalP. It predicts the subcellular locations of eukaryotic proteins based on their *N*-terminal amino acid sequence only. It uses analysis output from SignalP and feeds it into a decision neural network, which makes a final choice regarding the target compartment.

PSORT (http://psort.nibb.ac.jp/) is a web server that uses a nearest neighbor method to make predictions of subcellular localizations. It compares the query sequence to a library of signal peptides for different cellular localizations. If the majority of the closest signal peptide matches (nearest neighbors) are for a particular cellular location, the sequence is predicted as signal peptide for that location. It is functionally similar to TargetP, but may have lower sensitivity. An iPSORT is available in the same website that predicts *N*-terminal sorting signals and is an equivalent to SignalP.

PROTEIN–PROTEIN INTERACTIONS

In general, proteins have to interact with each other to carry out biochemical functions. Thus, mapping out protein–protein interactions is another important aspect of proteomics. Interprotein interactions include strong interactions that allow formation of stable complexes and weaker ones that exist transiently. Proteins involved in forming complexes are generally more tightly coregulated in expression than those involved in transient interactions. Protein–protein interaction analysis at the proteome level helps reveal the function of previously uncharacterized proteins on the basis of the "guilt by-association" rule.

Experimental Determination

Protein interactions are commonly detected by using the classic yeast two-hybrid method that relies on the interaction of "bait" and "prey" proteins in molecular constructs in yeast. In this strategy, a two-domain transcriptional activator is employed as a helper for determining protein–protein interactions. The two domains which are a DNA-binding domain and a trans-activation domain normally interact to activate transcription. However, molecular constructs are made such that each of the two domains is covalently attached to each of the two candidate proteins (bait and prey). If the bait and prey proteins physically interact, they bring the DNA-binding and trans-activation domains in such close proximity that they reconstitute the function of the transcription activator, turning on the expression of a reporter gene as a result. If the two candidate proteins do not interact, the reporter gene expression remains switched off.

This technique is essentially a low throughput approach because each bait and prey construct has to be prepared individually to map interactions between all proteins. Nonetheless, it has been systematically applied to study interactions at the whole proteome level. Protein–protein interaction networks of yeast and a small number of other species have been subsequently determined using this method. A major flaw in this method is that it is an indirect approach to probe protein–protein interaction and has a tendency to generate false positives (spurious interactions) and false negatives (undetected interactions). It has been estimated from proteome-wide characterizations that the rate of false positives can be as high as 50%. Another weakness is that only pairwise interactions are measured, and therefore interactions that only take place when multiple proteins come together are omitted.

There are many alternative approaches to determining protein–protein interactions. One of them is to use a large-scale affinity purification technique that involves attaching fusion tags to proteins and purifying the associated protein complexes in an affinity chromatography column. The purified proteins are then analyzed by gel electrophoresis followed by MS for identification of the interacting components. The protein microarray systems mentioned above also provide a high throughput alternative for studying protein–protein interactions. Although none of the methods

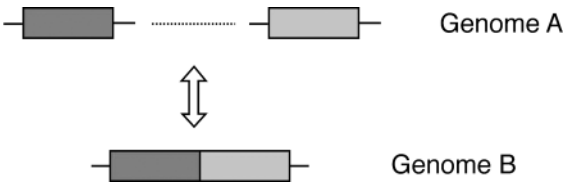


Figure 19.3: Rosetta stone method for prediction of genes encoding interacting proteins based on domain fusion patterns in different genomes. In genome A, two different domains exist in separate open reading frames. In genome B, they are fused together in one protein-encoding frame. Conversely, the two domains of the same protein encoded in genome B may become separate in genome A, but still perform the same function through physical interactions.

are guaranteed to eliminate false positives and false negatives, combining multiple approaches in theory compensates for the potential weaknesses of each technique and minimizes the artifacts.

Prediction of Protein–Protein Interactions

Decades of research on protein biochemistry and molecular biology has accumulated tremendous amount of data related to protein–protein interactions, which allow the extraction of some general rules governing these interactions. These rules have facilitated the development of algorithms for automated prediction of protein–protein interactions. The currently available tools are generally based on evolutionary studies of gene sequences, gene linkage patterns, and gene fusion patterns, which are described in detail next.

Predicting Interactions Based on Domain Fusion

One of the prediction methods is based on gene fusion events. The rationale goes like this: if A and B exist as interacting domains in a fusion protein in one proteome, the gene encoding the protein is a fusion gene. Their homologous gene sequences A′ and B′ existing separately in another genome most likely encode proteins interacting to perform a common function. Conversely, if ancestral genes A and B encode interacting proteins, they may have a tendency to be fused together in other genomes during evolution to enhance their effectiveness. This method of predicting protein–protein interactions is called the "Rosetta stone" method (Fig. 19.3) because a fused protein often reveals relationships between its domain components.

The further justification behind this method is that when two domains are fused in a single protein, they have to be in extremely close proximity to perform a common function. When the two domains are located in two different proteins, to preserve the same functionality, their close proximity and interaction have to be preserved as well. Therefore, by studying gene/protein fusion events, protein–protein interactions can be predicted. This prediction rule has been proven to be rather reliable and since successfully applied to a large number of proteins from both prokaryote and eukaryotes.

Predicting Interactions Based on Gene Neighbors

Gene orders, generally speaking, are poorly conserved among divergent prokaryotic genomes (see Chapter 16). However, if a certain gene linkage is found to be indeed conserved across divergent genomes, it can be used as a strong indicator of formation of an operon that encodes proteins that are functionally and even physically coupled. This rule of predicting protein–protein interactions holds up for most prokaryotic genomes. For eukaryotic genomes, gene order may be a less potent predictor of protein interactions than a tight coregulation for gene expression.

Predicting Interactions Based on Sequence Homology

If a pair of proteins from one proteome are known to interact, their conserved homologs in another proteome are likely to have similar interactions. The homologous pairs are referred to as *interologs*. This method relies on the correct identification of orthologs and the use of existing protein interaction databases. The method has potential to model protein quaternary structure if one pair of proteins have known structures.

InterPreTS (www.russell.embl-heidelberg.de/people/patrick/interprets/interprets.html) is a web server that has a built-in database for interacting domains based on known three-dimensional protein structures. Two protein sequences are used as query to search against the database for homologs. The alignment of the query sequences and database domains is carried out using HMMer (see Chapter 6). If the alignment scores for both sequences are above the threshold and the contact residues are found to be conserved, the two proteins are considered to be interacting proteins.

IPPRED (http://cbi.labri.fr/outils/ippred/IS_part_simple.php) is a similar web-based program that allows the user to submit multiple protein sequences. The program searches homologous sequences using BLAST in a database of known interacting protein pairs (BIND). If any two query sequences have strong enough similarity with known interacting protein pairs, they are inferred as interacting partners.

Predicting Interactions Based on Phylogenetic Information

Protein interactions can be predicted using phylogenetic profiles, which are defined as patterns of gene pairs that are concurrently present or absent across genomes. In other words, this method detects the copresence or co-absence of orthologs across a number of genomes. Genes having the same pattern of presence or absence across genomes are predicted as encoding interacting proteins. The logic behind the cooccurrence approach is that proteins normally operate as a complex. If one of the components of the complex is lost, it results in the failure of the entire complex. Under the selective pressure, the rest of the nonfunctional interacting partners in the complex are also lost during evolution because they have become functionally unnecessary. The rule based on concurrent gene loss or gene gain has proven to be less accurate than the rules based on gene fusion and gene neighbors. An example of using the phylogenetic profile method to predict interacting proteins is shown in Figure 19.4.

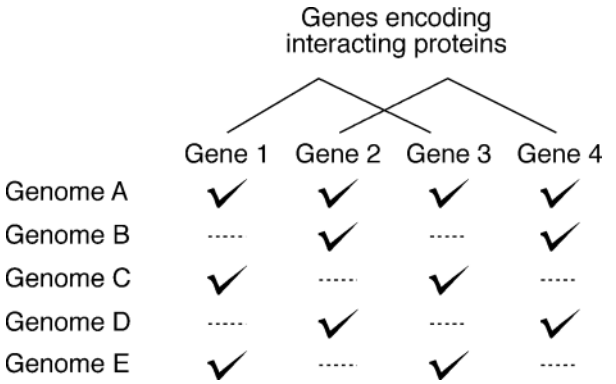


Figure 19.4: Phylogenetic profile method for predicting interacting proteins based on copresence and co-absence of the encoding genes across genomes. The presence is indicated by checks and absence by dashed lines. The protein pairs encoded by genes one and three as well as genes two and four are predicted as interacting partners.

A more quantitative phylogenetic method to predict protein interactions is the "mirror tree" method, which examines the resemblance between phylogenetic trees of two sequence families (Fig. 19.5). The rationale is that if two protein trees are nearly identical in topology and are highly correlated in terms of evolutionary rate, they are highly likely to interact with each other. This is because if mutations occur at

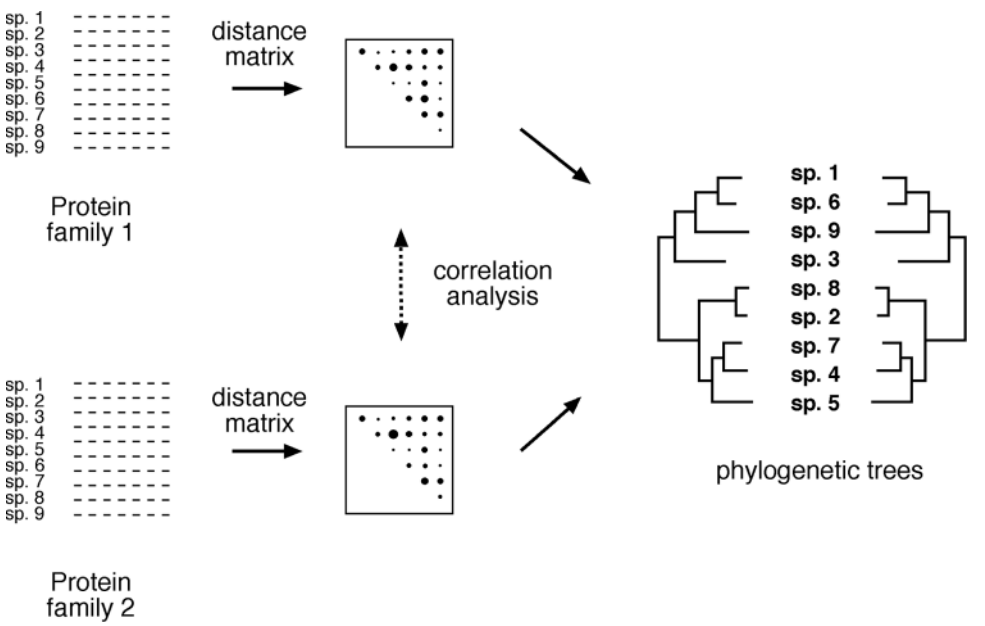


Figure 19.5: Mirror tree method for prediction of interacting proteins based on strong statistical correlation of evolutionary distance matrices used to build two phylogenetic trees for the two protein families of interest. The two trees have a near identical topology resulting in a near mirror image. The distance matrices used to construct the trees are compared using correlation analysis.

the interaction surface for one of the proteins, corresponding mutations are likely to occur in the interacting partner to sustain the interaction. As a result, the two interacting proteins should have very similar phylogenetic trees reflecting very similar evolutionary history. To analyze the extent of coevolution, correlation coefficients (r) of evolutionary distance matrices for the two groups of protein homologs used in constructing the trees are examined. It has been shown that if $r > 0.8$, there is a strong indication for protein interactions.

Matrix (http://orion.icmb.utexas.edu/cgi-bin/matrix/matrix-index.pl) is a web server that predicts interaction between two protein families. The server aligns two individual protein data sets (assuming each representing a protein family) using Clustal. It then derives distance matrices from the two alignment files and aligns the matrices to discover similar portions that may indicate interacting partners from the two protein families.

ADVICE (Automated Detection and Validation of Interaction based on the Co-Evolutions, http://advice.i2r.a-star.edu.sg/) is a similar web server providing prediction of interacting proteins using the mirror-tree approach. It performs automated BLAST searches for a given protein sequence pair to derive two sets of homologous sequences. The sequences are multiply aligned using CLUSTAL. A distance matrix for each set of alignment is then derived. The Pearson's correlation coefficient is subsequently calculated for detecting similarities between the two distance matrices. If the coefficient $r > 0.8$, the two query sequences are predicted to be a interacting pair.

Predicting Interactions Using Hybrid Methods

It needs to be emphasized that each of these prediction methods is based on a particular hypothesis and may exhibit a certain degree of bias associated with the hypothesis. Because it is difficult to evaluate the performance of each individual prediction method, the user of these prediction algorithms is recommended to use a combined approach that uses multiple methods to reduce bias and error rates and to yield a higher level of confidence in the protein interaction prediction. The following internet program is a good example of combining multiple lines of evidence in predicting protein-protein interactions.

STRING (Search Tool for the Retrieval of Interacting Genes/Proteins, http://www.bork.embl-heidelberg.de/STRING/) is a web server that predicts gene and protein functional associations based on combined evidence of gene linkage, gene fusion and phylogenetic profiles. The current version also includes experimental co-expression data as well as documented interactions resulted from literature mining. Functional associations include both direct and indirect protein-protein interactions. Indirect interactions can mean enzymes in the same pathway sharing a common substrate or proteins regulating each other in the genetic pathway. The server contains information for orthologous groups from 110 completely sequenced genomes. The query sequence is first classified into an orthologous group based on the COG classification (see Chapter 7) and is then used to search the database for known conserved linkage pattern, gene fusions, and phylogenetic profiles. The server uses a weighted

scoring system that evaluates the significance of all three types of protein associations among the genomes. To reduce false positives and increase reliability of the prediction, the three types of genomic associations are checked against an internal reference set. A single score of pairwise interactions is given as the final output which also contains all three types of evidence plus a summary of combined protein interaction network involving multiple partners. The server returns a list of predicted protein-protein associations and a graphic representation of the association network.

SUMMARY

Protein expression analysis at the proteome level promises more accurate elucidation of cellular functions. This is an advantage over genomic analysis, which does not necessarily lead to prediction of protein functions. Traditional experimental approaches to proteomics include large-scale protein identification using 2D-PAGE and MS. The identification process requires the integration of bioinformatics tools to search databases for matching peptides. Newer protein expression profiling techniques include DIGE and protein microarrays. Protein functions can be modulated as a result of posttranslational modifications. Sequence based prediction often results in high rates of false-positives owing to limited understanding of the structural features required for the modifications. A step toward minimizing the false-positive rates in prediction is the use of SVM. Another area of proteomics is defining protein subcellular localization signals. Several web tools such as TargetP, SignalP, and PSORT are available to give reasonably successful prediction of signal peptides. Protein–protein interactions are normally determined using yeast two-hybrid experiments or other experimental methods. However, theoretical prediction of such interactions is providing a promising alternative. The current prediction methods are based on domain fusion, gene linkage pattern, sequence homology, and phylogenetic information. The ability to predict protein interactions is of tremendous value in genome annotation and in understanding the function of genes and their encoded proteins. The computational approach helps to generate hypotheses to be tested by experiments.

FURTHER READING

Aebersold, R., and Mann, M. 2003. Mass spectrometry-based proteomics. *Nature* 422:198–207.

Cutler, P. 2003. Protein arrays: The current state-of-the-art. *Proteomics* 3:3–18.

Donnes, P., and Hoglund, A. 2004. Predicting protein subcellular localization: past, present, and future. *Genomics Proteomics Bioinformatics* 2:209–15.

Droit, A., Poirier, G. G., and Hunter, J. M. 2005. Experimental and bioinformatic approaches for interrogating protein-protein interactions to determine protein function. *J. Mol. Endocrinol.* 34:263–80.

Eisenhaber, F., Eisenhaber, B., and Maurer-Stroh, S. 2003. "Prediction of post-translational modifications from amino acid sequence: Problems, pitfalls, and methodological hints." In *Bioinformatics and Genomes: Current Perspectives,* edited by M. A. Andrade, 81–105. Wymondham, UK: Horizon Scientific Press.

Emanuelsson, O. 2002. Predicting protein subcellular localisation from amino acid sequence information. *Brief. Bioinform.* 3:361–76.

Huynen, M. A., Snel, B., Mering, C., and Bork, P. 2003. Function prediction and protein networks. *Curr. Opin. Cell Biol*. 15:191–8.

Mann, M., and Jensen, O. N. 2003. Proteomic analysis of post-translational modifications. *Nature Biotechnol*. 21:255–61.

Nakai, K. 2000. Protein sorting signals and prediction of subcellular localization. *Adv. Protein Chem*. 54:277–344.

Nakai, K. 2001. Review: Prediction of in vivo fates of proteins in the era of genomics and proteomics. *J. Struct. Biol.* 134:103–16.

Phizicky, E., Bastiaens, P. I. H., Zhu, H., Snyder, M., and Fields, S. 2003. Protein analysis on a proteomic scale. *Nature* 422:208–15.

Sadygov, R. G., Cociorva, D., and Yates, J. R. III. 2004. Large-scale database searching using tandem mass spectra: Looking up the answer in the back of the book. *Nat. Methods* 1:195–202.

Tyers, M., and Mann, M. 2003. From genomics to proteomics. *Nature* 422:193–7.

Valencia, A., and Pazos, F. 2002. Computational methods for the prediction of protein interactions. *Curr. Opin. Struct. Biol*. 12:368–73.

Valencia, A., and Pazos, F. 2003. Prediction of protein-protein interactions from evolutionary information. *Methods Biochem. Anal.* 44:411–26.