Terms and Conditions of Use of Digitised Theses from Trinity College Library Dublin

Copyright statement

All material supplied by Trinity College Library is protected by copyright (under the Copyright and Related Rights Act, 2000 as amended) and other relevant Intellectual Property Rights. By accessing and using a Digitised Thesis from Trinity College Library you acknowledge that all Intellectual Property Rights in any Works supplied are the sole and exclusive property of the copyright and/or other IPR holder. Specific copyright holders may not be explicitly identified. Use of materials from other sources within a thesis should not be construed as a claim over them.

A non-exclusive, non-transferable licence is hereby granted to those using or reproducing, in whole or in part, the material for valid purposes, providing the copyright owners are acknowledged using the normal conventions. Where specific permission to use material is required, this is identified and such permission must be sought from the copyright holder or agency cited.

Liability statement

By using a Digitised Thesis, I accept that Trinity College Dublin bears no legal responsibility for the accuracy, legality or comprehensiveness of materials contained within the thesis, and that Trinity College Dublin accepts no liability for indirect, consequential, or incidental, damages or losses arising from use of the thesis for whatever reason. Information located in a thesis may be subject to specific use constraints, details of which may not be explicitly described. It is the responsibility of potential and actual users to be aware of such constraints and to abide by them. By making use of material from a digitised thesis, you accept these copyright and disclaimer provisions. Where it is brought to the attention of Trinity College Library that there may be a breach of copyright or other restraint, it is the policy to withdraw or take down access to a thesis while the issue is being resolved.

Access Agreement

By using a Digitised Thesis from Trinity College Library you are bound by the following Terms & Conditions. Please read them carefully.

I have read and I understand the following statement: All material supplied via a Digitised Thesis from Trinity College Library is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of a thesis is not permitted, except that material may be duplicated by you for your research use or for educational purposes in electronic or print form providing the copyright owners are acknowledged using the normal conventions. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone. This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

A Bioinformatics Approach to (Intra-)Genome Comparisons

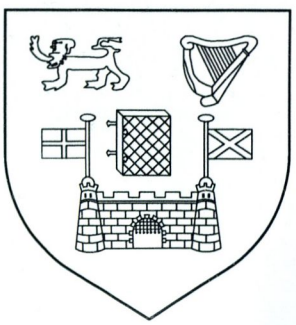
by

Karsten Hokamp

A thesis submitted to
The University of Dublin
for the degree of

Doctor of Philosophy

Department of Genetics
Trinity College
University of Dublin



October, 2001

Declaration

This thesis is submitted by the undersigned for the degree of Doctor in Philosophy at the University of Dublin. It has not been submitted as an exercise for a degree at any other university.

Apart from the advice, assistance, and joint effort mentioned in the acknowledgements and in the text, this thesis is entirely my own work.

I agree that the library may lend or copy this thesis freely upon request.

Karsten Hokamp

October 2001

Acknowledgements

The three years that I spent on this project were equally challenging and fascinating. It would have been much less fun and much more difficult without the contribution of a lot of people, which I would like to acknowledge here.

I owe a great debt to all of my family for their continuing support and faith. Meeting Fiona was the best thing that ever happened to me. She triggered the whole thing off and I'm forever grateful for the twist of fate that brought us together. Special thanks to Aoife for being the best lab partner I could have wished for. We complemented each other perfectly: I produced the bugs, you found them. Your work for chapter 4 was invaluable! Andrew, as the senior member of the lab, provided an infinite source of wisdom and anecdotes. My good friend Daniel made sure that I got enough physical distraction from the computer work. Best buddy Hugo managed to cheer me up when the working hours were long. Thanks to Lucy and Cathel for paving the way - you set a very high standard! Avril is the best organised student I have ever seen, and she is always there for sound advice. Thanks for your great enthusiasm! Simon, I was happy to see a Linuxer joining in, despite the red hat. Ronan, thanks for making sure that I practiced my German. The Tuesday lunch group was a brilliant forum for presenting my work, getting feedback and learn about loads of fascinating cow stuff. All the newcomers and lab guests I have to thank for bearing with me in the last weeks - it's about time you see me from a different side! Vera, the kitchen fairy - thanks so much for keeping that vital coffee club going. Kevin, it's a damn fine department that you have running here!

Last, but certainly not least, I have to thank Ken for granting me all the freedom that I wanted, for buying me all the computers that I needed, for involving me with PubCrawler, and for being such an incredibly brilliant boss!

Contents

List of Figures

List of Tables

Abbreviations

BAC	Bacterial Artificial Chromosome
BLAST	Basic Local Alignment Search Tool
BLOSUM	Blocks Substitution Matrix
CDS	Coding Sequence
CGI	Common Gateway Interface
CPU	Central Processing Unit
DNA	Deoxy-Ribonucleotide Acid
EBI	European Bioinformatics Institute
ECC	Error Correction Code
EMBL	European Molecular Biology Lab
EST	Expressed Sequence Tag
FPGA	Field Programmable Gate Arrays
FTP	File Transfer Protocol
GB	GigaBytes
GHz	GigaHertz
HSP	High-Scoring Segment Pair
HTML	Hyper-Text Markup Language
IP	Internet Protocol
Kb	Kilobases

KB	KiloBytes
Mb	Megabases
MB	MegaBytes
MHz	MegaHertz
Mya	Million years ago
Myr	Million years
NCBI	National Center for Biotechnology Information
NFS	Network File System
NIC	Network Interface Controller
OS	Operating System
PAM	Percent (or Point) Accepted Mutation
RAID	Redundant Array of Inexpensive Disks
RAM	Random Access Memory
RISC	Reduced Instruction Set Computer
RNA	Ribonucleotide Acid
SW	Smith-Waterman
STS	Sequence-Tagged Site
TFTP	Trivial File Transfer Protocol
VLSI	Very Large-Scale Integration
WWW	World-Wide Web
bp	basepairs
mRNA	messenger RNA
tRNA	transfer RNA

"If I have seen further it is by standing upon the shoulders of giants."

Sir Isaac Newton (1642-1727)

Summary

The analysis of large volumes of genomic data generates special computational needs. A Beowulf-type computer cluster was set up for high-performance computing. Improvements over existing tools for the efficient parallelisation of similarity searches on such systems were accomplished with the program `wrapid`.

To investigate the evolution of genomes on a molecular basis, a method for the detection of paralogous blocks was developed. Application to the yeast *Saccharomyces cerevisiæ* showed that fully automatic generated results approximated previously available, manually edited information very well. An improved method for the graphical presentation of duplicated regions was implemented which can be used through the World Wide Web and allows the highly interactive exploration of many aspects of the produced results.

Sequence and mapping data from the public Human Genome Project was subjected to intra-genomic comparison. Previously reported and new paralogous regions of statistically significant sizes were detected. A new resource for the interactive graphical presentation of these blocks at variable levels of resolution was implemented. Further phylogenetic analyses indicated that they contain an excess of gene pairs created in a burst of duplication activity that took place approximately 333-583 Mya, spanning the estimated time of the origin of vertebrates (about 500 Mya).

Tests with other genomes prove the benefits of the graphical presentation and its possible adaption to inter-genomic comparisons. The flexibility and

modularity of the approach warrant its usability for future projects.

Part II of this work describes the PubCrawler webservice, which arose from a side project and presents another example of how a bioinformatics tool can improve scientific research.

Part I

Chapter 1

General introduction

1.1 Bioinformatics

1.1.1 Definition and History

Despite its relatively young age as a scientific discipline compared to biology or even computer science, bioinformatics nevertheless has already reached an advanced degree of maturity, judging from the important role it plays in the life sciences. If one were to go by Danchin's definition (2000): "Bioinformatics was born when the use of computers was found to be necessary [in biology]", then one would have to date its beginnings to the 1960s. At around that time an increasing number of protein sequences started to accumulate, and according to Hagen (2000), several computing techniques were developed for their analysis, ranging from contig assembly to 3-D modelling. Ouzounis (2000) dates the emergence of the first algorithms and their computer implementations slightly later to the early 1970s.

What was first described as "biological computation" and "theoretical research in biology" received a new name in the late 80s. One of the first mentions of the word "bioinformatics" has been attributed to Arthur Lesk in 1986 (Sander, 2001). Even earlier, in 1985, grant applications were submitted for an Irish National Centre for BioInformatics (INCBI) (P. Sharp, personal

Box 1.1: Suggested definition of 'bioinformatics' for Oxford English Dictionary (Luscombe *et al.*, 2001)

(Molecular) **bio - informatics:** bioinformatics is conceptualising biology in terms of molecules (in the sense of physical chemistry) and applying "informatics techniques" (derived from disciplines such as applied maths, computer science and statistics) to understand and organise the information associated with these molecules, on a large scale. In short, bioinformatics is a management information system for molecular biology and has many practical applications.

communication). This came into existence in 1987, which shows, amongst other things, the official approval of the word. Sometimes it is used synonymously with "computational molecular biology" or even the whole of "computational biology" (Altman, 1998).

Initially, the boundaries of the field were quite loose, which is reflected by the variety of definitions that can be found in the literature (Bains, 1996; Benton, 1996; Altman, 1998; Kanehisa, 1998; Sansom and Smith, 2000; Danchin, 2000). Only recently a suggestion for an entry was submitted to the Oxford English Dictionary (Luscombe *et al.*, 2001, see Box 1.1).

The connection between biology and information technology occurred naturally following the identification of macromolecules as information carriers and with the appearance of problems that could only be solved through the number-crunching power of computers (Hagen, 2000). More advances in the field occurred after the development of rapid DNA sequencing methods in the mid-1970s by Maxam and Gilbert and by Sanger and Coulson (Trifonov (2000) and references therein) as well as increasing progress in information science and technology (Sansom and Smith, 2000). Many of the basic concepts of molecular sequence analysis were originally developed for studies of amino acids and then adapted to nucleic acids research (Claverie, 2000).

Although originally a lot of its methods were brought in from other fields, bioinformatics has emerged as a discipline with an increasing core of new and independent methods that are applied to data-rich problems in the life sciences. In the last couple of years, bioinformatics has penetrated every branch of biological and medical science, and computational planning and analysis has become an integral part of the biological discovery process (Sander, 2001). Its importance will continue to grow in the future with the need to analyse an ever-increasing quantity of data.

1.1.2 Research Areas and Applications

The list of bioinformatics core areas is long and still growing (Altman, 1998), and each field comes with its own specific set of methods and tools. Development of new research areas is often driven by advances in data generation and the emergence of new data sets (Wada, 2000). As mentioned before, the availability of protein sequences initially triggered computer-aided sequence analysis. X-ray crystallography led the way to structural analysis of proteins, 3-D modelling, as well as the study and the simulation of interactions between molecules. Prediction of secondary and tertiary structure is still an ongoing struggle. Approaches in these fields include *ab-initio* predictions according to the thermo-dynamical constraints of molecules, homology modelling, and protein threading – the alignment of amino acid sequences to known 3-D structures. International contests have been held regularly to evaluate the capabilities of state-of-the-art programs for protein structure prediction (Sippl *et al.*, 1999). Assessments of the results show that fully automated and human guided methods are getting better but still leave room for improvement (Fischer *et al.*, 1999; Sippl *et al.*, 1999).

The availability of fully sequenced genomes opened new research areas, such as computational genomics and proteomics: the large scale analysis of the complete set of genes and gene products of an organism. Identification

of genes and their products is a challenging task. Public (e.g., Lander *et al.*, 2001; Wright *et al.*, 2001) and private (e.g., Venter *et al.*, 2001) efforts are currently underway to establish a complete set of genes and proteins for the human genome. This is being carried out with elaborate algorithms (Burge and Karlin, 1997; Cuff *et al.*, 2000) on large computer farms, consisting of hundreds of powerful machines (Service, 2000). Availability of sequence data from a range of organisms supports not only the identification of genes but also the establishment of relationships among living things. The construction of molecular phylogenetic trees has played an important role ever since sequences of related proteins from different species became available (Hagen, 2000). Intra-genome comparisons have provided valuable information for inferences about the evolutionary history of complex organisms, such as yeast *Arabidopsis*, and human. Further knowledge can be obtained from inclusion of other organisms (rice, mouse, other yeasts). Particularly for microbial genomes, of which currently more than 59 completed sequences are publicly available, extensive studies of inter-genomic relationships have been carried out (Delcher *et al.*, 1999; Tekaia *et al.*, 1999). With the advent of complete sequences, this has been expanded to higher organisms as well (Chervitz *et al.*, 1998; Riechmann *et al.*, 2000; Rubin *et al.*, 2000).

The amount of data produced by EST sequencing (**E**xpressed **S**equence **T**ags, partial cDNA sequences (e.g., Adams *et al.*, 1991)) and micro-array experiments (Schena *et al.*, 1995) requires technologies for systematic analysis of gene-expression patterns on a large scale. Functional genomics is the new field that extends analysis from the individual building blocks of an organism to their interactions on an organismal level which lead to a manifold increase in complexity (Strausberg and Austin, 1999). New tools are needed that deal with massive amounts of data and adapt methods from multivariate statistics to the biological discovery process.

Of huge importance in bioinformatics is the capture, storage, and organisa-

tion of information. The basics of database design are well studied subjects of computer science. Due to the complicated nature of biological objects and their intricate dependencies specialised systems and access methods are required. The type of data that is of bio-medical relevance ranges from DNA and protein sequences via pattern and motifs to gene-expression information and even literature citations. The main repositories for DNA sequences can be found in the International Nucleotide Sequence Database Collection which consists of EMBL, GenBank, and DDBJ (Stoesser *et al.*, 2001). As of October 2001 (GenBank release 126.0) they hold approximately 13.6 million sequences comprising more than 14 billion nucleotides and are still growing at an enormous rate. Not only the size but also the variety increases: a recent overview reports 281 databases, many of them specialised (Baxevanis, 2001). These arise from the need to address a particular biological question or to present particular aspects of biological data. For example, shortly after the sequencing of ESTs began, a new database (dbEST) was created (Boguski *et al.*, 1993). In comparison to the corresponding entries in GenBank it contains value-added information, such as the latest homology and mapping data, as well as additional information about the libraries, cloning vectors, and source of mRNA used. A different aim is pursued by UniGene (Schuler, 1997). In this database ESTs are combined with known entries in GenBank to create a non-redundant set of genes.

Outside the academic environment, bioinformatics finds increasingly valuable commercial applications. In the pharmaceutical industry, years of practical research work for the identification of therapeutic targets have been saved by reducing massive quantities of sequence and gene-expression information to a manageable and useful amount of information (Fannon, 1996). Lyall (1996) lists the following stages in drug discovery in which bioinformatics is involved: information gathering and preliminary investigations, target identification, target validation, screening, and rational design. Not surprisingly, most of the pharmaceutical companies have established their own bioinformatics research

group or use services from bioinformatics companies that provide database access, research tools, and expertise.

As a recent study shows (Stevens *et al.*, 2001), the most often used bioinformatics analysis tools nowadays are similarity searches, motif searching, sequence retrieval, and multiple alignment. Many of these basic techniques need to be combined for more complex tasks such as phylogenetic analyses. An important aspect of bioinformatics tools lies in their ease of use and their accessibility. This has been greatly facilitated through the proliferation of the Internet (Boguski, 1994). Many programs do not need to be installed on a local computer anymore because they can be run through a WWW interface, using resources that are often far more powerful and up-to-date. Databases and access to them have also been greatly impacted, which becomes obvious from re-reading Doolittle's classic book "Of URFs and ORFs" (1986), probably the first bioinformatics book ever written. Doolittle talks about buying distributions of sequence databases on magnetic tape and provides postal addresses for GenBank, EMBL, and other centres (as opposed to the web sites and email addresses in use nowadays). Other advantages of the Internet related to bioinformatics include topical discussion groups and distance-learning courses (de la Vega *et al.*, 1996) which support training and the exchange of new ideas.

1.2 Gene and Genome Duplication

1.2.1 Evolutionary importance of gen(om)e duplication

In this thesis I apply bioinformatics methods to the human and other sequenced genomes to examine the history of gene duplications during their evolution. Genes form the basic building blocks of life and evolutionary changes occur through changes on the gene level. Differences in the size of genomic content between species caught the attention of geneticists early on and was referred to as the "C-value complexity paradox" (Lewin, 1983). It is known today

that the vast part of these variations is explicable through the existence of non-coding junk DNA rather than an increase in gene numbers. However, the fact remains that the number of genes differ even between closely related organisms. The question is, where do new genes come from? In his seminal book, Ohno (1970) draws out the theory of gene duplication as the major force of evolution. He argues that a redundant copy of a locus can escape the pressure imposed by natural selection, evolve into a new function, and become fixed in the genome. Alternative paths of gene accretion include transfer of genes from other organisms or formation *de novo* from surplus genetic material. The former mechanism does occur, but lateral gene transfer is far more likely to be successful in single-celled organisms, such as bacteria, than in multicellular eukaryotes, where it would be necessary for the transferred gene to integrate into the germ-line (Ochman *et al.*, 2000). *De novo* synthesis of genes through accumulation of mutations is possible but statistically so unlikely, that it could not be held responsible for the formation of large quantities of genes.

Not every duplication is tolerated because the balance of gene quantities can be very important. Trisomy of human chromosome 21 and subsequent overexpression of certain genes has been identified as the cause for Down syndrome (Korenberg *et al.*, 1994). Other duplications are not operative if regulatory objects or interacting counterparts of tightly integrated systems are missing. This favours the notion of wide-ranging duplications that involve chromosomal segments, whole chromosomes or, in the extreme, the complete genome (polyploidy). Sufficiently large duplicated regions prevent unbalanced gene expression and keep gene networks and arrangements of genes and regulatory elements intact. However, it should be noted that trisomy of individual chromosomes can lead to sometimes severe disabilities (e.g., Trisomy 18-like syndrome, Down syndrome).

The importance of gene duplication can be derived from the omnipresence of gene families and the large amount of similar genes within organisms. For re-

cently completed eukaryote genomes the number of duplicate gene pairs ranges from 30 % in *Saccharomyces cerevisiæ* (yeast) to 60 % in *Arabidopsis thaliana* (mustard weed) (Ball and Cherry, 2001). New genes can provide new functions and therefore evolutionary advances, as exemplified by the antifreeze gene in the giant Antarctic toothfish (Cheng and Chen, 1999). It has been speculated that polyploidisation in the yeast *Saccharomyces cerevisiæ* was responsible for the development of growth capabilities under anaerobic conditions (Wolfe and Shields, 1997). Ohno hypothesised that whole genome duplication was responsible for the Cambrian explosion, a leap in the evolution of vertebrates some 500 million years ago (Mya) (Ohno, 1997). Recent studies presented a model that emphasises the importance of gene loss or silencing after duplication, leading to speciation through divergent resolution (Lynch and Force, 2000; Lynch and Conery, 2000).

1.2.2 Homology and its variants

In 1844 the term "homology" was introduced by Richard Owen to identify and group similarities in nature (Tautz, 1998). Its precise meaning in biology today is of "having a common evolutionary origin" and in a loose sense of "possessing similarity or being matched" (Reeck *et al.*, 1987). A significant difference between the two terms consists in that fact that homology is an indivisible quality, whereas similarity can be quantified. It is important to note that similarity does not always imply homology. Sequences that reached a certain degree of similarity through convergence and not through a common evolutionary origin other than the cenancestor (the last common ancestor of all life) are called "analogues".

Several variations in the occurrence of homology led to the introduction of additional terminology for further specification. Fitch (1970) coined the terms "paralogy" and "orthology" to distinguish between homologies within and between species. The α-globin gene in mouse and human is an example of

an orthologous pair. Both genes stemmed from an α-globin gene in a common ancestor and evolved independently in both lineages after speciation. A paralogous gene pair arises from duplication within a species, as is the case with α- and β-globin. Because the duplication event happened before divergence of man and mouse, a paralogous pair can also be formed across the two species, for example, between mouse β-globin and human α-globin.

Following observations that many *Drosophila* genes correspond to two, three, four or more genes in higher vertebrates on different chromosomal locations, Spring (1997) proposed the term "tetralogue". It specifies a group of related genes that arose from multiple genome duplications as opposed to regional or tandem duplications. If the duplication events correspond to Ohno's proposed two rounds of polyploidy at the base of vertebrate evolution (1970), then the gene groups are also referred to as ohnologues (Wolfe, 2000).

1.2.3 Formation and fate of duplicated genes

Copies of a single gene can be formed via retrotransposition. This mechanism involves the reinsertion of a processed mRNA or its cDNA copy into a new chromosomal location (Vanin, 1985). Characteristics of these paralogues are their lack of introns, flanking by direct repeats, and their arbitrary position relative to the original gene. Most retransposons are not transcribed due to the lack of promoter regions and therefore form pseudogenes. Only a few intact examples of intronless, processed genes are known, such as the human phosphoglycerate kinase *PGK2* gene (McCarrey and Thomas, 1987) or the ribosome-binding protein *RBMXL* gene (Lingenfelter *et al.*, 2001).

Duplication of single genes or a group of adjacent genes can occur through unequal crossing-over. Due to errors in mitotic or meiotic processes two former alleles of the same gene (or group of genes) are placed on the same chromosome and are consequently replicated as a tandem array. Subsequent chromosomal rearrangements might increase the distance between the two copies but in

most cases they retain their physical proximity, as exemplified by the *Hox* genes (Brooke *et al.*, 1998) and the globin clusters (Wheeler *et al.*, 2001). A large numbers of tandem duplicates have been found in the genome sequences of *Arabidopsis* (The *Arabidopsis* Genome Initiative, 2000) and *Cænorhabditis elegans* (*C. elegans* Sequencing Consortium, 1998).

Duplicates are less affected by evolutionary constraints as long as one copy of the gene continues to carry out its assigned function. In most cases, accumulation of mutations will lead to gene silencing. From comparison of thousands of eukaryotic gene pairs it has been estimated by Lynch and Conery (2000) that more than 90% of duplicates dissappear before 50 million years have elapsed. Three scenarios are possible if the gene is not lost: (i) retention of the original function, for example if selective pressure for increased gene activity exists; (ii) "subfunctionalisation" - only a fraction of the original function is retained; and (iii) "neofunctionalisation" - development of a new function.

1.2.4 Polyploidy

The simultaneous duplication of all chromosomes in a genome is termed polyploidy. Two different mechanisms can lead to polyploidisation. In "allopolyploid" organisms genomes from two distinct parental species were fused to form a hybrid genome. The genome of "autopolyploids" was duplicated within the species, for example as a consequence of non-disjunction of all the daughter chromosomes following DNA replication (Li, 1997). Spring (1997) argues that allopolyloidisation presents small advantages, in that faster evolved genes reduce the likelihood of functional redundancy. Conversely, artificially produced autopolyploid plants appear to be generally inferior to their diploid progenitors (Li (1997) and references therein).

A diploid organism that undergoes polyploidy changes to a tetraploid state – it possesses four copies of each chromosome. The mechanism that re-establishes disomy, called diploidisation, is poorly understood. It involves extensive chro-

mosomal rearrangements leading to disruption of linkage of groups of genes and deletion of segments, resulting in loss of gene copies. Such events obscure evidence for whole genome duplication in paleopolyploids and aggravate the distinction between polyploidy as opposed to a series of aneuploidies. Despite accumulated evidence for whole genome duplication in *Arabidopsis thaliana* (Vision *et al.*, 2000) and *Saccharomyces cerevisiæ* (Wolfe and Shields, 1997) these explanations have not been met everywhere with full agreement.

Polyploidy in plants is not unusual with octoploid forms found, for example, in the sugarcane *Saccharum spontaneum* (Guimaraes *et al.*, 1997) and the cereal *Sorghum halpense* (Ming *et al.*, 1998). Polyploidy in mammals, in comparison, is extremely rare which has been attributed to their well-developed chromosomal sex-determining mechanism (Muller, 1925). However, one viable example of a tetraploid mammal was found recently in which chromosomal elimination has reconstituted the single XY sex-chromosome system (Gallardo *et al.*, 1999).

1.3 Large-scale homology search

The basic step for genome comparison consists in the detection of homologues amongst a set of sequences. A comprehensive analysis requires the rigorous comparison of each sequence against the rest. Several dynamic programming algorithms have been developed that are guaranteed to find the best match between two sequences through rigorous comparisons (Needleman and Wunsch, 1970; Smith and Waterman, 1981; Gotoh, 1982). Their computational demands make them too slow to be carried out with standard computers on large datasets in reasonable time. A variety of solutions have emerged in computer hardware and software to overcome this problem. Several algorithmic approaches are described in the next sections followed by descriptions of different computer architectures that provide speed-up for large-scale sequence comparisons.

1.3.1 Smith-Waterman implementations

The Smith-Waterman method (SW, 1981) in combination with an extension by Gotoh (1982) for affine gaps has evolved as the most widely used dynamic programming algorithm for rigorous sequence comparison. A very popular implementation can be found in SSEARCH, a program of the FASTA package (Pearson, 2000). The latest version is available with threading, which is a technique that allows parallel execution on multiprocessor machines or, with the help of special software layers, such as PVM (Sunderam, 1990) and MPI (Snir *et al.*, 1996), on networked computers.

Other algorithms exist that have managed to maximise usage of processor capabilities to increase the speed of SW searches. MPSRCH from Edinburgh Biocomputing Systems (`http://www.mpsrch.com`), which has been successfully used on the MasPar supercomputer (Blank, 1990), has recently been ported to more general hardware. In a demonstration of a beta version of the program, one billion comparisons per second were achieved on a 1GHz 64-bit Alpha chip (GenomeWeb, 2001), a task that would take more than a hundred times as long by SSEARCH. An implementation of Smith-Waterman showing six-fold speed-up on Intel Pentium III processors was reported by Rognes and Seeberg (2000).

1.3.2 Heuristics

To overcome the computational intensity of rigorous dynamic programming, fast alternatives have emerged using heuristics. Programs of the BLAST and FASTA packages (Altschul *et al.*, 1997; Pearson, 2000) are the most popular examples in this category. They achieve much higher speed in similarity searches by sacrificing some sensitivity. The short-cut consists of first scanning the sequences for small "words" (two or three residues) that match between the query and the database. Only these initial hits are subsequently expanded into so-called "high-scoring segment pairs" (HSPs). In certain cases

this might cause a weak hit to be missed that would be found by a rigorous search. The speed difference for searches between the methods are approximately 1 (BLAST) : 10 (FASTA) : 100 (SSEARCH).

Many studies have been dedicated to the assessment of sequence similarity searches through both rigorous and heuristic approaches (e.g., Pearson, 1995; Shpaer *et al.*, 1996; Brenner *et al.*, 1998; Pearson, 2000). Up until 1998 the BLAST program always performed the worst, that is, it detected the least number of homologues, mainly because no gaps were allowed in the alignment. FASTA performed better but not as well as Smith-Waterman implementations. The heuristics have been constantly improved, and particularly since the introduction of gapped BLAST, the differences in sensitivity have nearly disappeared (Pearson, 2000) with little concomitant loss of speed in the case of BLAST.

1.3.3 Multi-step searches

Beside the previously described methods, more sophisticated approaches have been developed which involve several steps. Hits from an initial homology search are combined to form objects such as Hidden Markov models (HMMs, Krogh *et al.*, 1994) or scoring matrices, which are then reapplied to the sequence database for retrieval of further hits. Although many of the HMM programs, such as SAM-T98 (Karplus and Hu, 2001), are primarily designed for multiple alignments, they are also suitable for homology searches, since an intermediate step involves the generation of protein families. Position-specific iterated BLAST (PSI-BLAST; Altschul *et al.*, 1997) creates a position-specific scoring matrix from a multiple alignment of the highest scoring hits in an initial BLAST search. This profile is used repeatedly in further queries which leads to a refinement and ideally ends in convergence. The extra effort that arises from training HMMs and multiple executions of PSI-BLAST pays out in better sensitivity, i.e. more distant relationships between sequences can be detected.

1.3.4 Integrated supercomputers

According to Seymour Cray, the father of supercomputers, "a supercomputer is defined simply as the most powerful class of computers at any point in time", a definition which can be stretched to fit a wide range of computer types. This section deals only with commercially produced supercomputers in which all components are combined into one closed compound. Traditionally these were of the PVP architecture (parallel vector processors; Strohmaier *et al.*, 1999) which benefits from the fact that scientific codes make extensive use of vector operations. Through a special vector processing unit (VPU) several steps that normally would be implemented as an explicit loop in machine code are combined into a single instruction implemented in hardware.

Vector architecture has now been nearly displaced by MPPs (massive parallel processors), where up to several thousand scalar processors are combined into one unit (Strohmaier *et al.*, 1999). Depending on their implementation these systems can be further subdivided into single instruction stream, multiple data stream (SIMD) and multiple instruction stream, multiple data stream (MIMD) parallel computers. This differentiation is made based on the processors which either execute the same instruction at the same time (SIMD) or carry out different instructions independently (MIMD). Accordingly, either threaded SSEARCH or specialised implementations of the SW algorithm (Yap *et al.*, 1995) can be run. The main manufacturers of high performance computers are Cray, Fujitsu, NEC, Hitachi, HP, and IBM.

1.3.5 Clusters

Compute clusters consist of multiple standalone computers connected together via a network system. The separate units can range from stripped-down PCs without even a hard-drive to multiprocessor computers, from dedicated nodes to workstations that are used when idle. The first experiments were carried out in the early 1980s at NASA (Castagnera *et al.*, 1994) from which even-

tually the Beowulf concept arose. It was originally envisioned as a number of commodity computers under the control of one master host, running a free operating system, which are connected via a small area network (Sterling *et al.*, 1995). The definition has now widened and includes high-speed switched networks and complete vendor-preconfigured rack-mounted systems with either Linux or Windows as an operating system. The term "pile of PCs" (PoPC, pronounced "pop-cee") is often associated with Beowulf clusters (see Figure 1.1). Constellations consisting of more sophisticated nodes spanning

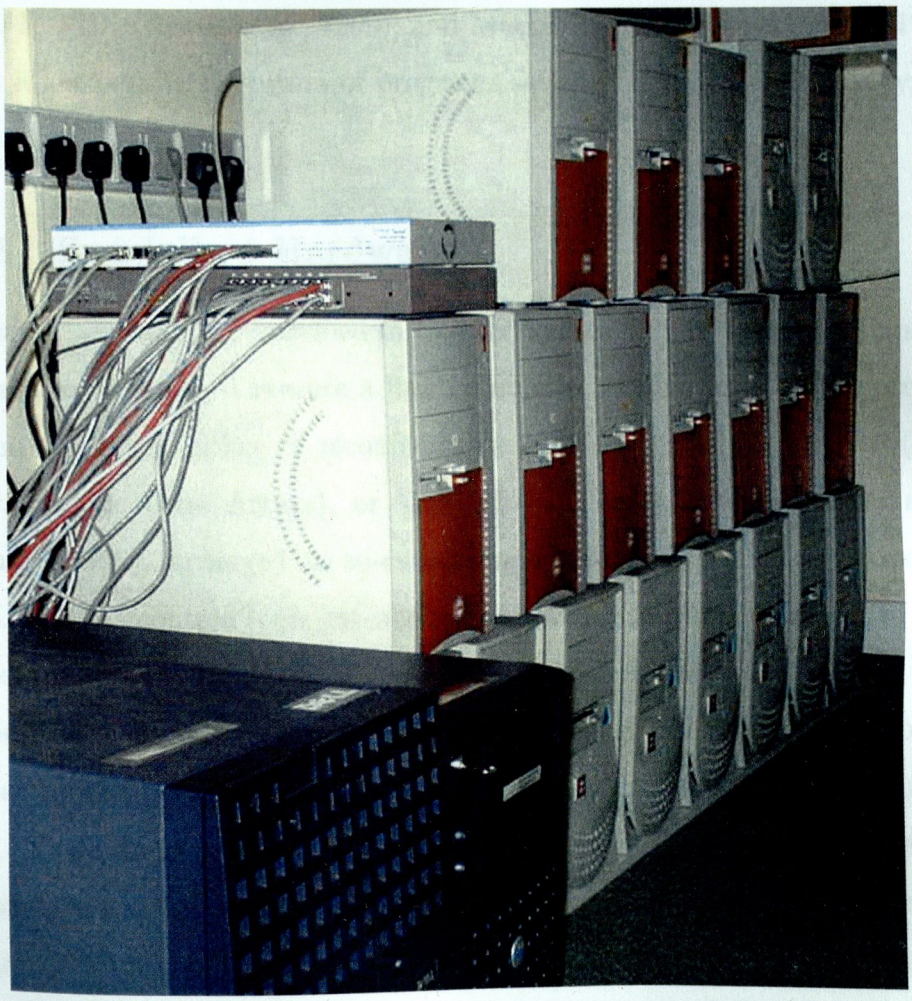


Figure 1.1: Our local version of a "Pile of PCs". The second version of our cluster came into existence at the beginning of the year 2000. The big dark box presents the Dell server; above it are the two switches, and the 20 clients are extending towards the back of our climatised machine room.

large area networks are also referred to as a "network of workstations" (NOW).

The evolution of these types of system was facilitated by the decreasing cost of PCs and the development of Linux (`http://www.linuxdoc.org`), a robust and free UNIX-based operating system targeted for microprocessors used in personal computers. Clusters consisting of more than a hundred computers (Gee, 2000) are not uncommon anymore. With the number of nodes, the peak performance grows as well, which is reflected in the increasing number of clusters that make their way into the Top500 list of supercomputers (Dongarra *et al.*, 2000). A variety of software models exist, such as PVM (Sunderam, 1990), MOSIX (Barak *et al.*, 1999), and MOLLUSCS (Jongeneel *et al.*, 1997), which allow parallel execution of programs such as SSEARCH on this type of architecture.

1.3.6 Specialised Hardware

The computer systems described in this section are highly specialised and are normally configured to execute a limited number of algorithms. The processing units consists either of reconfigurable hardware, such as FPGAs (Field Programmable Gate Arrays), or VLSI chips (Very Large-Scale Integration; Lavenier, 1996), arranged in so-called systolic arrays (Kung and Leiserson, 1980). FPGAs contain logic gates that can be hardware-programmed to carry out certain tasks. In VLSIs the algorithm is hardwired into the chip. This makes VLSIs faster but less flexible than FPGAs. Both type of chips are used as special-purpose functional units that very efficiently carry out limited tasks. Their usage in bioinformatics is often restricted to highly parallellised versions of the SW search algorithm. Through special design and programming techniques VLSIs and FPGAs can be optimised to perform Smith-Waterman at much higher speeds than general purpose processors. Besides the dedicated logical design, a speed advantage can also be achieved through utilisation of more than one processing unit in parallel. Two popular examples of this type of dedicated hardware are the Biocellerator developed by Compugen

(`http://www.compugen.co.il`), and GeneMatcher from Paracel (Shpaer *et al.*, 1996), in which up to 27,648 FPGAs are configured for rapid SW searches.

1.3.7 Comparison

Surveys have shown that SW searches in particular can be run fastest on dedicated computer systems with specialised hardware (Hughey, 1993; Shpaer *et al.*, 1996). These systems also offer the best price/performance ratio (Hughey, 1996; Jongeneel *et al.*, 1997). A disadvantage lies in their inflexibility to run other programs as well. FPGA-based machines might overcome this limitation by reprogramming of the processing units, but currently no system exists that provides a broad range of useful algorithms. Although dedicated hardware performs superbly for SW searches, many problems are sufficiently served by results produced by heuristics. Following recent technological advances these can now be produced in acceptable times on standard computers.

Industrial high performance systems offer great computing power, which can be used for a broad range of applications, but at a high price. Maintenance contracts add approximately 5 - 10% to the price of the systems per year (Warren *et al.*, 1997). In addition to this, the manufacturers have often developed specialised operating systems to which administrators, users, and programs need to be adjusted. The compact design of integrated systems allows for maximum communication speed between processors and memory and data sharing between parallel processes but makes upgrading of separate parts difficult and pricey.

As surveys have shown (Warren *et al.*, 1997; Ridge *et al.*, 1997; Jongeneel *et al.*, 1997), price/performance ratios for clusters are far better than for parallel machines. The current trend in ever-cheaper and faster PC components will further increase the gap. Another advantage is offered through the control that builders/users have over their system. A lot of scientific programs are available for Linux and Windows. These can be run immediately on each node

of a cluster. For automatic parallelisation, wrapper programs are required that organise the distribution of tasks and the assembly of results. Alternatively, programming models such as PVM (Sunderam, 1990) or MPI (Snir *et al.*, 1996) can be used to deal with the data and memory partitioning present in clusters. Another problem caused by clusters is that of energy consumption and heat emission. Since the components are by definition not purpose-built, large collections of PCs require extensive electrical supply and create so much heat that special air-conditioning might become necessary. On the other hand, the modularity of clusters allows a flexible design and effortless expansion. Every single part of each computer can be exchanged or updated without endangering the integrity of the system. More computers can be easily added to increase the overall computing power.

The system of choice for high-performance computing depends strongly on the availability of money and computer expertise as well as on the applications to be run. For very limited purposes specialised hardware offers a good solution. Speed-up for a wide range of problems can be comfortably achieved through commercial supercomputers where funding allows it. Beowulf systems are the least expensive high-performance systems and constitute the solution chosen by our group.

1.4 Tools for genome comparison

It is possible to carry out genome comparison *in vitro*. A technique called cross-species color banding (also termed Rx-FISH) was developed by Muller *et al.* (1998). It allows the detection of genome rearrangements through the generation of a distinctive colour banding pattern throughout the genome. The results, however, lack the necessary resolution to provide details on the gene level of the detected regions. This project focusses exclusively on software methods for genome comparison.

1.4.1 Software

Several tools for the comparison of large sequences exist already such as PAGEC (Courtois and Moncany, 1995), SIM3 (Chao *et al.*, 1997), and MUMmer (Delcher *et al.*, 1999). All of them work on the basis of large-scale DNA alignment. Their implementations vary from index algorithms via dynamic programming methods to the use of suffix trees. All approaches achieve alignment of large sequences in very short time but struggle with the amount of memory required for their calculations. For example, MUMmer requires about 4 Gigabytes working memory for the comparison of two genomes of 100 Mbp length. Scaling this to the human genome which has a size of approximately 3,200 Mbp would place enormous demands onto the required computer equipment.

Sequence comparison at the DNA level is useful for detection of highly similar sequences with only small differences, such as single nucleotide polymorphisms (SNPs). With regard to whole genome comparison as envisaged for this project such an approach would be very likely to miss interesting regions of remote homology. Protein sequences in general show a much higher degree of similarity than the corresponding DNA sequences, due to the effect of synonymous basepairs in the first and third codon positions. Pearson (2000) estimates that sequence similarity searches at the protein level can be five- to ten-times more sensitive than at the DNA level.

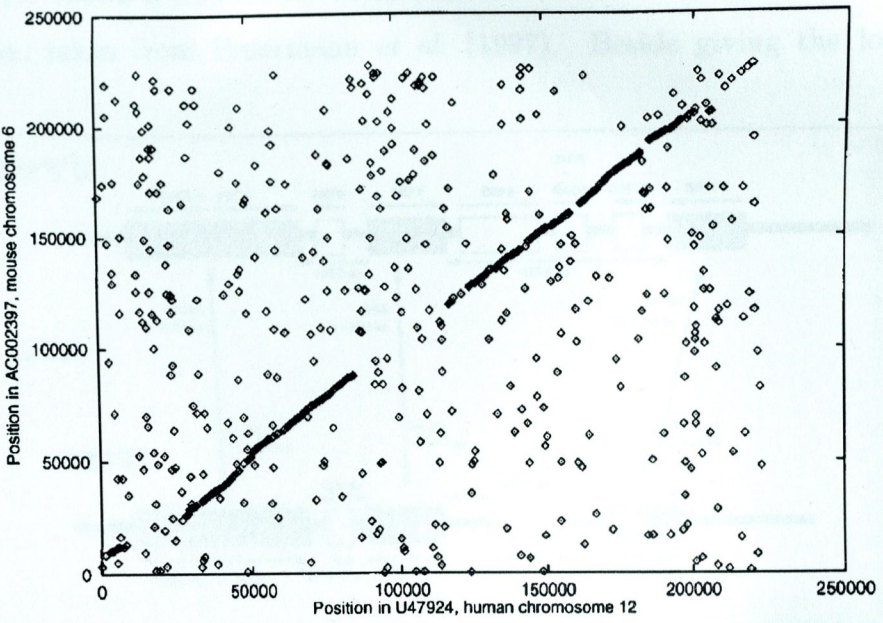


Figure 1.2: Dot-matrix plot between human and mouse segments. Squares indicate matching segments of 15 bp length. Conserved stretches of sequence similarity show up as diagonals, consisting of dense series of squares (alignment calculated by MUMmer; figure taken from Delcher *et al.* (1999))

1.4.2 Graphical presentation

The standard method for illustrating similarities between large sequences, where an alignment view at the basepair level proves unsuitable, is the dot-matrix plot. In this type of presentation, homologies are displayed as dots in a matrix defined by the two sequences that are under investigation. Stretches of neighbouring regions on both axes that share homology are revealed through a series of dots in the plot, which becomes visible in the form of a diagonal (Figure 1.2). Regions of homology in the same order on both sequences are clearly visible as diagonals. In case of chromosomal rearrangements, which interrupt the sequence order, these lines would be dispersed. For the presentation of distant homologies the dot-matrix presentation offers only limited practicality.

Good examples of the graphic display of duplicated blocks that procure the necessary balance between detail and abstraction can be found in the literature.

Figure 1.3 shows a duplicated block between chromosomes VII and II in *S. cerevisiæ*, taken from Feuermann *et al.* (1997). Beside giving the location

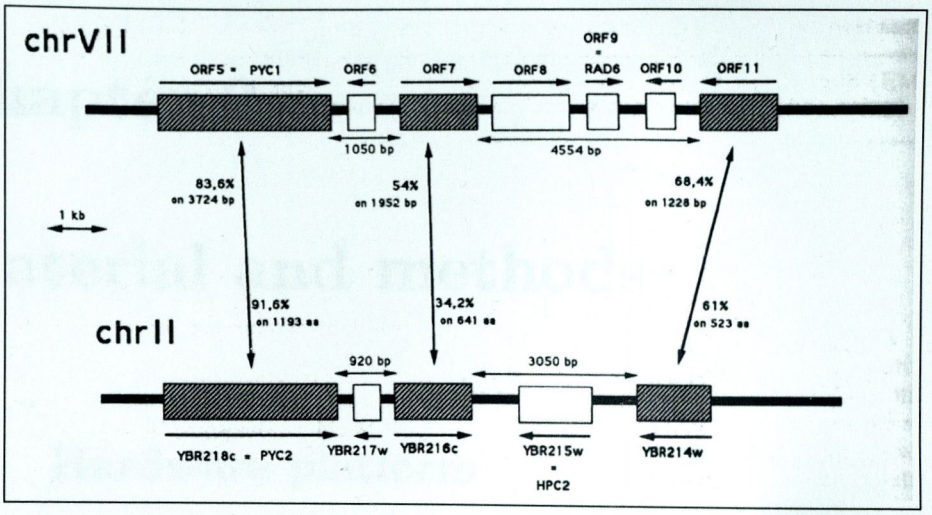


Figure 1.3: Example of a duplicated block in *S. cerevisiæ*. Representation of the *PYC* clusters on chromosomes VII and II in *S. cerevisiæ*. Sections from both chromosomes are shown that include pairs of homologues interspersed with unique genes. The grey boxes represent the genes with significant sequence identity, the white boxes represent genes that are unique. The degrees of identity at amino acid and nucleotide levels, the names and directions of transcription , as well as the distance between the different duplicated genes are indicated (taken from Feuermann *et al.*, 1997).

of genes, their names, and their direction of transcription, it also provides information about the similarity between matching sequences. Due to the usage of lines for the indication of homologous genes, regions where the gene order is not preserved stand out equally well.

Figures in journals are static and are often laboriously created by hand. An automatic method is desirable that produces interactive presentations and provides the mechanisms for navigating and zooming to explore regions of interest at different levels of resolution. This project presents an approach to that, which is described next.

Chapter 2

Material and methods

2.1 Hardware platform

The computing power of the workstations present in the lab at the beginning of the project was not sufficient to carry out whole genome scale similarity searches in acceptable time. Limited finances as well as familiarity with PCs and Linux were the decisive factors in the choice for a Beowulf cluster. In addition, a self-built system offers the opportunity to be easily expanded and therefore to grow with the scale of a problem. Since the amount of molecular data is enlarging at an ever increasing pace, this was an important factor to be considered.

2.1.1 Cluster components

The lab's original cluster was built in November 1999. The setup consisted of a dual processor server, acting as the master, and ten clients. The parts list can be seen in Table 2.1 overleaf. A switched FastEthernet LAN (Local Area Network) was set up with all the computers connected to a 16-port 100Mbit switch (Prime PS-1016, approx. £600[a]) via category 5 twisted wire network

[a] 1 £= 1.26974 Euros

cables (approx. £70). This brought the total cost of the system to less than £22,000.

Table 2.1: Parts list for first Beowulf setup

parts	server	client
main-board	Supermicro 2DGE	Matsonic MS7101C
case	Supermicro 760A	standard
CPU	2 x PII Xeon 500 MHz, 512 KB cache	Celeron 400 MHz, 128 KB cache
RAM	2 x 256 MB ECC	256 MB ECC
graphic	ATI XPERT, 8MB	S3 Trio 4 MB
NIC	Intel EtherExpress 100 Pro, 3Com 3905C	Intel EtherExpress 100 Pro
controller	Adaptec 2940 U2W	-
hard-drive	4 x 50 GB SCSI Seagate Barracuda	-
floppy	standard	standard
CD-ROM	standard, 40x speed	-
mouse	PS/2	-
keyboard	PS/2	-
monitor	CTX 19"	-
cost[a]	approx. £12,000	approx. £900

[a]prices in Irish Punt, dating from November 1999 (1 £= 1.26974 Euros)

One year later more funding became available for the expansion of this configuration. The master was replaced by a four-processor server of the Dell Poweredge family (`http://www.dell.ie`). The clients were upgraded to 800 MHz Pentium III processors, and ten more clients, with a configuration listed in Table 2.2 overleaf, were added. Thus, for an additional £32,000, the sum of processor speeds was nearly quadrupled from 5 GHz (2 x 500 MHz + 10 x 400 MHz) to 18.8 GHz (4 x 700 MHz + 20 x 800 MHz). An additional 16-port 100Mbit switch (Prime PS-1016, approx. £600) was added to connect the new clients to the second network card of the server.

Table 2.2: Parts list for additional cluster parts

parts	server	client
main-board	Dell	Matsonic MS7117C
case	Dell	standard
CPU	4 x PIII Xeon 700 MHz, 512 KB cache	Intel PIII 800 MHz, 256 KB cache
RAM	4 x 512 MB ECC	256 MB ECC
graphic	ATI XPERT, 8MB	S3 Trio 4 MB
NIC	3 x Intel EtherExpress 100 Pro	Intel EtherExpress 100 Pro
controller	RAID	-
hard-drive	72 GB RAID system	-
floppy	standard	standard
CD-ROM	standard, 40x speed	-
mouse	PS/2	-
keyboard	PS/2	-
monitor	Dell 17"	-
cost[a]	approx. £22,000	approx. £900

[a]prices in Irish Punt, dating from December 2000 (1 £= 1.26974 Euros)

The clients do not contain hard-drives, which brings several advantages:

- reduced cost of clients
- less noise and heat emission from clients
- fewer points of failure
- less administrative work

The last point in particular provides invaluable benefits: The operating system on the clients can be automatically kept in synchronisation with the one run on the server. Since the clients load their operating system through the network from the master, the easiest option is to export the same basic structure and files that are used on the server. Though for each client, some small modifications during boot-up is necessary to adjust network configuration and temporary files. Through this mechanism any changes or addition of programs on the master

will be proliferated through the whole cluster immediately. On the downside, this creates extra network traffic causing the connection between the server to the switch to become a likely bottle-neck. In our case however, the bandwidth provided by FastEthernet was sufficient to overcome these difficulties.

2.1.2 Network Configuration

Two private networks were set up to connect ten clients each through a switch to two network cards of the server. The current network configuration of the cluster is shown in Figure 2.1 on page 28. To prevent unbalanced usage of network parts, the clients were numbered sequentially and distributed in a nested manner to both switches (see Table 2.3 overleaf). Each switch is connected to a network card on the master (IP addresses 192.168.0.254 and 192.168.1.254, respectively) and on the file server (IP addresses 192.168.0.253 and 192.168.1.253, respectively). If a multi-process program is executed on the cluster, the first process is by default started on the client with the lowest number in its name (i.e. dc_001), and the rest are spread to clients with higher numbers in a sequential order (i.e. dc_002, dc_003, etc.). For sequence similarity searches, loading the database into memory of a client is the most data intensive step. The described network topology attempts to balance this load evenly over both switches. Regardless of the number of clients involved, as long as they are addressed in a sequential order the network traffic will be equally distributed amongst the two networks.

Table 2.3: Naming and numbering system of clients

Name	IP Address	Switch
dc_001	192.168.0.1	1
dc_002	192.168.1.2	2
dc_003	192.168.0.3	1
dc_004	192.168.1.4	2
dc_005	192.168.0.5	1
dc_006	192.168.1.6	2
dc_007	192.168.0.7	1
dc_008	192.168.1.8	2
dc_009	192.168.0.9	1
dc_010	192.168.1.10	2
dc_011	192.168.0.11	1
dc_012	192.168.1.12	2
dc_013	192.168.0.13	1
dc_014	192.168.1.14	2
dc_015	192.168.0.15	1
dc_016	192.168.1.16	2
dc_017	192.168.0.17	1
dc_018	192.168.1.18	2
dc_019	192.168.0.19	1
dc_020	192.168.1.20	2

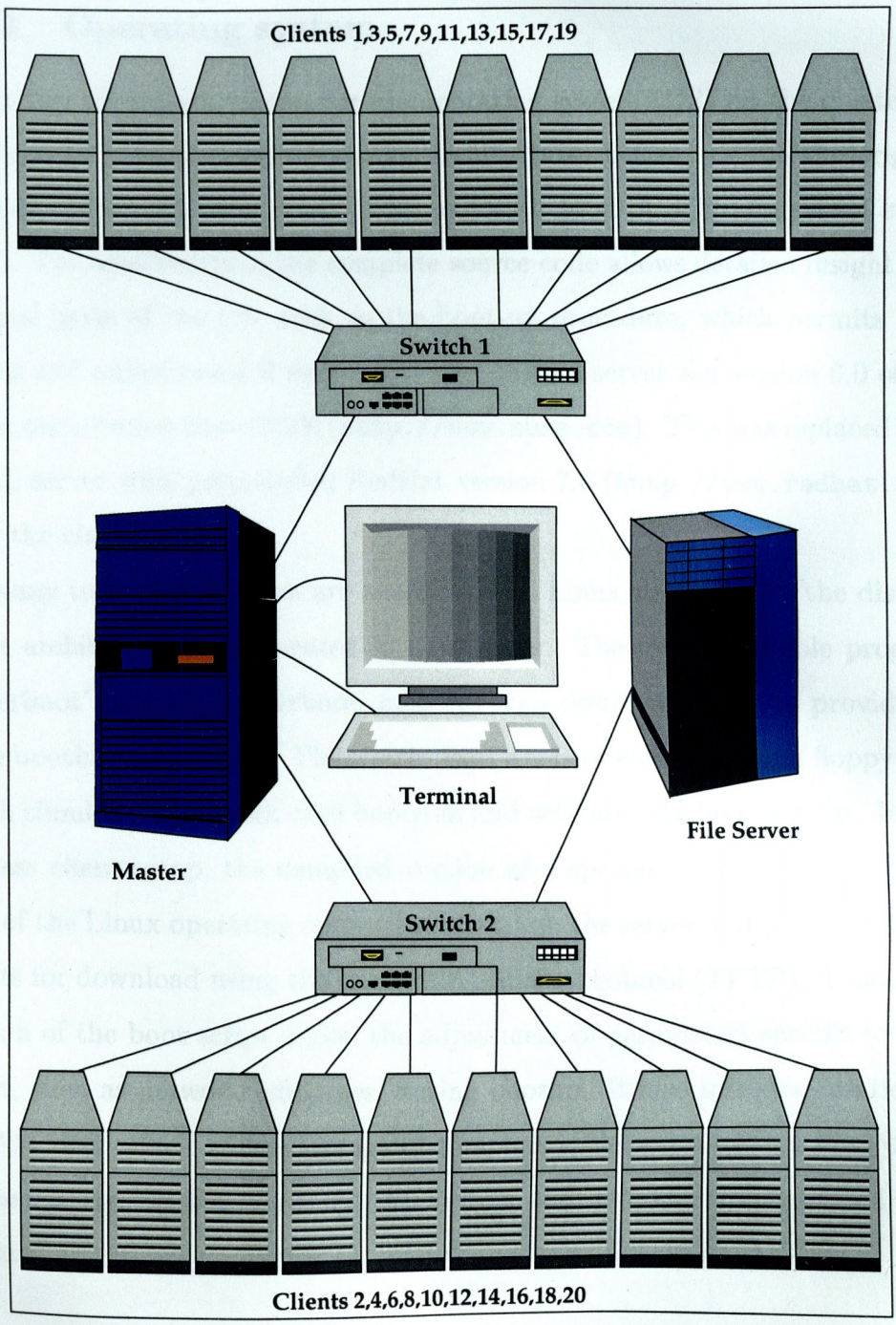


Figure 2.1: Local cluster configuration. Two sets of ten diskless clients are connected through FastEthernet switches to the master and the file server. Access and control of the cluster is enabled through the master server.

2.2 Software tools

2.2.1 Operating system

Linux was the natural choice for the operating system (OS) on the cluster. It has been successfully used on similar architectures before, e.g. on the original Beowulf cluster (Sterling *et al.*, 1995) and with MOLLUSCS (Jongeneel *et al.*, 1997). The availability of the complete source code allows detailed insight into integral parts of the OS, such as the boot-up procedure, which permits fine-tuning and adjustments if necessary. The original server ran version 6.0 of the Linux distribution from SuSE (`http://www.suse.com`). This was replaced by a DELL server with preinstalled RedHat version 7.0 (`http://www.redhat.com`) after the cluster upgrade.

Many tools and features are available with Linux that support the diskless client architecture implemented in our cluster. The freely available program 'Etherboot' (`http://etherboot.sourceforge.net/`) was used to provide remote booting capabilities. The clients load a software image from a floppy disk which simulates a network card bootrom and activates the boot process. In our diskless client setup, the compiled version of a specially prepared kernel (the core of the Linux operating system) is placed on the server and provided to the clients for download using the trivial file transfer protocol (TFTP). A modified version of the boot script allows the adjustment of parameters specific to each client, such as network addresses, during bootup. It also initiates creation of a RAM disk which allows storage of temporary files on a virtual file system located in the working memory. All system and user files are imported from the master and the file server through the network file system (NFS).

2.2.2 Parallelised sequence similarity search

To harness the power of a Beowulf cluster, a task must be broken down into sub-tasks that can be processed on the clients in parallel. In the case of se-

quence similarity searches this can be accomplished by searching different query sequences against the same database on each cluster node. A wrapper program is necessary for the handling of database organisation, remote execution calls, consistency checks and assembly of results. The structure of the three different files involved is as follows:

- **input:** a plain text file containing a list of aminoacid or DNA sequences in FASTA format (one-line header information, starting with '>', and the sequence string)
- **database:** usually a compressed binary file derived from a list of aminoacid or DNA sequences in FASTA format
- **output:** a plain text file containing the results of the sequence similarity search

An ideal programming language for this purpose is found in PERL (Larry Wall, 1996). Due to the lack of a compilation stage, it allows for rapid prototyping. Extensive software libraries enable communication between computers, and tight integration with the UNIX operating system guarantees smooth operability on Linux.

The system employed on our cluster was inspired by and is derived from the **MO**dular **L**ow-cost **L**inux-based **U**nified **S**equence **C**omparison **S**ystem (MOLLUSCS) developed by Jongeneel *et al.* (1997). It consists of a PERL script which is run from the master server. Our wrapper, entitled '`wrapid`' (**w**rapper for **r**apid **p**rallelised **i**nstruction **d**ispatching), is targeted towards parallelisation of the BLAST and FASTA programs, which are the primary methods used in our analysis.

Like in MOLLUSCS, care was taken that the command syntax for the search programs is preserved. Thus, users can continue to apply parameters they are accustomed to. Additionally, program calls can be checked on the server on

their own to ensure all necessary files are in place. After that, a simple prefix with the wrapper script will automate parallelisation.

Several options specific to `wrapid` are available which control its execution. These are listed in Table 2.4 and explained below.

Table 2.4: Command line options for `wrapid.pl`

option	effect
--dir <string>	output directory
--db <string>	explicit path to database
--q <string>	explicit path to query file
--cl <string>	specification of clients
--load <float>	maximum load allowed on clients
--perc <float>	divider for query fractions
--single	force single query per client
--no_assembly	prevents assembly of results
--compress	compresses results file(s)

For each parallelised run `wrapid` creates a directory in which results and temporary files are stored. By default a name consisting of the current date and time is used but it can also be specified using the `--dir` option. Currently, the filenames for each query and database are extracted from parameters submitted to the FASTA or BLAST programs. However, since these options differ for each search program, `wrapid` provides the `--q` and `--db` option to allow consistent specification of paths to the query file and the database, respectively.

During initialisation, the wrapper puts all known clients through several tests. If only a subset of clients is required, these can be specified using the `--cl` option. The tests involve network connection checks, load checks to make sure resources are available, access checks to ascertain the remote accessibility of files, and a check for the communication capability back to the server. Only clients that pass all these tests will be used. In a multi-user environment it makes sense to check for the load on each client and to only use those computers that are idle. However, this can be overridden with the `--load` option being

set to a value of 1.0 or higher.

Once the number of available clients is established, a temporary query file is prepared for each node. A limit for the number of residues r per client is calculated through the following equation:

$$r = \frac{S * f}{N} \tag{2.1}$$

Here, S stands for the sum of the length of all remaining queries, N denotes the number of available clients, and f is a factor between zero and one which prolongs the split up of the queries to prevent clients from running idle prematurely. By default f is set to 0.5 which means that initially, a subset of queries with lengths summing up to half of the sum of all lengths is distributed to the clients for processing. The `--single` option forces the program to send only one query to each client. This is the current default for FASTA programs, since they cannot handle input files containing multiple sequences.

Execution of sequence similarity searches in parallel on the cluster nodes is implemented by the `fork` mechanism in PERL and the remote shell program on Linux ('`rsh`'). A `fork` statement generates a child process that inherits the environment of the parent process and continues to run independently. The remote shell command enables execution of programs on machines connected through a network. Information stored in '.rhost' files obviate user authorisation. For networks that are not completely private the more secure version `ssh` can be used.

Once a client has finished its search, the results are checked for completeness. This became necessary when it was realised that sometimes processes are prematurely terminated, probably due to overheating, and evoke the wrong impression of a finished run. If the check is unsuccessful, the client is barred from further participation, and its last batch of queries is returned to the stack of unfinished queries. Otherwise, another remote shell call triggers the client to signal its availability for more jobs back to the server. This interprocess

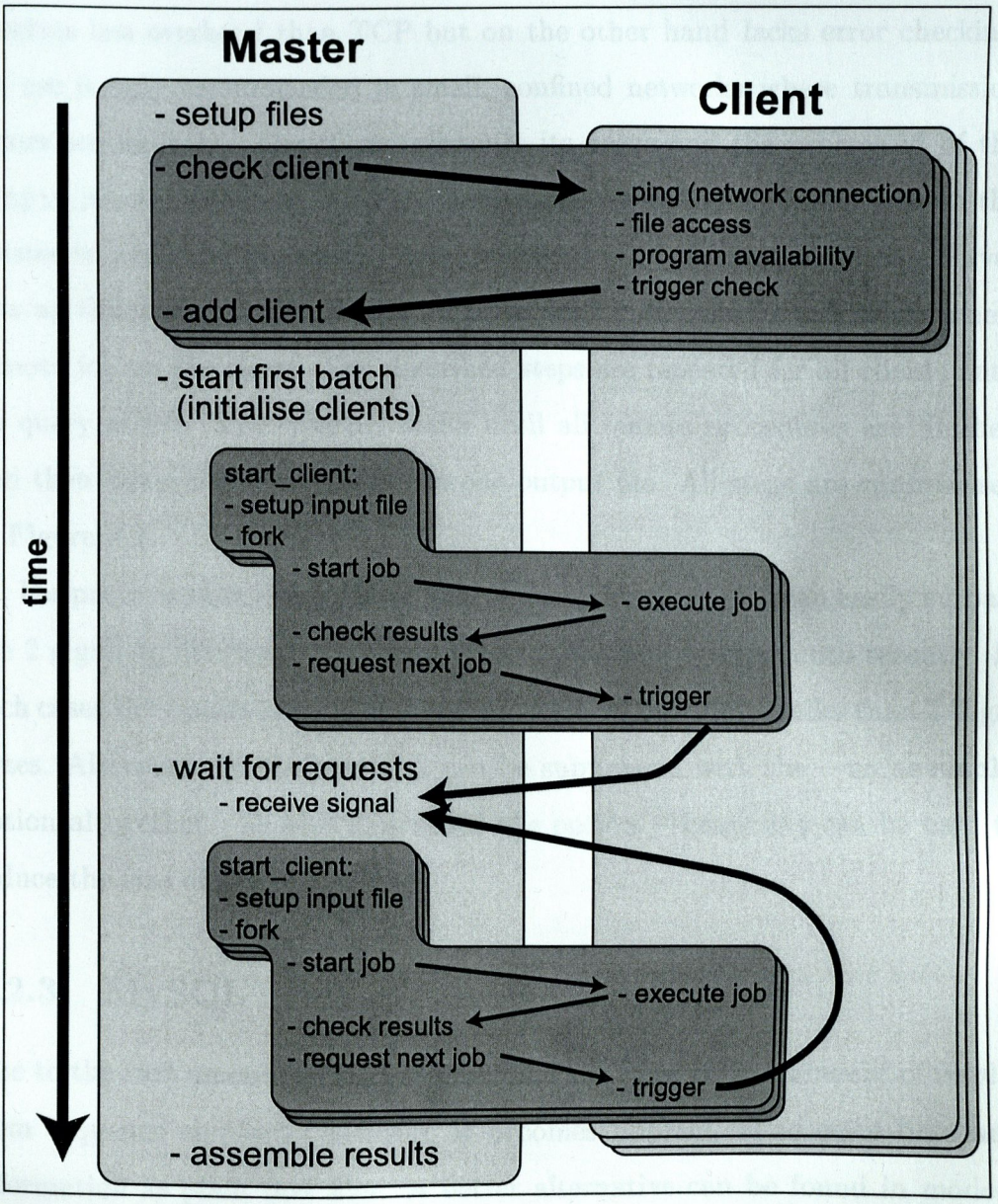


Figure 2.2: Execution steps of `wrapid`. The diagram shows key stages in the process carried out by `wrapid`. The execution is controlled from the master node. Dark grey boxes indiate parallelised steps that are carried out concurrently on multiple client nodes.

communication is realised through UDP or TCP packages sent through sockets. The socket mechanism was implemented in UNIX to allow programs to communicate, either on the same machine or across a network. The UDP protocol involves less overhead than TCP but on the other hand lacks error checking. Its use is only recommended in small, confined networks where transmission errors are unlikely. The client transmits its name and the process id of the `wrapid` job to the server. This mechanism allows multiple parallel jobs on the cluster to run simultaneously. After verification of the process id, the server sets up the next query file based on equation 2.1 on page 32 and starts a new remote job on the client. The described steps are repeated for all clients until no query is left. The wrapper waits until all remote procedures are finished and then assembles the results into one output file. All steps are summarised in Figure 2.2.

For projects that involve large sets of data, the results file can easily surpass the 2 gigabyte file size limit, which existed on Linux systems until recently. In such cases the results are automatically assembled into files smaller than 2 Gigabytes. Alternatively, the assembly can be suppressed with the `--no_assembly` option altogether. To save disk space the option `--compress` can be used to reduce the size of the results files.

2.2.3 MySQL

Due to the vast amount of data at hand and the even greater amount of results from sequence similarity searches, it becomes impractical to store data and information as plain text files. A better alternative can be found in modern database systems. They store data in compressed form and allow the creation of indices that enable retrieval of information from arbitrary location within the data set at high speed. The MySQL database system (`http://www.mysql.com`) is freely available for many operating systems, including Linux. All data files at Ensembl, the primary source for human annotation data, are available as

MySQL dumps and can be easily integrated into a local installation. The flexibility of an SQL system allows for data updates, which is particularly important in regard to the unfinished state of the human genome. An additional benefit arises from the available SQL interface for PERL programs. All of the programs developed for this project that need to store or access data interact directly with the MySQL database. This speeds up greatly the handling of large amounts of information and reduces necessary disk space.

2.3 Methods for analysis of genome data

The purpose of the methods developed in this section is to detect blocks of duplicated genes within a genome and to present these in an appropriate way. It is necessary to first remove tandem repeats which might obstruct the search for larger blocks. For the block detection a new algorithm has been developed that incorporates an extensive filter mechanism, which aims to separate true homologues from proteins with random similarity. For the storage and presentation of the results special means are required that allow fast access and have the capability to show widely differing levels of resolution.

2.3.1 Collapsing of tandem repeats

A preliminary step of the block detection involves the treatment of tandem repeats, multiple copies of a gene in close vicinity, most of which were probably formed recently through unequal crossing-over. Several reasons validate their detection:

- An occurrence of multiple copies of a gene suggests that it plays an important role and is needed in unusually large amounts. Such findings can point towards targets for further interesting research projects.
- Tandem repeats can lead to an inflation of block sizes, particularly when they originated from two homologues (see Figure 2.3).

- Most importantly, tandem repeats without related matches increase the distance between linked gene pairs and might lead to a transgression of the allowed gap size. This would have the undesirable effect of a block disruption, due to an evolutionary unrelated event.

This method detects groups of very similar genes in close vicinity and replaces them by a single representative. Information about each cluster is stored in a database and can be integrated into the results.

A PERL script was developed, which implements a method similar to the one used by Vision *et al.* (2000). It requires relative mapping positions (chromosome and rank) for proteins as well as their results from sequence similarity searches. For each protein the program scans the neighbourhood of a user-specific range for hits with an expectation value below a certain threshold. If a match is found the protein with the shorter sequence is removed from the map and stored in a separate database. This step is repeated until the neighbourhood of a protein contains no further similar hits. After all proteins are processed, a new map is made, which is then subjected to the actual block detection.

2.3.2 Block detection

For the detection of blocks a brute force algorithm was implemented in a PERL script, which means that without using any possible shortcuts, all of the available gene pairings are tested for potential paralogous clusterings. The data sources for the program comprise the following:

- a list of proteins and their relative genomic location (chromosome and rank)
- results of sequence similarity searches for the proteins

In combination with a variety of parameters, this information is processed to detect groups of similar genes in two or more distinct locations. A block consists

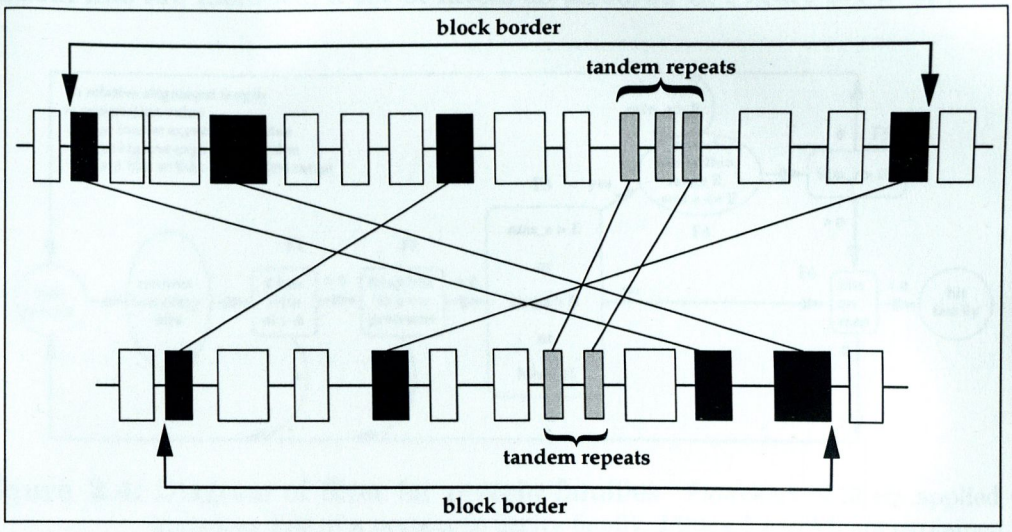


Figure 2.3: Definition of a block. Two regions with possibly duplicated genes (black boxes) are shown. Block borders are defined by the outermost linked genes (black). The number of continuous stretches of intermediate genes (white) must lie below the gap size. If tandem repeats (grey) are counted separately, this might lead to an transgression of the gap size and a subsequent split of paralogous regions. Cases where tandem repeats are paired between two regions can lead to an inflation of block size.

of two non-overlapping chromosomal regions whose boundaries are defined by the outermost genes that are linked between them (see Figure 2.3). The results strongly depend on the values chosen for the parameters listed in Table 2.5. The role that each parameter plays will become apparent from the following

Table 2.5: Available parameters for block detection

parameter	description
E	expectation threshold
G	maximal gap between linked proteins
R	maximal expectation range for homology hits
H	maximal number of hits within range
A	percentage of alignment overlap

description of the program.

The first step of the block detection consists of establishing protein families. To ensure that a maximum number of homologues and a minimum number of

random hits are included, a set of filters as pictured in Figure 2.4 is applied.

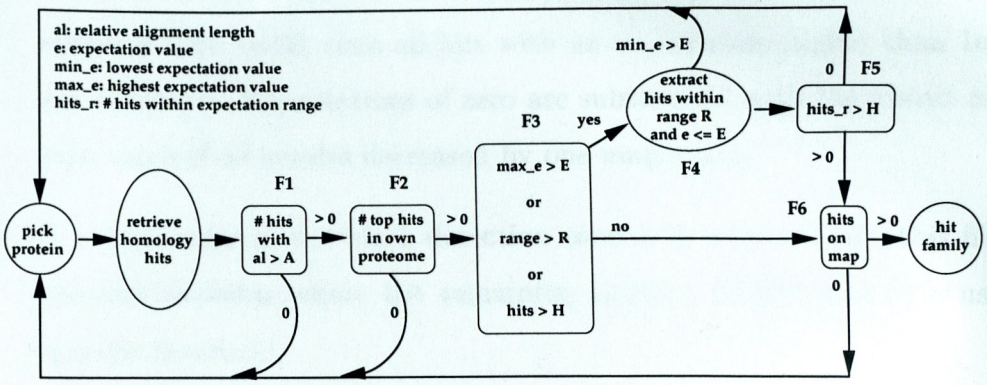


Figure 2.4: Diagram of filter for protein families. Flowchart of filters applied to homology hits of a protein to derive family. Filters F1 to F6 are explained in the text. Descriptions of parameters A,E,R,H are listed in Table 2.5 on the page before

The following listing provides a detailed description of the filters in sequential order:

F1: Alignments are required to overlap a certain percentage of the length of the longer protein sequence. This helps to avoid matches between proteins that only share similarity amongst short stretches, such as motifs or small domains.

F2: If sequences from other species are available, they can be used in the sequence similarity search as a natural orthology threshold. To filter out duplications that occurred before the divergence of different species, only families whose top hits come from the same proteome are accepted.

F3: This filter examines three properties: the highest expectation value, the range of expectation values and the number of hits. If any of those exceeds the threshold specified by according parameters (Table 2.5) the family undergoes further tests.

F4: Families with their best hit above the expectation threshold are discarded. To remove members of low significance, only those within a certain range

from the lowest expectation value are accepted. For example, if the best hit has an expectation value of 1e-50 and the allowed range is specified with 1e-20, then all hits with an expectation higher than 1e-30 are discarded. Expectations of zero are substituted with the lowest non-zero value of all results decreased by one magnitude.

F5: To reduce noise in the block detection caused by large families this filter discards proteins where the remaining number of hits exceeds a user-specific threshold.

F6: The last filter only has an effect in unfinished genomes. It removes families where none of the members have a genomic location assigned.

The resulting protein families depend greatly on the specified parameters used in the filters. Suitable values have to be evaluated for each organism to which the algorithm is applied.

After establishment of protein families, the block detection is started. Beginning with the first position in a chromosome, each protein with at least one other family member is considered a link and a possible seed for a block. Given a seed protein p_s under investigation at location s and a family member p_h at position h, several steps are carried out, which are shown in pseudocode:

1. set temporary counter $i = s$
2. add h to the set of hit positions H
3. set gap $g_s = 0$
4. increase i and g_s until a protein at $s + i$ with a hit family is encountered or until $g_s > g$
5. if a protein with family is found,
   - 5.1 check for all hits if one or more can be found maximal g positions away from the range between H_{min} and H_{max}

5.2 if a hit is found

5.2.1 set gap $g_s = 0$

5.2.2 add h_{s+i} to H

5.3 continue at 4 on the preceding page

6. else if H_{min} or H_{max} were extended

6.1 set $i = s + 1$

6.2 continue at 3 on the page before

7. else store block information and continue with new link or seed

The jump from position 6.2 to 3 is important, because an extension of the range of hit positions can cause inclusion of genes that previously lay outside the range. Biologically, this rule reflects the allowance for rearrangement of gene order in duplicated blocks.

The above steps are carried out for all the hits of every protein. Due to the asymmetry of results from sequence similarity searches, not every block is detected from both perspectives. For example, a region on chromosome *a* might have links to a region on chromosome *b*. It is possible that some of the links in the reverse direction from chromosome *b* to chromosome *a* cannot be established, and as a result the corresponding mirrored block is reduced in size or not detected at all. Changes in sequence length and composition between homologues can lead to differences in the number of hits, the alignment overlap and the expectation values as well. If any of those values are close to the user-specific thresholds, the resulting differences in families can result in loss of mutual links. To provide consistent annotation, a merging process is added in which varying parts of blocks are combined and reverse pairings of one-directional links are enforced. This sometimes introduces proteins that produce pairings with an expectation value below the specified threshold. This is tolerable as long as the pairing adheres to the constraints imposed by the

parameters in at least one direction. The block size *sm* is determined by the number of paired genes amongst both paralogous regions. Sometimes a gene can be connected to more than one other gene, i.e., the links appear in shape of a 'V', 'N', etc.) these networks of pairings are only counted as one link. Finally, details of the final set of blocks are stored in a MySQL database.

2.3.3 Results access

The development of a user interface to access blocks was guided by the intention of making results easily accessible and explorable to as many users as possible. The world wide web provides an ideal platform with its enormous scope and its ability to deal with graphical contents. Therefore, with the help of a set of CGI scripts in PERL a web interface was created that enables browsing of block details stored in MySQL tables, as well as graphical presentation of blocks. Additionally, hyper-links to other resources on the Internet were added to allow integration of external information.

MySQL browser

A vast amount of detailed information about the detected blocks and the underlying information (genes, blast results, maps, etc.) is stored in MySQL tables. This enables quick access and additional features, such as ordering, grouping and selecting subsets of data. Biologists who might be interested in the results are not necessarily familiar with the usage of relational databases. A CGI script was developed to provide a user-friendly database interface which allows browsing of the available information and also integrates extra functionality through hyper-links. The access page provides input fields which can be used to query the database with any valid SQL statement. Standard keywords are preselected and any changes to the settings will be kept for the next query. The output is presented as an HTML table. Table headers include links to help files which describe the content. General data fields, such as block identifiers,

contain links which allow quick access to graphical presentation of blocks. An additional feature of web based access allows to incorporate SQL statements into a hyper-link which can be placed around descriptive text onto a web page. This allows easy and comfortable access to datasets through predefined queries. A whole module is dedicated to the identification of genes by keywords. All the information contained in protein names, gene names, description of genes, as well as related identifiers from external databases can be searched through a simple interface. This overcomes difficulties arising from different nomenclature and aids in searches of groups of genes. Information of matching genes is presented in a formatted HTML page that contains links to blocks, alignments and external information.

Graphics and links

'A picture says more than thousands words.' This was the incentive behind the development of several CGI scripts which allow exploration of duplicated regions on different levels of detail. On the top level up to four different genomic regions can be drawn together with covering blocks and the connections between them. Colour codes help to highlight genes with a large number of links and to distinguish between the different chromosomes they lead to. Extensive control is given in respect to the amount of features that are presented. For example, any combination of linked genes, intermittent genes, and tandem repeats can be shown. The direction of a chromosome strand can be changed (reverse-complemented) to untwist links between inverted blocks. The highest graphical resolution allows zooming into gene-level presentation of blocks. From there, hyper-links lead to pages with detailed information and alignments of proteins. Additional links provide access to more information available on the Internet, such as sequence or disease databases. Example web pages and graphics providing further details are provided in the following chapters.

Chapter 3

Method assessment with yeast

Evaluation of the capabilities of the new block detection method and of the results quality is the objective of this chapter. The well-established map of duplicated blocks in *Saccharomyces cerevisiæ* served as a benchmark. After an overview of the available yeast data, the steps for creating and improving new blocks and for their assessment are described. The chapter closes with a presentation of results from comparisons of various data sets and their discussion.

3.1 Aim

The fungus *Saccharomyces cerevisiæ* is a well-studied organism of the yeast family. Its genomic sequence was completed in 1996 through an international collaboration (Goffeau *et al.*, 1996) and yielded the first full sequence of an eukaryotic organism. Extensive analyses have been carried out in our group before the start of this project (Wolfe and Shields, 1997) (Seoighe and Wolfe, 1998) (Seoighe and Wolfe, 1999), which resulted in detection of duplicated blocks and their explanation through a polyploidisation event about 100 million years ago. Speculations about an older age for the event (200 – 300 Mya) or independent duplications of large segments instead (Friedman and Hughes, 2001) (Llorente *et al.*, 2000) still remain controversial (Wolfe, 2001).

The last update of the yeast map of blocks maintained in our laboratory identified 84 paired regions, which contain 905, or 16%, of the protein-coding genes (Seoighe and Wolfe, 1999). A region is defined by at least three interlinked genes which are located less then 30 positions apart from each other. Two further restrictions were imposed: (i) conservation of gene order (except for small inversions) and (ii) conservation of transcriptional orientation.

In the Seoighe and Wolfe (1999) study high-copy Ty elements (retrotransposons) were removed from the dataset, and low-complexity regions were filtered through SEG (Wootton and Federhen, 1996). After that, queries of all protein-coding genes against themselves with SSEARCH (Pearson and Lipman, 1988) where carried out externally on a parallel supercomputer. Blocks were manually edited through addition of gene pairs with weak alignment scores and links between non-coding tRNAs. Overlapping blocks without strong similarity hits were removed as well as duplications amongst sub-telomeric regions, which are well known for their high similarity between chromosomes. A graphical overview of the blocks was placed on a publicly accessible web site, together with detailed listings of blocks, dot-plots, and search facilities.

The extensive manual post-processing of the blocks yields high fidelity in their quality. This makes them an ideal benchmark against which to measure the capabilities and qualities of the algorithm described in the present work. To enable the exact comparison of results obtained from both methods, the same dataset as in Seoighe and Wolfe (1999) was used to detect blocks. Attempts were made to reproduce the existing results as accurate as possible. Additionally, a variety of programs was applied in the generation of protein families to evaluate the effect and importance of different types of homology searches. Finally, an updated version of the block map was created from the latest available dataset of the yeast genome.

3.2 Data

The original data used in the previous study of yeast duplications consists of mapping positions for 5790 protein-coding genes and the results of an all-against-all SEG-filtered homology search through SSEARCH (version 3.0t from March 1996). The duplicated blocks for *S. cerevisiæ* were downloaded from the web site (`http://acer.gen.tcd.ie/cgi-bin/khwolfe/blocks.pl?block=ALL`). They comprise of 84 regions which are made from 905 linked genes.

For an update of the blocks the latest GenBank entries for *S. cerevisiæ* were downloaded from `ftp://ncbi.nlm.nih.gov/genbank/genomes/S_cerevisiae/`. Information of genes and proteins were extracted and stored in a database using a PERL script. Table 3.1 overleaf provides a summary of chromosome lengths and gene numbers. The data contains a set of 275 untranslated genes which consists entirely of tRNAs.

3.3 Analysis

The yeast blocks available on the web page were downloaded and converted into MySQL tables to improve their handling and to make them compliant with the format of the tables used in the new approach. This dataset will from hereon be called 'B^{so}_{orig}' (**orig**inal **B**locks from **o**ld **s**earch results).

3.3.1 Reproduction of duplicated blocks

We first tried to reproduce the original blocks by simulating the original analysis with then new approach as closely as possible. The same parameters for gap and score threshold were used (30, 17.5 respectively), and query/hit pairs were obtained from the original SSEARCH results. Application of the new approach produced a set of blocks which was stored under the name 'B^{so}_{copy}'. An initial inspection showed a discrepancy particularly in the sub-telomeric regions, where repetitive regions can be found amongst several chromosomes.

Table 3.1: Summary of genes from *S. cerevisiæ* used in the analysis. Values such as gene numbers, sequence length, and GenBank accession number are listed for each chromosome. The gene numbers of the original dataset as well as the numbers of coding and non-coding genes are shown.

accession	chr	genes	CDS[a]	old[b]	length (bp)
NC_001133	1	112	108	99	230,203
NC_001134	2	445	432	389	813,139
NC_001135	3	183	173	158	316,613
NC_001136	4	851	823	737	1,531,929
NC_001137	5	308	288	273	576,869
NC_001138	6	142	132	128	270,148
NC_001139	7	606	570	525	1,090,937
NC_001140	8	294	283	276	562,639
NC_001141	9	229	219	218	439,885
NC_001142	10	412	388	349	745,444
NC_001143	11	354	338	313	666,445
NC_001144	12	569	548	498	1,078,173
NC_001145	13	511	490	456	924,430
NC_001146	14	437	423	387	784,328
NC_001147	15	593	573	522	1,091,284
NC_001148	16	516	499	462	948,061
total	16	6562	6287	5790	12,070,527

[a]CDS = coding sequence
[b]number of protein-coding genes used in (Seoighe and Wolfe, 1999)

Blocks in those locations have been excluded from the Seoighe and Wolfe (1999) dataset, and the same was consequently applied to 'B^{so}_{copy}'.

3.3.2 Parameter optimisation

We tried to modify the search parameters to achieve results as similar as possible to the old ones. The original SSEARCH data does not provide the necessary details for the calculation of relative sizes of alignments in respect to sequence lengths. The new approach makes use of these to filter out matches which only overlap in short motifs or domains, which can easily have arisen by chance. To enable this feature, a new set of query/hit pairs was created. This was based on the original 5790 proteins but with a new version of SSEARCH

(version 3.4t05 from Aug 18, 2001). The output parameters necessary for inclusion of alignment borders were switched on (-m 9). Default values were used for the substitution matrix and the gap parameters (matrix: BLOSUM50, gap opening: -12, gap extension: -2). Sequences were filtered with SEG as before.

The number of parameters and the different values they can take on renders it unfeasible to analyse every single possible combination. To ease the task, only a limited range for each parameter was tested. Using varying values for score (6 - 16), score range (0.1 - 0.4), alignment length (25 - 40%), and hit numbers (8 - 20), the new SSEARCH results of the original data were subjected to the block detection method. After removal of blocks in telomeric regions, for each set of results the distance to B^{so}_{orig} was calculated for block numbers, coverage, overlap, median length, median density. The averages of these values were taken for categories $sm \geq 4$ to $sm \geq 8$. The choice of those limits is based on the grounds that only few larger blocks exist and that some of the blocks with sizes smaller than four do not show a high degree of certainty. The number of genes from B^{so}_{orig} that were missing in the new blocks ($sm >= 4$) were also calculated. This produced a set of six quality values for each run. Graphical presentation of their variation was used to facilitate the determination of optimal settings.

3.3.3 Significance test of block sizes

It is conceivable that a number of genes and their homologues are located in close vicinity and form paired regions by chance alone. To test for this, block detection with optimised parameters was carried out a thousand times on datasets with randomised gene order. To make the simulated genomes, all genes were shuffled and then arbitrarily lined up in groups with sizes equivalent to the original chromosomes. The frequency of occurring block sizes was then compared to those based on the real map. This method was developed by Aoife McLysaght for the determination of significance values for block sizes in the

human genome (McLysaght *et al.*, 2001). Exclusion of telomeric regions does not make sense after randomisation of genes. Therefore, all detected regions were retained in the randomised sets as well as in a set based on real data, which was used for comparison.

3.3.4 Evaluation of search programs

Another objective consisted of testing the effect of various search programs on the results. In addition to the updated SSEARCH set, further query/hit pairs were created through all-against-all searches with the latest editions of BLASTP (version 2.1.3 Apr 1, 2001) and FASTA (version 3.4t05 Aug 18, 2001). For FASTA the word size parameter (ktup) was set to 1 which makes the search more sensitive but slower. In both cases the default matrices (BLOSUM50 for FASTA, BLOSUM62 for BLASTP) and default gap parameters (FASTA: -12 for opening, -2 for extension, BLAST: -12 for opening, -1 for extension) were used.

Seoighe and Wolfe (1999) used log-normalised S-W scores as a measurement of sequence similarity, because this had been proposed by Shpaer *et al.* (1996) to be independent of sequence length. However, since BLAST results do not provide Smith-Waterman (SW) scores it became necessary to switch to expectation values as the common similarity measure. This required the mapping of the previously used score threshold to an equivalent E-value. 958 query/hit pairs with normalised SW scores ranging from 9.5 to 10.5 were examined. The median of their expectation values yielded 1e-19, which was subsequently used as the similarity threshold for block detection.

3.4 Results

An obvious way to categorise blocks is to group them according to their size. A comparison between datasets would produce exaggerated results if blocks

are shifted between small groups due to changes in size. To counteract this, the following results are all based on accumulative block sizes, starting from varying lower limits for *sm*.

3.4.1 Reproduction of duplicated blocks

Comparison of two sets of blocks was carried out on several levels: overall coverage (in percent), overlap factor, median length (in basepairs), and median density (*sm*/covered genes). Instead of taking averages, both values for the two regions of a block went into the calculation to provide as much detail as possible. Table 3.2 lists the results of the comparison between 'B^{so}_{orig}' and 'B^{so}_{copy}'. Overlap amongst blocks was not detected. In Figure 3.1 on page 51

Table 3.2: Results of comparison between old (1) and new (2) blocks

sm	b1[a]	b2	co1[b]	co2	%[c]	den1[d]	den2	%	len1[e]	len2	%
≥3	58	75	0.55	0.70	27.3	28.6	19.6	31.5	44	55	25.4
≥4	48	54	0.51	0.64	25.5	28.6	20.9	26.9	48	66	36.3
≥5	37	39	0.46	0.58	26.1	27.1	19.7	27.3	58	83	40.7
≥6	29	32	0.40	0.54	35	25.6	20.6	19.5	67	88	31.1
≥7	24	24	0.36	0.46	27.8	27.1	20.4	24.7	73	99	35.3
≥8	21	20	0.34	0.42	23.5	24.5	19.7	19.6	84	123	46.6
≥9	15	17	0.28	0.39	39.3	24.3	19.6	19.3	116	136	17.4
≥10	15	16	0.28	0.38	35.7	24.3	19.6	19.3	116	141	21.6
≥11	11	14	0.23	0.35	52.2	23.2	19.6	15.5	129	143	10.7
≥12	9	9	0.19	0.25	31.6	23.2	19.1	17.7	129	158	22.5
≥13	8	7	0.17	0.21	23.5	23.2	17.1	26.3	129	192	49.1
≥14	4	4	0.09	0.13	44.4	21.7	15.3	29.5	137	214	55.4
≥15	2	3	0.05	0.10	100	22.8	15.1	33.8	160	223	39.1
≥16	1	2	0.02	0.06	200	33.1	22.3	32.6	119	183	53.9
≥17	1	0	0.02	0	100	33.1	0	100	119	0	100

[a] b = number of blocks
[b] co = coverage
[c] % = difference in percent
[d] den = density
[e] len = median length in Kbp

these numbers are illustrated in separate graphs. In all categories but the block

numbers the two datasets show significantly differing values. The number of blocks is very alike in both sets, except for sizes three and four. This accounts for the higher number of possibly random blocks that are detected, of which only few are included in the old data. Despite the high similarity in block numbers a notable difference can be seen in the other categories. For all sizes the coverage and median length of the new blocks is bigger than in the old blocks, whereas the density is always lower. Examination of the blocks shows that sometimes links are added which connect from within a region to a gene that lies outside the boundary of the old blocks, thereby violating conservation of gene order (see Figure 3.4 on page 55 for an illustration). This expands some of the new blocks and causes larger coverage and median length. The higher density of the older blocks can be explained through the manual addition of links with low scores. Median numbers for large blocks vary a lot between the two datasets because only very few are found. This has a bigger effect on distances if the size of a block changes and the block moves into a different category.

A detailed comparison of blocks with $sm \geq 4$ reveals that all except one of the original blocks were found through the new approach. The additional old block contains a link pair with low score, which was added manually. The according block in the new data consists only of the three proper pairs and therefore did not make it into the comparison. Four blocks with $sm = 4$ and one with $sm = 5$ were found that have fewer links in B^{so}_{orig} and do not exceed size three.

A comparison on gene level shows that, although more genes are contained in B^{so}_{copy} (820 vs. 745), not every gene from the original blocks is present. All of the 58 genes exclusive to B^{so}_{orig} possess similarity scores below the threshold, ranging from 8 to 17.3, and were therefore discarded in the new method. Lowering the score threshold to eight would enable automatic retrieval of all the gene links that are found in B^{so}_{orig}. On the other hand, this would also draw

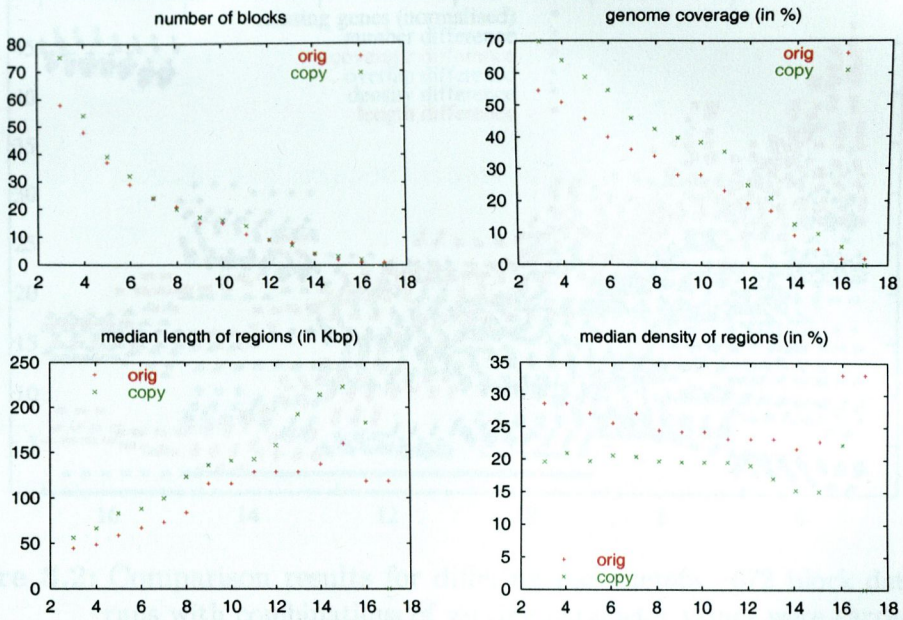


Figure 3.1: Comparison of old and new blocks. The numbers on the x-axis denote the lower border of accumulated block sizes. Values for blocks from B^{so}_{orig} are plotted in red, those for B^{so}_{copy} in green. Except for block numbers, differences in all other categories are very notable.

in a lot of spurious gene pairings, resulting in blocks of questionable quality.

3.4.2 Parameter optimisation

The results of comparisons between blocks from 672 parameter combinations and original yeast blocks are plotted in Figure 3.2 overleaf. They are sorted by score, hit number, range, and alignment length. Values for missing genes were normalised to fit into the range of the distance values (all in percent). As expected, the coverage of genes from B^{so}_{orig} grows with decreasing score threshold (black diamonds). However, this leads to larger and looser blocks, possibly including non-homologous gene pairs (coloured symbols). To minimise both number of missing genes and block differences, a threshold of 10 was chosen. This captures a large number of the old genes and also lies just before the point where block differences start to fan out. Following this decision,

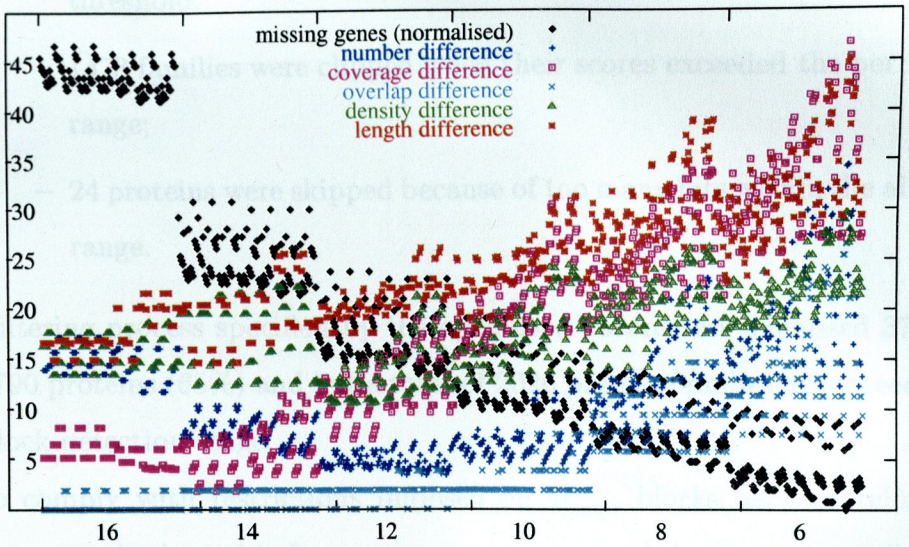


Figure 3.2: Comparison results for differing parameters. 672 block detection runs with combinations of varying parameter values were carried out. After comparison to B^{so}_{orig} differences in six categories were calculated for each set of blocks. These are plotted in different colours. Along the x-axis they are ordered first by score for which units are indicated. Values on the y-axis specify the difference to the original blocks in percent.

the according section of the graph could be examined separately, enlarged, and sorted by other parameters. The same criteria were applied again which resulted in the following settings: score threshold = 10, hit numbers = 14, range = 0.2, alignment length = 35%.

A new set of blocks was created and stored as B^{sn}_{opt} (**opt**imised **B**locks based on **n**ew **s**earch results). The chosen parameters were responsible for the following filter effect:

- 1993 proteins were discarded because none of their hits covered more than 35% of the longer sequence;
- 3288 of the remaining 3797 proteins required further sub-filtering because of breach of at least one criteria:
  - 1747 proteins were dropped because their top scores lay below the

threshold;

- 1478 families were clipped when their scores exceeded the permitted range;
- 24 proteins were skipped because of too many hits within the allowed range.

The filtering process specified by the optimised parameters dismissed 3764 of the 5790 proteins (65%) and left a total of 2026 protein families to proceed into the block detection stage.

To comply with restrictions imposed on B^{so}_{orig}, blocks between telomeric regions were eliminated before comparisons were carried out amongst all three sets. As can be seen from Figure 3.3 overleaf, the optimisation has successfully reduced the differences between old and new blocks in terms of coverage, median length, and median density. The dots coloured in blue, representing block characteristics for B^{sn}_{opt}, approximate the red ones (B^{so}_{orig}) much better than the green ones (B^{so}_{copy}). This becomes particularly obvious for median length and density in large blocks. These had been enlarged through links between proteins with weak similarities, which are now filtered out by the optimised parameters.

Improvements can also be observed at the gene level. Less then 5% of the 745 genes in blocks with $sm \geq 4$ are now exclusive to B^{so}_{orig}. 29 of them were not detected due to scores below the threshold. Three genes were discarded because their alignment length was too small. Another three fell out because their score lay outside the permitted range from the top hit. Finally, two genes came from a family with too many hits (HXT, hexose transporter family, 32 members, according to YPD at `http://www.proteome.com/databases/YPD/reports/HXT10.html`) and were therefore dismissed. Figure 3.4 on page 55 illustrates at an extreme example how differences in the resulting protein families can affect the consistency and borders of the blocks.

All of the 48 blocks from B^{so}_{orig} with $sm \geq 4$ can be found in the optimised

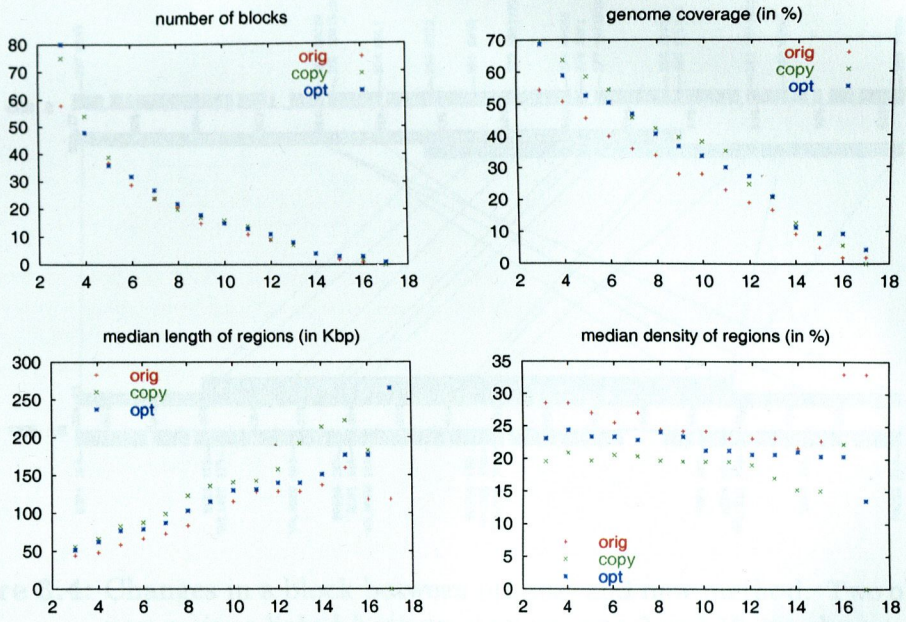


Figure 3.3: Comparison of original, copied, and optimised blocks. The numbers on the x-axis denote the lower border of accumulated block sizes.

set as well. They comprise of 785 genes, 78 of which are not found in the old pairs. Five blocks have increased from size two or three to four or more. For example, two connected regions with $sm = 4$ were found on chromosome seven, which lend support and expand two gene pairs that had been grouped into a possible block beforehand.

3.4.3 Significance test of block sizes

The frequency of block sizes in data with randomised gene order and their comparison to real data is shown in table 3.3 on page 56. Blocks with $sm \geq 5$ were pooled together because of their rare occurrence. The results show that numbers of blocks of sizes two and three are quite similar for randomised and real data. In contrast to this, larger blocks occur significantly more often in the real data and can therefore with high probability be regarded as genuine.

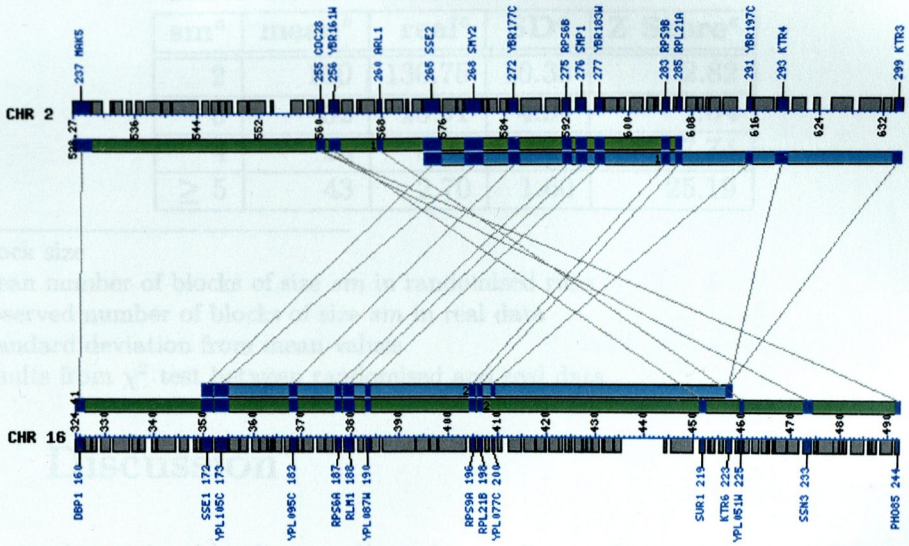


Figure 3.4: Changes in a block between original and new method. Two old and new regions linked between chromosomes 2 and 16 are shown. Block 7 from B^{so}_{orig} is coloured cyan, the block in green stems from B^{sn}_{opt}. Unlinked genes are grayed out. Linked genes are labelled with name and rank and are coloured in blue. Three of the previously detected pairs are missing: YBR197C/YPL077C due to low similarity score, the pairings between KTR3, KTR4, and KTR6 because of excess of the allowed score range. Five new pairs are found in the new block which breach the restriction of gene order that was imposed on the original blocks.

3.4.4 Evaluation of search programs

The graphs in Figure 3.5 on page 57 illustrate the block characteristics for sets derived from different underlying databases. Additionally, data from B^{so}_{orig} and B^{sn}_{opt} is shown. Both of these sets were created with the use of normalised SW scores, whereas the other three are based on expectation values instead. The dots from the original blocks (black) still stand out slightly but differences amongst all others are nearly indistinguishable over most parts of the graphs.

Table 3.3: Block sizes in randomised data

sm[a]	mean[b]	real[c]	SD[d]	Z Score[e]
2	160	130.75	10.38	2.82
3	32	23.91	4.94	1.64
4	25	5.97	2.46	7.74
≥ 5	43	2.70	1.60	25.19

[a]block size
[b]mean number of blocks of size *sm* in randomised runs
[c]observed number of blocks of size *sm* in real data
[d]standard deviation from mean values
[e]results from χ^2 test between randomised and real data

3.5 Discussion

We have shown in this chapter that the quality and arrangement of the manually edited blocks from (Seoighe and Wolfe, 1999) can be approximated but not entirely reached through our fully automated method. A trade-off exists between specificity and sensitivity, which is controlled by the parameters used in the analysis. When very strict values are set blocks are kept closer within the borders of the previously determined regions but more proteins are rejected. Loosening of the parameters leads to inclusion of more homologous genes with low similarity but also to widening of blocks through possibly unrelated proteins. The parameters with the strongest effect are obviously the score threshold and the alignment length, since they control the number of proteins coming through to the sub-filtering stage. A balanced combination between these parameters and the values for score range and hit numbers yields good results, as the blocks achieved through the optimisation process indicate.

In most cases the algorithm will be applied to data sets for which no previous map of blocks exists. Finding suitable parameters will then be more difficult. Two different strategies are possible:

1. If blocks for part of the genome are known and well studied, the parameter optimisation process as shown here can be carried out for that section only. Depending on the size and representative quality of the region this

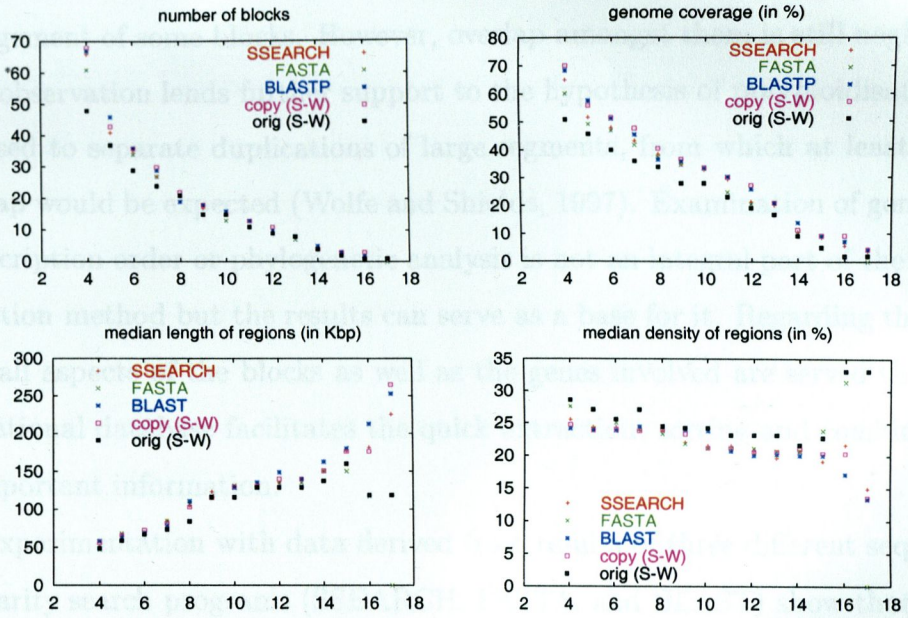


Figure 3.5: Effect of different search programs. The numbers on the x-axis denote the lower border of accumulated block sizes. B^{so}_{orig} (black) and B^{sn}_{opt} (pink) are based on normalised SW scores from SSEARCH databases. The other three sets were calculated with expectation scores from databases created with the according homology search program.

can result in a good approximation of parameters suitable for the whole genome.

2. Parameters can be tuned by starting with very strict values and stepwise relaxing them until the noise level in the results grows beyond a tolerance threshold. This approach is tedious and requires solid background knowledge to judge if detected links show homology or just similarity. Due to the facts that the program runs very fast, so that thousands of parameter combinations can be executed over night, and that graphical presentation of results facilitate their evaluation, this is still a feasible way to find suitable parameters.

One of the objectives in the earlier studies of duplicated blocks was to find out about the possibility of a polyploidy event in the evolution of *S. cerevisiæ*.

Breaking with the rule for conservation of gene order accounts mostly for the enlargement of some blocks. However, overlap amongst them is still negligible. This observation lends further support to the hypothesis of polyploidisation as opposed to separate duplications of large segments, from which at least some overlap would be expected (Wolfe and Shields, 1997). Examination of gene and transcription order or phylogenetic analysis is not an integral part of the block detection method but the results can serve as a base for it. Regarding the fact that all aspects of the blocks as well as the genes involved are served through a relational database facilitates the quick extraction, sorting and combination of important information.

Experimentation with data derived from results of three different sequence similarity search programs (SSEARCH, FASTA, and BLAST) show that:

1. normalised Smith-Waterman scores can be fairly well approximated through expectation values;

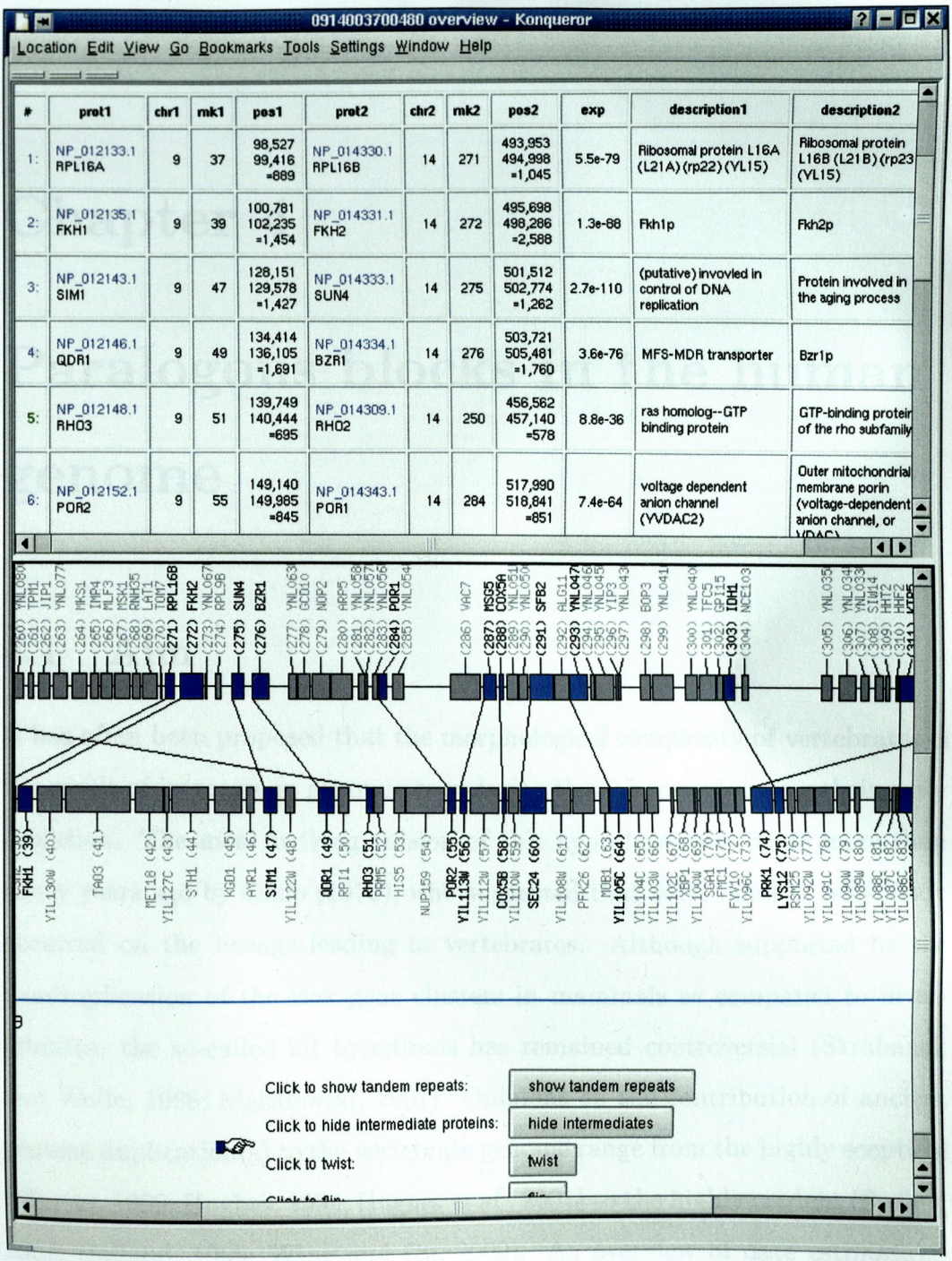
2. expectation values of these programs are very comparable;

3. results of block duplication based on query/hit pairs obtained from the three different programs differ only slightly.

This is generally consistent with findings by Brenner *et al.* (1998), who have compared the same programs. They noted that the E-values of SSEARCH and FASTA are very reliable. For BLAST a different version was assessed (WU-BLAST2), which tends to overrate the significance of hits. If this is the case with the BLAST version used in our analysis we would expect to see larger and possibly less denser blocks. This, however, is not obvious from the results.

Although this chapter dealt mainly with the evaluation of the method used for the detection of duplicated blocks, it is worth mentioning that, as a side product, a new WWW interface for graphical presentation and interactive exploration of the results has been created. This leads to an improvement of the mainly text-based listings that existed previously. A linear map of

blocks positioned along the chromosomes with the names of the duplicated genes and annotated dot-plots were the only graphical part of the web site at `http://acer.gen.tcd.ie/~khwolfe/yeast/`. The new interface combines listings of linked genes in a block with the detailed illustration of their arrangement within duplicated regions (see Figure 3.6 overleaf). Display of intermittent genes and navigation into surrounding areas allow further exploration of interesting regions. Finally, the ability to zoom from gene level to whole chromosomes and the presentation of connections between up to four chromosomes allows for a much broader view and the placement of blocks into a wider context.



#	prot1	chr1	mk1	pos1	prot2	chr2	mk2	pos2	exp	description1	description2
1:	NP_012133.1 RPL16A	9	37	98,527 99,416 =889	NP_014330.1 RPL16B	14	271	493,953 494,998 =1,045	5.5e-79	Ribosomal protein L16A (L21A) (rp22) (YL15)	Ribosomal protein L16B (L21B) (rp23 (YL15)
2:	NP_012135.1 FKH1	9	39	100,781 102,235 =1,454	NP_014331.1 FKH2	14	272	495,698 498,286 =2,588	1.3e-88	Fkh1p	Fkh2p
3:	NP_012143.1 SIM1	9	47	128,151 129,578 =1,427	NP_014333.1 SUN4	14	275	501,512 502,774 =1,262	2.7e-110	(putative) invovled in control of DNA replication	Protein involved in the aging process
4:	NP_012146.1 QDR1	9	49	134,414 136,105 =1,691	NP_014334.1 BZR1	14	276	503,721 505,481 =1,760	3.6e-76	MFS-MDR transporter	Bzr1p
5:	NP_012148.1 RHO3	9	51	139,749 140,444 =695	NP_014309.1 RHO2	14	250	456,562 457,140 =578	8.8e-36	ras homolog--GTP binding protein	GTP-binding protein of the rho subfamily
6:	NP_012152.1 POR2	9	55	149,140 149,985 =845	NP_014343.1 POR1	14	284	517,990 518,841 =851	7.4e-64	voltage dependent anion channel (YVDAC2)	Outer mitochondrial membrane porin (voltage-dependent anion channel, or

Figure 3.6: Screen shot of a block from the new yeast web page. Paralogous regions on chromosomes 9 and 14 are shown. Genes are drawn as filled blocks according to their chromosomal position. Paired genes that define the blocks are coloured blue, grey boxes are intermittent genes. Gene names and positions link to pages with additional information. The upper half provides details about gene positions, descriptions, and blast hits. Further integrated links provide retrieval of sequence alignments or acces to external databases.

Chapter 4

Paralogous blocks in the human genome

4.1 Aim

It has often been proposed that the morphological complexity of vertebrates is the result of increases in genomic complexity that happened during their early evolution. The most striking version of this hypothesis was made more than thirty years ago by Ohno (1970), who suggested that two rounds of polyploidy occurred on the lineage leading to vertebrates. Although supported by the quadruplication of the Hox gene clusters in mammals as compared to invertebrates, the so-called 2R hypothesis has remained controversial (Skrabanek and Wolfe, 1998; Makalowski, 2001). Opinions on the contribution of ancient genome duplication(s) to the vertebrate genome range from the highly sceptical (Martin, 1999; Hughes, 1999; Hughes *et al.*, 2001) to the highly credent (Spring, 1997; Holland, 1999; Wang and Gu, 2000). An overview of date estimations for duplication events can be seen in Figure 4.1 overleaf, taken from Skrabanek and Wolfe (1998). The draft version of the human genome is the most complete vertebrate sequence to date. Here we have analysed it to identify paralogous chromosomal regions and estimate the ages of duplicate genes.

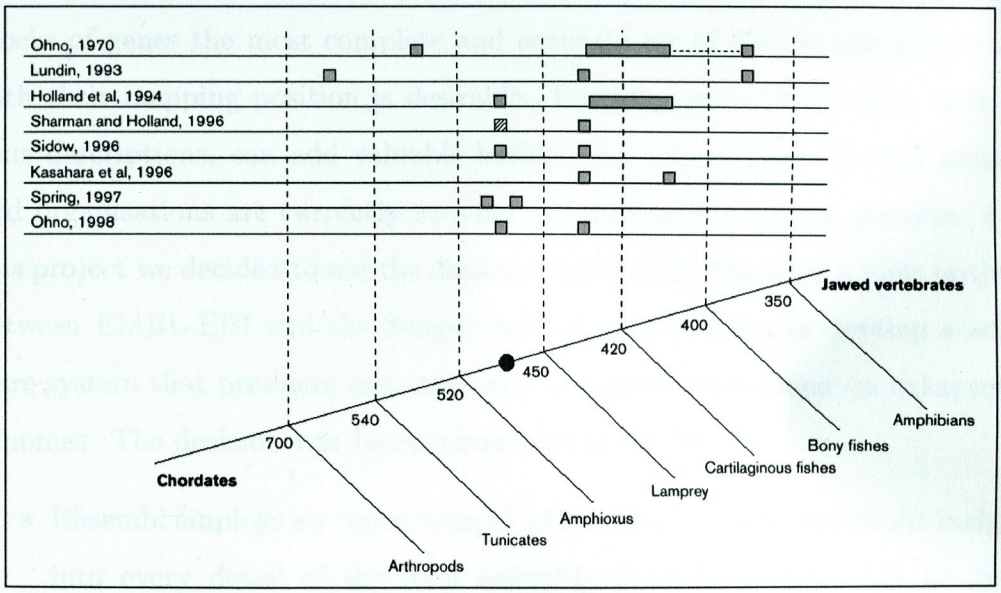


Figure 4.1: Date estimation for polyploidy events in vertebrates. Overview of the various proposals of the occurrences of polyploidy events in the vertebrate lineage. Divergence of various lineages from the vertebrate lineage are drawn schematically, not to scale. Shaded boxes indicate each proposed genome duplication and are drawn at the centre of possible time ranges. The hatched box indicates a proposed wave of tandem duplications as opposed to tetraploidisation event (Sharman and Holland, 1996). Ohno (1970) postulated a tetraploidisation event at the divergence of fish and/or amphibians, shown by the two boxes connected by a dotted line. The circle at $\sim$ 500 Myr denotes the origin of vertebrates.

Parts of this work (as acknowledged below) were done in collaboration with my fellow graduate student Aoife McLysaght.

4.2 Data

A final version of the human genome sequence will probably not be available until 2003 (Lander *et al.*, 2001). For analysis aimed at detecting duplicated blocks of genes the most complete and accurate set of human genes together with their mapping position is desirable. Further annotational data, such as gene descriptions, can add valuable background information. Several groups and organisations are currently striving to annotate the human genome. For this project we decided to use the data set provided by Ensembl, a joint project between EMBL-EBI and the Sanger Institute, which aims to develop a software system that produces and maintains automatic annotation on eukaryotic genomes. The decision was based upon several reasons:

- Ensembl employs an "open source" philosophy which allows public insight into every detail of the data assembly procedure. This can be very valuable to trace back steps that lead to decisions in the gene prediction process.
- Strategies and plans are discussed openly through a mailing list. This offers the chance to receive early information on data-related issues and also allows for interaction with the developers.
- The data are provided in SQL database format which facilitates their integration into our local computer system.
- Most importantly, all data and programs are freely available and can be used without any restrictions.

In the annotation process Ensembl integrates information of known genes and proteins from databases such as SwissProt, GenBank, HUGO and RefSeq

(Pruitt *et al.*, 2000). The GeneWise program (Birney and Durbin, 2000) converts these to exon structures on the human sequence. Additionally, new genes are detected through the GenScan (Burge and Karlin, 1997) program. This results in a set of predicted and confirmed genes. In this project only the latter was used which, beside known genes, also includes new members of existing families, predicted using GeneWise.

Sequencing of the human genome is still in progress and so is the annotation process which results in regular data updates. Our analysis is based on the 1.0 release (April 2001) of Ensembl. In August 2001 a new release has been made available (version 1.1) but after discovering some flaws in the data (Ewan Birney, personal communication) it was decided to continue with the older version.

The 1.0 dataset contains amino acid sequences for 27,615 proteins representing 24,046 confirmed genes. In case of alternative splices only the longest isoform was used. Additional annotation in form of product description is available for 18,922 genes. Ensembl provided mapping positions for 23,664 of the genes based on the "Golden Path" from December 2000. This is an arrangement of BAC-cloned human sequences assembled and maintained by the University of California at Santa Cruz (Kent and Haussler, 2001) which yields maximum genome coverage with minimal overlap between the clones. The underlying data comprise approximately 830 Mb of finished sequences and 2,300 Mb of draft sequences. A remaining 100 Mb had not been sequenced at the time of the data freeze. Data in the draft stage has only been sequenced once or twice and contains basepair ambiguities, gaps, and segments with unknown order or orientation. To achieve high quality sequences tenfold coverage is striven for.

The proteomes of two fully sequenced invertebrates were used in some parts of the analysis to enable evolutionary comparison. 14,335 proteins of *Drosophila melanogaster* (Adams *et al.*, 2000) and 19,835 proteins of *Cænorhabditis elegans*

(*C. elegans* Sequencing Consortium, 1998) were retrieved from GenBank release 123 (April 2001) and WormPep49, respectively. Removal of alternative splice variants (retaining the longest isoform) left 13,473 fly proteins and 18,685 worm proteins. All sequences were downloaded as FASTA files and converted into MySQL tables.

4.3 Preparations

4.3.1 Sequence similarity search

The basic information needed to find paralogous blocks lies in homology relationships amongst proteins. The first step in the analysis therefore requires the comparison of all human proteins with each other. Worm and fly proteins were added to the search database to provide additional evolutionary information. For speed reasons sequence similarity searches were carried out with gapped BLASTP (Altschul *et al.*, 1997) using the following parameters: expectation cutoff (E) = 1, matrix = BLOSUM45, one-line descriptions = 0, alignments = 500. SEG filtering was switched on to exclude low complexity regions. Gap opening and extension values were left at their default (-12 and -1, respectively). All of the information included in the one-line descriptions can also be found in the alignments reported by BLAST. Additionally, the latter provide information about the relative lengths of the sequence segments that form the overlap. The alignment value was set high enough to include all significant hits, even for large families. BLOSUM45 specifies a matrix which was derived from a set of proteins sharing about 45% sequence identity (Henikoff and Henikoff, 1992). Its use was preferred towards shallower matrices, such as BLOSUM62, because it is better suited for detecting remote homologues.

Employing all twenty nodes of the Beowulf cluster, the comparison of 27,615 sequences against a database of 56,204 sequences (24,046 human + 13,473 fly + 18,685 worm) took about 90 minutes and resulted in 4 Gigabytes of data. A

PERL script was applied to extract hit information and store them in a MySQL table, which subsequently held over 1.7 million query/hit entries. No hits were produced by 510 queries, and 3002 sequences only matched themselves.

4.3.2 ORF collapsing

To prevent interruptions of blocks due to inflated gaps from recent duplications, an attempt was made to detect and collapse tandem duplicates. This was achieved through a PERL script, which replaces all query/hit pairs with an E-value $\leq$ 1e-15 in a 30 position neighbourhood with the longest one. After scanning the whole genome the set of proteins was reduced by 12% to 20,842. The removed proteins and the collapsed ORFs were stored in a MySQL table for later reference.

A further twelve cases were identified where individual exons appeared to have been incorrectly annotated as complete genes. They were detected by looking for annotated genes less than 30 positions apart, dissimilar in sequence (E $\geq$ 1e-5), that both hit the same remote protein with expectation (E-value) $\leq$ 1e-15, and align with less than 20 amino acids overlap. Only the longest peptide of each group was retained, which resulted in a final set of proteins, representing 20,830 mapped genes. This data set is referred to as the collapsed map.

4.3.3 Parameter optimisation

The parameters used in the block detection play an important role in the outcome of the programs. A too strict set of values will only show highly conserved or recent blocks, whereas setting the parameters too loose can lead to inflated block sizes and numbers as a result of inclusion of insignificant pairings. Unlike with yeast, there is no reference map of duplicated blocks available for human that could serve as a template for the determination of suitable parameters. We decided to run the program with different combinations of

Table 4.1: Tested ranges and choices for parameters.

parameter[a]	value range	final choice
E: expectation threshold	1e-3 – 1e-15	1e-7
A: alignment length	20 – 50	30
G: gap length	10 – 40	30
R: expectation range	1e-10 – 1e-40	1e-20
H: allowed hits	10 – 40	20

[a]these quantities are defined on page 37

parameters, each varying within a sensible range. Table 4.1 lists the parameters and the values they were tested for, as well as the value that was eventually chosen.

The effect of different parameter constellations was evaluated with respect to block numbers, block density, genome coverage, block overlap, and block size. Through manually inspection a choice was made for values from within a range that produces stable results, i.e. small changes in parameter values do not affect the results significantly. In case of ambiguous situations it was attempted to keep the parameters rather conservative. This shows, for example, in the choice of 1e-07 as the expectation threshold, considering that values of 1e-02 are sometimes regarded as indications for homology (Pearson, 2000).

4.3.4 Filtering of protein families

The results from similarity searches serve as a measure for the construction of protein families which are subsequently assembled into blocks. There is no exact expectation threshold that separates remote homologues from similar analogues. Furthermore, some protein families, such as zinc fingers, have grown to large sizes that are unlikely to be useful for detecting ancient duplication events and are most likely to create 'noise' in the results. The combined effects of the parameters used in the block detection program act as filters that sort out unsuitable query/hit pairs. Sequences from the two invertebrates were included in the search to act as a natural orthology threshold: for each query

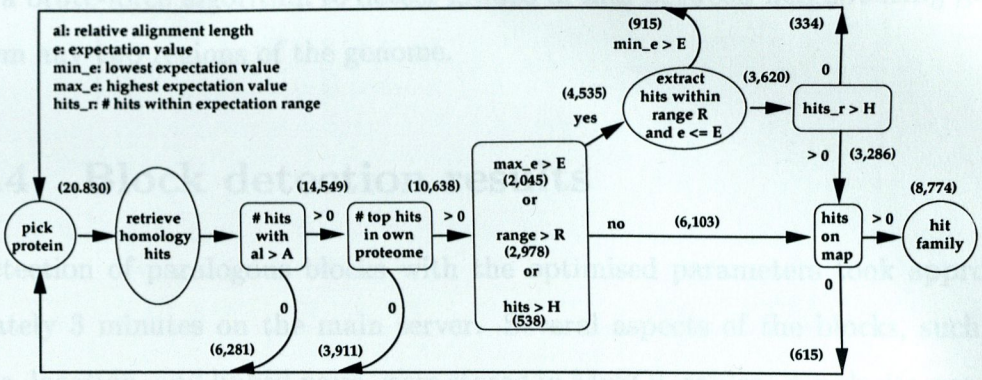


Figure 4.2: Filter for protein families in human. Flowchart of filters applied to similarity hits of a protein to derive its family. The different filter stages are explained in section 2.3.2 on page 36. Descriptions and values of parameters A,E,R,H are listed in Table 4.1 on the preceding page. Numbers in brackets indicate the number of proteins that are affected.

protein, any human BLASTP hits having less similarity than the best invertebrate orthologue were skipped, thus distinguishing gene family expansions that occurred in the chordate lineage from older paralogy relationships. Figure 4.2 illustrates the effect of the different filter stages. Out of the 20,830 proteins on the map, 6,281 did not produce hits with other proteins that aligned over at least 30% of the longer sequence length. Of the remaining 14,549 proteins 3,911 were excluded because their top hit was an invertebrate sequence. 915 proteins were discarded, because their best hits did not reach the expectation threshold. 334 proteins were dropped because they had at least 20 hits within a factor 1e20 of the top hit. 615 proteins for which none of the hits could be mapped where discarded, whereas 329 cases that matched only tandem repeats were referenced to the collapsed ORF. This left a set of 8,774 query proteins, each representing a separate gene, whose BLASTP results were used in the block detection process. In some cases human proteins that had been eliminated because their top hit was an invertebrate sequence were restored to the dataset because they were hit (more strongly than an invertebrate sequence) by another human protein. This resulted in a total number of 9,519 human proteins that were subjected

to a brute-force algorithm to detect groups of hits between neighbouring genes from any two regions of the genome.

4.4 Block detection results

Detection of paralogous blocks with the optimised parameters took approximately 3 minutes on the main server. Several aspects of the blocks, such as size, location, and linked pairs, were stored in MySQL tables. A web site was set up at `http://www.gen.tcd.ie/dup/` which allows access to the tables and also acts as a graphical interface to the blocks. Some helper applications have been developed which scan the blocks for interesting features and extract summary information. Table 4.2 overleaf provides an overview of results statistics. The entries contain data in accumulative form, i.e. values for blocks with sizes equal to or greater than a threshold are combined. This facilitates the evaluation of characteristics for blocks from a certain size upwards. For example, overall genomic coverage of more than 90 percent and nearly fourfold overlap by all blocks of size ≥ 2 is conspicuously high and sheds doubt on the validity of the inclusion of small blocks. Given the rate of gene loss after polyploidisation observed in other species (Wolfe and Shields, 1997; The *Arabidopsis* Genome Initiative, 2000; Vision *et al.*, 2000), an overlap of less then two is more likely, even after two rounds of genome duplication. A map of the location of all blocks is shown in Figures 4.9 - 4.12 at the end of this chapter.

4.4.1 Significance of block sizes

Further investigation into the significance of block sizes and expected background noise was carried out with the help of a method developed by Aoife McLysaght (McLysaght, 2001). With the objective of establishing the expected distribution of blocks that are formed by chance alone, one thousand simulations were carried out. Shuffling of genes was implemented in a way that retains

Table 4.2: Statistics of blocks in human found using the optimised parameters described in table 4.1

sm[a]	b[b]	g[c]	co[d]	ov[e]	den[f]	len[g]
≥ 2	1642	6120	91.4	3.6	14.3	1.8
≥ 3	504	3852	79.0	2.1	12.5	3.7
≥ 4	244	2730	64.2	1.7	12.5	5.0
≥ 5	151	2139	53.6	1.5	12.8	5.7
≥ 6	96	1662	44.1	1.3	12.85	6.8
≥ 7	65	1315	37.8	1.3	12.95	8.4
≥ 8	43	1030	30.1	1.2	13.1	9.4
≥ 9	33	894	27.5	1.2	12.75	12.2
≥ 10	25	775	25.1	1.1	12.9	15.5
≥ 11	18	640	22.4	1.1	12.7	19.6
≥ 12	16	596	20.5	1.1	13.05	20.7
≥ 13	14	547	18.3	1.1	13.05	20.7
≥ 14	12	498	17.0	1.0	13.05	23.3
≥ 15	9	423	14.7	1.0	13.1	25.8
≥ 16	8	393	13.3	1.0	13.1	25.8
≥ 18	7	357	12.5	1.0	13.1	27.0
≥ 19	6	320	10.5	1.0	13.1	26.7
≥ 22	5	278	8.4	1.0	13.5	26.7
≥ 24	3	182	4.7	1.0	14.15	22.9
≥ 27	2	126	3.1	1.0	13.5	22.9
≥ 29	1	63	1.9	1.0	13.5	30.6

[a]sm = size of blocks
[b]b = number of blocks
[c]g = linked genes (gene pairs that forming connections between paralogous regions)
[d]co = coverage (as percentage of 3,213,523,081 bp)
[e]ov = overlap (sum of lengths of all blocks / coverage)
[f]den = density (number of spanned genes / number of linked genes)
[g]len = length (measured in megabases)

Table 4.3: Block sizes in randomised data

sm[a]	real[b]	mean[c]	SD[d]	Z Score[e]
2	1138	1051.67	29.43	2.93
3	260	159.05	12.35	8.17
4	93	30.10	5.62	11.20
5	55	6.89	2.71	17.76
≥6	96	2.56	1.63	57.48

[a]block size
[b]observed number of blocks of size *sm* in real data
[c]mean number of blocks of size *sm* in randomised runs
[d]standard deviation in randomised runs
[e]number of S.D. by which the 'real' value exceeds the 'mean' value

characteristics of the real genome, such as gene family sizes and chromosome sizes, but completely randomises gene order. Any apparently paralogous regions detected in shuffled data must be artefacts. The data resulting from the simulations were collected and compared to the block distribution observed on real data (see Table 4.3). Although the number of blocks containing just two duplicated genes was similar in the simulations and the real data, the number of blocks with $sm \geq 3$ was consistently higher in the real data. This deviation is more marked for the larger blocks. The number of blocks containing at least 6 duplicated genes in the human genome is over 50 standard deviations greater than the mean observed in the shuffled genomes. This analysis indicates that any region with $sm \geq 6$ has almost certainly been formed by a single regional duplication and that $sm = 3$ is the borderline (with our parameter set) for statistical significance of a candidate paralogy region.

4.4.2 Paralogous regions

The minimal regions that define a block have $sm = 2$, and there are 1642 regions with $sm \geq 2$. The most extensive region found, which pairs a 41 Mb region of chromosome 1 (including the tenascin-R locus) with a 20 Mb region of chromosome 9 (including tenascin-C), has a block size of $sm = 29$. The regions with the next-highest numbers of duplicates are on chromosomes 7/17

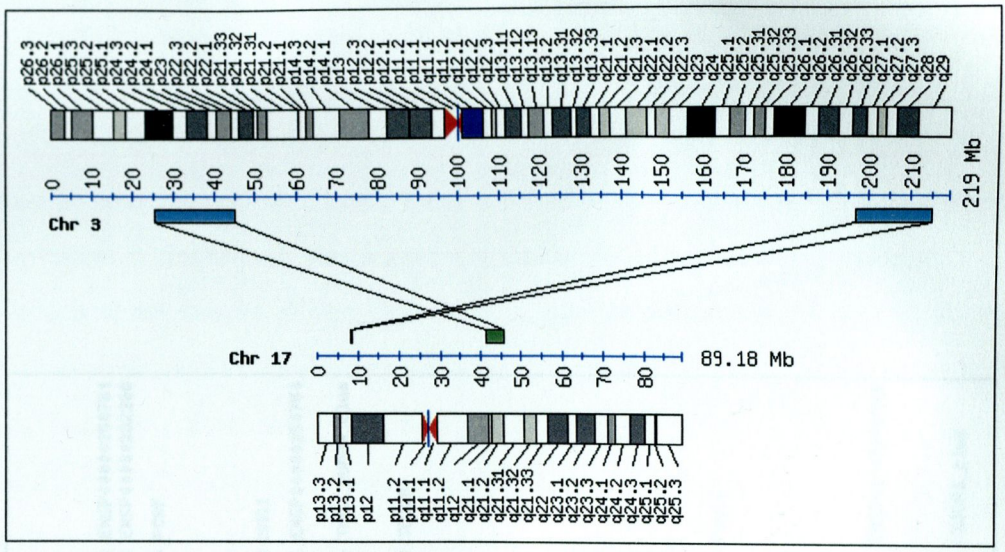


Figure 4.3: Paralogous regions between chromosomes 3 and 17. Chromosomes are drawn in karyotype format with a scale in megabasepairs beside it. Two pairs of connected regions are shown.

(sm = 28 around the HoxA/HoxB clusters), chromosomes 2/12 (sm = 26 around ITGA6/ITGA7), chromosomes 15/18 (sm = 23 around LIPC/LIPG), chromosomes 1/6 (sm = 23 around EYA3/EYA4), and chromosomes 5/15 (sm = 21 around IQGAP2/IQGAP1). The ten largest regions can be browsed at `http://www.gen.tcd.ie/dup`.

Figure 4.3 shows two sets of paralogous regions of sizes $sm = 8$ and $sm = 9$ between chromosomes 17 and 3. A consistent expansion is noticeable on the larger chromosome, which coheres with a general lower gene density (6.1 genes per Mb on chromosome 3 compared to 13.9 genes per Mb on chromosome 17).

The organisation of one of the blocks, containing ten gene pairs, is shown in detail in Figure 4.4 overleaf. This region includes duplicated genes for the histone acetyltransferases PCAF and GCN5L2 (Xu *et al.*, 2000) and for the alpha and beta forms of topoisomerase II (TOP2A and TOP2B; Lang *et al.* (1998)). Just over 10% of genes on both sides of the block are linked, which rises questions about the origin of the intermediate genes. An inspection of their top BLASTP results reveals that about a third of them either have no hits at all or hits with an invertebrate protein. They possibly account

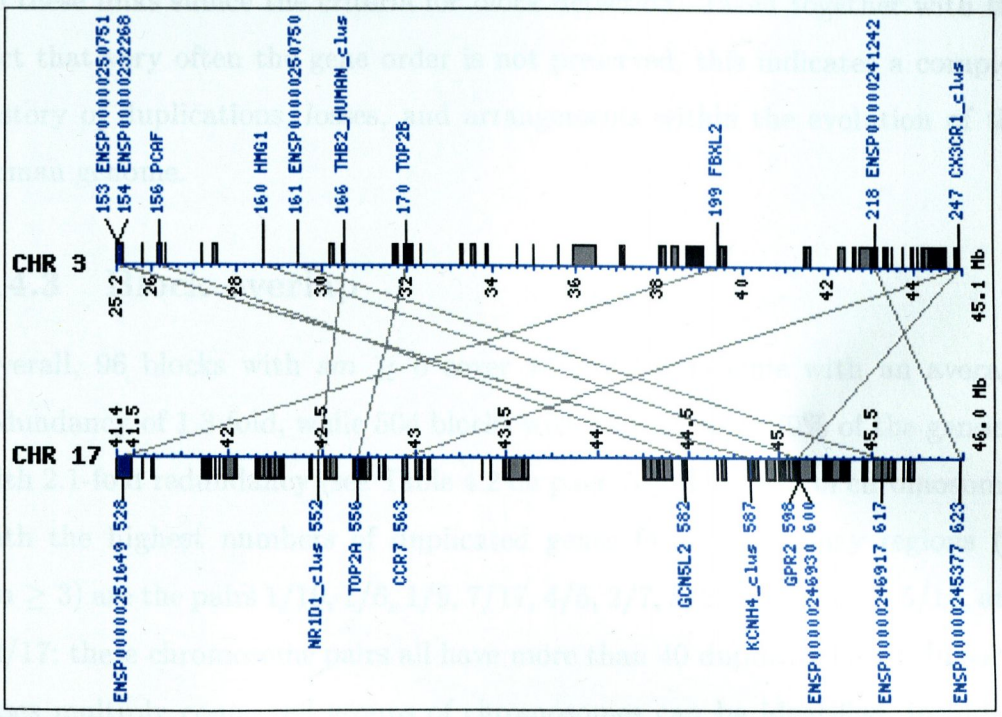


Figure 4.4: Paralogous block between chromosomes 3 and 17. Chromosome 17 is drawn at 5-fold larger scale than chromosome 3 to facilitate view of details. Linked genes are drawn as connected, blue-filled squares at positions according to their physical location on the chromosome. Intermediate genes are filled in grey. The labels show the rank of the linked genes and their name if known. Otherwise the Ensembl name is given. Representatives of tandem repeats have the appendix '_clus' (for cluster) attached to their names.

for ancient duplications and lost copies. Roughly a sixth of the hits stem from the same chromosome, and the remaining half comprise hits to other chromosomes. These proteins could have arrived in the paralogous region through rearrangements, both local and intra-chromosomal. A considerable proportion of intermediates shows similarity to genes on chromosomes 2 and 7, which share blocks in the Hox regions with chromosome 17, but in no case do these links suffice the criteria for block detection. Taken together with the fact that very often the gene order is not preserved, this indicates a complex history of duplications, losses, and arrangements within the evolution of the human genome.

4.4.3 Block overlap

Overall, 96 blocks with $sm \geq 6$ cover 44% of the genome with an average redundancy of 1.3-fold, while 504 blocks with $sm \geq 3$ cover 79% of the genome with 2.1-fold redundancy (see Table 4.2 on page 70). The pairs of chromosomes with the highest numbers of duplicated genes forming paralogy regions (of $sm \geq 3$) are the pairs 1/19, 1/6, 1/9, 7/17, 4/5, 2/7, 8/20, 2/12, 1/12, 5/15, and 12/17; these chromosome pairs all have more than 40 duplicate genes. In some cases multiply connected groups of chromosomes can be identified, including previously known sets, such as chromosomes 2/7/12/17 (Ruddle *et al.*, 1994), 1/6/9/19 (Kasahara *et al.*, 1997), and 4/5/10 (Pebusque *et al.*, 1998). The first of those groups covers the well-studied Hox clusters. It is reassuring that our method identifies them with good accuracy. In addition to previously known duplicated areas, some extra links were detected which increase the extent of paralogous regions. An overview of all blocks amongst chromosomes 2, 7, 12 and 17, including the Hox clusters, is shown in Figure 4.5 overleaf. In accordance with the 2R hypothesis no region in the genome is covered by more than three blocks with $sm \geq 6$ from other chromosomes. The only exception exists in a small segment near the Hox region of chromosome 17 that, aside from the

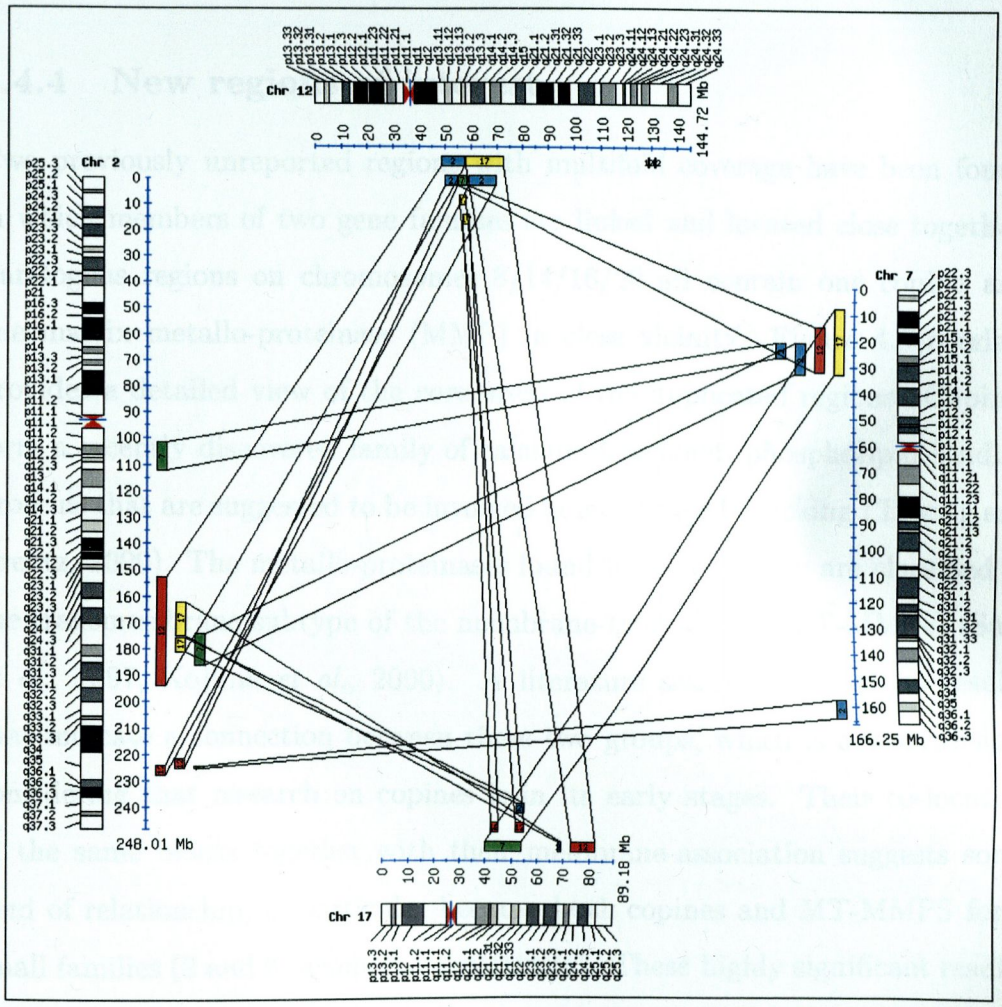


Figure 4.5: Blocks between Hox chromosomes. All detected paralogous blocks with size of $sm \geq 6$ between human chromosomes 2, 7, 12, and 17 are shown.

other Hox chromosomes, is also linked to an equally small and dense paralogous region on chromosome 1. Regarding the high number of paired genes (6 out of 8 on chromosome 1 and 6 out of 9 on chromosome 17) it appears to be a recent duplication that has not yet been loosened up by subsequent gene loss and rearrangements.

4.4.4 New regions of interest

Two previously unreported regions with multifold coverage have been found in which members of two gene families are linked and located close together. Paralogous regions on chromosomes 8/14/16/20 all contain one copine and one matrix metallo-proteinase (MMP) in close vicinity. Figure 4.6 overleaf provides a detailed view of the core areas of the duplicated regions. Copines form a recently discovered family of calcium-dependent, phospholipid-binding proteins that are suggested to be involved in membrane trafficking (Tomsig and Creutz, 2000). The metallo-proteinases found in their vicinity are classified as the transmembrane subtype of the membrane-type MMPs (MT-MMPs) (Sato *et al.*, 1997; Kojima *et al.*, 2000). A literature search produced no results that indicate a connection between these two groups, which is not surprising, considering that research on copines is in its early stages. Their co-location in the same blocks together with their membrane-association suggests some kind of relationship, in particular because both copines and MT-MMPS form small families (8 and 6 members, respectively). These highly significant results provide an interesting base for a separate research project.

Another set of blocks between chromosomes 1/11/19 exposes a similar phenomena. Each of the paralogous regions contains a galectin and one or two genes of the MAPK-family (mitogen-activated protein kinases) that are linked with each other. Members of both families are involved in a variety of cellular processes. A possible connection between them could be derived from a report of galectin-3 expression induced by mitogen (Joo *et al.*, 2001). As with

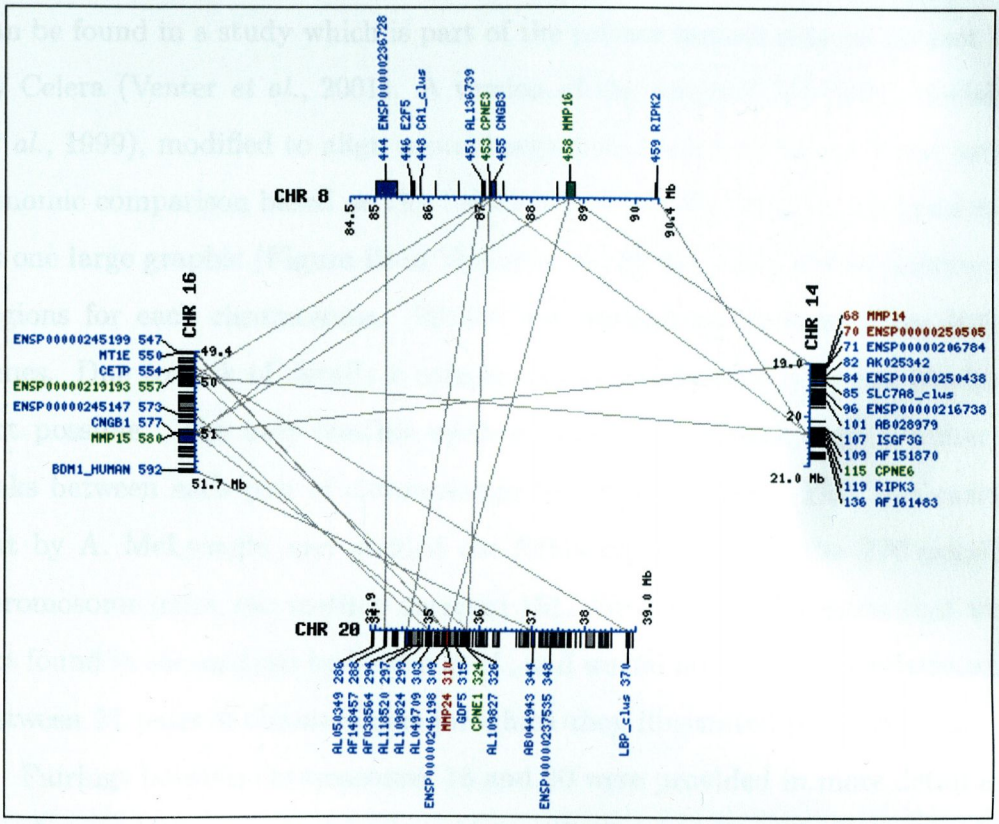


Figure 4.6: Blocks with copines and MMPs. The core areas of duplicated regions between human chromosomes 8,14,16, and 20 are shown. A copine (CPNE) and a matrix metallo-proteinase (MMP) can be found in each of them in close vicinity (protein ENSP00000219193 on chromosome 16 is predicted to be a copine). Linked genes are drawn and labelled in blue. Genes with links to two or three other blocks are highlighted in red and green, respectively. Intermediate genes are filled with grey.

the previous example, the findings in the blocks cannot give comprehensive answers but suggestions for the directions of future research.

4.4.5 Comparison with Celera data

The only equally comprehensive report on paralogous blocks in human so far can be found in a study which is part of the private human genome project led by Celera (Venter *et al.*, 2001). A version of the program MUMmer (Delcher *et al.*, 1999), modified to align protein sequences, was used to carry out intra-genomic comparison based on the Celera sequence data. Results are presented as one large graphic (Figure 13 in Venter *et al.*, 2001), which shows paralogous regions for each chromosome. Blocks were defined by at least three linked genes. Due to lack of details a comprehensive comparison with our blocks is not possible. The only feasible method consists of counting the number of links between each pair of chromosomes for both data sets. This was carried out by A. McLysaght and yielded the following results: Of the 276 possible chromosome pairs, our method detected 151. We detected 55 regions that were not found in the analysis by Venter *et al.*, and we did not detect any relationship between 21 pairs of chromosomes for which they illustrated pairings.

Pairings between chromosomes 18 and 20 were provided in more detail and are shown in Figure 4.7 overleaf together with the according blocks detected by our method. The overall appearance of cross-links seems to be the same except for a region near the centre of chromosome 20. The segment from gene "GATA rel" to "Krup rel" on chromosome 20 in the MUMmer graph might correspond to the far end of chromosome 20 (>64 Mb) in our graph, because both seem to be connected to roughly the same region on chromosome 18. A translocation such as this could occur from differences in the assembly process.

Large discrepancies exist in the underlying data: Celera reports 217 protein assignments on chromosome 18 and 322 on chromosome 20. This corresponds to 388 and 748 proteins in the Ensembl data. Venter *et al.* state that their

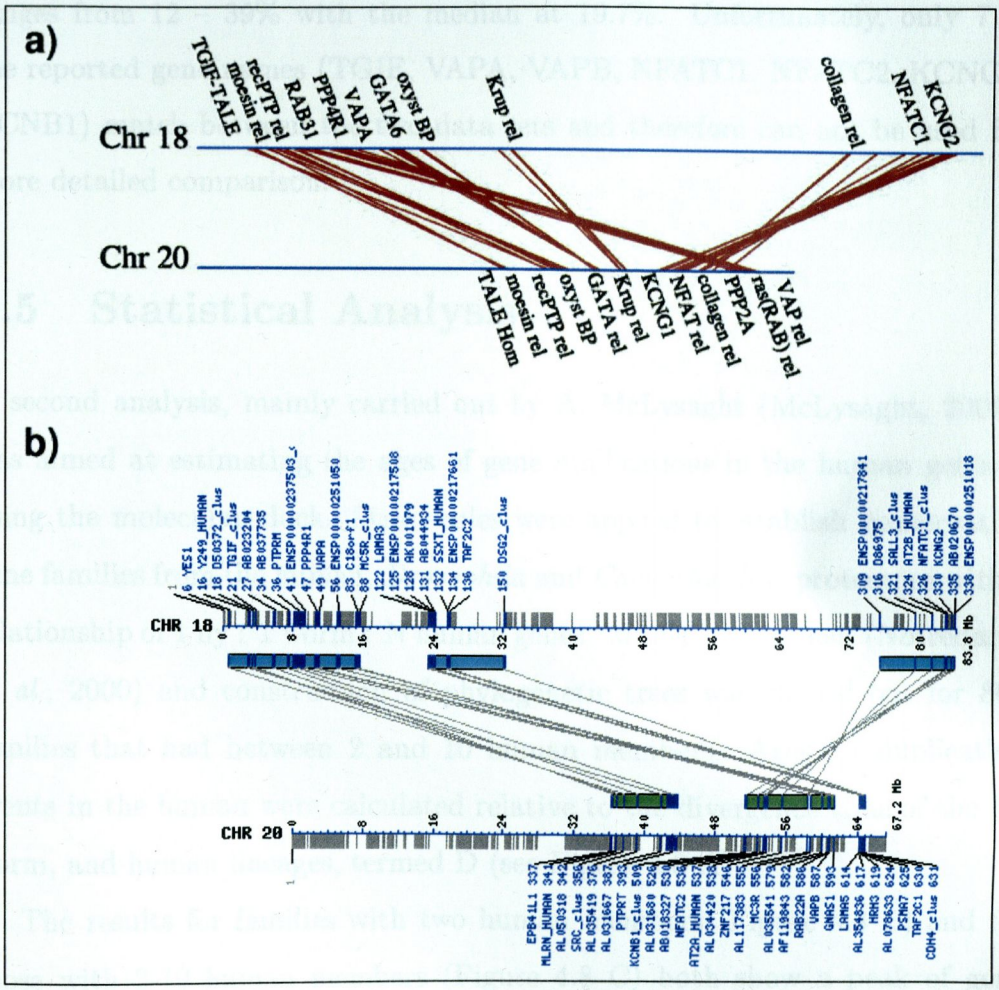


Figure 4.7: Comparison with Celera blocks between chromosomes 18 and 20. The upper figure is taken from Venter *et al.* (2001); the lower one shows a graphic resulting from our method. In both cases each cross-link represents a pair of homologous proteins. Figure (a) provides labels for a selection of genes. Information in Figure (b) includes a scale for each chromosome, gene details, labels for all linked pairs, and block boundaries.

analysis found 64 protein pairs in the blocks between chromosome 18 and 20 and that these blocks have a dduplicate gene density of 20 – 30%. In our case four blocks of sizes 6, 7, 7, and 8 are detected, which link a total of 29 and 28 genes on chromosome 18 and 20, respectively. The density of involved genes ranges from 12 – 39% with the median at 19.7%. Unfortunately, only 7 of the reported gene names (TGIF, VAPA, VAPB, NFATC1, NFATC2, KCNG2, KCNB1) match between the two data sets and therefore can not be used for more detailed comparison.

4.5 Statistical Analysis

A second analysis, mainly carried out by A. McLysaght (McLysaght, 2001), was aimed at estimating the ages of gene duplications in the human genome using the molecular clock. Strict rules were applied to establish conservative gene families from the human, *Drosophila* and *Caenorhabditis* proteomes with a relationship of 1 fly : 1 worm : N human genes. Multiple alignment (Notredame *et al.*, 2000) and construction of phylogenetic trees was carried out for 805 families that had between 2 and 10 human members. Ages for duplication events in the human were calculated relative to the divergence time of the fly, worm, and human lineages, termed D (see Figure 4.8 A).

The results for families with two human members (Figure 4.8 B) and for those with 2-10 human members (Figure 4.8 C) both show a peak of gene duplications with relative ages 0.4-0.7 D. The timing of the divergence of fly, worm, and vertebrate lineages is not certain, but taking a recent estimate of D = 833 Mya (Nei *et al.*, 2001), this corresponds to a burst of gene duplications in the period 333-583 Mya, which includes the origin of the vertebrates. The peak is more apparent in the data from two-membered families, where there is only one gene duplication event per tree, than in the larger families where there is more noise from the many duplication events that have shaped them. Additional phylogenetic analyses using other vertebrate sequences provide support

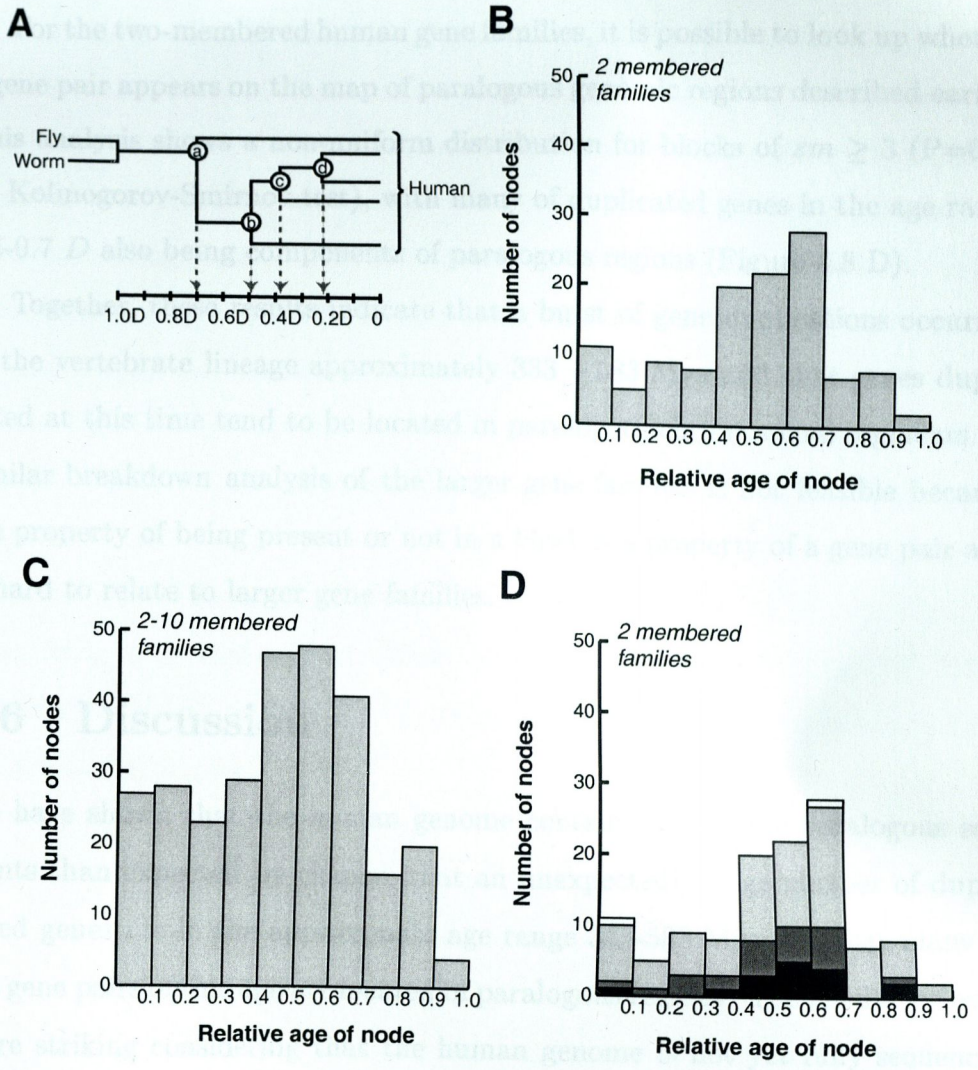


Figure 4.8: Estimation of gene duplication dates using linearised trees with fly and worm outgroups. **(A)** Model linearised tree of a five-membered gene family. The time of duplication for each of the nodes $a-d$ is indicated on the scale below the tree. Ages are expressed relative to the arthropod-chordate divergence (D); for example, the age of node a is $0.7D$. **(B,C)**Distribution of the estimated ages of vertebrate specific nodes of 2 membered and 2-10 membered families respectively. Each node represents a duplication event, and a family with N members has $N-1$ nodes. **(D)** Breakdown of estimated duplication dates among gene pairs mapped to paralogy blocks for two-membered gene families. The duplicated gene pairs in the histogram in (C) were placed into four categories: pairs making up paralogy regions with $\geq$6 duplicated genes (black); pairs making up paralogy regions with $\geq$3 duplicated genes (dark grey); pairs that appear on the map but are not present in blocks with at least 3 duplicated genes (light grey); and pairs for which one or both of the genes did not appear on the condensed gene map used for our analysis (white). All figures and legends were taken from McLysaght (2001).

for the approximate time scale used.

For the two-membered human gene families, it is possible to look up whether a gene pair appears on the map of paralogous genomic regions described earlier. This analysis shows a non-uniform distribution for blocks of $sm \geq 3$ (P=0.02 by Kolmogorov-Smirnov test), with many of duplicated genes in the age range 0.4-0.7 D also being components of paralogous regions (Figure 4.8 D).

Together, these results indicate that a burst of gene duplications occurred in the vertebrate lineage approximately 333 – 583 Mya and that genes duplicated at this time tend to be located in paralogous chromosomal segments. A similar breakdown analysis of the larger gene families is not feasible because the property of being present or not in a block is a property of a gene pair and is hard to relate to larger gene families.

4.6 Discussion

We have shown that the human genome contains more large paralogous segments than expected by chance, that an unexpectedly large number of duplicated genes are in the approximate age range 333-583 Mya and that many of the gene pairs of this age are located in paralogous regions. The results are even more striking considering that the human genome is not yet fully sequenced or annotated, which means that the detection of paralogous segments may have been hindered by many genes still being unidentified or assigned to the wrong location. The study by Hogenesch *et al.* (2001) reports large numbers of non-overlapping genes between the Celera and the Ensembl datasets, thereby indicating that large numbers of genes may still be undiscovered in the human genome.

Detections of previously reported blocks confirm the validity of our approach. Some newly discovered paralogous segments with conspicuous gene groupings may indicate functional links between genes (Hughes, 1998). The graphical interface to the results offers an unprecedented wealth of detail and

information and allows the interactive exploration of regions of interest. This will provide a valuable resource for studies related to the human genome.

Previous studies of duplicated regions in human lead to contradictory results, depending on whether approaches based on mapping data or phylogenetic analyses were carried out. Through the combination of both type of methods, applied to a large amount of data, we receive results that are consistent with the 2R hypothesis (Ohno, 1970; Holland *et al.*, 1994) but do not constitute proof of it. They are also consistent with many other possible scenarios (Martin, 1999; Smith *et al.*, 1999) including aneuploidy or an increased rate of production (or fixation) of duplicated chromosomal segments in an early chordate. Polyploidy is probably the most parsimonious explanation, but we do not see any specific evidence that two rounds of polyploidy occurred as opposed to one. Genome sequencing in species such as *Ciona*, *Amphioxus* or lamprey (Thornton, 2001) should throw light on the mechanism by which paralogous segments originated in chordate genomes.

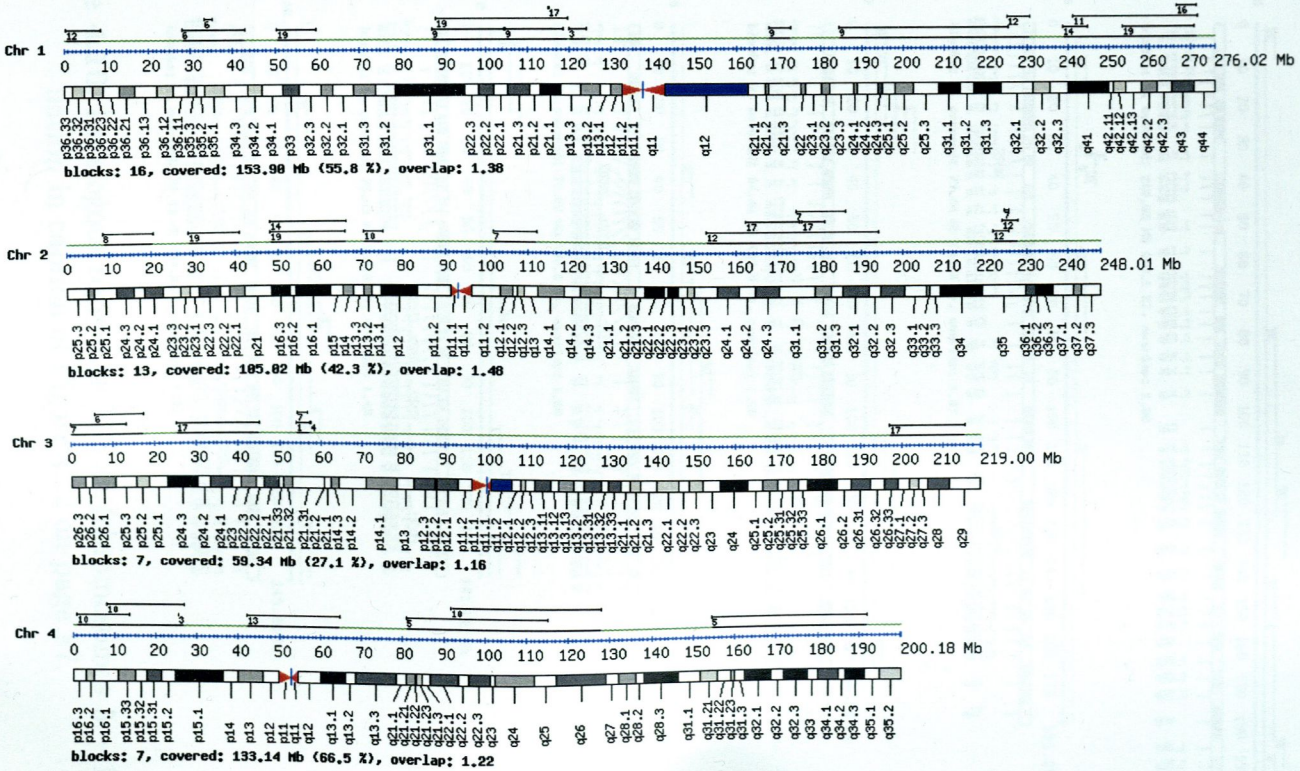


Figure 4.9: Paralogous regions in human (chromosomes 1 – 4); see explanation in caption of Figure 4.12 on page 87.

Figure 4.10: Paralogous regions in human (chromosomes 5 – 10); see explanation in caption of Figure 4.12 on page 87.

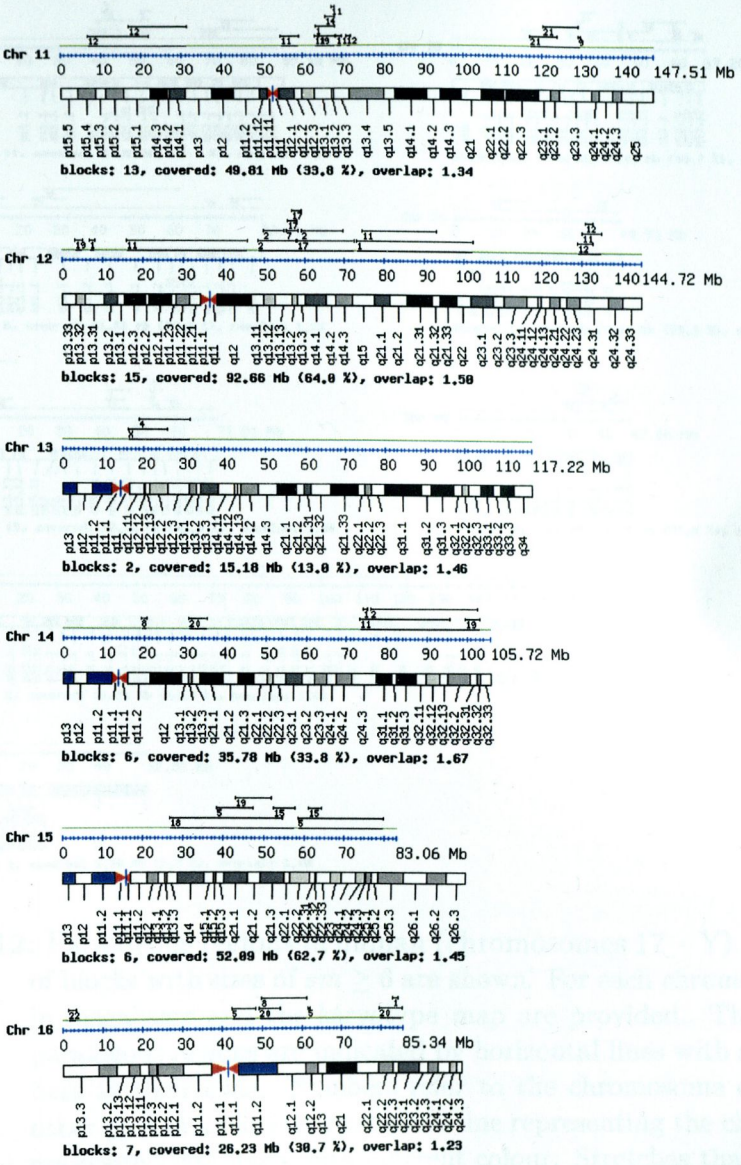


Figure 4.11: Paralogous regions in human (chromosomes 11 – 16); see explanation in caption of Figure 4.12 overleaf.

Figure 4.12: Paralogous regions in human (chromosomes 17 – Y). Only regions of blocks with sizes of $sm \geq 6$ are shown. For each chromosome a scale in megabases and the karyotype map are provided. The location of paralogous regions are indicated by horizontal lines with small vertical bars at their ends. Numbers refer to the chromosome on which the other part of a block is located. A line representing the chromosome is segmented into parts with different colour. Stretches that are covered by blocks are drawn in black, otherwise they are coloured green. A one-line description provides information for each chromosome regarding the number of blocks as well as their coverage and overlap.

Chapter 5

Genomes at a glance

In this chapter I will present applications of my software to whole genome comparisons and genome evolution with emphasis on the possibilities offered through the graphical interface as an interactive web-based tool.

5.1 Intra-genomic comparison in *Staphylococcus aureus*

5.1.1 Background information

It has been shown that bacteria increase their genomic content through lateral transfer and gene duplication, rather than through whole genome duplication (Mira *et al.* (2001) and references therein). In contrast to eukaryotes, a direct correlation between genome size and gene numbers can be observed. Although nearly half the number of genes in most genomes are made up of paralogues it has been reported that this can be modelled through random duplication of mainly single genes (Yanai *et al.*, 2000). Therefore, no large contiguous paralogous regions are expected to be found in bacterial genomes but rather dispersed gene copies or tandem duplicates. In this chapter the genome of *Staphylococcus aureus* is used as a model to demonstrate how our method can

contribute to the study of tandem repeats within microbial genomes.

S. aureus is a Gram-positive aerobic coccus. On the basis of low G+C content (33%) and of ribosomal RNA sequence it is grouped with the Bacillus species. The *S. aureus* genome sequence of strain N315 was recently sequenced (Kuroda *et al.*, 2001). The genome size is 2.81 Mbp and encodes 2594 open reading frames. The genome is fundamentally similar to other prokaryotic genomes, consisting of a circular chromosome with a range of accessory genetic elements including transposons, plasmids, IS-elements, and pathogenicity islands. Most of the antibiotic resistance genes and extracellular virulence determinants (exoenzymes and toxins) are carried on these mobile genetic elements.

5.1.2 Detection of tandem repeats

The complete annotated sequence for *S. aureus* N315 is available through GenBank entry NC_002745. After its download a PERL script was applied to extract gene locations, protein sequences, and additional information in form of description where available. All those data were stored in a MySQL table.

All-against-all search for sequence similarity was carried out through SEG-filtered SSEARCH. With an expectation threshold of $E = 1$, 482 out of 2594 proteins did not produce any hits except with themselves. This left a remaining 2112 proteins for the following analysis. Information for 13,786 query/hit pairs was extracted from the results and stored in a MySQL table.

Tandem duplicated genes were detected through the ORF collapsing method. The parameters used included a maximum allowed gap size of 15 and an expectation threshold of 1e-20. Results were stored in a MySQL table and drawn through the graphical interface. Using a sliding-window technique the tool allows a graphical representation of either the entire genome or particular sections of choice. Figure 5.1 overleaf shows an overview of all tandemly duplicated genes.

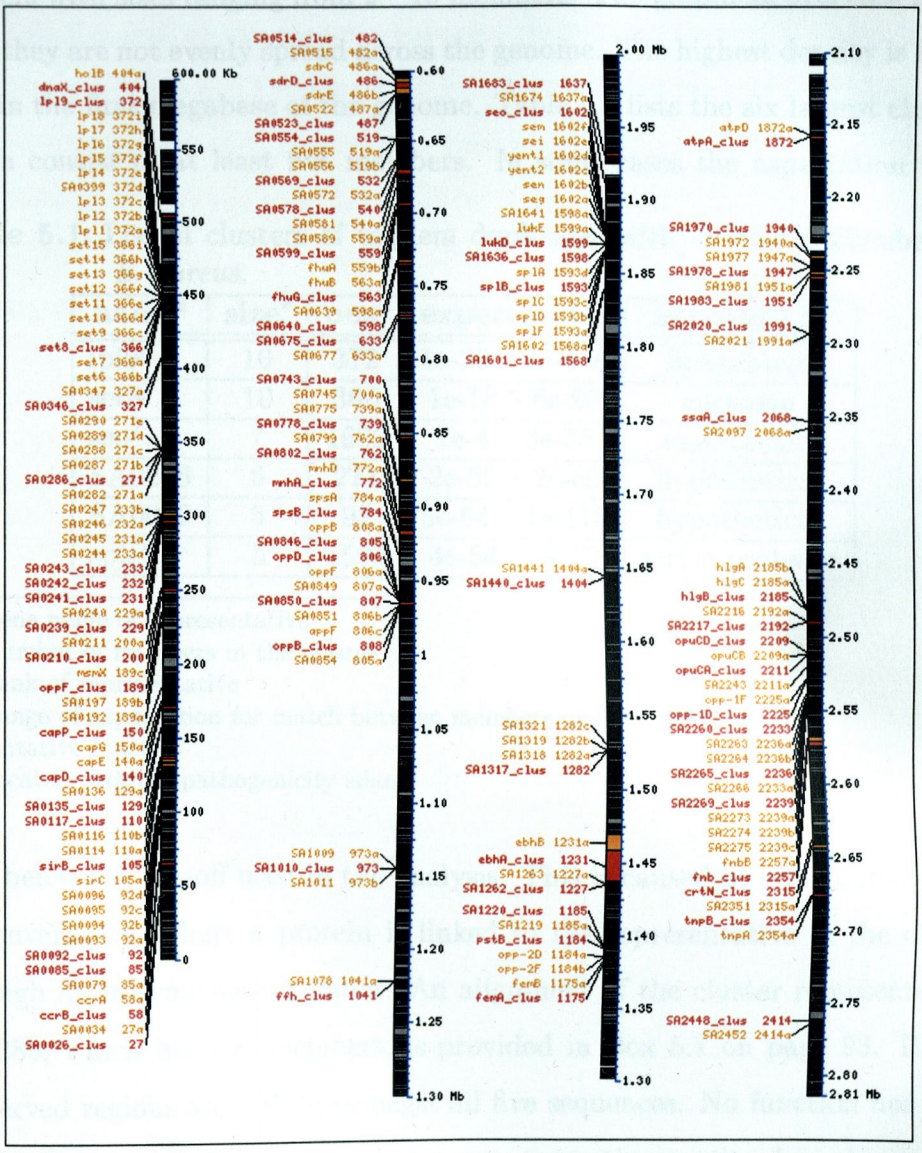


Figure 5.1: Tandem duplicates in *Staphylococcus aureus*. The complete genome is presented in four segments. The scale indicates basepair positions with genes located beside it. A cluster of tandem duplicates is represented by the longest protein, which is drawn in red and has '_clus' appended to its name. The according duplicates have the same position number as the representative with an additional letter appended.

5.1.3 Observations

A total of 198 proteins were detected by this method and grouped into 72 clusters with sizes ranging from 2 – 10 members. The graphical overview shows that they are not evenly spread across the genome. The highest density is found within the first megabase of the genome. Table 5.1 lists the six largest clusters which consist of at least five members. In some cases the expectation value

Table 5.1: List of clusters of tandem duplicates with at least 5 members in *S. aureus*.

name[a]	size[b]	pos[c]	expectation[d]	annotation[e]
lpl9[f]	10	372	1e-36 – 1e-124	lipoprotein
set8[f]	10	366	1e-18 – 6e-96	exotoxin
seo[f]	7	1602	2e-4 – 3e-55	enterotoxin
SA0286	6	271	2e-59 – 2e-69	hypothetical
SA0092	5	92	5e-94 – 1e-118	hypothetical
splB[f]	5	1593	4e-54 – 2e-72	serine protease

[a] gene name of representative
[b] number of members in the cluster
[c] rank of representative
[d] range of expectation for match between members and the representative
[e] putative function
[f] located within a pathogenicity island

falls below the cut-off used in the analysis. This is caused by consecutive steps of convergence, where a protein is linked to the representative of the cluster through an intermediate member. An alignment of the cluster represented by SA0286, which has five members, is provided in Box 5.1 on page 93. Highly conserved regions are visible amongst all five sequences. No function has been assigned to any of those sequences because currently no other homologues can be found in the databases.

In accordance with annotation data from Kuroda *et al.* (2001) it can be observed that large clusters of tandem repeats are located within pathogenicity islands. These islands are accessory genetic elements that carry virulence genes, such as toxins and exoenzymes, and lack essential genes (Hacker *et al.*, 1997). They comprise large DNA regions (up to 200 kb) and contain genes which allow

these elements to be mobile and to spread between bacteria by horizontal gene transfer. Duplication of virulence genes would therefore increase the virulence of strains carrying these elements. Most of the tandemly duplicated genes found in the rest of the genome have yet no function assigned. Considering the deletional bias and the maximised functionality in microbial genomes, any surplus genetic bacterial is generally quickly removed. It follows that multiple preserved copies of one gene must be carrying out important roles that are of substantial benefit to the organism. This recommends the currently unclassified tandem duplicates as possible research targets for future studies of *S. aureus*.

Box 5.1: ClustalW alignment of tandem duplicates represented by SA0286. Sequences are labelled with the GenBank accession number for the protein.

```
BAB41506.1 MTFEEKLSKIYNEIANEISNMIPVEWEKVYTMAYIDDGGGEVFFNYTKPGSDDLNYYTDI 60
BAB41511.1 MTFEEKISKLYNEIANEISSMIPVEWEKVYTMAYIDDGGGEVFFNYTKPGSDDLNYYTDI 60
BAB41513.1 MNFEEKLSQMYNEIANEISGMIPVEWEQVFTIAYVTDQAGEVIFNYTKPGSDELNYYTYI 60
BAB41514.1 MIFEEKLSQMYNEIANEISGMIPVEWEKVYTIAYVDDEGGEVVFNYTKPGSDELNYYTYI 60
BAB41512.1 MTFEEKLSQMYNEIANKISSMIPVEWEKVYAMAYVNERSGEVFYNYTEPRSDELFYYTSV 60
BAB41510.1 MTFEEKLSQMYNEIANKISSMIPVEWEKVYTMAYIDDEGGEVFYYYTEPGSNELYYYTSV 60
           * ****:*::******:**.*******:*:::**: : .***.: **:* *::* *** :

BAB41506.1 PKEYNISVQVFDDLWMDLYDLFEELRDLFKEEDLEPWTSCEFDFTREGKLKVSFDYIDWI 120
BAB41511.1 PKEYNISVQVFDDLWMDLYDLFEELRDLFKEEDLEPWTSCEFDFTSEGKLKVSLDYIDWI 120
BAB41513.1 PREYNVSEKVFYDLWTDLYRLFKKLRNAFKEEDLEPWTSCEFDFTSEGKLKVSFDYIDWI 120
BAB41514.1 PREYNVSEKVFYDLWTDLYRLFKKLRNAFK-EDLEPWTSCEFDFTREGNLKVSFDYIDWI 119
BAB41512.1 LNKYNISRSEFMDSVYELYKQFDKLRDLFKEEGLEPWTSCEFDFTRDGKLNVSFDYIDWA 120
BAB41510.1 LNKYDISESEFMDSAYELYKQFQNLRNIFKEEGYEPWTSCEFDFTKEGELKVSFDYIDWI 120
           .:*::* . * *   :**  *.:**: **.*. ********** :*:*:**:*****

BAB41506.1 NTEFDQLGRENYYMYKKFGVIPEMEYEMEEVKQIEQYIKEQEE--- 163
BAB41511.1 NTEFDQLGRENYYMYKKFGVIPEMEYEMEEIKEIEQYIKEQDEAEL 166
BAB41513.1 NTEFDQLGRENYYMYKKFGVIPEMEYEMEEVKEIEQYIKEQDEAEL 166
BAB41514.1 KLGFGPSGKENYYMYKKFGVLPETEYEMEEIREVEKYVKDQE---- 161
BAB41512.1 NSEFGQMGREHYYMYKKFGIWPEKEYAINWVKK------------- 153
BAB41510.1 NTEFDQLGRQNYYMYKKFGVIPEMEYEMEEVKEIEQYIKEQEEAEQ 166
           :  *.  *:::*********:  ** ** ::  :::  ..  .....
```

5.2 Inter-genomic comparison for selected bacteria

Whole genome comparisons provide a resource for addressing genetic diversity and are crucial to our understanding of the evolution of microbial genomes. Over 56 fully sequenced microbial genomes are currently stored in GenBank (October 2001). The availability of these sequences provides a wealth of new information, allowing researchers to understand the genetic diversity of bacteria and improve their knowledge of microbial evolution. With an ever increasing number of genomes being sequenced the availability of bioinformatics tools to analyse this data is becoming more important. At present there is no web-based interactive tool available to perform multiple genome comparisons. With our approach this gap can be filled. Another tool, made available by the Sanger Centre, is ACT (Artemis Comparison Tool). ACT is a DNA sequence comparison viewer written in Java and based on Artemis (Rutherford *et al.*, 2000). It allows the visualisation and annotation of DNA sequences but offers only pairwise genome alignments as opposed to a multiple genome comparison. Our interface, on the other hand, can be used for the simultaneous presentation of up to four genomes and has the potential to integrate important information including results from similarity searches.

Several systematic analyses have been carried out to study conserved features amongst a variety of microbial organisms but their results are only available as published data and are therefore static. With the escalating number of fully sequenced microbial genomes a public resource for interactive genome comparison would enable researchers to focus on regions within organisms specific to their research interests.

Table 5.2: Bacterial genomes used for inter-genomic comparison

organism	ID	size (bp)	genes[a]	species
Escherichia coli K-12	U00096	4639221	4289	Gram–
Streptococcus pneumoniæ R6	AE007317	2038615	2043	Gram+
Bacillus subtilis	AL009126	4214814	4100	Gram+
Borrelia burgdorferi	AE000783	910724	850	Spirochete

[a]only genes with CDS annotation were counted

5.2.1 Data

To demonstrate the usefulness of the approach for inter-genomic comparisons several fairly distant bacterial genomes were analysed for homologous blocks (see Table 5.2). After download of GenBank entries, coding sequences together with names, locations, and description of genes were extracted and stored in MySQL tables.

5.2.2 Analysis and results

For the block detection analysis the allowed maximum number of unlinked genes between two duplicated genes was set to ten and the expectation threshold to 1e-15. The alignment overlap needs to cover at least 30% of the longer sequence, only hits within an expectation range of 1e-20 are considered, and the number of these hits must not exceed 10. As expected, only very few conserved regions were found, consisting mainly of previously documented operon stuctures.

Figure 5.2 overleaf illustrates a whole genome comparison of four genomes displaying the ribosomal superoperon alignment. As shown previously this operon is highly conserved with some degrees of variation between species. The important feature of the interface presented here is that it shows not only the conserved genes but also additional genes in the neighbourhood. Colour codes illustrate immediately the differences between genomes. Navigation features allow to extend the displayed region into any direction by any size. Integrated hypertext links lead to detailed information on each gene including results

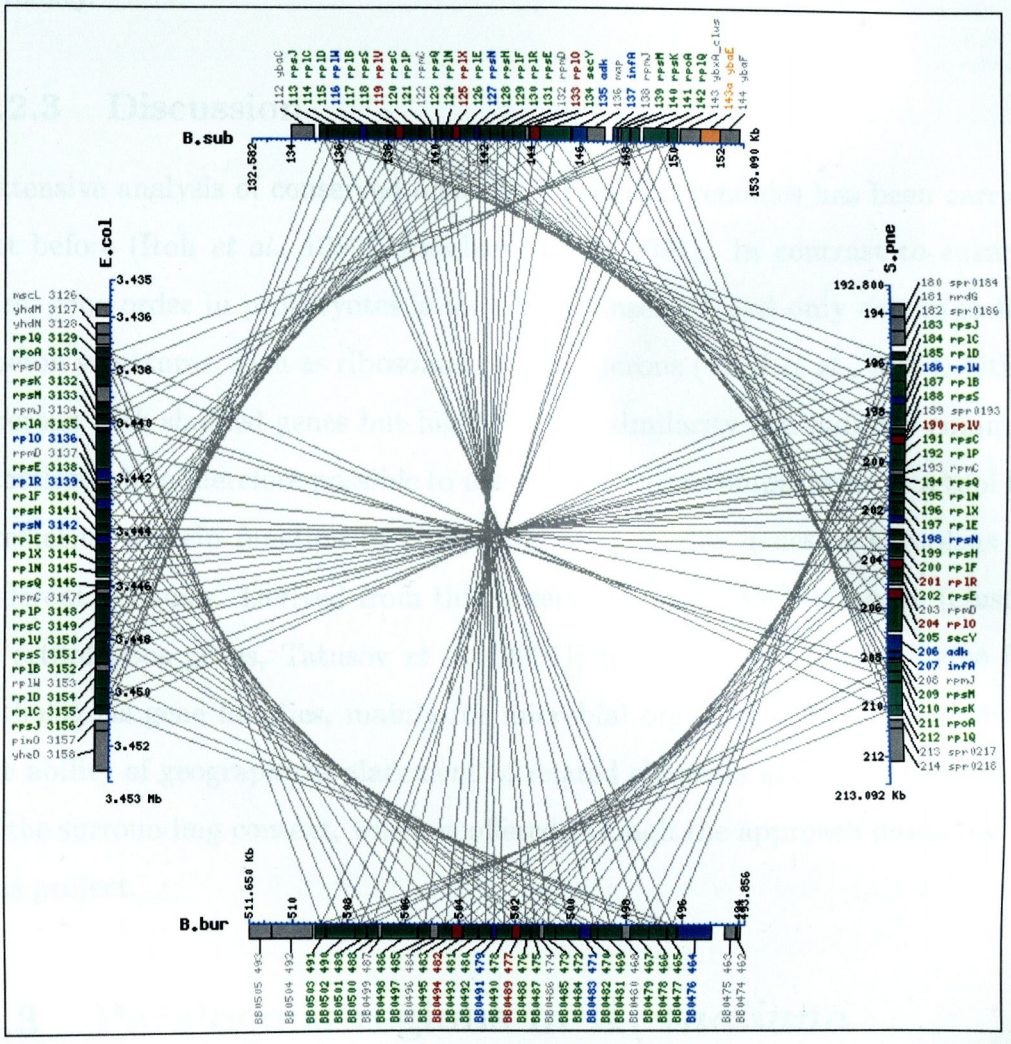


Figure 5.2: Alignment of the ribosomal superoperon in four bacteria. Sections from the genomes of *Escherichia coli*, *Streptococcus pneumoniæ*, *Bacillus subtilis*, and *Borrelia burgdorferi* are shown. Colour codes relate to the number of linked pairs found for a gene: grey = 0, blue = 1, red = 2, green = 3.

from sequence similarity searches and protein alignments. Using the presented example as a model, it is possible to explore any operon structure and study its stability between any microbial genome that has been fully sequenced, making it an extremely beneficial tool for those interested in studying microbial genome diversity.

5.2.3 Discussion

Extensive analysis of conserved operons in bacterial genomes has been carried out before (Itoh *et al.*, 1999; Ermolaeva *et al.*, 2001). In contrast to eukaryotes, gene order in prokaryotes is much less conserved and only occurs in few operon structures, such as ribosomal protein operons (Wolf *et al.*, 2001). Other operons with shuffled genes but high sequence similarity can indicate common functions. It is therefore possible to use whole genome comparisons as a tool for predicting protein function by comparing neighbouring genes of homologs on different genomes. Derived from this observation, the COG database (cluster of orthologous genes, Tatusov *et al.* (2001)) provides a valuable resource for orthologous gene families, mainly for microbial organisms. It lacks, however the ability of geographical placement of related elements and the presentation of the surrounding context, which is offered through the approach developed in this project.

5.3 Paralogous regions in *A. thaliana*

5.3.1 Background

Arabidopsis thaliana is a mustard-like weed of small size with a short life cycle and an enormous seed production. All these factors make it favourable for laboratory use. Despite its lack of agricultural benefits, *A. thaliana* was chosen for a sequencing project because of its small genome: 120 Mbp compared to 2,500 Mbp in maize and 16,000 Mbp in wheat. It is hoped that the genomic

sequence will provide insight into other plant genomes, including key crops. The completion of the *A. thaliana* genome sequence in late 2000 provided the first fully sequenced plant genome (The *Arabidopsis* Genome Initiative, 2000, AGI). Only very few gaps remain which yields high confidence in the quality of the sequence.

Dot-matrix plots from whole genome comparison carried out by AGI with MUMmer show large duplicated regions which support the hypothesis of a recent polyploidy event, as proposed by Ku *et al.* (2000). Extensive analysis by Vision *et al.* (2000) based on a molecular clock method produced classes of duplication ages with four distinctive peaks, hinting at multiple rounds of polyploidy. However, the assumptions of a constant rate of evolution made in these studies cast doubt on the validity of the results. A more reasonable approach to molecular clock based analysis carried out by Lynch and Conery (2000) resulted in evidence for a single, large-scale duplication event 65 Mya.

5.3.2 Analysis

For each chromosome the complete data record was downloaded from Genbank (`ftp://ncbi.nlm.nih.gov/genbank/genomes/A_thaliana/`). These entries were last updated August 13th this year by TIGR (`http://www.tigr.org`). Important data, such as gene names, location, and protein sequences, were extracted and stored in a database. Table 5.3 provides a summary.

Table 5.3: Summary of *A. thaliana* genes

accession	chr	genes	length (bp)
NC_003070	1	6605	29,640,317
NC_003071	2	4122	19,643,621
NC_003072	3	5161	23,333,883
NC_003073	4	3809	17,549,528
NC_003074	5	5845	26,269,328
total	5	25542	116,436,677

The program 'ssearch34' of the FASTA package was used for homology searches amongst all 25,542 proteins. The expectation threshold was set to 1, and the one-line descriptions were limited to 1,000, allowing the report of up to 1,000 hits per protein. Using ten nodes on the cluster, the process took about 15 hours to finish and resulted in a 300 Mb output file. Through a PERL script the relevant information, such as expectation value and alignment overlap, was extracted and stored in a database. 1,195 of the 25,542 query sequences only hit themselves. The rest yielded 1,910,890 query/hit pairs.

Following this, a search for tandem repeats was carried out. In accordance with Vision *et al.* (2000), 1e-20 for the expectation threshold and 15 as the maximum allowed gap between duplicated genes were used as parameters. This resulted in a reduction of genes by 2,960 (11.6%) to 22,582, very similar to the 2,796 genes excluded by Vision *et al.*.

Similar to the analysis of the human genome in the previous chapter, block detection was carried out with many different parameter constellations. Evaluation of the changes in the results produced the following set of parameters: alignment length $\geq 30\%$, expectation threshold $\leq 1e-15$, gap between linked genes ≤ 10, expectation range $\leq 1e-20$, and number of hits allowed within this range ≤ 20.

5.3.3 Results

Figure 5.3 overleaf shows an overview of blocks for all five chromosomes with size $sm \geq 7$ that resulted from the preceeding analysis. Several outstanding features can be observed immediately:

- The blocks are much less shuffled than in the human genome. This relates to the possible 8- to 9-fold difference in the age estimations for according duplication events.

- Nearly all of the genome seems to be covered by blocks, which indicates

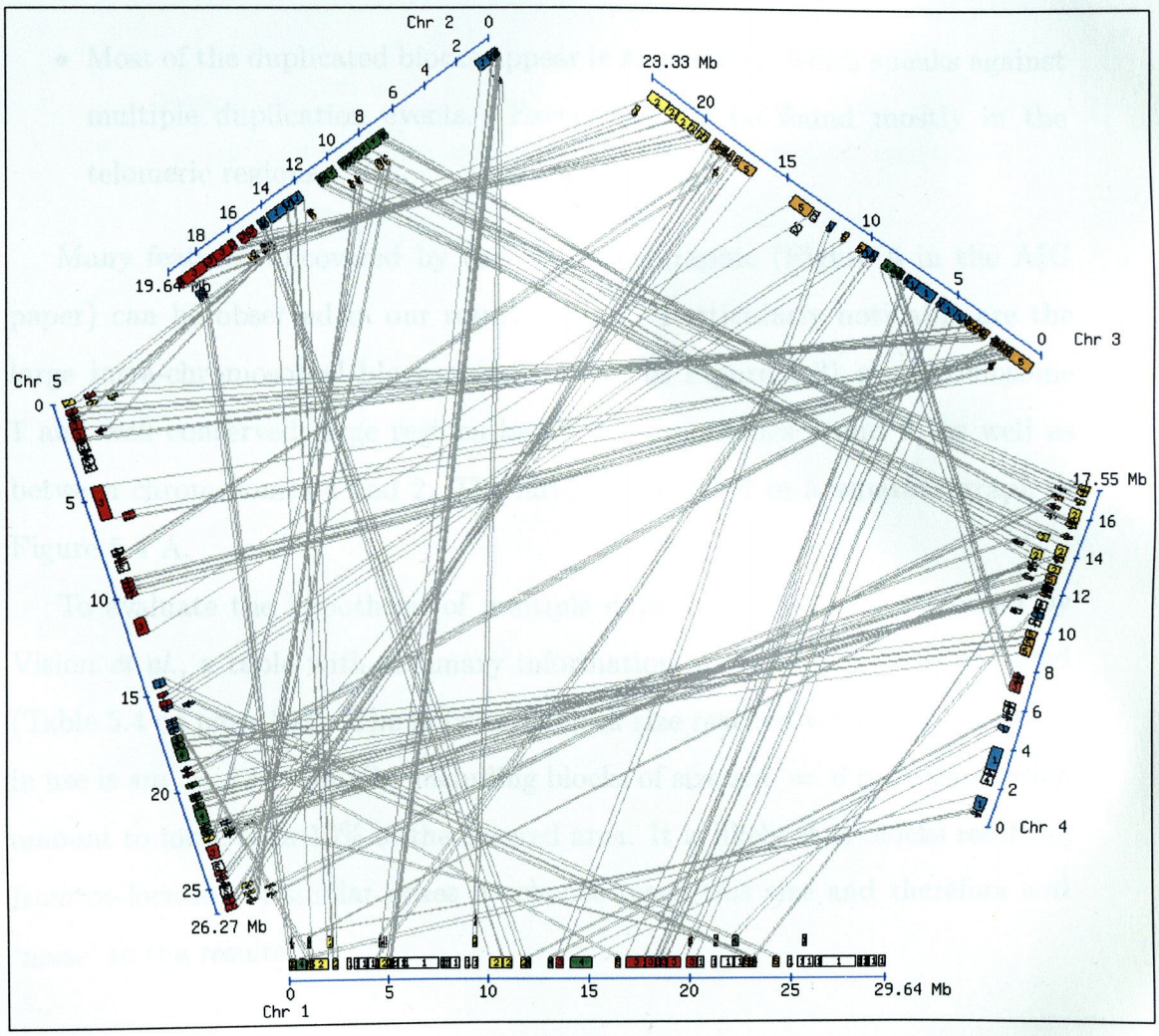


Figure 5.3: Paralogous regions in *A. thaliana*

duplications of large scale, possibly polyploidy.

- The centromeric regions have a consipicous absence of blocks. It shows clearly in form of clouds of dots in the plots produced by AGI (e.g. Figure 5.4B overleaf) that these regions contain a large amount of small, unordered segments with high similarity to other centromeric regions.

- Most of the duplicated blocks appear in single layer, which speaks against multiple duplication events. Exceptions can be found mostly in the telomeric regions.

Many features uncovered by the MUMmer graphic (Figure 4 in the AIG paper) can be observed in our results as well. Particularly noticable are the large intra-chromosomal blocks (white boxes in Figure 5.3) on chromosome 1 and well conserved large regions between chromosomes 3 and 2, as well as between chromosomes 4 and 2. The latter is presented in a separate graph in Figure 5.4 A.

To evaluate the hypothesis of multiple duplication events as proposed by Vision *et al.*, a table with summary information on block overlap is provided (Table 5.4 on page 103). The maximum block size reached with the parameters in use is $sm = 136$. Only by including blocks of size $sm = 6$ does the overlap amount to more than 10% of the covered area. It is likely that blocks resulting from co-location of similar genes by chance reach this size and therefore add 'noise' to the results.

5.3.4 Discussion

A preliminary glance at the results obtained from our block detection method indicates a solid agreement with results shown in The *Arabidopsis* Genome Initiative (2000). A positive aspect of our analysis is visible by the non-detection of blocks in the centromeric regions. They form examples of spurious data which are to be segregated from paralagous blocks. The multifold coverage

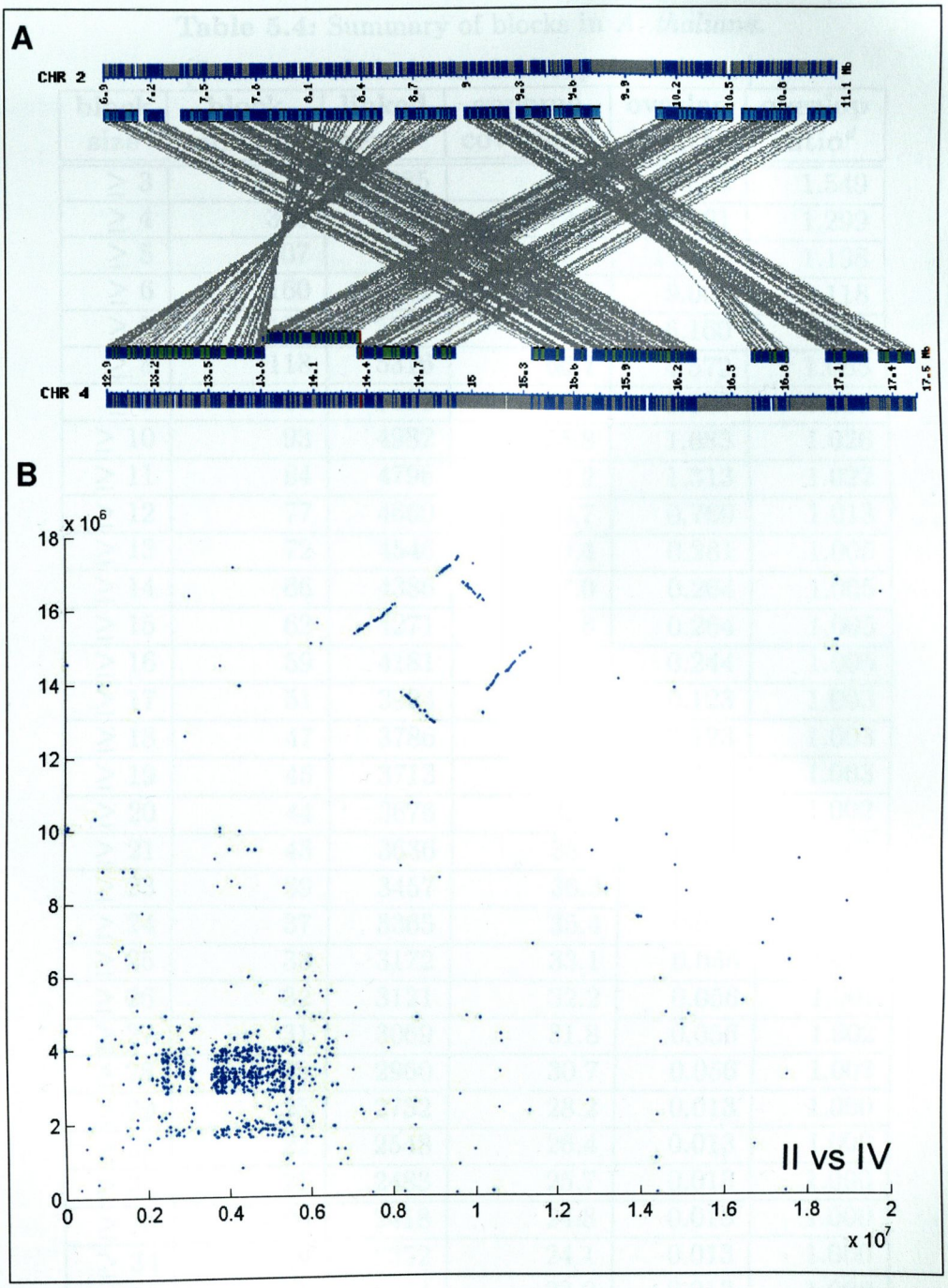


Figure 5.4: Large duplicated blocks between parts of chromosomes 2 and 4 *Arabidopsis*. Figure A shows blocks for parts of the chromosomes derived with our method. Blocks of sizes $sm \geq 7$ are shown together with the linked genes (in blue) and the connections between them. Regions with unlinked genes appear grey. Figure B was taken from The *Arabidopsis* Genome Initiative (2000, supplementary information) and presents the whole chromosomes. The region in the upper graph can be seen here as a diamond. A large cloud of blocks is visible around the location of the centromere.

Table 5.4: Summary of blocks in *A. thaliana*.
[Summary of blocks in *Arabidopsis thaliana*.]

block size	block numbers	linked genes[a]	genome coverage[b]	overlap length[c]	overlap ratio[d]
≥ 3	579	7385	78.0	49.832	1.549
≥ 4	300	6546	72.8	25.391	1.299
≥ 5	207	6081	69.8	16.127	1.198
≥ 6	160	5731	65.4	9.002	1.118
≥ 7	134	5500	62.9	6.160	1.084
≥ 8	118	5316	60.7	4.572	1.065
≥ 9	101	5092	56.8	2.607	1.039
≥ 10	93	4982	55.8	1.683	1.026
≥ 11	84	4796	52.2	1.313	1.022
≥ 12	77	4660	50.7	0.760	1.013
≥ 13	72	4546	49.4	0.281	1.005
≥ 14	66	4386	47.0	0.264	1.005
≥ 15	62	4271	45.8	0.264	1.005
≥ 16	59	4181	44.8	0.244	1.005
≥ 17	51	3924	41.7	0.123	1.003
≥ 18	47	3786	40.2	0.123	1.003
≥ 19	45	3713	39.3	0.123	1.003
≥ 20	44	3676	38.9	0.110	1.002
≥ 21	43	3636	38.3	0.110	1.002
≥ 23	39	3457	36.3	0.075	1.002
≥ 24	37	3365	35.4	0.075	1.002
≥ 25	33	3172	33.1	0.056	1.001
≥ 26	32	3121	32.2	0.056	1.001
≥ 27	31	3069	31.8	0.056	1.002
≥ 28	29	2960	30.7	0.056	1.002
≥ 29	25	2732	28.2	0.013	1.000
≥ 31	22	2548	26.4	0.013	1.000
≥ 32	21	2483	25.7	0.013	1.000
≥ 33	20	2418	24.8	0.013	1.000
≥ 34	19	2352	24.1	0.013	1.000
≥ 35	18	2284	23.2	0.013	1.000
≥ 39	14	1968	19.8	0.000	1.000
≥ 40	13	1889	19.1	0.000	1.000

[a] gene pairs forming connections between paralogous regions
[b] coverage in percentage of 116,436,677 bp
[c] amount of sequence covered more than once (in megabases)
[d] overlap = (sum of lengths of all blocks) / coverage

in some telomeric regions, on the contrary, was detected in form of blocks, which suggests that these regions should be excluded similar to the yeast analysis in chapter 3.

Much of the data suggests a fairly recent large-scale duplication event, possibly polyploidy, but we cannot see proof of multiple occurences of it. These results are not conclusive and require more extensive analysis. However, this section only served as a stimulator to show the capabilities of the new software. The continuation of the study of duplicated blocks in *A. thaliana* will involve considerable more time and effort and will be left for a subsequent project.

Chapter 6

Conclusion

The goal of this work was to develop a method for the detection of intra-genomic duplications, suitable for application to large genomes, such as that of man. The amount and complexity of the underlying biological data required special equipment and methods from the field of bioinformatics. I will conclude with a summary of the results achieved in this project.

6.1 Computing platform

To meet the demands of high-performance computing a Beowulf-type cluster was built. For the parallelisation of time-intensive procedures, such as sequence similarity searches, I developed a wrapper script called `wrapid` in the style of the MOLLUSCS system. Several improvements were achieved:

- Automatic load balancing was added, which became necessary after including heuristic search programs into the system. In MOLLUSCS the user needs to split up the database into chunks according to the speed of the computing nodes. This approach is not suitable for programs such as FASTA and BLAST, where the time for a search can vary according to the amount of similarity found, regardless of the length of the sequences. The load balancing automatically splits up the search input into chunks

that are small enough to avoid idle nodes in most cases.

- In a multi-user environment resources can easily be over-exploited. To avoid this, `wrapid` checks the CPU load on nodes before parallel jobs are started and only includes machines that have sufficient resources.
- Large computing jobs can last extensive time spans during which the system usage and requirements might change. The design of `wrapid` permits dynamic addition or removal of nodes from the group of clients in use. This allows resources to be freed up if required or more resources to be integrated if available and provides harmonious interaction with other users.
- Extra checks were added to assure that all parts of a task were completed properly. Again, this feature evolved from necessity when some of the nodes unexpectedly stopped the execution of their assigned jobs due to hardware errors.

These are important additions to an excellent program from which other users will benefit as well.

6.2 Flexibility

The overall approach to genome comparison was developed with the possibility of future expansions in mind. Considering the increasing amount of sequence data and the growing number of completed genomes, this is certainly an important consideration.

A Beowulf cluster provides the most flexible solution for high-performance computing. The expandability of the system was already demonstrated during the course of the project when an extensive upgrade of the cluster was carried out. Analogously, sequence similarity search – the basic step in the block detection method – was implemented as a separate module using external programs.

Developments in this field happen on a broad front, and the modularity of our approach facilitates integration of improved methods as soon as they become available.

6.3 Automation of block detection

For a project on the scale of the human genome, achievement of high confidence in the results despite full automation is of great importance. Application of our method to the yeast genome has shown that manually edited paralogous regions can be well approximated. Blocks detected in the analysis of the human genome produced previously reported regions, sometimes with further expansions, as well as additional regions. Statistical analysis of block sizes derived from randomised data proved the significance of the findings.

6.4 Search for polyploidy

The 2R hypothesis was tested in human through a combination of paralogous regions with phylogenetic analysis of gene families. This comprises the most extensive application of both approaches to a vertebrate genome to date. The general trend of the results shows supporting evidence for at least one round of whole genome duplication in an ancient vertebrate. However, these findings are not clear-cut and do not exclude alternative explanations, such as, for example, a series of large chromosomal duplications. More data, particularly from other species, and probably more sensitive methods are required to resolve the question of the validity of the 2R hypothesis.

6.5 Web resources

The WWW-interface for human and the updated webpages for *S. cerevisiæ* provide valuable resources for studies of molecular evolution and the development

of genes and gene families. Due to the popularity of the Internet, WWW-browsers have become common tools for modern researchers that are used on a daily basis. Thus, no new program needs to be learned to access the results, which can be explored interactively. This allows researchers to easily retrieve information on paralogous regions from different angles and perspectives. The graphical presentation is an ideal starting point to grasp the organisation of the blocks and their placement within a broader context. Graphics of multiple interconnected regions allow the visualisation of complex genome-wide correlations. Thanks to the online availability of a plethora of biological databases, it is possible to integrate links to external resources, which add further information and value to the results.

6.6 Outlook

The sequencing and annotation of the human genome is an on-going process. Updates of the detected blocks need to be carried out as soon as new data becomes available and will produce results of increasing quality and accuracy. Genome projects for other vertebrates, such as mouse and pufferfish, are already underway. They will provide interesting targets for future applications of our approach.

Great benefit can also be expected from inter-genomic comparisons. As was shown by the example of microbial genomes, the system can be adapted to expand the block detection and the presentation of results to multiple organisms. The availability of additional yeast genomes, particularly non-polyploids, such as *Candida albicans*, will help in resolving the origin of paralogous regions in *S. cerevisiæ*. Our studies indicate that intra-genomic analysis might not yield sufficient results for unveiling the mystery of polyploidisation in vertebrates. Therefore, interspecies comparison will be the best way to determine the history of the evolution of the human genome.

Due to the complexity of genome comparison the approach presented here

will never fully satisfy all possible aspects, but the results so far are a significant improvement on existing tools and resources and have good potential for synergistically enhancing genomics research.

Part II

Chapter 7

PubCrawler

7.1 Introduction

During the duration of the project, work was carried out on a side project that presents another example of how a bioinformatics tool can improve scientific research. The results were published (Hokamp and Wolfe, 1999) and are described in the following section.

7.2 What's new in the library? What's new in GenBank? Let PubCrawler tell you!

The scientific literature is growing so quickly that many scientists no longer have time to scan the latest issues of all the journals relevant to their interests. Thanks to online services, such as NCBI's Entrez (McEntyre, 1998), it has become possible to search huge libraries for specific articles without leaving the desk. Entrez provides free access to several scientific databases including PubMed (McEntyre and Lipman, 2001), the world's largest database of biomedical literature. PubMed currently holds the abstracts of approximately 10 million scientific journal articles, including the complete contents of MEDLINE. New articles in any research field can be expected nearly at a daily rate,

so staying abreast of the current state of science requires frequent electronic library searches. Interesting documents can be overlooked if searches are not made regularly, but carrying out searches can be uninteresting and laborious, particularly at times of day when traffic on the Internet is slow.

The repetitive querying of online databases can easily be automated by computer. The PubCrawler WWW service is an automated update alerting service for users of NCBI's PubMed (literature) and GenBank (DNA sequence) databases. PubCrawler carries out personalised searches at NCBI at regular intervals (e.g., daily), keeps track of what records have been seen previously, and produces a WWW page listing the latest hits that match the user's interests. The following sections describe several features of this service and how to access it.

Automated update delivery

PubCrawler is intended for scientists who want to be informed of the latest publications in their field of interest, as soon as they appear in PubMed. Its journal-monitoring function is similar to that of services that search commercial databases, such as Current Contents Online (Institute for Scientific Information, Philadelphia, PA), or SciSearch at LANL (Los Alamos National Laboratory). However, PubCrawler has the twin advantages of being free and of being able to monitor new DNA sequences in GenBank as well. Often, a new sequence appears in GenBank months before the paper describing it is published. PubCrawler searches the annotation text of GenBank entries, not the sequence data itself. This can be complemented by services performing BLAST searches against DNA and protein databases, such as SIB's Swiss-Shop `http://www.expasy.ch/swiss-shop/`, the Sequence Alerting System in Peer Bork's lab at EMBL `http://www.bork.embl-heidelberg.de/Alerting/`, or NCBI's XREFdb `http://www.ncbi.nlm.nih.gov/XREFdb/`.

A new PubCrawler user must first create a search profile, consisting of one or

more Entrez queries. For example, someone interested in fruitfly protein kinase genes could set up a profile that searches for papers in PubMed whose abstracts contain the words Drosophila, and gene or DNA, and the phrase protein kinase. A second query in the profile might search for papers with particular author names (e.g., rival fruitfly protein kinase labs). A third query might scan GenBank for new sequences where the organism is *Drosophila melanogaster* and the annotation contains the word kinase. Any number of queries can be combined into the search profile. A WWW site, the PubCrawler Configurator, has been set up to help with building and editing search profiles. This allows users to check the syntax of their queries and to get a feel for how many database entries might match each query in the profile. For PubCrawler to work effectively, the search profile should be neither too broad nor too specific, so that a manageable number of hits are returned each day. Because the search profile can include an unlimited number of queries, and is stored and can be edited at any time using the Configurator WWW page, users can build more detailed and comprehensive search profiles than they would by occasional use of Entrez.

When setting up a search profile, the user chooses how often the searches are to be run (for example, daily on Monday - Friday). They can then browse their results every day at the PubCrawler WWW site, or can have the results e-mailed to them as an HTML document (viewable with mail programs such as Netscape Mail or Microsoft Outlook Express) or as plain text (although this loses the usefulness of having hypertext links to NCBI). All personal information and profile data remains confidential and is password protected. Within the first three years more than 13,000 users from all over the world have registered with the PubCrawler's WWW-service.

The sample results page (Figure 7.2 on page 116) illustrates some of PubCrawler's useful features. The output for each user is a single HTML page, readable with any WWW browser. Each day's new results are presented exactly

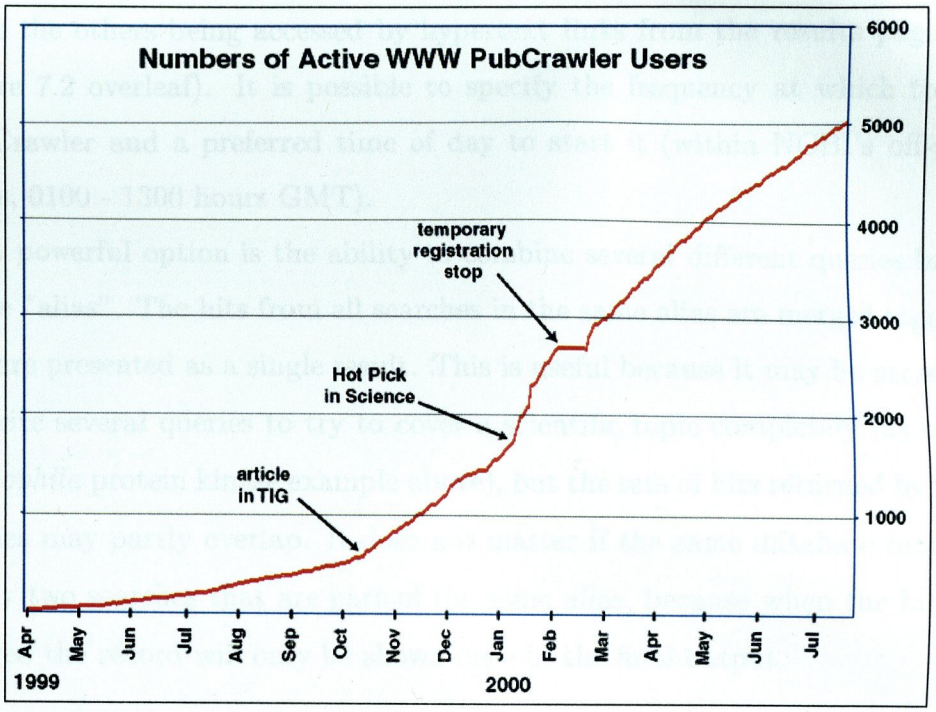


Figure 7.1: PubCrawler user numbers. The graphic shows the increase of PubCrawler registrations and several events which influenced this development.

the way they were received from the NCBI site, with hypertext links (e.g., to view complete Abstract or Sequence information at NCBI for any of the documents found). A quick index at the top shows at first glance how many documents were found for each search topic. PubCrawler keeps track of all the articles that have been presented, so every time it runs, only new hits matching the personal search profile (i.e., those not previously seen by PubCrawler) will be presented. This avoids the"have I read this before?" feeling common to absent-minded academics. The results page provides access to older results up to a few days old (the default is seven days, but that can be adjusted) via hypertext links, allowing people to catch up on missed days.

Options

The PubCrawler Configurator allows users to customise their searches in many ways. The variable parameters include the maximum number of documents to retrieve, their maximum age, and how many titles to show on the results page

(with the others being accessed by hypertext links from the results page; see Figure 7.2 overleaf). It is possible to specify the frequency at which to run PubCrawler and a preferred time of day to start it (within NCBI's off-peak hours, 0100 - 1300 hours GMT).

A powerful option is the ability to combine several different queries into a single "alias". The hits from all searches in the same alias are merged together and are presented as a single result. This is useful because it may be necessary to write several queries to try to cover a scientific topic completely (as in the *Drosophila* protein kinase example above), but the sets of hits returned by these queries may partly overlap. It does not matter if the same database record is hit by two searches that are part of the same alias, because when the hits are merged the record will only be shown once in the final output.

Availability

The PubCrawler WWW service `http://www.pubcrawler.ie` is offered without charge. Its home page provides a link to the PubCrawler Configurator for setting up personal search profiles.

Additionally, the PubCrawler program is freely available for stand-alone installation on a PC, Macintosh or Unix system. The program is written in Perl, which is available for every operating system at no cost. Detailed downloading and installation instructions are available at the same WWW site.

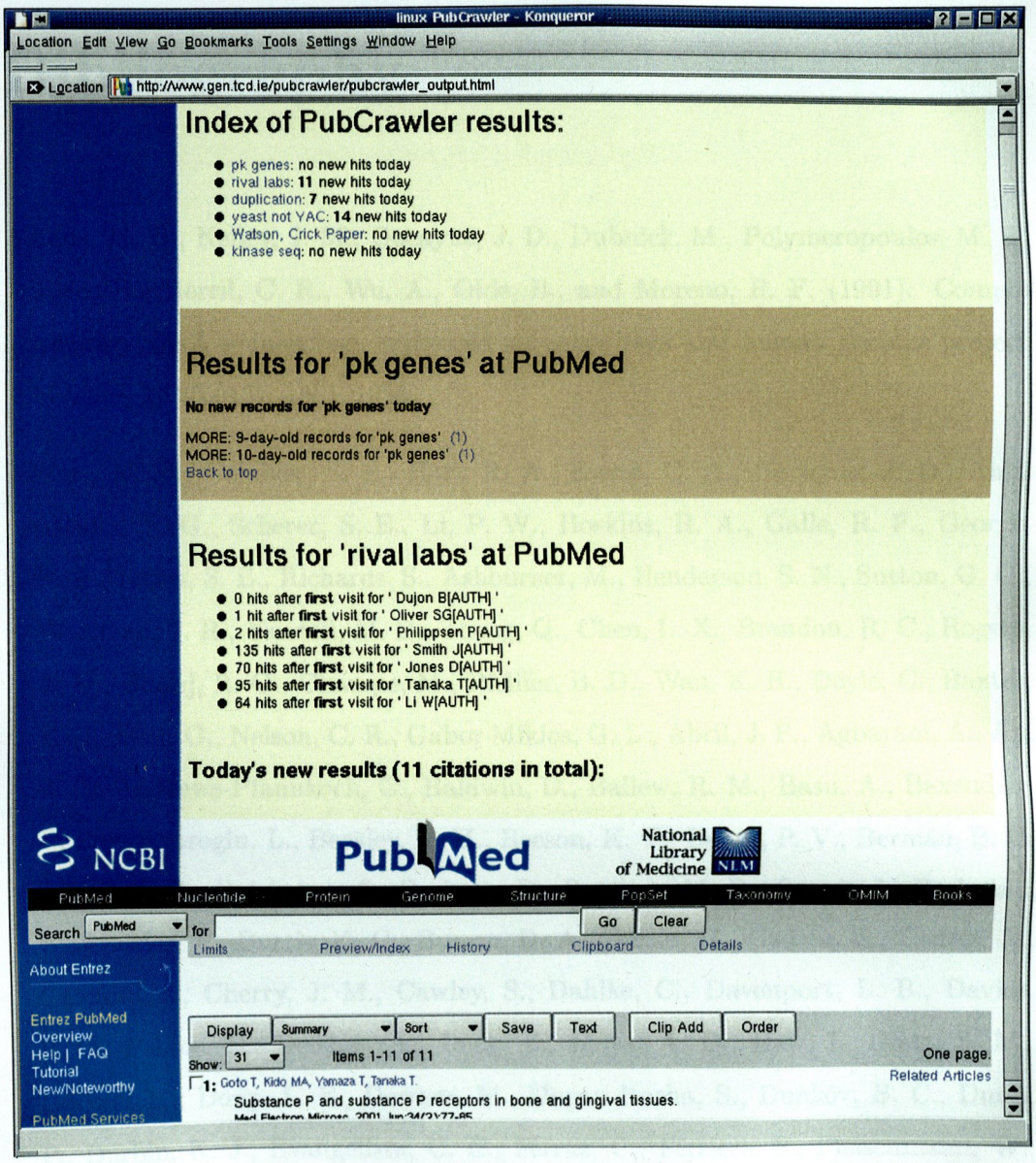


Figure 7.2: Screenshot of PubCrawler results. Sample PubCrawler results file. PubCrawler incorporates Entrez' original output into a web page and wraps it up with useful information and hypertext links.

Bibliography

Adams, M. D., Kelley, J. M., Gocayne, J. D., Dubnick, M., Polymeropoulos, M. H., Xiao, H., Merril, C. R., Wu, A., Olde, B., and Moreno, R. F. (1991). Complementary DNA sequencing: expressed sequence tags and human genome project. *Science*, **252**(5013), 1651–1656.

Adams, M. D., Celniker, S. E., Holt, R. A., Evans, C. A., Gocayne, J. D., Amanatides, P. G., Scherer, S. E., Li, P. W., Hoskins, R. A., Galle, R. F., George, R. A., Lewis, S. E., Richards, S., Ashburner, M., Henderson, S. N., Sutton, G. G., Wortman, J. R., Yandell, M. D., Zhang, Q., Chen, L. X., Brandon, R. C., Rogers, Y. H., Blazej, R. G., Champe, M., Pfeiffer, B. D., Wan, K. H., Doyle, C., Baxter, E. G., Helt, G., Nelson, C. R., Gabor Miklos, G. L., Abril, J. F., Agbayani, A., An, H. J., Andrews-Pfannkoch, C., Baldwin, D., Ballew, R. M., Basu, A., Baxendale, J., Bayraktaroglu, L., Beasley, E. M., Beeson, K. Y., Benos, P. V., Berman, B. P., Bhandari, D., Bolshakov, S., Borkova, D., Botchan, M. R., Bouck, J., Brokstein, P., Brottier, P., Burtis, K. C., Busam, D. A., Butler, H., Cadieu, E., Center, A., Chandra, I., Cherry, J. M., Cawley, S., Dahlke, C., Davenport, L. B., Davies, P., de Pablos, B., Delcher, A., Deng, Z., Mays, A. D., Dew, I., Dietz, S. M., Dodson, K., Doup, L. E., Downes, M., Dugan-Rocha, S., Dunkov, B. C., Dunn, P., Durbin, K. J., Evangelista, C. C., Ferraz, C., Ferriera, S., Fleischmann, W., Fosler, C., Gabrielian, A. E., Garg, N. S., Gelbart, W. M., Glasser, K., Glodek, A., Gong, F., Gorrell, J. H., Gu, Z., Guan, P., Harris, M., Harris, N. L., Harvey, D., Heiman, T. J., Hernandez, J. R., Houck, J., Hostin, D., Houston, K. A., Howland, T. J., Wei, M. H., Ibegwam, C., *et al.* (2000). The genome sequence of *Drosophila melanogaster*. *Science*, **287**(5461), 2185–2196.

Altman, R. B. (1998). A curriculum for bioinformatics: the time is ripe. *Bioinformatics*, **14**(7), 549–550.

Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*, **25**(17), 3389–3402.

Bains, W. (1996). Company strategies for using bioinformatics. *Trends Biotechnol*, **14**(8), 312–317.

Ball, C. A. and Cherry, J. M. (2001). Genome comparisons highlight similarity and diversity within the eukaryotic kingdoms. *Curr Opin Chem Biol*, **5**(1), 86–89.

Barak, A., La'adan, O., and Shiloh, A. (1999). Scalable cluster computing with MOSIX for LINUX. In *Proceedings of the 5-th Annual Linux Expo.*, pages 95–100.

Baxevanis, A. D. (2001). The molecular biology database collection: an updated compilation of biological database resources. *Nucleic Acids Res*, **29**(1), 1–10.

Benton, D. (1996). Bioinformatics–principles and potential of a new multidisciplinary tool. *Trends Biotechnol*, **14**(8), 261–272.

Birney, E. and Durbin, R. (2000). Using genewise in the drosophila annotation experiment. *Genome Res*, **10**(4), 547–548.

Blank, T. (1990). The MASPAR MP-1 architecture. In *Thirty-Fifth IEEE Computer Society International Conference - Compcon Spring 90*, pages 20–4, San Francisco, CA.

Boguski, M. S. (1994). Bioinformatics. *Curr Opin Genet Dev*, **4**(3), 383–388.

Boguski, M. S., Lowe, T. M., and Tolstoshev, C. M. (1993). dbEST – database for expressed sequence tags. *Nat Genet*, **4**(4), 332–333.

Brenner, S. E., Chothia, C., and Hubbard, T. J. (1998). Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc Natl Acad Sci U S A*, **95**(11), 6073–6078.

Brooke, N. M., Garcia-Fernandez, J., and Holland, P. W. (1998). The ParaHox gene cluster is an evolutionary sister of the Hox gene cluster. *Nature*, **392**(6679), 920–922.

Burge, C. and Karlin, S. (1997). Prediction of complete gene structures in human genomic DNA. *J Mol Biol*, **268**(1), 78–94.

Castagnera, K., Cheng, D., Fatoohi, R., Hook, E., Kramer, B., Manning, C., Musch, J., Niggley, C., Saphir, W., Sheppard, D., Smith, M., Stockdale, I., Welch, S., Williams, R., and Yip, D. (1994). Clustered workstations and their potential role as high speed compute processors. Tech. Rep. RNS-94-003, NAS Systems Division, NASA Ames Research Center.

Chao, K. M., Zhang, J., Ostell, J., and Miller, W. (1997). A tool for aligning very similar DNA sequences. *Comput Appl Biosci*, **13**(1), 75–80.

Cheng, C. H. and Chen, L. (1999). Evolution of an antifreeze glycoprotein. *Nature*, **401**(6752), 443–444.

Chervitz, S. A., Aravind, L., Sherlock, G., Ball, C. A., Koonin, E. V., Dwight, S. S., Harris, M. A., Dolinski, K., Mohr, S., Smith, T., Weng, S., Cherry, J. M., and Botstein, D. (1998). Comparison of the complete protein sets of worm and yeast: orthology and divergence. *Science*, **282**(5396), 2022–2028.

Claverie, J. M. (2000). From bioinformatics to computational biology. *Genome Res*, **10**(9), 1277–1279.

Courtois, P. R. and Moncany, M. L. (1995). A probabilistic algorithm for interactive huge genome comparison. *Comput Appl Biosci*, **11**(6), 657–665.

Cuff, J. A., Birney, E., Clamp, M. E., and Barton, G. J. (2000). ProtEST: protein multiple sequence alignments from expressed sequence tags. *Bioinformatics*, **16**(2), 111–116.

Danchin, A. (2000). A brief history of genome research and bioinformatics in France. *Bioinformatics*, **16**(1), 65–75.

de la Vega, F. M., Giegerich, R., and Fuellen, G. (1996). Distance education through the internet: the GNA-VSNS biocomputing course. *Pac Symp Biocomput*, pages 203–215.

Delcher, A. L., Kasif, S., Fleischmann, R. D., Peterson, J., White, O., and Salzberg, S. L. (1999). Alignment of whole genomes. *Nucleic Acids Res*, **27**(11), 2369–2376.

Dongarra, J. J., Meuer, H. W., and Strohmaier, E. (2000). Top500 computers. `http://www.top500.org`.

Doolittle, R. F. (1986). *Of URFs and ORFs.* University Science Books, Mill Valley, CA, USA.

C. elegans Sequencing Consortium (1998). Genome sequence of the nematode *C. elegans*: a platform for investigating biology. *Science*, **282**(5396), 2012–2018.

Ermolaeva, M. D., White, O., and Salzberg, S. L. (2001). Prediction of operons in microbial genomes. *Nucleic Acids Res*, **29**(5), 1216–1221.

Fannon, M. R. (1996). Gene expression in normal and disease states–identification of therapeutic targets. *Trends Biotechnol*, **14**(8), 294–298.

Feuermann, M., de Montigny, J., Potier, S., and Souciet, J. L. (1997). The characterization of two new clusters of duplicated genes suggests a 'lego' organization of the yeast *Saccharomyces cerevisiæ* chromosomes. *Yeast*, **13**(9), 861–869.

Fischer, D., Barret, C., Bryson, K., Elofsson, A., Godzik, A., Jones, D., Karplus, K. J., Kelley, L. A., MacCallum, R. M., Pawowski, K., Rost, B., Rychlewski, L., and Sternberg, M. (1999). CAFASP-1: critical assessment of fully automated structure prediction methods. *Proteins*, **Suppl 3**, 209–217.

Fitch, W. M. (1970). Distinguishing homologous from analogous proteins. *Syst Zool*, **19**(2), 99–113.

Friedman, R. and Hughes, A. L. (2001). Gene duplication and the structure of eukaryotic genomes. *Genome Res*, **11**(3), 373–381.

Gallardo, M. H., Bickham, J. W., Honeycutt, R. L., A., O. R., and Kohler, N. (1999). Discovery of tetraploidy in a mammal. *Nature*, **401**(6751), 341.

Gee, H. (2000). Homegrown computer roots out phylogenetic networks. *Nature*, **404**(6775), 214.

GenomeWeb (2001). Compaq and EBS break billion comparison-per-second barrier in 1 GHz alpha demo. `http://www.genomeweb.com/articles/view-article.asp?Article=200172311502%7`.

Goffeau, A., Barrell, B. G., Bussey, H., Davis, R. W., Dujon, B., Feldmann, H., Galibert, F., Hoheisel, J. D., Jacq, C., Johnston, M., Louis, E. J., Mewes, H. W., Murakami, Y., Philippsen, P., Tettelin, H., and Oliver, S. G. (1996). Life with 6000 genes. *Science*, **274**(5287), 546, 563–546, 567.

Gotoh, O. (1982). An improved algorithm for matching biological sequences. *J Mol Biol*, **162**(3), 705–708.

Guimaraes, C. T., Sills, G. R., and Sobral, B. W. S. (1997). Comparative mapping of andropogoneae: *Saccharum l.* (sugarcane) and its relation to *Sorghum* and maize. *Proc Natl Acad Sci U S A*, **94**(26), 14261–14266.

Hacker, J., Blum-Oehler, G., Muhldorfer, I., and Tschape, H. (1997). Pathogenicity islands of virulent bacteria: structure, function and impact on microbial evolution. *Mol Microbiol*, **23**(6), 1089–1097.

Hagen, J. B. (2000). The origins of bioinformatics. *Nat Rev Genet*, **1**(3), 231–236.

Henikoff, S. and Henikoff, J. G. (1992). Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A*, **89**(22), 10915–10919.

Hogenesch, J. B., Ching, K. A., Batalov, S., Su, A. I., Walker, J. R., Zhou, Y., Kay, S. A., Schultz, P. G., and Cooke, M. P. (2001). A comparison of the Celera and Ensembl predicted gene sets reveals little overlap in novel genes. *Cell*, **106**(4), 413–415.

Hokamp, K. and Wolfe, K. (1999). What's new in the library? What's new in GenBank? Let PubCrawler tell you! *Trends Genet*, **15**(11), 471–472.

Holland, P. W. (1999). Gene duplication: past, present and future. *Semin Cell Dev Biol*, **10**(5), 541–547.

Holland, P. W., Garcia-Fernandez, J., Williams, N. A., and Sidow, A. (1994). Gene duplications and the origins of vertebrate development. *Dev Suppl*, pages 125–133.

Hughes, A. L. (1998). Phylogenetic tests of the hypothesis of block duplication of homologous genes on human chromosomes 6, 9, and 1. *Mol Biol Evol*, **15**(7), 854–870.

Hughes, A. L. (1999). Phylogenies of developmentally important proteins do not support the hypothesis of two rounds of genome duplication early in vertebrate history. *J Mol Evol*, **48**(5), 565–576.

Hughes, A. L., da Silva, J., and Friedman, R. (2001). Ancient genome duplications did not structure the human Hox-bearing chromosomes. *Genome Res*, **11**(5), 771–780.

Hughey, R. (1993). Massively parallel biosequence analysis. Technical Report UCSC-CRL-93-14.

Hughey, R. (1996). Parallel hardware for sequence comparison and alignment. *CABIOS*, **12**(6), 473–479.

Itoh, T., Takemoto, K., Mori, H., and Gojobori, T. (1999). Evolutionary instability of operon structures disclosed by sequence comparisons of complete microbial genomes. *Mol Biol Evol*, **16**(3), 332–346.

Jongeneel, V., Junier, T., Iseli, C., Hofmann, K., and Bucher, P. (1997). INSECT and MOLLUSCS – supercomputing on the cheap. *EMBnet News*, **4**(3).

Joo, H. G., Goedegebuure, P. S., Sadanaga, N., Nagoshi, M., von Bernstorff, W., and Eberlein, T. J. (2001). Expression and function of galectin-3, a beta-galactoside-binding protein in activated T lymphocytes. *J Leukoc Biol*, **69**(4), 555–564.

Kanehisa, M. (1998). Grand challenges in bioinformatics. *Bioinformatics*, **14**(4), 309.

Karplus, K. and Hu, B. (2001). Evaluation of protein multiple alignments by SAM-T99 using the balibase multiple alignment test set. *Bioinformatics*, **17**(8), 713–720.

Kasahara, M., Nakaya, J., Satta, Y., and Takahata, N. (1997). Chromosomal duplication and the emergence of the adaptive immune system. *Trends in Genetics*, **13**(3), 90–92.

Kent, W. J. and Haussler, D. (2001). Assembly of the working draft of the human genome with GigAssembler. *Genome Res*, **11**(9), 1541–1548.

Kojima, S., Itoh, Y., Matsumoto, S., Masuho, Y., and Seiki, M. (2000). Membrane-type 6 matrix metalloproteinase (MT6-MMP, MMP-25) is the second glycosyl-phosphatidyl inositol (GPI)-anchored MMP. *FEBS Lett*, **480**(2-3), 142–146.

Korenberg, J. R., Chen, X. N., Schipper, R., Sun, Z., Gonsky, R., Gerwehr, S., Carpenter, N., Daumer, C., Dignan, P., Disteche, C., and a. l. .. et (1994). Down syndrome phenotypes: the consequences of chromosomal imbalance. *Proc Natl Acad Sci U S A*, **91**(11), 4997–5001.

Krogh, A., Brown, M., Mian, I. S., Sjolander, K., and Haussler, D. (1994). Hidden Markov models in computational biology. applications to protein modeling. *J Mol Biol*, **235**(5), 1501–1531.

Ku, H. M., Vision, T., Liu, J., and Tanksley, S. D. (2000). Comparing sequenced segments of the tomato and *Arabidopsis* genomes: large-scale duplication followed by selective gene loss creates a network of synteny. *Proc Natl Acad Sci U S A*, **97**(16), 9121–9126.

Kung, H. and Leiserson, C. (1980). *Introduction to VLSI Systems*. Addison-Wesley.

Kuroda, M., Ohta, T., Uchiyama, I., Baba, T., Yuzawa, H., Kobayashi, I., Cui, L., Oguchi, A., Aoki, K., Nagai, Y., Lian, J., Ito, T., Kanamori, M., Matsumaru, H., Maruyama, A., Murakami, H., Hosoyama, A., Mizutani-Ui, Y., Takahashi,

N. K., Sawano, T., Inoue, R., Kaito, C., Sekimizu, K., Hirakawa, H., Kuhara, S., Goto, S., Yabuzaki, J., Kanehisa, M., Yamashita, A., Oshima, K., Furuya, K., Yoshino, C., Shiba, T., Hattori, M., Ogasawara, N., Hayashi, H., and Hiramatsu, K. (2001). Whole genome sequencing of meticillin-resistant *Staphylococcus aureus*. *Lancet*, **357**(9264), 1225–1240.

Lander, E. S., Linton, L. M., Birren, B., Nusbaum, C., Zody, M. C., Baldwin, J., Devon, K., Dewar, K., Doyle, M., FitzHugh, W., Funke, R., Gage, D., Harris, K., Heaford, A., Howland, J., Kann, L., Lehoczky, J., LeVine, R., McEwan, P., McKernan, K., Meldrim, J., Mesirov, J. P., Miranda, C., Morris, W., Naylor, J., Raymond, C., Rosetti, M., Santos, R., Sheridan, A., Sougnez, C., Stange-Thomann, N., Stojanovic, N., Subramanian, A., Wyman, D., Rogers, J., Sulston, J., Ainscough, R., Beck, S., Bentley, D., Burton, J., Clee, C., Carter, N., Coulson, A., Deadman, R., Deloukas, P., Dunham, A., Dunham, I., Durbin, R., French, L., Grafham, D., Gregory, S., Hubbard, T., Humphray, S., Hunt, A., Jones, M., Lloyd, C., McMurray, A., Matthews, L., Mercer, S., Milne, S., Mullikin, J. C., Mungall, A., Plumb, R., Ross, M., Shownkeen, R., Sims, S., Waterston, R. H., Wilson, R. K., Hillier, L. W., McPherson, J. D., Marra, M. A., Mardis, E. R., Fulton, L. A., Chinwalla, A. T., Pepin, K. H., Gish, W. R., Chissoe, S. L., Wendl, M. C., Delehaunty, K. D., Miner, T. L., Delehaunty, A., Kramer, J. B., Cook, L. L., Fulton, R. S., Johnson, D. L., Minx, P. J., Clifton, S. W., Hawkins, T., Branscomb, E., Predki, P., Richardson, P., Wenning, S., Slezak, T., Doggett, N., Cheng, J. F., Olsen, A., Lucas, S., Elkin, C., Uberbacher, E., Frazier, M., Gibbs, R. A., Muzny, D. M., Scherer, S. E., Bouck, J. B., Sodergren, E. J., Worley, K. C., Rives, C. M., Gorrell, J. H., Metzker, M. L., Naylor, S. L., Kucherlapati, R. S., Nelson, D. L., Weinstock, G. M., Sakaki, Y., Fujiyama, A., Hattori, M., Yada, T., Toyoda, A., Itoh, T., Kawagoe, C., Watanabe, H., Totoki, Y., Taylor, T., Weissenbach, J., Heilig, R., Saurin, W., Artiguenave, F., Brottier, P., Bruls, T., Pelletier, E., Robert, C., Wincker, P., Smith, D. R., Doucette-Stamm, L., Rubenfield, M., Weinstock, K., Lee, H. M., Dubois, J., Rosenthal, A., Platzer, M., Nyakatura, G., Taudien, S., Rump, A., Yang, H., Yu, J., Wang, J., Huang, G., Gu, J., Hood, L., Rowen, L., Madan, A., Qin, S., Davis, R. W., Federspiel, N. A.,

Abola, A. P., Proctor, M. J., Myers, R. M., Schmutz, J., Dickson, M., Grimwood, J., Cox, D. R., Olson, M. V., Kaul, R., Raymond, C., Shimizu, N., Kawasaki, K., Minoshima, S., Evans, G. A., Athanasiou, M., Schultz, R., Roe, B. A., Chen, F., Pan, H., Ramser, J., Lehrach, H., Reinhardt, R., McCombie, W. R., de la Bastide, M., Dedhia, N., Blocker, H., Hornischer, K., Nordsiek, G., Agarwala, R., Aravind, L., Bailey, J. A., Bateman, A., Batzoglou, S., Birney, E., Bork, P., Brown, D. G., Burge, C. B., Cerutti, L., Chen, H. C., Church, D., Clamp, M., Copley, R. R., Doerks, T., Eddy, S. R., Eichler, E. E., Furey, T. S., Galagan, J., Gilbert, J. G., Harmon, C., Hayashizaki, Y., Haussler, D., Hermjakob, H., Hokamp, K., Jang, W., Johnson, L. S., Jones, T. A., Kasif, S., Kaspryzk, A., Kennedy, S., Kent, W. J., Kitts, P., Koonin, E. V., Korf, I., Kulp, D., Lancet, D., Lowe, T. M., McLysaght, A., Mikkelsen, T., Moran, J. V., Mulder, N., Pollara, V. J., Ponting, C. P., Schuler, G., Schultz, J., Slater, G., Smit, A. F., Stupka, E., Szustakowski, J., Thierry-Mieg, D., Thierry-Mieg, J., Wagner, L., Wallis, J., Wheeler, R., Williams, A., Wolf, Y. I., Wolfe, K. H., Yang, S. P., Yeh, R. F., Collins, F., Guyer, M. S., Peterson, J., Felsenfeld, A., Wetterstrand, K. A., Patrinos, A., Morgan, M. J., and Szustakowki, J. (2001). Initial sequencing and analysis of the human genome. *Nature*, **409**(6822), 860–921.

Lang, A. J., Mirski, S. E., Cummings, H. J., Yu, Q., Gerlach, J. H., and Cole, S. P. (1998). Structural organization of the human TOP2A and TOP2B genes. *Gene*, **221**(2), 255–266.

Larry Wall, Tom Christiansen, R. S. (1996). *Programming Perl.* O'Reilly & Associates, Inc., Sebastopol, CA.

Lavenier, D. (1996). Dedicated hardware for biological sequence comparison. *Journal of Universal Computer Science*, **2**(2), 77–86.

Lewin, B. (1983). *Genes II.* John Wiley & Sons, Inc, New York.

Li, W.-H. (1997). *Molecular Evolution.* Sinauer Associates, Inc., Sunderland.

Lingenfelter, P. A., Delbridge, M. L., Thomas, S., Hoekstra, H. E., Mitchell, M. J.,

Graves, J. A., and Disteche, C. M. (2001). Expression and conservation of processed copies of the RBMX gene. *Mamm Genome*, **12**(7), 538–545.

Llorente, B., Malpertuy, A., Neuveglise, C., de Montigny, J., Aigle, M., Artiguenave, F., Blandin, G., Bolotin-Fukuhara, M., Bon, E., Brottier, P., Casaregola, S., Durrens, P., Gaillardin, C., Lepingle, A., Ozier-Kalogeropoulos, O., Potier, S., Saurin, W., Tekaia, F., Toffano-Nioche, C., Wesolowski-Louvel, M., Wincker, P., Weissenbach, J., Souciet, J., and Dujon, B. (2000). Genomic exploration of the hemiascomycetous yeasts: 18. Comparative analysis of chromosome maps and synteny with *Saccharomyces cerevisiæ*. *FEBS Lett*, **487**(1), 101–112.

Luscombe, N. M., Greenbaum, D., and Gerstein, M. (2001). What is bioinformatics? a proposed definition and overview of the field. *Methods Inf Med*, **40**(4), 346–358.

Lyall, A. (1996). Bioinformatics in the pharmaceutical industry. *Trends Biotechnol*, **14**(8), 308–312.

Lynch, M. and Conery, J. S. (2000). The evolutionary fate and consequences of duplicate genes. *Science*, **290**(5494), 1151–1155.

Lynch, M. and Force, A. G. (2000). The origin of interspecific genomic incompatibility via gene duplication. *Am. Nat.*, **156**, 590–605.

Makalowski, W. (2001). Are we polyploids? A brief history of one hypothesis. *Genome Res*, **11**(5), 667–670.

Martin, A. P. (1999). Increasing genomic complexity by gene duplication and the origin of vertebrates. *American Naturalist*, **154**, 111–128.

McCarrey, J. R. and Thomas, K. (1987). Human testis-specific PGK gene lacks introns and possesses characteristics of a processed gene. *Nature*, **326**(6112), 501–505.

McEntyre, J. (1998). Linking up with Entrez. *Trends Genet*, **14**(1), 39–40.

McEntyre, J. and Lipman, D. (2001). Pubmed: bridging the information gap. *CMAJ*, **164**(9), 1317–1319.

McLysaght, A. (2001). *Evolution of vertebrate genome organisation.* Ph.D. thesis, University of Dublin.

McLysaght, A., Hokamp, K., and Wolfe, K. H. (2001). Genomic duplication during early chordate evolution. *Nature Genetics*, **submitted**.

Ming, R., Liu, S. C., Lin, Y. R., da Silva, J., Wilson, W., Braga, D., van Deynze, A., Wenslaff, T. F., Wu, K. K., Moore, P. H., Burnquist, W., Sorrells, M. E., Irvine, J. E., and Paterson, A. H. (1998). Detailed alignment of *saccharum* and *sorghum* chromosomes: comparative organization of closely related diploid and polyploid genomes. *Genetics*, **150**(4), 1663–1682.

Mira, A., Ochman, H., and Moran, N. A. (2001). Deletional bias and the evolution of bacterial genomes. *Trends Genet*, **17**(10), 589–596.

Muller, H. J. (1925). Why polyploidy is rarer in animals than in plants. *American Naturalist*, **9**, 346–353.

Muller, S., O'Brien, P. C., Ferguson-Smith, M. A., and Wienberg, J. (1998). Cross-species colour segmenting: a novel tool in human karyotype analysis. *Cytometry*, **33**(4), 445–452.

Needleman, S. B. and Wunsch, C. D. (1970). A general method applicable to the search of similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, **48**, 443–453.

Nei, M., Xu, P., and Glazko, G. (2001). Estimation of divergence times from multiprotein sequences for a few mammalian species and several distantly related organisms. *Proc Natl Acad Sci U S A*, **98**(5), 2497–2502.

Notredame, C., Higgins, D. G., and Heringa, J. (2000). T-coffee: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol*, **302**(1), 205–217.

Ochman, H., Lawrence, J. G., and Groisman, E. A. (2000). Lateral gene transfer and the nature of bacterial innovation. *Nature*, **405**(6784), 299–304.

Ohno, S. (1970). *Evolution by Gene Duplication.* George Allen and Unwin, London.

Ohno, S. (1997). The reason for as well as the consequence of the Cambrian explosion in animal evolution. *J Mol Evol*, **44 Suppl 1**, S23–S27.

Ouzounis, C. (2000). Two or three myths about bioinformatics. *Bioinformatics*, **16**(3), 187–189.

Pearson, W. R. (1995). Comparison of methods for searching protein sequence databases. *Protein Sci*, **4**(6), 1145–1160.

Pearson, W. R. (2000). Flexible sequence similarity searching with the FASTA3 program package. *Methods Mol Biol*, **132**, 185–219.

Pearson, W. R. and Lipman, D. J. (1988). Improved tools for biological sequence comparison. *Proc Natl Acad Sci U S A*, **85**(8), 2444–2448.

Pebusque, M. J., Coulier, F., Birnbaum, D., and Pontarotti, P. (1998). Ancient large-scale genome duplications: phylogenetic and linkage analyses shed light on chordate genome evolution. *Mol Biol Evol*, **15**(9), 1145–1159.

Pruitt, K. D., Katz, K. S., Sicotte, H., and Maglott, D. R. (2000). Introducing RefSeq and LocusLink: curated human genome resources at the NCBI. *Trends Genet*, **16**(1), 44–47.

Reeck, G. R., de Haen, C., Teller, D. C., Doolittle, R. F., Fitch, W. M., Dickerson, R. E., Chambon, P., McLachlan, A. D., Margoliash, E., Jukes, T. H., and et al. (1987). "Homology" in proteins and nucleic acids: a terminology muddle and a way out of it. *Cell*, **50**(5), 667.

Ridge, D., Becker, D., Merkey, P., and Sterling, T. (1997). Beowulf: Harnessing the power of parallelism in a pile-of-pcs. In *Proceedings, IEEE Aerospace*.

Riechmann, J. L., Heard, J., Martin, G., Reuber, L., Jiang, C., Keddie, J., Adam, L., Pineda, O., Ratcliffe, O. J., Samaha, R. R., Creelman, R., Pilgrim, M., Broun, P., Zhang, J. Z., Ghandehari, D., Sherman, B. K., and Yu, G. (2000). *Arabidopsis* transcription factors: genome-wide comparative analysis among eukaryotes. *Science*, **290**(5499), 2105–2110.

Rognes, T. and Seeberg, E. (2000). Six-fold speed-up of Smith-Waterman sequence database searches using parallel processing on common microprocessors. *Bioinformatics*, **16**(8), 699–706.

Rubin, G. M., Yandell, M. D., Wortman, J. R., Miklos, G. L. G., Nelson, C. R., Hariharan, I. K., Fortini, M. E., Li, P. W., Apweiler, R., Fleischmann, W., Cherry, J. M., Henikoff, S., Skupski, M. P., Misra, S., Ashburner, M., Birney, E., Boguski, M. S., Brody, T., Brokstein, P., Celniker, S. E., Chervitz, S. A., Coates, D., Cravchik, A., Gabrielian, A., Galle, R. F., Gelbart, W. M., George, R. A., Goldstein, L. S., Gong, F., Guan, P., Harris, N. L., Hay, B. A., Hoskins, R. A., Li, J., Li, Z., Hynes, R. O., Jones, S. J., Kuehl, P. M., Lemaitre, B., Littleton, J. T., Morrison, D. K., Mungall, C., O'Farrell, P. H., Pickeral, O. K., Shue, C., Vosshall, L. B., Zhang, J., Zhao, Q., Zheng, X. H., and Lewis, S. (2000). Comparative genomics of the eukaryotes. *Science*, **287**(5461), 2204–2215.

Ruddle, F. H., Bentley, K. L., Murtha, M. T., and Risch, N. (1994). Gene loss and gain in the evolution of the vertebrates. *Dev Suppl*, pages 155–161.

Rutherford, K., Parkhill, J., Crook, J., Horsnell, T., Rice, P., Rajandream, M. A., and Barrell, B. (2000). Artemis: sequence visualization and annotation. *Bioinformatics*, **16**(10), 944–945.

Sander, C. (2001). Bioinformatics - challenges in 2001. *Bioinformatics*, **17**(1), 1–2.

Sansom, C. E. and Smith, C. A. (2000). Computer applications in biomolecular sciences. Part 2: bioinformatics and genome projects. *Biochem. Educ*, **28**(3), 127–131.

Sato, H., Tanaka, M., Takino, T., Inoue, M., and Seiki, M. (1997). Assignment of the human genes for membrane-type-1, -2, and -3 matrix metalloproteinases (MMP14, MMP15, and MMP16) to 14q12.2, 16q12.2-q21, and 8q21, respectively, by *in situ* hybridization. *Genomics*, **39**(3), 412–413.

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

Schuler, G. D. (1997). Pieces of the puzzle: expressed sequence tags and the catalog of human genes. *J Mol Med*, **75**(10), 694–698.

Seoighe, C. and Wolfe, K. H. (1998). Extent of genomic rearrangement after genome duplication in yeast. *Proc Natl Acad Sci U S A*, **95**(8), 4447–4452.

Seoighe, C. and Wolfe, K. H. (1999). Updated map of duplicated regions in the yeast genome. *Gene*, **238**(1), 253–261.

Service, R. F. (2000). Proteomics. Can Celera do it again? *Science*, **287**(5461), 2136–2138.

Sharman, A. C. and Holland, P. W. H. (1996). Conservation, duplication and divergence of developmental genes during chordate evolution. *Neth. J. Zool.*, **46**, 47–67.

Shpaer, E. G., Robinson, M., Yee, D., Candlin, J. D., Mines, R., and Hunkapiller, T. (1996). Sensitivity and selectivity in protein similarity searches: a comparison of Smith-Waterman in hardware to BLAST and FASTA. *Genomics*, **38**(2), 179–191.

Sippl, M. J., Lackner, P., Domingues, F. S., and Koppensteiner, W. A. (1999). An attempt to analyse progress in fold recognition from CASP1 to CASP3. *Proteins*, **37**(S3), 226–230.

Skrabanek, L. and Wolfe, K. H. (1998). Eukaryote genome duplication - where's the evidence? *Curr Opin Genet Dev*, **8**(6), 694–700.

Smith, N. G., Knight, R., and Hurst, L. D. (1999). Vertebrate genome evolution: a slow shuffle or a big bang? *Bioessays*, **21**(8), 697–703.

Smith, T. F. and Waterman, M. S. (1981). Identification of common molecular subsequences. *J Mol Biol*, **147**(1), 195–197.

Snir, M., Otto, S. W., Huss-Lederman, S., Walker, D. W., and Dongarra, J. (1996). *MPI: the complete reference.* MIT Press, Cambridge, MA, USA.

Spring, J. (1997). Vertebrate evolution by interspecific hybridisation – are we polyploid? *FEBS Letters*, **400**(1), 2–8.

Sterling, T., Savarese, D., Becker, D. J., Dorband, J. E., Ranawake, U. A., and Packer, C. V. (1995). Beowulf: A parallel workstation for scientific computation. In *Proceedings of the 24th International Conference on Parallel Processing*, pages I:11–14, Oconomowoc, WI.

Stevens, R., Goble, C., Baker, P., and Brass, A. (2001). A classification of tasks in bioinformatics. *Bioinformatics*, **17**(2), 180–188.

Stoesser, G., Baker, W., van den Broek, A., Camon, E., Garcia-Pastor, M., Kanz, C., Kulikova, T., Lombard, V., Lopez, R., Parkinson, H., Redaschi, N., Sterk, P., Stoehr, P., and Tuli, M. A. (2001). The EMBL nucleotide sequence database. *Nucleic Acids Res*, **29**(1), 17–21.

Strausberg, R. L. and Austin, M. J. (1999). Functional genomics: technological challenges and opportunities. *Physiol Genomics*, **1**(1), 25–32.

Strohmaier, E., Dongarra, J., Meuer, H. W., and Simon, H. D. (1999). The marketplace of high-performance computing. *Parallel Computing*, **25**(13-14), 1517–1544.

Sunderam, V. S. (1990). PVM: a framework for parallel distributed computing. *Concurrency, Practice and Experience*, **2**(4), 315–340.

Tatusov, R. L., Natale, D. A., Garkavtsev, I. V., Tatusova, T. A., Shankavaram, U. T., Rao, B. S., Kiryutin, B., Galperin, M. Y., Fedorova, N. D., and Koonin, E. V. (2001). The COG database: new developments in phylogenetic classification of proteins from complete genomes. *Nucleic Acids Res*, **29**(1), 22–28.

Tautz, D. (1998). Evolutionary biology. Debatable homologies. *Nature*, **395**(6697), 17,19.

Tekaia, F., Lazcano, A., and Dujon, B. (1999). The genomic tree as revealed from whole proteome comparisons. *Genome Res*, **9**(6), 550–557.

The *Arabidopsis* Genome Initiative (2000). Analysis of the genome sequence of the flowering plant *Arabidopsis thaliana*. *Nature*, **408**(6814), 796–815.

Thornton, J. W. (2001). Evolution of vertebrate steroid receptors from an ancestral estrogen receptor by ligand exploitation and serial genome expansions. *Proc Natl Acad Sci U S A*, **98**(10), 5671–5676.

Tomsig, J. L. and Creutz, C. E. (2000). Biochemical characterization of copine: a ubiquitous Ca2+-dependent, phospholipid-binding protein. *Biochemistry*, **39**(51), 16163–16175.

Trifonov, E. N. (2000). Earliest pages of bioinformatics. *Bioinformatics*, **16**(1), 5–9.

Vanin, E. F. (1985). Processed pseudogenes: characteristics and evolution. *Annu Rev Genet*, **19**, 253–272.

Venter, J. C., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J., Sutton, G. G., Smith, H. O., Yandell, M., Evans, C. A., Holt, R. A., Gocayne, J. D., Amanatides, P., Ballew, R. M., Huson, D. H., Wortman, J. R., Zhang, Q., Kodira, C. D., Zheng, X. H., Chen, L., Skupski, M., Subramanian, G., Thomas, P. D., Zhang, J., Miklos, G. L. G., Nelson, C., Broder, S., Clark, A. G., Nadeau, J., McKusick, V. A., Zinder, N., Levine, A. J., Roberts, R. J., Simon, M., Slayman, C., Hunkapiller, M., Bolanos, R., Delcher, A., Dew, I., Fasulo, D., Flanigan, M., Florea, L., Halpern, A., Hannenhalli, S., Kravitz, S., Levy, S., Mobarry, C., Reinert, K., Remington, K., Abu-Threideh, J., Beasley, E., Biddick, K., Bonazzi, V., Brandon, R., Cargill, M., Chandramouliswaran, I., Charlab, R., Chaturvedi, K., Deng, Z., Francesco, V. D., Dunn, P., Eilbeck, K., Evangelista, C., Gabrielian, A. E., Gan, W., Ge, W., Gong, F., Gu, Z., Guan, P., Heiman, T. J., Higgins, M. E., Ji, R. R., Ke, Z., Ketchum, K. A., Lai, Z., Lei, Y., Li, Z., Li, J., Liang, Y., Lin, X., Lu, F., Merkulov, G. V., Milshina, N., Moore, H. M., Naik, A. K., Narayan, V. A., Neelam, B., Nusskern, D., Rusch, D. B., Salzberg, S., Shao, W., Shue, B., Sun, J., Wang, Z., Wang, A., Wang, X., Wang, J., Wei, M., Wides, R., Xiao, C., Yan, C., Yao, A., Ye, J., Zhan, M., Zhang, W., Zhang, H., Zhao, Q., Zheng, L., Zhong, F., Zhong, W., Zhu, S., Zhao, S., Gilbert, D., Baumhueter, S., Spier, G., Carter, C., Cravchik, A., Woodage, T., Ali, F., An, H., Awe, A., Baldwin, D., Baden, H., Barnstead, M., Barrow, I., Beeson, K., Busam, D., Carver, A., Center, A., Cheng, M. L., Curry, L., Danaher, S., Davenport, L., Desilets, R., Dietz, S., Dodson, K.,

Doup, L., Ferriera, S., Garg, N., Gluecksmann, A., Hart, B., Haynes, J., Haynes, C., Heiner, C., Hladun, S., Hostin, D., Houck, J., Howland, T., Ibegwam, C., Johnson, J., Kalush, F., Kline, L., Koduru, S., Love, A., Mann, F., May, D., McCawley, S., McIntosh, T., McMullen, I., Moy, M., Moy, L., Murphy, B., Nelson, K., Pfannkoch, C., Pratts, E., Puri, V., Qureshi, H., Reardon, M., Rodriguez, R., Rogers, Y. H., Romblad, D., Ruhfel, B., Scott, R., Sitter, C., Smallwood, M., Stewart, E., Strong, R., Suh, E., Thomas, R., Tint, N. N., Tse, S., Vech, C., Wang, G., Wetter, J., Williams, S., Williams, M., Windsor, S., Winn-Deen, E., Wolfe, K., Zaveri, J., Zaveri, K., Abril, J. F., Guigo, R., Campbell, M. J., Sjolander, K. V., Karlak, B., Kejariwal, A., Mi, H., Lazareva, B., Hatton, T., Narechania, A., Diemer, K., Muruganujan, A., Guo, N., Sato, S., Bafna, V., Istrail, S., Lippert, R., Schwartz, R., Walenz, B., Yooseph, S., Allen, D., Basu, A., Baxendale, J., Blick, L., Caminha, M., Carnes-Stine, J., Caulk, P., Chiang, Y. H., Coyne, M., Dahlke, C., Mays, A., Dombroski, M., Donnelly, M., Ely, D., Esparham, S., Fosler, C., Gire, H., Glanowski, S., Glasser, K., Glodek, A., Gorokhov, M., Graham, K., Gropman, B., Harris, M., Heil, J., Henderson, S., Hoover, J., Jennings, D., Jordan, C., Jordan, J., Kasha, J., Kagan, L., Kraft, C., Levitsky, A., Lewis, M., Liu, X., Lopez, J., Ma, D., Majoros, W., McDaniel, J., Murphy, S., Newman, M., Nguyen, T., Nguyen, N., Nodell, M., Pan, S., Peck, J., Peterson, M., Rowe, W., Sanders, R., Scott, J., Simpson, M., Smith, T., Sprague, A., Stockwell, T., Turner, R., Venter, E., Wang, M., Wen, M., Wu, D., Wu, M., Xia, A., Zandieh, A., and Zhu, X. (2001). The sequence of the human genome. *Science*, **291**(5507), 1304–1351.

Vision, T. J., Brown, D. G., and Tanksley, S. D. (2000). The origins of genomic duplications in *Arabidopsis*. *Science*, **290**(5499), 2114–2117.

Wada, A. (2000). Bioinformatics–the necessity of the quest for 'first principles' in life. *Bioinformatics*, **16**(8), 663–664.

Wang, Y. and Gu, X. (2000). Evolutionary patterns of gene families generated in the early stage of vertebrates. *J Mol Evol*, **51**(1), 88–96.

Warren, M., Becker, D., Goda, M., Salmon, J., and Sterling, T. (1997). Parallel

supercomputing with commodity components. In *Proceedings of PDPTA '97*, pages 1372–1381.

Wheeler, D., Hope, R., Cooper, S. B., Dolman, G., Webb, G. C., Bottema, C. D., Gooley, A. A., Goodman, M., and Holland, R. A. (2001). An orphaned mammalian beta-globin gene of ancient evolutionary origin. *Proc Natl Acad Sci U S A*, **98**(3), 1101–1106.

Wolf, Y. I., Rogozin, I. B., Kondrashov, A. S., and Koonin, E. V. (2001). Genome alignment, evolution of prokaryotic genome organization, and prediction of gene function using genomic context. *Genome Res*, **11**(3), 356–372.

Wolfe, K. (2000). Robustness–it's not where you think it is. *Nat Genet*, **25**(1), 3–4.

Wolfe, K. H. (2001). Yesterday's polyploids and the mystery of diploidization. *Nat Rev Genet*, **2**(5), 333–341.

Wolfe, K. H. and Shields, D. C. (1997). Molecular evidence for an ancient duplication of the entire yeast genome. *Nature*, **387**(6634), 708–713.

Wootton, J. C. and Federhen, S. (1996). Analysis of compositionally biased regions in sequence databases. *Methods Enzymol*, **266**, 554–571.

Wright, F. A., Lemon, W. J., Zhao, W. D., Sears, R., Zhuo, D., Wang, J. P., Yang, H. Y., Baer, T., Stredney, D., Spitzner, J., Stutz, A., Krahe, R., and Yuan, B. (2001). A draft annotation and overview of the human genome. *Genome Biol*, **2**(7), 1–18.

Xu, W., Edmondson, D. G., Evrard, Y. A., Wakamiya, M., Behringer, R. R., and Roth, S. Y. (2000). Loss of Gcn5l2 leads to increased apoptosis and mesodermal defects during mouse development. *Nat Genet*, **26**(2), 229–232.

Yanai, I., Camacho, C. J., and DeLisi, C. (2000). Predictions of gene family distributions in microbial genomes: evolution by gene duplication and modification. *Phys Rev Lett*, **85**(12), 2641–2644.

Yap, T. K., Munson, P. J., Frieder, O., and Martino, R. L. (1995). Parallel multiple sequence alignment using speculative computation. In *Proceedings of the 24th International Conference on Parallel Processing*, pages 60–67, Oconomowoc, WI.