

Methods for Identifying and Mapping Recent Segmental and Gene Duplications in Eukaryotic Genomes

Razi Khaja, Jeffrey R. MacDonald, Junjun Zhang,
and Stephen W. Scherer

Summary

The aim of this chapter is to provide instruction for analyzing and mapping recent segmental and gene duplications in eukaryotic genomes. We describe a bioinformatics-based approach utilizing computational tools to manage eukaryotic genome sequences to characterize and understand the evolutionary fates and trajectories of duplicated genes. An introduction to bioinformatics tools and programs such as BLAST, Perl, BioPerl, and the GFF specification provides the necessary background to complete this analysis for any eukaryotic genome of interest.

Key Words: Bioinformatics; BLAST/MegaBLAST; gene duplication; gene ontology; genome assembly; genomic disorder; GFF (Generic Feature Format); homology; neofunctionalization; paralogous; Perl/BioPerl; pseudogene; RefSeq; RepeatMasker; segmental duplication; sequence alignments; subfunctionalization.

1. Introduction

With the completion of the human genome sequence and the increasing availability of whole genome shotgun sequences (WGS) for numerous other eukaryotic species, we are poised to begin to understand the complexity and dynamic nature of chromosomes. Segmental duplications are nearly identical segments of DNA at two or more sites in a genome; for human they comprise about 3.5 to 5% of the total DNA content (1,2). Segmental duplications also account for 1.2 to 2% of the mouse genome (3,4) and approx 3% of the rat genome (5). Segmental duplications (also called low copy repeats [LCRs]) can be predisposition sites for increased opportunity of nonallelic homologous recombination leading to deletion, inversion, or duplication of large segments of DNA (6).

From: *Methods in Molecular Biology*, vol. 338: *Gene Mapping, Discovery, and Expression: Methods and Protocols*

Edited by: M. Bina © Humana Press Inc., Totowa, NJ

These structural alterations may lead to the gain or loss of dosage-sensitive genetic material and may result in a spectrum of diseases defined as genomic disorders (7–9).

The presence of segmental duplications is a common feature of many mammalian genomes, and their involvement in chromosome evolution and natural variation is an area of active investigation (10–12). Duplication of large segments of DNA can generate duplicate genes in whole (13), or in part (14), and may lead to an expanding repertoire of similar gene products. The identification of recent segmental duplication therefore gives us the ability to map the origin and fate of duplicate genes, which are a driving force in species evolution (see **Note 1**).

Here we define recent segmental duplications as paralogous regions of a genome having a length greater than 5000 nucleotides (nt) and having greater than 90% DNA sequence identity. We present a computational protocol for identifying and mapping recent segmental and gene duplications in eukaryotic genomes. The major procedures involved in identifying recent segmental and gene duplications include comparing genomic sequences using BLAST (15), parsing and filtering BLAST alignments, and mapping genes to segmental duplications to identify gene duplicates. We note that much of our methodologies have arisen in an ongoing initiative to map segmental duplications accurately in the human (2), chimpanzee, mouse (3), and other mammalian genomes as displayed at publicly available websites (<http://projects.tcag.ca/humandup> and <http://projects.tcag.ca/xenodup>).

2. Materials

1. A modest-sized cluster-computer or super-computer with 4 GB of RAM per CPU running any variant of a UNIX or Linux operating system.
2. Internet connection, ftp utilities (e.g., ftp, ncftp, wget).
3. Archiving utilities (e.g., unzip).
4. An assembled genome sequence of a eukaryotic organism that is lower case masked for repetitive elements.
5. The BLAST suite of programs (particularly formatdb and MegaBLAST).
6. Perl, BioPerl.
7. Approximately 5 to 20 GB of disk space to store sequence data, blast databases, alignment data, and parsed output.

3. Methods

The methods described below outline: (1) the prerequisites and assumptions required to perform this analysis, (2) where to obtain genome assemblies of eukaryotic genomes, (3) the process for installing the BLAST suite of programs, and (4) the procedure for creating BLAST databases. To identify segmental

duplications in eukaryotic genomes, the methods summarize: (5) the procedure for performing sequence alignments of all possible pairs of chromosomes using MegaBLAST, (6) how to convert MegaBLAST alignments into Generic Feature Format (GFF) format, and (7) the criteria for filtering GFF records and (8) chain alignments together. Furthermore, we describe how to identify gene duplicates by (9) mapping RefSeq genes to segmental duplications and (10) using the Gene Ontology to characterize gene duplicates by function.

3.1. Prerequisites/Assumptions

To perform segmental duplication analysis of eukaryotic genomes, the reader needs access to a modest-sized cluster-computer or super-computer with a minimum of 4 GB of RAM available to each CPU (*see Note 2*) running any variant of a UNIX or Linux operating system (*see Note 3*). Competency in using UNIX command line utilities and programming in Perl is also a necessity (*see Note 4*). It is also a prerequisite that the BioPerl package (*see Note 5*) be available in the computing environment. Furthermore, the reader should be capable of using BioPerl to convert MegaBLAST alignment files into GFF records and should be familiar with the GFF version 3 specification (*see Note 6*).

3.2. Download Genome of Interest

This protocol requires that the genome sequence being targeted for the identification of segmental and gene duplications be assembled and masked for repetitive elements.

Although this protocol is applicable to all eukaryotic genomes (*see Note 7*), the mouse genome will be used as our example. The May 2004 mouse genome assembly (referred to as mm5 by UCSC or Build 33 by NCBI) can be downloaded from UCSC (<http://genome.ucsc.edu>) as a zip file by executing the following command:

```
% wget http://hgdownload.cse.ucsc.edu/goldenPath/mm5/bigZips/chromFa.zip
```

This zip file contains the mouse genome assembly with one FASTA file for each chromosome. Repetitive elements within each chromosome sequence have been identified with RepeatMasker (<http://www.repeatmasker.org>) and are represented in lower case letters; nonrepeating DNA sequences are shown in upper case letters. Once the genome has been downloaded, the zip file is uncompressed by executing the following command:

```
% unzip chromFa.zip
```

Uncompressing this file will extract one FASTA file for each chromosome sequence. For the mouse genome, this should extract files: chr1.fa to chr19.fa, chrX.fa, chrY.fa, and chrM.fa (mitochondrial dna), as well as chr1_random.fa

to chr19_random.fa, chrX_random.fa, chrY_random.fa, and chrUn_random.fa (see **Note 8**).

3.3. Download and Install the BLAST Suite of Programs

To perform sequence alignments for identification of segmental duplications in the genome, download and install the BLAST suite of programs on your computing environment. The BLAST suite of programs is available from the NCBI as precompiled binary distributions or as source code. The precompiled binaries are available from <ftp://ftp.ncbi.nlm.nih.gov/blast/executables/LATEST/>. These are compiled for many operating systems and hardware architectures (see **Note 9**). Installation is a simple matter of downloading and then uncompressing the distribution for your computing environment. Documentation supplied with the BLAST suite of programs describes command line options for each of the utilities. In this protocol, the formatdb and MegaBLAST (**16**) command line tools are used to identify segmental duplications in the genome. Formatdb is used to create BLAST databases, and MegaBLAST is used to perform sequence alignments.

3.4. Create BLAST Databases, One for Each Chromosome

Once the genome has been downloaded and the BLAST suite has been installed, create BLAST databases for each of the chromosome FASTA files using the formatdb command line utility. The formatdb command line utility must be used to format a FASTA file such as chr7.fa into a BLAST database before it can be searched by MegaBLAST. The following command is an example of using formatdb to create a BLAST database:

```
% formatdb -i chr7.fa -p F
```

Executing this command will create the files: chr7.fa.nhr, chr7.fa.nin, and chr7.fa.nsq, which collectively represent the BLAST database for mouse chromosome 7. This database will be searched by MegaBLAST in order to produce sequence alignments for the purpose of identifying segmental duplications in the genome. BLAST databases must be created iteratively for every FASTA file for each chromosome sequence in the genome, including the pseudo chromosomes (see **Note 8**).

In the example above, “-i chr7.fa” specifies the name of the input file, and “-p F” specifies that the sequence contained within the file is nucleotide. Below is a detailed description of the command line options used:

formatdb 2.2.10 arguments:

- i Input file(s) for formatting (this parameter must be set)
[File In]

- p Type of file
 - T - protein
 - F - nucleotide [T/F] Optional
 - default = T

A full description of this command and its options is included with the documentation supplied with the BLAST suite and is also available at the NCBI website (<http://www.ncbi.nih.gov/BLAST/docs/formatdb.html>).

3.5. Perform Sequence Alignments of All Possible Pairs of Chromosomes Using MegaBLAST

The MegaBLAST program is used to perform sequence alignments because it was designed to identify long alignments efficiently between similar sequences. Since we have defined recent segmental duplications as long stretches of DNA (>5000 nt) having greater than 90% sequence identity, MegaBLAST is ideal at identifying these paralogous regions of the genome. After creating the BLAST databases for each chromosome, MegaBLAST is used to perform sequence alignments between all possible pairs of chromosomes. In other words, each FASTA file is compared with each of the BLAST databases (*see Note 10*).

The following command is an example of using MegaBLAST to find sequence alignments between mouse chromosome 7 and mouse chromosome 3.

```
% megablast -d chr7.fa -i chr3.fa -D 2 -F 'm' -U T -o chr7.3.blast
```

In the example above, the “-d chr7.fa” option specifies that MegaBLAST use the mouse chromosome 7 BLAST database as the subject of this comparison and the “-i chr3.fa” option specifies mouse chromosome 3 as the query sequence. Sequence alignments are stored in the chr7.3.blast output file as specified by the option “-o chr7.3.blast” and the format of output generated is “traditional BLAST output” as specified by the “-D 2” option. Furthermore, “-U T” specifies that lower case letters in the query sequence should be recognized as a repetitive element. The “-F ‘m’” option denotes that the MegaBLAST algorithm should not find word matches in the repetitive regions of the query sequence but should allow for extension of sequence alignments through these regions.

Below is a detailed description of the command line options that are required to perform sequence alignments using MegaBLAST to identify segmental duplications in a genome:

megablast 2.2.10 arguments:

- d Database [String]
 - default = nr
- i Query File [File In]
- D Type of output:

- 0 - alignment endpoints and score
- 1 - all ungapped segments endpoints
- 2 - traditional BLAST output
- 3 - tab-delimited one-line format [Integer]
- default = 0
- F Filter query sequence [String]
- default = T
- U Use lower case filtering of FASTA sequence [T/F] Optional
- default = F
- o BLAST report Output File [File Out] Optional
- default = stdout

A full description of this command and its options is included with the documentation supplied with the BLAST suite of programs and is also available at the NCBI website (<http://www.ncbi.nih.gov/BLAST/docs/megablast.html>).

Sequence alignments generated by MegaBLAST between a subject database and a query sequence of the same chromosome are used to identify intra-chromosomal segmental duplications (i.e., duplications that occur within the same chromosome). Sequence alignments generated by MegaBLAST between a subject database and a query sequence of different chromosomes are used to identify interchromosomal segmental duplications (i.e., duplications that occur between different chromosomes). Executing MegaBLAST on a subject database and query sequence generates many sequence alignments. Not all of these represent sequences involved in segmental duplications, so further steps are required to convert, filter, and process these alignments based on a variety of criteria. These criteria are described in the sections below.

3.6. Convert MegaBLAST Alignments Into GFF Format

In the previous step, MegaBLAST was used to generate traditional BLAST output for all pairs of chromosomes. Sequence alignments in this format are extremely informative since they visualize detailed information about homologous DNA, showing locations of nucleotide mismatches and small insertions and deletions (**Fig. 1**).

However, programmatically it is difficult to identify duplications from blast results in this format as this output is generated for visual inspection. In order to identify segmental duplications from blast results without loss of information it is necessary to transform traditional BLAST output into a tabular format. The current Generic Feature Format version 3 (GFF3) specification (<http://song.sourceforge.net/gff3.shtml>) is a widely accepted tabular format for describing genes and other features associated with DNA, RNA, and protein sequences. The BioPerl project (<http://www.bioperl.org>) supports the parsing of different output formats, including traditional BLAST output into GFF3.

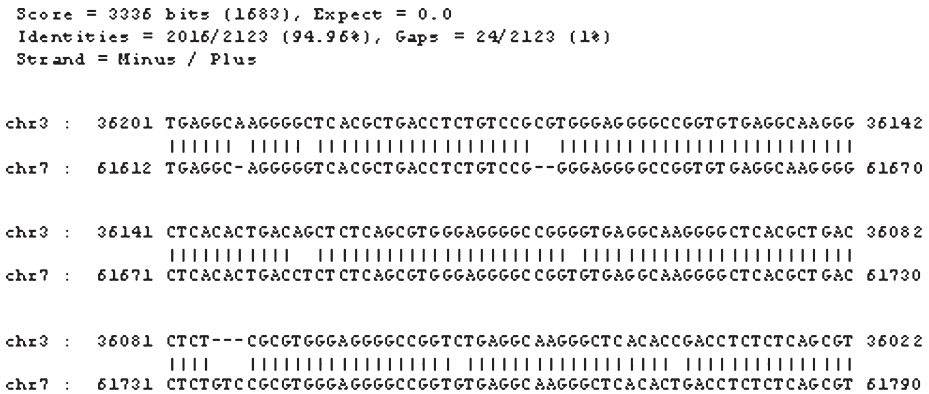


Fig. 1. Traditional BLAST output as generated by MegaBLAST.

Using the Bio::SearchIO module that is part of the BioPerl package, it is required that BLAST alignment files for each pair of chromosomes be converted into GFF3 records. Below is an example of the result of converting the alignment shown in **Fig. 1** as a GFF3 record:

chr7 UCSC_hg17 match 61612 61790 0.0 - . Target=chr3 36022 36201;Gap=M6 I1
M25 I2 M90 D3 M53;percentId=94.96;alnLength=2123;matches=2016;gaps=24;
bitScore=3336;rawScore=1683

To understand how to generate records in GFF3 format, the reader should understand the GFF3 specification. This will enable the user to apply the Bio::SearchIO module to convert BLAST alignment files to generate this output. This format allows storage of all information from the traditional BLAST output including: subject sequence start and stop coordinates, query sequence start and stop coordinates, e-value, strand, percent identity, alignment length, matched nucleotides, gaps, bit score, raw score and detailed alignment information.

3.7. Filter GFF Records Based on Many Criteria

After converting the traditional BLAST alignments into GFF format, some alignments are excluded since not all are components of recent segmental duplications. To identify sequences meeting a stringent categorization of being a “recent segmental duplication,” GFF records are filtered based on the criteria described below.

3.7.1. Filter Sequence Alignments With Less Than 90 Percent Identity

Recent segmental duplications are defined as paralogous sequences that share greater than 90% sequence similarity. Remove GFF records in which the percent identity attribute does not meet this minimum percent identity cutoff. This

filtering criterion is applicable to both inter- and intrachromosomal sequence alignments.

3.7.2. Filter Suboptimal Sequence Alignments

Suboptimal sequence alignments occur when one sequence alignment is redundant in the sense that the subject and query elements are completely covered or spanned by another alignment. Remove the GFF record with the smaller span, which is considered a suboptimal alignment. This filtering step is applicable to both inter- and intrachromosomal sequence alignments.

3.7.3. Filter Identical Sequence Alignments

This filtering step is only applicable to intrachromosomal sequence alignments. Exclude self-self matches, whose GFF records have subject sequence coordinates that are identical to the query sequence coordinates.

3.8. Identify Segmental Duplications by Chaining Alignments Together

To define the boundaries of segmental duplications, alignments whose coordinates are monotonically increasing are chained together to form larger contiguous alignments. This compensates for short and fragmented alignments, which have arisen because of insertion or deletion events that have modified paralogous copies of DNA. Since we defined segmental duplications as regions of the genome having length greater than 5000 nt, we need to filter chained alignments that do not meet this minimum length requirement.

1. Sort GFF records by subject and query coordinates.
2. For records of the same subject and query chromosome pair, if adjacent sequence alignments are separated by less than 3000 nt, chain the alignments together
3. Remove chained alignments that are smaller than 5000 bp.

This step concludes the identification of large regions of the genome involved in recent segmental duplications. Large segmental duplications can often contain duplicate genes and/or be implicated in genomic disease and structural rearrangements; hence they have an inherent biological interest. **Subheadings 3.9. and 3.10.** discuss mapping genes to segmental duplications, identifying duplicate gene pairs, and characterizing gene duplications using the Gene Ontology.

3.9. Map RefSeq Genes to the Mouse Genome and to Segmental Duplications

To identify and characterize recent gene duplicates in the mouse genome, you will first need to obtain the most current curated gene data set, map the location of the gene to the genome of interest, and perform a positional colocalization of genes and duplications to detect gene paralogs.

3.9.1. Obtain RefSeq Gene Set and Mapping Location in the Mouse Genome

1. Obtain the mouse gene data set (refGene.txt.gz) from the University of California at Santa Cruz (<http://hgdownload.cse.ucsc.edu/goldenPath/mm5/database/>).
2. Extract the gene mapping information from the above file, and store in GFF3 format. A description of the refGene.txt table format from UCSC can be found at <http://genome.ucsc.edu/goldenPath/gbdDescriptions.html#GenePredictions>.

3.9.2. Identify Recent Gene Duplicates

Identifying recent gene duplications generated via a segmental duplication event can be accomplished by localizing the genes that lie within the boundaries of the duplications detected and determining the paralogous gene pair in the corresponding duplicon. The genes may be duplicated in whole or part along with the surrounding genomic DNA.

1. Identify the genes that reside completely within the defined boundary of the duplications (whole gene duplication). Compare the transcriptional start and end coordinates stored in the GFF3 file and identify those genes that fall completely within the coordinates of the duplication.
2. Identify the genes that lie partially within the defined boundary of the duplication (partial gene duplication). Compare the transcriptional start and end coordinates stored in the GFF3 file and identify those genes that overlap one or both boundaries of the duplication (as defined by either the feature start, the feature end, or those transcripts that span the entire duplication).
3. Now that you have found all RefSeq genes, which reside within or span the boundaries of segmental duplications, you will need to search for the paralogous gene pair within the related segmental duplication loci. The duplicated gene may be supported by a curated RefSeq mRNA, an unannotated full-length mRNA, or an expressed sequence tag (EST).
 - a. Download EST (all_est.txt.gz) and mRNA (all_mrna.txt.gz) data sets from UCSC (<http://hgdownload.cse.ucsc.edu/goldenPath/mm5/database/>).
 - b. Extract the EST and mRNA mapping information from the above file, and store in GFF3 format. A description of the all_est.txt and all_mrna.txt table format from UCSC can be found at <http://genome.ucsc.edu/goldenPath/gbdDescriptions.html#GenePredictions>.
 - c. Identify the transcripts (EST and mRNA) that map completely within the defined boundary of the duplications (whole gene duplication). Compare the transcriptional start and end coordinates stored in the GFF3 file, and identify those EST or mRNA sequences that fall completely within the coordinates of the duplication.
 - d. Identify the transcripts (EST and mRNA) that are located partially within the defined boundary of the duplication (partial gene duplication). Compare the transcriptional start and end coordinates stored in the GFF3 file and identify those EST or mRNA sequences that overlap one or both boundaries of the duplication (as defined by either the feature start, the feature end, or those transcripts that span the entire duplication).

4. You now have a list of all RefSeq genes and EST and mRNA sequences that reside within duplications. This data set will represent all transcribed sequences that are candidates of recent gene duplication events. To determine the relationship between duplicate genes, a pairwise comparison of all transcripts within related duplications is required.
 - a. To determine whether two transcripts are related (i.e., a duplicated gene pair), you will need to BLAST pairs of transcript sequences.
 - b. Based on our criteria, genes that share greater than 90% DNA sequence similarity for greater than 50% of the length of the transcript can be categorized as a duplicated gene pair.

3.10. Functional Characterization of Genes by Gene Ontology

Duplicate genes may undergo pseudogenization, subfunctionalization, or neofunctionalization (17). To identify the putative function and fates of duplicate genes, an in silico analysis of gene function should be undertaken using the Gene Ontology (GO) resource (18).

1. Obtain the geneID (extract the ID from the gene2refseq.gz file) for each duplicated gene from the NCBI website (<ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/>). The geneID is a unique NCBI identifier (previously Locus Link ID) for each curated RefSeq entry. The GO database can be searched by this unique ID to extract pre-computed gene ontology information. Additional information on the GO project is available at this website <http://www.geneontology.org/>.
2. Using the unique geneID, assign each gene to its GO annotations from each of the three GO taxonomies (biological processes, cellular component, and molecular function) by utilizing the GO Tree Machine (<http://genereg.ornl.gov/gotm/>). You will need to create an account. (Registration is free and will allow the user to save and retrieve analyses.)
3. Create a text file with the list of the geneIDs and save to a file.
 - a. Log onto the GO Tree Machine site, and give the analysis a relevant name for future access.
 - b. From the drop-down menu for “Select the ID type in your file,” select Locus Link ID (same as geneID).
 - c. For “What kind of analysis do you want to do?” select “single gene list” to perform a functional characterization of the duplicated genes.
 - d. You will need to upload the text file with the list of geneIDs previously created and select “MAKE TREE.”
 - e. Alternatively, if, for **step 3c**, you select “interesting gene list vs. reference gene list” you can perform a statistical analysis of duplicated genes to detect GO terms that are relatively enriched compared with the full RefSeq data set. You will need to choose the “MOUSE” reference list.

4. Notes

1. Gene duplication allows for relaxed selection owing to redundancy, and this may allow for processes such as subfunctionalization, neofunctionalization, and pseu-

dogenization. Subfunctionalization occurs when two gene copies specialize to perform complementary functions. Neofunctionalization involves gene duplication whereby one of the genes acquires a new biochemical function. Furthermore, pseudogenization occurs when one of the duplicated genes acquires mutations rendering it nonfunctional.

2. Since chromosome sequence FASTA files are quite large and range in size from 50 to 250 Mb, a significant amount of computational power and memory is required to perform the sequence alignments using MegaBLAST.
3. We will explain how to perform this analysis in a serial manner. It is up to the reader to understand the nuances of their particular cluster or supercomputing installation in order to parallelize the algorithm and achieve the desired results in less time. This means understanding whether using MPI or forking and executing processes is suitable.
4. This protocol can be written in any programming language such as Perl, Java, Python, Ruby, C, or C++. However, typically in bioinformatic applications, algorithms are written in Perl.
5. The BioPerl package is available from <http://www.bioperl.org/>.
6. The current Generic Feature Format version 3 (GFF3) specification is available at <http://song.sourceforge.net/gff3.shtml>.
7. Assembled genomes of several species such as: human, rat, chimpanzee, dog, chicken, and others are available from the download page of the University of California at Santa Cruz (UCSC), <http://hgdownload.cse.ucsc.edu/downloads.html>.
8. The main chromosome sequence assemblies are found in the chrN.fa files, where N is the name of the chromosome. The chrN_random.fa files are pseudo chromosomes containing sequences that are not yet finished or cannot be localized with certainty at any particular place in the chromosome assembly. The chrUn_random.fa file is another pseudo chromosome containing clones that have not been localized to a particular chromosome in the genome. These pseudo chromosomes should not be overlooked since they can often contain sequences that are involved in segmental duplications and have not been included in the main genome assembly perhaps because of their duplicated nature.
9. If the precompiled binaries do not match your computing environment, source code is available from NCBI at ftp://ftp.ncbi.nlm.nih.gov/toolbox/ncbi_tools/ncbi.tar.gz. The instructions detail how to compile and install this suite of tools for your particular computing environment.
10. A total of N^2 sequence alignments are performed for all sequence files where N is the number of files in the genome (i.e., chr1.fa vs chr2 BLAST database and chr2.fa vs chr1 BLAST database). Sequence comparisons are required for all chromosomes in the genome including the pseudo chromosomes.

References

1. Bailey, J. A., Gu, Z., Clark, R. A., et al. (2002) Recent segmental duplications in the human genome. *Science* **297**, 1003–1007.

2. Cheung, J., Estivill, X., Khaja, R., et al. (2003) Genome-wide detection of segmental duplications and potential assembly errors in the human genome sequence. *Genome Biol.* **4**, R25.
3. Cheung, J., Wilson, M. D., Zhang, J., et al. (2003) Recent segmental and gene duplications in the mouse genome. *Genome Biol.* **4**, R47.
4. Bailey, J. A., Church, D. M., Ventura, M., Rocchi, M., and Eichler, E. E. (2004) Analysis of segmental duplications and genome assembly in the mouse. *Genome Res.* **14**, 789–801.
5. Tuzun, E., Bailey, J. A., and Eichler, E. E. (2004) Recent segmental duplications in the working draft assembly of the brown Norway rat. *Genome Res.* **14**, 493–506.
6. Lupski, J. R. (1998) Genomic disorders: structural features of the genome can lead to DNA rearrangements and human disease traits. *Trends Genet.* **14**, 417–422.
7. Stankiewicz, P. and Lupski, J. R. (2002) Genome architecture, rearrangements and genomic disorders. *Trends Genet.* **18**, 74–82.
8. Eichler, E. E. (2001) Recent duplication, domain accretion and the dynamic mutation of the human genome. *Trends Genet.* **17**, 661–669.
9. Ji, Y., Eichler, E. E., Schwartz, S., and Nicholls, R. D. (2000) Structure of chromosomal duplicons and their role in mediating human genomic disorders. *Genome Res.* **10**, 597–610.
10. Iafrate, A. J., Feuk, L., Rivera, M. N., et al. (2004) Detection of large-scale variation in the human genome. *Nat. Genet.* **36**, 949–951.
11. Armengol, L., Pujana, M. A., Cheung, J., Scherer, S. W., and Estivill, X. (2003) Enrichment of segmental duplications in regions of breaks of synteny between the human and mouse genomes suggest their involvement in evolutionary rearrangements. *Hum. Mol. Genet.* **12**, 2201–2208.
12. Bailey, J. A., Baertsch, R., Kent, W. J., Haussler, D., and Eichler, E. E. (2004) Hotspots of mammalian chromosomal evolution. *Genome Biol.* **5**, R23.
13. Ohno, S. (1970) *Evolution by Gene Duplication*. Springer, New York, NY.
14. Buiting, K., Korner, C., Ulrich, B., Wahle, E., and Horsthemke, B. (1999) The human gene for the poly(A)-specific ribonuclease (PARN) maps to 16p13 and has a truncated copy in the Prader-Willi/Angelman syndrome region on 15q11→q13. *Cytogenet. Cell Genet.* **87**, 125–131.
15. Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990) Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410.
16. Zhang, Z., Schwartz, S., Wagner, L., and Miller, W. (2000) A greedy algorithm for aligning DNA sequences. *J. Comput. Biol.* **7**, 203–214.
17. Prince, V. E. and Pickett, F. B. (2002) Splitting pairs: the diverging fates of duplicated genes. *Nat. Rev. Genet.* **3**, 827–837.
18. Ashburner, M., Ball, C. A., Blake, J. A., et al. (2000) Gene ontology: tool for the unification of biology. *Nat. Genet.* **25**, 25–29.



<http://www.springer.com/978-1-58829-575-0>

Gene Mapping, Discovery, and Expression
Methods and Protocols

Bina, M. (Ed.)

2006, XIV, 334 p. 94 illus., 1 illus. in color., Hardcover

ISBN: 978-1-58829-575-0

A product of Humana Press