Chapter 2
Whole-Genome Sequencing Recommendations

Toni Gabaldón and Tyler S. Alioto

2.1 Introduction to Genome Sequencing

2.1.1 Introduction

The recent revolution in sequencing technologies has democratized genome sequencing projects. What once was a daunting endeavor reserved for large international consortia backed by strong funding bodies is now a reasonable goal for a moderately sized research project and can be performed by small teams backed by public or private sequencing and bioinformatic centers. However, the decrease in sequencing costs and the increased availability to groups of sequencing and computing platforms has also brought about the necessity of keeping up with recent developments and strategies, as the sequencing technologies and bioinformatic tools for downstream analyses keep evolving at a fast pace. Sequencing approaches are thus a moving target. However, some general principles can be drawn that can guide the design of a successful genome sequencing project. Common considerations include evaluating known information about size and genome complexity of

T. Gabaldón, Ph.D. (✉)
Bioinformatics and Genomics Programme, Centre for Genomic Regulation (CRG),
Dr. Aiguader, 88, 08003 Barcelona, Spain

Universitat Pompeu Fabra (UPF), 08003 Barcelona, Spain

Institució Catalana de Recerca i Estudis Avançats (ICREA),
Pg. Lluís Companys 23, 08010 Barcelona, Spain
e-mail: tgabaldon@crg.es

T.S. Alioto, B.S., Ph.D.
Centro Nacional de Análisis Genómico, Centre de Regulació Genòmica (CRG-CNAG),
Baldiri Reixac, 4, 08028 Barcelona, Spain
e-mail: talioto@gmail.com; talioto@pcb.ub.es; tyler.alioto@cnag.crg.eu

A.M. Aransay, J.L. Lavín Trueba (eds.), *Field Guidelines for Genetic Experimental Designs in High-Throughput Sequencing*,
DOI 10.1007/978-3-319-31350-4_2

the target genome, obtaining samples with minimal sequence polymorphism, and assessing the needs in terms of contiguity, coverage, and quality of the assembly to address the desired research questions. Here we will provide some general guidelines and recommendations for planning whole-genome sequencing project while focusing on the two most extended applications of whole-genome sequencing. Genome sequencing projects can be grossly subdivided in two broad groups: (1) de novo genome sequencing, in which the objective is obtaining a high-quality genome assembly that can serve as a reference for a species or variety, and (2) resequencing, when there is an available reference genome and the objective is to map sequence variation of an individual or a set of individuals. As we will see below, these two objectives differ in the type of sequencing strategies, in the amount of initial material, as well as in the bioinformatics processing of the data. Despite these differences, all whole-genome sequencing projects have, nevertheless, a similar overall workflow. Four main steps can be defined: (1) sample collection and DNA extraction, (2) sequencing library preparation, (3) sequencing, and (4) bioinformatics data processing. After the data has been processed, this has to be interpreted and additional analyses should be performed. These additional analyses will depend on the particular question under study and will not be the focus of this book chapter.

2.1.2 Sample Collection and DNA Extraction

The first crucial step for whole-genome sequencing is the isolation and quality control of the extracted nucleic acids. The ability to obtain sufficient quantity of fresh samples may sometimes be compromised by the very nature of the organisms under study. For instance, whereas it is simple to obtain enough quantities of material from organisms that can be grown in the lab or that are easily accessible in nature, others may pose serious problems. Examples of problematic materials are material from museum specimens of recently extinct (or rare) species and species that cannot be grown in the laboratory or that are intimately associated with other organisms (e.g., symbionts, obligate parasites). Once samples are collected, DNA should be extracted. The extraction of sufficient quantities of pure, intact, double-stranded, highly concentrated, and uncontaminated genomic DNA is desirable for a reliable whole-genome analysis. The collection and DNA extraction protocols will depend on the organism under study. For instance, the presence of a cell wall in plant and fungal cells makes necessary the use of physical (vortexing in the presence of beads, heating) or biochemical (e.g., cellulase or zymolyase for plants and fungi, respectively) means to break this barrier. Thus a sensible approach for planning of the sample collection and DNA extraction is to survey existing methods that have been previously used for the genetic study of that particular species. In general, standard DNA extraction methods can be used, as long as the necessary quality and quantity of DNA of the target species is produced. These requirements depend on each specific application and sequencing strategy. Sections 2.4 and 2.5 provide some specific guidelines.

2.1.3 DNA Library Preparation

The preparation of sequencing libraries from DNA comprises a series of standard molecular biology reactions, such as fragmentation, amplification, or ligation. In general terms library preparation protocols include the fragmentation of the target DNA and the selection of fragments within a determined size range using gel or bead purification. The size range of the fragments depends on the specific whole-genome sequencing and/or assembly strategy. Subsequent amplification and ligation steps ensure the addition of the specific adaptors at the 5′ and 3′ ends, required for the sequencing phase (see below). Alternatively, "tagmentation" combines the fragmentation and ligation reactions into a single step, which can greatly increase the efficiency of the library preparation. Adapter-ligated fragments are then amplified by polymerase chain reaction (PCR) and purified in gel. Preparation of high-quality libraries and obtaining high yields require a good initial material (see point above) and a careful execution of the library preparation protocol. A number of kits that ease the preparation of libraries are available, and some are provided by the company that manufactures the sequencer. Potential problems in the library preparation phase include biases in the inclusion of genomic regions into the library and the creation of chimeric fragments by artifactual ligation of fragments originating from different genomic regions (Van Dijk et al. 2014).

2.1.4 Sequencing

The principle of next-generation sequencing (NGS) is similar to that of capillary electrophoresis (Sanger) sequencing: sequencing by synthesis, in which the addition of each nucleotide is monitored while DNA polymerase copies a DNA template. However, the critical difference in NGS is that instead of sequencing a single DNA fragment, millions of fragments can be processed in parallel. In the most widely used sequencing-by-synthesis NGS technology, Illumina, DNA polymerase catalyzes the incorporation of fluorescently labeled deoxyribonucleotide triphosphates (dNTPs) into a DNA template strand during a number of cycles of DNA synthesis. At each cycle, the incorporated nucleotides are identified by fluorophore excitation. Sanger sequencing is now obsolete due to its high cost, and some of the earlier generations of NGS technologies are disappearing in favor of newer ones. For instance, Roche has announced that its support for 454 sequencing will be discontinued in 2016. This turnover of sequencing technologies is likely to continue in the coming years. The interested reader is encouraged to read a recent review of current sequencing technologies (Reuter et al. 2015).

2.1.5 Bioinformatics and Data Processing

The sequencing process produces a significant amount of data. For instance, a single run of an Illumina HiSeq2500 will produce 1 terabyte of data in about 6 days. The raw data is primarily provided in the form of plaintext files containing the sequences with associated quality scores. The general format used is the so-called FASTQ format which bundles a FASTA sequence file to its quality data codified as ASCII characters. The information of the quality scores is generally used for an initial quality clipping of the data, in which reads with low qualities are removed or trimmed. Subsequently, in a whole-genome analysis, there are two basic operations with this data. In de novo genome sequencing, reads are assembled into larger contigs by means of detecting sequence overlap between the reads. Alternatively, in genome "resequencing," reads are mapped (i.e., aligned) to a reference genome sequence in order to subsequently detect the desired variations (see below). Both assembly and mapping processes may require significant computational resources. Mapping can be easily parallelized but assembly needs to consider large amounts of data simultaneously which requires access to large amounts of RAM. Currently, 1 terabyte RAM, 32 core servers are often used.

2.2 Review of Achievable Objectives

2.2.1 De Novo Genome Sequencing

The ultimate goal of a de novo whole-genome sequencing project is to obtain a good quality reference assembly and sequence for a representative genome of a given species. What is understood as "good quality" may vary depending on the subsequent application. Generally, one major goal of high-quality genome references is to obtain high-quality gene model annotation. If there is interest in the large-scale organization of the genome and/or the dynamics of repetitive elements, high contiguity is also needed. Ideally, one would wish for a final assembly that contains a single scaffold per chromosome, encompassing all sequence information, from telomere to telomere, and containing no sequencing or assembly errors.

2.2.2 Resequencing

The goal of a genome resequencing project is to annotate, for a given sample (individual, cell line, tissue, etc.), the variations (polymorphisms) in the genome with respect to the reference (or to another sample). These variations may comprise all or a subset of the following types: single-nucleotide changes, including

polymorphisms (SNPs), rare variants (SNVs), or simple somatic mutations (SSMs), insertions and deletions, copy number variations (CNVs), and other rearrangements broadly categorized as structural variants (SVs).

2.3 Recommended Sequencing Platforms

Sequencing platforms are evolving continuously at a fast pace (Reuter et al. 2015). The recommendations outlined here will necessarily be limited to the current available techniques which may soon be surpassed by newer technologies. In general we will phrase our recommendations in terms of read length, throughput, and read pairing strategies. The Illumina platforms give high-quality sequence at the lowest cost per Mb. The main disadvantage is that read length is limited to shorter reads (100–300 bp) because of phasing issues and size restrictions on bridging amplification. Single-molecule sequencing (Pacific Biosciences and Oxford Nanopore Technologies) can achieve longer reads at the expense of error rate, throughput, and cost. Coverage can offset problems in high error rate, at least for de novo assembly.

2.4 Experimental Design Guidelines (Best Practices)

2.4.1 De Novo Genome Sequencing

For a de novo genome sequencing, the most crucial part is to perform the assembly. This process is based on finding sequence overlaps between reads that allow their assembly into contigs and scaffolds that represent longer sequences (Simpson and Pop 2015). The presence of sequence variants within the sequenced DNA sample complicates this process, because these variants create mismatches between reads that correspond to the same genomic locus. The source of sequence variants can originate from the presence of a genetically heterogeneous set of organisms in the sample. Thus one first recommendation is to use a genetically homogeneous source of genomic DNA. In large organisms it is easy to obtain enough material from a single individual. For smaller ones, the use of several individuals from clonal populations is preferred. In diploid organisms (or organisms with higher ploidy) sequence variants of the same locus can be present in the same organism. When possible, the use of inbred lines with reduced heterozygosity levels is recommended.

Once the appropriate source for the DNA has been selected, the next important consideration is the sequencing strategy. This will be determined mainly by the size and complexity of the target genome. For the same sequencing error rate, longer reads and higher sequencing coverage facilitate the assembly process. However, different technologies or sequencing strategies differ in throughput, read length, and

error rate in a way that a combination of several of them is generally the optimal solution. To inform this process, it is highly recommended to learn from previous efforts in sequencing the genomes of highly related species and to gather as much information on the complexity of the target genome in terms of size, level of heterozygosity, and abundance of highly repetitive regions. As the number of sequencing projects increases, such guidelines and learned best practices are starting to be available for more diverse sets of organisms (Richards and Murali 2015). When this information is not available in the literature for that species or closely related ones, one sensible approach is to perform a small sequencing test involving one single run. Simple analysis of k-mers (a short DNA sequence consisting of a fixed number (K) of bases) can inform us on parameters such as estimated genome size, presence of repetitive regions, and heterozygosity, among others (Simpson 2014). A common practice in the era of Sanger sequencing was to clone a few bacterial artificial chromosomes (BACs) and shotgun sequence them first and annotate them with repeats and genes.

2.4.2 Genome Resequencing

Genome resequencing generally involves fewer constraints on the data than de novo sequencing. When the main objective is mostly to determine single-nucleotide polymorphisms and copy number variations, the accuracy and sequence depth of coverage is instrumental, and thus sequencing strategies that provide a higher throughput are preferred. When information on genome rearrangements is required, the design needs to include sequencing strategies that provide information of the relative position of sequences over larger genomic distances. This includes technologies providing long reads or library preparation strategies that capture long genomic fragments from which the extremes are sequenced (mate-pair (MP) or clone end sequencing). Optical mapping (e.g., BioNano Genomics and OpGen) shows potential in this arena, but is not yet standard (Howe and Wood 2015; Tang et al. 2015).

2.5 Technique Overview (Wet Lab Protocol Overview: Library Construction Recommendations)

As mentioned above, sequencing involves DNA extraction and sequencing library preparation. DNA extraction should be performed with protocols that are appropriate to the particularities of the biological material available so that a sufficient quantity and quality of DNA is obtained. A first step that precedes the preparation of the library is the quality control (QC) of the DNA samples. QC involves quantification of the amount of DNA, checking the 260:280 absorbance ratio (ratios between 1.8

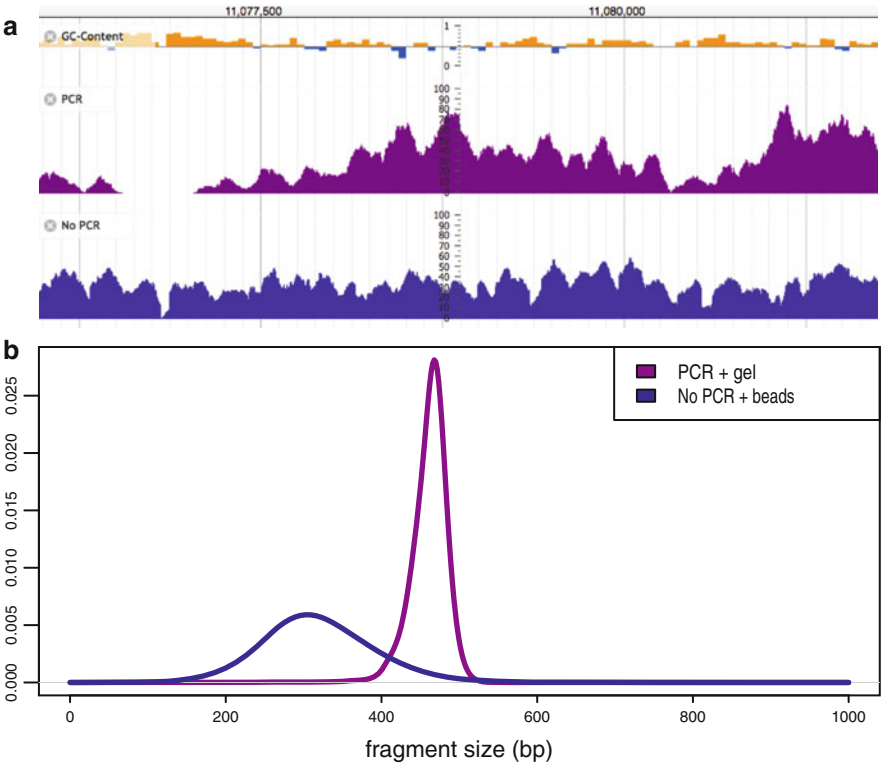

Fig. 2.1 No-PCR library preparation results in more even coverage across wide range of GC content. Panel A shows the coverage profile (both sets of reads were downsampled to 30x at the locus shown) while panel B shows the fragment-size distributions. In *magenta* is the standard PCR protocol (10 cycles of PCR) and in *blue* the no-PCR protocol. While the fragment-size distribution is not as tight, the no-PCR protocol leads to more even coverage, for the most part independent of GC content

and 2 are considered to indicate relatively pure DNA), and running an aliquot on a gel to check integrity and detect ribosomal bands. Ideally, there should be a sufficient amount of DNA to proceed with a no-PCR protocol, which reduces the GC bias effect. The difference in coverage of a particular locus affected by PCR-dependent GC bias is shown in Fig. 2.1. For Illumina SBS sequencing, sample preparation proceeds starting with DNA fragmentation (e.g., with Covaris), A-tailing, adapter ligation, and then size selection (column/beads for automation and consistency or gel for tighter size selection). An aliquot should then be run on a Bioanalyzer or similar instrument in order to choose the most promising libraries for sequencing. Longer fragments are not amplified as well by bridging PCR on the Illumina flow cell, so smaller fragments need to be removed by column purification if longer (>500 bp) fragment libraries are to be sequenced.

2.6 Decision Tree for Good Sequencing Strategy Selection

The most important aspects that anticipate the difficulty of an assembly in a de novo genome sequencing project is the complexity of the target genome, in terms of size, repeat structure, and level of heterozygosity. Determination of the correct sequencing approach is difficult if no prior knowledge is available. Fortunately, depending on the genome size, a lane or two of Illumina sequencing can be analyzed using k-mer counting approaches (Simpson 2014). This can be done using specific software (Preqc, gce) or by using the simple 17-mer counting approach described in Figure S8 of the giant panda genome supplementary information (Li et al. 2010) with a k-mer counter such as Jellyfish (Marçais and Kingsford 2011). See Fig. 2.2

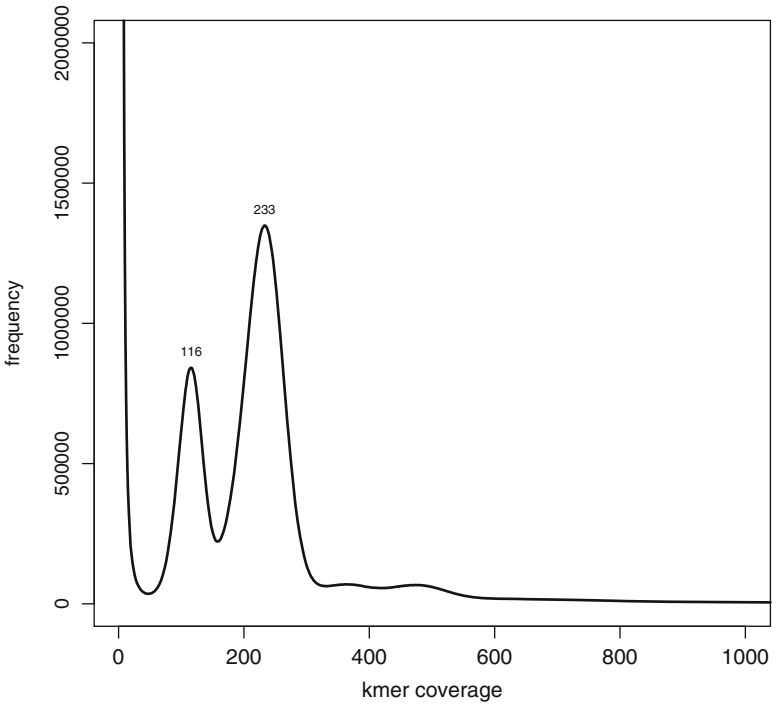

Fig. 2.2 The k-mer frequency plot for a heterozygous genome. Unique 17-mers were counted with Jellyfish. The number of unique 17-mers is plotted according to the number of times they are seen in the input set of Illumina reads (k-mer depth). The highest peak occurs at depth of one. These k-mers that appear only once in the set of reads correspond to sequencing errors. The next highest peak (at k-mer depth of 233) is the main peak, which is correlated with the depth of sequencing. In this case we see a substantial minor peak at half the depth (k-mer depth of 116), which is induced by the presence of polymorphisms. This is a diploid genome, so we only see one minor peak. In genomes of higher ploidy, it is possible to see additional peaks. To the right of the main peak, one can observe a wavelike pattern corresponding to repetitive elements. Larger peaks here are sometimes observed indicating a higher fraction of repetitive content. To estimate the genome size (without correcting for major sequencing biases like GC bias), one can simply divide the total number of k-mers by the depth of the main peak

Table 2.1 Provides several examples of sequencing and assembly strategies

Case	Sequencing strategy	Assembly strategy	Reference
Haploid fungal genome (*Penicillium digitatum*) 26 Mb	Illumina pair-end (PE) 2×50	SOAPdenovo	Marcet-Houben et al. (2012)
	Illumina mate-pairs 2×50 5 kb inserts		
Diploid fungal hybrid (highly heterozygous) (*Candida orthopsilosis*) 12.6 Mb	Illumina pair-end 2×75	SOAPdenovo	Pryszcz et al. (2014)
		REDUNDANS	
Giant panda	Illumina paired-end 2×50 and 2×75	SOAPdenovo	Li et al. (2010)
	Illumina mate-pairs 2×50 2 kb, 5 kb, 10 kb inserts		
Loblolly pine (22 Gb)	Illumina MiSeq paired-end 2×255	MaSuRCA	Neale et al. (2014)
D. melanogaster, *A. thaliana*, *S. cerevisiae*, cell line CHM1	PacBio SMRT sequencing	Celera Assembler with MHAP	Berlin et al. (2015)
E. coli	Oxford Nanopore	Nanocorrect (DALIGNER + POA), Celera Assembler, nanopolish	Loman et al. (2015)

Several different sequencing and assembly strategies are shown from examples taken from a diversity of organisms

for an example. Genome size, repeat content, and heterozygosity can all be estimated with such an approach. Table 2.1 lists some real examples that illustrate different genome complexities and the sequencing strategy that led to good quality assemblies.

One strategy that helps with highly repetitive genomes and highly heterozygous genomes (Fig. 2.3) is to divide the genome into smaller pieces by cloning fragments in BACs or fosmid vectors and sequence them either individually (antiquated Sanger-based clone-by-clone approach) or in pools (more easily managed and cost-effective on the Illumina platform). Drawbacks include cost of making the fosmid library, dividing into pools and preparing the DNA as well as the cost of sequencing, which depends on the target clone coverage. 5× clone coverage (necessary to cover 99 % of the genome) would cost at least five times as much as a standard whole-genome shotgun approach. Perhaps soon, long single-molecule reads may present a fast and cheap replacement for this approach; however, the goal remains the same—to reduce the problems caused by repeats and to deal with polymorphism. With regard to genome resequencing projects, the constraints are fewer, and the characteristics of the genome are generally known for that species, as there is a reference genome available. The genome size determines the required amount of sequencing so that variations can be called with sufficient confidence.

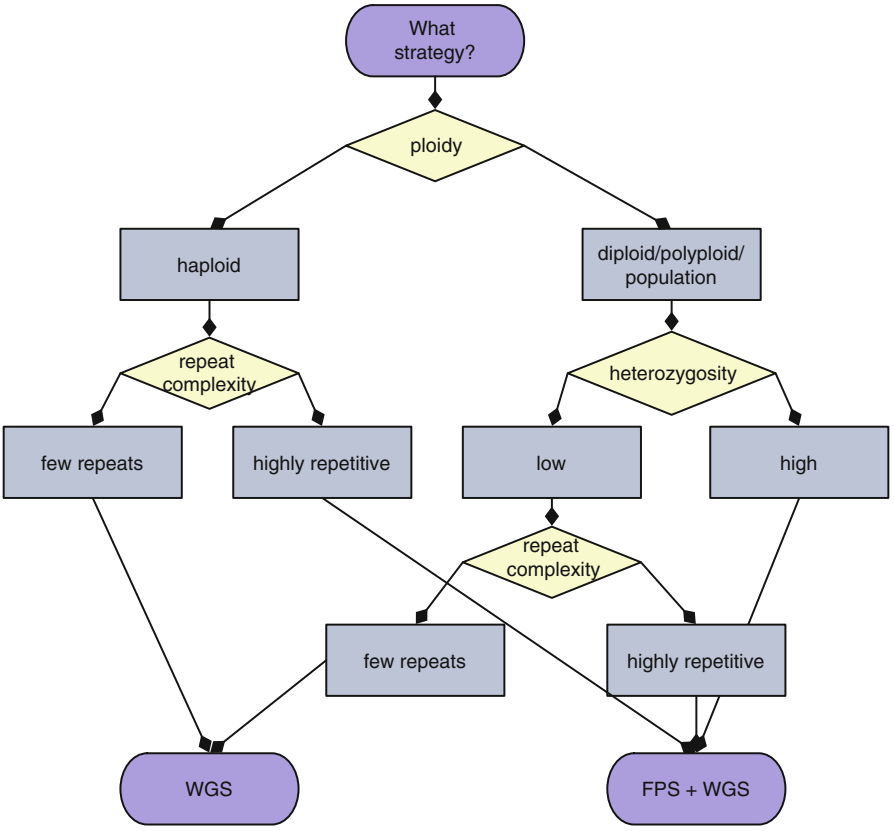

Fig. 2.3 Deciding between sequencing pools of clones vs. pure whole-genome shotgun approach. FPS = fosmid pool sequencing. WGS = whole-genome shotgun sequencing

2.7 Potential Bottlenecks of the Methodology

The sequencing itself is no longer a bottleneck for genome sequencing projects. Depending on the strategy taken, if cloning steps are involved (e.g., fosmid or BAC libraries) or if experimental sequencing library preparation is to be carried out, one can expect delays on the front end. However, the conversion of the raw sequencing data into a high-quality, finished genome assembly is generally one of the major bottlenecks in a de novo genome sequencing project. This process is complicated by the different read lengths, read counts, and error profiles that are produced by different sequencing technologies. In addition, biases in sample preparation, sequencing, and genomic alignment and assembly may result in genomic regions without coverage (i.e., gaps) and in regions with much higher or lower coverage than theoretically expected. GC-rich regions, such as CpG islands, can particularly suffer from low coverage because such regions remain annealed during the amplification step.

Highly repetitive regions, which are prominent in multicellular organisms with large genome sizes, are hard to assemble. In theory, one needs to bridge the repetitive regions by sequencing fragments that expand the whole region and its boundaries, either by using long reads or long mate-pair libraries. Due to its large size and high redundancy, some regions may remain unresolved at any given fragment size. These would need to be closed by targeted approaches that are costly and time consuming. Depending on the expected use of the assembly, this can tolerate the presence of gaps or unresolved regions, and most projects reach a compromise that would satisfy most general applications. Recently, duplicated regions, such as those deriving from tandem gene duplications, are also problematic and most assemblers would collapse these regions into a single one. The same type of regions is problematic in genome resequencing projects, for the same reasons: some regions are less covered among sequenced reads, giving rise to gaps and coverage biases. In addition, short reads may map in multiple loci leading to ambiguity in the localization of a particular variant.

2.8 Bioinformatic Analyses (Best Practices)

2.8.1 Bioinformatician Consulting for Experimental Design

It is important to consult with the team that will perform the bioinformatic analysis earlier on. Poorly designed experiments or sample collection will introduce analytical challenges in downstream analyses; to minimize these complications, bioinformatic teams can provide useful recommendations based on previous experiences. Ideally, a bioinformatic team that has previous expertise in similar analyses and that is easily accessible would be involved in the project from the beginning. Many teams doing bioinformatics research may be recruited to the project if they have a scientific interest in the project. A recommendation is to try to involve them from the start of the project and make them aware of the research interest, asking them to contribute to its solution, rather than simply using them for subsidiary help in the tedious task of "simply" processing the data. This will ensure a high level of implication and a true interest in producing the best results. An important guideline in this respect is to reward the help of bioinformatic collaborators with due recognition in terms of authorship (Chang 2015). Beyond collaborations from other groups, bioinformatic support can be obtained from core services at many large institutions or companies that specialize in bioinformatic analyses. Assessing what is the expertise of these teams in projects similar to the one at hand is crucial to ensure a successful experience. Finally, it is advisable to envision the hiring of bioinformaticians in the project. If bioinformatic expertise is lacking in the host group, these specialists could ideally be embedded (at least for some time) in teams of data analysis collaborators or cores, so that he/she benefits from expert knowledge accumulated in experienced teams.

2.8.2 *Analysis Workflow Overview (From Raw Reads QC to Functional Characterization)*

2.8.2.1 Quality Clipping, Filtering, and Error Correction

Invariably, the first step of data analysis is the quality clipping and filtering of the raw sequencing results. An efficient filtering of low-quality data will minimize problems in downstream analysis. One first filtering that must be done is to remove any partial adapter sequence that may have been sequenced. This can occur when a given sequenced fragment was shorter than the read length. In addition it is possible that concatenated adapter-only sequences have been sequenced. These sequences must be removed. Subsequently it is highly advised to perform a control of the quality of the reads which may lead to filtering or trimming reads of regions thereof that have low quality. As mentioned above raw sequencing reads are made available as FASTQ text files, in which each short read takes up four lines: the read identifier (starting with an @), the DNA sequence itself, another identifier (same as line 1, but starting with a + (or sometimes only consisting of a +)), and the Phred quality score for each base in the read. The quality score is encoded with an ASCII character code (http://www.ascii-code.com/). Illumina and other manufacturers currently (as of v1.8) use the Sanger Phred ASCII encoding offset of 33, so that the ASCII code 33 (!) is 0, and ASCII code 74 (J) is 41. Quality scores are defined as $Q_{\text{phred}} = -10\log10(p)$, where p is the estimated probability of a wrong base call. So a Q_{phred} of 20 corresponds to a 99 % probability of a correctly identified base (1 % error; see Table 2.2).

One of the first evaluation routines is to assess how the distribution of quality scores and nucleotides looks like. This is generally done by summarizing and plotting the data (typically with FASTQC or a similar software). A typical plot includes the quality score per residue (see Fig. 2.4 for an example of a 250 nt HiSeq2500 read 1). Quality scores generally decrease over the length of a read (i.e., first incorporated nucleotides are determined with higher accuracy), and how fast these declines occur can vary from one sequencing run to the next. This plot will reveal whether the sequencing run maintained an overall high quality during the whole procedure or whether trimming the last residues of the reads would be advisable. Q30% (average percent of bases >Q30) is a frequently used metric to determine the overall quality of a run, while error rate (estimated by spiking in PhiX DNA as a control) is probably the most relevant metric for downstream analyses. Quality

Table 2.2 Relationships between Phred quality scores and accuracy

Phred quality score	Probability of incorrect base call	Base call accuracy
10	1 in 10	90 %
20	1 in 100	99 %
30	1 in 1000	99.9 %
40	1 in 10,000	99.99 %
50	1 in 100,000	99.999 %
60	1 in 1,000,000	99.9999 %

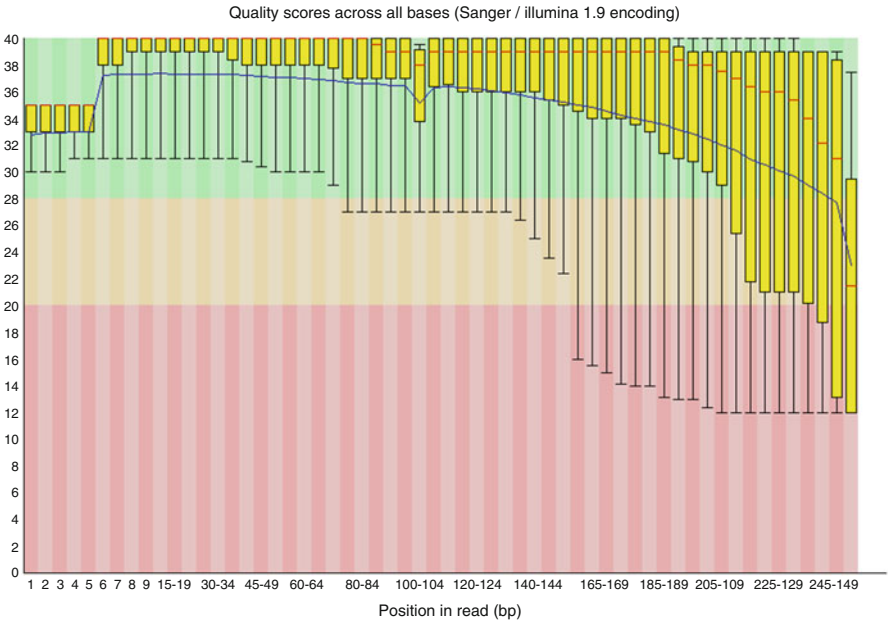

Fig. 2.4 FASTQC per-base quality report. This example is from read one of a typical 2×250 nt run of a HiSeq2500. The first few cycles typically show some sequence bias and lower quality. Sometimes a particular cycle (in this case around base 100) shows a slight dip in quality, perhaps due to a temperature fluctuation. Gradual decrease in quality is observed after 150 cycles, falling more rapidly after 200 cycles

scores and error rates are related, of course, but not perfectly, which is why some analyses recalibrate base qualities using packages such as the Genome Analysis Toolkit (GATK) from the Broad Institute (McKenna et al. 2010). Another informative plot is the base composition per residue, e.g., what fraction of A's, C's, G's, and T's has a given position in a read. A perfectly random sampling of reads along a genome should render horizontal lines for each residue, with their values in accordance to the overall base composition of the genome (e.g., with GC content). Nonuniform patterns reveal biases in the composition of the reads and may indicate strong amplification biases or the presence of sequenced adapters in the reads. In addition, it is recommended to assess the fraction of duplicate reads (identical reads present that are present in the dataset), as they may originate from primer or PCR bias, and thus a large fraction of duplicate reads may be indicative of a poor cDNA library. Several tools and packages are available for performing the quality assessment and trimming of FASTQ files. Some currently popular options include FASTX, FASTQC, Trimmomatic (Bolger et al. 2014), cutadapt, trim_galore, or PRINSEQ (Schmieder and Edwards 2011).

In addition to trimming, another way to deal with errors is to correct them. For de novo genome assembly, error correction can reduce memory consumption and lead to simpler assembly graphs. Popular assembly tools SOAPdenovo (Luo et al. 2012),

ALLPATHS-LG (Gnerre et al. 2011), and SGA (Simpson and Durbin 2012) have built-in error correction. Some tools such as QUAKE (Kelley et al. 2010) can be run stand-alone. The basic idea behind most of these approaches is that low-coverage k-mers (presumably caused by sequencing errors) can be corrected by high-coverage k-mers within a low edit distance of the low-coverage k-mer.

2.8.2.2 Genome Assembly

Essentially, there have been two successful approaches to the assembly of sequencing reads into a genome sequence: those based on the basic overlap-layout-consensus (OLC) algorithm and those based primarily on de Bruijn graphs. For detailed reviews, see Miller et al. (2010) and Compeau et al. (2011). Archetypal OLC assemblers include Phrap, TIGR assembler, PCAP, JASS, Phusion, Arachne, Newbler, and the Celera Assembler. In the era of Sanger sequencing-based genome projects, these programs were successful in producing high-quality draft genomes, although the final contiguity reported was often achieved by combining clone-based approaches and lots of manual "finishing" work. The basic approach taken by Celera Assembler, for example, is as follows:

1. Overlap
   (a) Overlaps are computed among the set of all reads ("all against all") using a BLAST-like seed and extend algorithm. *ovl* (classic) or *mer* (for 454) are used as the overlapper. Both use a seed and extend approach, but with parameters tuned to Sanger or 454 read length and error profiles, respectively. Other assemblers use similar seed approaches (like BLAST) and usually process the initial overlaps with Smith-Waterman alignment.
   (b) Such overlap computations use the majority of CPU time.
2. Layout
   (a) The genomic order or "layout" of the reads is determined by computing a Hamiltonian path in which reads are represented as vertices in a graph, the overlaps are edges, and a path is found that visits each vertex once and only once.
   (b) The CA module unitigger is used to compute initial high-confidence contigs.
   (c) Scaffolder uses additional mate-pair data to join unitigs with estimated gaps.
   (d) The layout step often uses the most memory.
3. Consensus
   (a) The optimal multiple sequence alignment is usually unattainable. Heuristics are used to guide the alignment and output a consensus. Variants can sometimes be output. Depending on the length and pairing of input data, the variants can be phased.

Practically speaking, most OLC software cannot be run efficiently on NGS data. However, the cost of hardware (CPUs and memory) has fallen and the algorithms and implementations improved so much that, for example, the Celera Assembler can now be run on Illumina data, although still not as efficiently as the k-mer graph (de Bruijn graph)-based assemblers.

With the introduction of massively parallel sequencing, which is characterized by the production of a very large number of short reads, OLC approaches became computationally infeasible, necessitating new algorithmic development. Fortunately, the mathematics had already been worked out and only required co-opting for the assembly problem. A Eulerian path, in particular the k-mer version of the de Bruijn graph, is similar to a Hamiltonian path, but where vertices are the k-mers and edges are the k—one overlaps and each edge is visited at least once. The solution to this problem is more computationally feasible and has become popular for assembling NGS data. However, the problems that complex genomes present, such as repeats and heterozygosity, become even harder to resolve. Extra attention must be paid to read trimming and error correction and to cleaning of the assembly graph (pruning tips, popping bubbles, etc.).

To generate high-quality assemblies from NGS data, one more or less follows the general workflow depicted in Fig. 2.5. It is important to preprocess the read data as described above and to do quality control checks (e.g., FASTQC) and plot k-mer frequencies to estimate genome size and complexity. Then, the overlap graph (as discussed above the more efficiently computed by de Bruijn graph) is created. To generate unitigs using a de Bruijn graph, k-mers of different lengths should be tested. K-mers that are too short will result in an assembly broken by short tandem repeats, while k-mers that are too long will result in assemblies broken at regions of low coverage. Moreover, longer k-mers often require more memory to store k-mer counts, as errors create a number of unique k-mers equal to the k-mer size each time an error occurs in a read. Then pairing information from short fragment paired-end reads and/or long fragment mate-pair reads is used to join unitigs into longer contigs and these contigs into scaffolds containing gaps of estimated size using the mean and standard deviation of the fragment lengths for each sequencing library. It is important to detect potential misassemblies along the way by trying to detect chimeras, aberrant depth (repeat) contigs, or compression/expansion errors either by determining the consistency or support of the read data aligned back to the intermediate assembly (e.g., using REAPR (Hunt et al. 2013)) or by using external information such as physical or genetic maps or alignment to phylogenetically close high-quality reference genomes. After misassembly correction, one can fill scaffold gaps using either built-in modules or stand-alone programs such as GapFiller (Boetzer and Pirovano 2012). Polishing, or fixing small errors such as single-nucleotide substitutions or indel errors like homopolymers, can be achieved using approaches nearly identical to variant calling of resequencing data. Finally, if genetic, physical, or optical maps have been generated, the assembly can be "anchored" to chromosomes/linkage groups/pseudomolecules by mapping the positioned markers onto the scaffolds and then ordering and orienting them if possible to create a final anchored assembly.

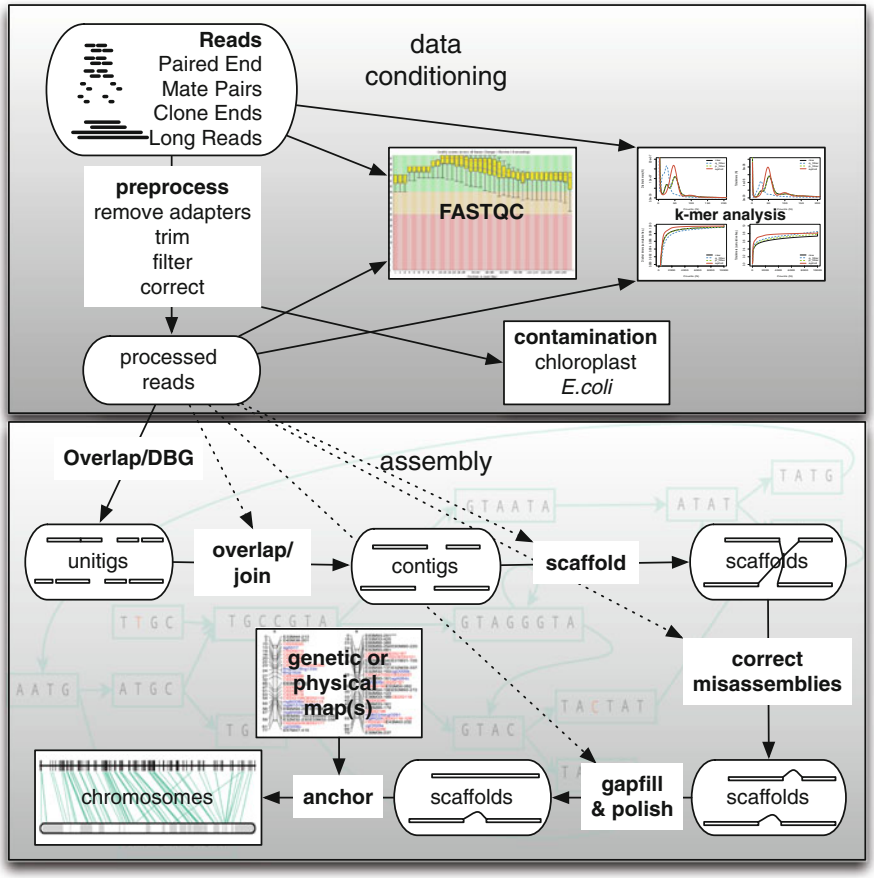

Fig. 2.5 General assembly workflow

2.8.2.3 Read Mapping and Variant Calling

Read mapping refers to the process of aligning short sequencing reads to a reference sequence, which is generally a complete genome, but can also be a transcriptome. A plethora of computer programs have been developed that map (also called align) reads to a reference sequence. These programs use different algorithms that vary in speed and accuracy (Fonseca et al. 2012). The majority of fast mapping algorithms perform indexing on the read sequences or the reference sequence, or sometimes both. Similar to Google's indexing of websites, a preprocessing of sequence data creates an index data structure that accelerates the search for a near-exact match. Depending on the nature of the index, mapping algorithms can be roughly grouped into three categories: algorithms based on hash tables, algorithms based on suffix trees, and algorithms based on merge sorting (Li and Homer 2010). Most existing algorithms belong to the first two types. All algorithms based on hash tables keep

the position of each k-mer subsequence (a sequence of k residues) of the query in a table (hash table) and scans the databases for k-mer exact matches (called seeds). Algorithms based on suffix trees first identify exact matches using a data structure that stores all the suffixes of a string and then build inexact alignments around the exact matches. Different mappers diverge in their particular implementation of the strategy and in their inclusion of additional parameters that enable more efficient mapping of dissimilar types of data, for instance, the ability to perform alignments containing gaps, or split alignments, or the possibility to incorporate information from pair-end or mate-pair reads. The most immediate goal of read mapping is to create an alignment file also known as a sequence alignment/map (SAM) file. The SAM file contains one line per mapped read indicating the reference sequence and position to which it maps, as well as a Phred-scaled quality score of the mapping, among other details (Li et al. 2009). The SAM format is human readable and easier to process by conventional processing programs. The BAM format provides binary versions of most of the same data and is designed to provide higher compression.

One of the main purposes of genome resequencing is to discover genetic variation among related individuals or samples in a large scale. This inference is generally done after the mapping of the reads is completed. Again, a number of algorithms and computer programs are available that are designed to call variants from SAM/BAM files. Most are focused on the detection of single-nucleotide polymorphisms (SNPs) or small insertions and deletions. A variation from the reference sequence will result in mismatches, gaps, or a significantly different coverage, and most algorithms perform a statistical analysis of mapping results to provide a call of present variants. Most prevalent types of sequence variation, including SNPs, indels, and larger structural variants, are generally stored in a specific format denoted as variant call format (VCF). Larger variations such as copy number variants (CNVs) and genomic rearrangements are generally detected with specific programs. For instance, CNVs can be detected by methods that assess the depth of coverage, by piling up aligned reads against genomic coordinates and then calculate the depth of coverage along windows and compare it with the average coverage of the region (Consortium 2012). Genomic rearrangements can be assessed by using information of the mappings of mate-pair or pair-end reads (Xi et al. 2010).

2.8.3 Sequencing Depth (Number of Aligned Reads Required for a Reliable Analysis)

2.8.3.1 Introduction

Despite significant drops in price, sequencing costs still set limits to the total amount of sequence that can be generated. In addition, various analyses may require different minimal sequence coverage to provide reliable results. These factors are keys for the experimental design of a whole-genome sequencing project (Sims et al. 2014). Here we will provide an overview of current guidelines and precedents with

respect to sequence coverage. The empirical per-base coverage (or sequencing depth) is the exact number of times that a base in the reference is covered by a high-quality aligned read in a given sequencing experiment. However, when planning a whole-genome sequencing project, we must deal with the expected coverage, which is the average number of times that each nucleotide in the genome is expected to be sequenced given a certain number and length of reads and with the assumption that reads will be randomly distributed across the genome. Lander and Waterman (1988) described this as $c = LN/G$, where L is the read length, N is the number of reads, and G is the haploid genome length. Sequencing depth is generally expressed in fold coverage units (e.g., 10× means that an average base is covered by ten reads).

Redundancy in sequencing data is necessary to overcome sequencing errors and biases. If a sequencing method would be 100 % accurate and perfectly balanced over the entire genome sequence, then a 1× depth of coverage would suffice for all downstream analyses. However, in reality, sequencing errors are not negligible. To distinguish errors from sequence variants, one needs to assess all reads mapped to a given residue. For instance, at a 1 % error rate, the combination of ten identical reads that cover the location of the variant will produce a strongly supported variant call with an associated error rate of 10^{-20}. It must be noted, however, that increased depth of coverage cannot solve other sequencing problems such as gaps or ambiguous alignments in repetitive regions. Thus, sequencing depth must be considered in combination with alternative sequencing strategies (e.g., paired-end, mate-pairs).

2.8.3.2 De Novo Sequencing

The required depth in a de novo genome sequencing project is determined by several factors including the sequencing method and strategy, read length, the assembly approach, and the complexity in terms of repetitive regions of the genome (length, similarity, and abundance of the repetitive regions). For instance, Sanger-based sequenced genomes such as dog and human provide good reference assemblies at low coverage (7–10×), whereas much higher sequencing depths (~73×) using short reads rendered poor assembly qualities in the giant panda, a genome of similar size and complexity to that of dog (Lindblad-Toh et al. 2005; Li et al. 2010). For Illumina data, the depth and library types need to be matched to the assembly algorithm, which can have very specific requirements. For example, ALLPATHS-LG requires a 2 × 100 PE library of fragment length 180 bp (20 bp overlap) at >50× coverage and at least one MP library of 3 kb fragment length also at 45–50× coverage. Larger mate-pair libraries are necessary for more contiguous assemblies. It can also take advantage of long PacBio reads at about 50× coverage. This software is being replaced by DISCOVARdenovo, which requires 50–80× coverage by a single 450 bp fragment PE library sequenced in 2 × 250 PE mode on a HiSeq2500. Of course additional scaffolding with MP libraries or other means can and should be carried out with stand-alone scaffolding software.

SOAPdenovo and ABySS are more flexible in the number of input libraries and coverage. ABySS is able to use distributed memory and thus has more flexibility in

terms of the number of reads you give it. However, best results are achieved when at least 100× coverage in PE reads (all PE libraries combined) is used for the initial de Bruijn graph construction, with a minimum of 20–30× per library for scaffolding. Higher coverage can give better scaffolding results, but with diminishing returns.

For PacBio-only assemblies, one can use the MHAP algorithm (Berlin et al. 2015) that is now available as part of the Celera Assembler. Required coverage is a minimum of 50–70×. This strategy is able to reconstruct whole chromosome arms of the *D. melanogaster* genome. A similar approach for Oxford Nanopore Technologies two-directional reads has been implemented in a pair of packages called *nanocorrect* and *nanopolish* (Loman et al. 2015). At least 25× coverage is necessary, with higher depth likely to yield better results. For both technologies, the error rate is typically too high to run self-alignments with more traditional aligners, a step necessary for calculating overlaps; thus they utilize new alignment algorithms (the MinHash Alignment Process (MHAP) and DALIGNER (https://github.com/thegenemyers/DALIGNER), respectively) that are roughly based on the idea of shared k-mer content.

2.8.3.3 Resequencing

Early resequencing studies of humans using Illumina short-read approach showed that the required sequencing depth to detect most of the SNPs and short indels was 15× when they were homozygous and 33× if they were heterozygous (Bentley et al. 2008). Subsequent studies have provided similar estimates, and thus depths exceeding 30× have become the de facto standard in resequencing analyses (Ajay et al. 2011). The use of low base qualities and nonuniform coverage may challenge the detection of variants, so these numbers should be considered after filtering reads by quality and assuming a uniform coverage over the genome. For the detection of CNVs, uniformity of sequencing coverage is instrumental to avoid false positives. In addition, accurate inference of break points and absolute copy number estimation improve with increasing read depth.

2.8.4 Difficulties of the Bioinformatic Analyses

Although an increasing number of user-friendly solutions are becoming available, the difficulty of the bioinformatic analyses required remains high. Attempts to undergo a genomic analysis without the required expertise can lead to frustration and dangerous misinterpretations of the data. Thus, it is highly advisable to include in the team the necessary human resources with sufficient expertise. As mentioned above, this can be achieved through collaborations with bioinformatic teams, service cores, or companies.

2.8.5 Expected Results

2.8.5.1 De Novo Sequencing

The expected result for a de novo genome sequencing project is a high-quality genome assembly, which is annotated to some satisfactory level. The quality of the assembly in terms of contiguity depends on the expected use. As mentioned above, the optimal target is an end-to-end, one chromosome one contig, no-gap containing accurate sequence. However, such an objective has only been accomplished for small genomes, and larger genomes containing repetitive sequences are generally incomplete, despite extensive effort. As an example, the human genome still contains hundreds of large, unresolved gaps that correspond to repetitive or heterochromatic regions. Fortunately, not all applications of de novo genome sequencing require full completion of the assembly. For instance, protein-coding regions of the genome, which remain the main focus of de novo genome sequencing, are generally well recovered. However, a highly fragmented genome may split genes across different contigs. If the interests lie on higher-scale properties of the genome such as gene order, high contiguity in the assembly is required, although the presence of undetermined sequences may be allowed. Finally, some analyses are highly demanding on the assembly completion, for instance, when the focus is in determining the content and distribution of transposable elements.

2.8.5.2 Resequencing

The expected results for a genome resequencing analysis would be a comprehensive catalog of genetic variations in individuals, samples, or populations with respect to a given reference. This includes single-nucleotide variants, small insertions and deletions (indels), larger structural variants (such as inversions and translocations), and copy number variants (CNVs).

2.8.6 Effective Result Reporting

2.8.6.1 De Novo Sequencing

Genome assemblies are reported and shared as a set of files including:

1. A set of FASTA files corresponding to contigs, scaffolds, and/or chromosomes. Scaffold FASTA files are the most common and useful of these.
2. One or more AGP files describing the structure of the assembly with contigs as the building blocks. An AGP (acronym for “A Golden Path”) is a commonly

used file format for describing assemblies. This format was originally conceived by the International Human Genome Sequencing Consortium and used to describe the genome assembly of human. It is now the most commonly used format for specifying assembly information (http://www.ncbi.nlm.nih.gov/projects/genome/assembly/agp/AGP_Specification.shtml). There can be multiple AGP files: one for scaffolds, another for superscaffolds, and another for pseudomolecules/chromosomes/linkage groups.

2.8.6.2 Assembly Metrics

A wide range of basic statistics are available that serve to describe the quality of a given assembly. The most basic one is the total size of the assembly (assembly size), which reports the total number of bases contained in the genome. When compared to the estimated or known size of the target genome, this metric can be transformed into the coverage of a given assembly over the genome of interest. Another set of useful metrics inform on the contiguity of the assembly, that is, whether the assembly is formed by many, small contigs or by few large ones. These statistics can refer to contigs or scaffolds, being the simplest metric the total number of contigs and scaffolds in that assembly. Rather than the mean contig length, a metric known as the N50 is often used to describe the contiguity of an assembly. It is defined as the length N for which at least 50 % of all bases in the sequences are contained in sequences of length N or longer. An easy way to compute it is to order your sequence lengths from longest to shortest and compute the cumulative sum of their lengths; when the sequence is reached, which brings the sum to greater than or equal to half of the total length of the assembly, the N50 equals the length of that sequence. The metric can also be computed for other proportions of the assembly, for example, N10 or N90 (where 90 % of the assembled bases are in scaffolds/contigs of length N90 or longer). When comparing multiple assemblies or assembly methods on a genome with an accurate size estimate, the assembly length can be substituted by the estimated genome length to give NG50 (NG10, NG80, NG90, etc.) values.

As many would point out, contiguity is good to have but not at the expense of correctness. It would be easy to make an assembly of one single contig by joining all sequences end to end, yet it would be highly inaccurate. Aggressive scaffolding requiring low support can inflate N50 values and the expense of more misassemblies. Thus other metrics should be considered. Gene content (both the completeness of the gene set and the connectivity of exons) is a very important point to consider. The CEGMA (Parra et al. 2007) or BUSCO (Simão et al. 2015) pipelines which search for a conserved set of core eukaryotic genes in draft genomes can report on both completeness of the genome and its connectivity. Several other analysis suites aim to provide a more complete picture of quality. FRCurve (Vezzi et al. 2012) can be

run on assemblies to which at least one paired-end library and, optionally, one mate-pair library have been mapped and provided in BAM format. QUAST (Gurevich et al. 2013) is another useful tool for plotting a number of contiguity and gene content metrics.

2.8.6.3 Genome Resequencing

Efficient reporting of a resequencing study includes making available the raw reads, the variant calling files (VCFs, (Danecek et al. 2011)), as well as a statistical analysis that will depend on the focus of study (detection of disease variants, population structure, etc.). Quality metrics for call sets are lacking. Pipelines can be benchmarked (e.g., using the Genome in a Bottle materials (https://www-s.nist.gov/srmors/view_detail.cfm?srm=8398)), but individual call sets, unless independently validated with an orthogonal technology, cannot. As such, it is important to report base frequencies, base qualities, mapping qualities, allele frequencies, strand bias, positional bias, etc. so that the data may be reanalyzed at a future date by more up-to-date pipelines, perhaps tuned to return few false positives or few false negatives, depending on the goal of the resequencing experiment. It must be noted that variant/mutation calling procedures may vary depending on the frequency of the alternate allele.

2.8.6.4 Repositories to Upload Research Results Data for Publication

The European Nucleotide Archive (ENA (Leinonen et al. 2011)) is Europe's primary nucleotide-sequence repository. It comprises the Sequence Read Archive (SRA) where raw reads from different sequencing experiments can be submitted. The European Genome-phenome Archive (EGA) is the appropriate repository for human resequencing data. Raw data (FASTQs), alignments (BAMs), and genotypes and structural variants (VCFs) can all be submitted. Access is governed by a data access committee.

2.9 Main Remarks and Conclusions

To summarize, successful whole-genome sequencing requires the ability to think ahead and develop a strategy that accomplishes the goals of the project. Specifically, we recommend the following:

Before the Project Starts

- Survey existing genomic literature in search of required information (genome complexity, heterozygosity, size)
- Study previous projects on similar organisms.
- In the absence of related studies, consider a sequencing test to obtain preliminary data on genomic characteristics.
- Plan the sequencing strategy according to the assembly/analysis strategy that you will use afterward.
- Make a concerted effort to obtain high-quality DNA material, from samples of minimal polymorphism if possible (for genome assembly).
- Engage collaborators that will participate in the analysis from the beginning.
- Consider data storage and processing costs in addition to library preparation and sequencing costs.
- Balance cost with desirable depth of sequencing, most useful library fragment sizes, and longest reads possible (for genome assembly). Underfunding a project will achieve suboptimal results. In some cases, additional sequencing can save a project; however, depending on the strategy, it may have been a waste.

During the Project

- Revise and optimize as you go. If a strategy is not working, try to diagnose the problem and fix it as early as possible.
- Coordinate the work of the different teams involved, avoid redundant analysis, and establish clear dependencies and workflows.
- Freeze assembly and annotation at the time downstream analyses and start to avoid multiple recomputations due to constant minor updates.

After the Project

- Use efficient reporting and standard formats.
- Submit assemblies, annotations, raw data, and main analyses to public repositories.

Annex: Quick Reference Guide

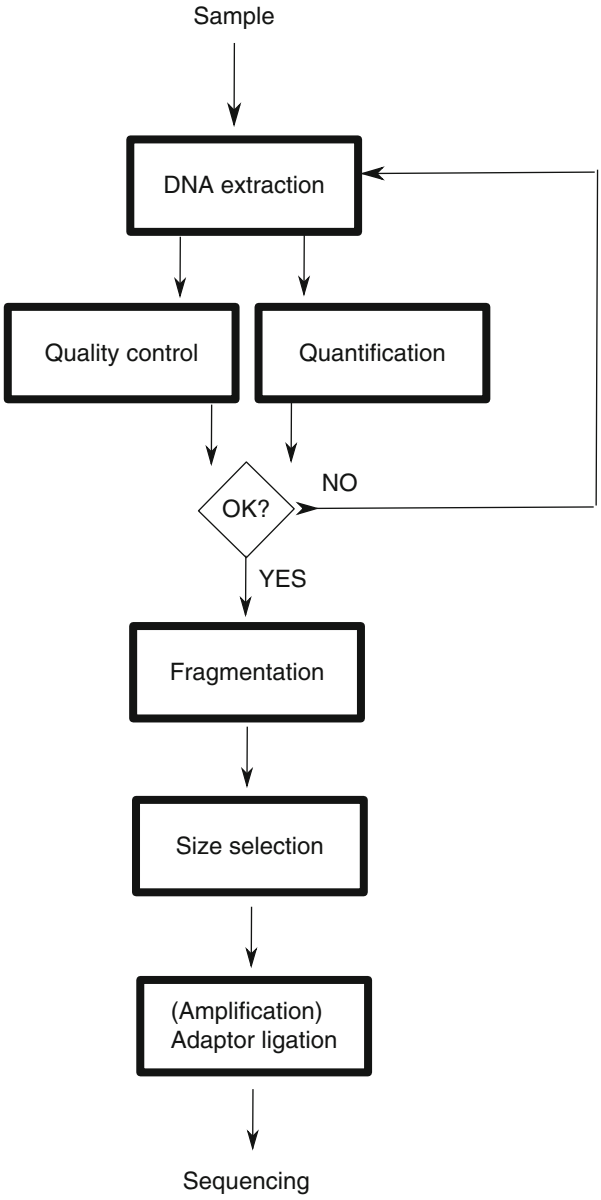

Fig. QG2.1 Representation of the wet lab procedure workflow

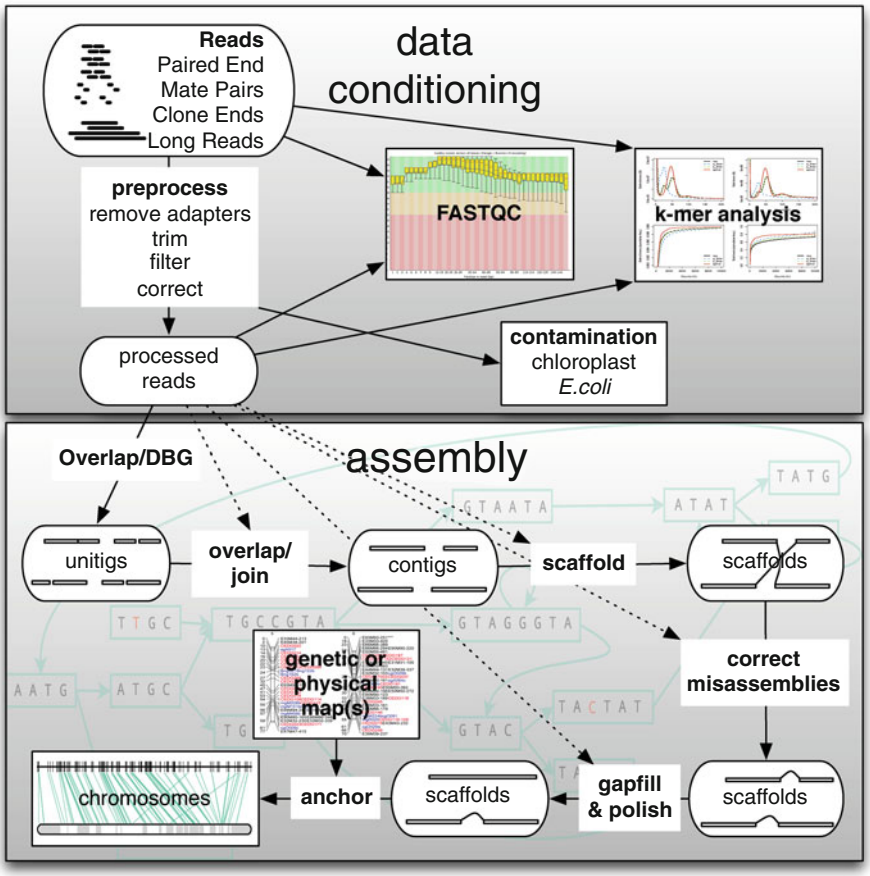

Fig. QG2.2 Main steps of the computational analysis pipeline

Table QG2.1 Experimental design considerations (I)

Project phase	Recommendations
Sample	1. Reduce expected genetic variability of the sample by using minimal number of inbreed individuals if possible
Sequencing strategy	1. Determine early size, heterozygosity, and repetitive structure of the target genome
	2. Consider recent experiences in similar organisms
	3. Consider contiguity and coverage needed to address the specific questions
	4. Combine throughput with long-range approach (FOSMIDS, longer read technology)
Bioinformatic analyses	1. Engage expert collaborators from the beginning
	2. Survey state-of-the-art methodology
	3. Consider specificities of the project (e.g., high heterozygosity)
Efficient reporting	1. Deposit all possible data (raw reads, assemblies, annotations) in public repositories
	2. Link data to publication
	3. Report standard quality parameters for assembly and annotation
	4. Use standard formats when possible

De novo genome sequencing hints

Table QG2.2 Experimental design considerations (II)

Project phase	Recommendations
Sample	1. Plan balanced sampling of a sufficient size to address the questions driven by the project
Sequencing strategy	1. Consider required sequencing depth depending on size of the target genome and required coverage for efficient variant calling
	2. Consider whether determination of structural variants is needed and use required strategy (e.g., pair-end, mate-pair libraries)
Bioinformatic analyses	1. Engage expert collaborators from the beginning
	2. Survey state-of-the-art methodology
	3. Consider specificities of the project (e.g., high heterozygosity)
Efficient reporting	1. Deposit all possible variation data in public repositories
	2. Link data to publication
	3. Use standard formats when possible

Whole-genome resequencing hints

Table QG2.3 Available software recommendations

Software	Function	Input	Reference	Result output	Result format
FASTQC	Quality control	FASTQ files	http://www.bioinformatics.babraham.ac.uk/projects/fastqc/	Quality control tables/text	text
SOAP denovo2	Genome assembly	FASTA, FASTQ files	Luo et al. (2012) SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler. GigaScience 1:18.	Genome assembly	Scaffold sequences
Velvet	Genome assembly	FASTA, FASTQ files	Zerbino, D. R. (2010) Using the Velvet de novo Assembler for Short-Read Sequencing Technologies. Current Protocols in Bioinformatics. 31:11.5:11.5.1–11.5.12.	Genome assembly	FASTA, Graph
MaSuRCA	Genome assembly	FASTQ files	Zimin et al. (2013) The MaSuRCA genome assembler *Bioinformatics* 29 (21): 2669–2677	Genome assembly	FASTA
ABySS	Genome assembly	FASTA, FASTQ, qseq, SAM files	Simpson et al. (2009) ABySS: a parallel assembler for short-read sequence data. Genome Res. 19(6):1117–23	Genome assembly	FASTA
SPADES	Genome assembly	FASTA, FASTQ, BAM	Bankevich et al. (2012). "SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing" *Journal of Computational Biology* 19: 455–477	Genome assembly	FASTA, FASTG

Table displaying a selection of the recommended software available for the computational analysis of data yielded by this technique
This table has been generated by the editors for the quick reference guide corresponding to this chapter

References

Ajay SS, Parker SCJ, Abaan HO, Fajardo KVF, Margulies EH (2011) Accurate and comprehensive sequencing of personal genomes. Genome Res 21:1498–1505

Bentley DR, Balasubramanian S, Swerdlow HP, Smith GP, Milton J, Brown CG, Hall KP, Evers DJ, Barnes CL, Bignell HR et al (2008) Accurate whole human genome sequencing using reversible terminator chemistry. Nature 456:53–59

Berlin K, Koren S, Chin C-S, Drake JP, Landolin JM, Phillippy AM (2015) Assembling large genomes with single-molecule sequencing and locality-sensitive hashing. Nat Biotechnol 33:623–630

Boetzer M, Pirovano W (2012) Toward almost closed genomes with GapFiller. Genome Biol 13:R56

Bolger AM, Lohse M, Usadel B (2014) Trimmomatic: a flexible trimmer for Illumina sequence data. Bioinformatics 30:2114–2120

Chang J (2015) Core services: reward bioinformaticians. Nature 520:151–152

Compeau PEC, Pevzner PA, Tesler G (2011) How to apply de Bruijn graphs to genome assembly. Nat Biotechnol 29:987–991

Consortium T 1000 GP (2012) An integrated map of genetic variation from 1,092 human genomes. Nature 491:56–65

Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, Handsaker RE, Lunter G, Marth GT, Sherry ST et al (2011) The variant call format and VCFtools. Bioinformatics 27:2156–2158

Fonseca NA, Rung J, Brazma A, Marioni JC (2012) Tools for mapping high-throughput sequencing data. Bioinformatics 28:3169–3177

Gnerre S, Maccallum I, Przybylski D, Ribeiro FJ, Burton JN, Walker BJ, Sharpe T, Hall G, Shea TP, Sykes S et al (2011) High-quality draft assemblies of mammalian genomes from massively parallel sequence data. Proc Natl Acad Sci U S A 108:1513–1518

Gurevich A, Saveliev V, Vyahhi N, Tesler G (2013) QUAST: quality assessment tool for genome assemblies. Bioinformatics 29:1072–1075

Howe K, Wood JM (2015) Using optical mapping data for the improvement of vertebrate genome assemblies. Gigascience 4:10

Hunt M, Kikuchi T, Sanders M, Newbold C, Berriman M, Otto TD (2013) REAPR: a universal tool for genome assembly evaluation. Genome Biol 14:R47

Kelley DR, Schatz MC, Salzberg SL (2010) Quake: quality-aware detection and correction of sequencing errors. Genome Biol 11:R116

Lander ES, Waterman MS (1988) Genomic mapping by fingerprinting random clones: a mathematical analysis. Genomics 2:231–239

Leinonen R, Akhtar R, Birney E, Bower L, Cerdeno-Tárraga A, Cheng Y, Cleland I, Faruque N, Goodgame N, Gibson R et al (2011) The european nucleotide archive. Nucleic Acids Res 39:D28–D31

Li H, Homer N (2010) A survey of sequence alignment algorithms for next-generation sequencing. Brief Bioinform 11:473–483

Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R (2009) The sequence alignment/map format and SAMtools. Bioinformatics 25:2078–2079

Li R, Fan W, Tian G, Zhu H, He L, Cai J, Huang Q, Cai Q, Li B, Bai Y et al (2010) The sequence and de novo assembly of the giant panda genome. Nature 463:311–317

Lindblad-Toh K, Wade CM, Mikkelsen TS, Karlsson EK, Jaffe DB, Kamal M, Clamp M, Chang JL, Kulbokas EJ, Zody MC et al (2005) Genome sequence, comparative analysis and haplotype structure of the domestic dog. Nature 438:803–819

Loman NJ, Quick J, Simpson JT (2015) A complete bacterial genome assembled de novo using only nanopore sequencing data. Nat Methods 12:733

Luo R, Liu B, Xie Y, Li Z, Huang W, Yuan J, He G, Chen Y, Pan Q, Liu Y et al (2012) SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler. Gigascience 1:18

Marçais G, Kingsford C (2011) A fast, lock-free approach for efficient parallel counting of occurrences of k-mers. Bioinformatics 27:764–770

Marcet-Houben M, Ballester A-R, de la Fuente B, Harries E, Marcos JF, González-Candelas L, Gabaldón T (2012) Genome sequence of the necrotrophic fungus Penicillium digitatum, the main postharvest pathogen of citrus. BMC Genomics 13:646

McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M et al (2010) The genome analysis toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome Res 20:1297–1303

Miller JR, Koren S, Sutton G (2010) Assembly algorithms for next-generation sequencing data. Genomics 95:315–327

Neale DB, Wegrzyn JL, Stevens KA, Zimin AV, Puiu D, Crepeau MW, Cardeno C, Koriabine M, Holtz-Morris AE, Liechty JD et al (2014) Decoding the massive genome of loblolly pine using haploid DNA and novel assembly strategies. Genome Biol 15:R59

Parra G, Bradnam K, Korf I (2007) CEGMA: a pipeline to accurately annotate core genes in eukaryotic genomes. Bioinformatics 23:1061–1067

Pryszcz LP, Németh T, Gácser A, Gabaldón T (2014) Genome comparison of Candida orthopsilosis clinical strains reveals the existence of hybrids between two distinct subspecies. Genome Biol Evol 6:1069–1078

Reuter JA, Spacek DV, Snyder MP (2015) High-throughput sequencing technologies. Mol Cell 58:586–597

Richards S, Murali SC (2015) Best practices in insect genome sequencing: what works and what doesn't. Curr Opin Insect Sci 7:1–7

Schmieder R, Edwards R (2011) Quality control and preprocessing of metagenomic datasets. Bioinformatics 27:863–864

Simão FA, Waterhouse RM, Ioannidis P, Kriventseva EV, Zdobnov EM (2015) BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. Bioinformatics 31:3210

Simpson JT (2014) Exploring genome characteristics and sequence quality without a reference. Bioinformatics 30:1228–1235

Simpson JT, Durbin R (2012) Efficient de novo assembly of large genomes using compressed data structures. Genome Res 22:549–556

Simpson JT, Pop M (2015) The theory and practice of genome sequence assembly. Annu Rev Genomics Hum Genet 16:153

Sims D, Sudbery I, Ilott NE, Heger A, Ponting CP (2014) Sequencing depth and coverage: key considerations in genomic analyses. Nat Rev Genet 15:121–132

Tang H, Lyons E, Town CD (2015) Optical mapping in plant comparative genomics. Gigascience 4:3

Van Dijk EL, Jaszczyszyn Y, Thermes C (2014) Library preparation methods for next-generation sequencing: tone down the bias. Exp Cell Res 322:12–20

Vezzi F, Narzisi G, Mishra B (2012) Feature-by-feature--evaluating de novo sequence assembly. PLoS One 7:e31002

Xi R, Kim T-M, Park PJ (2010) Detecting structural variations in the human genome using next generation sequencing. Brief Funct Genomics 9:405–415

http://www.springer.com/978-3-319-31348-1

Field Guidelines for Genetic Experimental Designs in High-Throughput Sequencing

Aransay, A.M.; Lavín Trueba, J.L. (Eds.)

2016, XI, 399 p. 72 illus., 42 illus. in color., Hardcover
ISBN: 978-3-319-31348-1