Making sense of genomes of parasitic worms: Tackling bioinformatic challenges

Pasi K. Korhonen*, Neil D. Young, Robin B. Gasser*

Faculty of Veterinary and Agricultural Sciences, The University of Melbourne, Parkville, Victoria 3010, Australia

A B S T R A C T

Billions of people and animals are infected with parasitic worms (helminths). Many of these worms cause diseases that have a major socioeconomic impact worldwide, and are challenging to control because existing treatment methods are often inadequate. There is, therefore, a need to work toward developing new intervention methods, built on a sound understanding of parasitic worms at molecular level, the relationships that they have with their animal hosts, and/or the diseases that they cause. Decoding the genomes and transcriptomes of these parasites brings us a step closer to this goal. The key focus of this article is to critically review and discuss bioinformatic tools used for the assembly and annotation of these genomes and transcriptomes, as well as various post-genomic analyses of transcription profiles, biological pathways, synteny, phylogeny, biogeography and the prediction and prioritisation of drug target candidates. Bioinformatic pipelines implemented and established recently provide practical and efficient tools for the assembly and annotation of genomes of parasitic worms, and will be applicable to a wide range of other parasites and eukaryotic organisms. Future research will need to assess the utility of long-read sequence data sets for enhanced genomic assemblies, and develop improved algorithms for gene prediction and post-genomic analyses, to enable comprehensive systems biology explorations of parasitic organisms.

* Corresponding author. Tel.: +61-3-97312283. 
E-mail address: robinbg@unimelb.edu.au (R. B. Gasser) or pasi.korhonen@unimelb.edu.au (P. K. Korhonen)
1. Introduction – importance of nematode genomics

Parasitic worms (helminths) of humans and animals cause diseases of major socioeconomic importance globally (Fenwick, 2012, Utzinger, 2012). In humans, the global disease burden caused by worms in disability-adjusted life years (DALYs) is comparable to that caused by malaria and tuberculosis (Hotez et al., 2009, Hotez et al., 2006), and represents ~1% of DALYs worldwide (Murray et al., 2012). These diseases can be transmitted via water, food, soil or other means, and a considerable number of them are zoonotic (Bethony et al., 2006, Pozio, 2003). In livestock, the annual economic losses due to death, poor health and reduced productivity caused by roundworms (nematodes), such as *Haemonchus* spp., are estimated at billions of dollars per annum worldwide (Newton and Munn, 1999, Preston et al., 2015). In Australia alone, endemic and gastrointestinal worms of sheep and cattle cause economic losses of 410 to 530 million dollars (Lane et al., 2015, Sackett et al., 2006).

As an example of an important zoonotic nematode, the intestinal nematode *Toxocara* has major health importance in carnivores (dogs or cats), humans and other hosts. Indeed, millions of people are infected with *T. canis*, and the worldwide prevalence of serum antibodies against *Toxocara* in humans ranges from 7% to 82% (Maizels, 2013). Also foodborne nematodes, such as *Trichinella*, are of human and animal health importance (Pozio and Zarlenga, 2013). The prevalence of trichinellosis in humans is estimated at 11 million (Dupouy-Camet, 2000), and the incidence (annual) is estimated 2,500-10,000 (Gottstein et al., 2009, Murrell and Pozio, 2011, Pozio, 2007). In addition, *T. spiralis* causes a major economic burden (> $1.5 billion per annum in Europe and the USA) associated with preventative measures in human food chain (Murrell and Pozio, 2000). Clearly, these examples illustrate how parasitic nematodes impact on human and animal health.

Despite current knowledge of parasitic nematodes and the diseases that they cause (Anderson, 2001), little is known about their molecular biology, genetics and evolution; how they invade and establish in the host, cause disease, interact, suppress and evade their host’s immune responses; and how resistance develops and evolves against current treatments. For instance, although resistance to most commonly used anthelmintics is widespread (Kaplan and Vidyashankar, 2012) and seriously compromises nematode control, it is not possible to accurately estimate the prevalence of drug resistant genotypes of nematodes (Howell et al., 2008, Kaplan and Vidyashankar, 2012, Sczesny-Moraes et al., 2010). The related annual cost of anthelmintic treatment worldwide is estimated at $3 billion, and this cost, together with the production losses caused by nematode infections/diseases, has stimulated research towards developing improved intervention methods. Thus far, the development of vaccines against parasitic worms has been relatively unsuccessful (Knox, 2011, 2013), and the development of novel anthelmintic drugs has been relatively slow, with emodepside (patent WO 1997/02256) (Harder and von Samson-Himmelstjerna, 2001), monepantel (patent WO2005/44784) (Kaminsky et al., 2008) and derquantel (patent WO1997/03988) (Lee et al., 2001) being new anthelmintics developed in the past 20 years.

There is a clear imperative to gain an improved understanding of the biology of socioeconomically important parasitic nematodes at the molecular level, in order to underpin new methods of treatment, diagnosis and control. Advanced genomic and bioinformatic technologies should provide useful tools to investigate nematode genomes and transcriptomes, to elucidate the global “molecular landscapes” of nematodes as well as key aspects of their biochemistry and physiology. In spite of technological advances, there have been considerable challenges in the sequencing and annotation of nematode genomes and post-genomic analyses of genomic data sets. In the present article, we (i) critically appraise genome sequencing technologies as well as algorithms and approaches used for genome assembly and annotation (highlighting their advantages and disadvantages), (ii) summarize developments in sequencing and drafting of genomes of parasitic nematodes using various bioinformatics tools; (iii) describe enhanced pipelines constructed and employed in our laboratory, which have proven to be practical and efficient for the assembly
and annotation of worm genomes, and, finally, (vi) provide a perspective on future research toward improved genomics/bioinformatics of parasitic nematodes.

2. Appraisal of genomic and bioinformatics approaches

In terms of bioinformatics, a draft genome project for a parasite encompasses genome assembly and annotation (Stein, 2001, Yandell and Ence, 2012). The success of a genome assembly depends on the sequencing technology and assembly algorithm(s) used. However, typically, short-read, shotgun assemblies do not lead to complete chromosomal assemblies of eukaryote genomes, mainly because of the difficulties in the resolution of repeat regions (Alkan et al., 2011), and a lack of uniform read coverage usually associated with the suboptimal quality of genomic DNA and of library construction (Gasser, 2013, Gasser et al., 1993, Gasser et al., 2006, Raghunathan et al., 2005).

Genome annotation can be divided into structural (computational) and functional (annotation) phases (Ekblom and Wolf, 2014, Stein, 2001). During structural annotation, genomic features, such as genes, repeat regions and RNA species, are predicted, and their composition and location in the genome recorded. For gene prediction, usually the results of both ab initio and evidence-based predictions (from mRNA, cDNA or proteomic data) are combined (Yandell and Ence, 2012). In contrast, functional annotation, also called function prediction, infers and assigns a function(s) to a gene or genome element (Stein, 2001). This process usually predicts the function of proteins, RNA species and repeat regions (Stein, 2001). The functions of the predicted genome elements, particularly genes, are predicted using similarity searches, structural comparisons, phylogenetic approaches, genetic interaction networks and/or machine learning approaches (Erdin et al., 2011). Here, we concentrate on reviewing the computational prediction of the functions of protein-encoding genes.

The quality of a genome assembly is central to gene predictions (Alkan, Sajjadian, 2011), and the quality of gene prediction is critical for both the functional annotation (Sleator, 2010) and the identification of evolutionarily distant homologous genes. An improvement in the recognition of distant homologs improves the success and reduces the cost and time required for post-genomic analyses. For instance, the quality of the computational prediction of drug targets can have a direct influence on the success of expensive and time-consuming biochemical assays for target validation (Keiser et al., 2009).

The following sub-sections discuss commonly used algorithms and recent methods employed for genome assembly, gene prediction and functional annotation of predicted proteins. Genomic elements, such as RNA molecules (Kang and Friedlander, 2015, Ma et al., 2012, Nawrocki and Eddy, 2013, Stadler, 2014), gene regulatory elements (Hecker et al., 2009, Wasserman and Sandelin, 2004) and repeat regions (Diethard, 1993, McClintock, 1950, Saha et al., 2008, Wicker et al., 2007) as well as some methods integrated into automated software pipelines, such as those employed for the pre-processing of reads (Del Fabbro et al., 2013), scaffolding (Hunt et al., 2014) and the removal of putative contamination (Schmieder and Edwards, 2011), are not covered here.

2.1. Genomic sequencing and assembly

The overall success of genome sequencing depends on a multitude of parameters, such as the quality of genomic DNA and DNA library construction, the cost per nucleotide sequenced, the amount of sequence data produced per time unit, the quality and lengths of fragments sequenced, and the software employed for the analysis of sequence/read data. In this sub-section, both sequencing technologies and algorithms used for genomic assembly are reviewed.
2.1.1. Sequencing

Genome sequencing strategies can be divided into whole-genome and hierarchical shotgun sequencing approaches (Lander et al., 2001). For whole-genome shotgun sequencing (Staden, 1979), total genomic DNA is first fragmented. Fragments are then sequenced, resulting in reads, and these reads are computationally assembled into a draft genome. Using the hierarchical shotgun sequencing approach, the genome is first divided into segments, from which cloned vector libraries, such as bacterial artificial chromosome (BAC) (Shizuya et al., 1992), are prepared. Shotgun sequencing is then applied separately to individual BAC clones, consequently dividing the assembly task into subregions of the genome. Resultant BAC clone assemblies are then computationally combined into consensus genomic sequences. Typically, whole-genome shotgun sequencing is applicable to genomes with relatively low repeat contents, whereas hierarchical shotgun sequencing can be applied to genomes with relatively high repeat contents (Beier et al., 2015, Lander, Linton, 2001).

Historically, dideoxy sequencing (Sanger and Coulson, 1975) was applied to sequence DNA, and is often referred to as a first-generation sequencing method. Sanger sequencing was used for the BAC-based hierarchical shotgun sequencing employed in the human genome project, which took ten years to complete (Lander, Linton, 2001). Parallel to this public, federally funded human genome project (based on years of solid science), J. Craig Venter and his colleagues undertook a competing human genome assembly (Venter et al., 2001), based on the use of the paradigm shifting whole-genome shotgun method. Despite advances achieved in the original human genome project and the ability to produce high quality read data (length $\leq 900$ bp) (Morozova and Marra, 2008), Sanger sequencing is a slow and costly approach for genomic sequencing.

Consequently, next-generation sequencing (NGS), also called massively parallel or second-generation sequencing technologies were introduced. Successfully employed second-generation technologies include Genome Analyser (GA) by Illumina, Supported Oligonucleotide Ligation and Detection (SOLiD) by Applied Biosystems (ABI) and Genome Sequencer (GS) by Roche. Since their launch, these platforms have evolved significantly, in terms of read length, throughput and cost of sequencing (Table 1). The common feature of these technologies is massively parallel, iterative sequencing using PCR-amplified templates in wash-and-scan flow cells (Holt and Jones, 2008), producing read lengths of 35-700 bp. Both GS and SOLiD use emulsion PCR technology for template preparation, whereas GA uses a bridge-PCR method (Holt and Jones, 2008) for the same purpose. To capture the sequence, GS employs pyrosequencing (Nyren et al., 1993, Ronaghi et al., 1996), whereas GA uses a four-colour sequencing-by-synthesis method, with reversible terminators (Holt and Jones, 2008); finally, SOLiD uses oligonucleotide ligation, with two base-encoding and four-colour imaging (Barany and Gelfand, 1991, Shendure et al., 2005). The features of these platforms (Henson et al., 2012, Holt and Jones, 2008) are summarised in Table 1.

The main advantages of NGS sequencers are the speed and the cost of sequencing – a human genome can now be sequenced at a cost of $< 2,000$ (Hayden, 2014, Reuter et al., 2015) (http://www.illumina.com/systems/hiseq-x-sequencing-system/system.html). However, limitations are the high cost of the sequencers themselves and the short length of sequence reads. These short-read data sets typically result in fragmented genome assemblies, which is particularly evident in eukaryotic genomes replete with long, highly repetitive repeat regions. Using short-read data only, the genome assembly algorithms can neither resolve long repeat regions nor extend contigs to chromosomal completeness due to the multitude of very similar anchor sequences in repeat regions. To partially resolve this problem, paired-end (PE) read libraries (Roach et al., 1995) and mate pair (MP) libraries (Venter, Adams, 2001) and Chicago libraries by Dovetail are supported by all NGS platforms. Data from these libraries bridge genomic regions using a pair of reads relating to a known insert size between the pairs. Typically, insert size range from 20 bp to 800 bp for PE reads and from 2 kb to 40 kb for MP libraries.
Some NGS platforms, such as Illumina, are in the process of resolving these problems. However, third-generation sequencers are likely to supersede the second-generation methods in the next years, and promise to sequence long reads (≤ 100 kb) from single DNA molecules from tiny amounts of genomic DNA template at reduced throughput time and cost of sequencing (and the expense of purchasing sequencers). Currently, available third-generation sequencing is still costly and produces error-prone, long reads (error rate: 2-15%) (Artyomenko et al., 2014, Quail et al., 2012). These sequencers include Ion Torrent from Life Technologies (Rothberg et al., 2011) and PacBio real-time sequencer (RS) from Pacific BioSciences (Flusberg et al., 2010). Ion Torrent can produce reads of 400 bp in length by using a complementary metal–oxide–semiconductor (CMOS) integrated circuit, named ion chip, to recognize a pH change caused by the release of hydronium ion (H$_3$O$^+$) side-products from a polymerase attaching nucleotides to the template sequence. Ion Torrent might be applicable to single molecule sequencing, but, currently, the DNA template is amplified by emulsion PCR. The PacBio RS platform is based on the use of a single-molecule real-time (SMRT) sequencing cell and produces reads of > 4,500 bp. Presently, the cost of PacBio sequencing is high, and the error rate of every single pass sequence is reported to be high (13-15%) (Artyomenko, Mangul, 2014, Quail, Smith, 2012). However, at a coverage of > 100, sequencing can be almost error-free, and the technology can be used to assemble genomic regions that cannot be resolved using NGS platforms. Importantly, used in combination with short read data, this technology can achieve improved genomic assemblies (Koren et al., 2012).

Nanopore-based methods, which represent third-generation technology, can sequence a single DNA molecule directly, without the need for an intervening PCR amplification or chemical labelling step, or the need for optical instrumentation to identify the chemical label, and have potential for high-throughput, low-cost sequencing. The principle of such methods are that a molecule is translocated through a pore, driven either by a salt gradient, an electric field or by an enzyme, thereby modulating measurable ionic flow (Niedringhaus et al., 2011, Schneider and Dekker, 2012). This flow has particular characteristics for each base, and allows the sequence in the DNA strand to be recorded. Currently, technical challenges include blockages in the pores, enzyme dissociation from the nucleic acid strand and inaccuracy of base recognition (Cracknell et al., 2013, Goodwin et al., 2015, Schneider and Dekker, 2012). To date, none of the Nanopore methods have been commercialised. However, Oxford Nanopore™ has successfully distributed the Minion™ platform to selected customers for evaluation in an early access program. This platform has been successfully used to assemble the bacterial genome of Escherichia coli (Loman et al., 2015). The error rate (18-35%) for single-pass DNA sequences (Goodwin, Gurtowski, 2015, Jain et al., 2015) is at least similar to that of PacBio, but high sequence coverage can help to overcome this problem.

Second-generation sequencers, such as HiSeq and MiSeq from Illumina, and GS Junior from Roche (Henson, Tischler, 2012), are presently competing with third-generation methods through enhancements, such as longer read lengths and reduced sequencing cost (Table 2). Recently, Illumina also launched a Moleculo Long Read service, producing data that can achieve localised assemblies of ≤ 10 kb in length (Voskoboynik et al., 2013). Other technologies, such as direct imaging of DNA, are also being developed, but are not yet in routine use. Theoretical limits regarding the achievable accuracy of sequencing technologies exist (Strippoli et al., 2005). Technology-specific sequencing errors and biases, such as GC-bias (Ross et al., 2013), substitution errors in Illumina, poor resolution of long homopolymers in pyrosequencing and Ion Torrent (Reuter, Spacek, 2015), need to be accounted for in genome assembly algorithms.

2.1.2. Assembly

The data produced by the sequencers are short nucleotide sequence fragments, usually accompanied by a quality score for each nucleotide. These fragments are called sequence reads or reads, and the computational challenge is to correctly assemble these reads together to form
multiple contiguous sequences (contigs), sequences composed of overlapping reads combining the nucleotide sequence information contained in the reads to form long contiguous nucleotide sequences, with the goal of resolving the sequences of complete chromosomes. Assembly algorithms follow the constraints posed by sequencing technologies, and can be roughly divided into overlap, layout, consensus (OLC) (first-generation), de Bruijn graph-based (second-generation) and greedy algorithms (Table 3).

**Assembly using OLC algorithms.** The assembly algorithms used for data produced by first-generation sequencers are composed of overlap, layout and consensus phases (Kececioglu, 1991, Myers, 1995, Peltola et al., 1984) (see Table 3). In brief, pairwise overlaps are first approximated and converted into a graph. Through maximising a cost function, this graph is then used to create a consistent layout representing contiguous overlapping reads. Finally, these reads are aligned, and the consensus sequence is established (Kececioglu, 1991).

To implement the overlap phase, all reads should aligned in pairwise manner, but this is not computationally feasible, for instance, using traditional Needleman-Wunsch and Smith-Waterman dynamic programming alignment algorithms (Needleman and Wunsch, 1970, Smith and Waterman, 1981). To circumvent this issue of all versus all pairwise alignments, reads are usually first transformed into k-mers (= all ordered sub-sequences of length k contained in all reads). To identify the high-confidence pairwise matches, the alignment can then be applied to a subset of reads with identical k-mers. This method substantially reduces the number of computed, pairwise alignments needed to create an overlap graph. The algorithms resolving the edges (= different orientations among overlapping reads) in the overlap graph strive to maximise a non-deterministic polynomial-time (NP) complete (Garey and Johnson, 1979) cost function (not resolvable exhaustively for large data), to achieve optimal order of overlaps (Myers, 1995). Therefore, an optimal layout can only be approximated using heuristic methods. Finally, the reads are multi-aligned to the optimised layout, and consensus sequences are then established by majority voting from these alignments.

Examples of algorithms that implement the OLC scheme for Sanger sequence data include SEQAID (Peltola, Söderlund, 1984), TIGR (Sutton et al., 1995), CAP3 (Huang and Madan, 1999) and Celera (Myers et al., 2000). SEQAID was the first algorithm to implement the OLC scheme, TIGR implements k-mer based pairwise alignment to speed up the overlap phase, and CAP3 supports both paired-end read data (forward-reverse constraints) and PHRED quality score-based read filtering (Ewing and Green, 1998, Ewing et al., 1998). Celera introduced the unitig concept, in which high confidence reads are first assembled into contigs (called unitigs), which are then used for the subsequent assembly.

**Assembly using greedy and de Bruijn graph-based algorithms.** Compared with first-generation sequencers, second-generation sequencers produce massive amounts of read data, which led to the development of new assembly algorithms (Table 3). Large data creates challenges for OLC-based assemblers, but programs such as Newbler (developed by 454 Life Sciences) (Margulies et al., 2005) and CABOG (Miller et al., 2008) have resolved this issue for pyrosequence (GS) data (cf. Table 2). Newbler optimises memory management and uses the unitig concept, whereas CABOG extends the implementation of the Celera assembler (Myers, Sutton, 2000) to cover also GS data, in addition to Celera-supported Sanger data.

The first programs to support NGS data were the greedy assembler implementations SSAKE (Warren et al., 2007) and SHARCGS (Dohm et al., 2007), in which the contigs are iteratively extended by best-matching reads. A problem with such greedy approaches is that the best match of a read in one contig might lead to a better assembly if this read were assigned to another contig. SSAKE was the first to support the assembly of short-read data, and SHARCGS the first to filter, identify and remove base-call errors.
To avoid computationally demanding pairwise and multiple alignment phases of OLC-based implementations, a graph theory-based Eulerian path fragment assembly method (Pevzner et al., 2001) was soon applied to second-generation assemblers. The crucial algorithmic improvement of this method was the linear-time computation of the Eulerian path (Fleischner, 1990), while nondeterministic polynomial time (NP-) complete problems remain for the OLC-based methods (Pevzner, Tang, 2001). In short, instead of representing reads in an overlap-graph, these assembly algorithms represent all k-mers from these reads as a de Bruijn graph. This graph is then processed and converted to a Eulerian graph, in which Eulerian walk (crossing each edge once) reconstructs the genome (i.e. Eulerian path). The use of the de Bruijn graph reduces the algorithmic phases to graph construction, graph simplification and error removal; this approach does not contain the NP-complete layout phase of the OLC method.

The assembled contigs can then be readily inferred from the de Bruijn graphs and, consequently, do not require a multiple-alignment. Compared with the OLC method, a limitation of the de Bruijn graph-based method is the artificial reduction of read length to the selected size of the k-mer and a consequent compromised resolution of similar sequences, such as repeat regions. To overcome this limitation of de Bruijn graph-based assemblers, the OLC-based String Graph Assembler (SGA) was developed (Simpson and Durbin, 2012), in which the FM-index (Ferragina et al., 2010) (compressed Burrows Wheeler indexing) is employed to reduce memory usage in string graphs (Myers, 2005) for the processing of short read data.

An early study (Idury and Waterman, 1995) described the first Eulerian path-based fragment assembly algorithm using de Bruijn graphs. This algorithm was improved in the program EULER (Pevzner, Tang, 2001), which embeds repeat handling and nucleotide error-correction in its implementation. Subsequent, successful implementations were through the programs Velvet (Zerbino and Birney, 2008), ABySS (Simpson et al., 2009), ALLPATHS (Butler et al., 2008) and SOAPdenovo (Luo et al., 2012). In de Bruijn graph implementations, to accommodate the artificial shortening of reads, reads are probed against a k-mer graph (read-threading) to validate each subgraph. Paired-end and mate pair reads are similarly threaded, and subgraphs are fitted for each pair. In the latest implementation of algorithms utilising de Bruijn graphs, namely SPAdes (Bankevich et al., 2012), k-bimer adjustments allow for variation in insert size and paired assembly graphs, which were inspired by paired de Bruijn graphs (PDBG) (Medvedev et al., 2011). The advantage of this method over traditional de Bruijn graphs is that it allows the insert size to be accurately estimated for most of k-bimers.

Assembly of repeat regions. The assembly of repeat regions remains problematic for both first- and second-generation assembly algorithms. In theory, the accurate assembly of a repeat region becomes challenging when the length of the repeat longer or equal than the length of the reads (Medvedev, Pham, 2011). To partially overcome this challenge, mate pairs or jumping pairs (paired reads with long insert sizes), potentially spanning repeat regions, were introduced (Weber and Myers, 1997). To completely resolve repeat regions, read lengths must match the repeat lengths.

Third-generation assemblers. For third-generation sequencers, which are capable of producing significantly longer reads than first and second-generation sequencers, assembler implementations are moving toward improved read pre-processing to eliminate (excessive) errors from sequence data. This pre-processing of data results in relatively small numbers of long sequences of high quality, thus inspiring the reuse of OLC-based algorithms to avoid the artificial shortening of read lengths encountered using de Bruijn graph-based assemblers. Therefore, this approach has the capacity to assemble and resolve long repeat regions. The first assembler to support third-generation data is the single molecule assembler PBcR (Koren, Schatz, 2012), in which high quality, short reads produced by second-generation sequencers (see Sub-section 2.1.1) are used to resolve error-prone long read data originating from third-generation sequencers. The PBcR assembler is based on
the OLC-based CABOG Celera implementation (Miller, Delcher, 2008) (Table 3). The use of a highly parallelizable SGA should enable rapid and effective assembly of large genomes from short and long read data sets. In the future, an improvement in the quality of data produced by third-generation sequencers might eliminate the need for second-generation sequencing.

2.2. Gene prediction

Assembled genomic sequence data are of little biological value without further bioinformatic processing. Gene prediction, also called structural annotation or gene finding, aims to identify structural elements in a genomic sequence that represent a gene. In eukaryotes, these elements consist of 5'- and 3'- untranslated regions (UTRs), exons and introns and their splice sites, splicing variants and respective protein-coding sequences (CDS) of a gene, in which UTR-regions may span multiple exons (Fig. 1). Gene regulatory regions may be part of the structural gene annotation, but they are not the focus in annotations in the present review article.

Gene prediction methods can be intrinsic (ab initio) or extrinsic (similarity search) (Do and Choi, 2006). Intrinsic methods use statistical patterns to identify gene regions in the genomic sequence, whereas extrinsic techniques align transcriptomic, protein sequence and/or other evidence data sets for gene prediction. Predicted gene element data are represented by a unified General Feature Format (GFF) specification (http://www.sanger.ac.uk/resources/software/gff/spec.html). Unfortunately, this format often has different interpretations, depending on the gene prediction software employed, thus requiring implementations of GFF parsers (i.e. software program analysing the syntax of a GFF file).

Intrinsic methods. Intrinsic (ab initio) gene prediction methods are divided into consensus (signal sensor) and non-consensus (content sensor) categories (Sleator, 2010). Consensus-based methods predict known nucleotide patterns (signals) in gene elements, such as splice sites, start and stop codons, and the Kozak consensus sequence (relates to the initiation of translation) (Sleator, 2010, Wilson and Hunt, 2008). Methods utilising the Weighted Matrix Method (WMM), called Position Weigh Matrix (PWM) (Staden, 1984), Weighed Array Model (WAM) (Zhang and Marr, 1993), Maximal Dependence Decomposition (MDD) (Burge and Karlin, 1997) and Windowed Weight Array Model (WWAM) (Burge and Karlin, 1997), are used to recognise these signals. In brief, WMM calculates the signal probability and assumes independency among the nucleotides; WAM assumes dependencies between adjacent nucleotides; MDD implements a decision tree of WMMs, extending the dependency considerations across non-adjacent nucleotides; and WWAM assumes dependencies across three consecutive nucleotides, and averages related conditional probabilities among five consecutive nucleotides.

In contrast, non-consensus methods use nucleotide composition (content) to recognise gene elements and sequence areas, such as coding and non-coding regions. Hidden Markov Models (HMM) using hexamer sequence composition has proven to be the most successful discriminator between these two regions, when predicting nucleotide-by-nucleotide (Fickett and Tung, 1992). However, to extend the prediction capability of this single nucleotide approach to versatile gene elements and even complete gene structures, the prediction algorithms are enhanced with three-period, fifth order generalized HMMs (GHMM) (Kulp et al., 1996), in which hexamers are used together with the built-in knowledge of codon structure, to ensure the preservation of the reading frame. For instance, the programs GENSCAN (Burge and Karlin, 1997) and GeneMark-ES (Lomsadze et al., 2005, Ter-Hovhannisyan et al., 2008) use a GHMM-based three-period, fifth order Markov chain model (Borodovsky and McIninch, 1993); GENSCAN also uses MDD as a signal sensor. To further improve predictions, interpolated Markov Model (IMM), in which Markov Models of different orders are interpolated (Salzberg et al., 1999), is used in gene finders such as AUGUSTUS (Stanke, 2004, Stanke and Waack, 2003) and GlimmerHMM (Majoros et al., 2004). AUGUSTUS uses WWAM for signal prediction, but also has a dedicated HMM model for intron
prediction, whereas GlimmerHMM uses the program GeneSplicer (Pertea et al., 2001) to improve splice site prediction. The program SNAP (Korf, 2004) combines signal predictions (WWM and WAM) with GHMM content predictions by supporting genome-specific training to create HMM models, whereas GeneID (Parra et al., 2000) combines signal (PWM) and exon content predictions (GHMM) using dynamic programming (Guigo, 1998), and scores predictions based on log-likelihood values of predicted gene elements. GeneMark-ES is an unsupervised gene prediction algorithm for eukaryotic genomes, and is based on Hidden Semi-Markov Model (HSMM) (Rabiner, 1989), in which model parameters are estimated iteratively using Viterbi training and Gibbs Recursive Sampler (Thompson et al., 2003). In general, HMMs tend not to have enough statistical power to reliably predict short exons with relatively modest numbers of nucleotides (Picardi and Pesole, 2010).

Usually, sophisticated HMM-based content sensors are used to predict the complete structure of a gene (Majoros, Pertea, 2004, Stanke, 2004, Stanke and Waack, 2003), and signal sensors are only used to estimate the parameters for the prediction models employed in content sensors (Korf, 2004, Parra, Blanco, 2000). However, signal sensors alone can be used to predict particular gene elements, and some predictors, such as GeneID and mGene (Schweikert et al., 2009), apply signal sensor methods to genomic sequence data and then combine the resultant predictions with those of content sensors to produce a final, consensus prediction.

Ab initio prediction algorithms (cf. Table 4) have been enhanced using information from syntenic regions among multiple genomes. Algorithms such as TWINSCAN (Korf et al., 2001) and N-SCAN (Gross and Brent, 2006) are based on GHMM, but also consider syntenic sequence conservation. While TWINSCAN can accommodate two genomes, N-SCAN can handle multiple genomes. The use of a high quality reference genome in TWINSCAN can produce results that are comparable with those achievable using multiple genomes in N-SCAN (Gross and Brent, 2005). When using these tools, it is advisable to employ genomes from taxonomically closely related species (Boffelli et al., 2003). TWINSCAN is capable of creating superior results compared with predictors in which only a single genome is used, and N-SCAN can produce superior results compared with TWINSCAN. However, a limitation is that N-SCAN is dependent on programs such as BLASTZ (Schwartz et al., 2003) or MULTIZ (Blanchette et al., 2004) for the multiple alignment at the beginning of the prediction process. Producing this alignment can be very time consuming and may not succeed if highly fragmented de novo-assembled genomes are employed, thus limiting the applicability of this sophisticated gene prediction method.

To create functional prediction models, ab initio predictors have to be trained using reliable training data sets, which are specific to each genome. If training data are not available, parameter values for prediction models can be estimated by predicting genes first using sub-optimal parameter values, and then by recalculating new values using the predicted genes (Korf, 2004). Suboptimal parameter values can be copied from prediction models for closely related species. In addition, such values can be inferred from the structures of Core Eukaryotic Genes (CEG) using the program CEGMA (Parra et al., 2007) and/or from unsupervised gene prediction programs, such as GeneMark.

Extrinsic methods. In contrast to intrinsic techniques, extrinsic methods (Table 5) do not infer or predict gene structures from the genomic sequence. Instead, for these methods, evidence data (i.e. transcriptomic and protein sequences) are aligned to the genomic sequence, and genes are predicted based on the alignment success. Clearly, full-length transcriptomic (cDNA) sequences are the “gold standard” for gene prediction and provide a means of accurately predicting a gene structure, but sometimes their availability is limited (Brent, 2005). Therefore, fragmented transcriptomic sequences, assembled from RNA-seq data, and protein sequences of taxonomically closely related species can be used. The mRNA sequences are typically derived from the species under
investigation and thus match the genomic sequence (“native alignment”), whereas the protein sequences of closely related species are not expected to match exactly conceptually translated genomic sequences (trans alignment).

The accuracy of these alignments, the fragmented nature of evidence (mRNA or protein sequence) data, and splice variants from genes are challenges for extrinsic methods. Many of the alignment algorithms, such as EST_GENOME (Mott, 1997), AAT (Huang et al., 1997) and Exonerate (Slater and Birney, 2005), use BLAST (Altschul et al., 1997) to gain seed alignments. These seeds are then extended using different dynamic programming variants (Needleman and Wunsch, 1970, Smith and Waterman, 1981). Pair HMM (Kent and Zahler, 2000) aligners, such as GeneWise (Birney et al., 2004b) and Pairagon (Lu et al., 2009), align evidence data accurately across exons and introns, but alignment speed is suboptimal, even when the Viterbi algorithms used in HMM are replaced with heuristic dynamic programming algorithms. Of these aligners, Pairagon combines WMMs to recognize splice sites. Exonerate processes data relatively rapidly, is capable of aligning both nucleotide and protein sequences, and is, thus, widely used for alignment tasks.

**Combining gene predictions from multiple sources.** The consensus gene predictions, combined from both ab initio and evidence-based gene structure predictions from multiple sources (cf. Table 6), provide the most confident results (Coghlan et al., 2008). The first attempts to combine the prediction data from multiple sources were implemented in the program COMBINER (Allen et al., 2004). Here, consensus prediction algorithms are based on both linear and statistical combinations of the prediction data from multiple sources. JIGSAW (Allen and Salzberg, 2005), the successor of COMBINER, has internal support for ab initio predictions with GHMM, and expresses external evidence of structural elements of a gene using feature vectors. These vectors give a weighting coefficient to each prediction source, and dynamic programming (combined with decision trees) is used to establish optimal gene structures. Other frameworks include Ensembl (Birney et al., 2004a), EVIGAN (Liu et al., 2008), GLEAN (Elsik et al., 2007) and MAKER2 (Cantarel et al., 2008, Holt and Yandell, 2011). Ensembl prefers evidence-based over ab initio predictions, thus achieving high quality annotations at the cost of sensitivity. EVIGAN predicts gene structures using Dynamic Bayes Networks (Murphy, 2002), for which the parameters are estimated with the maximum likelihood (ML) method (Savage, 1976), and GLEAN uses latent class analysis (LCA) algorithm to give consensus predictions (Goodman, 1974, Lazarsfeld and Henry, 1968, McCutcheon, 1987). In GLEAN-based LCA implementation, latent classes (gene structures) are predicted from multivariate categorical data (gene structural elements). Interestingly, MAKER2 uses Annotation Edit Distance (AED) (Eilbeck et al., 2009) to estimate the share of evidence data for the consensus prediction, therefore offering the advantage of estimating the reliability of any prediction. Upgraded versions of ab initio programs, such as GENSCAN, TWINSCAN, NSCAN and AUGUSTUS, can utilise evidence data for gene predictions. These new versions are GenomeScan (Yeh et al., 2001), TWINSCAN-EST (Wei and Brent, 2006), NSCAN-EST (Wei and Brent, 2006) and AUGUSTUS+ (Stanke et al., 2006b), respectively.

**Machine learning methods.** The limitations of HMM-based methods include the assumption of conditional independence, non-discriminative training and explicit feature representation, often leading to ambiguous prediction results (Altun et al., 2003). This means that kernel-based discriminative machine learning methods, such as Support Vector Machine (SVM) (Cortes and Vapnik, 1995), cannot be used because features cannot be represented in non-linear combinations. However, a method called Hidden Markov SVM (HM-SVM), which combines both Viterbi algorithm (used in HMM) and discriminative learning of SVM, has been proposed (Altun, Tsochantaridis, 2003). Moreover, Hidden semi-Markov SVM (Ratsch and Sonnenburg, 2007) further generalises HM-SVM by accepting non-Markovian behaviour, in which a state is allowed to
persist through multiple time steps, similar to GHMM. This approach enables the length modelling of gene structural elements and non-linear content sensors.

Despite this progress, machine learning is not yet widely used for the prediction of genes. SVM-based gene prediction algorithms include CONTRAST (Gross et al., 2007) and mGene (Schweikert, Zien, 2009). CONTRAST uses quadratic kernels to predict signal features and combines the predictions with gene structures, employing conditional random fields (CRF) (Lafferty et al.). CRF does not allow semi-Markov dependencies among gene structural elements (Schweikert, Zien, 2009) and, therefore, the lengths of the predicted gene elements can be compromised. CONTRAST supports the use of both evidence data (e.g., mRNAs) and synteny among multiple genomes. By contrast, mGene uses string kernels (Ben-Hur et al., 2008) to predict signal and content features, and employs Hidden semi-Markov Model (HSM) SVM, with dynamic programming, to predict gene structures.

2.3. Functional annotation of predicted proteins

To conduct various analyses of pathways, essentiality and drug targets, the functions of genes need to be predicted. In this sub-section, only computational predictions of the functions of protein-encoding sequences and their inferred products are considered. Computational prediction can rely, for instance, on sequence similarity, clustering and phylogenomics, structure and structural similarity, protein-protein interaction (PPI) networks and/or machine learning (Erdin, Lisewski, 2011). Commonly used annotation tools, such as BLAST (Altschul, Madden, 1997) and the InterPro software framework (Hunter et al., 2012), use sequence similarity.

The success or reliability of functional prediction is influenced by the accuracy of an alignment of homologous characters in two or more sequences. Typically, two randomly generated protein sequences of > 100 amino acids in length, have 10–20% sequence identity. This identity range imposes practical limits on the success of identifying homologous protein sequences, and has raised a concept called the “twilight zone” (identity range: 15–25%) (Doolittle, 1986). The reliability of the prediction of homologous protein sequences in the twilight zone is only 10% (Rost, 1999). In contrast, the reliability of a prediction is 90% when sequence identity is > 30%. However, for enzyme pairs, < 30% have identical EC numbers, when sequence identity is > 50% (Rost, 2002). To confidently transfer the three digits of an EC annotation scheme, 40% sequence identity is required, and even 60% identity is needed to transfer all four digits at 90% accuracy (Tian and Skolnick, 2003). On the other hand, only 86% of BLAST matches with E-values of < 10^-50 have an identical EC annotation (Rost, 2002). By contrast, the structure of a protein is imposed by the amino acids covering only 3-4% of the sequence (Rost, 1997, 1999), and structural homologs with a sequence identity of < 15% can be identified (“midnight zone”). Similar structure is likely at identities of > 33%.

Errors in functional annotation of genes are widespread in current databases. Estimates indicate that 5–63% of gene annotations in public databases are incorrect or misleading, and are propagated via analyses of new genomes (Schnoes et al., 2009). These errors originate from various sources including genome assembly and gene prediction. Erroneous or incomplete genome assembly might relate to truncated or chimeric sequences as well as single nucleotide errors (Alkan, Sajjadian, 2011), which can cause a poor prediction of genes and gene functions (Do and Choi, 2006). RefSeq (Pruitt et al., 2009) and UniProt/SwissProt (Magrane and Consortium, 2011) are the best-curated databases.

Classification of proteins based on structure and function. The description of an accurate, uniform and coherent annotation of protein function(s) is achieved using a common set of rules, defined in unified classification schemes such as Gene Ontology (GO) (Ashburner et al., 2000), Enzyme Commission (EC) (IUBMB, 1992, McDonald et al., 2009) and Kyoto Encyclopedia of Genes and
Genomes (KEGG) BRITE functional hierarchies (Kanehisa et al., 2012). For GO, the proteins are classified according to three hierarchical domains relating to molecular function (MF), biological process (BP) and cellular component (CC). EC classifies enzymes based on their catalytic function. The EC classification scheme has four levels. The first level consists of oxidoreductases, transferases, hydrolases, lyases, isomerases and ligases, followed by three levels, which further refine enzyme function. KEGG BRITE functional hierarchies represent genes and proteins as orthologous protein families, arranged in a clear and versatile hierarchy.

The structure of a protein often relates to the function but differs in hierarchy from functional classification schemes. Three classification schemes have been devised for protein structures, namely Structural Classification of Proteins (SCOP) (Andreeva et al., 2004, Murzin et al., 1995), Class, Architecture, Topology, Homologous (CATH) superfamily (Cuff et al., 2009, Orengo et al., 1997) and Families of Structurally Similar Proteins (FSSP) (Holm and Sander, 1994, 1997). SCOP classifies both structure and function of a protein to a hierarchy of class, fold, superfamily and family. CATH sorts the arrangements of secondary structures to folds and to homologous superfamilies, and is used in the Protein Data Bank (PDB) (Berman et al., 2000). FSSP is a database representing structural alignments of PDB proteins with a sequence similarity of < 70% aligned using the structural alignment program DALI (Holm and Sander, 1993).

**Annotation based on sequence similarity.** Functional annotation methods rely on sequence similarity comparisons with proteins of known function(s). These methods establish similarities either by global alignment of full-length sequences or by local alignments of domains, motifs (partial sequences) or fingerprints (motif sets). The concepts of domain, motif and fingerprint in sequences contained in these databases and functional prediction methods are illustrated in Fig. 2.

A widely used approach is to transfer the annotation of an homologous protein identified by gapped BLAST (Altschul, Madden, 1997). The sensitivity of BLAST is comparable with that of Smith-Waterman (Altschul, Madden, 1997, Smith and Waterman, 1981), but evolutionarily distant homologs are not always recognised. To address this latter issue, iterative PSI-BLAST using position-specific score matrix (PSSM), was devised. PSSM is reconstructed for individual iterations using sequences from previous iterations (Altschul, Madden, 1997). BLAST-based methods do not annotate proteins based on features, but rather they transfer an annotation from another protein predicted to be a homolog. In some cases, homologs may only align over a small portion of their overall lengths, which could lead to an erroneous transfer of function. In addition, it is possible that the homolog from which function is being transferred might originally have been mis-annotated. Because BLAST is also commonly used to predict orthologous proteins from KEGG databases, any mis-annotation can result in erroneous predictions of metabolic pathways and protein families.

For sequence-based protein annotation, there are many publicly available databases and methods for the prediction of function. These essential databases and methods are collectively integrated in the widely used InterPro database (Hunter, Jones, 2012). InterPro supports databases such has Pfam (Punta et al., 2012), PRINTS (Attwood et al., 2012), PROSITE (Sigrist et al., 2013), SMART (Letunic et al., 2012), ProDom (Bru et al., 2005), PIRSF (Nikolskaya et al., 2006), SUPERFAMILY (de Lima Morais et al., 2011), PANTHER (Mi et al., 2013), CATH-Gen3D (Lees et al., 2012), TIGRFAM (Haft et al., 2013) and HAMAP (Lima et al., 2009) as well as the signal peptide and transmembrane domain prediction algorithms SignalP (Petersen et al., 2011) and TMHMM (Sonnhammer et al., 1998).

PANTHER, PIRSF, TIGRFAM and HAMAP databases are used to annotate full-length protein sequences. The protein data in PANTHER database are divided to superfamilies and families, which are then positioned in a phylogenetic tree. PIRSF requires homeomorphic (i.e. topologically isomorphic) domain architecture, and thus accounts for proteins with multiple functions. TIGRFAM and HAMAP are limited to data for bacteria and archaea. The short functional sequence fragments (domains or motifs) are predicted using either profile HMM (Eddy, 1998) or regular expressions.
(Thompson, 1968) to identify patterns in sequence strings. Domain databases using profile HMM are Pfam, SMART, ProDom, SUPERFAMILY and Gene3D, whereas motifs (identified by regular expressions) are used in the PROSITE database. A group of conserved motifs in a protein form a “fingerprint”, a concept used in the PRINTS database. PIRSF, PRINTS and PROSITE are reliable, manually curated sources for protein annotation, but the number of proteins found in these databases is limited. SignalP and TMHMM predict signal peptides and transmembrane domains, respectively, from a protein sequence.

Databases with curated or predicted domains and motifs deserve detailed consideration, because they are concerned with the functional components of proteins. Pfam has a curated (PfamA) and a computationally generated (PfamB) domain family database. Clusters of domain families are defined using the program ADDA (Heger and Holm, 2003), in which the clusters are formed from pairwise comparisons of profiles of domains inferred by penalizing splits and partial overlaps in a pairwise, BLAST aligned protein similarity matrix. SMART requires manual intervention during annotation and is linked to a database called Search Tool for the Retrieval of Interacting Genes (STRING) (Szklarczyk et al., 2011). ProDom compares the results from PSI-BLAST against the UniProtKB database (Magrane and Consortium, 2011) using ASTRAL compendium (Chandonia et al., 2004), and infers domain information from resultant data. ProDom complements domain databases, such as Pfam, PROSITE and SMART. SUPERFAMILY resource uses the SCOP classification scheme for inferred protein domain superfamilies, and assigns GO terms to these families using Gene Ontology Annotation (GOA) (Barrell et al., 2009), in which GO annotations are assigned to proteins in UniProtKB. Gene3D combines both structural and functional information to annotate domains found in sequences in UniProtKB, RefSeq and Ensembl (Kersey et al., 2010) databases. Gene3D uses the structural CATH classification scheme, and clusters annotated superfamilies into functional subfamilies using GeMMA (Lee et al., 2010). PROSITE recognizes protein motifs using regular expressions and weight matrix profiles, augmented by the annotation rule database ProRule (Sigrist et al., 2005). ProRule increases the reliability by imposing rules, such as essential amino acids in the active sites of an enzyme. The annotation of proteins encoded in a genome can be inferred by combining results from searches of all of the above, curated public databases containing information/data on protein sequences, domains and motifs. However, currently, these resources are far from complete, and the domains of some genes/proteins of parasites are likely to differ significantly from those found in these databases, or are not yet defined, thus compromising their recognition.

InterPro is commonly used to assign hierarchical GO terms for annotated proteins. However, a number of programs, such as GOblet (Groth et al., 2004, Hennig et al., 2003), OntoBlast (Zehetner, 2003), GoFigure (Khan et al., 2003), GOtcha (Martin et al., 2004) and Blast2GO (Gotz et al., 2008), annotate proteins directly using GO terms by combining BLAST results. Blast2GO also integrates the annotation results from databases including InterPro and KEGG. Some methods, such as Protein Function Prediction (PFP) (Hawkings et al., 2009, Hawkings et al., 2006) and Extended Similarity Group (ESG) (Chitale et al., 2009), rely on PSI-BLAST and aim to further extend remote homolog identification. As these latter annotation methods have not gained wide acceptance, they are not considered further in this review.

GO terms provide a hierarchical framework for the genome, in terms of functionality linked to the domains of the predicted proteins, and provide better annotation coverage compared with protein-level annotation using, for example, the KEGG BRITE database. The hierarchical representation of proteins offered by the KEGG BRITE database is useful for analyses of genomic data, particularly because it offers a direct link to an alternative, pathway-level representation supported by the KEGG database. However, this representation is limited to the orthologous proteins of some species, and thus has limited coverage. InterPro or KEGG BRITE results can also be readily used to infer EC numbers of enzymes in a predicted proteome.
**Annotation based on clustering and phylogenomics.** The methods for predicting orthologous (genes in different species having common ancestor) and paralogous (duplicated genes within a genome) (Fitch, 1970) protein groups among multiple species can be divided into those that rely on clustering based on similarity and those that use a phylogenetic approach (Table 7). The resultant groups are expected to represent proteins with the same or similar functions, and can thus be used for protein annotation.

Clustering methods, such as OrthoMCL (Li et al., 2003), InParanoid (Remm et al., 2001) and MultiParanoid (Alexeyenko et al., 2006), and databases OrthoDB (Waterhouse et al., 2013) and Clusters of Orthologous Groups of proteins (COG) (Tatusov et al., 2003) use all-versus-all similarity matrices, created based on pairwise alignments of protein sequence using algorithms such as BLAST, FASTA and Smith-Waterman. OrthoMCL uses BLAST to present scores in a similarity matrix. This matrix then serves as input to Markov clustering algorithm (Van Dongen, 2000), in which the orthologous protein clusters are formed. Similarly, InParanoid first calculates pairwise similarity scores using BLAST and then identifies reciprocal matches (= hits) that are used as seeds to define orthologous protein clusters. To avoid erroneous, high scoring matches for short alignments, a pairwise protein sequence alignment length of > 50% of the longer protein sequence is required. MultiParanoid combines pairwise clustering results from InParanoid among multiple genomes, using a variant of the hierarchical agglomerative single linkage clustering method (Sneath, 1957). In short, this variant strives to maximize cluster sizes and stops after all pairwise clusters have been processed, thus not proceeding up to the root of the hierarchical tree. The clustering algorithm of eukaryotic orthologous groups (KOGs) in the COG database also uses BLAST to calculate pairwise similarities among protein sequences, but first filters known repetitive domains and low complexity regions to avoid non-homologous pairing. In the clustering step, the reciprocal best hits among groups of three species (Tatusov et al., 1997) are extracted and merged into KOGs. OrthoDB uses a Smith-Waterman alignment to create a similarity matrix, and, like KOG, forms clusters based on reciprocal best hits among groups of three species. To avoid false-positive results from high-scoring, short alignments, a minimum overlap length of 30 amino acids among sequences is required.

The largest publicly available all-versus-all protein sequence similarity score matrix is called Similarity Matrix of Proteins (SIMAP) (Rattei et al., 2010). SIMAP imports proteins from multiple databases, such as Ensembl, NCBI GeneBank, NCBI RefSeq, UniProt/TrEMBL, UniProt/SwissProt and PDB, and is updated monthly. SIMAP encompassed > 48 million sequences in the release of January 2015. The pairwise scores in this matrix are calculated from pairwise alignments made using FASTA and Smith-Waterman algorithms. SIMAP is employed, for instance, in the database called “evolutionary genealogy of genes: Non-supervised Orthologous Groups” (eggNOG) (Jensen et al., 2008), in which the proteins are assembled into inparalogous groups by comparing sequence similarities within and among clades. Orthologous proteins are then identified amongst these inparalogous groups by creating and merging reciprocal best hits among three species (Tatusov, Koonin, 1997).

These clustering methods can be improved by replacing the amino acids substitution models, such as BLOSUM (Henikoff and Henikoff, 1992) and PAM (Dayhoff et al., 1978), with models that better estimate phylogenetic distances, such as JTT (Jones et al., 1992) and WAG (Whelan and Goldman, 2001), and by reconciliation of the deduced phylogenetic tree of each gene to the phylogenetic tree of species, in order to predict orthologs and paralogs (El-Mabrouk and Sankoff, 2012). Synteny (Wusirika and Shaik, 2014) and incongruence of gene trees (Leigh et al., 2011) play a role in these predictions. This approach has implications in terms of the prediction of protein function, because paralogs tend to have divergent functions compared with orthologs (Ohno, 1970). The reconciliation (El-Mabrouk and Sankoff, 2012) is accounted for in methods and databases such as SYNERGY (Wapinski et al., 2007), Phylogenetically Inferred Groups (PhIG) (Dehal and Boore, 2006), TreeFam (Li et al., 2006, Ruan et al., 2008) and PANTHER (Mi, Muruganujan, 2013).
SYNERGY aims to represent orthologous genes as ortho-groups (OGs) using a single hypothetical ancestral gene in the phylogenetic tree constructed from the deduced protein sequences of these genes. In this method, the Gene Similarity Graph (GSG) graph, equivalent to similarity matrix, is created using the program FASTA and by weighting pairwise phylogenetic distances of resultant aligned protein pairs, employing amino acid substitution rates defined in JTT model and by synteny in the genomic neighbourhood of the genes encoding the aligned protein pair. For a particular gene, an OG is then inferred both from GSG and a phylogenetic tree of the species under study. Unfortunately, SYNERGY does not have a large-scale database associated with it. For the PhIG database, BLAST is used to identify similar protein pairs, and the pairwise phylogenetic distance of these pairs is calculated using the JTT substitution model. These pairs are further clustered (using single linkage) into multiple aligned protein families (seeds). For each family, a phylogenetic tree, based on the JTT substitution model, is then calculated using the Maximum Likelihood (ML) algorithm TREE PUZZLE (Schmidt et al., 2002). Finally, orthologs and paralogs are deduced from the trees of protein families and the phylogenetic tree of species being studied. TreeFam uses protein family seeds identified by PhIG. These seeds are then expanded with hits found in BLAST and HMMER searches against multiple protein databases. Expanded protein families are then multi-aligned, and the phylogenetic distances among aligned protein families are estimated by constructing phylogenetic trees using the program PhyML (Guindon and Gascuel, 2003) employing WAG and NKY (Hasegawa et al., 1985) substitution models, and the Neighbour Joining (NJ) method (Saitou and Nei, 1987). Orthologs and paralogs are then inferred from the resultant phylogenetic trees, together with the species tree, and results are manually curated. The PANTHER protein families must contain at least five members, and the resultant profile HMM must be sensitive enough to recognize the protein family members among all other families in the database. If the program GIGA (Thomas, 2010) is used to construct a phylogenetic tree, and to infer speciation and duplication events, then ≥ 75% of all amino acids in the multiple alignment of sequences representing a protein family need to align.

Databases offered by these methods provide a compelling option to rapidly predict protein function, but these databases are limited in their coverage of both species and proteins. Furthermore, using sequence similarity searches to position the query sequence in phylogenetic trees in these databases, in which substitution models and taxonomic information are utilised for construction, seems questionable. Instead, the query sequence should be positioned in a phylogenetic tree using methodologies commensurate with the database construction. However, even a perfect positioning does not guarantee an accurate prediction of function for the query protein sequence, because homologous proteins do not always have the same function (Doolittle, 1986, Rost, 1999, Rost, 2002, Tian and Skolnick, 2003). The methods presented in this section attempt to address this issue by reconciliation of orthologous protein trees with a phylogenetic tree of species, in order to assign proteins as orthologs or paralogs. However, clearly, a more accurate means of identifying conserved functions is needed.

**Annotation based on protein structure.** The sequence-based annotation methods described thus far can accurately predict the function of a protein as long as a well-annotated homologous protein exists. Although this is often not the case, function can still be predicted (Table 8) by comparing folds in a protein against structural homologs in databases such as PDB (Bernstein et al., 1977). The function assigned to such folds offers an initial prediction, even though it is known that the structure of a protein does not always correlate with its function – only around 60% of structurally similar proteins without significant sequence similarity share a binding site location. (Martin et al., 1998, Russell et al., 1998). PDB has more than 100,000 macromolecular structures, most of which relate to proteins whose structures have been established via the Protein Structure Initiative (PSI) (http://www.pdb.org), a Structural Genomics Initiative (SGI)
established in 2000. Unfortunately, the functional knowledge of these protein structures is lacking. Clearly, the prediction of protein function, based solely on structure, needs to be complemented with other methods. However, it is also well known, that, in divergent evolution, the structure and function of even distantly related orthologous proteins are relatively well conserved (Doolittle, 1994, Martin, Orengo, 1998, Murzin, 1993), therefore supporting structure-based function prediction. By contrast, in convergent evolution, the same function is observed with differing folds (Russell, 1998), thus discouraging such an approach for the prediction of function. Enzyme function is best inferred based on conserved amino acids in the active site, instead of relying on the shape of the fold around it, which mostly correlates with ligands (Martin, Orengo, 1998, Rost, 1997). Nonetheless, the global structure of a protein also has to be considered when predicting enzyme function.

The widely used sequence-to-structure-to-function annotation paradigm consists of four steps: (i) the search and selection of a three-dimensional (3D) template of a protein, (ii) model building from well-aligned template fragments, (iii) subsequent quality evaluation of resultant model(s) and (iv) the prediction of function. Ab initio structure predictions can be used in the absence of matching structural templates or to infer the structure of non-aligned regions, such as loops. Typically, rapid protein secondary structure-based profile-profile (Wallner et al., 2004) threading algorithms (Shi et al., 2001, Wu and Zhang, 2008, Zhou and Zhou, 2005) and meta-predictors, combining multiple threading algorithms, such as LOMETS (Wu and Zhang, 2007), are used to search for and select templates (Bujnicki et al., 2001, Rychelewski and Fischer, 2005). A model is then built from such templates or template fragments using rigid body, segment matching or spatial constraint methods, and loops are modelled separately either from loop libraries or ab initio predictions (Kelley and Sternberg, 2009, Marti-Renom et al., 2000, Roy et al., 2010). The evaluation of the quality of a model is based on various alignment metrics, such as TM-score and root mean square deviation (RMSD), but these scores are always dependent on similarity comparisons between or among multiple structures, thereby compromising their accuracy (Hasegawa and Holm, 2009).

Subsequently, the prediction of function is achieved by aligning the predicted final model to known protein structures using structural alignment techniques, such as Combinatorial Extension (CE) (Shindyalov and Bourne, 1998), DALI (Holm and Sander, 1993), Vorolign (Birzele et al., 2007), TM-align (Zhang and Skolnick, 2005) and the phenotypic plasticity method (PPM) (Csaba et al., 2008). Vorolign uses tessellation (Voronoi, 1908) to infer neighbouring alpha-carbons for each sequential position in two protein structures, and scores the similarity of these neighbours by aligning them with dynamic programming. These scores for each neighbourhood are then transferred into a 2D-matrix covering all positions in both proteins being compared, and dynamic programming in this matrix is used to achieve the final alignment as well as the alignment score for the protein pair. CE searches for the longest continuous path to the aligned fragment pairs (AFP) and accounts for all combinations in the extension of the alignment. DALI decomposes protein structures into contact points (distance sub-matrices), which are used to infer the alignment path iteratively with the Metropolis algorithm (Metropolis et al., 1953). TM-align implements an iterative dynamic programming algorithm to achieve optimal TM-score in aligned structural elements. Finally, when these methods have identified one or more high scoring structural matches, a function can be transferred to the final predicted model. Structural aligners, which provide some degree of flexibility, seem to model dynamic protein structures better and, therefore, perform better than rigid aligners (Hasegawa and Holm, 2009).

To increase the accuracy of structure-based function prediction, conserved amino acids in active and binding sites need to be evaluated. For enzymes, catalytic residues and their 3D location and orientation in active sites are usually conserved (Zvelebil et al., 1987), and do not always associate with structural variation, thereby allowing the functional annotation of distantly related homologs (Torrance et al., 2005). The identification of conserved residues in protein families is based on Multiple Sequence Alignment (MSA); the evolutionary trace (ET) paths method (Innis et al., 2000,
Lichtarge et al., 1996, Yao et al., 2003) extracts conserved residues in 3D space and uses entropy considerations in real-valued ET scores (Mihalek et al., 2004). Other methods, such as ConFunc (Wass and Sternberg, 2008), predict conserved residues among GO specific groups following MSA, and directly assign query sequences to GO domains. The program JESS (Barker and Thornton, 2003) estimates residue conservation of catalytic sites in 3D space, and provides catalytic structural templates (Torrance, Bartlett, 2005) for use by Catalytic Site Atlas (CSA) (Porter et al., 2004). Implementations such as COFACTOR (Roy et al., 2012) in I-TASSER (Roy, Kucukural, 2010) first predict function based on the global structure of a protein, estimate conserved residues in a protein and then compare these residues to the catalytic structural templates in CSA, and to the known functional residues in protein templates. In the Phyre2 (Kelley et al., 2015) implementation, confident protein structure and functional residue predictions are submitted to the program 3DLigandSite (Wass et al., 2010) for the inference of ligand binding sites.

**Other annotation methods.** In addition to the protein functional annotation methods described above, a large number of other computational annotation methods are available. Among them are experimentally evaluated and computationally predicted protein-protein Interaction (PPI) networks and protein-protein complexes (Sowmya and Ranganathan, 2013), for which results are found in databases such as DIP (Salwinski et al., 2004) and STRING (Szklarczyk, Franceschini, 2011). Most promising annotation appears to be achievable using machine learning-based, supervised classification methods, and unsupervised clustering methods (Alpaydin, 2009). These methods can be applied to predict individual features of proteins (e.g., domain boundaries, subcellular location and conserved residues), to collectively predict function(s) based on data integrated from different sources (sequence, structure, transcription, taxonomy, and metabolic and protein-protein interaction networks), or to enhance an existing homology-based annotation (Bernardes and Pedreira, 2013). The functional classification of proteins can be critically evaluated in an annual Critical Assessment of Functional Prediction (CAFA) challenge (http://biofunctionprediction.org).

### 2.4. Post-genomic analyses

Genome assembly and annotation provide a basis for additional analyses. Post-genomic analyses encompass a plethora of areas, including differential transcription analysis, comparative genomics, gene essentiality predictions, drug and vaccine target discovery, drug resistance, metabolic pathway and gene regulatory network analyses, pathogenesis of disease, phylogenomics and population genetics. Currently, there are few tools for the post-genomic analyses of *de novo*-assembled and -annotated genomes, and their integrated application to the genomes, transcriptomes and proteomes of parasitic worms is lacking. In this section, a select number of approaches are reviewed, which are pertinent to investigations of parasitic worms.

**Transcriptomic analyses.** Presently, RNA-sequencing (RNA-seq) is the preferred technology to study transcriptomes linked to *de novo*-assembled genomes, because it offers genome-wide transcription profiles for any species, and can also identify genes not detected in draft genome assemblies (Wang et al., 2009). In addition, unlike hybridisation- and PCR-based microarrays (Wang, Gerstein, 2009), RNA-seq does not require the use or design of oligonucleotide probes, and platforms used for genome sequencing can also sequence transcribed RNA molecules from a sample. The resultant RNA-seq data can then be used to explore transcription qualitatively and quantitatively, in order to detect differentially transcribed genes and gene transcription profiles, and to detect splice junctions. Importantly, RNA-seq data can be assembled to serve as training and evidence data for the prediction of genes (Consortium, 2014, Li et al., 2014).

RNA-seq data can have multiple sources of bias, which need to be considered before analyses are conducted. Technical sources for bias can include library size, GC content, template shearing
and the transfer of technical errors introduced during library construction (Dillies et al., 2013, Li, Labaj, 2014). Methods that account for technical biases using data normalisation include sva (Leek et al., 2012) and PEER (Stegle et al., 2010), and Trimmed Mean of M-values (TMM) (Robinson and Oshlack, 2010) and DESeq (Anders and Huber, 2010) for Illumina data (Dillies, Rau, 2013). However, presently, there are no normalisation methods for comparative analyses of transcription among species, and biases in data sets, in this context, are unknown. Transcription profiling and the analysis of differential transcription using RNA-seq data can be undertaken employing parametric EdgeR (McCarthy et al., 2012) and DESeq (Anders and Huber, 2010), and non-parametric NOIseq (Tarazona et al., 2011) programs. The sensitivity and accuracy of RNA-seq depends on sequencing depth and the read mapping method selected for the analyses, but also on the accuracy and completeness of the gene models used for gene prediction (Consortium, 2014).

**Phylogenetic analyses.** Historically, morphological characters have been used for phylogenetic studies of organisms. However, such characters are not uniformly applicable across a myriad of taxa, and particularly in nematodes are often not sufficiently informative for systematic investigations (Dorris et al., 1999). Instead, due to the adaptation of organisms to challenging environments, evolutionarily conserved features among a wide range of disparate taxa can be found in molecules. Nucleic acid and protein sequences offer a uniform metric (mutation rate) for phylogenetic analyses; this rate is often referred to as “molecular clock” (Zuckerkandl and Pauling, 1962). Fossil records are needed to calibrate this clock but are rare for nematodes (Conway, 1981, De Baets and Littlewood, 2015).

Molecular methods have become the mainstay to reconstruct phylogenies, although the use of data for only one or a few genetic loci, such as ribosomal DNA (rDNA) (Dorris, De Ley, 1999), does not always provide strong support for the relationships of taxa due to coverage of merely a fraction of the total nucleotide content of an organism. Instead, the use of a large number of loci results in more confident phylogenies, which is proposed to relate to the “cancelling out” of conflicting signals among multiple loci (Rokas and Carroll, 2005, Rokas et al., 2003). Obviously, the availability of genome-wide data sets provides unique opportunities for phylogenetic studies. Such data sets contain a wealth of phylogenetically informative sequences, including those encoded by single copy orthologous (SCO) genes (Aguileta et al., 2008, Ciccarelli et al., 2005, Pisani et al., 2007). SCOs can be concatenated, partitioned into gene groups or processed in a gene-wise manner and then subjected to analyses. However, It is not yet clear whether concatenation or partitioning of data is most suitable (Gadagkar et al., 2005, Gontcharov et al., 2004, Rokas, Williams, 2003), and the process of partitioning is considered as art rather than science by some researchers (Yang and Rannala, 2012). Importantly, taxon sampling (i.e. the selection of species for analysis) can also have an impact on the resultant phylogenetic relationships (Rokas and Carroll, 2005), the preference being for large numbers of taxa (Dunn et al., 2008, Heath et al., 2008, Zwickl and Hillis, 2002). Multi-gene congruence analysis strives to resolve incongruence among different phylogenetic trees derived from data from orthologous molecules, and might be used to group molecules sharing similar sequence patterns but resulting in different, yet well-supported trees (Leigh, Lapointe, 2011).

From a computational perspective, for the construction of a consensus tree, it is advantageous to perform phylogenetic analyses employing concatenated sequences or groups of sequences with particular evolutionary characteristics, such as mutation rate and codon position (Yang and Rannala, 2012), rather than individual sequences.

The most commonly used methods to predict phylogenetic relationships among species are character-based maximum likelihood (ML) (Savage, 1976), Bayesian inference (BI) (Bayes and Price, 1763, Fienberg, 2006, Yang and Rannala, 1997), maximum parsimony (MP) (Camin and Sokal, 1965, Fitch, 1971, Hartigan, 1973) and distance-based (Haubold, 2014) neighbour joining (NJ) (Saitou and Nei, 1987) tree-building algorithms (cf. Table 9). The critical step for ML, BI and
MP analyses is a multiple sequence alignment, in which the nucleotide or amino acid sequences representing the species under study are aligned based on their similarity and substitution matrices (Penn et al., 2010, Phillips et al., 2000). The MP method, which minimises the total tree length, is computationally feasible, but suffers from long branch attraction (LBA) caused by an inability to account for multiple mutations in the same column (Felsenstein, 1978, Yang and Rannala, 2012). The ML method estimates optimal constant parameters (branch lengths and substitution parameters) for a statistical model (topology) matching the data (aligned sequences), whereas for BI, the parameters are random variables, and a prior distribution is assigned to these variables (Yang and Rannala, 2012). The ML and BI methods support many different evolutionary models, but both are computationally intensive (Yang and Rannala, 2012).

To estimate divergence or speciation times, methods based on molecular clock models are used (programs supporting such models are listed in Table 9). These models are divided into strict (global), relaxed multi-rate local and discrete, and relaxed auto-correlated and uncorrelated (independent-rates) models (Ho and Duchene, 2014). In the strict clock model, the rate of evolution is assumed to be the same for all organisms and, therefore, has restricted practical use for model selection (Ho and Duchene, 2014). The local and discrete models relax the assumption of rate homogeneity (Ho and Duchene, 2014), allow different rates of evolution on different branches and are capable of grouping adjacent (local) or non-adjacent (discrete) branches (Ho and Duchene, 2014). The auto-correlated, relaxed methods assume limited variations on the rate homogeneity between adjacent branches, whereas the clock models of independent-rates do not (Ho and Duchene, 2014, Rannala and Yang, 2007). Consequently, auto-correlated models are likely to best suit the construction of phylogenetic trees in which species are taxonomically closely related, and independent-rates models best suit taxonomically distant ones, emphasising the dependency on taxon sampling. The method partitioning and/or selection of sequence data for genome-wide analyses are similar to those used for phylogenetic analyses. The calibration of these clock models is usually based on fossil records or tectonic events known to relate to speciation events and, because of an uncertainty regarding dating, are modelled as distributions rather than fixed time points (Warnock, 2014).

Synteny analyses. Synteny is a concept in which conserved segments (orthologous chromosomal regions) in different genomes preserve their order (Nadeau and Taylor, 1984). Syntenic blocks are defined as nucleotide sequence segments that can be converted to conserved segments by small rearrangements, called micro-arrangements (Pevzner and Tesler, 2003). Such blocks include orthologous genes or non-coding conserved nucleotide sequence tracts called anchor-nucleotide regions (Pevzner and Tesler, 2003). The identification of such regions and the subsequent prediction of syntenic blocks are achieved through a genome alignment, which is usually a challenging computational task (Hardison, 2003), and can be complicated by the presence of repeats in genomes (Bourque et al., 2004). The last step in a synteny analysis is to rearrange syntenic blocks among the genomes and to visually display the result. The rearrangement algorithms are typically heuristic, parsimony-based implementations (Bourque, Pevzner, 2004). For fragmented de novo-assembled genomes, the arrangement of the contigs on chromosomes is not known and, therefore, adds a level of complexity to the prediction of synteny and the display of results; currently available programs do not cope well with these issues. Synteny can assist gene prediction and functional annotation (Wusirika and Shaik, 2014) as well as the prediction of chromosomal duplications, deletions and rearrangements, serving as a basis, for instance, for predicting ancestral
genomes (Ma, 2011). The programs used to resolve and/or visually display synteny are listed in Table 10. These programs either align the genomic sequences using alignment implementations, such as LASTZ (Harris, 2007), BLASTN (Camacho et al., 2009) and BLASTP (Camacho, Coulouris, 2009), or require files with genomic positions of anchors.

**Metabolic pathway analyses.** Metabolic pathway and enrichment analyses can be conducted using gene and transcription data encoded in de novo-assembled and annotated genomes. The key metabolic pathway databases and computational tools for this purpose are summarized in Table 11. Pathway analyses can be carried out using information available in public databases, such as Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa, Goto, 2012), MetaCyc (Casi et al., 2014), BioCyc (Casi, Altman, 2014), ConsensusPathDB (Kamburov et al., 2011) and Reactome (Croft et al., 2014), and the commercial Ingenuity Pathway Analysis™ (IPA) (http://www.qiagen.com/ingenuity) and/or MetaCore™ (http://lsresearch.thomsonreuters.com) databases employing available analysis software (embedded within these databases). Typically, these databases assume that the organism of interest is readily integrated in the database, and thus are usually not able to accommodate data from an annotated genome from a new species. Only the KEGG database and the MetaCore™ suite allow the tailoring of analyses to an unknown organism. In the KEGG database, genes are first mapped to KEGG orthologs, which can then be used to extract enriched pathways, whereas MetaCore™ supports the construction of a customised database. However, these analyses are limited to identifying known pathways rather than inferring pathways ab initio. The database EuPathDB (Aurrecoechea et al., 2013) aims to predict or identify pathogen-specific pathways, but lacks support for the integration of de novo-assembled genomes.

Nevertheless, novel metabolic pathways could be inferred or predicted using the methods implemented in programs such as Pathway Tools (Dale et al., 2010, Karp et al., 2010, Paley and Karp, 2002) and/or PathMiner (McShan et al., 2003). These programs are typically based on sub-graph predictions from all-encompassing metabolic networks (reactions and compounds) (Faust et al., 2011). However, the construction of complete metabolic networks (Thiele and Palsson, 2010), which involves numerous manual steps and might take years to achieve, would substantially improve the quality of pathway predictions. The prediction of synthetic pathways that would produce particular metabolites might also be useful (Latendresse et al., 2014, Medema et al., 2012). However, pathway predictions are highly dependent on accurate enzyme annotations (Dale, Popescu, 2010), which relates back to the original quality of gene predictions and functional annotations.

3. Considerations regarding the sequencing, annotation and assembly of nematode genomes

Since the introduction of Illumina sequencing technology (Henson, Tischler, 2012), there has been a major expansion in the amount of genomic sequence data for parasitic worms (Table 12). To date, in numerous investigations, 38 draft genomes of nematodes have been assembled, annotated and employed (Table 12). These studies show that these genomes vary considerably in size and composition; therefore, no “reference” genome exists for members of the phylum Nematoda. Consequently, each new nematode genome must be assembled de novo. A factor contributing to the quality of the final genome assembly is the quality and quantity of genomic DNA isolated for subsequent library construction and sequencing. For some tiny nematodes, there are significant limitations in extracting high quality, high molecular weight genomic DNA (Gasser, 2013, Gasser, Chilton, 1993, Gasser, Hu, 2006). If small quantities of such DNA are available, techniques, such as multiple displacement amplification (MDA), can be used to produce (up to a billion-fold) genomic DNA (even from a single cell) (Dean et al., 2002). This amplification process can lead to a bias, toward CG pairing, and some DNA elements may be “lost”, particularly when single cells are used
as templates (Raghunathan, Ferguson, 2005). However, a few thousand copies of genomic DNA are required to minimise possible amplification bias in MDA (Hosono et al., 2003). Following amplification, AT rich areas, such as those found sometimes in repeat regions (Diethard, 1993), might lack the necessary sequencing depth for successful genome assembly. Amplification bias should be detected using third-generation sequencing approaches, where read length relates directly to the quality of genomic DNA extracted and/or library construction (see Section 2.1.1). However, for short read data produced by second-generation sequencers (see Section 2.1.1), long repeat regions cannot be assembled reliably (Alkan, Sajjadian, 2011), and this bias might go unnoticed.

In addition, other factors, such as high levels of sequence heterogeneity in genes within some species of parasitic nematodes, including *Haemonchus contortus*, *Teladorsagia circumcincta* and *Ostertagia ostertagi* (Blouin et al., 1995, Yin et al., 2013) and chromatin diminution in ascaridoids (Boveri, 1887, Goday and Pimpinelli, 1993, Jex et al., 2011, Wang et al., 2012a) can affect the quality of a genome assembly. In addition, genomic DNA isolated from parasites typically contains some host and bacterial contamination from their habitat. However, endosymbiotic bacteria, such as *Wolbachia* in *Brugia malayi*, might be considered part of a genome project, like a mitochondrial genome (Ghedin et al., 2007).

The annotation of a genome is also affected by peculiarities of parasites relating to their evolution. For instance, the adaptation of parasites to their hosts might be a reason for relatively high proportion of uncharacterised genes (often 24-35%) in their genomes (Foth et al., 2014, Ghedin, Wang, 2007, Jex, Liu, 2011, Mitreva et al., 2011, Srinivasan et al., 2013, Tang et al., 2014). These “unknown” genes are of considerable interest, because they are likely to encode proteins that are involved in host-parasite interactions and/or parasitism. The integration of genomic data in metabolic pathway databases (Table 11) for nematodes is lacking, and many pathways that are unique to parasitic worms are not represented in these databases. Therefore, unravelling the function for these genes could provide new insights into the biological pathways involved in parasitism, and this understanding might assist in the development of new strategies to block or disrupt the relationship of the parasite with its host. Moreover, comparisons of biological pathways between parasites and hosts might enable parasite-host interactions to be predicted computationally (Durmus et al., 2015).

4. The establishment of improved bioinformatic pipelines

Based on recent experiences made in several nematode genome projects (Korhonen et al., 2015, Schwarz et al., 2013, Zhu et al., 2015), we have now constructed improved genome assembly and annotation pipelines that have proven to be practical and time-efficient. The assembly pipeline (Fig. 3) pre-processes the sequence data, assembles these data into contigs, scaffolds the contigs and then closes gaps in the resultant scaffolds (scaffolds are DNA sequences composed of multiple contigs in estimated order and orientation having putative gaps between these contigs). The gene prediction pipeline (Fig. 4) employs the program MAKER2, such that the training of two *de novo*-predictors is automated utilising nucleotide and protein sequences, which serve as “evidence data” for the prediction of genes. The functional annotation component supports the application of BLAST against user-selected protein and nucleotide databases, predicts signal peptides and integrates the results with InterPro predictions into one file. Using these pipelines, a worm genome can be annotated in three consecutive runs; some manual post-processing steps are needed to refine the results from each run. Finally, although not yet integrated into the pipeline, we have established a set of scripts for the submission of annotated genomic datasets to the NCBI database.

4.1. Genome assembly pipeline
Our pipeline for the assembly of parasite genomes (Fig. 3) accepts raw Illumina sequence reads as input data, pre-processes these reads, assembles them into contigs and then combines these contigs to form scaffolds. Single-end (SE) and paired-end (PE) reads are used for contig assembly, whereas mate pair (MP) reads are used for scaffolding. Reads are presented in the FASTQ format, and each nucleotide has a PHRED quality score. The first step in the pre-processing is the trimming of reads, according to their quality value, such that only reads and read fragments of high quality are retained. Assemblers can cope, to some extent, with erroneous data. Therefore, quality trimming is a mandatory step for a successful assembly. The trimmer program selected for the pipeline is Trimmomatic (Bolger et al., 2014). Although some errors remain in trimmed reads, many of them can be statistically differentiated from true variations, such as single nucleotide polymorphisms (SNPs) or insertion/deletion events, and are thus corrected (Yang et al., 2013). The error-correction programs that have been integrated into the assembly pipeline (Fig. 3) are the error corrector module from the SOAPdenovo suite (Li et al., 2010) and BayesHammer (Nikolenko et al., 2013) embedded in the assembler SPAdes (Bankevich, Nurk, 2012). SPAdes was selected for contig assembly and integrated into the pipeline. Assembled contigs are then scaffolded in an iterative loop employing the program Opera (Gao et al., 2011), and gaps are closed utilising the program GapFiller (Boetzer and Pirovano, 2012). The rationale for this loop is that the closed gaps increase the nucleotide count, and the scaffolding result therefore improves in next iteration(s). The resultant, final assembly typically requires post-processing, such as the removal of putative contaminants and short contigs. If the assembly contains a significant amount of extraneous sequences (e.g., microbial and/or host contamination), they can be removed by mapping the genomic reads to putative contaminant sequences in public databases. Subsequently, the worm genome can be assembled again, in order to provide a simpler graph structure, thus resulting in an improved assembly. The quality of a resultant de novo assembly is difficult to estimate (Phillippy et al., 2008), and our quality considerations have been restricted to CEGMA completeness (Parra, Bradnam, 2007) and metrics implemented in QUAST analysis (Gurevich et al., 2013), although these components have not yet been integrated into the assembly pipeline (Fig. 3).

4.2. Genome annotation pipeline

Our pipeline for the annotation of parasite genomes is divided into two components: gene prediction and functional annotation of protein-encoding genes (Fig. 4). In this pipeline, the MAKER2 framework (Holt and Yandell, 2011) is employed for gene prediction. First, training and evidence data for MAKER2 are created from non-redundant transcriptomic assemblies of RNA-seq data utilising the programs Velvet (Zerbino and Birney, 2008) and Oases (Schulz et al., 2012) to produce multiple assemblies with different parameter values for k-mer and coverage cut-off. The resultant transcriptomic assemblies are combined, and transcript redundancy is decreased using the program CD-HIT-EST (Li and Godzik, 2006). Non-redundant transcripts are then mapped to the genome, and those with valid splicing sites are combined with the CEGMA predictions (Parra, Bradnam, 2007) and used as training data in the ab initio predictors SNAP (Korf, 2004) and AUGUSTUS (Stanke et al., 2006a), and then as evidence data for gene prediction. The evidence data are expanded to include also known proteomes of relatively closely related taxa, and provided to the MAKER2 framework, together with the trained HMM models for the ab initio gene predictors. Moreover, the results of ab initio GeneMark (Borodovsky and Lomsadze, 2011) and evidence-based Tophat/Cufflinks (Trapnell et al., 2009, Trapnell et al., 2010) gene predictions are added to the input data for MAKER2, which then executes ab initio gene predictions, maps the evidence data to the genome, delivers consensus gene predictions following the combining of all of these data, and finally formats them into a GFF file, which defines the structures of the predicted genes in the genome. Genes are then filtered for putative transposons, unusually short genes and exons, and nested or overlapping loci. Typically, transposons are predicted by identifying repeat
regions in the genome (Finnegan, 1989, Jurka et al., 2005, Wicker, Sabot, 2007). The subsequent functional annotation of predicted genes runs BLAST against databases defined by the user (e.g., UniProt/SwissProt and NCBI non-redundant protein databases), predicts signal peptides employing the packages SignalP (Petersen, Brunak, 2011) and Phobius (Kall et al., 2004), identifies transmembrane regions using the program TMHMM (Sonnhammer, von Heijne, 1998), and finally combines these results with those for InterPro into one annotation file. Any possible extraneous (“contaminating”) sequence data can be substantially reduced using functional annotation results.

5. What have we learned, and where to from now?

Based on results from some of our recent genome projects (Korhonen, Pozio, 2015, Schwarz, Korhonen, 2013, Zhu, Korhonen, 2015) and experiences over the last few years, we discuss issues that might be relevant for future genomic investigations of a wide range of nematodes and/or invertebrates.

5.1. Genome assembly

The genomes of parasitic nematodes differ substantially from one another in size and in the amount of sequence variability among individuals and populations. Such variability can compromise the success of the genome assembly, as was observed for H. contortus, in which extensive pre-processing, filtering of low and high coverage reads and removing putatively contaminated reads was required to accommodate high levels of population heterogeneity during genome assembly (Schwarz, Korhonen, 2013). However, despite all of these efforts, H. contortus genome remained fragmented (Velvet/Oases programs were used in assembly). In the future, this issue might be addressed by sequencing from single worms or eggs (unfertilised), rather than from a pool of hundreds of worms. Whole genome amplification (WGA) from small amounts of genomic DNA from tiny individual worms and advances in library construction (Dean, Hosono, 2002, Young et al., 2012) (http://www.qiagen.com/REPLI-g) make this proposal feasible. Indeed, for T. canis, single worms were used for sequencing and genome assembly (Zhu, Korhonen, 2015). Following the removal of putative bacterial contamination, we produced substantially longer scaffolds using the program SOAPdenovo, and thus considerably improved the N50 value (Schwarz, Korhonen, 2013, Zhu, Korhonen, 2015).

To assemble 16 genomes representing all 12 currently recognised Trichinella taxa (Korhonen, Pozio, 2015), we employed an automated genome assembly pipeline, utilising the SPAdes assembler (Fig. 3). Although a chromosomal-level completeness was not achieved (730-2697 scaffolds), the assemblies showed consistent characteristics for encapsulated and non-encapsulated clades, and their estimated CEGMA completeness (Parra, Bradnam, 2007) was > 95%/97% for complete/partial genes, in contrast to 73%/92% for H. contortus (Schwarz, Korhonen, 2013) and 67%/98% for T. canis (Zhu, Korhonen, 2015). Taken together, the paired de Bruijn graph-based SPAdes assembler provided a high level of CEGMA completeness for CEG genes (>95%) predicted from the assembled genomes; the lower figures for complete vs. partial CEGs for H. contortus (73% vs. 92%) and T. canis (67% vs. 98%) might relate to: (i) fragmented assemblies using the plain de Bruijn graph-coupled assemblers, (ii) structural variation in some CEGs of these parasites or (iii) smaller genome size (resulting in simpler de Bruijn graphs) of Trichinella taxa compared with H. contortus and T. canis.
5.2. Gene prediction

The prediction of genes is the most critical step in the annotation of a genome, and needs to be continuously improved to strive toward accurate genome annotation. Even though *ab initio* prediction methods have advanced, they are not perfect, and evidence data are required to achieve reliable gene predictions (see Section 2). High quality evidence data, such as full-length mRNA and/or protein sequences (from closely related taxa) can be combined with *ab initio* predictions as well as with the syntenic relationships of genes of related taxa. The latest gene predictors, utilising discriminative machine learning algorithms (see Section 2), might provide greatly improved gene predictions, but they are not yet widely used, and need to be enhanced to achieve high quality results.

In the gene prediction pipeline implemented for recent studies (Anstead et al., 2015, Jex et al., 2014, Korhonen, Pozio, 2015, Schwarz, Korhonen, 2013, Young et al., 2014, Zhu, Korhonen, 2015), a pragmatic approach was applied, in which established gene predictors and prediction combiners were used (see Sub-section 4.1). This pipeline is based on MAKER2 prediction-combiner implementation and was applied to the genomes of *H. contortus* and *Trichinella* (Korhonen, Pozio, 2015, Schwarz, Korhonen, 2013). For *H. contortus*, only assembled transcriptomes were used as evidence data, whereas for members of the *Trichinella* complex, available cDNAs and the proteomes of other nematode taxa were also included (Korhonen, Pozio, 2015). The genomic and transcriptomic data for *H. contortus* (Schwarz, Korhonen, 2013) were from distinct strains, which led to some discrepancies in the predictions of some genes. For *T. canis* (Zhu, Korhonen, 2015), the program GLEAN was used to combine the prediction results from multiple sources, but protein data were not employed as evidence data.

For all of these projects, considerable post-processing outside of the automated prediction pipeline was needed. Such ‘external’ processing was first applied to the *H. contortus* genome (Schwarz, Korhonen, 2013) and then gradually improved prior to application to the *T. canis* and *Trichinella* genomes (Korhonen, Pozio, 2015, Zhu, Korhonen, 2015). This processing contained steps to correct or remove genes with putatively erroneous exon structures, to remove nested and overlapping genes, and to filter out putative transposons and extraneous or “contaminating” sequences (Korhonen, Pozio, 2015, Zhu, Korhonen, 2015).

Despite extensive efforts, to date, there is no prediction combiner that is able to reliably combine the results from numerous sources, and there are no suitable methods for post-processing and the evaluation of the quality of gene prediction results such as genome-wide reliable full-length estimation even an attempt is made for instance by MAKER2 quality codes. Thus, there is a need to develop an improved approach that can, for instance, re-predict small sets of genes of interest by accounting for issues such as scaffold fragmentation and/or micro-exons, possibly by utilising comparative genomics (Wusirika and Shaik, 2014) and/or machine learning algorithms. In addition, the use of N-SCAN (Gross and Brent, 2006), accounting for genomic synteny among closely related species, would also increase the accuracy of predictions. However, fragmentation in genomic assemblies and consequent limitations in aligning genomes would complicate this approach. Therefore, producing substantially improved genomic assemblies using both long- and short-read sequence data produced by second- and third-generation sequencers has the potential to resolve this issue. For the evaluation of the quality of *de novo* assembled genomes, programs such as PAGIT (Swain et al., 2012), iCORN (Otto et al., 2010) and Pilon (Walker et al., 2014) are commonly applied.

5.3. Prediction of gene/protein function
In the functional annotation of the *H. contortus* genome (Schwarz, Korhonen, 2013), the InterPro results were combined with the BLAST results obtained from public databases, such as UniProt/SwissProt and NCBI reference sequences, and the classification of proteins into families followed the same approach as applied previously to the *A. suum* genome (Jex, Liu, 2011). This approach, which relied on text searching of gene annotations to identify the protein families, was not optimal. In subsequent projects (Korhonen, Pozio, 2015, Schwarz, Korhonen, 2013, Zhu, Korhonen, 2015), we elected to use an improved approach that employs KEGG BRITE protein families. In addition, for *T. canis* and *Trichinella* taxa, OrthoMCL and structural function predictions were used to confirm the predicted annotations for small subsets of proteins of biological interest (Korhonen, Pozio, 2015, Zhu, Korhonen, 2015).

Methods used for the functional annotation of genomes might propagate annotation errors found in public databases. Particularly for helminths, protein families can differ significantly from other eukaryotes for which annotated data sets exist in public databases, leading to mis-annotations of proteins that are variable in sequence (but perhaps not in structure and function) (Sub-section 2.3). Consequently, accurate HMM models for protein families of parasitic worms and machine learning classification methods should be applied to such proteins (Campos et al., 2014). Currently, many genes remain uncharacterized, and constitute ~29% of predicted genes in *H. contortus*, 22% in *T. canis* and 25-41% in *Trichinella* (depending on taxon) (Korhonen, Pozio, 2015, Schwarz, Korhonen, 2013, Zhu, Korhonen, 2015). These percentages of uncharacterised (unknown or orphan) genes are similar to those (28-35%) estimated for other *de novo*-assembled genomes of parasitic worms (Foth, Tsai, 2014, Ghedin, Wang, 2007, Jex, Liu, 2011, Mitreva, Jasmer, 2011, Srinivasan, Dillman, 2013, Tang, Gao, 2014). It is possible that the inability to annotate some genes might relate to genomic mis-assemblies or imprecise gene predictions. However, these orphans are considered true genes, as many display abundant transcription. This relatively large portion of orphan genes compromises post-genomic explorations, such as pathway analyses and drug discovery, and remains an unresolved and vexed issue. Comparative genomic techniques might help to predict functions for at least some orphan genes (Wusirika and Shaik, 2014).

### 5.4. Transcriptional analyses

We have conducted transcriptional analyses of various nematodes. In *H. contortus*, all developmental stages (egg, L1, L2, L3, L4 and adult as well as sexes) were explored (Schwarz, Korhonen, 2013), but we were not able to reliably assess differential transcription of genes between stages, because a lack of biological replicates; instead, an arbitrary threshold was used to identify “upregulated” genes in the transition from stage to stage, and putative key protein families relating to these genes were identified. However, we combined data for non-parasitic (egg, L1, L2 and L3) and parasitic (L4 and adults) groups to emulate replicates and identified statistically significant genes that were upregulated in each group (Schwarz, Korhonen, 2013). For *T. canis*, four biological replicates were used for each developmental stage and tissue studied (Zhu, Korhonen, 2015). In addition, the interpretation of differentially transcribed genes was improved through the implementation of a method that predicts the statistical significance for KEGG pathways and KEGG BRITE protein families found in *T. canis* (Zhu, Korhonen, 2015). However, the latter approach was not applicable to the *Trichinella* complex (Korhonen, Pozio, 2015), because the comparative analysis of transcription for *Trichinella* taxa was restricted to a subset of genes (i.e. single copy orthologs, SCOs), which was too limited to confidently assess the statistical significance of transcribed KEGG pathways and protein families. Consequently, the analysis was constrained to single occurrences of genes, such as those predicted to encode excretory/secretory (ES) proteins (Korhonen, Pozio, 2015). As for *H. contortus* (Schwarz, Korhonen, 2013), only one (pooled) sample per *Trichinella* taxon was available; therefore, groups of encapsulated (9 samples) and non-encapsulated (6 samples) taxa were formed to emulate two groups with biological
replicates (Korhonen, Pozio, 2015). The absence of a normalisation method for Trichinella taxa (with differing gene sets) was resolved by applying an exhaustive analysis of transcribed SCOs against three samples representing each clade, and then by selecting the most frequently occurring differentially transcribed genes as a final result (Korhonen, Pozio, 2015).

5.5. Secretomes and parasite-host interactions

In studies of parasitic nematodes, we have been particularly interested in investigating ES proteins in the secretome, which are likely to be involved in the interplay between the nematodes and their host animal. For instance, based on the presence of predicted signal peptides, we predicted totals of 1,457 and 870 ES proteins for H. contortus and T. canis, respectively (Schwarz, Korhonen, 2013, Zhu, Korhonen, 2015). Although these figures might seem somewhat high, these gene sets provide a basis for further studies of particular protein groups, such as the cysteine-rich secretory proteins, antigen 5 and pathogenesis-related 1 (CAP), NIM (abbreviation for: “not in Marleen’s data set) and transthyretin-like (TTL) proteins (Geldhof et al., 2005, Schwarz, Korhonen, 2013, Yatsuda et al., 2003); (Zhu, Korhonen, 2015). Interestingly, in T. canis, the highly transcribed genes ant-3, -5, -30 and -34 (Callister et al., 2008) were not detected in the genome, and might represent a symbiotic virus that assists T. canis to suppress the immune system of the host animal (Zhu, Korhonen, 2015). For members of the Trichinella complex, we stringently filtered ES molecules to exclude those targeting the nucleus or organelles in the cytosol (Korhonen, Pozio, 2015). The resultant numbers of ES proteins ranged from 314 to 414, and were significantly smaller than those estimated for H. contortus and T. canis. For Trichinella, we propose that some of these predicted ES proteins have key roles in the invasion of the muscle cell, subsequent uncoupling of mitochondria, and the suppression of antigen presentation and collagen production (Korhonen, Pozio, 2015). In our opinion, the gene sets predicted to encode ES proteins provide a useful starting point for future studies of parasite-host interactions.

5.6. Phylogenetic reconstruction

In a recent study (Korhonen, Pozio, 2015), we conducted post-genomic analyses to explore phylogenetic and biogeographical relationships of members of the Trichinella complex. First, we reconstructed the phylogeny using nuclear genomic data sets representing all SCOs (2,855) shared by 16 genomes, rather than the conventional approach of using ribosomal and/or mitochondrial DNA sequence data (Dorris, De Ley, 1999, Zarlenga et al., 2006). The (consensus) phylogenetic tree constructed using the SCO dataset was strongly supported (at all nodes), and future studies of larger numbers of isolates representing individual Trichinella taxa will assess the robustness of the topology of this tree.

Second, the analysis of the timescales of speciation events based on a molecular clock and using SCO data (Korhonen, Pozio, 2015) gave very similar results to the analyses conducted previously using nuclear ribosomal and mitochondrial DNA sequence data sets (Zarlenga, Rosenthal, 2006). However, the selection of a molecular clock model and taxon sampling tend to influence dating (Ho and Duchene, 2014). Moreover, being intracellular parasites, Trichinella taxa might have different, possibly more rapid, evolutionary rates compared with nematodes with a free-living phase in their life cycle, as indicated for endosymbiotic microbes (Bastolla et al., 2004). Consequently, to correctly evaluate speciation times, it would be beneficial to have a fossil record for Trichinella and/or other, closely related nematodes, but an accurate record will be challenging to discover.

Third, to illustrate synteny among the 16 Trichinella genomes, a minimum cross-over algorithm was implemented with Integer Linear Programming (ILP), to organise syntenic scaffolds between two genomes sharing at least 10 SCOs (Korhonen, Pozio, 2015). This approach eliminated the need to align the scaffolds of fragmented draft genomes and resulted in a useful display of synteny.
However, the limitations of this implementation are (i) a restriction to pairwise comparisons as well as (ii) an exponential increase in processing (computing) time required for increased resolution using a reduced number of SCOs shared among scaffolds. The algorithm could be extended to support analyses using multiple genomes simultaneously, and processing time could be improved by employing heuristic cross-over algorithms, but a display of exact synteny would not be achievable because of the fragmented nature of draft genome assemblies, in which the order and orientation of scaffolds is not known. However, to resolve latter issue, the algorithm could be used to propose anchor points of high confidence to close gaps between scaffolds of an assembly using a PCR-coupled approach, similar to that offered in the program ABACAS (Assefa et al., 2009). The resultant, longer scaffolds would facilitate the use of conventional algorithms in establishing synteny and could also assist through the adoption of gene prediction algorithms, such as N-SCAN (Gross and Brent, 2006). This approach might partly replace optical mapping (Dong et al., 2013, Shiguo et al., 2007) to resolve the order of scaffolds, but would only work for genomes of taxonomically closely related species.
6. Concluding remarks

Taken together, the assembly, gene prediction and functional annotation pipelines established and implemented (Section 4) were very successful, and produced consistent and high quality results for nematode genomes. Clearly, these genomes now provide a solid foundation for fundamental research in areas such as the pathogenesis of nematodiases, and should facilitate applied research toward the development of new interventions and diagnostic methods. However, to further improve the quality of assembly and annotation, third-generation sequencing technologies might be employed to produce long read sequence data to complement the assembly of short-read data sets in an enhanced, automated pipeline. In addition, further components could be integrated into the annotation pipelines, including extensive post-processing of predicted genes, an accurate gene re-prediction (targeting small sets of genes), and the use of phylogenetic, structural and machine learning prediction methods in functional annotation to account for orphan gene groups in parasitic worms, which are entirely new and different from those in current public databases. The genomes of parasitic nematodes studied to date (Table 12) encode large numbers of highly transcribed, yet uncharacterised genes, which might be critical for the survival of a worm or for parasitism and/or the parasite-host interplay (Korhonen, Pozio, 2015, Schwarz, Korhonen, 2013, Zhu, Korhonen, 2015). To further study such genes, carefully designed biochemical assay together with sophisticated bioinformatic analyses are essential to unravel their functions and roles in biological pathways. Studies of such molecules using metabolic and proteomic tools should also provide complementary data sets, which could be integrated into worm-specific databases essential for post-genomic analyses. Understanding such orphan molecules could provide a paradigm shift in our understanding of parasitism, and could enable the design of radically new strategies to treat and control parasitic nematodes. In the future, third-generation sequencing should support improved fundamental investigations of the population genetics, systematics of nematodes and the evolution/genetics of drug resistance.

In conclusion, the genomic and transcriptomic data sets, together with the software pipelines for genome assembly, annotation and analyses developed recently, provide significant resources for the scientific community to accelerate genome-wide molecular discovery in parasitic worms as well as approaches to explore parasite-host interactions and disease, and could lead to a considerable improvement in our understanding of these parasites at the molecular level. These resources should also underpin applied areas, including the design of new drugs, vaccines and diagnostics. Importantly, it is expected that the in silico tools and pipelines established will be applicable or readily adaptable to a broad range of socioeconomically important parasitic worms and other eukaryotic pathogens.

Acknowledgements

Research funding from the National Health and Medical Research Council (NHMRC) and Australian Research Council (ARC) is gratefully acknowledged (R.B.G.). Other support from the Australian Academy of Science, the Australian-American Fulbright Commission, Alexander von Humboldt Foundation, Melbourne Water Corporation as well as the Victorian Life Sciences Computation Initiative (VLSCI) and WormBase (www.wormbase.org) is gratefully acknowledged. N.D.Y is an NHMRC Early Career Research (ECR) Fellow.
References


Harris, R.S. Improved pairwise alignment of genomic DNA. ProQuest, 2007.: The Pennsylvania State University; 2007.


Lane, J., Jubb, T., Shephard, R., Webb-Ware, J., Fordyce, G. 2015, Priority list of endemic diseases for the red meat industries. North Sydney, NSW 2059, Australia: Meat and Livestock Australia Limited.


Van Dongen, S. Graph clustering by flow simulation. The Netherlands: University of Utrecht; 2000.


**Table 1**

Second-generation sequencing technologies.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Vendor</th>
<th>Technology*</th>
<th>Read length (bp)</th>
<th>Cost (USD/Gbp)</th>
<th>Error type +</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA</td>
<td>Solexa</td>
<td>BR</td>
<td>35</td>
<td>2,300</td>
<td>S</td>
</tr>
<tr>
<td>SOLiD</td>
<td>ABI</td>
<td>EL</td>
<td>35</td>
<td>850</td>
<td>A</td>
</tr>
<tr>
<td>GS-FLX</td>
<td>Roche</td>
<td>EP</td>
<td>238</td>
<td>85,000</td>
<td>I</td>
</tr>
<tr>
<td>SOLiD 5500xl</td>
<td>ABI</td>
<td>EL</td>
<td>75</td>
<td>80</td>
<td>A</td>
</tr>
<tr>
<td>HiSeq2000</td>
<td>Illumina</td>
<td>BR</td>
<td>100</td>
<td>40</td>
<td>S</td>
</tr>
<tr>
<td>GS-FLX Titanium XL+</td>
<td>Roche</td>
<td>EP</td>
<td>700</td>
<td>7,000</td>
<td>I</td>
</tr>
</tbody>
</table>

* E = emulsion PCR, B = bridge PCR, P = pyrosequencing, R = reversible terminators, L = ligation-based sequencing
+ I = indel, A = AT-bias, S = substitution
Table 2
Key second- and third-generation sequencing technologies.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Vendor</th>
<th>Read length (bp)</th>
<th>Sequencer cost (USD)</th>
<th>Sequencing cost (USD/Gbp)</th>
<th>Error type+</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>GS Junior</td>
<td>Roche (454)</td>
<td>400</td>
<td>108,000</td>
<td>22,000</td>
<td>I</td>
<td>2010</td>
</tr>
<tr>
<td>MiSeq</td>
<td>Illumina</td>
<td>150</td>
<td>125,000</td>
<td>740</td>
<td>S</td>
<td>2011</td>
</tr>
<tr>
<td>IonTorrent</td>
<td>Life Technologies</td>
<td>&lt; 400</td>
<td>50,000</td>
<td>&lt; 925</td>
<td>I</td>
<td>2012</td>
</tr>
<tr>
<td>PacBio RS</td>
<td>ABI</td>
<td>4,600</td>
<td>695,000</td>
<td>1,500</td>
<td>I</td>
<td>2012</td>
</tr>
<tr>
<td>Moleculo Long Read</td>
<td>Illumina</td>
<td>&lt; 10k*</td>
<td>HiSeq machine</td>
<td>-</td>
<td>S</td>
<td>2013</td>
</tr>
<tr>
<td>MiniON</td>
<td>Oxford Nanopore™</td>
<td>&lt; 100k</td>
<td>&lt; 10,000</td>
<td>Very low</td>
<td>IS</td>
<td>2014</td>
</tr>
</tbody>
</table>

* The “reads” in Moleculo Long Read technology are local contig assemblies.
+ I = indel, A = AT-bias, S = substitution.
Table 3
Comparison of genome assembly algorithms.

<table>
<thead>
<tr>
<th>Name</th>
<th>Category*</th>
<th>PE support</th>
<th>Distinguishing feature</th>
<th>Read fix**</th>
<th>Data support*</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEQAID</td>
<td>OLC</td>
<td>No</td>
<td>First graph based</td>
<td>-</td>
<td>S</td>
<td>1984</td>
</tr>
<tr>
<td>TIGR</td>
<td>OLC</td>
<td>No</td>
<td>k-mer alignment</td>
<td>-</td>
<td>S</td>
<td>1995</td>
</tr>
<tr>
<td>CAP3</td>
<td>OLC</td>
<td>Yes</td>
<td>PE support</td>
<td>P</td>
<td>S</td>
<td>1999</td>
</tr>
<tr>
<td>Celera</td>
<td>OLC</td>
<td>Yes</td>
<td>Unitig concept</td>
<td>P</td>
<td>S</td>
<td>2000</td>
</tr>
<tr>
<td>Newbler</td>
<td>OLC</td>
<td>Yes</td>
<td>454 Life Sciences assembler</td>
<td>IP</td>
<td>P</td>
<td>2005</td>
</tr>
<tr>
<td>CABOG</td>
<td>OLC</td>
<td>Yes</td>
<td>Sanger and SOLiD support</td>
<td>P</td>
<td>RS</td>
<td>2008</td>
</tr>
<tr>
<td>SSAKE</td>
<td>Greedy</td>
<td>No</td>
<td>First short read assembler</td>
<td>-</td>
<td>I</td>
<td>2007</td>
</tr>
<tr>
<td>SHARCGS</td>
<td>Greedy</td>
<td>No</td>
<td>Read filtering</td>
<td>P</td>
<td>I</td>
<td>2007</td>
</tr>
<tr>
<td>EULER</td>
<td>DBG</td>
<td>MT</td>
<td>DBG repeat handling</td>
<td>GPS</td>
<td>IRS</td>
<td>2001</td>
</tr>
<tr>
<td>Velvet</td>
<td>DBG</td>
<td>GB</td>
<td>Long read threading</td>
<td>GS</td>
<td>I</td>
<td>2008</td>
</tr>
<tr>
<td>ALLPATHS</td>
<td>DBG</td>
<td>GB</td>
<td>Large genome support</td>
<td>GPS</td>
<td>I</td>
<td>2008</td>
</tr>
<tr>
<td>ABySS</td>
<td>DBG</td>
<td>MT and GB</td>
<td>Full parallel processing</td>
<td>GPS</td>
<td>I</td>
<td>2009</td>
</tr>
<tr>
<td>SOAPdenovc</td>
<td>DBG</td>
<td>GB</td>
<td>Compact graph structure</td>
<td>GPS</td>
<td>I</td>
<td>2010</td>
</tr>
<tr>
<td>SPAdes</td>
<td>DBG</td>
<td>Build-in</td>
<td>paired assembly graphs</td>
<td>GPS</td>
<td>I</td>
<td>2012</td>
</tr>
<tr>
<td>SGA</td>
<td>OLC</td>
<td>Yes</td>
<td>FM indexed string graph</td>
<td>P</td>
<td>I</td>
<td>2012</td>
</tr>
<tr>
<td>PBcR</td>
<td>OLC</td>
<td>Yes</td>
<td>First to support 3rd gen.</td>
<td>PIS</td>
<td>SIPRO</td>
<td>2012</td>
</tr>
</tbody>
</table>

**** OLC = Overlap-layout-consensus, DBG = de Bruijn Graph
*** MT = Mate threading, GB = Graph based
** I = indel, A = AT-bias, S = substitution, D = GC detection, P = preprocessing, G = graph simplification
* I = Solexa, Illumina, R = Roche 454, S = Sanger, P = PacBio
Table 4
Key *ab initio* gene prediction algorithms.

<table>
<thead>
<tr>
<th>Name</th>
<th>Content sensors</th>
<th>Signal sensors</th>
<th>Features</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>GENSCAN</td>
<td>GHMM</td>
<td>WMs, MDD</td>
<td>First to predict multiple genes</td>
<td>1997</td>
</tr>
<tr>
<td>GeneID</td>
<td>GHMM</td>
<td>PWM</td>
<td>DP gene structure prediction</td>
<td>2000</td>
</tr>
<tr>
<td>TWINSCAN</td>
<td>GHMM</td>
<td>-</td>
<td>Conservation probability</td>
<td>2001</td>
</tr>
<tr>
<td>AUGUSTUS</td>
<td>IMM</td>
<td>WWAM, HMM</td>
<td>Improved intron model</td>
<td>2003</td>
</tr>
<tr>
<td>GlimmerHMM</td>
<td>IMM</td>
<td>WMs, MDD</td>
<td>Uses GeneSplicer and GlimmerM</td>
<td>2004</td>
</tr>
<tr>
<td>SNAP</td>
<td>GHMM</td>
<td>WAM, WWM</td>
<td>User friendly training procedure</td>
<td>2004</td>
</tr>
<tr>
<td>GeneMark-ES</td>
<td>HSMM</td>
<td>-</td>
<td>Unsupervised training</td>
<td>2005</td>
</tr>
<tr>
<td>N-SCAN</td>
<td>GHMM</td>
<td>-</td>
<td>Phylogenetic Bayes networks</td>
<td>2006</td>
</tr>
<tr>
<td>CONTRAST</td>
<td>CRF</td>
<td>SVM</td>
<td>Combines signals to gene structure</td>
<td>2007</td>
</tr>
<tr>
<td>mGene</td>
<td>SVM</td>
<td>SVM</td>
<td>Structure prediction by HSM SVM</td>
<td>2009</td>
</tr>
</tbody>
</table>
Table 5
Key evidence data aligner implementations.

<table>
<thead>
<tr>
<th>Name</th>
<th>Category</th>
<th>Features</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>EST_GENOME</td>
<td>BLAST seeds</td>
<td>Accurate but slow</td>
<td>1997</td>
</tr>
<tr>
<td>AAT</td>
<td>BLAST seeds</td>
<td>Nucleotide and amino acid support</td>
<td>1997</td>
</tr>
<tr>
<td>GeneWise</td>
<td>Pair HMM</td>
<td>Integrated to Ensembl annotation system</td>
<td>2004</td>
</tr>
<tr>
<td>Exonerate</td>
<td>BLAST seeds</td>
<td>Supports both nucleotides and amino acids</td>
<td>2005</td>
</tr>
<tr>
<td>Pairagon</td>
<td>Pair HMM</td>
<td>Uses WMM to accurately align splice sites</td>
<td>2009</td>
</tr>
<tr>
<td>Name</td>
<td>Category</td>
<td>Features</td>
<td>Year</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
<td>-----------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>GenomeScan</td>
<td>Predicting</td>
<td>GHMM with evidence by BLASTX</td>
<td>2001</td>
</tr>
<tr>
<td>TWINSCAN-EST</td>
<td>Predicting</td>
<td>Nucleotide evidence data</td>
<td>2006</td>
</tr>
<tr>
<td>N-SCAN-EST</td>
<td>Predicting</td>
<td>Nucleotide evidence data</td>
<td>2006</td>
</tr>
<tr>
<td>AUGUSTUS+</td>
<td>Predicting</td>
<td>Nucleotide evidence data</td>
<td>2006</td>
</tr>
<tr>
<td>COMBINER</td>
<td>Combining</td>
<td>Linear and statistical combination</td>
<td>2004</td>
</tr>
<tr>
<td>Ensembl</td>
<td>Combining</td>
<td>Prefers evidence data</td>
<td>2004</td>
</tr>
<tr>
<td>JIGSAW</td>
<td>Combining</td>
<td>GHMM</td>
<td>2005</td>
</tr>
<tr>
<td>GLEAN</td>
<td>Combining</td>
<td>Latent Component Analysis (LCA)</td>
<td>2007</td>
</tr>
<tr>
<td>Evigan</td>
<td>Combining</td>
<td>Bayes network with ML parameter estimation</td>
<td>2008</td>
</tr>
<tr>
<td>MAKER2</td>
<td>Combining</td>
<td>Annotation Edit Distance (AED)</td>
<td>2008</td>
</tr>
</tbody>
</table>
Table 7
Clustering and phylogenomic annotation algorithms.

<table>
<thead>
<tr>
<th>Name</th>
<th>Inferring orthology*</th>
<th>Pairwise similarity matrix</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>InParanoid</td>
<td>PR</td>
<td>BLAST based</td>
<td>2001</td>
</tr>
<tr>
<td>OrthoMCL</td>
<td>MCL</td>
<td>BLAST based</td>
<td>2003</td>
</tr>
<tr>
<td>MultiParanoid</td>
<td>PR</td>
<td>Uses InParanoid</td>
<td>2006</td>
</tr>
<tr>
<td>OrthoDB</td>
<td>TR</td>
<td>Smith-Waterman (3-reciprocal hits)</td>
<td>2013</td>
</tr>
<tr>
<td>KOG</td>
<td>TR</td>
<td>BLAST based (3-reciprocal hits)</td>
<td>2003</td>
</tr>
<tr>
<td>SIMAP</td>
<td>-</td>
<td>FASTA and Smith-Waterman. Very large</td>
<td>2010</td>
</tr>
<tr>
<td>eggNOG</td>
<td>TD</td>
<td>Uses SIMAP</td>
<td>2008</td>
</tr>
<tr>
<td>SYNERGY</td>
<td>TD</td>
<td>FASTA weighed by phylogenetic distance</td>
<td>2007</td>
</tr>
<tr>
<td>PhIG</td>
<td>TF</td>
<td>BLAST weighed by phylogenetic distance</td>
<td>2006</td>
</tr>
<tr>
<td>TreeFam</td>
<td>TF</td>
<td>Uses PhIG</td>
<td>2006</td>
</tr>
<tr>
<td>PANTHER</td>
<td>TF</td>
<td>Profile HMM</td>
<td>2013</td>
</tr>
</tbody>
</table>

* T = uses taxonomy to infer orthologs and paralogs, D = uses pairwise phylogenetic distance,
  F = uses phylogenetic distance between protein families, MCL = Markov Clustering Algorithm,
  PR = Pairwise reciprocal, TR = Three-wise reciprocal.
**Table 8**
Protein structure based functional annotation methods.

<table>
<thead>
<tr>
<th>Name</th>
<th>Pairwise similarity matrix</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phyre2</td>
<td>Fast structure and function prediction</td>
<td>2003, 2015</td>
</tr>
<tr>
<td>JESS</td>
<td>Prediction of conserved residues in 3D space</td>
<td>2003</td>
</tr>
<tr>
<td>I-TASSER</td>
<td>Winner of CASP competition over many years</td>
<td>from 2006</td>
</tr>
<tr>
<td>ConFunc</td>
<td>Predicts conserved residues among GO enrichments</td>
<td>2008</td>
</tr>
<tr>
<td>COFACTOR</td>
<td>Function prediction module in I-TASSER</td>
<td>2012</td>
</tr>
</tbody>
</table>
Table 9  
Key programs implementing NJ, MP, ML, BI and molecular clock methods.

<table>
<thead>
<tr>
<th>Program</th>
<th>Phylogenetic methods</th>
<th>Molecular clocks</th>
<th>Publication</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NJ</td>
<td>MP</td>
<td>ML</td>
</tr>
<tr>
<td>PHYLIP</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>PAUP</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>MEGA</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>RAxML</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>PhyML</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>MrBayes</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>PhyloBayes</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>BEAST</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>PAML</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
</tbody>
</table>

* NJ = neighbour joining, MP = maximum parsimony, ML = maximum likelihood, BI = Bayesian inference
** SC = strict clock, AC = autocorrelated relaxed clock, UC = uncorrelated relaxed clock
### Table 10
Key programs commonly used to resolve and/or display synteny.

<table>
<thead>
<tr>
<th>Program</th>
<th>Publication</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoGraph</td>
<td>Derrien et al. (2007)</td>
<td>Web service. Limited to three user defined species.</td>
</tr>
<tr>
<td>FISH</td>
<td>Calabrese et al. (2003)</td>
<td>Web service. Limited to three user defined species.</td>
</tr>
<tr>
<td>Mummer</td>
<td>Kurtz et al. (2004)</td>
<td>Genome wide alignment. Limited to two genomes only.</td>
</tr>
<tr>
<td>MCScanx</td>
<td>Wang et al. (2012b)</td>
<td>Command line. Requires blast results and gene coordinates.</td>
</tr>
<tr>
<td>Circos</td>
<td>Krzywinski et al. (2009)</td>
<td>Visualization program. Supports only circular chromosomes.</td>
</tr>
</tbody>
</table>
Table 11
Metabolic pathway databases and programs for metabolic pathway analyses.

<table>
<thead>
<tr>
<th>Database of program</th>
<th>Features*</th>
<th>Publication or URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>KEGG</td>
<td>I</td>
<td>Kanehisa, Goto (2012)</td>
</tr>
<tr>
<td>MetaCyc</td>
<td>I</td>
<td>Caspi, Altman (2014)</td>
</tr>
<tr>
<td>BioCyc</td>
<td>I</td>
<td>Caspi, Altman (2014)</td>
</tr>
<tr>
<td>ConsensusPathDB</td>
<td>I</td>
<td>Kamburov, Pentchev (2011)</td>
</tr>
<tr>
<td>Reactome</td>
<td>I</td>
<td>Croft, Mundo (2014)</td>
</tr>
<tr>
<td>IPA™</td>
<td>N</td>
<td><a href="http://www.qiagen.com/ingenuity">http://www.qiagen.com/ingenuity</a></td>
</tr>
<tr>
<td>MetaCore™</td>
<td>N</td>
<td><a href="http://lsresearch.thomsonreuters.com">http://lsresearch.thomsonreuters.com</a></td>
</tr>
<tr>
<td>RouteSearch</td>
<td>S</td>
<td>Latendresse, Krummenacker (2014)</td>
</tr>
<tr>
<td>FMM</td>
<td>S</td>
<td>Chou et al. (2009)</td>
</tr>
<tr>
<td>RetroPath</td>
<td>S</td>
<td>Carbonell et al. (2011)</td>
</tr>
</tbody>
</table>

* I = pathway identification, N = novel pathway prediction, S = synthetic pathway prediction
<table>
<thead>
<tr>
<th>Nematode</th>
<th>Publication</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pristionchus pacificus</td>
<td>Dieterich et al. (2008)</td>
<td>2008</td>
</tr>
<tr>
<td>Meloidogyne incognica</td>
<td>Abad et al. (2008)</td>
<td>2008</td>
</tr>
<tr>
<td>Meloidogyne hapla</td>
<td>Opperman et al. (2008)</td>
<td>2008</td>
</tr>
<tr>
<td>Caenorhabditis angaria</td>
<td>Mortazavi et al. (2010)</td>
<td>2010</td>
</tr>
<tr>
<td>Trichinella spiralis</td>
<td>Mitreva, Jasmer (2011)</td>
<td>2011</td>
</tr>
<tr>
<td>Bursaphelenchus xylophilus</td>
<td>Kikuchi et al. (2011)</td>
<td>2011</td>
</tr>
<tr>
<td>Dirofilaria immitis</td>
<td>Godel et al. (2012)</td>
<td>2012</td>
</tr>
<tr>
<td>Haemonchus contortus</td>
<td>Laing et al. (2013), Schwarz, Korhonen (2013)</td>
<td>2013</td>
</tr>
<tr>
<td>Heterorhabditis bacteriophora</td>
<td>Bai et al. (2013)</td>
<td>2013</td>
</tr>
<tr>
<td>Loa loa</td>
<td>Desjardins et al. (2013)</td>
<td>2013</td>
</tr>
<tr>
<td>Panagrellus redivivus</td>
<td>Srinivasan, Dillman (2013)</td>
<td>2013</td>
</tr>
<tr>
<td>Romanomermis culicivorax</td>
<td>Schiffer et al. (2013)</td>
<td>2013</td>
</tr>
<tr>
<td>Meloidogyne floridensis</td>
<td>Lunt et al. (2014)</td>
<td>2014</td>
</tr>
<tr>
<td>Trichuris muris</td>
<td>Foth, Tsai (2014)</td>
<td>2014</td>
</tr>
<tr>
<td>Trichuris trichiura</td>
<td>Foth, Tsai (2014)</td>
<td>2014</td>
</tr>
<tr>
<td>Trichuris suis</td>
<td>Jex, Nejsum (2014)</td>
<td>2014</td>
</tr>
<tr>
<td>Ancylostoma ceylanicum</td>
<td>Schwarz et al. (2015)</td>
<td>2015</td>
</tr>
</tbody>
</table>
Fig. 1. Structural elements of a gene. Exons encompass both transcribed untranslated regions (UTRs: 5’-UTR, 3’-UTR, light grey) and coding sequences (CDSs, dark grey) whereas introns between exons are regions that are spliced from the gene product.
**Sequence**: Panther, PIRSF, TIGRFAM, HAMAP

**Domain**: Pfam, SMART, ProDom, SUPERFAMILY, Gene3D, ProSite (profile)

**Motif**: ProSite

**Fingerprint**: PRINTS

**Signal peptide**: SignalP

**Transmembrane region**: TMHMM

**Fig. 2.** Sequence (light grey), domain (grey), motif (dark grey) and fingerprint databases, integrated in InterPro together with signal peptide and transmembrane region (black), are shown in the context of a protein sequence.
Fig. 3. Flowchart of genome assembly pipeline used in our laboratory. Parallelograms represent the data; rectangles represent data processing. Dashed lines represent the data flow and solid lines the flow of the data processing.
Fig. 4. Flowchart of gene prediction and functional annotation pipelines used in our laboratory. Parallelograms represent the data; rectangles represent data processing. Dashed lines represent the data flow and solid lines the flow of the data processing.
Author/s:
Korhonen, PK; Young, ND; Gasser, RB

Title:
Making sense of genomes of parasitic worms: Tackling bioinformatic challenges

Date:
2016

Citation:

Persistent Link:
http://hdl.handle.net/11343/124176