Genomics and Phylogeny of Cytoskeletal Proteins: Tools and Analyses

Dissertation

zur Erlangung des mathematisch-naturwissenschaftlichen Doktorgrades

„Doctor rerum naturalium“

der Georg-August-Universität

vorgelegt von

Björn Hammesfahr

aus Ludwigsburg

Göttingen

2012

Thesis Committee:

PD Dr. Martin Kollmar (Referent)

AG System-Biologie von Motorproteinen,

Max-Planck-Institut für Biophysikalische Chemie

Göttingen

Prof. Dr. Burkhard Morgenstern (Co-Referent)

Institut für Mikrobiologie und Genetik,

Abteilung Bioinformatik,

Georg-August-Universität

Göttingen

Dr. Dirk Fasshauer

Associate Professor

Department of Cell Biology and Morphology

Faculty of Biology and Medicine

University of Lausanne

1005 Lausanne

Switzerland

Tag der mündlichen Prüfung: 05.11.2012

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig angefertigt habe. Es wurden nur die in der Arbeit ausdrücklich benannten Quellen und Hilfsmittel benutzt. Wörtlich oder sinngemäß übernommenes Gedankengut habe ich als solches kenntlich gemacht.

Datum, Ort	Unterschrift

Talks:

March 2010
Alpbach Meeting, Myotin & Muscle and Many other Motors
“CyMoBase – The Reference Database For Cyotskeletal And Motor Proteins”

Poster presentations:

February 2010
Biophysical Society 54th Annual Meeting, San Francisco (USA)
“CyMoBase – The Reference Database For Cytoskeletal And Motor Proteins”

March 2010
Alpbach Meeting, Myosin & Muscle and Many other Motors, Alpbach (Austria)
“CyMoBase – The Reference Database For Cyotskeletal And Motor Proteins”

October 2010
2nd Bio-IT World Europe, Hannover (Germany)
“CyMoBase – The Reference Database For Cyotskeletal And Motor Proteins”

July 2011
ISMB/ECCB, Vienna (Austria)
“CyMoBase and diArk - Resources for cytoskeletal and motor protein sequence information and eukaryotic genome research”

October 2011
3rd Bio-IT World Europe
“diArk and CyMoBase – Resources for eukaryotic genome research, and cytoskeletal and motor protein sequence information”

Table of contents

Erklärung 5

Table of contents 7

1 Introduction 11
1.1 Bits of Life 11
1.1.1 Genome 11
1.1.2 Gene 14
1.2 Cytoskeletal Proteins 16
1.2.1 Cytoskeleton 16
1.2.2 Microfilaments 16
1.2.3 Intermediate filaments 17
1.2.4 Microtubules 17
1.2.5 Motor proteins 17
1.3 Protein Family Analyses 20
1.3.1 Why analysing a protein family? 20
1.3.2 How analysing a protein family? 20
1.3.3 Issues during a protein family analysis 21
1.3.4 Correctly predict and assemble proteins 22
1.3.5 Phylogenetic analysis 23
1.4 Databases and Web applications 25
1.4.1 Database 25
1.4.2 Web application 25
1.4.3 Development 26
1.4.4 Deployment 27
1.4.5 Set up a new database 27
1.4.6 NMR 28

2 Publications 30
2.1 Evolution of the eukaryotic dynactin complex, the activator of cytoplasmic dynein 30
2.1.1 Abstract 30
2.1.2 Background 31
2.1.3 Results 33
2.1.4 Discussion 49
2.1.5 Conclusions 57
2.1.6 Methods 57
2.1.7 Competing interests 59
2.1.8 Authors' contributions 59
2.1.9 Acknowledgements 60

2.1.10 Additional files 60
2.1.10.1 Additional file 1 as ZIP 60
2.1.10.2 Additional file 2 as PDF 60
2.1.10.3 Additional file 3 as PDF 60
2.1.10.4 Additional file 4 as PDF 60
2.1.10.5 Additional file 5 as PDF 61
2.1.10.6 Additional file 6 as PDF 61
2.1.10.7 Additional file 7 61
2.1.10.8 Additional file 8 as PDF 64
2.2 A holistic phylogeny of the coronin gene family reveals an ancient origin of the tandem-coronin, defines a new subfamily, and predicts protein function 65
2.2.1 Abstract 65
2.2.2 Background 66
2.2.3 Results 67
2.2.4 Discussion 82
2.2.5 Conclusions 86
2.2.6 Methods 86
2.2.7 Acknowledgements 88
2.2.8 Authors' contributions 88
2.2.9 Additional files 88
2.2.9.1 Additional file 1 – Sequence alignment of the coronins 88
2.2.9.2 Additional file 2 – MrBayes tree of the coronin family 88
2.2.9.3 Additional file 3 – RAxML tree of the coronin family 89
2.2.9.4 Additional file 4 – Coronin repertoire of all eukaryotes analyzed 89
2.2.9.5 Additional file 5 – Conserved residues in the coronin domain 89
2.3 diArk 2.0 provides detailed analyses of the ever increasing eukaryotic genome sequencing data 90
2.3.1 Abstract 90
2.3.2 Background 91
2.3.3 Methods 93
2.3.4 Results and Discussion 98
2.3.5 Conclusions 106
2.3.6 Availability and Requirements 106
2.3.7 Competing interests 107
2.3.8 Authors’ contributions 107
2.3.9 Acknowledgements and Funding 107
2.3.10 Additional files 107
2.3.10.1 Additional file 1 - Database scheme. 107
2.4 Cross-species protein sequence and gene structure prediction with fine-tuned Webscipio 2.0 and Scipio 108
2.4.1 Abstract 108
2.4.2 Background 109
2.4.3 Methods 110

2.4.4 Results and Discussion 115
2.4.5 Conclusions 136
2.4.6 Availability and requirements 136
2.4.7 List of abbreviations 137
2.4.8 Competing interests 137
2.4.9 Authors' contributions 137
2.4.10 Acknowledgements and Funding 137
2.4.11 Additional files 138
2.4.11.1 Additional file 1 – Activity flow of the hit processing step 138
2.4.11.2 Additional file 2 – Protein – DNA alignments corresponding to the example searches 138
2.4.11.3 Additional file 3 – Table with detailed data of the results of the cross-species search of the human DHC genes in the elephant genome. 138
2.4.11.4 Additional file 4 – Detailed evaluation values used for Tables 2, 3, and 4. 138
2.4.11.5 Additional file 5 – Software versions and run parameters of the gene reconstruction and prediction tools 138
2.5 Predicting mutually exclusive spliced exons based on exon length, splice site and reading frame conservation, and exon sequence homology 139
2.5.1 Abstract 139
2.5.2 Background 140
2.5.3 Methods 142
2.5.4 Results and Discussion 147
2.5.5 Conclusions 161
2.5.6 Abbreviations 162
2.5.7 Authors' contributions 162
2.5.8 Acknowledgements 162
2.5.9 Additional files 162
2.5.9.1 Additional file 1 - Detailed activity diagram 162
2.5.9.2 Additional file 2 – Search for non-mutually exclusive exons sharing similar length, same reading frame and sequence homology 162

3 Manuscripts in revision 163
3.1 Peakr: Predicting solid-state NMR spectra of proteins 163
3.1.1 Abstract 163
3.1.2 Introduction 164
3.1.3 Methods 165
3.1.3.1 Protein sequences 165
3.1.3.2 Chemical shifts 165
3.1.3.3 Conformations 167
3.1.3.4 Correlations 167
3.1.3.5 Calculated Spectra 169
3.1.3.6 Experimental Spectra 169
3.1.3.7 Output 169
3.1.4 Results and Discussion 170

3.1.5 Conclusions 175
3.1.6 Acknowledgements 175
3.2 ShereKhan – Calculating exchange parameters in relaxation dispersions data from CPMG experiments 176
3.2.1 Abstract 176
3.2.2 Introduction 176
3.2.3 Features 177
3.2.3.1 Applying different exchange regimes 178
3.2.3.2 Displaying and exporting results 178
3.2.4 Implementation 179
3.2.5 Acknowledgements 179
3.3 GenePainter: Aligning gene structures for phylogenetic analyses 180
3.3.1 Abstract 180
3.3.2 Background 181
3.3.3 Implementation 182
3.3.4 Results and Discussion 183
3.3.5 Conclusion 188
3.3.6 Competing interests 189
3.3.7 Availability and requirements 189
3.3.8 Authors' contributions 189
3.3.9 Acknowledgements 189
3.3.10 Additional files 189
3.3.10.1 Additional file 1 189

4 Conclusions 191

5 Acknowledgements 197

A Appendix 198

A.1 References 198
A.2 Curriculum vitae 222

1 Introduction

1.1 Bits of Life

1.1.1 Genome

During evolution, organism got more and more complex. Today, prokaryotes, archaea, and eukaryotes populate this planet. On one hand there are organisms composed of only one cell, while on the other hand they are composed of millions of cells. The necessary mechanisms to build up a cell are complicated. In the course of time, many proteins with different structure, function, size, etc. developed. The blueprint of all of them is stored within the genome as Deoxyribonucleic acid (DNA). The DNA is composed of two long chains forming a double helix. Each chain consists of four different types of nucleotide subunits. Each of them being composed of a sugar, a phosphate, and one of four bases: adenine, thymine, guanine, and cytosine. The chains are complementary to each other, with adenine and thymine forming two hydrogen bonds, whereas guanine and cytosine bind to each other via three hydrogen bonds.

The sizes of the genomes differ significantly depending on the organism. For eukaryotes, its size varies from ~2Mbp for *Encephalitozoon intestinalis* to ~3500Mbp for *Monodelphis domestica* (Figure 1.1-1).

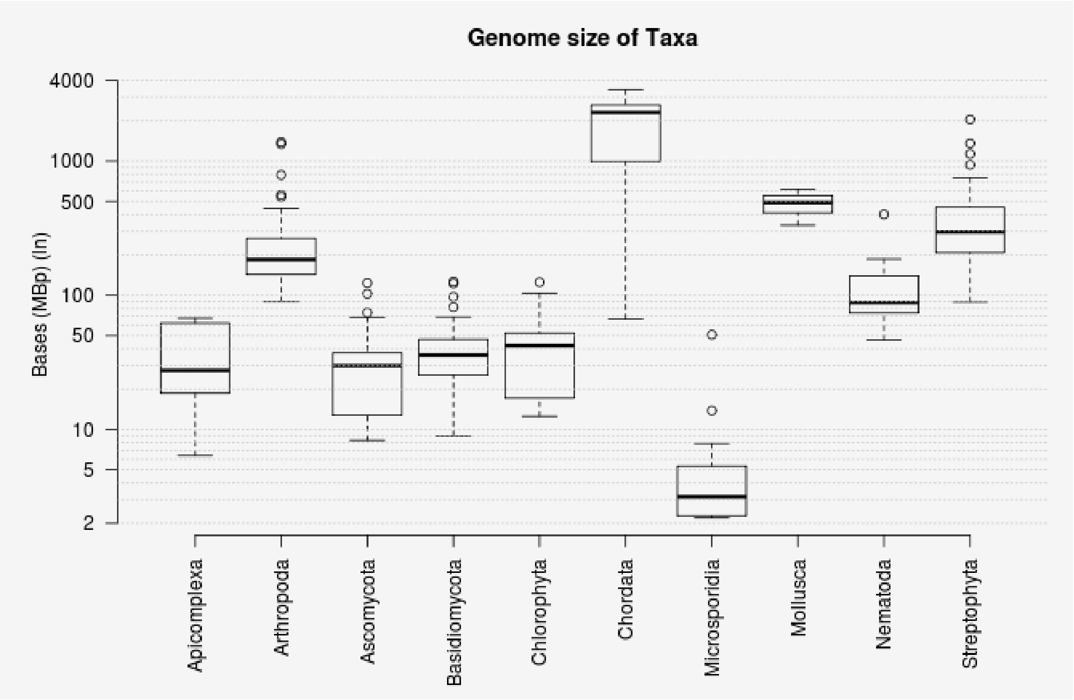


Figure 1.1-1: ***Genome size of Taxa****. This figure illustrates the size of all genomes available at diArk for different eukaryotic taxa. The smallest genome was found in Microsporidia, whereas the larges genome was found in Chordata. The y-axis is scaled logarithmically.*

All these cells express different kind of proteins. Some of them are acting directly within the cell (e.g. Dynein). Some of them form large complexes (e.g. Dynactin (1)). But all these proteins have a common feature that their blueprints have to be stored. Protein encoding parts of the genome are called genes.

One gene was not invented independently in different organisms. Genes were passed on to the offspring. During evolution, different kinds of mutations occur: mutations of just one base; deletion or insertion of parts of genes; gene duplications, or gene loss; genome duplication events. Over time, many different organisms appeared and vanished. But all of them have many proteins in common. This conservation of one protein in many different organisms is called a protein family.

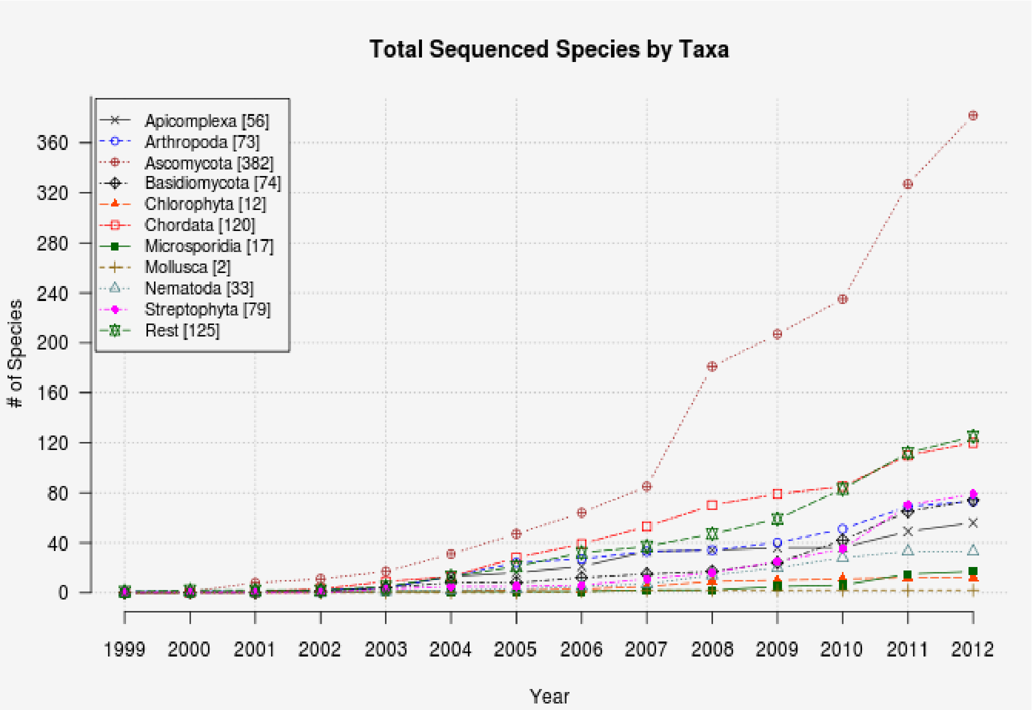


Figure 1.1-2: Completely sequenced eukaryotic organisms between 1999 and 2012, sorted by taxa.

In the year 1996, the first completely sequenced eukaryotic genome of *Saccharomyces cerevisiae* was published (2). Six years later, in the year 2002, about 20 eukaryotic genomes were decoded (Figure 1.1-2). All these genomes were sequenced by the Sanger chain-terminator method (3). Most of them were representatives of the fungi branch, except for the vase tunicate *Ciona intestinalis* (4) and three Cordata, namely the mouse *Mus musculus* (5), the pufferfish *Takifugu rubripes* (6), and *Homo sapiens* (7,8). Based on this small number of genomes, it was not possible to investigate the evolution of most protein family distributed over the eukaryotic tree of life. But time has changed. During the last 10 years, the number of available genomes increased nearly exponentially. Venter et al. developed a so called shotgun sequencing method (9), which allowed a four times faster sequencing of the human genome than the International Human Genome Project needed (10), reducing the sequencing time from 12 to three years.

Around the year 2007, new sequencing techniques, so called Next-generation sequencing (NGS), were developed. These techniques further improved sequencing speed and as a result reduced the costs for decoding a eukaryotic genome (Figure 1.1-3). Today, the most important ones are Illumina and Roche 454 sequencing. But still, the Sanger method is frequently use, especially for parts of the genome with repeating sequence elements.

Nowadays, the sequencing of genomes is fast. The combination of the incoming sequenced read to contigs, supercontigs, or even to chromosomal assemblies is difficult. Thereby, coverage is an important factor. By means of the Sanger technique, only a small coverage (6-10) is needed to combine reads to contigs and supercontigs, while NGS techniques require higher coverage. The reason for this lies in the length of the reads produced by the different techniques. Sanger produces reads with a length of ~1000bp, whereas the length of reads produced by NGS techniques varies between a few tens up to ~800bp (11).

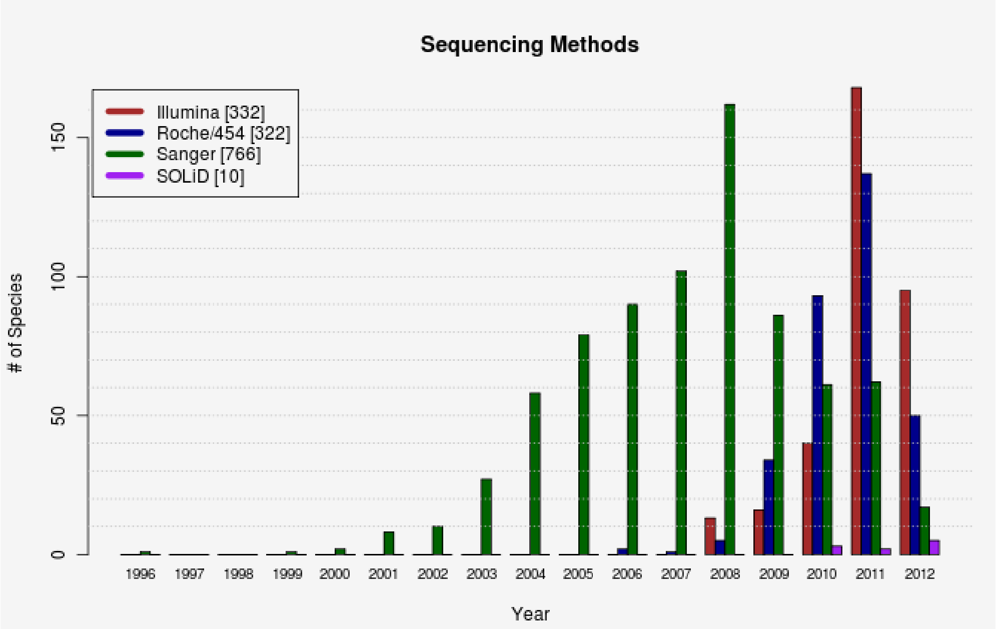


Figure 1.1-3: Sequencing methods used during 1996 and 2012 to sequence eukaryotic genomes.

1.1.2 Gene

As stated before, a genome contains among other elements the blueprint of proteins, called genes. In eukaryotes, a gene usually consists of exons and introns. The exons of a gene are spliced together and subsequently translated to the corresponding protein, while on the other hand the introns of the gene are spliced out by the spliceosome. In prokaryotes, no spliceosome is available.

The spliceosome is a protein complex composed of five ribonucleoproteins (snRNPs). Together with more than 100 other proteins (12), this complex recognizes the exon-intron borders and splice-sites of pre-mRNA and splices out the introns.

Two additional ways of splicing can be found in both eukaryotes and prokaryotes. They are called the self-splicing and the tRNA splicing.

To obtain the exon and intron structure, the so called gene structure, of eukaryotic genes, WebScipio (13) can be used. WebScipio requires the protein sequence as input. In addition, the corresponding genome has to be available in diArk (chapter 2.3, page 90) and to be selected. Optionally, different parameter can be set. Based on these settings, the gene structure of the protein of interest will be computed (e.g. Figure 2.1-3, page 42). During constitutive splicing, every exon of the gene is present in the corresponding mRNA (14). But only 5% of all genes in humans are spliced this way. 95% of the genes with more than one exon are alternatively spliced (15).

Exon skipping

Exon skipping is one way of alternative splicing. During the splicing process, one exon can be spliced out together with the surrounding introns. This leads to two different isoforms of the same gene: One with the normal set of exons and the other without the skipped exon. In higher eukaryotes, this kind of alternative splicing is observed in nearly 40% of all genes (16,17). In lower eukaryotes, exon skipping occurs very rarely (18). Exons that are skipped are more often surrounded by long introns than exons that are not skipped (19).

Alternative 3’- and 5’-splicing

Two further splicing mechanisms are alternative splicing of the 3’- and 5’-splice sites, respectively. The alternatively spliced exons have a constitutive splice site one the one side and on the other side at least two alternative splice sites. For both, the alternative 3’- and the 5’-splice site, two isoforms for each differently spliced exon are possible. The isoforms include, or miss the sequence between the alternative splice sites. In higher eukaryotes, alternative 3’- and 5’-splicing happens in ~18% and ~8% of all cases of alternative splicing, respectively (20).

Intron retention

The fourth possibility of alternative splicing is intron retention. This type of alternative splicing describes the remaining of the intron between two exons in some of the mRNA transcript. In higher eukaryotes, this type of alternative splicing is not quite often. Less than 5% of the genes are affected (16,18,21). However, in plants, protozoa, and fungi, intron retention is the most observed type of alternatively splicing (18).

Alternative promoters and alternative polyadenylation

Furthermore, genes with alternative promoters, or alternative polyadenylation were found. The results are for each of them this leads to two different isoforms of the gene. Thereby, only the first or the last exon is different. These two types of alternatively splicing are less often (18,22,23).

Mutually exclusive splicing

One case of alternative splicing is the mutually exclusive splicing. In a gene, only exactly one exon out of a cluster with two or more exons next to each other is included into the mRNA and further on translated into a protein. No isoform is available, where none or more than one exon of the corresponding cluster is included. With this kind of alternative splicing, the organism is able to produce different isoforms of the same protein and to fine-tune the function. It is not necessary to have two slightly different copies of the same gene in the genome. One example is the muscle myosin heavy chain gene of *Drosophila melanogaster* (DmMhc1) (24). In this gene, 28 exons were found. 16 exons are mutually exclusive. For instance, for exon 10, five mutually exclusive spliced exon variants are found (Figure 1.1-4).

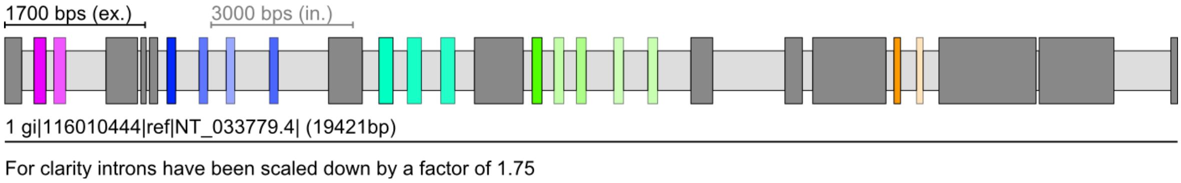

1700 bps (ex.)
3000 bps (in.)
1 gi|116010444|ref|NT_033779.4| (19421bp)
For clarity introns have been scaled down by a factor of 1.75

Figure 1.1-4: ***Mutually exclusive spliced exons of DmMhc1.*** *Dark grey bars represent exons, light grey bars represent introns. Coloured bars represent mutually exclusive spliced exons. For clarity introns have been scaled down by a factor of 1.7.*

Five criteria describe the mutually exclusive exons: A) the exons have to be in a cluster, meaning that they are located next to each other in the gene; B) the splice sites, e.g. GT--AG have to be similar. Otherwise, the spliceosome would not recognize the exons correctly; C) the exons have to be translated in the same reading frame; since the exons are found at the same position of the protein structure, and used to fine-tune the function, they have to have D) the same length and E) a high sequence similarity.

To find these exon clusters in a gene, WebScipio (13) can be used. Therefore, the “Search for Mutually Exclusive Exons” has to be enabled (chapter 2.5, page 139).

1.2 Cytoskeletal Proteins

1.2.1 Cytoskeleton

Eukaryotic cells contain many different types of content, such as a nucleus, membranes, organelles, and lipids, just to name a few of them. Many of them have in common that they do not diffuse in the cell. They have fixed positions. This task fulfils the cytoskeleton. Furthermore, it is essential for the cell structure and stability, thus the cytoskeleton interacts with the membranes of the cell (25). Even some mechanisms developed for movement, like cilia, are based on the functionality of the cytoskeleton. This is the reason why it is also called the skeleton, or the scaffold of the cell.

The cytoskeleton is based on different types of proteins. Furthermore, many proteins are associated with the cytoskeleton and its function. Three different types of filaments can be found in the cell: microfilaments, intermediate filaments, and microtubules.

1.2.2 Microfilaments

The microfilaments are the smallest and shortest types of filaments having a diameter of about 6nm (26). Microfilaments are composed of actin subunits that polymerize and form a double helical structure. Since these actin filaments are highly versatile, they play an important role during cytokinesis, cell movement, stability and structure, as well as the transport of cargo inside the cell. Furthermore, actin filaments are essential for muscle function.

One important protein for several actin-dependent processes, like cytokinesis, or cell motility (27,28), is Coronin (chapter 2.2, page 65). The coronin family includes four classes that are highly conserved and can be found in all branches of the eukaryotic tree of life, except for plants. Class 1 and 2 are specific classes of the metazoan/choanoflagellate branch, whereas class 3 can be found in all eukaryotic branches, except for plants. Class 4 coronins were identified in excavates and opisthokonts, but are absent in the other branches. This coronin class regulates the actin cytoskeleton. Coronin class 1 has the ability to bind to F-actin, the polymer form of actin. Furthermore, this class interacts with the Arp2/3 complex that is important for the regulation of the actin cytoskeleton. Coronin can depending on concentration activate and inhibit the Arp2/3 complex (29).

1.2.3 Intermediate filaments

The intermediate filaments are not as small as microfilaments (about 10nm in diameter) and are among others important for anchoring of organelles and establishing the three-dimensional structure of the cell. Most of them can be found in the cytoplasm, except for intermediate filaments which are made up of lamins. These are found in the nucleus where they are essential for its stability.

1.2.4 Microtubules

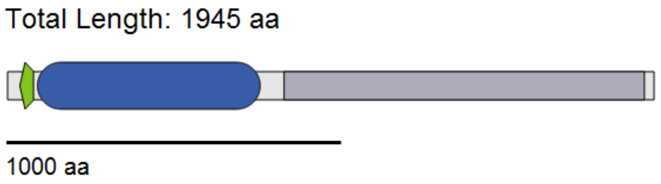
The largest filaments found in eukaryotic cells are microtubules. They are composed of tubulin subunits and are about 25 nm in diameter. Microtubules are important for many different purposes, including stabilizing the structure of the cell, intracellular transport, and the mitotic spindle, just to name a few. Furthermore, microtubules are essential for the function of cilia and flagella.

1.2.5 Motor proteins

In terms of intracellular transport, three motor proteins are well known; myosin, kinesin, and dynein. Motor proteins are important for the active transport of various cargos, cytokinesis, and movement (30). All three of them have in common that they include a specific motor domain that hydrolyses ATP. The released chemical energy is then transformed into directed movement (mechanical energy) using conformational changes. Furthermore, these head domains reversibly interact with microfilament, or microtubule. The tail regions of the proteins are essential for cargo binding (31).

Myosin

Myosin, a large superfamily, uses actin microfilaments for movement. Some of them walk over a short distance through the cell; others are muscle specific and important for its function (32). Not muscle specific myosins transport cargo, like organelles (33), are



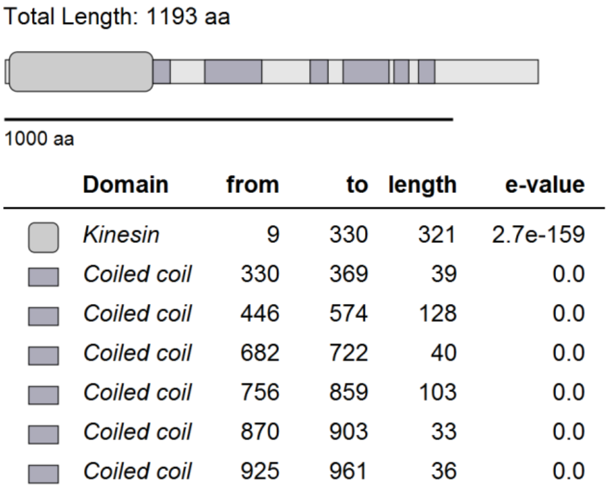
	Domain	from	to	length	e-value
⬟	*Myosin_N*	37	78	41	1.1e-20
⬬	*Myosin_head*	90	760	670	0.0
▭	*Coiled coil*	832	1916	1084	0.0

*Figure 1.2-1****: Domain composition of Homo sapiens Myosin heavy chain 20 (HsMhc20)****. Based on Pfam (34) and provided by CyMoBase (35).*

important for cytokinesis (36), and play an important role for the cell structure (37). As all motor proteins, myosin has a motor head domain that is responsible for the attaching and relating to actin microfilaments. This head domain includes the actin and the ATP binding sites (38,39). Furthermore, myosin has a neck domain and a tail domain (Figure 1.2-1). The tail domain interacts with the cargo.

Kinesin

Kinesin uses microtubules as rails to transport the cargo through the cell. Like all motor proteins, it hydrolyses ATP to produce movement (40). Kinesin is important for transport biomolecules over long distance. Furthermore, it is important for several function of the cell, including meiosis and mitosis (41,42). Kinesins can be grouped into three types: N-kinesins, M-kinesins, and C-kinesins. The naming is based on the location of the kinesin motor domain. N-kinesins, M-kinesins, and C-kinesins have their motor domain in the N-terminal region, in the middle, and in the C-terminal region of the protein, respectively. Most kinesins (N-kinesins, e.g. Figure 1.2-2) move towards the (+) end of microtubules that implies the transport from the cell centre towards the cell membrane (43). Man kinesins have a motor, a neck, a stalk, and a tail domain. The motor and the stalk domains are linked to each other by the neck domain. The long stalk is linked to the tail domain. Like the motor domain of myosin, the motor domain of dynein hydrolyses ATP to perform conformal changes of the head domain to induce movement. The tail domain binds different cargos like lipids (44).

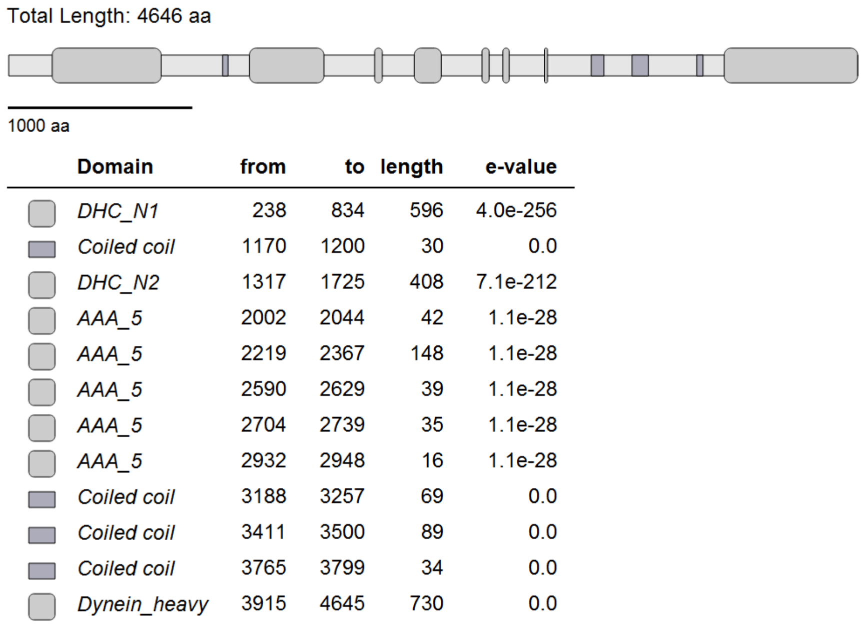


	Domain	from	to	length	e-value
▢	*Kinesin*	9	330	321	2.7e-159
▬	*Coiled coil*	330	369	39	0.0
▬	*Coiled coil*	446	574	128	0.0
▬	*Coiled coil*	682	722	40	0.0
▬	*Coiled coil*	756	859	103	0.0
▬	*Coiled coil*	870	903	33	0.0
▬	*Coiled coil*	925	961	36	0.0

Figure 1.2-2: ***Domain composition of Dictyostelium discoideum Kinesin 3****. Based on Pfam (34) and provided by CyMoBase (35).*

Dynein

Dynein is the biggest motor protein in eukaryotic cells. In human, the dynein heavy chain 1 protein sequence is about 4600 amino acids long (Figure 1.2-3). For comparison, in human the myosin heavy chain 1 gene is composed of about 1900 aas and the kinesin 1 gene is composed of about 1000 aas. Like kinesin, cytoplasmic dynein uses microtubules to transport the cargo over long distances. Dyneins move from the cell membrane to the centre of the cell, towards the (-) of microtubules (30,45). Furthermore, dynein is essential for cell division (46). The second kind of dynein, the axonemal dynein, is important for the movement of flagella and cilia (47). Dynein includes a head domain that is composed of six AAA domains forming a ring. Here, ATP (chemical energy) is hydrolysed to produce conformational changes. These changes produce the power stroke and the resulting movement (mechanical energy) of the protein. Additionally, dynein has a tail, a neck and a linker domain that connect the head to the tail. Furthermore, the stalk domain transmits conformational information between AAA domains and the microtubule binding domain.



	Domain	from	to	length	e-value
▢	*DHC_N1*	238	834	596	4.0e-256
▬	*Coiled coil*	1170	1200	30	0.0
▢	*DHC_N2*	1317	1725	408	7.1e-212
▢	*AAA_5*	2002	2044	42	1.1e-28
▢	*AAA_5*	2219	2367	148	1.1e-28
▢	*AAA_5*	2590	2629	39	1.1e-28
▢	*AAA_5*	2704	2739	35	1.1e-28
▢	*AAA_5*	2932	2948	16	1.1e-28
▬	*Coiled coil*	3188	3257	69	0.0
▬	*Coiled coil*	3411	3500	89	0.0
▬	*Coiled coil*	3765	3799	34	0.0
▢	*Dynein_heavy*	3915	4645	730	0.0

Figure 1.2-3: ***Domain composition of Homo sapiens Dynein heavy chain 1*** *Based on Pfam (34) and provided by CyMoBase (35).*

Cytoplasmic dynein does not have the ability to transport cargo from the periphery to the cell centre. It needs to be activated by dynactin (chapter 2.1, page 30), a multi subunit protein complex composed of eleven different proteins. Dynactin acts as an adapter between dynein and the cargo (48–50), and influences dyneins processivity (51–54). Dynactin is also important during cell division (46) and can also bind to kinesin-2 (55,56) and kinesin-5 (57).

1.3 Protein Family Analyses

1.3.1 Why analysing a protein family?

To understand the role, the structure, and the function of the protein of interest, a protein family analysis and the study of its evolution help. With a protein family analysis in as many organisms as possible and the resulting sequence alignment, it is possible to recognise errors in the starting sequence, or to identify wrong or missing parts of the sequence. This kind of analysis can minimize the probability to get artificial results because of using a wrong protein sequence. Even more, information about conserved amino acids is received. These amino acid positions might have higher influence to the protein structure and function as other position of the protein. To represent the conservation of amino acid positions in a sequence alignment, the tool WebLogo (58) (e.g Figure 2.1-2D, page 39) can be used. This knowledge helps to plan mutation experiments. The scientist saves time, money, and resources. Even more, doing a protein family analysis may also lead to different classes of the same protein that might have different functions, or are expressed in different tissues. One additional aspect is that searching for homologous proteins in e.g. all eukaryotic genomes (59) might lead to an idea, how the Ur protein of the last common ancestor might have looked like.

1.3.2 How analysing a protein family?

To do such a protein family analysis, a starting sequence is necessary. If no sequence is available, one can be get from the protein database of NCBI (National Center for Biotechnology Information). Using this sequence, it is possible to use BLAST (60), or Webscipio (13) to search in the phylogenetic next related organism for the homolog protein. If the genome of an organism is available at NCBI, tblastn instead of blastn can be used. This modification of blast uses a protein sequence as query and searches in all six reading frame translations of the genomic sequence.

It is important to take all hits with a significant small e-value as possible homologs into account. The e-value describes the number of random hits that can be found by chance searching in a database. The bigger the database is the smaller should this value be. They may belong to different classes of the protein family. To align these sequences, BioEdit (61) and ClustalW (62) can be used. But the best way to improve the alignment, mainly if many sequences are already collected and aligned is to do it manually. Especially if additional knowledge about domains or even the protein structure is available, a manually annotation improves the quality of the protein alignment significantly. Automatic alignment tools, such like ClustalW, normally do not use biological information. They try to align the sequences based on scoring matrices. This may destroy biological conserved domains with specific function, like the motor domain, by aligning e.g. loops with different lengths and low homology to each other.

During the analysis, the number of sequences of different species is gaining. If a more related sequence is already available, adding a new sequence gets faster.

1.3.3 Issues during a protein family analysis

Small analysis

Doing a small analysis is faster. It is easier to search in only 10 to 20 species distributed in all branches of the eukaryotic tree. This way is acceptable, if only information for e.g. mutation experiments is needed. With this small number of sequences, it might be possible to determine conserved amino acid position in the sequence of the protein of interest. But to understand the evolution of protein families in detail, a complete analysis of as many as possible e.g. eukaryotes is essential. It might be that all analysed species of a sub branch are missing a specific protein class. The analysis would lead to the result that the last common ancestor (LCA) of this sub branch had lost this specific protein class. But it happens that a further analysed species of this sub branch still has this protein class (e.g. Myosin class 22 in *Spizellomyces punctatus*). The LCA did not lose the specific protein class; all species of the branch have lost this protein except the new analysed one.

Furthermore, it happens that each species of a branch has a different set of classes (63). To select only one of these species as the representative of the branch will lead to a misinterpretation. For instance, if only the kinesins of *Drosophila melanogaster* as the representative of the Drosophila branch were analysed, it would lead to the suggestion that none organism of the branch have Kinesin 4D and that the LCA of all of them had lost this protein class (Kinesin 4D is present in e.g. *Aedes aegypti*). But adding *Drosophila persimilis* as another representative of this branch will neglect the first assumption. Kinesin 4D is present in this organism.

Problems and pitfalls with genome sequencing data

The major issue with genomic data is that normally it is not known whether the genomic sequence is complete. For instance, EST clones are not full-length, or EST data does not reflect all proteins that are really expressed by the organism. Furthermore, whole sequenced genomes sometimes are not completely sequenced (having a low coverage) or are incorrectly assembled. This leads to gaps, frame shifts, or even artificially missing genes. Even more, different sequence databases often contain different sequences of the same organism. One example for this kind of issues is the human reference assembly. One of the Dynein heavy chain genes was partially found in the human reference assembly. A sequence analysis with the same gene in the J. C. Venter chromosome assembly leads to the result that about 40,000 base pairs were missing in the human reference genome.

Furthermore, even published sequences have to be reanalysed. Many of them, especially the sequences based on automatically annotation, are wrongly predicted. One reason can be that for the published protein sequence no sufficient protein family analyses was done, or even skipped. This means that parts or complete exons could have been missed. Additionally, automatic annotation programs usually do not predict other spicing sides than GT--AG. The splice sides GC--AT or AT--AC are ignored. Or, if more than one possible splice site is predicted, these tools may take the wrong one. In addition, small exons are often not recognised. Another issue is that these tools often have problems with missing bases based on sequencing errors and the thereupon supposed frame shifts. They do not add Ns, or gaps at these positions.

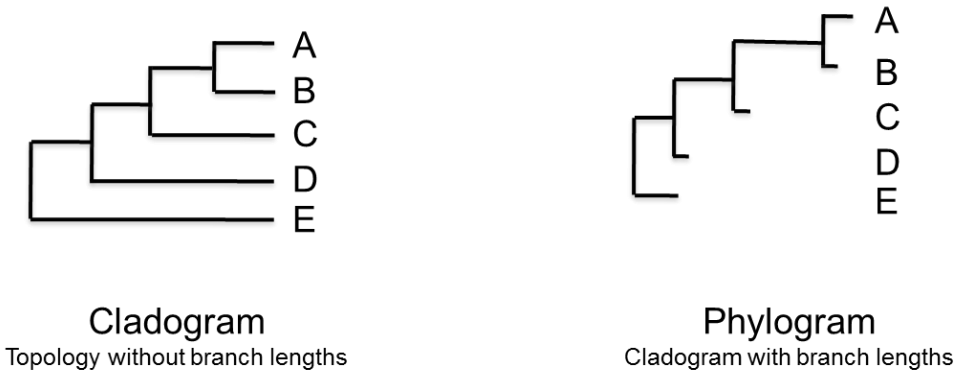
One further reason for doing a complete protein family analysis is that it might be that the collected sequences are not correctly annotated. Often, the starting amino acids of all collected sequences seem to be correct, because all of them start with a methionine. Sometimes, a further protein sequence might not have a methionine at the first suggested amino acid position. This protein sequence might have additional amino acids upstream the suggested protein start position. Reanalysing the start position of all already collected sequences with the upstream sequence of the newly found homolog sometimes leads to another starting methionine upstream that all sequences have in common.

1.3.4 Correctly predict and assemble proteins

It is not easy to know, if the protein of interest and the corresponding sequence is correctly predicted and assembled. One way is to do a protein family analysis in as many species as possible and to manually validate all sequences by comparing them to each other. Even very closely related species might contain different sets of homologs due to gene and genome duplication events, e.g. at the emergence of the vertebrates (64). Furthermore, it is useful to compare different assembly versions of the same organism, if available (chapter 2.3, page 90), especially if the gene seems not to be complete or correct. Additionally, it is helpful to compare the gene structures and intron positions of all sequences (chapter 3.3, page 180). In general, the intron positions are conserved and mistakes in the corresponding protein sequence are directly visible.

1.3.5 Phylogenetic analysis

To fully understand the evolution of a protein and the corresponding sequence alignment, it is important to know which sequences and classes are more related to each other. With this knowledge the manual annotation and correction of a sequence alignment is easier, because related sequences can be put beneath each other. Correlating sequence positions and wrong annotations are more prominent, e.g. if all sequences except one have nearly the same amino acids at the specific position. To achieve this knowledge, a phylogenetic tree is essential.



Cladogram
Topology without branch lengths

Phylogram
Cladogram with branch lengths

Figure 1.3-1: ***Schematic representation of a phylogenetic tree.***

A phylogenetic tree represents the relationship among different organisms. As show in Figure 1.3-1, a phylogenetic tree has many nodes. There are two types of nodes. The first type of nodes is at the end and represents a specific organism (A-E). These nodes are also called leaves. The second kind of nodes is connected to two or more leafs. They represent the last common ancestor (LCA) of the (sub) tree. The nodes are connected to each other via branches. The length of a branch represents the evolutionary time elapsed between its endpoint. A phylogenetic tree without branch lengths is called a cladogram. A phylogram represents a phylogenetic tree with branch lengths.

Organisms can have different types of relationships. If all of them as the LCA share a specific characteristic, it is called monophyly. If some organisms out of this branch do not have this specific characteristic, the grouping of the remaining organisms is called paraphyly. If organisms have the same characteristic, but do not share a LCA with the same characteristic, it is called polyphyly.

During a protein family analysis, homologous sequences of the protein of interest are searched. Homolog means that the sequences or genes share a common origin. The division of homologous sequences into orthologous and paralogous sequences is the most important one for eukaryotes. Orthologous sequences originated from a single ancestral gene in the last common ancestor. Paralogous genes are related via gene duplication. Based on linage-specific gene loss events, to paralogous sequences might behave like orthologs.

This can happen, if the LCA had a duplication of one gene and the analysed offspring has lost either one or the other version of this gene. These genes are called pseudo-orthologs. In bacteria, xenologous and pseudo-paralogous genes can be found (65). These are based on horizontal gene transfer.

To calculate a phylogenetic tree, there are many methods and tools available. To get a first tree, the Neighbor-Joining method, for instance implemented in ClustalW (66), can be used. This distance based method needs a short calculation time and a first classification of different protein classes is possible. But often, this method is not successful in solving the relationship of less related species and sequences. Uncharacterised sequences, so called orphans, remain. They do not group to any protein class, or to each other.

To reduce the number of orphans, the Maximum-Likelihood method can be tried. This method is robust to many violations and statistically well founded, but it is very CPU and time consuming. Even using multiple CPUs for one phylogenetic analyses, e.g. using RAxML (67), the analysis can take days, weeks, or even months to generate a phylogenetic tree. It mostly depends on the number of sequences in the sequence alignment. One way to reduce the computation time is to reduce the number of sequences with CD-HIT (68). Setting the parameter of CD-HIT to select only one sequence representative of a cluster with more than 85% sequence similarity, reduced the number of myosin sequences from more than 6000 to 2200. Another way to trim the alignment is to remove too similar, or too divergent positions from the sequences with the tool Gblocks (69). Comparing the time that is saved by CD-HIT and Gblocks, CD-HIT has significant more influence. The reason is that the computational time is increasing exponentially with the number of sequences and linear with the number of amino acids in the sequences. But still, a phylogenetic analysis of the remaining 2200 sequences of myosin took 140 days with RAxML and 1000 bootstrap steps on server with 48 CPUs.

Both methods, Neighbor-Joining and Maximum-Likelihood have in common that the most adequate substitution matrix should be used. Therefore, the tool ProtTest (70) can be applied. It generates a phylogenetic tree for each selected substitution matrix and lists the Akaike information criterion (AIC) (71) for each result. The AIC describe the quality, how good the selected model fits the underlying data. Based on these values the best fitting substitution matrix can be chosen for the phylogenetic analysis.

One additional method for phylogenetic analysis is the Bayesian method. For such an analysis, the tool MrBayes (72) can be used. But like RAxML, this type of analysis is slow. And for a phylogenetic analysis of a protein family, it does not produce more accurate results. It is not necessary to use ProtTest to select the best fitting substitution matrix. MrBayes can select if on its own.

1.4 Databases and Web applications

1.4.1 Database

During the analysis of a protein family, different type of data is accrued. In our case, information about the species, genomes, sequencing projects, publications, and the sequence itself with different statistical analyses will be collected. To store this data without losing the relation between them, it is necessary to use a relational database, like PostgreSQL (73). This database is open source and free to use.

For each type of data, a database table with specific columns has to be created. For all above named data types, a table is designed to store the corresponding information in. There are two ways to store the relation between tables. If the relationship is *1:n*, a foreign key has to be used. For instance, one sequence belongs to one species, but one species has *n* sequences. This implies storing the corresponding species ID in the sequence table. If the relationship is *n:m*, an additional table has to be designed to store the two foreign keys. For instance, one species is listed in many different publications, and a publication lists different species. This leads storing the species ID and the publication ID in the additional table.

Much information found on our web pages is automatically generated by different tools running in the background. If a sequence is changed or added, the molecular weight, the amino acid statistics, and the domain structure using hmmpfam (74) and Pfam (34) are calculated. If a new species is added, the NCBI taxonomy is added and refreshed once a day. Furthermore, the link to the detailed species page is added to NCBI LinkOut (75) and to the Encyclopedia of Life (76). Additionally, the available Blast (77) page of CyMoBase has to be updated and therefore, the underlying Blast database has to be refreshed as well.

To offer other scientist the possibility to benefit of sequence data and the corresponding statistics, CyMoBase (35) was developed. In CyMoBase, all published data and additional analyses of our group can be found.

1.4.2 Web application

It is not only important to store the incurred information and the corresponding meta data in a database. For instance, it is not user friendly to access the data using a terminal and the SQL language. Furthermore, it could be quite dangerous to allow everybody direct access to the database. It is quite easy to change or even delete stored entries.

One elegant way to share the data with colleagues is to create a web application. This kind of applications has the advantage that the user does not have to install a tool and all its dependencies on the local computer. Only a web browser and internet connection is necessary.

To deploy such a service, the first step is to decide, which programming language fits best to the given task. In our case, the programing language Ruby (78) was selected for nearly every project. This language has the advantage of being object-oriented. Furthermore, source code written in Ruby is easy to read and to understand, even without knowledge of this language. But the main reason for using Ruby in our group is the web framework Ruby on Rails (79).

The Ruby on Rails framework has the advantage of agile and rapid application development, using generators, engines, and gems. Furthermore, the Ruby on Rails community is big and code already exists for many tasks. This framework uses the Model/View/Controller concept that allows the software engineer to structure the different parts of the applications. A Model is used for creating and handling the data. Often, a Model is associated with a database table. The benefit is that e.g. the programmer does not have to code SQL queries by hand, like "SELECT `sequence FROM sequence WHERE sequence_id = 123;`" to get the sequence of interest. He just have to code "`Sequence.find(123).sequence`". Everything is prepared by ActiveRecord implemented in the Ruby on Rails framework; the security checks, validations, creating the SQL-query, and preparing the result as an object. The Controller is important to collect and to prepare all necessary information for the View, which represents the data.

One additional reason for using Ruby on Rails is its ability to handle huge data. In our internal database, we have more than 29,000 manually annotated sequences, 50 proteins, and 1200 species. One reason is the ability to create different caches. That means that e.g. the statistics page of diArk (Chapter 2.3, page 90) does not have to be recalculated every time, a user enters the page. Only, if new data was added to the database behind diArk, the graphs will be recalculated, which takes about 30 seconds. Using the cached version, the user will see the statistics page in less than 1 second.

1.4.3 Development

To keep track of the source code changes made by different members of our group, we use the source code versioning and revision control and management software Subversion (80) and git (81). In principal, this means that the software developer first has to check out the latest version of the source code out of the repository. After adding new features or bug fixes, the developer has to commit the changes with comments back to the repository.

Furthermore, an editor for the source code is essential. Of course, the editor of the operating system can be used. But normally, these editors do not have syntax highlighting, can manage all files of a project, or highlight the changes of the source code compared to the repository. Therefore, a more powerful editor, like Netbeans, TextMate, or Sublime Text 2 should be used.

Normally, only one or two browsers are installed on the local computer. But to be sure that every user of the web application sees the same design and has the same functionality, the application has to be tested with different web browsers and even different versions of them. Therefore, virtual machines with different operating systems and different browsers can be created.

In our group, different web applications are developed. Nearly none of them uses the same Ruby version and set of gems. To avoid conflicts we use the Ruby Version Manager (82) to set up a unique environment for each project.

1.4.4 Deployment

Using the same server for development and for the public access is risky. If there is a change in the source code of the application that disturbs or even breaks the application, it will be directly passed to the user. Furthermore, if there is a security hole in one of the applications and servers running, it could be possible that the main server gets hacked. Therefore, we are using two different servers; one for developing and one for the public. But using this setting, it can be trick to get the latest data and source code to the public server. The source code of the application has to be deployed, the genomes and the corresponding images have to be copied, the users' rights have to be set correctly, and the database and caching servers have to be restarted. Furthermore, the database has to be copied and cleaned up, because not every data in our internal database is already published and therefore should not be public available. To do all these steps by hand take about 30 minutes. Therefore, we use Capistrano (83) for deployment. Different 'receipts' were developed for each of the steps mentioned above. Now, it takes only one command in the terminal to perform every step in the background and to deploy the application.

1.4.5 Set up a new database

Internal, we do not have only one database to store information about cytoskeleton and motor proteins. To set up a complete new database does not only mean to create a new one with the existing database schema. Because all information about species, genome files, and sequencing projects are the same, it would be time consuming to undertake each database the same changes. Therefore, we use the replication system Slony (84) for PostgreSQL. This system uses one master and many slave databases. Each change in a replicated table of the master database will be forwarded to the slaves, immediately.

Furthermore, our web applications are designed to use different databases. Only a few minor changes have to be made in configuration files and the complete web interface and all analyses are available for the new database.

1.4.6 NMR

To solve the structure of proteins, the nuclear magnetic resonance (NMR) technique can be used. Today, there are two major techniques, the liquid-state and the solid-state NMR. Whereas the resonances of a liquid state NMR spectrum are usually separated and an assignment to the corresponding amino acids and atoms is quite easy, the spectra produced with solid state NMR are difficult to interpret. In solid state NMR, the resonances of the atoms fuse and an assignment gets harder. One solution for this issue is to predict the spectrum of the protein of interest and to plot the resulting peaks to the experimental spectrum.

Nowadays, different tools exist to predict the shifts of amino acids atoms (e.g. 81,82). But no software was developed to predict the corresponding spectra based on different experimental settings. Therefore, Peakr and the corresponding web application Webpeakr were developed (chapter 3.1, page 163). Like the other mentioned web applications, Webpeakr only requires a modern web browser.

One feature of NMR is the possibility to study the dynamics of a protein. Based on these studies, the function of the protein can be understood. These exchange processes can be observed. One technique for this goal is the Carr-Purcell-Meiboom-Gill (CPMG) experiment. But the analysis of such an experiment is not easy. To avoid issues during analysis, the web application ShereKhan was developed (chapter 3.2, page 176).

The publications are ordered chronologically, beginning with the newest.

2 Publications

2.1 Evolution of the eukaryotic dynactin complex, the activator of cytoplasmic dynein

Björn Hammesfahr[1] and Martin Kollmar[1§]

[1] Abteilung NMR basierte Strukturbiologie, Max-Planck-Institut für Biophysikalische Chemie, Am Fassberg 11, D-37077 Göttingen, Germany
[§] Corresponding author

BMC Evolutionary Biology Highly accessed
Published: 22 June 2012
BMC Evolutionary Biology 2012 Jun 22;12(1):95 doi:10.1186/1471-2148-12-95 This article is available from http://www.biomedcentral.com/1471-2148/12/95

2.1.1 Abstract

Background

Dynactin is a large multisubunit protein complex that enhances the processivity of cytoplasmic dynein and acts as an adapter between dynein and the cargo. It is composed of eleven different polypeptides of which eight are unique to this complex, namely dynactin1 (p150$^{\text{Glued}}$), dynactin2 (p50 or dynamitin), dynactin3 (p24), dynactin4 (p62), dynactin5 (p25), dynactin6 (p27), and the actin-related proteins Arp1 and Arp10 (Arp11).

Results

To reveal the evolution of dynactin across the eukaryotic tree the presence or absence of all dynactin subunits was determined in most of the available eukaryotic genome assemblies. Altogether, 3061 dynactin sequences from 478 organisms have been annotated. Phylogenetic trees of the various subunit sequences were used to reveal sub-family relationships and to reconstruct gene duplication events. Especially in the metazoan lineage, several of the dynactin subunits were duplicated independently in different branches. The largest subunit repertoire is found in vertebrates. Dynactin diversity in vertebrates is further increased by alternative splicing of several subunits. The most prominent example is the dynactin1 gene, which may code for up to 36 different isoforms due to three different transcription start sites and four exons that are spliced as differentially included exons.

Conclusions

The dynactin complex is a very ancient complex that most likely included all subunits in the last common ancestor of extant eukaryotes. The absence of dynactin in certain species coincides with that of the cytoplasmic dynein heavy chain: Organisms that do not encode cytoplasmic dynein like plants and diplomonads also do not encode the unique dynactin subunits. The conserved core of dynactin consists of dynactin1, dynactin2, dynactin4, dynactin5, Arp1, and the heterodimeric actin capping protein. The evolution of the remaining subunits dynactin3, dynactin6, and Arp10 is characterized by many branch- and species-specific gene loss events.

2.1.2 Background

Dynactin is a multisubunit protein complex in eukaryotic cells required as an activator of cytoplasmic dynein, the major minus end-directed microtubule motor (48,87). Dynactin acts as an adapter between dynein and the cargo (48–50) and enhances the movement of dynein by increasing its processivity (51–54). The dynein-dynactin complex plays an important role during mitosis (88,89) and is necessary for synapse stabilization (90). It is involved in nuclear migration, and during cell division in mitotic spindle positioning (91–93) and organization of spindle microtubule arrays (94). Although most of dynactins functions are in conjunction with cytoplasmic dynein it also binds to and modulates kinesin-2 (55,56) and kinesin-5 (57).

Dynactin is composed of eleven different subunits ranging in size from 22 to 150 kDa (95). Several components are present as dimers or oligomers in the complex resulting in an overall molecular weight of 1.2 MDa. The novel dynactin subunits have initially been named according to the molecular weights of the vertebrate subunits in SDS gels (87). However, as the molecular weights differ between species the original naming is not adequate to describe the protein family relation of the subunits in all eukaryotes. Therefore and because these subunits are unique to the dynactin complex we adopt and use the nomenclature dynactin1 to dynactin6 (symbols DCTN1 to DCTN6), which has recently been established by the HUGO Gene Nomenclature Committee (HGNC; (96)), throughout this analysis.

The structure of the complex can be divided into two distinct domains: the Arp1 rod and the projecting arm (97,98). The projecting arm (consisting of the so-called sidearm and shoulder complex) links dynactin to cytoplasmic dynein, kinesin motors, and microtubules. It is composed of two dynactin1 ($p150^{Glued}$), four dynactin2 (p50 or dynamitin), and two dynactin3 subunits (p24 and p22 have been used for the mouse and the human ortholog, respectively). The Arp1 rod is built of eight Arp1 molecules forming a short actin-like filament, probably one β-actin molecule, and the conventional actin capping proteins Capα and Capβ, which are located at the barbed-end of the mini-filament. The other end of the filament is terminated by Arp10 (the name Arp11 is synonymously used for the vertebrate orthologs (96)) and dynactin4 (p62), to which the dynactin5 (p25) and dynactin6 (p27) subunits are associated. The heterotetrameric complex of dynactin4, dynactin5, dynactin6 and Arp10 is also called pointed-end complex.

Dynactin1 is the largest subunit of the dynactin complex (99) and belongs to the microtubule plus end-binding protein family (100). The microtubule-binding CAP-Gly (cytoskeleton-associated protein-glycine-rich) domain is located at the N-terminus (99,101). The CAP-Gly domain is connected to the other subunits of the complex via two long coiled-coil regions. The first coiled-coil region following the CAP-Gly domain binds to the intermediate chain of cytoplasmic dynein (102,103). Dynactin2 is the connection between the projecting arm and the Arp1 rod (104,105) and its over-expression *in vivo* causes disruption of the dynactin complex (104,97,106). Dynactin3 is required for attachment of dynactin1 to dynactin2 (107). Arp1 is the actin-related protein most similar to actin and forms an actin-like mini-filament (98) that represents the backbone of dynactin, to which the other dynactin subunits bind. It is supposed that membranous cargoes bind to dynactin via the Arp1 rod (50,108,109).

The first studies on dynactin have been performed with chicken brain samples (48,87). Subsequently, dynactin subunits have been identified and analyzed in the model organisms *Neurospora crassa* (110–115), *Saccharomyces cerevisiae* (116–119), *Drosophila melanogaster* (120,121) and *Caenorhabditis elegans* (121–123). Although the composition of the dynactin complex in vertebrates gradually became apparent, a thorough analysis of the complex and its subunits in terms of gene duplicates, alternatively spliced isoforms, and phylogenetic evolution is still missing. That a surprising diversity might be found has been shown by a recent study of the motor protein repertoire of 21 insect genomes uncovering a branch specific duplication of the well-known dynactin1 ($p150^{Glued}$) gene in *Drosophila* species (63).

Building such a multi-protein complex with a filament of fixed size seems rather complicated. Because most of the analyses of the complex have been done with vertebrate samples, it would be interesting to see whether the various unicellular protists that often have smaller gene repertoires, may have evolved compacted versions of the dynactin complex. Vice-versa, there could have been a minimal dynactin complex at the origin of the eukaryotes that multicellular eukaryotes expanded to accomplish more and different tasks. Here, we examined every known protein of the complex and determined its absence and presence in all eukaryotic genomes as available in September 2011. Furthermore, we inspected all genes to identify alternatively spliced exons and their appearance during evolution. For our analysis, we manually assembled and annotated more than 4,700 dynactin and actin-related protein sequences from about 550 species. All sequences were inspected and validated at the genomic DNA level to remove wrongly predicted sequence regions, to manually fill gaps in gene predictions, and to reveal the correct exon/intron boundaries. The sequences and related data like gene structure reconstructions and biochemical properties are available through CyMoBase (http://www.cymobase.org).

2.1.3 Results

Identification of dynactin genes

Dynactin protein sequences are not as strongly conserved as for example tubulins, and three of the dynactin subunits are relatively short complicating their identification if they were spread on several exons. In addition, dynactin contains two actin-related proteins of which Arp1 is closely related to actin while Arp10 is a very divergent member thus hindering their immediate identification. The dynactin subunits might have been duplicated in single species or certain branches, like the *Drosophila* dynactin1 gene (63). These events can only be revealed through the phylogenetic analysis of the corresponding protein sequences. Thus, it is of major importance to obtain the best sequence data possible and to create the most accurate multiple sequence alignments. Automatic gene predictions are error-prone (for example, automatic gene prediction programs do not recognize GC---AG intron splice sites), and even those gene predictions are available for only a small subset of all sequenced eukaryotic genomes (124). Therefore, we manually assembled and annotated all dynactin and actin-related sequences used in this study. Manual identification and assembly means that we started from a set of sequences verified by cDNA and used those for searches with standard tools like TBLASTN in the genome assemblies. Unfortunately, only a few full-length mRNA/cDNA sequences for dynactin subunits are available, which served as representatives for correct sequences. Every search hit has further been analyzed by manual inspection of the corresponding genomic DNA sequence either to reveal the correct intron/exon boundaries or to extend hits that only covered short parts of the search sequence. Those sequences were excluded, for which some local similarity was identified (e.g. similarity to the dynactin1 CAP-Gly domain) but for which the remaining parts of the respective subunits could not be found although the genomic sequences of the respective contigs were long enough. Genomes, for which the respective dynactin subunits could not unambiguously be assembled in the first instance, were reanalysed as soon as further data was added to the multiple sequence alignments. In this way the completeness of the search for dynactin subunits and the accuracy of the gene assembly and annotation has continuously been re-evaluated and improved. In addition to manually assembling all sequences, the multiple sequence alignments of the dynactin sequences have been created and were maintained and improved manually (Additional File 2.1.10.1).

Sequences of which small parts were missing due to gaps in the genome assemblies (up to 5%) were termed “Partials”. “Partials” are not expected to considerably influence the phylogenetic tree computations. Sequences of which more than 5% were missing due to genome assembly gaps or incomplete EST data but that are otherwise unambiguous orthologs or paralogs were termed “Fragments”. "Fragments" are important to denote the presence of the subunits in the respective species in the qualitative analysis. Dynactin genes were termed pseudogenes if they contain more features like frame shifts and in-frame stop codons and miss more conserved sequence regions than can be attributed to sequencing or assembly errors.

In total, the dynactin dataset contains 3061 sequences from 478 organisms (Table 2.1.1, Additional File 2.1.10.2), of which 2872 have been derived from 353 WGS sequencing projects. 2668 sequences are complete, and an additional 191 sequences are partially complete. In addition, 1766 actin and actin-related proteins from 323 species have been assembled to finally reveal the subfamily relationship of potential Arp1 and Arp10 orthologs in questionable cases. For plotting the presence or absence of dynactin subunits across the tree of the eukaryotes we only included those species whose genomes have been sequenced with high coverage and which provided reliable data in many other cases (63,125–127). Nevertheless, low-coverage genomes have also been analyzed because every single piece of sequence could be very important to resolve ambiguous regions in related species or to clarify phylogenetic question. For example, we also analyzed the incomplete genome of the agnath *Petromyzon marinus* to reveal at which stage alternative splice forms had been evolved in vertebrate evolution. To infer the phylogenetic relationship of duplicated dynactin subunits we calculated phylogenetic trees using the Maximum Likelihood and Bayesian methods. Gene structures were reconstructed for all sequences using WebScipio (13) and can be inspected via CyMoBase (www.cymobase.org) for further investigation.

Table 2.1.1: Data statistics

	total	**dynactin1 p150 ropy-3 Nip100**	**dynactin2 p50 Jnm1**	**dynactin3 p24 ropy-10 Ldb18**	**dynactin4 p62 ropy-2**	**dynactin5 p25 ropy-12**	**dynactin6 p27**	**Cap1 (Capα)**	**Cap2 (Capβ)**	**Arp1 ropy–4**	**Arp10 Arp11 ropy–7**
Sequence											
Total	3061	321	312	278	322	326	258	368	299	306	271
From WGS	2872	300	280	248	289	296	229	359	298	304	269
Pseudogenes	60	1	6	3	19	12	5	14	0	0	0
Completeness											
Complete	2668	246	259	250	250	273	220	345	286	293	246
Partials	191	27	14	23	28	31	27	10	4	10	17
Fragments	181	48	37	5	44	12	9	6	9	3	8
Species											
Total	2863	288	306	274	301	313	246	289	292	287	267
WGS-projects	2567	257	261	238	256	271	215	270	278	269	252
EST-projects	960	86	106	96	103	113	93	102	102	87	72
WGS- and EST-projects	1314	124	135	116	134	148	115	146	149	133	114
Sequences in Taxa											
Metazoa	1339	155	132	141	159	158	132	167	94	101	100
Fungi	1339	144	140	118	124	118	96	138	141	165	155
Apusozoa	9	0	1	1	1	1	1	1	1	1	1
Amoeba	56	5	5	4	5	5	5	10	8	4	5
SAR	227	15	28	10	27	32	17	30	29	30	9
Excavata	21	0	1	0	1	5	2	4	7	1	0
Viridiplantae	53	0	0	0	0	0	0	27	26	0	0
Rhodophyta	0	0	0	0	0	0	0	0	0	0	0
Glaucophyta	5	0	1	1	1	2	0	1	1	0	0
Cryptophyta	3	0	1	0	0	1	1	0	0	0	0
Haptophyta	0	0	0	0	0	0	0	0	0	0	0

Dynactin1

Dynactin1 plays a major role for the function of the dynactin complex as it connects the Arp1 rod, and thus the cargo binding sites, to cytoplasmic dynein, the transporter protein complex, and to microtubules, the track. It can hardly be imagined to build a functional dynactin complex without a dynactin1 subunit. However, dynactin1 is also the least conserved of the dynactin subunits (Figure 2.1-1). This is most likely due to its domain structure that consists of a short N-terminal globular CAP-Gly domain followed by two coiled-coil regions, which account for two thirds of its primary sequence. Both the region separating the two coiled-coil regions and the C-terminal region are not even conserved between metazoan and fungal dynactin1 subunits, which belong to the opisthokont branch. Given the functional importance of dynactin1 we were surprised not to be able to identify homologs in any Apicomplexa, in the Heterolobosea *Naegleria gruberi*, and the Apusozoa *Thecamonas trahens* (Table 2.1.1). When searching for dynactin1 homologs in these organisms we analysed all TBLASTN and PSI-BLAST hits showing sequence similarity to CAP-Gly domains but we only found other CAP-Gly domain containing proteins like CLIP-170/restin (128), and the tubulin-specific chaperones B and E (129,130).

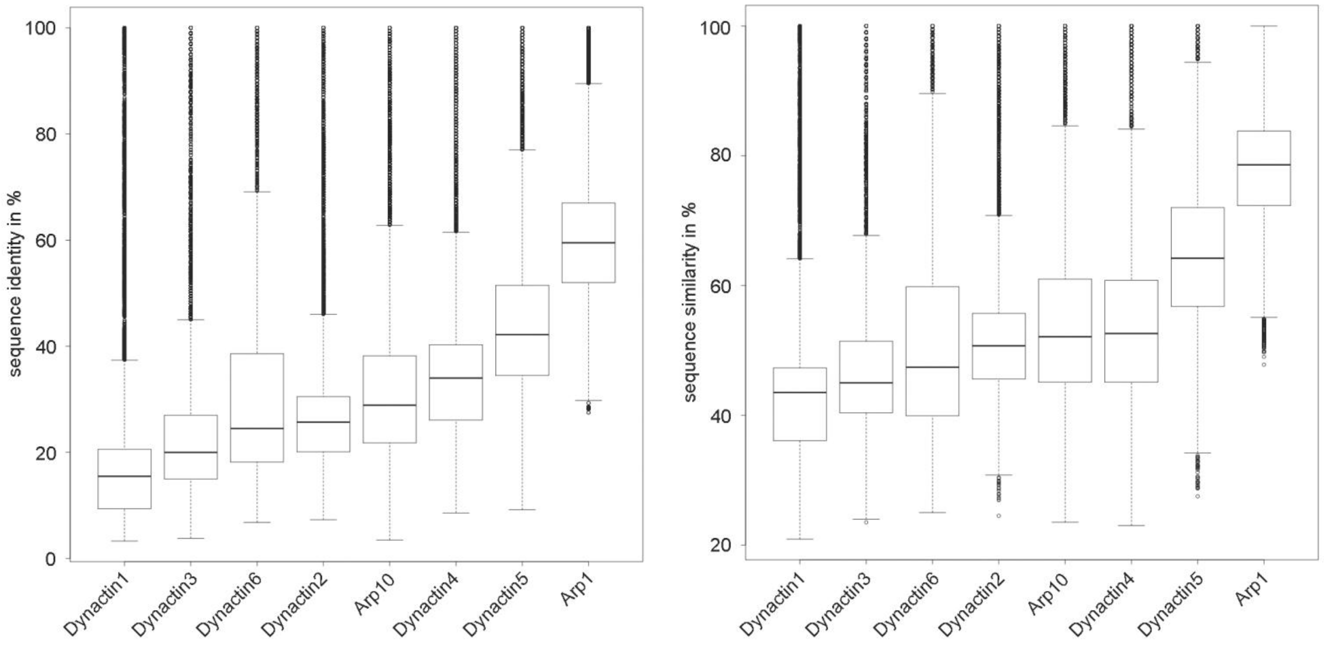


Figure 2.1-1: ***Sequence conservation in dynactin subunits****. Box plots of the sequence identities and similarities of the dynactin subunits.*

Duplicates of dynactin1 have been found in independent branches of the eukaryotic tree (Additional File 2.1.10.3). In the Brachycera branch (including the *Drosophila* clade) the dynactin1 gene has been duplicated once (63). Another duplication of dynactin1 was found in the Actinopterygii branch, supported by *Brachydanio rerio*, *Takifugu rubripes*, and *Gasterosteus aculeatus*. Some of the nematods like *Brugia malayi* also encode two versions of dynactin1. Two duplications of dynactin1 were found in the genome of the fungus *Rhizopus arrhizus*, and one additional dynactin1 in *Mucor circinelloides*. The variant A and B subunits each grouped together, suggesting a gene duplication predating the separation of the two species. Variant C of *Rhizopus arrhizus* grouped to variant B indicating another *Rhizopus*-specific duplication.

The dynactin1 gene of *Homo sapiens* is encoded in 32 exons on chromosome 2 (Figure 2.1-2A, (131)). All exons are constitutively expressed and present in all dynactin1 transcripts, except for exon 5 (“RGLKPKK”), the second part of exon 6 (“APTARK”), exon 7 (“TTTRRPK”), and exon 27 (“EEQQR”) that are alternatively spliced (Figure 2.1-2B). Some alternative transcripts have already been described based on the analysis of a fetal human cDNA library (dynactin1-Δ5; dynactin1-Δ5,6: dynactin1-Δ5,6,7; (132)) suggesting that exons 5–7 are each differentially included. In order to reveal a more general view of possible transcripts we extensively searched for corresponding sequences of vertebrate species in the available EST and cDNA databases and found the following combinations for exons 5–7 (Figure 2.1-2C):

- none of the alternative exons is included in the transcript (Δ5,6,7)
- exon 5 included, resulting in four additional positively charged residues (lysines or arginines, Δ6,7)
- exon 7 included, three additional positively charged residues (Δ5,6)
- exon 5 and 7 included, seven additional positively charged residues (Δ6)
- exon 6 and 7 included, five additional positively charged residues (Δ5)
- exon 5, 6 und 7 included, nine additional positively charged residues

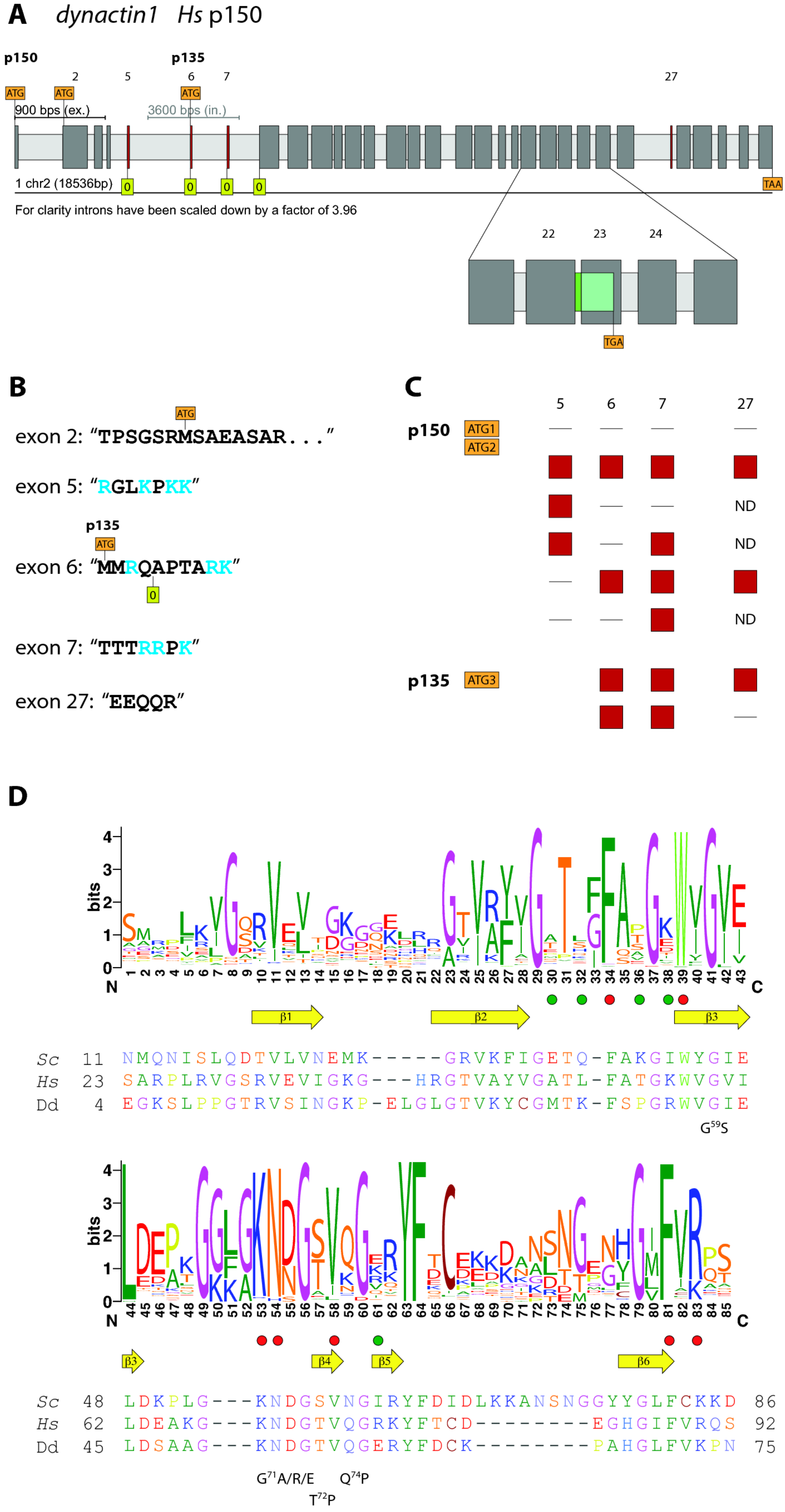

A
dynactin1 Hs p150
p150
p135
2
5
6
7
27
ATG
900 bps (ex.)
3600 bps (in.)
1 chr2 (18536bp)
0
TAA
For clarity introns have been scaled down by a factor of 3.96
22
23
24
TGA
B
exon 2: "TPSGSRMSAEASAR. . ."
exon 5: "RGLKPKK"
p135
exon 6: "MMRQAPTARK"
exon 7: "TTTRRPK"
exon 27: "EEQQR"
C
5
6
7
27
p150
ATG1
ATG2
ND
p135
ATG3
D
bits
4
3
2
1
0
N
C
β1
β2
β3
Sc 11 NMQNISLQDTVLVNEMK-----GRVKFIGETQ-FAKGIWYGIE
Hs 23 SARPLRVGSRVEVIGKG---HRGTVAYYVGATL-FATGKWVGVI
Dd 4 EGKSLPPGTRVSINGKP-ELGLGTVKYCGMTK-FSPGRWVGIE
G59S
β3
β4
β5
β6
Sc 48 LDKPLG---KNDGSVNGIRYFDIDLKKANSNGGYYGLFCKKD 86
Hs 62 LDEAKG---KNDGTVQGRKYFTCD--------EGHGIFVRQS 92
Dd 45 LDSAAG---KNDGTVQGERYFDCK--------PAHGLFVKPN 75
G71A/R/E
Q74P
T72P

Figure 2.1-2: ***Gene structure and isoforms generated by alternative splicing of dynactin1.*** *The gene structure was reconstructed with WebScipio and represents the dynactin1 (p150) homolog of Homo sapiens encoded by 32 exons including four alternatively spliced exons (A). Dark grey bars represent exons, light grey bars indicate introns, and coloured bars symbolize the alternatively spliced exons. For better visualisation, exons and introns are scaled differently. ATG in orange rectangles represent translation start positions. Translation start codons exist in exons 1, 2, and 6, respectively. The zero in green rectangles represents the first reading frame. A zoomed view on the exons 21–25 shows intron retention of intron 22 (dark-green bar) that results in the translation of exon 23 in a different reading frame leading to a premature stop codon (light-green bar). The protein sequences for the alternative exons are given (B) as well as a short summary of the combinations of the alternatively spliced exons that have been found in full-length cDNA data (C). Due to missing full-length cDNA sequences the inclusion or exclusion of alternative exon 27 could not be determined for all combinations of exons 5 to 7 (ND = not determined). The sequence logos (D) illustrate the sequence conservation within the multiple sequence alignment of the CAP-Gly domain. For better orientation, the sequences of three representative CAP-Gly domains are shown: the human CAP-Gly domain as the main target of disease associated mutations, the Saccharomyces cerevisiae and the Dictyostelium CAP-Gly domains as representatives of widely used model organisms. β-strands as determined from the crystal structure are drawn as yellow arrows. Green dots point to amino acids of the human CAP-Gly domain that have been proposed to constitute the second EB1-binding site (133) and red dots highlight residues that are part of the conserved EEY/F motif binding site (101,133,134). Some mutations as found in human diseases are given below the reference sequences with numbering referring to human dynactin*

We did not find EST or cDNA-data for transcripts including only exon 6 (Δ5,7), or EST-data including exons 5 and 6 without exon 7 (Δ7). Exon 27 is also a differentially included exon. Maybe because of lack of more full-length cDNA data or maybe because of tight regulation, exon 27 is found to be absent in dynactin1-Δ5,6,7, and to be present in dynactin1-Δ5 and dynactin1 (Figure 2.1-2C). In addition, transcripts are generated from three alternative start positions. The first is at the beginning of exon 1, the second is at the beginning of exon 2, and the third possible transcript starts with exon 6 (“MMRQAPTARK…”), which corresponds to the “p135” construct. While transcript start sites 1 and 2 are found in all described combinations of exons 5–7 and exon 27, transcript start site 3 (exon 6) is only found in combination with exon 7 included and either exon 27 included or spliced out.

Interestingly, the alternative exons encode different numbers of basic residues, arginines and lysines. Although only six of the eight possible combinations of the alternative exons have been found in EST and cDNA data so far, vertebrates seem to be able to stepwise increase the number of basic residues in this region from zero to nine. The basic residues influence the sliding behaviour of dynactin along the microtubules with fewer charges allowing a faster diffusion (132). The function of the region including the fourth differentially included exon, exon 27, which is located subsequent to the second coiled-coil region and thus behind a proposed Arp1 binding site (99), has not been analysed so far. While the third transcription start site produces a dynactin1 without a CAP-Gly domain (“p135”) the functional difference between transcripts of the two other transcription start sites is not known yet. The longer N-terminus (about 20 residues) is not visible in any of the available crystal structures of dynactin1 CAP-Gly domains (101,133–135). In addition, a solution state structure (PDBid 2COY) revealed that the N-terminus is an unstructured and unordered coil.

There is another alternative transcript generated by retention of intron 22 (Figure 2.1-2A). This intron retention results in a premature stop codon and has only been found in combination with transcription start site 2. The resulting transcript includes the CAP-Gly microtubule binding domain and the dynein intermediate chain binding site but stops before the second proposed coiled-coil region. The C-terminal part of dynactin1 starting with the second coiled-coil region has been proposed to bind to Arp1 and truncation mutants of *Drosophila* dynactin1 have been shown not to be incorporated into dynactin (99,136). This most likely also accounts for the alternative transcripts including intron 22 of vertebrate dynactin1.

The alternatively spliced exons and transcription start sites are conserved in all vertebrates and were also found in the agnath *Petromyzon marinus* the sistergroup of all gnathostomes representing the deepest separation in extant vertebrates. Especially the lysines and arginines and their positions are invariant. However, in the fish type A dynactin1 subunits the exons 5 have been lost, as well as the third potential translation start in exon 6. Instead, exon 6 encodes only the part that is alternatively spliced in type B dynactin1. Thus a “p135”-like isoform cannot be built from fish type A dynactin1 subunits. Alternatively spliced isoforms have not been identified in any other of the analyzed species.

The sequence conservation plot across all dynactin1 CAP-Gly domains shows that the core structure consisting of six beta-strands and several key residues for binding microtubule plus end-tracking proteins is highly conserved (Figure 2.1-2D). The key residues for binding the C-terminal EEY/F tail motifs of CLIP170, EB1 proteins, and α-tubulines are F52, W57, K68, N69, and R90 (human dynactin1 numbering, (101,133)). These are almost invariant from stramenopiles to alveolates to humans (Figure 2.1-2D). In contrast, the residues of the proposed second EB1-binding site A49, L51, T54, K56, and R76 (human dynactin1 numbering, (133)) are not conserved (Figure 2.1-2D). EB1 proteins are present in all eukaryotes (plants, *Giardia*, stramenopiles, Alveolata, *Trichomonas*, Opisthokonts, data not shown). Thus, this proposed second EB1-binding site could be specific to mammals or, most likely, be an artefact from crystal packing effects. The latter is supported by another crystal structure of the complex of the dynactin CAP-Gly domain and the C-terminus of EB1, in which only the C-terminal EEY motif binds to dynactin1 (134). Several mutations in the CAP-Gly domain of human dynactin1 are associated with diseases. The G59S mutation has been identified in patients with distal spinal bulbar muscular atrophy (dSBMA, (137)) and the G71R/E/A, T72P, and Q74P mutations have been found in patients with Perry’s syndrome (138). All mutations lead to destabilization of the CAP-Gly domain (139). The two glycines G59 and G71 are invariant in all dynactin1 CAP-Gly domains. While the threonine and glutamine are variable across the eukaryotes prolines are never found at these positions (Figure 2.1-2D).

Dynactin2

Dynactin2 was found in almost all branches of the eukaryotic tree that contain a dynactin complex (Table 2.1.1). The only two species containing a likely functional dynactin complex without dynactin2 are the closely related yeasts *Ogataea parapolymorpha* and *Ogataea angusta*. Because two different species of *Ogataea* have been sequenced it is unlikely that dynactin2 could be missed because of gaps in the assemblies. None of the genomes analysed encodes more than one functional dynactin2 gene. Some mammals and *Caenorhabditis brenneri* contain dynactin2 pseudogenes.

Dynactin2 from *Homo sapiens* is encoded in 16 exons on chromosome 12 (Figure 2.1-3A). Two of the exons, the very short exons exon 3 (“FAQ”, residues 36–38) and exon 4 (“EL”, residues 39 and 40), are alternatively spliced. Both exons are independently differentially included and many EST and cDNA clones from many vertebrates exist excluding exons 3 and 4 (dynactin2-Δ3, 4) as well as including each exon separately (dynactin2-Δ3 and dynactin2-Δ4) and both exons together. The two alternatively spliced exons were also found in the agnath *Petromyzon marinus*, but not in any invertebrate and thus seem to be an invention of the most ancient vertebrate. While the up- and downstream coding sequence around exons 3 and 4 is slightly variable in vertebrates, the sequence of the two short exons is invariant. In contrast to dynactin1 we could not identify any further transcription start sites. The analysis of the available EST/cDNA data do not support alternatively spliced isoforms in any other species than vertebrates.

The first dynactin2 cDNA sequences were isolated from rat and human, and consisted of the long form including both alternative exons (isoform-1, (104)). Although immunobiochemical studies of the dynactin2 expression in various adult rat tissues have been interpreted to result from the same transcript (104) the slightly different sizes of the dynactin2 bands in the SDS-gels could in retrospect originate from the tissue-specific expression of the alternative splice forms. Later, isoform-1 and the dynactin2 isoform excluding the two alternative exons (isoform-2, dynactin2-Δ3,4) have been shown to be tissue specific transcribed (140), and very recently isoform-2 from human has been compared to chicken dynactin2-Δ3 with respect to determinants for self-oligomerization and interactions with other dynactin subunits (105).

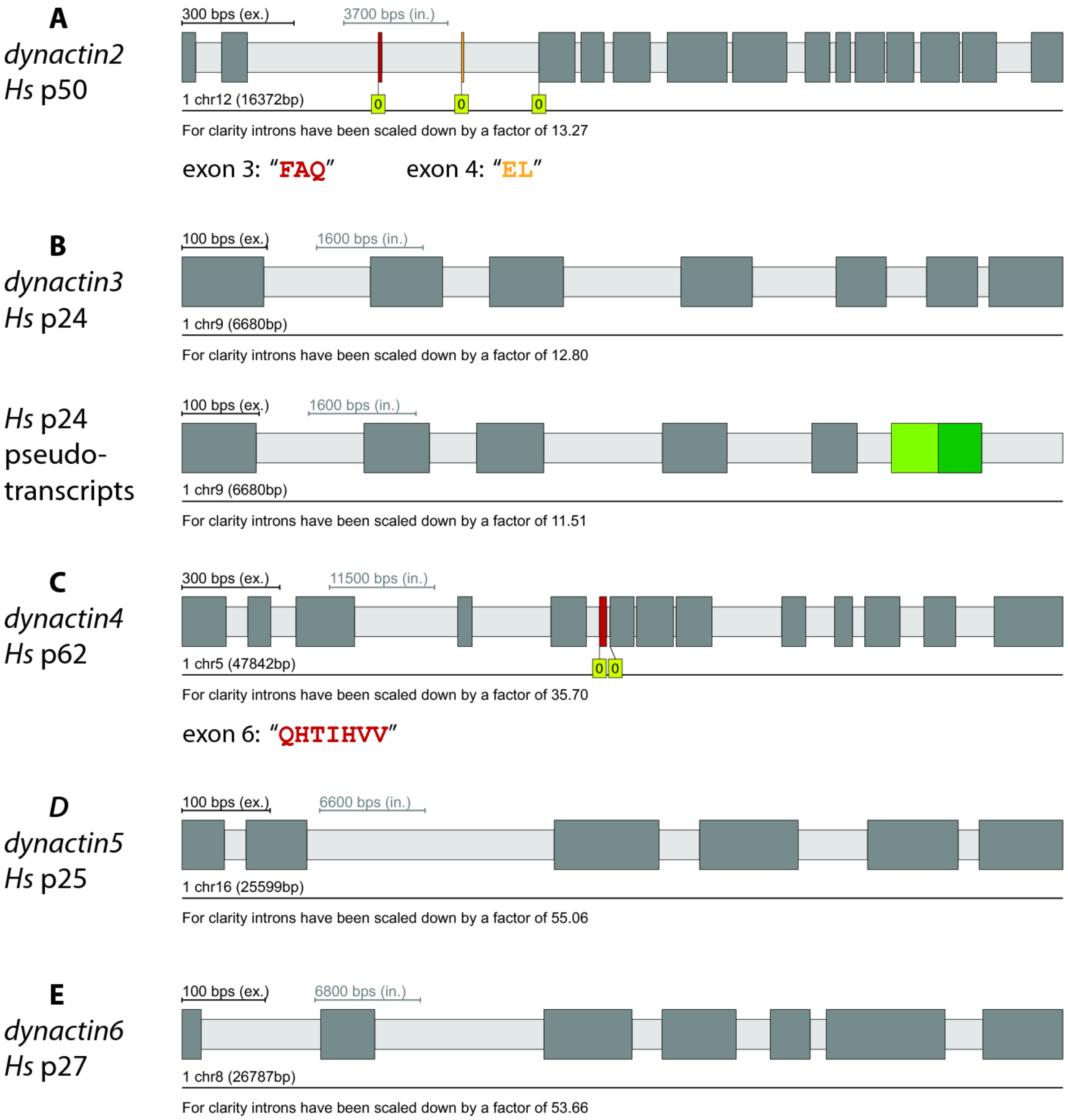


Figure 2.1-3: ***Gene structures and alternatively spliced exons of dynactin subunits.*** *The gene structures including alternatively spliced exons of the dynactin subunits of Homo sapiens were reconstructed with WebScipio. The colour coding is the same as in Figure* ***2 A)*** *The scheme shows the gene structure of dynactin2 (p50) consisting of 16 exons including the differentially included exons 3 and 4.* ***B)*** *Gene structure of dynactin3 (7 exons). For dynactin3, pseudo-transcripts were identified (for detailed information see Additional File 4).* ***C)*** *The dynactin4 (p62) gene is comprised of 14 exons of which exon 6 is alternatively spliced.* ***E)*** *and* ***F)*** *Gene structures of dynactin5 (6 exons) and dynactin6 (7 exons), respectively.*

The residues encoded by the alternative exons (residues 36 to 40) are located in the N-terminal region of dynactin2 but have not been the specific focus of any biochemical study yet. Both the N-terminal and the C-terminal part of dynactin2 are needed for proper self-assembly and binding to dynactin3. The N-terminal 100 residues seem to be required and sufficient for binding to Arp1 (105,141). Binding essays showed that determinants for the optimal recruitment of dynactin1 are located in the N-terminal half of dynactin2 but that the N-terminal 100 residues alone are not sufficient (105). It could thus be possible that a certain combination of alternatively spliced exons in dynactin2 correlates with the differentially inclusion of exon 27 of dynactin1. More specific experiments will be necessary to reveal how such small modifications of two to five residues could modify dynactin2's binding to Arp1, dynactin1, and dynactin3.

Dynactin3

We were not able to identify dynactin3 homologs in Ustilaginomycetes, Chytridiomycota, *Naegleria gruberi*, *Bigelowiella natans*, Ciliophora, plants, and Stramenopiles (Table 2.1.1). Dynactin3 homologs could also not be identified in the Schizosaccharomyces branch and most of the analyzed yeast species. It has been proposed that dynactin3 is the least conserved of the dynactin subunits (121). This analysis has been based on the comparison of the sequence identities of the dynactin subunits of chicken, *Drosophila*, *C.elegans*, and *Neurospora crassa* to the mouse subunits. In order to determine the least conserved dynactin subunit based on all eukaryotes we calculated sequence identity and similarity matrices for all subunits (Figure 2.1-1). Because the data includes sequences from all branches of the eukaryotes each subunit shows a broad distribution. The comparison of the medians of the populations shows that dynactin1 is the least conserved dynactin subunit followed by dynactin3 and dynactin6. Because we were able to identify dynactin3 in almost all opisthokonts the dynactin3 subunits have most likely been lost independently in most Saccharomyctes, the Basidiomycote *Ustilago maydis*, and in the fungi of the Chytridiomycota. Similarly we should have been able to find the dynactin3 homologs in ciliates based on the dynactin3 subunits from the Apicomplexa. The other branches, for which we could not find dynactin3 homologs, have either lost the gene or the dynactin3 proteins must be very different from the known dynactin3 subunits. *Naegleria*, *Bigelowiella*, and stramenopiles species normally do not contain intron-rich genes. Thus, it is unlikely that we missed dynactin3 subunits because they were not present in gene prediction datasets (that are available for some species and that we searched with PSI-BLAST) or because the scores of short exon hits were too low to be detected with TBLASTN.

Dynactin3 has been duplicated in *Rattus norvegicus*. The translations of both genes are identical except for three amino acids that are conserved substitutions. However, the gene of homolog B does not contain any introns and is not supported by EST data. Therefore, it is most likely the result of a recent retro-transcription of a processed pseudogene. Human dynactin3 is encoded on chromosome 9 in 7 exons, which are constitutively spliced (Figure 2.1-3B). A few EST clones suggest the alternative transcription of exon 6 that, however, leads to pseudo-transcripts (Figure 2.1-3B, Additional File 2.1.10.4). Alternatively spliced isoforms have also not been identified in any other species.

Dynactin4

Dynactin4 was found in all branches of the eukaryotic tree that contain dynactin. However, homologs could not be identified in many yeast and most of the Schizosaccharomyces species. Dynactin4 proteins are much longer than dynactin3 proteins and we would expect to identify homologs in the yeast and Schizosaccharomyces species based on the supposed homology to the identified dynactin4 proteins. Missing dynactin4 genes are therefore rather the result of gene loss than the result of identification problems.

The published dynactin4 sequence from *Neurospora crassa* (ropy-2 or RO2 gene) contains a sequencing error that led to a predicted N-terminal extension of 173 residues (113). The genomic sequence encodes another methionine 62 residues upstream of the translation start site. Homologous sequence to these 62 residues including the methionine could only be found in *Neurospora* species and the closely related *Sordaria macrospora* but not in other Sordariales (e.g. *Chaetomium*, *Thielavia*) or any other fungi. The sequence starting from the second methionine is highly conserved in all fungi and thus this methionine is most probably also the translation start site in *Neurospora* and *Sordaria* (Additional file 2.1.10.5).

Dynactin4 from *Homo sapiens* is encoded in 14 exons on chromosome 5 (Figure 2.1-3C). Exon 6 (“QHTIHVV”) is a differentially included alternatively spliced exon. Different isoforms have already been reported for rat (142) but not further evaluated. The alternatively spliced exon is conserved in sequence, length, and reading frame in all vertebrates and was also found in the agnath *Petromyzon marinus*, but not in cephalochordates (*Branchiostoma floridae*), tunicates (e.g. *Ciona intestinalis*), and other invertebrates. The exon invention event therefore predates the separation of the Gnathostomata and the Hyperoartia. There is not enough EST/cDNA data available to proof the alternative character of the exon in all vertebrates. For example, there is only one EST clone from *Petromyzon* that covers the respective sequence region and includes exon 6 but none without exon 6. But because there are EST/cDNA clones for several fish, mammals, and *Xenopus* with and without exon 6 it is highly probable that exon 6 is alternatively spliced in all vertebrates. Alternatively spliced isoforms have not been identified in any other species.

Dynactin4 subunits have been predicted to contain N-terminal LIM (142) or RING domains (143), which are short domains consisting of two zinc fingers of the treble clef fold group arranged in tandem (144,145). The treble clef fold is characterized by a β-hairpin at the N-terminus and an α-helix at the C-terminus that both contribute two ligands for zinc binding (146). In LIM domains these ligands are almost exclusively cysteins while cysteins could be replaced by histidines in RING domains. In addition, the tandem treble clef fingers are separated by a two-residue spacer in LIM domains, which is invariant in length and seems to be essential for LIM-domain function (147). Dynactin4 subunits from almost all species contain eight CxxC motifs, of which the seventh and eights motif are separated by about 150 residues. The cysteins are never substituted by histidines and a two-residue spacer exists only between the fifth and sixth motif. A multiple sequence alignment based secondary structure prediction using Jpred (148) did not reveal any α-helical propensity close to the CxxC motifs (data not shown). Thus, dynactin4 can bind up to four zinc ions but it is unlikely that these zinc fingers adopt the treble clef fold and form LIM or RING domains. Rather, the CxxC motifs will form so-called zinc ribbons, which are composed of two β-hairpins forming two structurally similar zinc-binding sub-sites. These sites are often separated by even protein domains (146). Thus, as long as structural data is not available it is not possible to predict to which of the other motifs the eighth CxxC motif of dynactin4 might fold to build a zinc finger. The highly probable contribution of the eighth motif to the structure of the zinc-finger domain might also explain why expression of only the N-terminal 130 residues of dynactin4 resulted in aggregates (142).

Dynactin5

Dynactin5 was found in all eukaryotic branches that contain dynactin, except for species of the Schizosaccharomycetes clade and some yeast species of the Saccharomyces clade. In the yeast species *Vanderwaltozyma polyspora*, none of the dynactin subunits were identified, except for dynactin5 and the capping proteins. *Vanderwaltozyma* also does not encode a cytoplasmic dynein homolog. Thus, the *Vp*Dynactin5 might either be an artefact (unlikely) or it gained a species-specific function outside the dynactin complex (needs experimental verification). The absence of dynein and dynactin in *Vanderwaltozyma* is most likely related to the specific phenotypic feature that spores are formed by extra mitotic replications after meiosis independent of bud formation (149). Sole dynactin5 subunits have also been found in Euglenozoa, and dynactin6 additionally in *Trypanosoma cruzi*. Euglenozoa also contain cytoplasmic dynein heavy and intermediate chains. The presence of only dyanctin5 (and also dynactin6) is in accordance with the report of a freely soluble pool of these subunits in cells (95). In addition, a dynactin5 homolog was found for the plant *Vitis vinifera* in the cDNA database. This sequence could not be identified in the genome assembly and grouped to a cluster containing parasitic Nematodes in the phylogenetic tree (data not shown) indicating that it is most likely a contamination of the *Vitis vinifera* cDNA library. Some mammals contain one or more pseudogenes resulting from retro-transcripts.

Dynactin5 from *Homo sapiens* is encoded in six exons on chromosome 16 (Figure 2.1-3D). The available EST and cDNA data do not provide evidence for any alternatively spliced exons in human dynactin5 as well as dynactin5's from any species.

It has been reported that the subunits dynactin4, dynactin5, and Arp11 from mouse, *Drosophila*, and *C.elegans* have conserved alkaline pIs (97). It has been suggested that one or all may interact electrostatically with negatively charged membrane lipids or other acidic cargoes such as lipid droplets or viral nucleocapsids (97). Recently, dynactin5 from *Neurospora crassa* was shown to be required for early endosome interaction (150). However, *Drosophila* Arp11 and dynactin4 and Arp11 from *C.elegans* actually have acidic pIs (5.16, 6.61, and 6.4, respectively). These contradictions were perplexing and we decided to determine whether potential electrostatic interactions between dynactin subunits and membranes are conserved across the eukaryotes. Therefore, we have analyzed the distribution of pI values of the pointed-end complex subunits of all species (Figure 2.1-4). Dynactin4, dynactin6, and Arp10/Arp11 show broad distributions from acidic to alkaline pIs. In contrast, almost all dynactin5 subunits have alkaline pIs suggesting a dynactin5 specific interaction within the complex or to other cellular components. The other pointed-end subunits might have conserved functions that are, however, most likely independent from electrostatic interactions.

Dynactin6

Dynactin6 is encoded in all eukaryotic branches, except for Aconoidasida (including *Plasmodium species*), plants, Rhizaria, Bacillariophyta, Saccharomycotina and Schizosaccharomycetes. In the *Euteleostei* branch (containing part of the fish), the dynactin6 gene has been duplicated. The dynactin6 gene from *Homo sapiens* is located on chromosome 8 (Figure 2.1-3E). It consists of 7 exons all of which are constitutively spliced. Alternatively spliced isoforms have not been identified in any species.

Arp1

The only species encoding dynactin subunits except Arp1 were the yeast *Vanderwaltozyma polyspora DSM 70294*, the stramenopiles *Aureococcus anophageferens*, and the cryptophyte *Guillardia theta*. Duplicates have been identified in mammals and anole lizard (Additional File 2.1.10.6) grouping to two types, variant A (also known as α-centractin, (151,152)) and variant B (also known as β-centractin, (153)). Because the fish Arp1s are most closely related to variant B while the bird and frog Arp1s are most closely related to variant A, the Arp1 duplication event must have been at the origin of the vertebrates, most probably as part of the two whole genome duplications (WGDs) that happened at the emergence of the vertebrates (64). Unfortunately, EST or genomic DNA data is not available for any Arp1 in *Petromyzon marinus*. Therefore, we cannot conclude yet whether the Arp1 gene duplication happened at the basis of the vertebrates or the Gnathostomata. Subsequent to the duplication, the ancestors of the fish, birds, and frogs each lost one additional Arp1 paralog, while the mammals and the anole lizard retained both of them. Arp1A and Arp1B have both been shown to be part of dynactin and were found in a constant ratio of about 15:1 in the cytosolic fraction (153). There was no evidence for a free pool of either isoform and it could not be resolved whether Arp1A and Arp1B appear in distinct or mixed complexes. A recent proteomics study of microtubule associated genes in brain tissue also showed that both paralogs are part of the dynactin pool (154). Formally it could be possible that all combinations of the two Arp1 paralogs are present in dynactin complexes in mammalian cells. However, because the two paralogs are 90% identical (96% similar) and the few differences are distributed over the length of the Arp1 molecule and because most vertebrates retained only one Arp1 homolog it is likely that even mixed dynactin complexes are functionally identical.

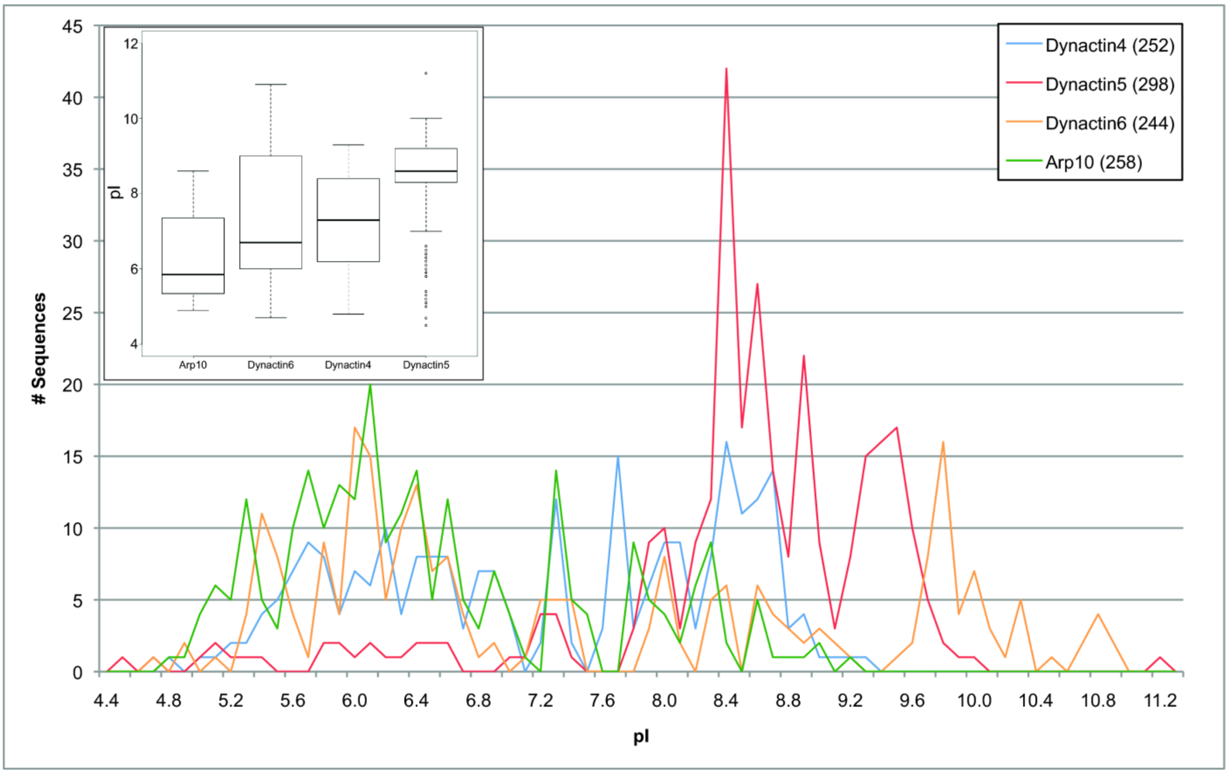


Figure 2.1-4: ***Distribution of the pI values of the dynactin4, dynactin5, dynactin6, and Arp10 subunits.*** *The isoelectric points of the dynactin subunits were rounded to the first decimal place and the number of sequences having a pI of a certain range (increment of 0.1) plotted. The numbers next to the subunits in the legend denote the total numbers of sequences used in the graph. The inlet contains box plots of the data for each dynactin subunit.*

Arp10 (Arp11)

Ten actin-related proteins have been found in the completed genome sequence of *Saccharomyces cerevisiae*, namely Arp1 to Arp10 (155). Subsequently, next to Arp1 a second actin-related protein has been identified in the vertebrate dynactin complex. It has been named Arp11 although its closest grouping homologs in a phylogenetic tree of actin and actin-related proteins were the yeast Arp10 and ropy-7 from *Neurospora crassa* (97). Most probably, the support for a potential subfamily grouping was not as significant as for other groups of actin-related proteins. Along the same lines a comparative analysis of 20 completely sequenced eukaryotic genomes did not reveal compelling evidence for grouping Arp10 and Arp11 into one subfamily but recognized that, until then, the appearance of Arp10 and Arp11 was mutually exclusive (156). It has been suggested that both should be grouped together if yeast Arp10 was found in the dynactin complex or to separate them if both Arp10 and Arp11 were found in a single organism (156). Recently, yeast Arp10 has been shown to be an integral part of the dynactin complex (117).

Arp10 and Arp11 proteins are very divergent, not only in comparison to the other actin-related proteins but also in between the subfamily. In order to determine their presence or absence in species not encoding unambiguous orthologs we assembled all actin related genes of these species for comparison with complete Arp repertoires of representative organisms. Altogether more than 2,300 Arp proteins have been assembled and analyzed including all previously designated Arp classes (156). Thus, Arp11 orthologs have been identified in the Metazoa, the Fungi (except yeasts), the Amoebozoa, and Oomycetes branch. Arp10 orthologs have been identified in almost all species of the Saccharomycotina branch. Exceptions are *Zygosaccharomyces rouxii*, *Vanderwaltozyma polyspora*, *Candida glabrata*, and *Lodderomyces elongisporus*. Both Arp10 and Arp11 have been found in a mutually exclusive manner and group together in the phylogenetic tree of all Arp proteins (Additional file 2.1.10.1). Therefore, and because representatives of both have been shown to be present in dynactin, both are orthologs. According to HUGO this group should be named Arp10 (symbol ACTR10) and Arp11 can be used as synonym (96). As with the other dynactin genes we will follow the HUGO recommendation and use the name Arp10 for orthologs of this group of actin-related proteins throughout the rest of the analysis. Arp10 has been duplicated in *Gallus gallus*, and two Arp10 homologs were identified in the pseudotetraploid *Xenopus laevis*. In the other branches of the eukaryotes, none of the assembled actin-related proteins clearly belongs to the Arp10 subfamily.

Capping proteins Capα (Cap1) and Capβ (Cap2)

The ubiquitous actin capping proteins Capα (Cap1) and Capβ (Cap2) are part of the dynactin complex but except for capping the Arp1 minifilament they do not seem to have dynactin-specific functions and will therefore be discussed in Additional File 2.1.10.7.

2.1.4 Discussion

Here, we have performed an exhaustive analysis of all known dynactin subunits in all eukaryotic genomes available until September 2011. The presence of dynactin subunits is always coupled to the presence of a cytoplasmic dynein heavy chain (DHC1). Some branches do not contain a DHC1 and accordingly do not contain any dynactin subunit (Figure 2.1-5A): plants, diplomonads (e.g. *Giardia lamblia*), Haptophyceae (e.g. *Emiliania huxleyi*), Entamoebidae, some of the Microsporidia, and Rhodophyta (e.g. *Cyanidioschyzon merolae* and *Galdieria sulphuraria*). While the presence of dynactin is coupled to the presence of a DHC1 there are a few species that contain cytoplasmic dyneins but do not encode dynactin subunits: Piroplasmida (e.g. *Babesia* and *Theileria* species), some Microsporidia, and Parabasalia (e.g. *Trichomonas vaginalis*). The DHC1s of the known Piroplasmida and Microsporidia are, however, extremely divergent and shortened (about 3,200 instead of the usual 4,500 residues) and it is not known whether these are functional motors at all. Together, these results demonstrate the strong functional interconnection between dynactin and cytoplasmic dynein. In addition, both were most probably already present in the last common ancestor of the eukaryotes. Although dynein-independent functions have been reported for dynactin these are most likely sub-functionalization in specific branches of the eukaryotic tree in which either dyneins partnership became obsolete for certain functions or in which dynactin acquired additional specific binding partners.

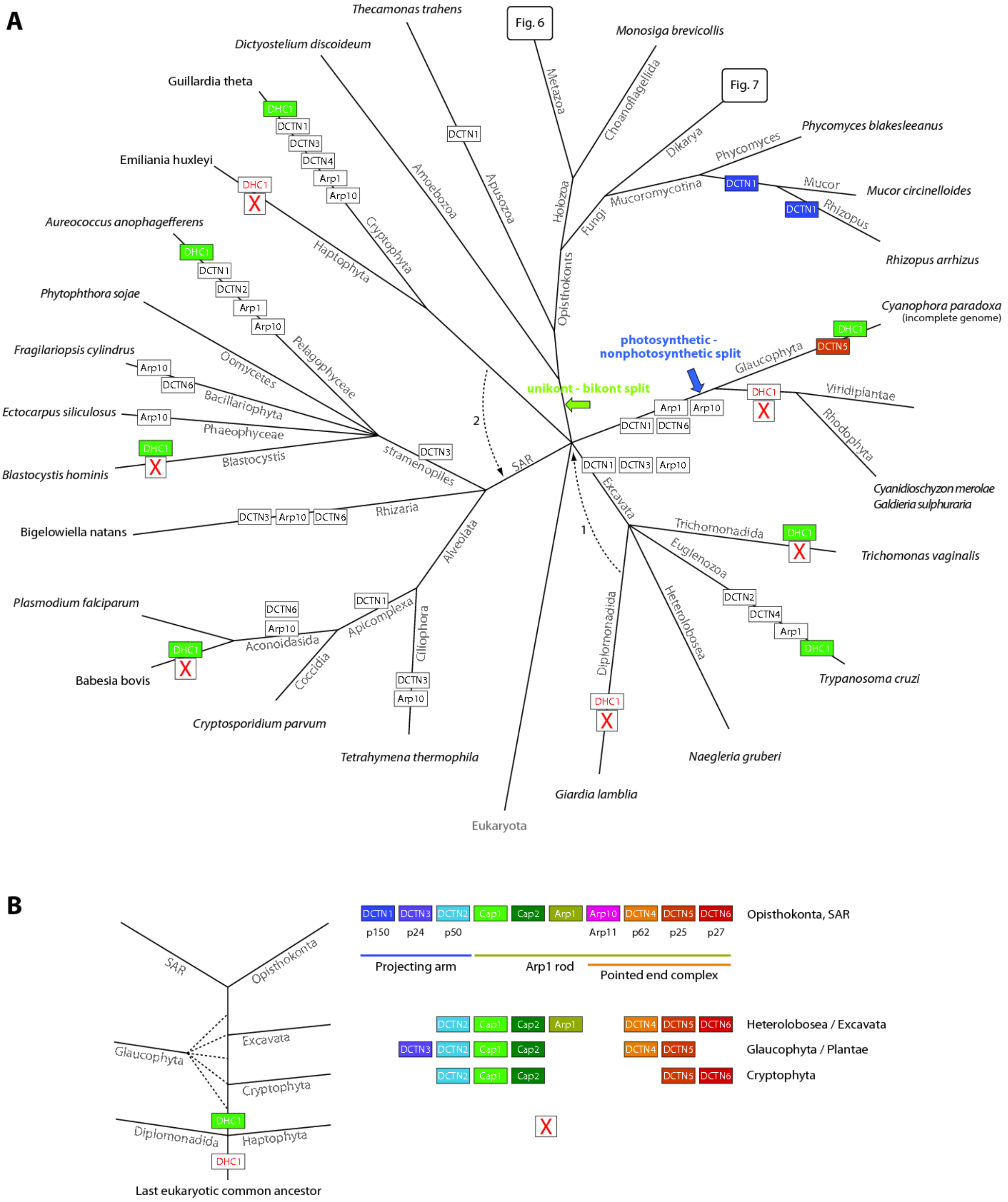


Figure 2.1-5: ***Evolution of the dynactin complex with respect to the species evolution. A)****. The tree represents the most widely accepted phylogenetic tree of the eukaryotes. However, especially the grouping of taxa that emerged close to the origin of the eukaryotes remains highly debated. Therefore, alternative branchings are also indicated in the tree. The phylogeny of the supposed supergroup Excavata is the least understood because only a few species of this branch have been completely sequenced so far. While the grouping of the Heterolobosea, Trichomonada, and Euglenozoa into the Excavata is found in most analyses, the grouping of the Diplomonadida as separate phylum or as part of the Excavata is still debated (arrow 1 (157)). According to most of the recent phylogenetic analyses, the Alveolata, Rhizaria, and Stramenopiles form the superfamily SAR (158,159). The placement of the Haptophyceae and Cryptophyta to the SAR is still highly debated. Although several analyses are in favour of this grouping (arrow 2; (160–162)) most analyses are in contrast (158,159,163–165). At each leaf of the tree one representative species of the branch is printed. Branch lengths are arbitrary. The tree illustrates the presence and absence of each subunit of the dynactin complex in the corresponding species under the hypothesis of five eukaryotic supergroups and the position of the LECA as indicated. Alternative eukaryotic roots are indicated by coloured arrows. Coloured boxes show gene duplications and white boxes show gene loss events of dynactin subunits. The presence (green box) or absence (white box) of the cytoplasmic dynein heavy chain gene DHC1 is indicated for those species that do not have dynactin or miss most of the subunits.* ***B)*** *A possible tree of some major branches of the eukaryotes is shown together with the subunits encoded by the respective taxa. The tree is based on the most parsimonious way the branches could have diverged based on the assumption that during this evolution subunits have only been gained and not lost.*

All of dynactins known eleven subunits were already present in the last common ancestor of the eukaryotes because all of them have been identified in at least two of the major lineages (Figure 2.1-5A). However, in many genomes single subunits are missing. Is this due to gene loss events or due to problems in their identification? The dynactin complex and the dynactin subunits have first been identified and characterized in vertebrates and insects, and these constitute the reference sequences. It could be possible, that some subunits have not been identified in several branches and species, which diverged very early in eukaryotic evolution, because of their low similarity to the subunits of the metazoan species that prevented their identification. However, unambiguous homologs have been identified and annotated in every major lineage of the eukaryotes demonstrating that the sequence similarity in general dates back to the last common ancestor. Even when we searched with these homologs instead of the reference sequences, missing homologs in closely related species could not be identified. For example, although a dynactin1 has been found in *Tetrahymena* we were not able to identify dynactin1 homologs in *Toxoplasma gondii*, *Plasmodium* and *Cryptosporidium* species. Therefore, we rather assume that subunits have been lost during evolution although we cannot exclude that we might have missed divergent homologs that can only be revealed in experiments, but not sequence based analyses. In addition, subunits might be missing because of gaps in the sequence assemblies.

Evolution of the dynactin complex in eukaryotes

The evolution of the dynactin complex in eukaryotes is characterized by many branch- and species-specific gene loss and gene duplication events (Figure 2.1-5A). The monophyly of the SAR branch is well established now (166) as well as the monophyly of the Opisthokonts (and even unikonts, (167)) and Excavata (167). The last common ancestors of both the SAR and the unikonts contained all eleven dynactin subunits (Figure 2.1-5B). If the unikont-bikont hypothesis, that combines all major kingdoms except the unikonts into a supergroup called bikonts and places the eukaryotic root between these two supergroups (166,168), were true the last common ancestor of all extant eukaryotes (LECA) must have contained the complete dynactin complex (Figure 2.1-5A). Another hypothesis places the origin of the eukaryotes between Plantae and the rest (photosynthetic-nonphotosynthetic split, (169,170)). Unfortunately, the genome sequence of the Glaucophyte *Cyanophora paradoxa* is not complete, but it seems that based on the latter hypothesis the LECA would have contained only dynactin2, dynactin3, dynactin4, dynactin5, and the CAP proteins. The dynactin data does not help in resolving the issue of unambiguously placing the eukaryotic root because its analysis involves eleven subunits and is biased by the very small number of sequenced species in the taxa Cryptophyta, Haptophyta, Glaucophyta and Excavata except Euglenozoa. Thus it could be possible that more complete dynactin inventories will be found in newly sequenced species of these taxa like in the SAR branch in which all dynactin subunits were found in total but not in a single species. Building a parsimonious tree from the presence and absence of the dynactin subunits alone in all species is not possible without breaking established monophyletic groups like the sistergroups Fungi and Holozoa, or the sistergroups Blastocystis and Oomycetes. However, if we try to reconstruct a tree of the eukaryotes based on the major taxa by only breaking the still debated phylogenetic groupings of the Haptophyta and the Diplomonadida but leaving established supergroups intact, the following scenario can be imagined (Figure 2.1-5B). Diplomonadida and Haptophyta both do not contain cytoplasmic dynein and dynactin and were therefore the first to diverge in eukaryotic evolution. The LECA would have not contained dynein and dynactin in this case. Next, the dynactin5 and dynactin6 subcomplex and dynactin2 were invented and the Cryptophyta separated. This would be consistent with the finding of a freely soluble pool of dynactin5 and dynactin6 in cells (95). The placing of the Glaucophyta (as part of the Plantae) is not yet clear due to the incomplete genome of the single representative *Cyanophora paradoxa*. The Glaucophyta do have already dynactin3 and dynactin4 but miss dynactin6. Subsequently, Arp1 and dynactin4 evolved completing the Arp1 rod in Heterolobosea (Excavata). Finally, the projecting arm had been completed in SAR and Opisthokonta. However, this model is based on the assumption that dynactin subunits had only been gained and not lost during early eukaryotic evolution, and the model contradicts the unikont-bikont and the photosynthetic-nonphotosynthetic split hypotheses. Given the many dynactin gene loss events in later separating branches it is more likely that the LECA already contained all dynactin subunits. This assumption could be combined with both split hypotheses and is in agreement with analyses of other protein complexes in which the reconstructed complexes of the LECA contained most of the present-day subunits (171,172).

From the stage of the SAR and Opisthokonta the subsequent evolution of the branches is determined by many and specific gene loss events. Especially Arp10, dynactin6, dynactin1, and dynactin3 have been lost independently in many branches. The Arp1 filament capping function of Arp10 might have been taken over by one of the so far unclassified actin-related proteins or dynactin4. Dynactin6 forms a tight complex with dynactin5 in vertebrates (97) but because also yeasts have, if at all, only dynactin5 it might be possible that dynactin5 forms a homodimer in the species lacking dynactin6. Dynactin1 subunits have independently been lost by many species or their dynactin1 homologs have lost the CAP-Gly domain hindering their identification because of the low sequence similarity of the coiled-coil regions. Vertebrate and *Drosophila* dynactin1 transcripts without a CAP-Gly domain (corresponding to “p135”) are very well versed to bridge the Arp1 rod to dynein and microtubules showing similar intracellular trafficking of organelles (94,132,173). Thus, so far unknown dynactin1s without CAP-Gly domains could still be present in Apicomplexa, Heterolobosea and Apusozoa. Dynactin3 is necessary for the incorporation of dynactin1 into the yeast dynactin complex (116). However, it has been shown that dynactin1 in vertebrates and *Drosophila* contains an independent Arp1 binding site, and therefore dynactin3 might not be essential for the dynactin complex in all species. This might explain dynactin3's absence in many branches that have dynactin1. Other reasons could be that we were not able to identify all dynactin3 subunits because of their low sequence conservation or that dynactin3 has diverged in independent branches so far that homology cannot be detected any more. In any case, strong changes happened to this subunit independently in many early branching eukaryotic lineages and also in closely related branches.

Expansion of dynactin complexity in metazoan

Dynactin complex diversity in metazoa is greatly enhanced by branch specific gene duplications and the introduction of alternative splice forms (Figure 2.1-6). The dynactin1 gene has been duplicated independently in the nematods of the Spirurida branch, in the Brachycera including the *Drosophila* species (63), and in fish genomes (Figure 2.1-6). Thus, dynactin complexes with different properties could be generated in these species by assembling two different homodimers or a heterodimer of their different dynactin1 subunits. Dynactin6 and Arp10 have also been duplicated in the Otocephala branch (including *Brachydanio rerio*) and in birds, respectively. Arp1 has been duplicated early in vertebrate history and subsequently fish, birds, and amphibians lost different types of the duplicates. An alternative but less likely scenario would be that all dynactin subunit duplications were part of the two whole genome duplications at the origin of the vertebrates followed by numerous independent gene losses in the extant species.

Interestingly, alternatively spliced exons have been invented in vertebrate dynactin1, dynactin2, and dynactin4 genes either before, in between or after the two whole genome duplication events happened but before the divergence of the agnaths and the Gnathostomata (Figure 2.1-6). Thus, complexity and fine-tuning of dynactin1 can considerably be enhanced by differential inclusion of four exons that can be combined with three alternative start sites for translation (Figure 2.1-2). The alternative start sites affect the inclusion/exclusion of the CAP-Gly domain, and three of the alternative exons encode consecutive pieces of the basic region between the CAP-Gly domain and the first coiled-coil domain. These alternative splice forms therefore do not affect the binding of dynactin to cytoplasmic dynein but only the region attaching dynactin to microtubules (174). Altogether, 36 different transcripts can theoretically be generated for each vertebrate dynactin1 gene. It is not known yet whether heterodimers of dynactin1 isoforms are possible, which would multiply the theoretically possible number of different dynactin complexes. However, it seems unlikely that a transcript with certain functionality, e.g. a transcript without the CAP-Gly domain, would be combined with a subunit that could in part reconstitute the missing function. This conclusion is consistent with findings in rat brain that showed distinct complexes of dynactin with either full-length dynactin1 (p150) or CAP-Gly diminished dynactin1 (p135, (173)). The three alternative splice forms analyzed so far (p150-Δ5; p150-Δ5,6: p150-Δ5,6,7) also showed a tissue specific expression pattern (132) demonstrating that most likely only a limited number of different dynactin1 subunits and not all possible combinations are present in a single cell. However, all combinations will most likely be present in each organism.

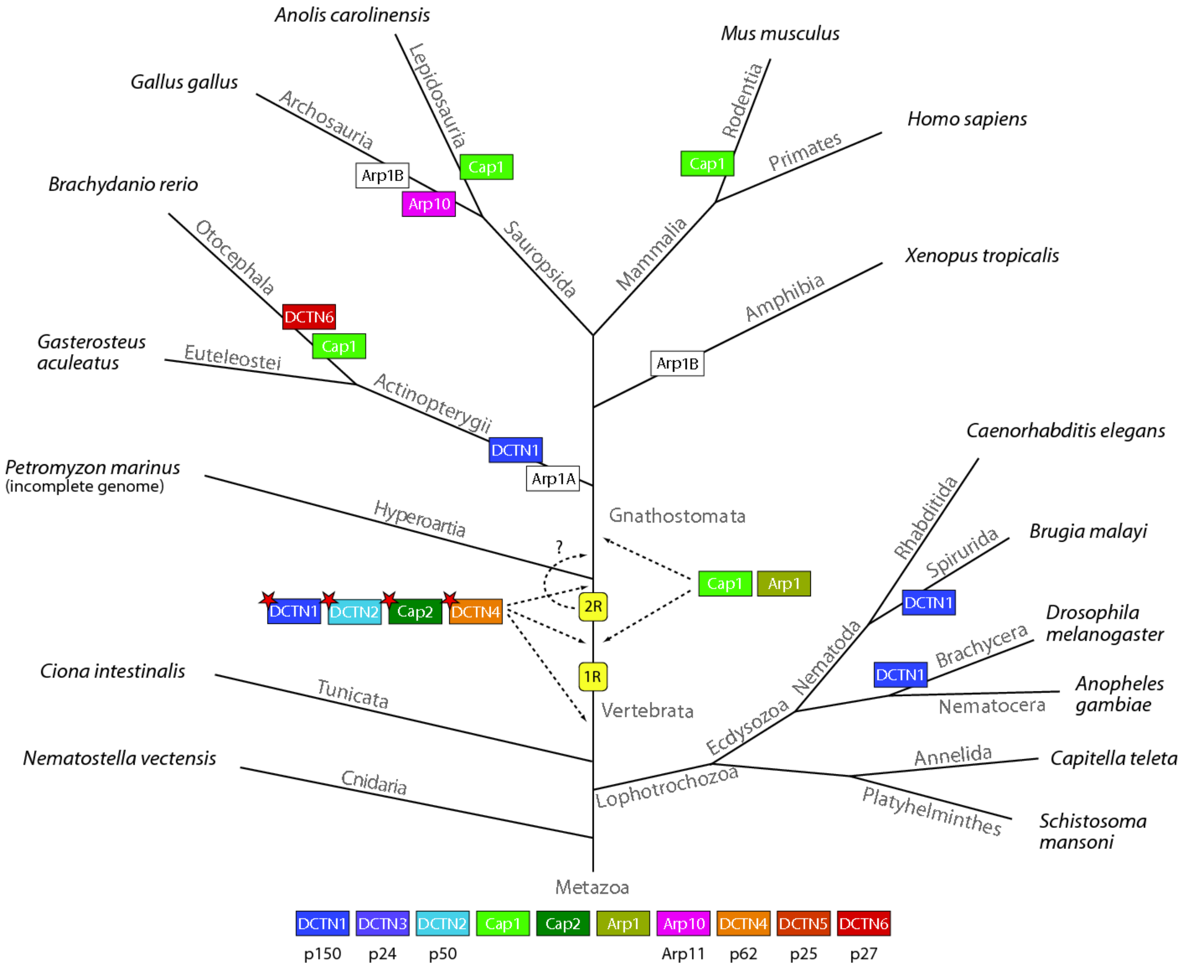


Figure 2.1-6: ***Evolution of the dynactin complex with respect to the species evolution in Metazoa.*** *The tree represents the most widely accepted phylogenetic tree of the Metazoa. At each leaf one representative species of the branch is printed. Branch lengths are arbitrary. Coloured boxes show gene duplications of dynactin subunits. Coloured boxes with a red asterisk illustrate the introduction of alternatively spliced isoforms of the corresponding protein in vertebrates. White boxes denote gene loss events. The two whole genome duplication events at the origin of the vertebrates are shown (1R and 2R). The second duplication might also have happened in the Gnathostomata branch (dashed arrow). Due to missing data the duplication of the Cap1 and Arp1 genes cannot unambiguously be dated and could have happened either after the 1R event or after the divergence of the Hyperoartia.*

Because most subunits are present in multiple copies in the complex, a single gene duplication of one subunit would already result in two, three, or more different complexes, if complexes were built not only from distinct but also mixed subunit compositions. Based on their gene content, the vertebrates can theoretically build thousands of different dynactin complexes considering all combinations of the genes and splice forms. However, most of the differences are introduced by tiny changes. For example, all identified alternatively spliced exons contain only between two and seven residues. In addition, the two Arp1 paralogs differ in only a few residues that are distributed over the length of the molecule. Thus, these small changes are not expected to considerably alter the overall structure of dynactin. However, it is well known that even single posttranslational modifications can dramatically change the functions of proteins from activating/deactivating enzymes or binding/non-binding other proteins or membranes (e.g. phosphorylation of dynactin1 strongly reduces its microtubule affinity, (175)). Concerning the two Arp1 paralogs it is hard to imagine how defined combinations could be generated in the cell given that eight to nine Arp1 subunits comprise the Arp1 minifilament. This would require a strong regulation of the protein level of both paralogs as well as a strong regulation of the position-specific incorporation into the minifilament. The 15:1 ratio of the paralogs could be regulated at the transcription level but it is very likely that both are just randomly incorporated into dynactin without influencing its structure, stability, and function. Therefore, dynactins functions will most likely only be modulated through the various alternative transcripts. The differences seem small but have not been studied at a molecular level yet.

Reduction of dynactin complexity in yeasts

In general, gene loss in yeasts only affects the pointed-end complex subunits and dynactin3, which mediates association of dynactin1 to dynactin2 (Figure 2.1-7) The Schizosaccharomycetes and Saccharomycotina both have lost the dynactin6 subunit. Dynactin5 and dynactin6 are predicted to fold into left-handed β-helical structures (176), and are supposed to form a tight heterodimeric complex in vertebrates (97). They show low sequence similarity but can still be aligned to each other (data not shown). Therefore, it could be possible that dynactin5 forms homodimers in those species that do not encode dynactin6. Dynactin3 and dynactin5 have been lost in Schizosaccharomycetes and many Saccharomycotina subbranches. The loss of dynactin3 in yeasts is surprising because it has been found to be essential to recruit dynactin1 to the dynactin complex in *Saccharomyces cerevisiae* (116). The dynactin1 genes in yeasts have about the same lengths, which is also true for the dynactin2 genes. A missing dynactin3 is therefore not compensated by additional domains in the other dynactin subunits. Either changes at the surface of dynactin1 or dynactin2 may supersede dynactin3 or we were not able to detect the missing dynactin3 subunits yet. Dynactin5 is required for the interaction of dynein with a subset but not all membranous vesicles, which is supported by the conserved basic pI of all dynactin5 subunits and by membrane-flotation essays (150). This is also consistent with findings in *Saccharomyces cerevisiae*, which does not encode a dynactin5 subunit, that dynein is necessary for nuclear migration and spindle orientation but does not perform vesicle transport (177,178). In addition, dynactin4, Arp10, and dynactin2 have been lost in several, two, and 1 branch of the yeasts, respectively. Arp10 is needed for the stability and capping of the Arp1 filament (117) and its absence should thus affect dynactins integrity. Both *Zygosaccharomyces* and *Lodderomyces* lack Arp10 and dynactin4, the other pointed-end capping protein and it is unclear how the Arp1 filament could be stabilized in these species.

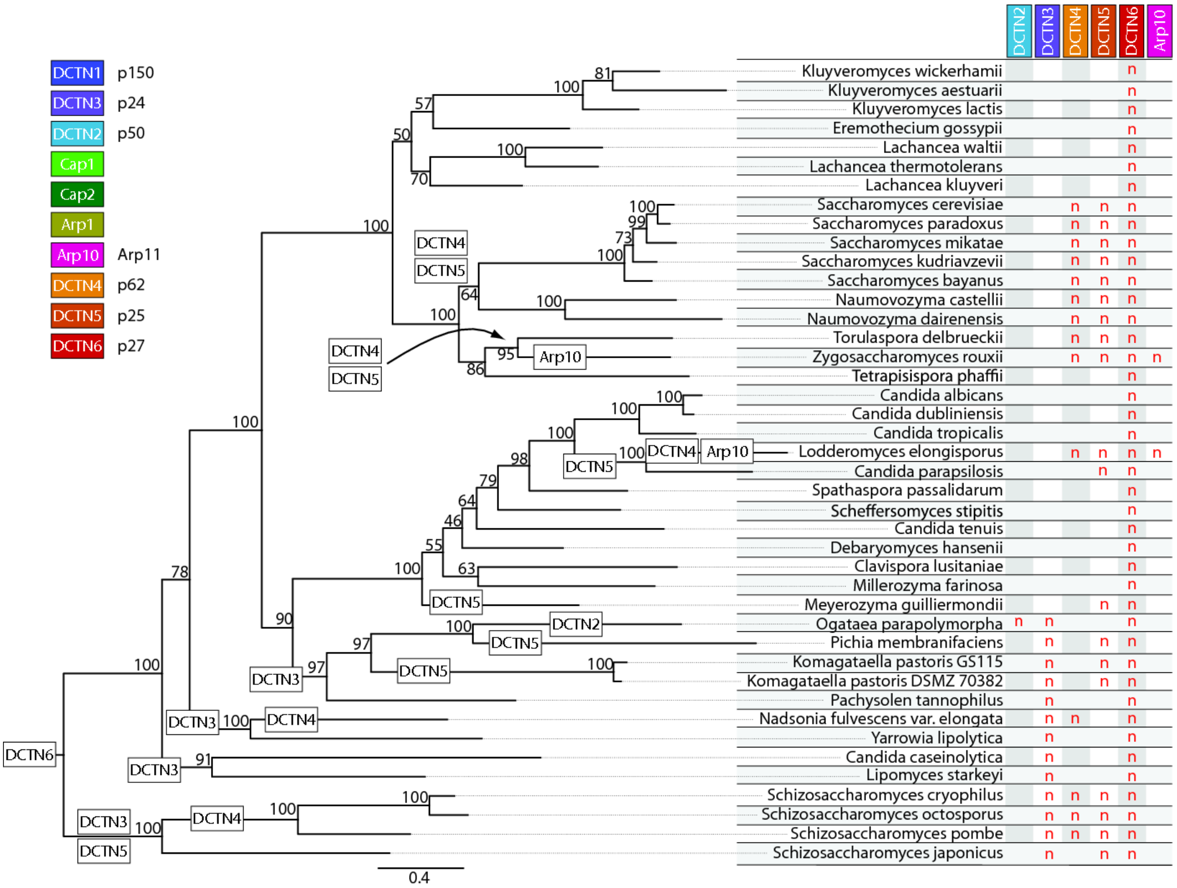

DCTN1 p150
DCTN3 p24
DCTN2 p50
Cap1
Cap2
Arp1
Arp10 Arp11
DCTN4 p62
DCTN5 p25
DCTN6 p27
DCTN2 DCTN3 DCTN4 DCTN5 DCTN6 Arp10
Kluyveromyces wickerhamii n
Kluyveromyces aestuarii n
Kluyveromyces lactis n
Eremothecium gossypii n
Lachancea waltii n
Lachancea thermotolerans n
Lachancea kluyveri n
Saccharomyces cerevisiae n n n
Saccharomyces paradoxus n n n
Saccharomyces mikatae n n n
Saccharomyces kudriavzevii n n n
Saccharomyces bayanus n n n
Naumovozyma castellii n n n
Naumovozyma dairenensis n n n
Torulaspora delbrueckii n n n
Zygosaccharomyces rouxii n n n n
Tetrapisispora phaffii n
Candida albicans n
Candida dubliniensis n
Candida tropicalis n
Lodderomyces elongisporus n n n n
Candida parapsilosis n n
Spathaspora passalidarum n
Scheffersomyces stipitis n
Candida tenuis n
Debaryomyces hansenii n
Clavispora lusitaniae n
Millerozyma farinosa n
Meyerozyma guilliermondii n n
Ogataea parapolymorpha n n n
Pichia membranifaciens n n n
Komagataella pastoris GS115 n n n
Komagataella pastoris DSMZ 70382 n n n
Pachysolen tannophilus n n
Nadsonia fulvescens var. elongata n n n
Yarrowia lipolytica n n
Candida caseinolytica n n
Lipomyces starkeyi n n
Schizosaccharomyces cryophilus n n n n
Schizosaccharomyces octosporus n n n n
Schizosaccharomyces pombe n n n n
Schizosaccharomyces japonicus n n n
81
100
57
50
100
70
100
100
99
73
100
DCTN4
DCTN5
64
100
100
DCTN4
DCTN5
95
Arp10
86
100
100
100
100
100
DCTN4
Arp10
98
DCTN5
79
64
46
55
100
63
DCTN5
78
90
100
DCTN2
97
DCTN5
DCTN3
97
DCTN5
100
100
DCTN3
100
DCTN4
DCTN6
DCTN3
91
100
DCTN3
100
DCTN4
DCTN5
100
0.4

Figure 2.1-7: ***Evolution of the dynactin complex with respect to the yeast species evolution.*** *The phylogenetic tree of the Saccharomycetes and Schizosaccharomycetes is based on the Maximum-Likelihood tree (RAxML) of the concatenated dynactin2, Cap1, Cap2, Arp1, and Arp10 subunits. Bootstrap support values (100 randomisations) are given for every node. The phylogenetic distribution of the sampled species is in overall agreement with other recent yeast phylogenies (179,180). Small differences are most likely due to the different genes (LSU rRNA, SSU rRNA and EF-1α DNA sequences in (179), 542 putative orthologous proteins in (180), and dynactin protein sequences in our analysis) and methods used (NJ and MP in (179), ML in (180) and in our analysis). On the left the phylogenetic tree is shown with the corresponding species at each leaf. White boxes at branches represent gene loss events. On the right those subunits of the dynactin complex are tabulated that show differential inclusion within the analysed species. Dynactin subunits that are present in all species have been omitted for clarity. The abbreviation 'n' denotes the absence of the corresponding subunit in the respective genome while blanks indicate their presence.*

2.1.5 Conclusions

The dynactin complex is a very ancient complex that already existed in the last common ancestor of extant eukaryotes. It consists of eleven subunits of which at least seven comprise the core structure: dynactin1, dynactin2, dynactin4, dynactin5, the heterodimeric capping protein, and Arp1. The presence of the dynactin complex coincides with that of the cytoplasmic dynein heavy chain: Organisms that do not encode cytoplasmic dyneins like plants and diplomonads also do not encode dynactin subunits either. In the metazoan lineage, several of the dynactin subunits were duplicated independently in different branches. The largest repertoire is found in vertebrates. Also at the origin of the vertebrates, several alternatively spliced exons have been invented providing the basis for modulating the core functions. The most prominent example is the dynactin1 gene, from which 36 different transcripts could be generated. In contrast, ascomycetous yeasts have reduced subunit compositions. In general they have reduced pointed-end complexes, which in return let to the loss of the functions coupled to the specific subunits.

2.1.6 Methods

Identification and annotation of the genes of the dynactin subunits

Dynactin genes have been identified in iterated TBLASTN and PSI-BLAST searches of the completed or almost completed genomes of about 600 organisms starting with the protein sequences of the human dynactin subunits. All hits were manually analyzed at the genomic DNA level. The correct coding sequences were identified with the help of multiple sequence alignments of the respective dynactin subunits. As the amount of dynactin sequences increased (especially the number of sequences from taxa with few representatives), many of the initially predicted sequences were reanalysed to correctly identify all exon borders. Where possible, EST data has been analyzed to help in the annotation process. In addition to the analysis of these large-scale sequencing projects, all dynactin sequences in the “nr” database at NCBI have been collected and reanalysed.

Several of the genes contain alternative splice forms. The different splice forms were not considered independently in the analysis but in all cases the same splice forms were taken for homologous dynactin proteins. All sequence related data (names, corresponding species, GenBank ID's, alternative names, corresponding publications, domain predictions, sequences, and gene structure reconstructions) and references to genome sequencing centres are available through the CyMoBase (http://www.cymobase.org, (181)). A list of the species analyzed, their abbreviations as used in the alignments and trees, as well as detailed information and acknowledgments of the respective sequencing centres is also available as Additional File 2.1.10.8. Webscipio (127,182) was used to reconstruct the gene structure (exon/intron pattern) of each sequence.

Generating the multiple sequence alignment

The multiple sequence alignments of the dynactin subunits have been built and extended during the process of annotating and assembling new sequences. The initial alignments have been generated from the first about 20 sequences obtained from NCBI using the ClustalW software with standard settings (62). During the following correction of the sequences (removing wrongly annotated sequences and filling gaps) the alignment has been adjusted manually. Subsequently, every newly predicted sequence has preliminarily been aligned to its supposed closest relative using ClustalW, the aligned sequence added to the multiple sequence alignment of the respective dynactin subunit, and the alignment adjusted manually during the subsequent sequence validation process. Still, many gaps in sequences derived from low-coverage genomes remained. In those cases, the integrity of the exons next to gaps has been maintained (gaps in the genomic sequence are reflected as gaps in the multiple sequence alignment). The sequence alignments of the dynactin subunits can be obtained from CyMoBase or Additional File 2.1.10.1.

Comparison of the sequence identities and similarities

Sequences designated “Fragment”, “Partial”, or “Pseudogene” were removed from the multiple sequence alignments of the dynactin subunits. Poorly aligned positions and divergent regions of the alignments were removed using Gblocks (69) with the following parameters: A) The minimum number of sequences for a conserved position and the minimum of sequences for a flank position were set to the minimum (e.g. half the number of sequences plus one). B) The maximum number of contiguous nonconserved positions was set to 32000 and the minimum length of a block was set to 2. C) The parameter for the allowed gap position was set to ‘all’.

Sequence identity matrices (2D-matrix tables containing sequence identities scores for each pair of sequences) were calculated for each alignment using the method implemented in BioEdit (Tom Hall, http://www.mbio.ncsu.edu/bioedit/bioedit.html). Shortly, the reported numbers represent the ratio of identities to the length of the longer of the two sequences after positions where both sequences contain a gap are removed. Sequence similarity matrices were calculated with MatGAT (183) using the BLOSUM62 substitution matrix and setting the gap opening and extending penalties to 12 and 2, respectively.

Computing and visualizing phylogenetic trees

For calculating phylogenetic trees of single dynactin subunits only complete and partial sequences were included in the dataset. For the calculation of the tree of the yeast species, the sequences of the Arp1, Arp10, Cap1, Cap2, and dynactin2 subunit of each species of the Saccharomyces and the Schizosaccharomyces branch were concatenated. Missing protein sequences (*Zygosaccharomyces rouxii* Arp10, *Lodderomyces elongisporus* Arp10, *Ogataea parapolymorpha* dynactin2, and *Naumovozyma dairenensis* Cap2) were substituted by gaps. The phylogenetic trees were generated using two different methods for each dataset: 1. ProtTest was used to determine the most appropriate of the available 112 possible amino acid substitution models (70). The tree topology was calculated with the BioNJ algorithm and both the branch lengths and the model of protein evolution were optimized simultaneously. The Akaike Information Criterion with a modification to control for small sample size (AICc, with alignment length representing sample size) identified the LG model (184) to be the best for the dynactin3, dynactin5, dynactin6, Arp, and Cap datasets and the JTT model (185) for dynactin1, dynactin2, and dynactin4. Maximum likelihood (ML) analysis with estimated proportion of invariable sites and bootstrapping (1,000 replicates) were performed using RAxML (67). 2. Posterior probabilities were generated using MrBayes v3.1.2 (72) with the MPI option (186). Two independent runs with 5,000,000 generations, four chains, and a random starting tree were computed using the mixed amino-acid option. MrBayes used the WAG model (187) for all protein alignments. Trees were sampled every 1.000th generation and the first 25% of the trees were discarded as “burn-in” before generating a consensus tree. Phylogenetic trees were visualized with the CLC Sequence Viewer (http://www.clcbio.com) and FigTree (http://tree.bio.ed.ac.uk/software/figtree/) and are available as Additional File 2.1.10.1.

2.1.7 Competing interests

The authors declare that they have no competing interests.

2.1.8 Authors' contributions

BH performed all database related work, adjusted the CyMoBase software for specific dynactin related needs, did all data analysis and drafted the manuscript. MK assembled and annotated all sequences, and assisted in writing the manuscript. Both authors read and approved the final version of the manuscript.

2.1.9 Acknowledgements

First we would like to thank the Editor Henner Brinkmann and the anonymous reviewers for their very constructive comments that helped to considerably improve the manuscript. We also would like to thank the many sequencing centres and funding agencies for making unpublished sequence data available. Last, we want to thank Dr. Florian Odronitz and Klas Hatje for their help with CyMoBase and WebScipio, and Prof. Christian Griesinger for his continuous generous support. Sebastian Becker helped in the annotation of some of the sequence. This work has been funded by grants KO 2251/3-1 and KO 2251/3-2 of the Deutsche Forschungsgemeinschaft.

2.1.10 Additional files

2.1.10.1 Additional file 1 as ZIP

Zip archive of the Maximum Likelihood and Bayesian inference trees, and the sequence alignments of the dynactin subunits. The file includes all Maximum Likelihood and Bayesian trees of all dynactin proteins in the Newick format. The sequence alignments of the proteins are included in fasta format.

The zip file can be found in the corresponding publication.

2.1.10.2 Additional file 2 as PDF

Dynactin inventory of the analysed species. The file lists the presence and number of orthologs for each dynactin subunit for each analysed organism in taxonomic order.

The pdf file can be found in the corresponding publication.

2.1.10.3 Additional file 3 as PDF

Phylogenetic tree of dynactin1. The file contains the phylogenetic tree of dynactin1 highlighting the species- and branch-specific gene duplication events.

The pdf file can be found in the corresponding publication.

2.1.10.4 Additional file 4 as PDF

Additional file 4 Detailed description of the pseudo-transcripts of dynactin4. The file contains details about the pseudo-transcripts of dynactin4.

The pdf file can be found in the corresponding publication.

2.1.10.5 Additional file 5 as PDF

Sequence alignment of fungal dynactin4 proteins. The file contains the sequence alignment of the N-termini of several fungal dynactin4 (p62) subunits showing that the upstream methionines in Neurospora and Sordaria are most likely not the translation start sites.

The pdf file can be found in the corresponding publication.

2.1.10.6 Additional file 6 as PDF

Phylogenetic tree of Arp1. The file contains the phylogenetic tree of Arp1 focused on the vertebrate branch highlighting the Arp1 gene duplication event and subsequent branch-specific losses of Arp1 subtypes.

The pdf file can be found in the corresponding publication.

2.1.10.7 Additional file 7

Evolution of the conventional actin capping protein Cap

CapZ, the heterodimeric cytoplasmic actin-capping protein, caps the barbed-end of the Arp1 mini-filament. Because of their ubiquitous cytoplasmic function, the CapZ subunits Cap! (Cap1) and Cap" (Cap2) have also been found in plants and algae that do not contain any other dynactin subunit. CapZ homologs have been found in all available eukaryotes except the diatom *Thalassiosira pseudonana*. The reason is unknown but unlikely to be an unexpectedly large divergence because homologs were identified in the closely related species *Fragilariopsis cylindrus* and *Phaeodactylum tricornutum*. The reason could be a gap in the genome assembly although it is unlikely that both capping proteins were missing because of gaps.

Cap1 (Capα)

368 Capα sequences were identified in 289 species (Table 2.1.1, Additional File 2.1.10.1). All eukaryotes encode at least one copy of Cap1 (Capα). Several Cap1 homologs have been identified in vertebrates (Figure 2.1-1). However, the duplication pattern of the homologs of the Actinopterygii, Amphibia, Sauropsida, and Mammalia branches cannot easily be explained by the two whole genome duplications that happened at the origin of the vertebrates (64). Instead, there must have been either additional single gene duplications with further gene losses in certain branches, or the homologs must have diverged so far that their phylogenetic relationship cannot be resolved with current phylogenetic methods. Therefore, variant designations do not necessarily correspond to evolutionary relationships between these branches. The most remarkable exceptions in the branches listed above are: in between the mammalia branch, the rodents show another Cap1 duplication, which is related to Cap1A (Capα1), the lizard *Anolis carolinensis*

encodes an additional homolog compared to the other Sauropsidae, which is related to Cap1C (Capα3), and the Cap1 variants of *Brachydanio rerio* do not group to the variants of the other fish. Both major mammalian Cap1 isoforms, Cap1A (Capα1) and Cap1B (Capα2), are present in dynactin from brain (154) although their ratio is unknown.

The three Cap1 variants of *Homo sapiens* are located on chromosome 1 (variant A), chromosome 7 (variant B), and chromosome 12 (variant C), respectively (Figure 2.1-2). The variant A and variant B genes consist of 10 exons while variant C is encoded by a single exon. The identical gene structures of the variant A and variant B genes support their origin from a common ancestor. Variant C most probably derived by retrotransciption of a processed ancestral Cap1 gene, a process leading to pseudo-genes in most cases. However, because several dozens of EST and cDNA clones strongly support its cellular expression, the ancestral variant C gene must have gained a new promoter region.

Cap2 (Capβ)

299 Capβ sequences have been assembled from 292 species (Table 2.1.1, Additional File 2.1.10.1). Cap2 was found in all eukaryotic species except the diatom *Thalassiosira pseudonana*. Duplicates have been identified in *Trichomonas vaginalis* (five Cap2 homologs) and *Paramecium tetraurelia* (two Cap2 homologs). Mutually exclusive splicing (188) increases the vertebrates Cap2 diversity. The Cap2 gene of *Homo sapiens* is located on chromosome 1 and consists of ten exons of which the first eight are constitutively transcribed and exons 9A and 9B are mutually exclusive spliced (Figure 2.1-2). The transcripts including exon 9A are also called Capβ1 and the transcripts including exon 9B Capβ2. Many EST and cDNA clones support both mutually exclusive spliced exons in most vertebrates while invertebrates do not encode alternatively spliced Cap2 genes. Unfortunately, the Cap2 gene is not included in the fragmented draft assembly of *Petromyzon marinus* so that it is not clear yet whether the alternatively spliced form has been invented at the origin of the vertebrates or the Gnathostomata. Of the two Cap2 isoforms, only the Capβ1 isoform (including exon 9A) has been found in the dynactin complex (189).

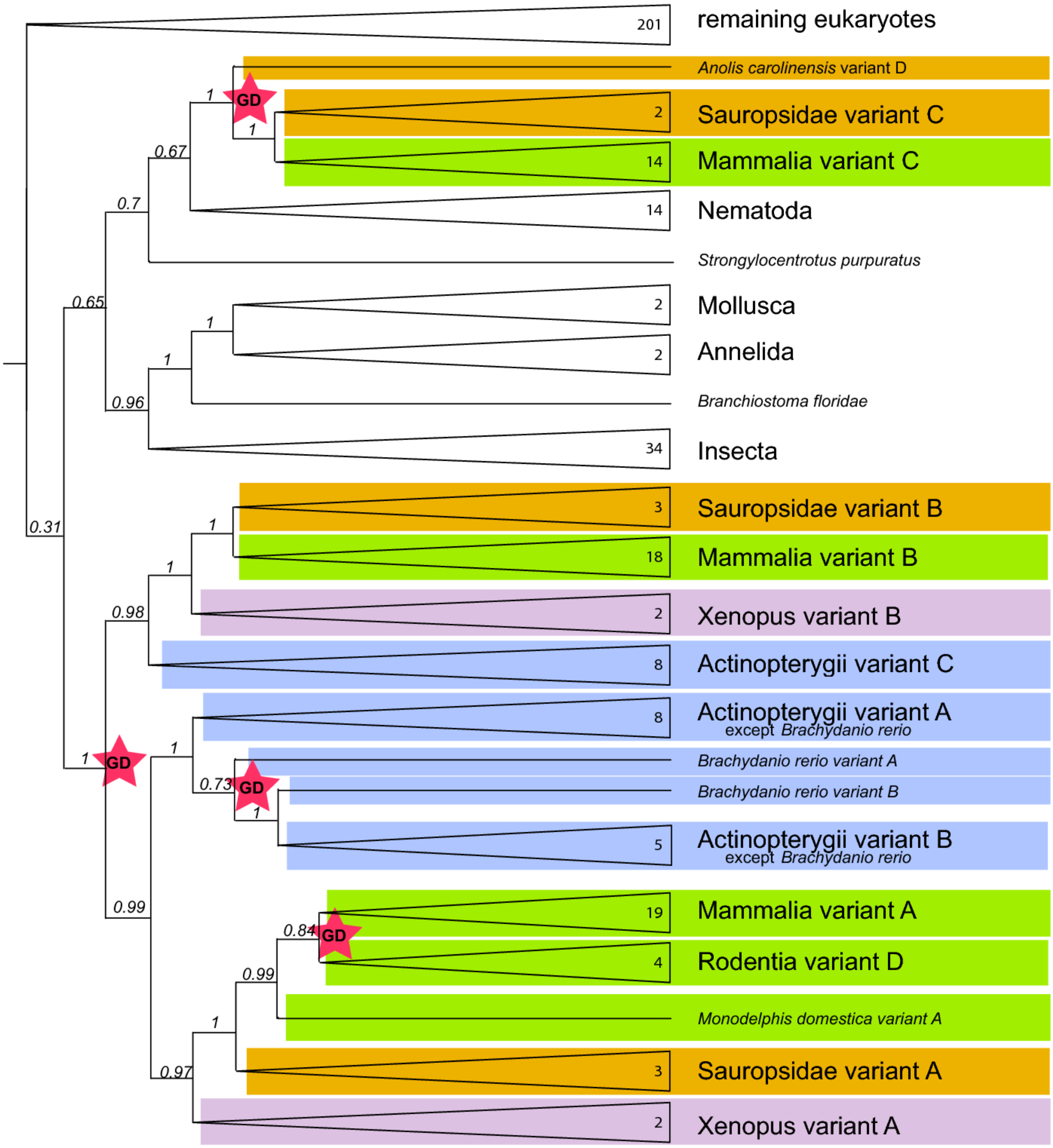


Figure 2.1-8: ***Phylogeny of Cap1 (Capα) in vertebrates.*** *Bayesian tree (MrBayes, WAG model) of the Cap1 protein dataset (368 sequences). Major branches have been collapsed to highlight the duplication events of Cap1 in Metazoa (red stars with "GD"). Small numbers within the triangles denote the number of included leafs. Branch support values correspond to Bayesian posterior probabilities*

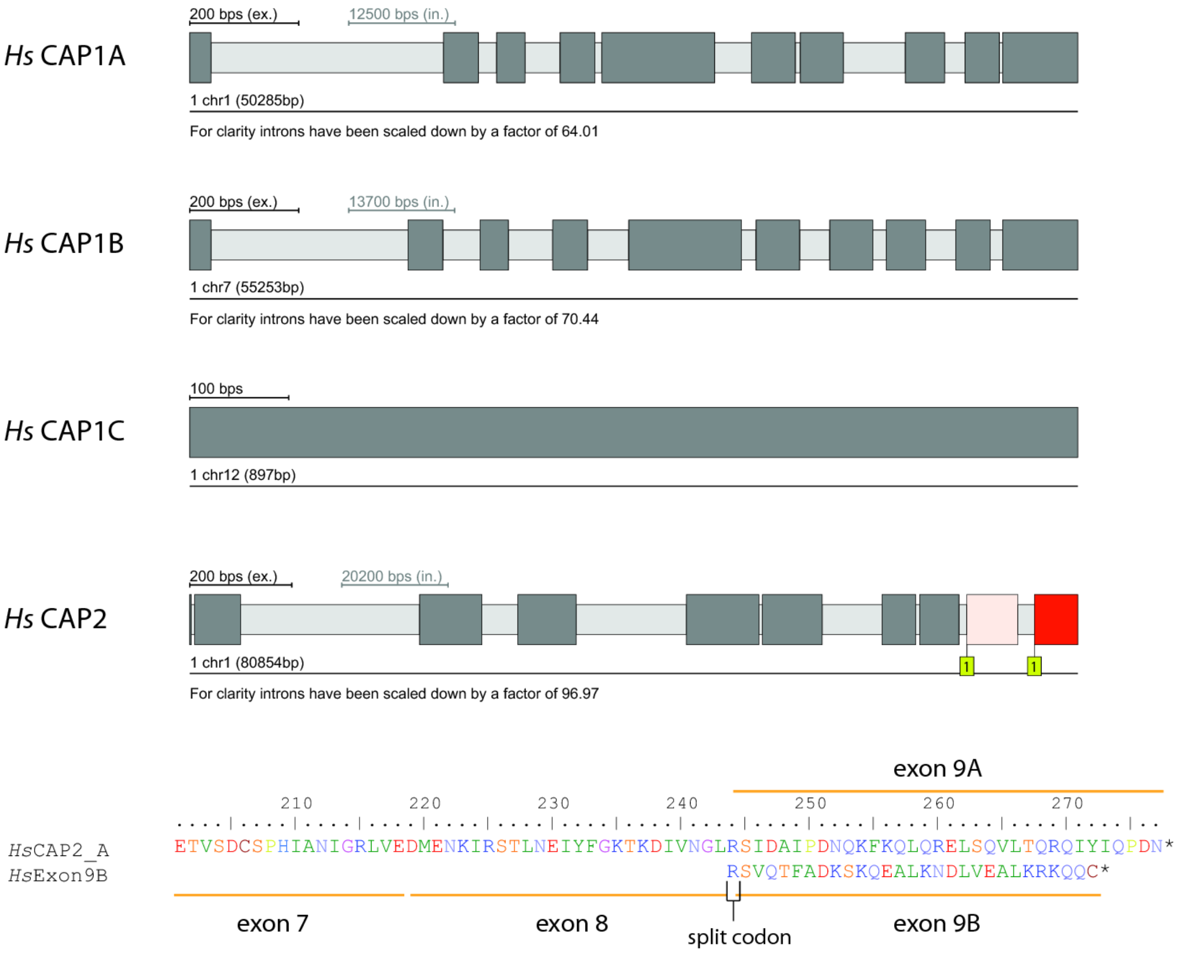

Hs CAP1A
200 bps (ex.)
12500 bps (in.)
1 chr1 (50285bp)
For clarity introns have been scaled down by a factor of 64.01
Hs CAP1B
200 bps (ex.)
13700 bps (in.)
1 chr7 (55253bp)
For clarity introns have been scaled down by a factor of 70.44
Hs CAP1C
100 bps
1 chr12 (897bp)
Hs CAP2
200 bps (ex.)
20200 bps (in.)
1 chr1 (80854bp)
1
1
For clarity introns have been scaled down by a factor of 96.97
exon 9A
210
220
230
240
250
260
270
HsCAP2_A
HsExon9B
ETVSDCSPHIANIGRLVEDMENKIRSTLNEIYFGKTKDIVNGLRSIDAIPDNQKFKQLQRELSQVLTQRQIYIQPDN*
RSVQTFADKSKQEALKNDLVEALKRKQQC*
exon 7
exon 8
split codon
exon 9B

Figure 2.1-9: ***Gene structures of the Capα and Capβ homologs of Homo sapiens.*** *Three variants of Capα have been identified in Homo sapiens. Variant A (found on chromosome 1) and B (found on chromosome 7), both have 10 exons. Variant C (found on chromosome 12) consists of one exon. For Capβ, one copy was found in the human genome comprised of 10 exons, of which exons 9A and 9B are alternatively spliced.*

2.1.10.8 Additional file 8 as PDF

Species table. The file contains all species of the analysis, their scientific names, the abbreviation as used in the sequence alignments and trees, the species taxonomy, references to sequencing centres, and publications if genome analyses have already been published.

The pdf file can be found in the corresponding publication.

2.2 A holistic phylogeny of the coronin gene family reveals an ancient origin of the tandem-coronin, defines a new subfamily, and predicts protein function

Christian Eckert, Björn Hammesfahr and Martin Kollmar[§]

Department of NMR-based Structural Biology, Max-Planck-Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Goettingen, Germany
[§] Corresponding author

BMC Evolutionary Biology

Published: 25 September 2011

BMC Evolutionary Biology 2011, 11:268 doi:10.1186/1471-2148-11-268 This article is available from http://www.biomedcentral.com/1471-2148/11/268

2.2.1 Abstract

Background

Coronins belong to the superfamily of the eukaryotic-specific WD40-repeat proteins and play a role in several actin-dependent processes like cytokinesis, cell motility, phagocytosis, and vesicular trafficking. Two major types of coronins are known: First, the short coronins consisting of an N-terminal coronin domain, a unique region and a short coiled-coil region, and secondly the tandem coronins comprising two coronin domains.

Results

723 coronin proteins from 358 species have been identified by analysing the whole-genome assemblies of all available sequenced eukaryotes (March 2011). The organisms analyzed represent most eukaryotic kingdoms but also cover every taxon several times to provide a better statistical sampling. The phylogenetic tree of the coronin domains based on the Bayesian method is in accordance with the most recent grouping of the major kingdoms of the eukaryotes and also with the grouping of more recently separated branches. Based on this “holistic” approach the coronins group into four classes: class-1 (Type I) and class-2 (Type II) are metazoan/choanoflagellate specific classes, class-3 contains the tandem-coronins (Type III), and the new class-4 represents the coronins fused to villin (Type IV). Short coronins from non-metazoans are equally related to class-1 and class-2 coronins and thus remain unclassified.

Conclusion

The coronin class distribution suggests that the last common eukaryotic ancestor possessed a single and a tandem-coronin, and most probably a class-4 coronin of which homologs have been identified in Excavata and Opisthokonts although most of these species subsequently lost the class-4 homolog. The most ancient short coronin already contained the trimerization motif in the coiled-coil domain.

2.2.2 Background

The coronin proteins, which were originally isolated as a major co-purifying protein from an actin-myosin-complex of the slime mold *Dictyostelium discoideum* (190), have since been identified in other protists (191,192), fungi (193), and animals (194), but are absent in plants. Coronins are a conserved family of actin binding proteins (195–197) and the first family member had been named coronin based on its strong immunolocalization to the actin rich crown like structures of the cell cortex in *Dictyostelium discoideum* (190). Coronins belong to the superfamily of the eukaryotic-specific WD40-repeat proteins (198,199) and play a role in several actin-dependent processes like cytokinesis (27), cell motility (27,28), phagocytosis (200,201), and vesicular trafficking (202).

WD-repeat motifs are minimally conserved regions of approximately 40-60 amino acids typically starting with Gly-His (GH) dipeptides 11-24 residues away from the N-terminus and ending with a Trp-Asp (WD) dipeptide at the C-terminus. WD40-repeat proteins, which are characterized by the presence of at least four consecutive WD repeats in the middle of the molecule, fold into beta propeller structures and serve as stable platforms for protein-protein interactions (198).

The coronin proteins have five canonical WD-repeat motifs located centrally. Since the region encoding the WD repeats is similar to the sequence of the beta-subunit of trimeric G-proteins the formation of a five-bladed beta-propeller was assumed for coronins (203). However, the determination of the structure of murine coronin-1 (*Mm*Coro1A (204)) demonstrated that the protein, analogous to the trimeric G-proteins, forms a seven-bladed beta-propeller carrying two potential F-actin binding sites. Apart from the central WD-repeats, almost all coronin proteins have a C-terminal coiled-coil sequence that mediates homo-oligomerization (205–207), and a short N-terminal motif that contains an important regulatory phosphorylation site in coronin-1B (28). In addition, each coronin protein has a unique region of variable length and composition following the conserved extension to the C-terminus of the beta-propeller.

Based on their domain composition coronins have originally been divided into two subfamilies, namely short and long coronins (208). Short coronins consist of 450 - 650 amino acids containing one seven-bladed beta-propeller and a C-terminal coiled-coil region. Furthermore, the N-terminal region of most known short coronins contains 12 basic amino acids. Since this motif is only present in coronin molecules, it has been suggested as a novel coronin signature (208). The longer types of coronin, also called POD or Coronin 7, possess two complete core domains in tandem but lack a coiled-coil motif. In the longer coronins, the sequence of the basic N-terminal motif is reduced to 5 amino acids. Based on phylogenetic relationships among the coronins, the Human Genome Organization nomenclature committee (HGNC) proposed a system in 2001 that grouped the short coronins into two classes resulting in a total of three subtypes (197). Very recently, a new nomenclature has been suggested dividing the coronins into twelve subclasses based on the analysis of about 250 coronins from most taxa (209). In contrast to previous systems, every mammalian coronin (and corresponding vertebrate homologs) was designated an own class resulting in seven vertebrate classes. Invertebrates were grouped into two classes, the fungi got an own class, coronins from alveolates were grouped with those from Parabasalids (class 10), and the remaining coronins from Amoeba, Heterolobosea, and Euglenozoa were combined into the twelfth class. This study constituted the first major phylogenetic analysis of the coronin family. However, this classification was not consistent with the latest phylogeny of the eukaryotes and homologs of some major branches like the stramenopiles were missing.

Here, we present the analysis of the complete coronin repertoires of all eukaryotic organisms sequenced and assembled so far. The distribution of all coronin homologs is in accordance with the latest taxonomy of the eukaryotes and reveals the origin of the tandem-coronin and another newly defined class in the last common ancestor of the eukaryotes.

2.2.3 Results

Identification and annotation of the coronin

The coronin protein genes were identified by TBLASTN searches against the corresponding genome data of the different species. The list of sequenced eukaryotic species as well as access information to the corresponding genome data has been obtained from diArk (210). Species that missed certain orthologs in the first instance were later searched again with supposed-to-be orthologs of other closely related species. In this iterative process all coronin family proteins have been identified or their loss in certain species or taxa was confirmed. Because verified cDNA sequences and protein predictions, which often contain mispredicted exons and introns even in the “annotated” genomes, are not available for most of the sequenced species, the protein sequences were assembled and assigned by manual inspection of the genomic DNA sequences. Exons have been confirmed by the identification of flanking consensus intron-exon splice junction donor and acceptor sequences (211). In addition, the gene structures of all coronin genes were reconstructed using WebScipio (13,212). Through comparison of the intron positions and splice-site phases in relation to the protein multiple-sequence alignment, several suspicious exon border predictions could be resolved and the protein sequences subsequently be corrected. The genomic sequences of many species contain several gaps due to the low coverage of the sequencing or problems in the assembly process. Only some of the gaps could be closed at the amino-acid level by analysing EST data.

Table 2.2.1: Data statistics

		coronin
Sequence		
	Total	723
	From WGS	700
	Domains	7
	Amino acids	
	Total pseudogenes	7
	Pseudogenes without sequence	3
Completeness		
	Complete	614
	Partials	44
	Fragments	62
Species		
	Total	358
	WGS-projects	323
	EST-projects	112
	WGS- and EST-projects	152

The coronin dataset contains 723 sequences from 358 organisms (Table 2.2.1). 614 sequences are complete, and an additional 44 sequences are partially complete. Sequences for which a small part is missing (up to 5%) were termed “Partials”, while sequences for which a considerable part is missing were termed “Fragments”. This difference has been introduced because Partials are not expected to considerably influence the phylogenetic analysis. Several of the genes were termed pseudogenes because they contain too many frame shifts, in-frame stop codons, and missing sequences to be attributed to sequencing or assembly errors.

Multiple sequence alignment, phylogenetic analysis, and classification

A multiple sequence alignment of all coronin family members has been created and extensively manually improved (Additional file 2.2.9.1). The basis of the alignment was the conserved coronin domain that consists of the β-propeller region and a subsequent conserved extension, which packs against the “bottom” surface of the propeller (204). This entire domain is conserved in all coronin homologs and we would therefore suggest naming it coronin-domain. The unique regions following the coronin-domain could only be aligned for homologs of closely related species. The C-terminal predicted coiled-coil regions were aligned again for all corresponding sequences to analyse potential oligomerization patterns (see below). The second coronin-domains of the tandem-coronins were also aligned to the coronin-domains for the phylogenetic analysis. One part of the coronin-domain in coronin-1D is encoded by a cluster of mutually exclusive exons (see below) and therefore the exon with the higher sequence identity to related homologs has been included in the alignment. The phylogenetic tree of the coronin family was calculated for 764 coronin-domains, including both coronin-domains of the tandem-coronins separately, using the Bayesian (Additional file 2.2.9.2) and the maximum-likelihood method (Additional file 2.2.9.3). The resulting trees were almost identical. However, the relations of the innermost nodes representing the most ancient relationships were best resolved using the Bayesian approach (Figure 2.2-1). The resulting phylogenetic tree is in accordance with the latest phylogenetic grouping of the six kingdoms of the eukaryotes (158,164,165) of which five are covered by the data analysed here. Thus, coronins of phylogenetic related species group together in the coronin family tree. In the coronin tree, not only the grouping is retained but also the evolutionary history of the branches. For example, the fungi separate as monophyletic group before the metazoans, and after the Amoeba.

The classification into subfamilies should at best include both the phylogenetic grouping of the protein family members and the domain organisation of the respective homologs. However, because most coronins contain a unique region between the coronin-domain and the C-terminal coiled-coil regions, several sub-branch specific domain organisation patterns evolved. To keep the coronin classification as simple as possible and to provide the highest consistency with previous classification schemes, the following classification is proposed: The classification should solely be based on the phylogenetic tree of the coronin-domains because it is in accordance with the phylogeny of the eukaryotes and contains the conserved part of the proteins that is the basis of the protein family. Metazoan species encode two phylogenetically distinct groups of coronins that have historically been named class-1 and class-2 coronins. Further variants of these classes should be named alphabetically, e.g. class-1A, class-1B, etc.. However, due to the independent whole-genome, genomic region, and single gene duplication events of certain phylogenetic branches these variant designations do not always refer to orthologs. For the mammalian coronins, which are the best analysed coronins, the suggested classification is almost entirely consistent with previous classifications (197) and the HGNC nomenclature except for “CORO6” and “CORO7”, which are here classified as coronin-1D and coronin-3, respectively. Class-3 comprises the tandem coronins. All members of this class group together in the phylogenetic tree, and only single homologs have been found in all species analysed. Class-4 is a newly defined class that contains coronins with variable numbers of C-terminal PH, gelsolin, and VHP domains, but also coronins with only very short sequences outside the coronin-domain. The other coronins group in accordance with the latest taxonomy of the species (Figure 2.2-1). In our opinion it does not add information or help the scientific community if those coronins were classified separately. In contrast to the metazoans, gene duplications in the branches of Amoeba, Excavata, and SAR are species-specific and do not warrant further subclassification at the moment. For example, instead of talking about a “class-11 coronin” and long explanations what type of coronins would belong to such a class, it would be easier, shorter, and less confusing to just say a “*Naegleria* coronin”, an “apicomplexan coronin” or a “yeast coronin”. The distribution of the coronins analysed here is summarized for some example species in Figure 2.2-2 including previously used names and classification schemes. The distribution of all coronins is found in Additional file 2.2.9.4. Coronin homologs are absent in Rhodophyta (*Cyanidioschyzon*, *Galdieria*), Viridiplantae, Microsporidia, Formicata (*Giardia*), and Haptophyceae (*Emiliania*).

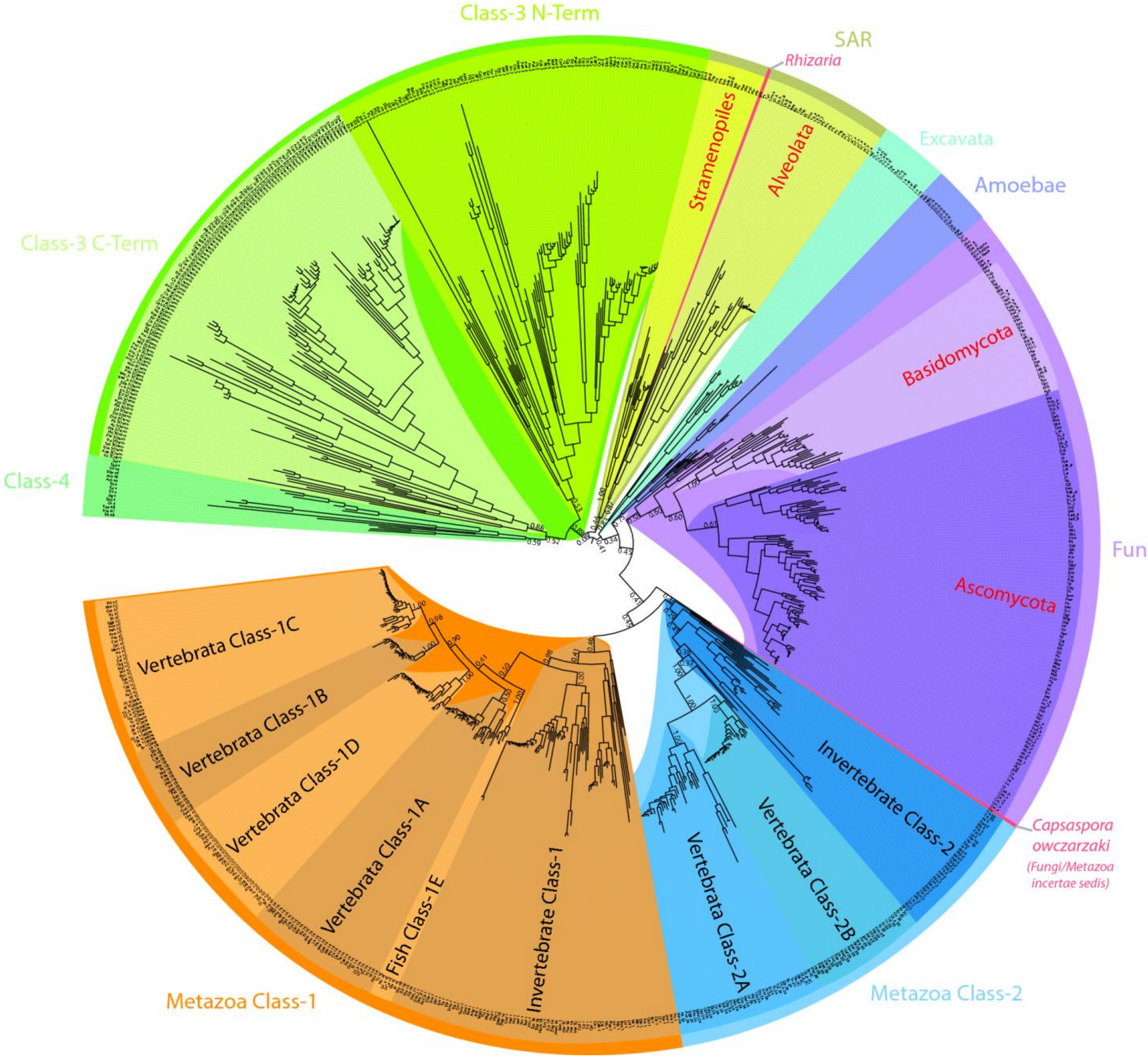


Figure 2.2-1: ***Phylogenetic tree of the coronin family.*** *The phylogenetic tree of the coronin family was calculated from the multiple sequence alignment of the conserved coronin domain using the Bayesian method. The unrooted tree was drawn with iTOL (213) and branches were coloured according to class and taxonomic distributions. For an extended representation of the tree including all posterior probability values see Additional file 2.2.9.3.*

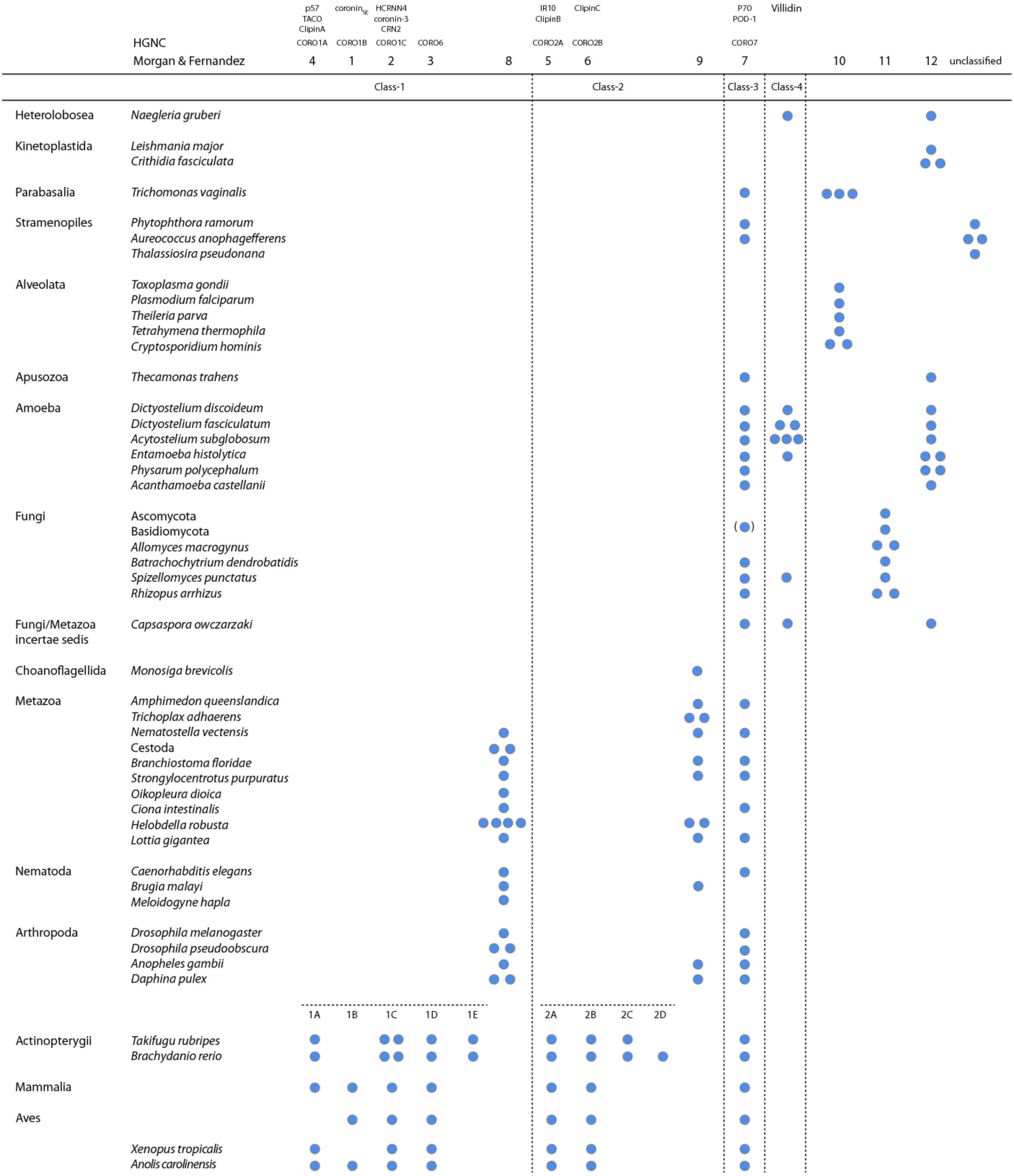


Figure 2.2-2: ***Coronin repertoire of selected species of major taxa and branches.*** *The coronins of several representative species for most eukaryotic taxa and branches are listed (for the list of all species see Additional file 2.2.9.4). On top, alternatively used names and classification schemes are given for better comparison and orientation.*

Short coronins (class-1, class-2, and unclassified coronins)

The domain organisations of most short coronins (class-1, class-2, and unclassified coronins) are similar. They consist of the 390 amino-acid long coronin-domain followed by a short unique domain and a C-terminal short coiled-coil region (about 30-40 amino acids, Figure 2.2-3). The unique regions are conserved in branches (e.g. the vertebrates have similar regions, as do the arthropods, the nematodes, etc.), but are not conserved for major taxa (e.g. fungi, Metazoa, stramenopiles).

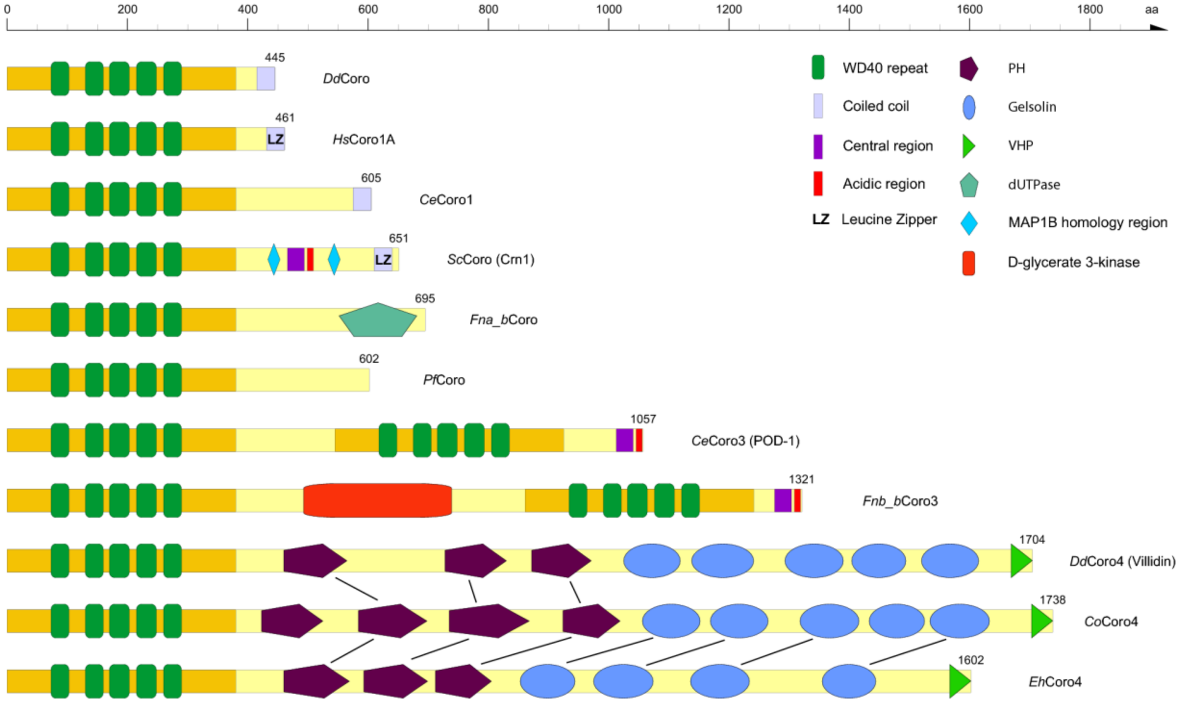


Figure 2.2-3: ***Domain organisation of representative coronins.*** *A colour key to the domain names and symbols is given on the right except for the coronin domain that is coloured in orange. The abbreviations for the domains are: WD, WD repeat; PH, pleckstrin-homology domain; LZ, leucine zipper; VHP, villin headpeace domain.*

The *Saccharomyces cerevisiae* coronin, ScCoro (CRN1), is known to bind to microtubules via its unique region between the β-barrel domain and the coiled-coil oligomerization region (Figure 2.2-3, (214)). Two short regions showing homology to the microtubule-binding regions of MAP1B mediate this interaction. However, the MAP1B sequence motif is very short (about ten residues) and not very specific comprising mainly glutamate and lysine residues (214). If the corresponding motifs in ScCoro are responsible for microtubule-binding then all yeast and *Schizosaccharomyces* coronins should be able to bind to microtubules because they contain motifs with similar amino acid compositions. A similar motif or region could not be identified in the Pezizomycotina coronins. While these supposed microtubule-binding regions mainly consist of glutamate, lysine, proline, serine, and threonine and are not even conserved in very closely related yeast species, the *Saccharomyces cerevisiae* coronin, *Sc*Coro, has very recently been described to contain a CA domain (C: central; A: acidic; (215)). This domain, with which *Sc*Coro activates and inhibits the ARP2/3 complex depending on concentration (215), is similar to CA domains in WASP family proteins (216). The CA domain is well conserved but distinct within the Saccharomyceta clade (Pezizomycotina and Saccharomycotina, Figure 2.2-4).

Surprisingly, the coronins of the Tremellomycetes (e.g. *Filobasidiella/Cryptococcus* species) that belong to the Basidiomycota encode a C-terminal dUTPase domain (deoxyuridine triphosphatase domain) instead of the coiled-coil region (Figure 2.2-3). These coronin sequences are supported by many EST/cDNA clones for several of the *Filobasidiella* species extending from the coronin domain to the stop-codon. In addition to this dUTPase domain, the *Filobasidiella* species contain a further dUTPase in the genome that is conserved in the other Basidiomycotes, and also the other fungi. The dUTPase domains of the Tremellomycetes coronins contain all characteristic dUTPase domain motifs (217) and are therefore supposed to constitute enzymatically active domains. dUTPases typically form homotrimer active site architectures with all monomers contributing conserved residues to each of the three active sites (217). Except for the prediction of trimerization of these coronins, which could be mediated by the dUTPase domains instead of the coiled-coil domains in the other coronins, it needs experimental data to link the function of actin filament structure remodelling by coronins to dUTP nucleotide hydrolysis in DNA repair by dUTPases.

Class-3 coronins

Class-3 coronins (Type Ⅲ coronins) comprise homologs that encode two coronin domains arranged in tandem (197). These two coronin domains are separated by unique regions, and class-3 coronins do not encode coiled-coil domains. As recently reported (215) the class-3 coronins also encode a CA domain similar to the CA domain of the WASP family proteins at their C-termini (Figure 2.2-3). Based on the multiple sequence alignment of 112 class-3 coronins from all major branches of the eukaryotes the position of the C-region has slightly been adjusted in comparison with a previous analysis (Figure 2.2-4; (215)). Although the C-region of the class-3 coronins is not as conserved as similar regions in the yeast short coronins or in WASP family proteins, the characteristic pattern of hydrophobic residues concluded by a basic residue is visible in the homologs of all species (Figure 2.2-4). In contrast to the short coronins, the unique region between the C-terminal coronin-domain and the conserved CA-domain is short (20-30 amino acids).

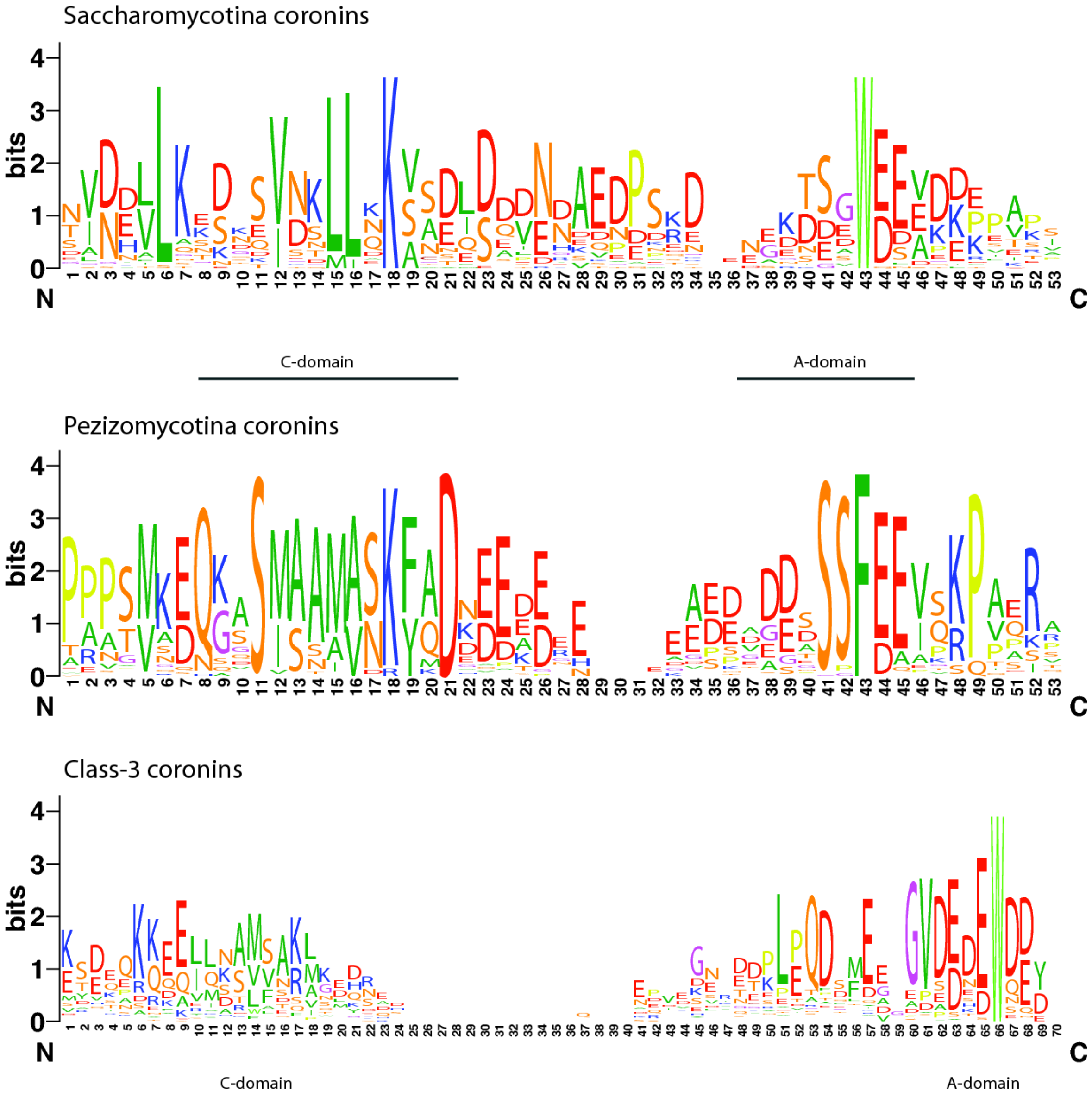


Figure 2.2-4: ***Sequence conservation in the CA domains.*** *The sequence logos illustrate the sequence conservation within the multiple sequence alignments of the CA domains of the Saccharomycotina, the Pezizomycotina, and the class-3 coronins. The CA domains of the Saccharomycotina and the Pezizomycotina are located within the unique regions of the short coronins while the CA domain of the class-3 coronins is at the C-termini of the proteins like in WASP family proteins. The regions between the C and the A domains are of variable length.*

Like for the short coronins the *Filobasidiella* species have surprising and species-specific tandem-coronins. The *Filobasidiella* class-3 coronins have a D-glycerate 3-kinase domain between the two coronin-domains (Figure 2.2-3). Only the termini of the *Filobasidiella* class-3 coronins are supported by EST/cDNA data, but long exons bridge the N-terminal coronin-domain and the glycerate 3-kinase domain, as well as the glycerate 3-kinase domain and the C-terminal coronin-domain. As found for the dUTPase domain of the short coronins, the *Filobasidiella* species contain an additional D-glycerate 3-kinase that has homologs in the other fungi and also in plants. Why it is advantageous to connect an actin-filament binding function to a glycerate 3-kinase needs experimental evaluation. The glycerate 3-kinase domain is not found in the class-3 coronins of the other Basidiomycotes. Except for the *Filobasidiella* species only the insects have long insertions between the two coronin-domains of their class-3 coronins. These insertions are highly conserved, about 300 residues long, and do not show any homology to known domains, sequence motifs, and other proteins.

In contrast to the related species *Rhizopus arrhizus* and *Phycomyces blakesleeanus* the coronin-3 of *Mucor circinelloides* consists of only the second coronin-domain of the tandem. We can exclude the possibility of this being an artefact of the genome assembly for three reasons. First, the genome sequence is continuous around *Muc*Coro3. Secondly, there is no homology to any part of the N-terminal coronin-domain of *Rha*Coro3 or *Phb*Coro3 in the genome although the sequence identity of the C-terminal coronin-domains is about 65%. And finally, there is a TATA-box shortly upstream of the *Muc*Coro3 gene. Because a coronin-3 has already been present in the most ancient eukaryote the loss of the N-terminal coronin-domain must be specific to *Mucor circinelloides*.

Class-4 coronins

Based on the phylogenetic tree (Figure 2.2-1) and the domain composition of the protein homologs, another coronin class can be defined for which the *Dictyostelium discoideum* homolog, also called villidin (29), would be a representative (Figure 2.2-3). We suggest naming members of this class class-4 coronins. Most class-4 coronins consist of an N-terminal coronin-domain followed by three to four PH domains, four to five gelsolin domains, and a C-terminal villin headpiece domain (VHP). Class-4 coronins were identified in two of the major kingdoms of the eukaryotes, in excavates and opisthokonts. Furthermore, they are found in several of the sub-branches of the opisthokonts, in amoebae, fungi, and the fungi/metazoa incertae sedis branch. Because class-4 coronins from different species often contain different numbers of PH and gelsolin domains, domain gain and loss events must have happened in the respective branches or single species. However, there are not enough coronin-4 homologs identified yet to reconstruct the evolution of these regions. In addition to these multi-domain class-4 coronins there is a group of class-4 coronins that just consists of the conserved coronin domain and is restricted to some Amoebae species yet.

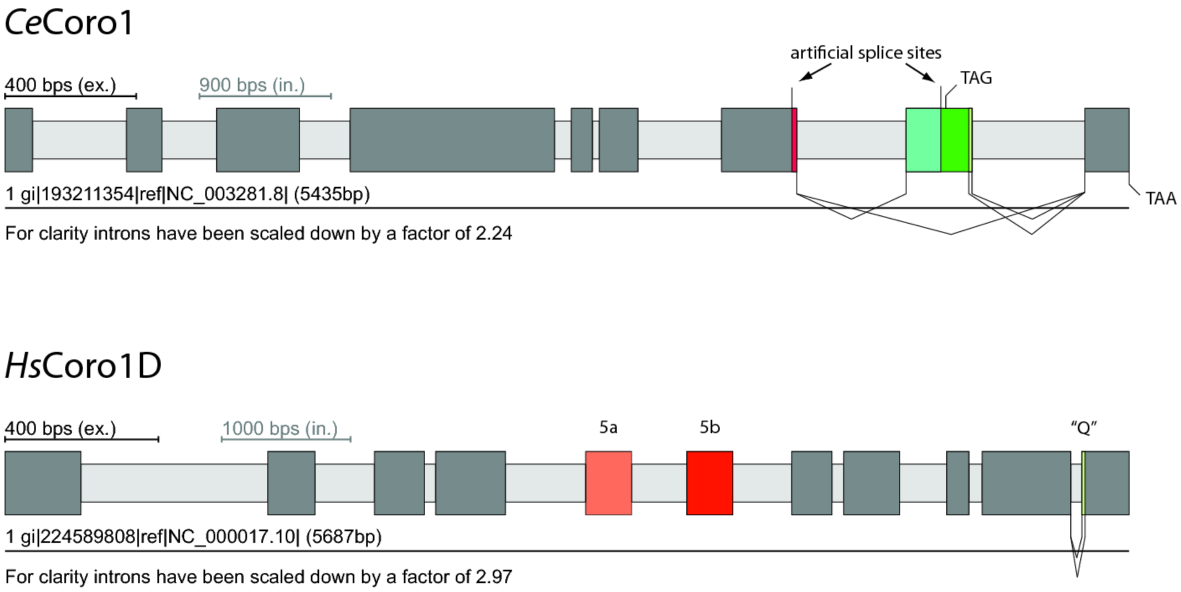
Alternatively spliced coronins

Alternative splice forms have been reported for two coronin homologs: five variants of coronin from *Caenorhabditis elegans* (218), *Ce*Coro1 (Figure 2.2-5), and three variants for coronin-1C from human (219), *Hs*Coro1C. The described splice variants do not concern the beta-barrel domain but the structurally low-complexity region prior to the coiled-coil region in *Ce*Coro1 and elongations of the N-terminus of *Hs*Coro1C, respectively. In the reported analysis of *Ce*Coro1 (218) two splice sites (the alternative 3’-splice site of exon7 and the alternative 5’-splice site of exon8) do not obey the conventional splicing rules. Alternative 5’-splicing of exon8 would lead to a premature stop-codon. In the four additional *Caenorhabditis* strains analysed here, *C. briggsae*, *C. japonica*, *C. remanei*, and *C.brenneri*, alternative 5’-splicing of exon8 would not lead to a premature stop-codon at the same position as in *C. elegans* but to transcripts of various lengths. The same accounts for several of the other available nematode coronin-1 genes. Given the high conservation of the nematode coronin-1 genes, especially the *Caenorhabditis* genes, and the completely uncommon nature of the potential splice sites, the reported alternative 3’-splice site of exon7 and 5’-splice site of exon8 are most probably artificial results. An alternative 3’-splice site has been reported for exon8 of *Ce*Coro1 comprising two amino acids (218). Similar splice sites were identified in the genes of the other analyzed *Caenorhabditis* species but not in other nematodes. This splice site is thus also either an artificial result or specific for the *Caenorhabditis* branch. In addition, skipping of exon8 has also been reported to lead to an alternative transcript (218). The intron position and reading frame of exon8 of *Ce*Coro1 is conserved in all analyzed nematode coronin-1’s except for the *Strongyloides rattii* coronin-1, which consists of only one exon, and the *Pristionchus pacificus* coronin-1, which has introns at different positions. Compared to the full-length transcript, the other alternative splice forms of *Ce*Coro1 are of low abundance (see Fig. 2 in (218)). Because the integrity of exon8 of *Ce*Coro1 (intron positions around the conserved coding sequence of exon8) is not conserved in nematodes but the corresponding amino-acid sequence, alternative splicing of nematode coronin-1 is either restricted to some sub-branches or an artificial result of the *Ce*Coro1 analysis.

Alternative splicing of human coronin-1C results in two additional transcripts derived from alternative transcription start sites encoded by an additional upstream exon, compared to the normal start site as found and conserved in all other coronin proteins (219). These alternative splice forms seem to be restricted to modern primates (human, chimpanzee, gorilla, orang-utan, and gibbon) and have been discussed in detail elsewhere (219).

We have identified alternative splice variants for coronin-1D (Figure 2.2-5), a coronin subfamily restricted to vertebrates. A cluster of two mutually exclusively spliced exons, exon5a and exon5b, was identified in all tetrapods. The amino-acid sequences corresponding to exon5 of the fish genes are more similar to exon5b than to exon5a. Thus, exon5a is the result of an exon duplication event that either occurred after the separation of tetrapods from fishes or at the onset of the vertebrates, where exon5a has been lost in the ancestor of the fishes. Exon5 represents the sequence of almost the entire fourth WD repeat (fifth blade in the β-propeller) starting in the middle of the fourth β-strand of blade four. By exchanging the fourth WD repeat the vertebrates could fine-tune the function of the coronin-1D beta-barrel domain. Vertebrate coronin-1D (CORO6) has not been analyzed experimentally yet and its specific function is unknown.

Further alternative transcripts are derived from mammalian coronin-1D genes by alternative 5’-splicing of the last exon, exon10. This alternative splicing results in one additional glutamine residue and is conserved in all 22 analyzed mammalian coronin-1D’s except for *Ailuropoda melanoleuca* (giant panda), *Loxodonta africana* (elephant), *Myotis lucifugus* (little brown bat), and *Bos taurus* (cow).


*Ce*Coro1

*Hs*Coro1D

artificial splice sites
TAG
400 bps (ex.)
900 bps (in.)
1 gi|193211354|ref|NC_003281.8| (5435bp)
TAA
For clarity introns have been scaled down by a factor of 2.24
400 bps (ex.)
1000 bps (in.)
5a
5b
“Q”
1 gi|224589808|ref|NC_000017.10| (5687bp)
For clarity introns have been scaled down by a factor of 2.97

Figure 2.2-5: ***Gene structures of alternatively spliced coronins.*** *The cartoons outline the gene structures of the alternatively spliced coronin-1 gene from Caenorhabditis elegans, CeCoro1, and the coronin-1D gene from Homo sapiens, HsCoro1D. The alternatively spliced CeCoro1 gene contains a differentially included exon8, which has an additional alternative 3’-splice site, leading to three transcripts. The other two described splice sites, an alternative 3’-splice site of exon7 and an alternative 5’-splice site of exon8 (218), are most probably artificial. The HsCoro1D gene contains a cluster of two mutually exclusive spliced exons, exon5a and exon5b, and an alternative 5’-splice site of exon10. Dark grey bars and light grey bars mark exons and introns, respectively, and alternative exons and splice sites are coloured.*

Oligomerization

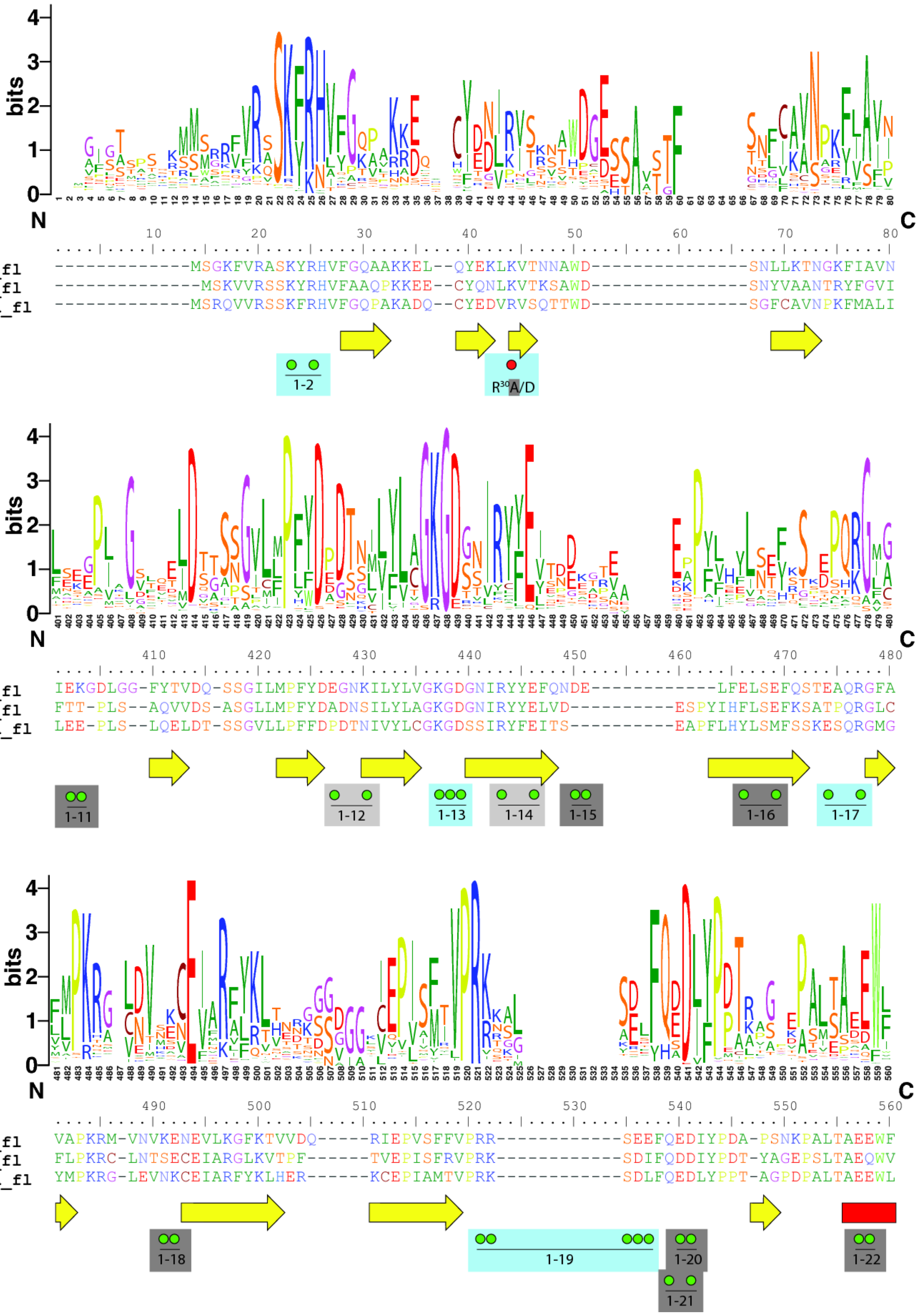
Most of the short coronins have predicted coiled-coil domains at the C-terminus that are the bases for their supposed oligomerization. Initially, coronins have been proposed to form dimers (203), the most common form of coiled-coil multimerization. In the last decade, a few coronin homologs were biochemically purified and analyzed. Accordingly, the *Xenopus laevis* coronin-1C (XcoroninA) has been shown to form a dimer (220) while an oligomeric state has been found for human coronin-1C (coronin 3; (206), and the *Saccharomyces cerevisiae* coronin (CRN1) trimerizes (215). Parallel trimer formation has also been shown in a crystal structure of the coiled-coil domain of mouse coronin-1A (207) revealing a conserved motif determining the trimeric structure: R_1-$[ILVM]_2$-X_3-X_4-$[ILV]_5$-E_6. In this motif arginine forms a salt-bridge with glutamate at the surface of the coiled-coil structure and the aliphatic side chain moieties of arginine and glutamate pack against the hydrophobic residues at positions 2 and 5 of the motif shielding them from solvent. Mutation of the arginine to lysine leads to a concentration-dependent equilibrium between trimers and tetramers with tetramers forming at high concentration, while mutation to alanine or norleucine leads to tetramers (207). Mutation of the invariant arginine to glutamine in the trimerization motif of human matrilin-1 leads to tetramers (221). Unfortunately, the switching of arginine and glutamate in the respective positions has not been analyzed yet. We would expect that such a switch should be as stable as the original motif. Thus, to predict the oligomerization state we have analyzed all coronin coiled-coil regions for the presence of the trimerization motif. Accordingly, all 233 class-1 coronins have the classical motif, except for *Dp*Coro1B and *Drp*Coro1B (*Drosophila pseudoobscura* and *persimilis*; Lys at position 1), and *Nv*Coro1 (*Nematostella*; Cys at position 2), and are thus predicted to form trimers. This would include the *Xenopus* Coro1C that has, however, been shown to exist as a dimer (220). The situation is more diverse for the class-2 coronins. The invertebrate coronins contain the trimerization motif, except for *Amq*Coro2 (*Amphimedon*; Ser at P1), *Her*Coro2A (*Helobdella*; Lys at P1), *Her*Coro2B (Phe at P1), *Myd*Coro2 (*Mayetiola*; Gln at P6), and the nematode class-2 coronins (Cys at P2). Almost all fish class-2 coronins contain the trimerization motif, but the other vertebrate class-2 coronins have conserved mutations. The tetrapod class-2A coronins encode a glutamine instead of the invariant arginine, which would turn them to tetramers in analogy to matrilin-1 (221). The tetrapod class-2B coronins contain glutamine instead of the glutamate at position 6 of the motif, a substitution whose effect has not been analyzed yet.

About half of the analyzed fungal coronins have the classical trimerization motif. The most common substitutions that are found in all Schizosaccharomyces and most Basidiomycota coronins are lysines or glutamines instead of the arginine at position 1. While the coiled-coil region is conserved in general, substitutions happened in specific species but not in whole branches (except for the Schizosaccharomyces). Therefore, we would expect all fungal coronins to form trimers. All Amoeba coronins, the Stramenopiles coronins (exceptions: *Frc*Coro a His at P1, *Bh*Coro_B a Asn at P6, *Aua*Coro_B a Lys at P1), the *Trichomonas* and *Naegleria* coronins contain the classical trimerization sequence motif in the coiled-coil region. Interestingly, the kinetoplastid coronins have the salt-bridge switched in the motif and should thus also be able to form trimers. From the Alveolata, only the Ciliophora (e.g. *Tetrahymena*) and Coccidia (e.g. *Toxoplasma gondii*) coronins contain coiled-coil domains, and only the Coccidia contain the trimerization motif.

These are, however, predictions based on the existence of the proposed trimerization motif. The motif has been identified in 86% of all short, autonomous, and parallel three-stranded coiled-coils while it is also observed in 9% of the antiparallel trimers and in 5% of the parallel and antiparallel dimers (207). Thus, although most short coronins are predicted to form trimers some might nevertheless function in other oligomeric states in the cell. The oligomerization state can ultimately only be shown in experiments, which have, however, been done for just a few of the coronins yet.

F-actin binding

F-actin binding is one of the common properties of coronin proteins. The extended multiple sequence alignment presented here together with the recently determined crystal structure of murine coronin-1A (204) now allows a reevaluation of previous mutagenesis studies. Truncation studies have shown that the coronin domain, including the β-propeller and its C-terminal extension, is necessary for F-actin binding (214,222). Mapping the sequence conservation within 13 short coronin members onto the surface of the crystal structure revealed two regions, one formed by blades 1, 6, and 7 and one formed by blades 6 and 7 and a portion of the C-terminal extension, to represent possible actin binding sites (204). Subsequently, several surface-exposed charged amino acids have been mutated to alanine or substituted by reversed charges in human coronin-1B and their F-actin binding affinity has been analyzed ((223), Figure 2.2-6 red dots, see also Additional file 2.2.9.5). Only the R30D mutation abolished actin binding in vitro. Although an arginine is the most prevalent amino acid at this position it is often substituted by a lysine or a proline (Figure 2.2-6). The multiple sequence alignment of the coronins also does not show a trend towards a class-specific substitution. For example, while a proline is found at this position in all vertebrate class-2 coronins, arginines, lysines, asparagines, prolines, threonins, and tyrosins are found in invertebrate class-2 coronins. At least negatively charged amino acids are not found in any of the coronin domains at this position. Recently, systematic mutagenesis of charged surface-exposed residues of yeast coronin revealed a patch of residues extending over the top and one side of the β-propeller that abolished actin binding when mutated to alanine (Figure 2.2-6, green dots (224)). The analysis of the conservation within the coronin proteins shows that many of the substitutions in both studies have been performed on marginally conserved residues (e.g. E^{215}A/K, K^{216}A/E, 1-11, 1-15, 1-16). Thus, it is not surprising that coronins with mutations of these residues are able to bind F-actin. As actin binding is one of the common functions of coronins and actins belong to the highest conserved protein families the actin binding surface of the coronins is also expected to be highly conserved. Most of the residues that were found to abolish actin binding when mutated to alanine are strongly conserved (Figure 2.2-6). The few residues that are highly conserved but do not influence actin binding might be interaction sites for other proteins like cofilin.



```
             10        20        30        40        50        60        70        80
ScCoro__fl   -------------MSGKFVRASKYRHVFGQAAKKEL--QYEKLKVTNNAWD---------------SNLLKTNGKFIAVN
DdCoro__fl   --------------MSKVVRSSKYRHVFAAQPKKEE--CYQNLKVTKSAWD---------------SNYVAANTRYFGVI
MmCoro1A_fl  -------------MSRQVVRSSKFRHVFGQPAKADQ--CYEDVRVSQTTWD---------------SGFCAVNPKFMALI

             410       420       430       440       450       460       470       480
ScCoro__fl   IEKGDLGG-FYTVDQ-SSGILMPFYDEGNKILYLVGKGDGNIRYYEFQNDE------------LFELSEFQSTEAQRGFA
DdCoro__fl   FTT-PLS--AQVVDS-ASGLLMPFYDADNSILYLAGKGDGNIRYYELVD----------ESPYIHFLSEFKSATPQRGLC
MmCoro1A_fl  LEE-PLS--LQELDT-SSGVLLPFFDPDTNIVYLCGKGDSSIRYFEITS----------EAPFLHYLSMFSSKESQRGMG

             490       500       510       520       530       540       550       560
ScCoro__fl   VAPKRM-VNVKENEVLKGFKTVVDQ-----RIEPVSFFVPRR------------SEEFQEDIYPDA-PSNKPALTAEEWF
DdCoro__fl   FLPKRC-LNTSECEIARGLKVTPF------TVEPISFRVPRK------------SDIFQDDIYPDT-YAGEPSLTAEQWV
MmCoro1A_fl  YMPKRG-LEVNKCEIARFYKLHER------KCEPIAMTVPRK------------SDLFQEDLYPPT-AGPDPALTAEEWL
```

Figure 2.2-6: ***Sequence conservation within the actin binding region.*** *The sequence logos illustrate the sequence conservation within the multiple sequence alignments of the coronin domains. Here, only the N- and C-termini of the coronin domains are shown because most of the residues implicated in actin binding map to these regions. For the representation of the entire coronin domain see Additional file 2.2.9.5. For better orientation, the sequences of three representative coronins are shown: the yeast coronin as the main target of mutagenesis experiments, the Dictyostelium coronin as the founding member of the protein family, and the murine coronin-1A of which the crystal structure is known. Secondary structural elements as determined from the crystal structure are drawn as yellow arrows (β-strands) and red boxes (α-helices). Green dots point to amino acids of ScCoro that have been mutated to alanine (224) and red dots highlight mutagenesis studies in HsCoro1B (223). Light-blue boxes highlight mutations that abolished actin binding, dark-grey boxes represent mutations that did not influence actin binding, and light-grey boxes point to mutations in yeast coronins that could not be expressed and tested.*

2.2.4 Discussion

Here, we have analyzed 723 coronins from 358 species. For 323 species whole genome sequence data was available allowing a “holistic” analysis of the coronin protein family. In addition, the whole genome assemblies of 69 species have been analyzed that in the end did not contain any coronin homolog. These species include Rhodophyta (*Cyanidioschyzon*, *Galdieria*), Viridiplantae, Microsporidia, Formicata (*Giardia*), and Haptophyceae (*Emiliania*). A sequence alignment of the coronin proteins was created and extensively improved manually. The phylogenetic analysis of the conserved coronin domain, which is also included in the crystal structure (204), using the Bayesian method showed that the grouping of the coronins is completely in accordance with the latest phylogeny of the eukaryotic species (Figure 2.2-1, (158,164,165)). Subsequently, we analyzed the coronin tree with respect to established and proposed classifications defining subfamilies. Two major schemes are currently in use, the old one established by the HGNC (197) and a more recent one expanding the number of classes from three to twelve (209). Essentially, the later classification re-defines subclasses of the HGNC scheme as separate classes, e.g. 1A and 1B become class-4 and class-1, respectively, and groups some branches to new classes. However, some coronins still remained unclassified and several classes have been proposed, like the invertebrate metazoan classes 8 and 9, although the contributing members did not form monophyletic branches in the underlying protein family tree. The proposed classes 10 and 12 contain members of unrelated taxonomic branches, probably because these coronins were adjacent in the tree figure. In addition, the Entamoeba tandem-coronin did not group to the other tandem-coronins. Thus, this classification is not consistent with the taxonomy of the eukaryotes. In addition, homologs of major branches were missing in the analysis like those from stramenopiles. We do not intend to add confusion to the classification of the coronin family but want to suggest a reliable and, considering future genome sequencing projects, expandable scheme. Two major reasons support the future use of the HGNC scheme although it needs some minor adjustments. The classification by Morgan and Fernandez (209) of coronins outside the metazoans is not consistent with the latest taxonomy of the eukaryotes and therefore not adaptable to our more comprehensive coronin tree. In addition, it is well known that two whole-genome-duplications are the reason for the expansion of gene homologs at the origin of the vertebrates (64), while another whole-genome-duplication happened at the origin of the Actinopterygii (225,226). Thus retaining the orthology between non-vertebrate and vertebrate coronins in class-numbers would be desirable but has also been abandoned by Morgan and Fernandez (209). Here, we adapted the HGNC classification except for renaming CORO6 and CORO7 (HGNC) to coronin-1D and coronin-3, respectively, numbering additional fish coronins as coronin-1E, coronin-2C, and coronin-2D, and defining the new coronin class-4. The term “class” is equivalent to the term “Type” used by the Bear group in recent reviews (197,227). However, we prefer the term "class" to be consistent with the terminology used for other protein families (e.g. the myosin family (125,228)) and therefore to facilitate the work with databases and search engines in the future.

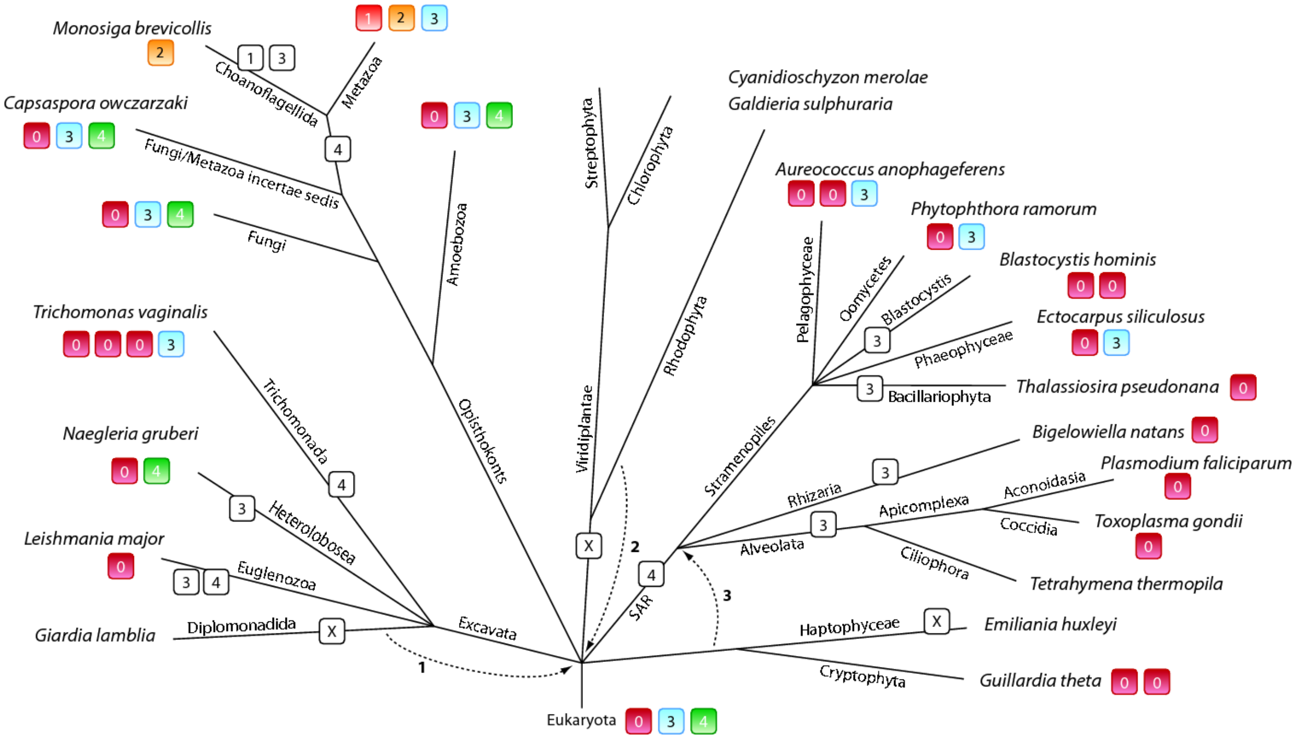


Figure 2.2-7: ***Evolution of the coronin protein family with respect to the species evolution.*** *Schematic representation of the most widely accepted eukaryotic tree of life. Branch lengths are arbitrary. The coronin inventories of certain taxa and specific species have been plotted to the tree with class numbers given in colour-coded boxes. "O" stands for "Orphan", the unclassified short coronins. The numbers on the arrows refer to alternative placing of the respective taxa:* ***1****: The independence of the Diplomonadida (instead of grouping them to the superkingdom Excavata) is supported by (157).* ***2****: The monophyly of the Rhodophyta is supported by (160,164).* ***3****: Grouping the Haptophyceae and Cryptophyta to the SAR is supported by (160–162).*

Class-4 coronins represent a new type of coronins that are present in Excavata (*Naegleria gruberi*), Amoebae, fungi (*Spizellomyces punctatus*), and the Fungi/Metazoa incertae sedis branch. Most class-4 coronins consist of the N-terminal coronin domain followed by two to three PH, four to five gelsolin, and a C-terminal VHP domain. The first representative of this subfamily has been identified in *Dictyostelium discoideum* and called villidin because of the homology of its gelsolin and VHP domains to villin (29). The homology of villidins WD-repeat region to coronin has been recognized later on (229,230) suggesting villidins origin through a fusion of the coronin domain with villin. Villin is the founding member of a superfamily of proteins containing three to six gelsolin domains (reviewed in (229,231)). Like villidin (class-4 coronins), villin, supervillin, and protovillin also contain a C-terminal VHP domain. Alignment of villin to the class-4 coronins gelsolin domains shows that the class-4 coronins have lost the first gelsolin domain of villin. The first gelsolin domain of villin is associated with dimerization, actin filament capping, nucleation, and bundling, and G-actin binding (231). Thus, class-4 coronins do not play a role in these activities via their gelsolin domains (29). However, villin contains three phospholipid-binding domains, two preceding the second gelsolin domain and one overlapping with the VHP domain. These phospholipid-binding domains are conserved in class-4 coronins and are most probably responsible for their association with internal membranes like Golgi-structures and ER-membranes (29).

To reveal the evolution of the coronin family and to determine the coronin repertoire of the last common ancestor of the eukaryotes, we plotted the coronin inventory of several representative species, whose genome sequences are available and whose coronin inventories are therefore complete, on the most widely agreed tree of the eukaryotes (Figure 2.2-7). However, especially the grouping of taxa that emerged close to the origin of the eukaryotes remains highly debated. Therefore, alternative branchings are also indicated in the tree. The phylogeny of the supposed supergroup Excavata is the least understood because only a few species of this branch have been completely sequenced so far. While the grouping of the Heterolobosea, Trichomonada, and Euglenozoa into the Excavata is found in most analyses, the grouping of the Diplomonadida as separate phylum or as part of the Excavata is still debated (arrow 1 (157)). Also, some analyses group the red algae of the Rhodophyta branch to the Viridiplantae (159,163,232) and others support their independence (arrow 2; (160,164)). According to most of the recent phylogenetic analyses, the Alveolata, Rhizaria, and Stramenopiles form the superfamily SAR (158,159). The placement of the Haptophyceae and Cryptophyta to the SAR is still highly debated. Although several analyses are in favour to this grouping (arrow 3; (160–162)) most analyses are in contrast (158,159,163–165). Short coronins containing the N-terminal coronin domain and the C-terminal oligomerization domain have been found in all branches except Diplomonadida, Haptophyceae, and Viridiplantae/Rhodophyta. The phylogenetic grouping of the species based on the phylogenetic tree of the coronin domains showed that the coronins with different domain compositions (containing dUTPase domains, ARP2/3 binding domains, no coiled-coil regions) are species-specific developments based on domain loss and gain events while the corresponding species correctly group together inside the respective branches. Class-3 coronins are also found in all major eukaryotic superkingdoms that contain coronins. We did not identify any species that contains exclusively a class-3 coronin suggesting that encoding a class-3 coronin is a plus for the species but not a necessity. Class-4 coronins were found in two of the four coronin-containing superkingdoms, the Excavata and the Opisthokonts. Several major sub-branches of the Opisthokonts contain class-4 coronins, the Amoebozoa, the Fungi, and the Fungi/Metazoa incertae sedis branch. However, the evolution of the class-4 coronins rather seems to be determined by gene-loss events. This distribution of the coronin classes demonstrates that the last common ancestor of the eukaryotes must have contained a short coronin as well as a tandem coronin (class-3), and most probably even a class-4 coronin. In the coronin-family tree (Figure 2.2-1) the C-terminal coronin-domains of the class-3 coronins group closer to the short coronins than the N-terminal coronin-domains.

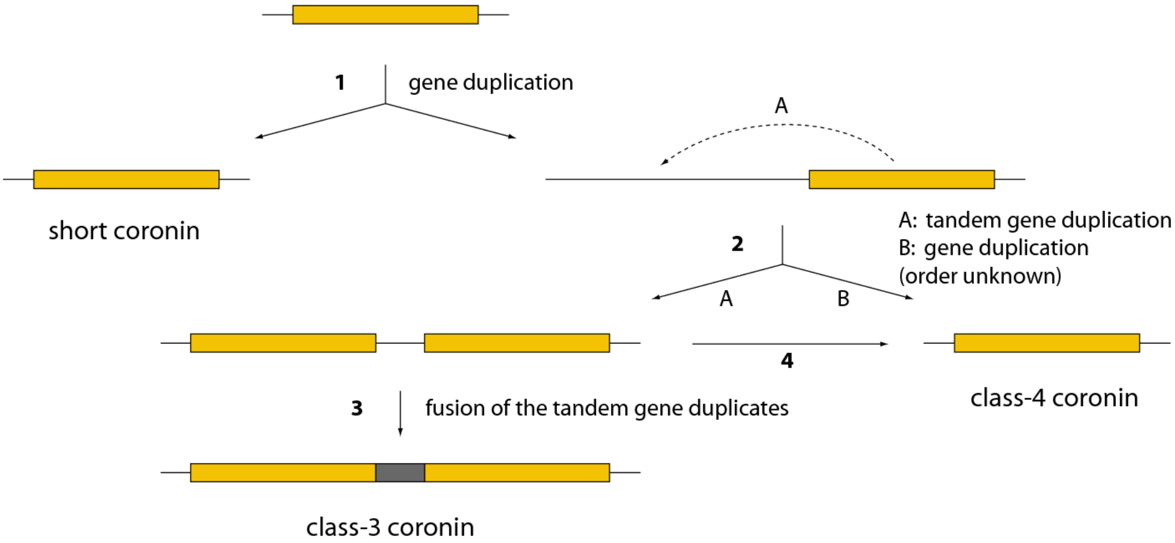


Figure 2.2-8: ***Evolution of the coronin classes.*** *The cartoon shows the different gene duplication and fusion events that led to the formation of the short coronins, the class-3 coronins, and the class-4 coronins.*

This suggests a three-step invention of the class-3 coronin (Figure 2.2-8): First a gene duplication of the short coronin happened (**1**). The new copy was subsequently copied twice but the order of these events could not be determined (**2**). One copy has been distributed in a different genomic region resulting in the class-4 coronin after fusion to a copy of the villin gene (**2B**). The other copy resulted in a tandem gene duplicate in which the new copy was placed at the 5' site of the original gene (**2A**). The tandem gene duplicate subsequently fused to build the class-3 coronin prototype (**3**). It could also be possible that the coronin domain copy, which led to the class-4 coronin, would have been produced as a copy of the 3' coronin of the then already existing tandem gene duplicate (**4**).

At the origin of the Metazoa and Choanoflagellida branches another gene duplication event led to two distinct classes, class-1 coronins and class-2 coronins (Figure 2.2-7). The further evolution of the short coronins in the invertebrate branches is determined by species-specific gene-loss and gene-duplication events (Figure 2.2-2). This view is, however, based on the species whose genomes are available today and might change as soon as sequencing of more related species reveals subtypes of the class-1 and class-2 coronins in major invertebrate branches. At the origin of the vertebrates the two well-known whole-genome duplications (2R, (64)) resulted in several subtypes of both the class-1 and class-2 coronins. The subsequent third whole genome duplication in the fish-lineage (225,226) led to even more gene duplicates. Subsequent to this boost of coronin homologs at the onset of the vertebrates branch-specific gene deletions happened, like the loss of the class-1B variants in fishes and the class-1A loss in birds (Figure 2.2-2).

The short coiled-coil region including the trimerization motif R-[VILM]-X-X-[VIL]-E is an accomplishment of the most ancient short coronin because it is found in coronins of all branches of the eukaryotic tree. It has been retained without major mutations for a long evolutionary time. This is exemplified by the fact that changes, which might lead to other oligomerization states, are species-specific or have been introduced in very recently separated branches.

2.2.5 Conclusions

The phylogenetic tree based on the coronin domains of 723 homologs from 358 species allowed grouping the coronin proteins into four classes: Class-1 (Type I) and class-2 (Type II) comprise short coronins and resulted from a gene duplication of a short coronin at the onset of the halozoans. Short coronins are characterized by an N-terminal coronin domain followed by a unique domain and a C-terminal short coiled-coil region. The coiled-coil domain of almost all short coronins contains a trimerization motif that must therefore have already existed in the last common ancestor of the eukaryotes. Class-3 (Type III) coronins comprise coronins with two coronin-domains arranged in tandem and have been found in species of all eukaryotic kingdoms that contain coronins. Class-4 (Type IV) coronins encode fusions of the coronin domain to villin and have been identified in Excavata and Opisthokonts although most of these species subsequently lost the class-4 homolog. Hence, the last common ancestor of the eukaryotes must have contained a short coronin and a class-3 coronin, and most probably a class-4 coronin.

2.2.6 Methods

Identification and annotation of the coronin family proteins

The coronin genes have been identified by TBLASTN searches against the sequenced eukaryotic genomes, which have been obtained via lists available from the diArk database (210,233). All hits were manually analyzed at the genomic DNA level. Datasets of predicted proteins produced by the sequencing consortia often miss homologs, and predicted proteins contain mispredicted exons and introns in many cases, necessitating manual assembly and annotation. The correct coding sequences were identified with the help of the multiple sequence alignments of all coronin proteins. As the amount of protein sequences increased (especially the number of sequences in taxa with few representatives), many of the initially predicted sequences were reanalysed to correctly identify all exon borders. Where possible, EST data available from the NCBI EST database has been analyzed to help in the annotation process. In addition, coronin homologs from cDNA projects or single-gene analyses have been obtained by TBLASTN searches against the NCBI nr database (60). Gene structures have been reconstructed using WebScipio (13) as far as genomic sequence data was available. All sequence related data (names, corresponding species, GenBank ID's, alternative names, corresponding publications, domain predictions, gene structure reconstructions, and sequences) and references to genome sequencing centres are available through CyMoBase (35,181).

Generating the multiple sequence alignment

The multiple sequence alignment of the coronin family has been built and extended during the process of annotating and assembling new sequences. The initial alignment has been generated from the first about 50 non-validated sequences obtained from NCBI using the ClustalW software with standard settings (62). During the following correction of the sequences (removing wrongly annotated sequences and filling gaps) the alignment has been adjusted manually. Subsequently, every newly predicted sequence has been preliminary aligned to its supposed closest relative using ClustalW, the aligned sequence added to the multiple sequence alignment of the coronins, and the coronin alignment adjusted manually during the subsequent sequence validation process. We have also retained the integrity of the primary sequence within the secondary structural elements that have been determined from the crystal structure (e.g. sequence gaps have only been introduced in known loop regions). Still, many gaps in sequences derived from low-coverage genomes remained. In those cases, the integrity of the exons surrounding the gaps has been maintained (gaps in the genomic sequence are reflected as gaps in the multiple sequence alignment). The unique and coiled-coil regions are completely divergent in sequence and length and were therefore aligned manually. The domain compositions of the short coronin, the class-3, and the class-4 coronins are different and regions outside the N-terminal coronin domain were only aligned within these groups. The C-terminal coronin domains of the class-3 coronins were separately included in the multiple sequence alignment of the coronins, in addition to being aligned as part of their class.

Building trees

For calculating phylogenetic trees only full-length and partial sequences were included in the alignment. The phylogenetic trees were generated based on the conserved coronin domains (corresponding to amino acids 1-386 of *Hs*Coro1A) using two different methods: 1. Maximum likelihood (ML) using the LG model with estimated proportion of invariable sites and bootstrapping (1,000 replicates) using RAxML (67). 2. Posterior probabilities were generated using MrBayes v3.1.2 (72) with the MPI option (186). Two independent runs with 15,000,000 generations, four chains, and a random starting tree were computed using the mixed amino-acid option. From the 32,000th generation MrBayes used the Wag model (187). Using ProtTest (70), the LG model (184), which is, however, not implemented in MrBayes, was determined to provide a slightly better fit to the data than the Wag model. Trees were sampled every 1,000th generation and the first 25% of the trees were discarded as “burn-in” before generating a consensus tree.

Domain and motif prediction

Protein domains were predicted using the SMART (234) and Pfam (34) web server. The leucine zipper motifs have been identified using the Prosite database (235). The CA domains have been identified by visual inspection of the manual sequence alignment of the coronins and motif comparisons with CA domains of WASP family proteins available at CyMoBase (unpublished data, (35)). Graphical representations of the sequence patterns have been generated with WebLogo (58).

2.2.7 Acknowledgements

This work has been funded by grants KO 2251/3-1 and KO 2251/3-2 of the Deutsche Forschungsgemeinschaft.

2.2.8 Authors' contributions

CE and MK assembled coronin sequences, performed data analysis and wrote the manuscript. BH performed the phylogenetic analysis. All authors read and approved the final manuscript.

2.2.9 Additional files

2.2.9.1 Additional file 1 – Sequence alignment of the coronins

The file contains the alignment of the full-length sequences of the coronins in fasta-format. The data can also be downloaded from CyMoBase (35).

The file can be found in the corresponding publication.

2.2.9.2 Additional file 2 – MrBayes tree of the coronin family

This file contains the phylogenetic tree calculated with MrBayes including posterior probability values that has been the basis for Figure 2.2-1. Here, the tree is plotted in an extended way so that every coronin can be found and compared easily.

The file can be found in the corresponding publication.

2.2.9.3 Additional file 3 – RAxML tree of the coronin family

This file contains the phylogenetic tree calculated with RAxML including bootstrap values. The tree is plotted in an extended way so that every coronin can be found and compared easily.

The file can be found in the corresponding publication.

2.2.9.4 Additional file 4 – Coronin repertoire of all eukaryotes analyzed

Complete table of the coronin inventories of 358 eukaryotes.

The file can be found in the corresponding publication.

2.2.9.5 Additional file 5 – Conserved residues in the coronin domain

This figure contains the sequence conservation of the entire coronin domain including all mutagenesis experiments as described in Cai *et al.* (223) and Gandhi *et al.* (224).

The file can be found in the corresponding publication.

2.3 diArk 2.0 provides detailed analyses of the ever increasing eukaryotic genome sequencing data

Björn Hammesfahr[1*], Florian Odronitz[1*], Marcel Hellkamp[1] and Martin Kollmar[1§]

[1] Abteilung NMR basierte Strukturbiologie, Max-Planck-Institut für Biophysikalische Chemie, Am Fassberg 11, D-37077 Göttingen, Germany

[*] These authors contributed equally to the work.
[§] Corresponding author

BMC Research Notes Highly accessed
Published: 9 September 2011
BMC Research Notes 2011 4:338 doi:10.1186/1756-0500-4-338 This article is available from http://www.biomedcentral.com/1756-0500/4/338

2.3.1 Abstract

Background

Nowadays, the sequencing of even the largest mammalian genomes has become a question of days with current next-generation sequencing methods. It comes as no surprise that dozens of genome assemblies are released per months now. Since the number of next-generation sequencing machines increases worldwide and new major sequencing plans are announced, a further increase in the speed of releasing genome assemblies is expected. Thus it becomes increasingly important to get an overview as well as detailed information about available sequenced genomes. The different sequencing and assembly methods have specific characteristics that need to be known to evaluate the various genome assemblies before performing subsequent analyses.

Results

diArk has been developed to provide fast and easy access to all sequenced eukaryotic genomes worldwide. Currently, diArk 2.0 contains information about more than 880 species and more than 2350 genome assembly files. Many meta-data like sequencing and read-assembly methods, sequencing coverage, GC-content, extended lists of alternatively used scientific names and common species names, and various kinds of statistics are provided. To intuitively approach the data the web interface makes extensive usage of modern web techniques. A number of search modules and result views facilitate finding and judging the data of interest. Subscribing to the RSS feed is the easiest way to stay up-to-date with the latest genome data.

Conclusions

diArk 2.0 is the most up-to-date database of sequenced eukaryotic genomes compared to databases like GOLD, NCBI Genome, NHGRI, and ISC. It is different in that only those projects are stored for which genome assembly data or considerable amounts of cDNA data are available. Projects in planning stage or in the process of being sequenced are not included. The user can easily search through the provided data and directly access the genome assembly files of the sequenced genome of interest. diArk 2.0 is available at http://www.diark.org.

2.3.2 Background

The International Human Genome Project needed almost 13 years for the sequencing of the first human genome (10). While Celera, using the same Sanger technique, already accelerated human genome sequencing to three years by applying a whole genome shotgun instead of the primer based approach (9), the sequencing of even the largest mammalian genomes has become only a matter of days with current next-generation sequencing methods (236). The bottleneck for providing the analysis of a eukaryotic genome is thus not the sequencing process anymore (237). The most time consuming part is the assembly and even more the annotation of genes, RNA, and other genetic features (238). Nevertheless, while only a few genome assemblies have been made public per year at the beginning of the century, dozens of genome assemblies are released per month today. A further increase in the speed of releasing genome assemblies may be expected because of the increasing number of next-generation sequencing machines worldwide (239), together with the announcement of major sequencing plans (see for example the 1000 human genomes project (240), the 1001 arabidopsis genomes project (241), the 1,000 Plant & Animal reference genomes project (242), and the 10,000 vertebrates genomes project (243)).

There are many steps to produce a complete and gap-less genome sequence of an organism. First draft versions often contain sets of so-called contigs that have been built from the assembly of whole genome shotgun reads. The genome coverage is the most important factor determining contig length. In the following steps during the assembly process the contigs are organised into supercontigs and finally into chromosomes. In the finishing process, gaps are filled by direct sequencing of the corresponding regions. However, the publication of the genome sequence of an organism does not correlate with the status of the assembly process. Some genome assemblies have been published although they are very fragmented and represent rather early draft assemblies (e.g. (244–247)), while finishing and gap-closing have already been done for other genomes still waiting to be published. It is obvious that analyses based on genes, genomic regions, or proteins need high coverage genome sequences and assemblies to very long contigs or even supercontigs. This is especially true for the analysis of genes of higher eukaryotes that are often spread over hundred thousands of base pairs.

How can a researcher find out which organisms have already been sequenced, how good the quality of the latest assembly is, and what the differences between the sometimes many different assemblies of the same genome are? To provide access to genome data, five major databases have been developed: GOLD (248), NCBI Genome Project (will soon be reorganized into NCBI BioProject) (249), National Human Genome Research Institute (NHGRI) (250), International Sequencing Consortium (251), and diArk (210). The GOLD database monitors finished and on-going genome and metagenome sequencing projects of all branches of the tree of life (248). The largest part of the database is related to prokaryotes for which most of the about 130 metadata fields have been designed. GOLD's strength therefore is the listing of the prokaryotes, while it is outdated for eukaryotes. For example, GOLD announces 156 eukaryotes as published (although several of these are listed as "unpublished" in the table, status: March 10, 2011) while genome assemblies of 358 eukaryotes have been published according to diArk (status: March 10, 2011). The NCBI Genome Project pages list all sequencing centres participating in a certain sequencing project and provides many links to other species resources (species databases, BLAST and genome browser pages, publications, etc.). However, the list of these projects is far from being up-to-date. Here, 431 eukaryotes are available and listed as complete or draft assembly, while diArk provides assemblies for 613 species. The NHGRI hosts a list of approved sequencing targets (almost exclusively eukaryotic) with limited additional information. However, most eukaryotic projects are not listed, and the project status (not started, in process, complete) is often not up-to-date. For example, the sequencing of *Geomyces destructans* is still listed as "not started" although a very good draft assembly is already available. The International Sequencing Consortium hosts a list of comparable information to the NHGRI.

diArk 2.0 is the most up-to-date database for eukaryotic sequencing projects, providing in the latest version many meta-data like sequencing and read-assembly methods, sequencing coverage, GC-content, extended lists of alternatively used scientific names and common species names, and various kinds of statistics. diArk only lists those projects, for which genome assemblies or considerable amounts of cDNA data are available. diArk does not list projects that are planned, and does not track the various stages of the genome sequencing process (species targeted, awaiting DNA, DNA library prepared, etc.) as it is done by GOLD (248). Due to the next-generation sequencing methods sequencing has become so fast and cheap that the time frame between planning and finishing sequencing projects is in the order of weeks and not years anymore. Although independent groups have not sequenced too many identical species yet, sequencing has started to become competitive so that project plans are often not announced anymore and finished sequences claimed by press releases (252). The virtue of the sequencing projects is the data, and thus the intention of diArk is to provide easy and fast access to where and which eukaryotic data may be obtained.

2.3.3 Methods

The technologies

The system is running on Linux. The database management system is PostgreSQL (73) supported by pgpool-II (253). The web application framework is Ruby on Rails (79), which is based on the object orientated programming language Ruby (79). In order to present the user with a feature rich interface while minimizing the amount of transferred data the site makes extensive use of modern Web 2.0 techniques like Ajax (Asynchronous JavaScript and XML) using Prototype (254), and Lightwindow (255). Graphs are drawn using the graphical toolkit Protovis (256,257), the statistical programming language R (258), and SVG (259). Ruby together with BioRuby (260) is also used for scripts that automatically retrieve data via the NCBI-API, reconstruct the phylogenetic tree of diArk's species, and analyse genome assembly files. All technologies used are freely available and open source.

The database

diArk has been developed with a custom database schema due to the unique requirements of the system (210). Initially, three interconnected tables had been at the centre of the database: species, projects, and publications. This basic concept has significantly been extended by more than doubling the number of database tables and by increasing the number of fields in existing tables (Additional file 2.3.10.1). Most importantly, a table for genome file data has been added to which several further tables are connected representing sequencing and assembly methods (Figure 2.3-1, Additional file 2.3.10.1).

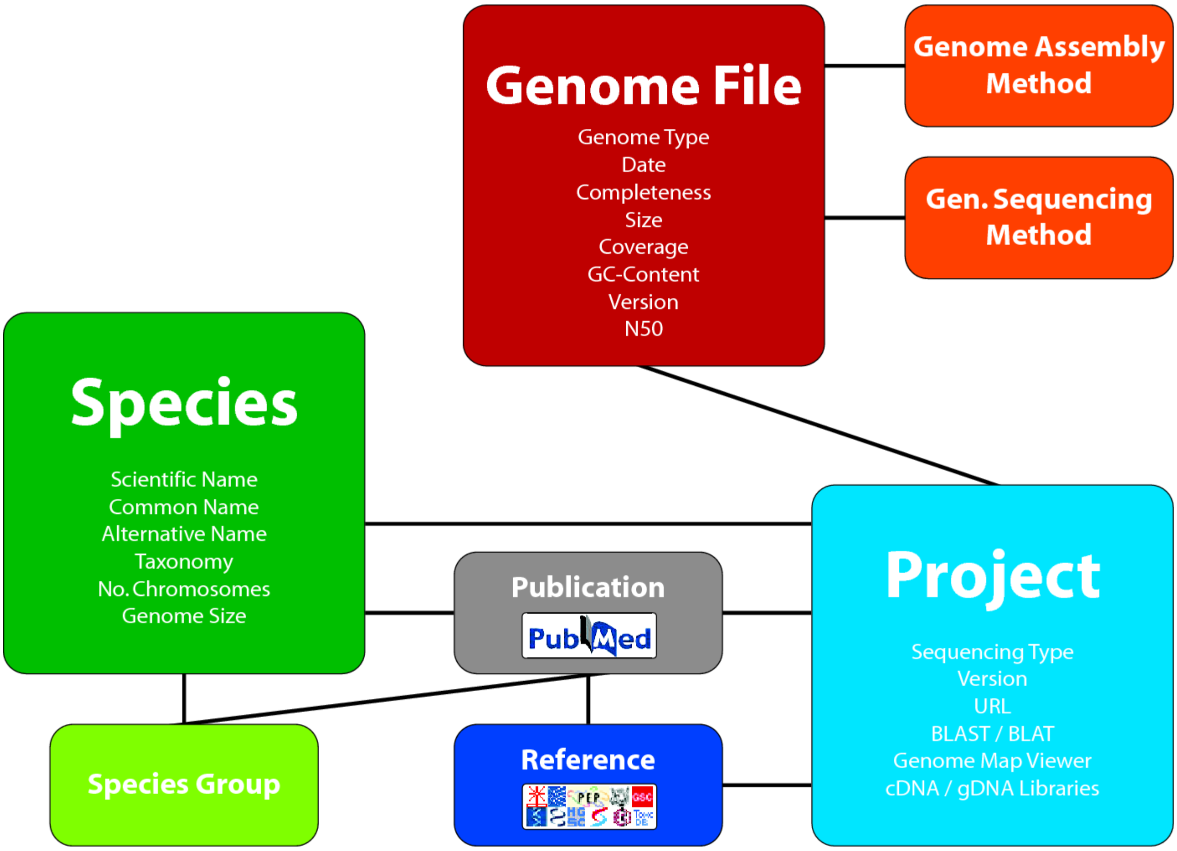


Figure 2.3-1: ***Schematic organisation of the database.*** *The diagram shows the major tables of the database and their connections. Some of the content fields of the three main tables Species, Project, and Genome File are listed. Details to publications are obtained from NCBI via their API. The References table contains the major sequencing centres and species project web pages.*

The genome file table contains information about genome assemblies. Genome assembly files are retrieved from sequencing centres, dedicated species/taxa sequencing pages, or from the NCBI database. While some information is directly calculated from the assembly files, other information is manually added to the genome file table. Every assembly file gets a genome type identifier based on the fasta-entries. The most important genome types are Chromosome, Uchromosome (these files contain contigs/supercontigs, which could not be mapped to any (unknown chromosome) or anchored (random chromosome) to a certain chromosome), Supercontigs, Contigs, Ureads (unplaced reads), Apicoplast, Chloroplast, Kinetoplast, and Mito (mitochondrial DNA). In addition, there are some special extensions to the file types, for example “assembly1", "assembly2", etc.. These extensions indicate that different assemblies for the same genome are available. For example, if assemblies were produced from different sequencing data like in the case of *Drosophila pseudoobscura* (assembly1: (261); assembly2: unpublished assembly of The Institute for Genomic Research) or if the same reads were assembled using different methods/software like in the two *Bos taurus* genome assemblies (assembly1: (262); assembly2: (263)).

If possible, the version of the assembly as well as the release date of the data is provided. In general, the versions and release dates are entered manually as given by the sequencing centres. Otherwise the dates are used at which the files were saved in the ftp-directories.

For NCBI-assembly data, we store the dates at which the data has been submitted to NCBI. Please note that the version numbers do not correlate among sequencing centres and NCBI. Also, we rank the completeness of the genome assemblies as a rough estimate of the quality of the data. If provided by the sequencing centres, the genome coverage of the assembled sequence data is given. For some assemblies, comments are written that provide further background information about differences to earlier assemblies and problems during the assembly process, for example.

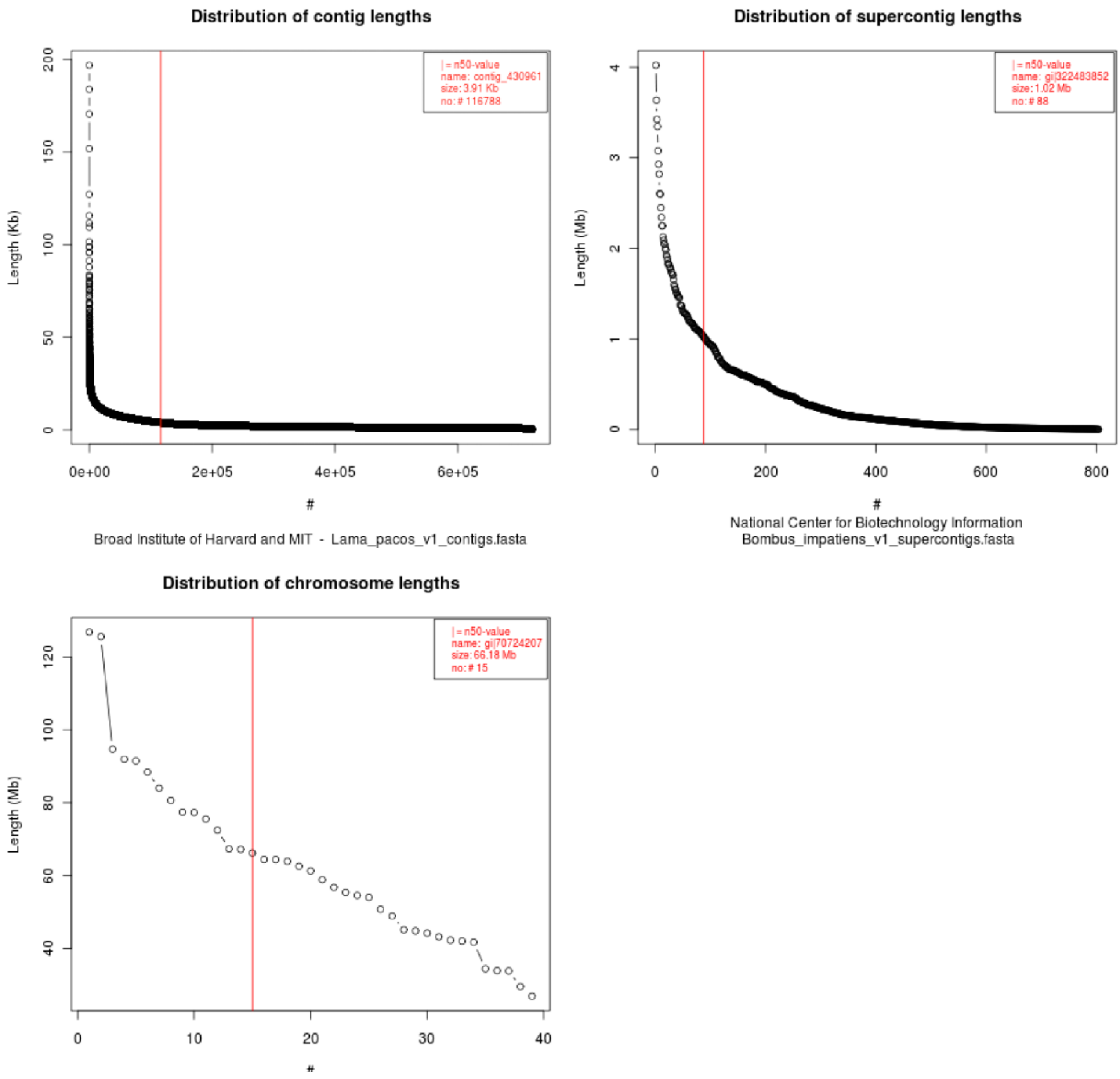


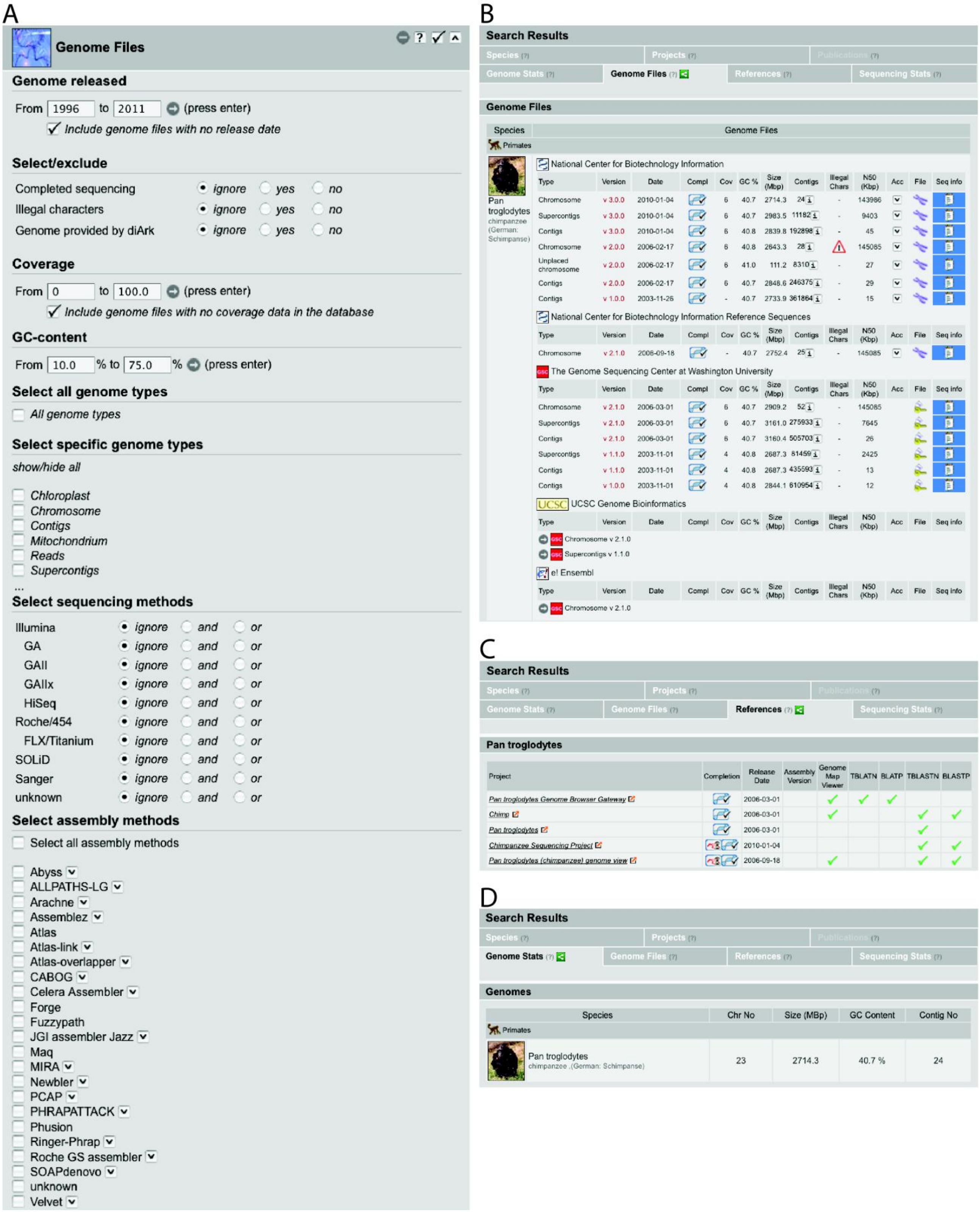
Figure 2.3-2: ***Contig distribution for three sample genome assemblies.*** *A) Example of a low-coverage mammalian genome. B) Example of a high-coverage insect genome. C) Example of a chromosome assembly. All chromosomes are plotted as separate entries.*

In addition to this manually collected information, the GC content, the size in Giga-base-pairs, the number of fasta-entries, the occurrence of illegal characters in the sequences (not being g/G, a/A, t/T, c/C, or n/N), and the N50 of the assemblies are calculated from the fasta files. The N50 value is a measure of contig length and is calculated by adding up contig lengths starting with the longest contig. The length of that contig, which leads to at least half of the assembly, is the N50 value. The longer the contigs are the longer is the contig that overcomes the half-genome barrier. All contig lengths are counted and plotted in decreasing length together with the N50 value (Figure 2.3-2). These graphs provide additional information to the user to judge the quality of the assembly. Accession numbers are only stored from NCBI data.

For every genome file the sequencing methods and the assembly software were collected, if available. The next-generation sequencing methods strongly differ in their usefulness concerning de-novo assemblies, and therefore this information together with the sequencing coverage and the library types used for sequencing is absolutely essential to judge the quality of the data.

The web interface

The web interface always represents the current state of the database, and all tables and graphs are calculated on-the-fly depending on users' requests. The database is searched using any of the six *search modules*, or a combination of them. We have added a new module, called "Genome Files", for searching the data content of the genome file table and associated tables (Figure 2.3-3A).



Genome Files

Genome released

From 1996 to 2011 (press enter)
☑ *Include genome files with no release date*

Select/exclude

Completed sequencing	● *ignore*	○ *yes*	○ *no*
Illegal characters	● *ignore*	○ *yes*	○ *no*
Genome provided by diArk	● *ignore*	○ *yes*	○ *no*

Coverage

From 0 to 100.0 (press enter)
☑ *Include genome files with no coverage data in the database*

GC-content

From 10.0 % to 75.0 % (press enter)

Select all genome types

☐ *All genome types*

Select specific genome types

show/hide all

☐ *Chloroplast*
☐ *Chromosome*
☐ *Contigs*
☐ *Mitochondrium*
☐ *Reads*
☐ *Supercontigs*
...

Select sequencing methods

Illumina	● *ignore*	○ *and*	○ *or*
GA	● *ignore*	○ *and*	○ *or*
GAII	● *ignore*	○ *and*	○ *or*
GAIIx	● *ignore*	○ *and*	○ *or*
HiSeq	● *ignore*	○ *and*	○ *or*
Roche/454	● *ignore*	○ *and*	○ *or*
FLX/Titanium	● *ignore*	○ *and*	○ *or*
SOLiD	● *ignore*	○ *and*	○ *or*
Sanger	● *ignore*	○ *and*	○ *or*
unknown	● *ignore*	○ *and*	○ *or*

Select assembly methods

☐ Select all assembly methods

☐ Abyss
☐ ALLPATHS-LG
☐ Arachne
☐ Assemblez
☐ Atlas
☐ Atlas-link
☐ Atlas-overlapper
☐ CABOG
☐ Celera Assembler
☐ Forge
☐ Fuzzypath
☐ JGI assembler Jazz
☐ Maq
☐ MIRA
☐ Newbler
☐ PCAP
☐ PHRAPATTACK
☐ Phusion
☐ Ringer-Phrap
☐ Roche GS assembler
☐ SOAPdenovo
☐ unknown
☐ Velvet

Search Results

Species (?) | Projects (?) | Publications (?)
Genome Stats (?) | **Genome Files** (?) | References (?) | Sequencing Stats (?)

Genome Files

Species: Primates — Pan troglodytes chimpanzee (German: Schimpanse)

National Center for Biotechnology Information

Type	Version	Date	Compl	Cov	GC %	Size (Mbp)	Contigs	Illegal Chars	N50 (Kbp)	Acc	File	Seq info
Chromosome	v 3.0.0	2010-01-04		6	40.7	2714.3	24	-	143986			
Supercontigs	v 3.0.0	2010-01-04		6	40.7	2983.5	11182	-	9403			
Contigs	v 3.0.0	2010-01-04		6	40.8	2839.8	192898	-	45			
Chromosome	v 2.0.0	2006-02-17		6	40.8	2643.3	28	⚠	145085			
Unplaced chromosome	v 2.0.0	2006-02-17		6	41.0	111.2	8310	-	27			
Contigs	v 2.0.0	2006-02-17		6	40.7	2848.6	246375	-	29			
Contigs	v 1.0.0	2003-11-26		-	40.7	2733.9	361864	-	15			

National Center for Biotechnology Information Reference Sequences

Type	Version	Date	Compl	Cov	GC %	Size (Mbp)	Contigs	Illegal Chars	N50 (Kbp)	Acc	File	Seq info
Chromosome	v 2.1.0	2006-09-18		-	40.7	2752.4	25	-	145085			

The Genome Sequencing Center at Washington University

Type	Version	Date	Compl	Cov	GC %	Size (Mbp)	Contigs	Illegal Chars	N50 (Kbp)	Acc	File	Seq info
Chromosome	v 2.1.0	2006-03-01		6	40.7	2909.2	52	-	145085			
Supercontigs	v 2.1.0	2006-03-01		6	40.7	3161.0	275933	-	7645			
Contigs	v 2.1.0	2006-03-01		6	40.7	3160.4	505703	-	26			
Supercontigs	v 1.1.0	2003-11-01		4	40.8	2687.3	81459	-	2425			
Contigs	v 1.1.0	2003-11-01		4	40.8	2687.3	435593	-	13			
Contigs	v 1.0.0	2003-11-01		4	40.8	2844.1	610954	-	12			

UCSC Genome Bioinformatics

Type	Version	Date	Compl	Cov	GC %	Size (Mbp)	Contigs	Illegal Chars	N50 (Kbp)	Acc	File	Seq info

GSC Chromosome v 2.1.0
GSC Supercontigs v 1.1.0

e! Ensembl

Type	Version	Date	Compl	Cov	GC %	Size (Mbp)	Contigs	Illegal Chars	N50 (Kbp)	Acc	File	Seq info

GSC Chromosome v 2.1.0

Search Results

Species (?) | Projects (?) | Publications (?)
Genome Stats (?) | Genome Files (?) | **References** (?) | Sequencing Stats (?)

Pan troglodytes

Project	Completion	Release Date	Assembly Version	Genome Map Viewer	TBLATN	BLATP	TBLASTN	BLASTP
Pan troglodytes Genome Browser Gateway		2006-03-01		✓	✓	✓		
Chimp		2006-03-01		✓			✓	✓
Pan troglodytes		2006-03-01					✓	
Chimpanzee Sequencing Project		2010-01-04					✓	✓
Pan troglodytes (chimpanzee) genome view		2006-09-18		✓			✓	✓

Search Results

Species (?) | Projects (?) | Publications (?)
Genome Stats (?) | Genome Files (?) | References (?) | Sequencing Stats (?)

Genomes

Species	Chr No	Size (MBp)	GC Content	Contig No
Primates				
Pan troglodytes chimpanzee ,(German: Schimpanse)	23	2714.3	40.7 %	24

*Figure 2.3-3: **Screenshots of diArks “Genome Files” search module and several result views.** A) The new “Genome Files” search module of diArk allows a detailed search for species that were sequenced with a specific sequencing method, for certain assembly methods, for specific genome types, for the completeness of the assembly, for illegal characters (not a/A, t/T, g/G, c/C, n/N), and for genomes provided by diArk. Furthermore, the data can be filtered by the GC-content, by the sequence coverage, and the release date of the genome assemblies. B) The “Genome Files” result view provides an overview about the different genome assemblies generated by the sequencing centres. Clicking on the symbols provides further details and the possibility to download the genome file. C) The “References” result view provides an overview about some data analysis options the species project pages offer, like BLAST pages or access to genome browsers. D) The “Genome Stats” result view gives a species based overview about several genome statistics, like the chromosome numbers and the GC-contents, with the species ordered according to their taxonomy so that closely related organisms can be compared.*

The results of the search can be browsed in *result views*. Previously, three result views had been offered, the “Species”, the “Publications” and the “Projects” result view. The new “Genome Stats” result view provides a fast overview of important genome characteristics in direct comparison of evolutionarily related species and includes chromosome numbers (if known), genome sizes (as calculated from the assembly files, given as number of base pairs included in the chromosome-, supercontigs-, or contigs-file, in descending priority), the GC-contents, and the number of contigs (Figure 2.3-3D). The “Genome Files” result view provides a direct comparison of the data related to the assembly files (Figure 2.3-3B). Here, data as provided from NCBI and the sequencing centres can be downloaded (in accordance with the Bermuda principles and the Ford Lauderdale agreement (264)) and the graphs presenting the size distribution of the contigs/supercontigs/etc. can be viewed (Figure 2.3-2). The “References” result view provides information about tools and material as provided by the species sequencing pages, for example, whether certain species homepages provide BLAST search possibilities or access to genome browsers (Figure 2.3-3C). The “Sequencing Stats” result view provides many graphs presenting various aspects of the data (in total or according to the selection by the user; see also below).

In addition to the modular search, which allows a powerful and very detailed definition of the search, diArk provides a “Fast Search” just offering the main search options: the search for a single species, the selection of model organisms or given taxa, the selection for sequencing type, completed genome sequencing, and retrieval of NBCI genome data. This search should be more suited for beginners.

Stay informed – inform others

To stay up-to-date with newly sequenced genomes without repeatedly accessing diArk we offer an RSS-feed. To easily inform others, diArk offers options that allow the user to send content to facebook-, twitter-, and email-accounts.

2.3.4 Results and Discussion

diArk is the most comprehensive and complete database for eukaryotic sequencing projects. The number of sequenced species and projects has more than doubled since the first version of diArk went online (Figure 2.3-4, (210)). diArk now (March 2011) contains 806 species (415 in 2007; numbers in parenthesis refer to database content in 2007), of which 613 (209) were subject to whole genome sequencing. Genome sequence data is referenced by 1911 (824) species project pages that are organized into 101 (73) sequencing centres.. The number of sequenced species is not as strongly increasing as might have been expected (Figure 2.3-4B).

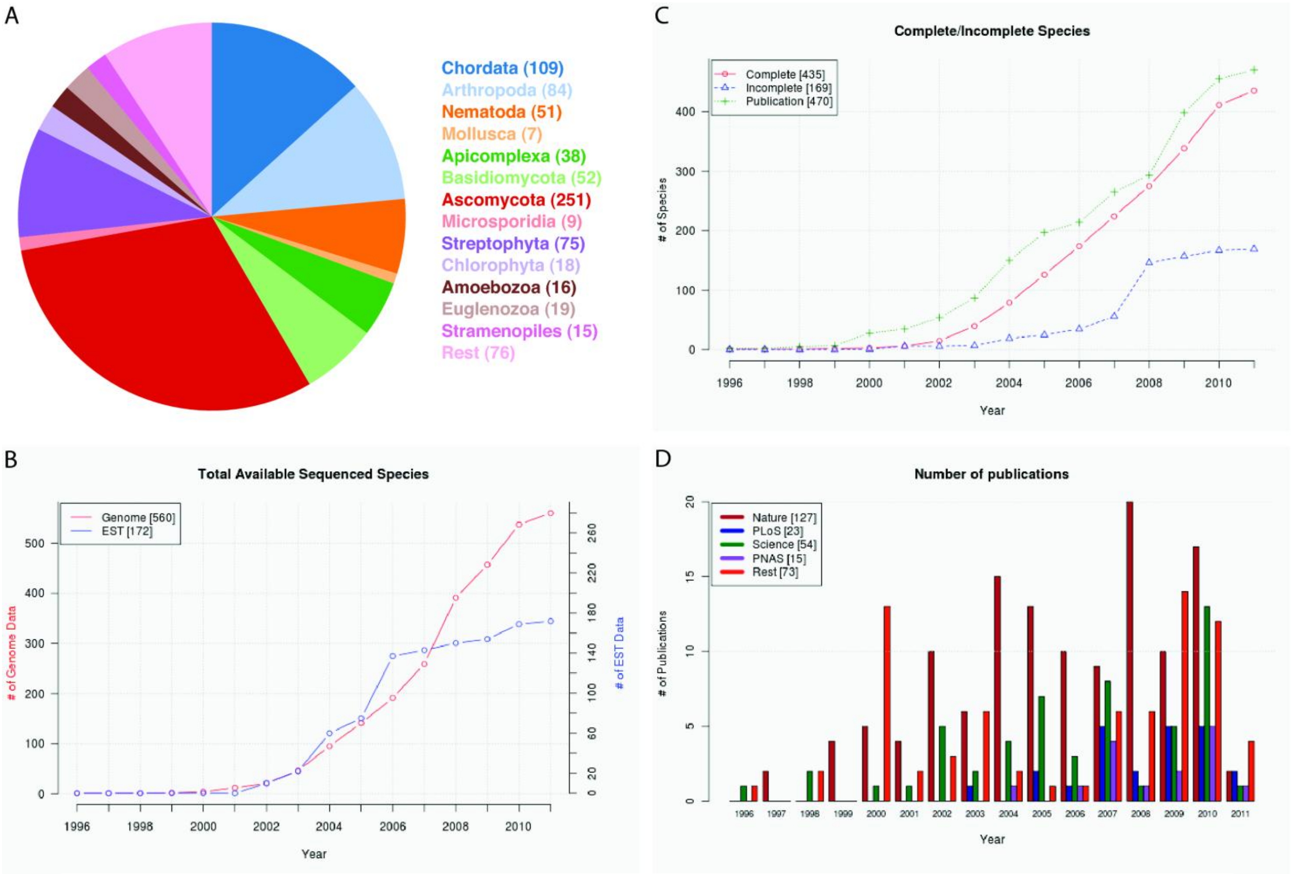


*Figure 2.3-4: **Eukaryotes sequenced worldwide.** A) The pie chart shows the sequenced species sorted by taxa for which genome assemblies have been released. B) The graph shows the increase of total sequenced eukaryotes, genome data as well as EST data, in dependence of the year. Note that the lower numbers in the figures compared to the numbers given in the text are due to the fact that dates, at which genomes had been made available, are not known for every genome assembly. C) The graph shows the sequenced eukaryotes separated according to complete and incomplete (low-coverage genomes) genome assemblies. In addition, publications of genome assemblies are plotted. D) The diagram shows the number of publications of genome assemblies separated to four major publishing groups, the Nature Journals, the PLoS Journals, Science, and the Proceedings of the National Academy of Science (PNAS).*

The discrepancy between the expected sequencing throughput and the only slightly exponential increase of sequenced species is best explained by the increased use of next-generation sequencing machines for other projects then de-novo sequencing of eukaryotes, like for human sequencing in the course of the 1000 Genomes Project (240) and for metagenome projects, which are not covered by diArk. Also, most likely due to next-generation sequencing the number of incomplete genomes (genomes sequenced with very low coverage) does not increase as strongly as before (Figure 2.3-4C). The strong increase between 2007 and 2008 is due to the low coverage sequencing of more than 60 Saccharomyces strains (265). Although some sequenced genomes are awaiting analysis and publication since years, most genome sequences are published shortly after their generation (Figure 2.3-4C). The genomes of most sequenced species are still published in the high-impact journals Science, those of the Nature group, PNAS, and the PLoS journals (Figure 2.3-4D).

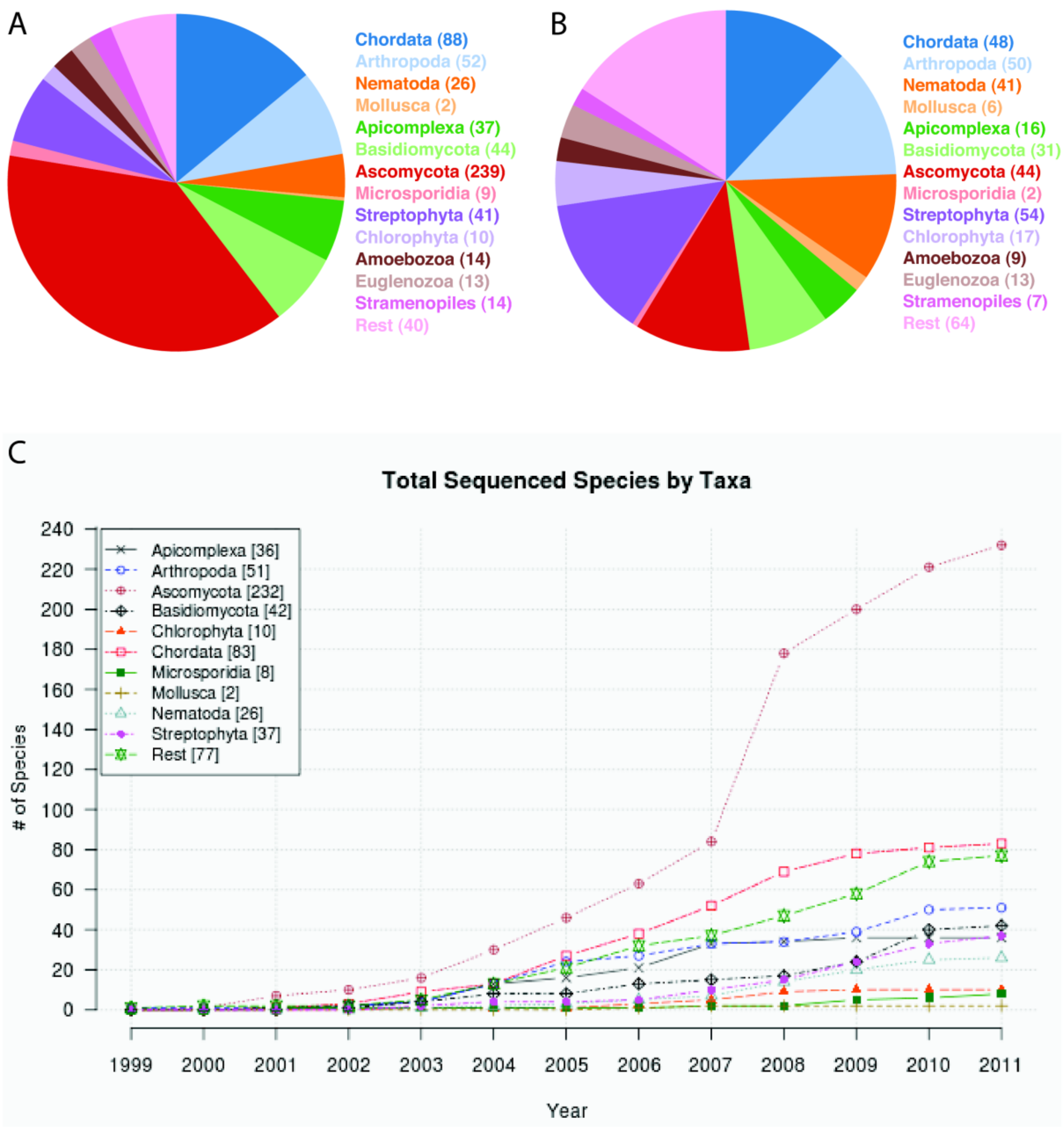


*Figure 2.3-5: **Species sequenced in relation to taxa.** A), B) The pie charts show the number of sequenced species ordered by several major taxa. Graphs were drawn separately for species A) whose genome was sequenced and B) for which transcriptome data is available. C) Species are plotted according to the year in which the first genome assembly has been released. The species are combined to the same taxa as in A) and B).*

Taxonomic distribution

As in 2007, whole genome sequencing is still strongly biased towards sequencing of fungi (especially ascomycotes) and chordates (Figure 2.3-5A). However, in 2007 we pointed out (210) that sequencing of nematodes and plants is far underrepresented, and this has changed dramatically. The number of sequenced nematodes and plants increased fivefold in the last years while the number of the other sequenced species doubled to tripled (Figure 2.3-5C). The taxonomic distribution is still better balanced for transcriptome sequencing (Figure 2.3-5B).

Sequencing methods

Since the first sequencing of a genome using massively parallel DNA sequencing (266) the Sanger method has increasingly been substituted by the high-throughput methods Roche/454, Illumina Solexa, and SOLiD (Figure 2.3-6). These methods pose several restrains to de-novo species sequencing like the need for a far higher sequencing coverage (some species like *Oreochromis niloticus* are sequenced with a coverage of more than 200 using Illumina) and specific assembly software. Both characteristics have been included in diArk.

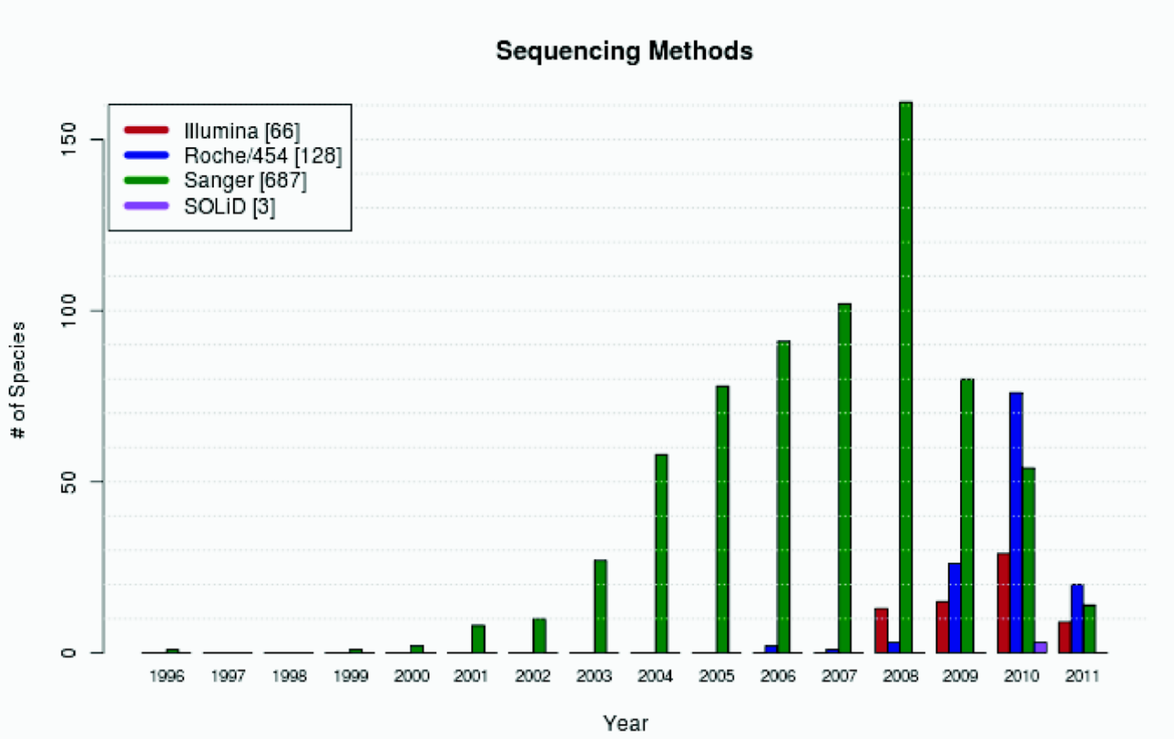


Figure 2.3-6: ***Number of species sequenced by a certain sequencing method per year.*** *The diagram shows the number of species sequenced with different sequencing methods. For species that were sequenced using several methods (e.g. the whole genome library was sequenced with 454 and the BAC library sequenced with Sanger), every method is counted.*

Genome characteristics

Based on the genome assembly files diArk calculates several genome assembly characteristics like the number of contigs, N50 values, GC-content, and genome size. The plot of the genome sizes of completed genome assemblies against their GC-content shows taxa specific distributions (Figure 2.3-7A). Chordates have the largest genomes (and also a wide distribution of genome sizes, Figure 2.3-7B/C) but a narrow distribution of their GC-contents between 37 - 47%. Apicomplexa have the broadest distribution with GC-contents ranging from 20 - 55%, while Chlorophyta have the highest GC-contents (52 - 67%).

diArk in comparison to other databases

Important parameters describing diArk's content in comparison to that of GOLD, NHGRI, NCBI Genome, and ISC are listed in Table 2.3.1. Because diArk, NHGRI, and ISC exclusively contain eukaryotes only those data were compared. Most obviously, the total number of species differs by up to a factor of ten. At diArk, information about 806 species is available (numbers have been obtained on March 10, 2011) while GOLD provides data for 2153 eukaryotes with 1876 species unique. NHGRI lists 187 (total 248), NCBI Genome 986 (total 1090), and ISC 287 (total 360) unique species, respectively. In total, GOLD and NCBI Genome list more species than diArk, but this is mainly due to the different philosophies. GOLD and NCBI Genome include species for which genome projects are planned or which are in very early stages ("DNA received" or "sequencing in progress") of the project while diArk only lists projects for which genome assemblies or considerable amounts of cDNA/EST data are available. In addition, GOLD, NHGRI, NCBI Genome, and ISC list the same species multiple times if for example different sequencing centres sequence different genome libraries (e.g. three entries are available for sequencing *Bos taurus* at GOLD), while diArk combines these data.

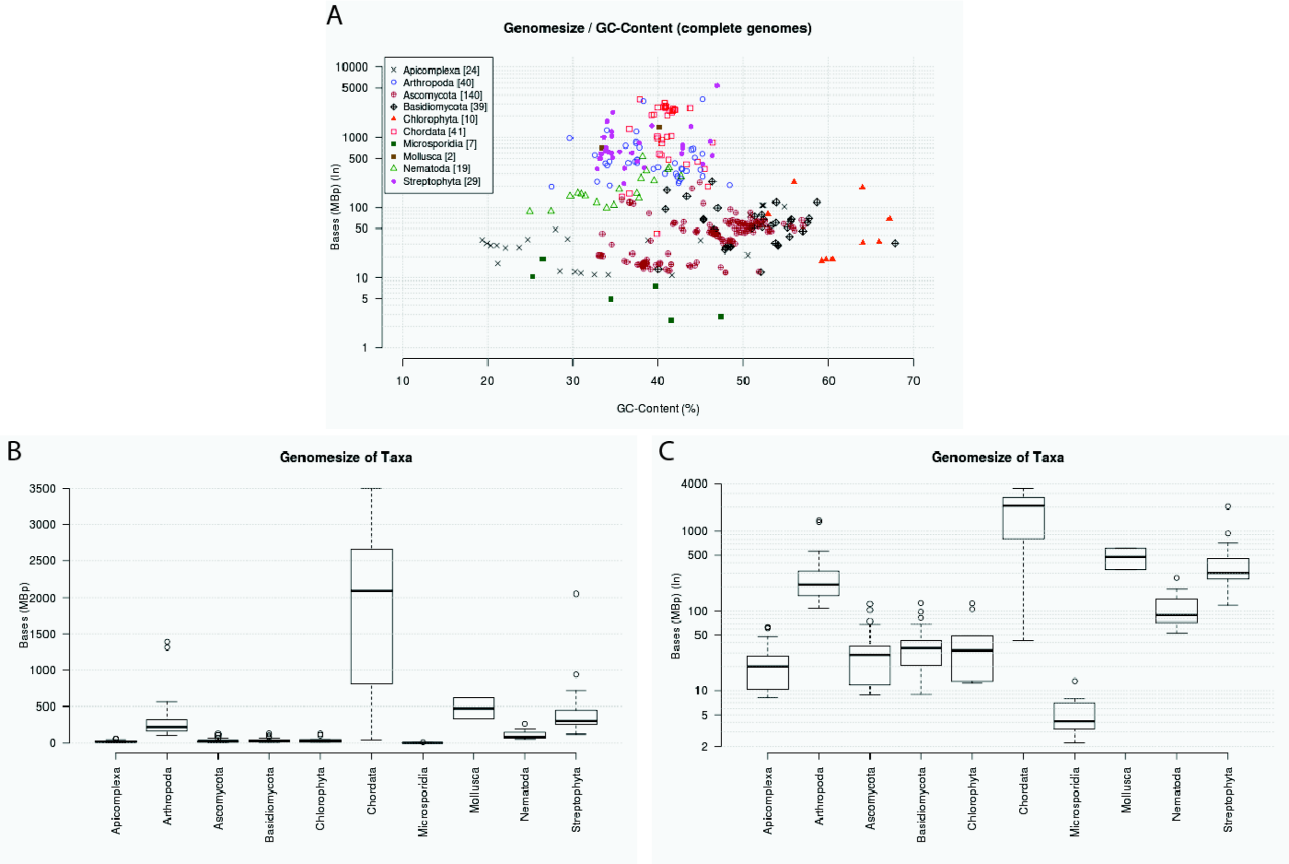


Figure 2.3-7: ***Genome assembly characteristics.*** *A) The graph shows the GC-content and the genome size of completed genome assemblies (thus excluding low-coverage genomes). For better visualisation the genome size is plotted logarithmically. B) The diagram shows the box plot of the genome sizes of some major taxa for which many completed genome assemblies are available. C) Same as B) but the genome sizes are plotted logarithmically to better visualize the sizes of the smaller genomes.*

Different strains of a species (e.g. *Saccharomyces cerevisiae YS2* and *YS4*) are treated separately in all databases. Thus, the up-to-dateness of the databases can only be compared at the level of draft, finished, and published genomes. In diArk, 613 of 806 species are completely sequenced and 358 are published. In contrast, GOLD assigned 358 of the 2153 species as completed and 156 as published genomes. Publications for species are missing in GOLD for example (chosen alphabetically) for the pea aphid *Acyrthosiphon pisum* (267), the giant panda *Ailuropoda melanoleuca* ((268), still marked as "in progress"), the fungus *Ajellomyces capsulatus NAmI WU24* (269), the American malaria mosquito *Anopheles darlingi* ((270), still marked as "in progress"), and the fungus *Ascosphaera apis* (271), while the list of 156 "published genomes" also contains species marked as "unpublished" (e.g. *Arthroderma benhamiae*) and those, for which no information at all is given (e.g. the four *Arabidopsis thaliana* ecotypes Bur-0, C24, Ler-1, and Kro-0). At NCBI Genome, 431 completed and 285 published eukaryotes were found. Because species projects and publications are entered manually into diArk and the other databases, the lower numbers by GOLD and NCBI Genome might mainly result from oversight and lack of manpower by the curators. diArk includes all publications listed in GOLD and NCBI Genome. Furthermore, diArk is unique in providing additional information for most of the sequenced genomes like the method(s) used for sequencing, the method(s) used to create the assembly, and assembly details like the sequencing coverage or the assembly version. For each assembly, the GC-content and the assembly size are computed while NCBI Genome and GOLD provide these data for only a small subset of their species. Based on these data, diArk presents the most comprehensive and complete dataset of sequenced eukaryotic species worldwide.

Table 2.3.1: diArk's content in comparison to other databases

	diArk	GOLD	NHGRI	NCBI Genome	ISC
# species (unique/total)	806	1876/2153	187/248	986/1090	287/360
# mRNA sequencing projects	562	350 (EST) 88 (Transcriptome)	11 (RNA) 1 (cDNA)	-	6 (cDNA) 1 (EST)
# genome sequencing projects	1499	1705	160	1078	-
# genomes marked as "sequenced" 1)	613	358 (completed)	88 (completed)	431	105
# genomes marked as "published" 2)	358	156	-	285	-
taxonomy	full taxonomy	two major taxa	one major taxon	two major taxa	one major taxon
sequencing method	✓	-	-	-	-
assembly method	✓	-	-	-	-
GC-content (# species)	589/613	142/1876	-	-	-
genome size (# species)	589/613	510/1876	-	✓	-
assembly details	✓	-	-	-	-
genome assembly files analysed	2109	-	-	-	-
species common names	✓	✓	✓	-	✓
links to species pages	✓	✓	-	-	-
detailed info about species pages	✓	-	-	-	-
sequencing centre reference	✓	✓	✓	✓	✓
funding agency	-	✓	✓	-	✓
target (survey sequencing, draft, etc.)	-	-	✓	✓	✓
project status	-	✓	✓	✓	✓
database search options	✓	✓	-	limited	limited
database content view options	7 result tabs	1 table	1 table	1 table	1 table
accessibility / speed	fast	slow	fast	fast	fast

(1) In this analysis, all genomes, for which assemblies were announced, are regarded as “sequenced” independently of the various status that the different databases give (draft, completed, published) and independently of the genome coverage.

(2) The numbers of published genomes have been retrieved as follows: **diArk**: 1) Using the Search page, select Projects_Search_Module, select “Sequencing type” Genome, and “Select all references” All Projects; 2) Add Search_Module, select Publications_Search_Module, and select “Select all publications” All Publications. **GOLD**: The number of published genomes is given, separated by kingdoms, in the “Complete Published” list. **NCBI Genome**: The number of published genomes has been derived by counting the links to PubMed.

NHGRI: http://www.genome.gov/10002154 (acquisition of data: 2011-03-10)
NCBI Genome Projects: http://www.ncbi.nlm.nih.gov/genomeprj, http://www.ncbi.nlm.nih.gov/genomes/leuks.cgi (acquisition of data: 2011-03-10)
ISC: http://www.intlgenome.org/viewDatabase.cfm (acquisition of data: data as of 2011-03-10)

2.3.5 Conclusions

Due to the next-generation sequencing methods genome data of eukaryotes is increasing rapidly. Technically, all methods have their advantages and disadvantages, and it is therefore important to know how the genome of interest has been sequenced. Also, different assemblies have been generated for several species using either the same raw data but different assembly methods (262,263,272), or incorporating data from different sources (see for example the latest *Rattus norvegicus* assembly, version 4.1, generated at the Human Genome Sequencing Center at Baylor College of Medicine). diArk stores all genome assemblies that are available worldwide and provides several assembly related metadata: assembly version, assembly release date, completeness of the assembly, GC-content, assembly size, number of contigs, N50-value (including graphical representation of the contig distribution), accession numbers of the contigs, genome assembly files, sequencing method, and assembly method. diArk also provides many statistical analyses of its content based on the selection of the data. Currently, diArk contains data associated to 806 species. For 611 of them, genome assemblies are available, in most cases in different versions and types (contigs, supercontigs, chromosomes, etc.) amounting to 2109 genome assembly files. Of these 611 genome assemblies, 358 have already been published. Compared to other databases diArk 2.0 provides the most recent and comprehensive eukaryotic genome assembly data.

2.3.6 Availability and Requirements

Project name: diArk – a resource for eukaryotic genome research

Project home page: http://www.diark.org/

Operating system: Platform independent

Programming language: Ruby

Other requirements: The current version of diArk was designed for Firefox, but has been tested on all recent versions of Safari, Internet Explorer, and Chrome. It requires cookies and JavaScript enabled.

Web-service: To use the web service via SOAP, the WSDL-file can be obtained at http://www.diark.org/diark_backend/service.wsdl. For using XML-RPC, users can connect to the endpoint URL http://www.diark.org/diark_backend/api.

License: The database schema, the web application and all scripts can be obtained upon request and used under a GNU General Public License.

2.3.7 Competing interests

The authors declare that they have no competing interests.

2.3.8 Authors' contributions

MK specified the requirements from a user's perspective, defined the rules for data handling, and collected all the data. BH and FO designed the database scheme and set up the technical requirements. BH, FO, and MH did the technical design and the programming. MK and BH wrote the manuscript. All authors read and approved the final manuscript.

2.3.9 Acknowledgements and Funding

This work has been funded by grants KO 2251/3-1, KO 2251/3-2, and KO 2251/6-1 of the Deutsche Forschungsgemeinschaft.

2.3.10 Additional files

2.3.10.1 Additional file 1 - Database scheme.

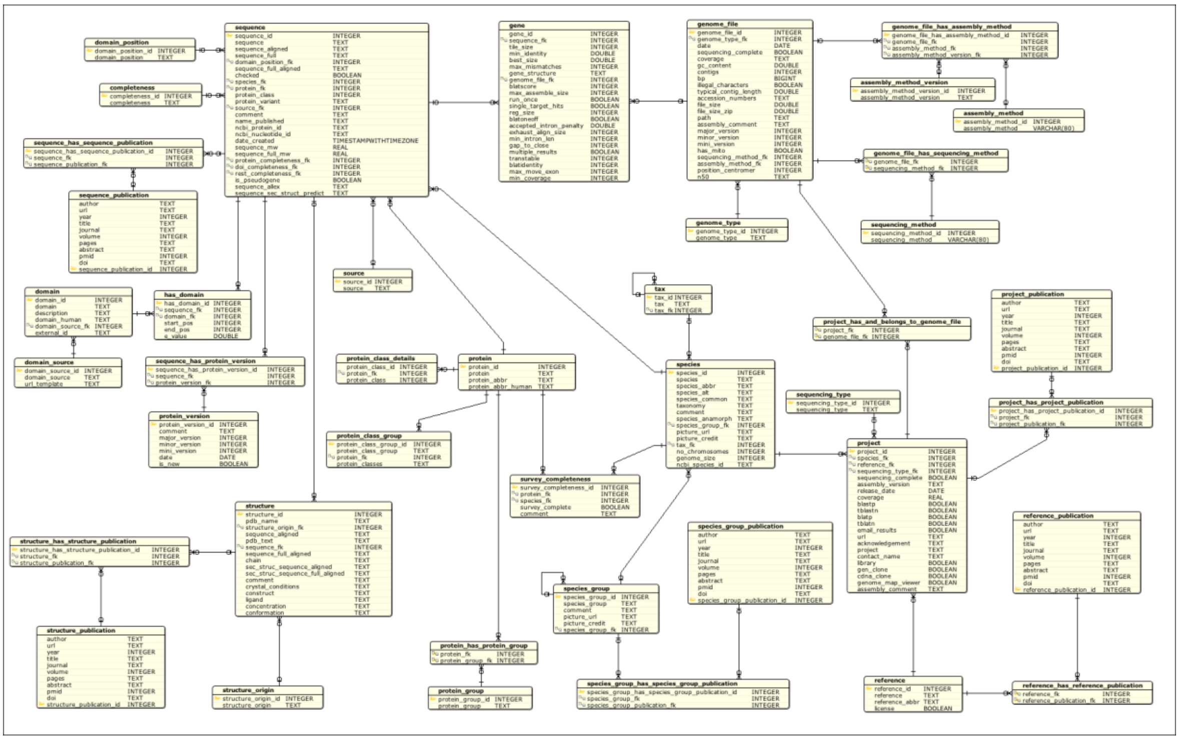


Figure 2.3-8: ***The schema shows the database tables and their relations.*** *Content related tables are grouped and coloured according to Figure 2.3-1 of the main manuscript. For each table the columns are listed with their name and data type. Yellow keys in front of the names signify columns with unique identifiers. Blue window-symbols mark foreign key columns that contain values of id-columns of other tables. Symbols at the right side of the column names designate indices for better performance. Lines are relations between tables. Two unary (recursive) relationships are defined: One linking taxa to their parent taxon and one linking species groups to their parent group.*

2.4 Cross-species protein sequence and gene structure prediction with fine-tuned Webscipio 2.0 and Scipio

Klas Hatje[1*], Oliver Keller[2*], Björn Hammesfahr[1], Holger Pillmann[1], Stephan Waack[2] and Martin[1§]

[1] Abteilung NMR basierte Strukturbiologie, Max-Planck-Institut für Biophysikalische Chemie, Am Fassberg 11, D-37077 Göttingen, Germany
[2] Institute of Computer Science, University of Göttingen, Goldschmidtstr. 7, 37077 Göttingen, Germany
[*] KH and OK contributed equally to the work
[§] Corresponding author

BMC Research Notes
Published: 28 July 2011
BMC Research Notes 2011 4:265 doi:10.1186/1756-0500-4-265 This article is available from http://www.biomedcentral.com/1756-0500/4/265

2.4.1 Abstract

Background
Obtaining transcripts of homologs of closely related organisms and retrieving the reconstructed exon-intron patterns of the genes is a very important process during the analysis of the evolution of a protein family and the comparative analysis of the exon-intron structure of a certain gene from different species. Due to the ever-increasing speed of genome sequencing, the gap to genome annotation is growing. Thus, tools for the correct prediction and reconstruction of genes in related organisms become more and more important. The tool Scipio, which can also be used via the graphical interface WebScipio, performs significant hit processing of the output of the Blat program to account for sequencing errors, missing sequence, and fragmented genome assemblies. However, Scipio has so far been limited to high sequence similarity and unable to reconstruct short exons.

Results
Scipio and WebScipio have fundamentally been extended to better reconstruct very short exons and intron splice sites and to be better suited for cross-species gene structure predictions. The Needleman-Wunsch algorithm has been implemented for the search for short parts of the query sequence that were not recognized by Blat. Those regions might either be short exons, divergent sequence at intron splice sites, or very divergent exons. We have shown the benefit and use of new parameters with several protein examples from

completely different protein families in searches against species from several kingdoms of the eukaryotes. The performance of the new Scipio version has been tested in comparison with several similar tools.

Conclusions

With the new version of Scipio very short exons, terminal and internal, of even just one amino acid can correctly be reconstructed. Scipio is also able to correctly predict almost all genes in cross-species searches even if the ancestors of the species separated more than 100 Myr ago and if the protein sequence identity is below 80 %. For our test cases Scipio outperforms all other software tested. WebScipio has been restructured and provides easy access to the genome assemblies of about 640 eukaryotic species. Scipio and WebScipio are freely accessible at http://www.webscipio.org.

2.4.2 Background

Whole genome sequences of eukaryotes are generated with increasing speed (273). While the focus at the beginning of high-throughput DNA sequencing was on model organisms and the human genome, for which tremendous amounts of secondary data was available, the aims have shifted to organisms of medical or economic relevance (e.g. *Plasmodium falciparum* (274) or *Phytophthora ramorum* (275)), to the comparative analysis of entire taxa (e.g. the Drosophila clade (276) or Candida species (277)), and, very recently, to organisms of evolutionary interest (e.g. *Trichoplax adhaerens* (278) or *Volvox carteri* (279)). However, gene catalogues are only available for a small part of the sequenced organisms and a precise and complete set of genes is still unavailable for even a single species. In the first instance the gene annotation is done with automatic gene prediction programs that either predict only isolated exons, or reconstruct the complete exon-intron structures of the protein-coding genes, or even try to predict 5’ and 3’ untranslated regions. *Ab-initio* gene prediction programs only use the assembled DNA sequences as input, having precomputed models for nucleotide distributions, while evidence-based programs consider alignments of ESTs, cDNAs, or annotated sequences from closely related organisms, with the target sequence (reviewed in (280)). The highest accuracy is reached by programs that combine model-based and alignment-based approaches (281,282).

For many biological applications like the phylogenetic analysis of a protein family (e.g. (125)) or the comparative analysis of the exon-intron structure of a certain gene from different species (e.g. (283)), it is necessary to obtain translated transcripts of homologs of closely related organisms or the reconstructed exon-intron patterns of the genes, respectively. The protein sequences of homologs of a certain protein can be obtained in several ways. Annotations based on *ab-initio* gene predictions, sometimes supplemented by EST data, are available for about half of the sequenced eukaryotic genomes, although it is often tedious to find the corresponding data via the FTP-pages of the sequencing centers. In addition, automatic predictions are not complete and in many cases not correct. For very few eukaryotes, full-length cDNA data can be accessed. However, these data never cover the complete transcriptome of the species. Another possibility is to manually annotate the protein homologs in the genomes of choice by comparative genomics. This is certainly the most accurate way. By this approach a multiple sequence alignment of as many as possible homologs is created, and based on this sequence alignment mispredicted sequence regions (insertions and missing regions) are easily detected. Further homologs are added by manual inspection of the corresponding genomic DNA regions and manual reconstruction of intron splice sites. Splice sites are in most cases conserved throughout the eukaryotes (284) and therefore their position and frame can be used for gene reconstructions by comparing gene structures from known and to be annotated genes.

To assist in the task of the manual annotation of eukaryotic genomes, and to provide options for genomes for which gene prediction data is not available, we have recently developed Scipio (13,212). Scipio is a post-processing script for the Blat output (285) and maps a protein sequence to a genomic DNA sequence. Blat has been developed for the fast alignment of very similar DNA or protein sequences. However, Blat is not able to identify very short exons (two or three amino acids, or exons of just the N-terminal methionine), it is not able to assemble genes spread on more than one contiguous DNA sequence, it misses exons that are too divergent, it does not apply biological sequence models to determine exact splice site locations on nucleotide level, or to distinguish introns from insertions caused by frameshifts or in-frame stop codons (13,212,286). Scipio is able to address most of these issues resulting in considerably improved gene structure reconstructions (13,212). Its initial intention was to cope with sequencing errors, to assemble genes from highly fragmented genome assemblies, and to reconstruct intron splice sites. Scipio was not able to correctly reconstruct very short exons or to correctly reconstruct genes in cross-species searches if these were not highly identical.

Here, we present the fundamentally improved version of the Scipio software that has been extended for the use in cross-species searches. In addition, very short exons and divergent regions at intron borders are now correctly reconstructed. Scipio can be used via the web-interface WebScipio that provides access to 2111 genome assembly files for 592 species (end of February 2011).

2.4.3 Methods

The presented software consists of two programs that form a pipeline for the output of the external program Blat, which is executed first. The Blat results are post-processed by the Scipio script written in Perl (287). WebScipio provides a graphical user interface for Scipio that we have developed using the web framework Ruby on Rails (78,79). The workflow was optimized to direct the user to the necessary input parameters. This was implemented with the technique of Asynchronous Javascript and XML (AJAX). Visual effects were realized with the help of Prototype (254) and script.aculo.us (288) that are JavaScript libraries, which are integral parts of Ruby on Rails.

Scipio

The Scipio Perl script itself, which can also be run standalone, has undergone numerous extensions that are based on our extensive experience in manual gene annotation (35,63,125). The general setup of the script that aimed to handle all the various sequencing and assembly errors has already been described (212); here, we present an implementation of the Needleman-Wunsch algorithm which is the main extension to the previous version.

The Needleman-Wunsch Algorithm used in Scipio

In the updated version of Scipio, we use a modified Needleman-Wunsch style dynamic programming (DP) algorithm to perform an exhaustive search for the best-scoring spliced alignment between the query and target sequence fragments that were left unmatched by Blat. Like the original Needleman-Wunsch algorithm, it calculates an optimal global alignment between the sequences, but it is adjusted to find an optimal *spliced* alignment between a protein query sequence s and a genomic target sequence t. Given the computational cost of $|s|$ and $|t|$, it is executed only on very short sequence fragments s and t. We introduce different categories of penalties depending on the type of matching. Any alignment can be represented by a *parse* Φ: a collection of pairs of strings (s_1, t_1), … , (s_k, t_k), such that the aligned sequences are the concatenations: $s = s_1 \ldots s_r$, $t = t_1 \ldots t_r$. A *penalty score* $p(s_k, t_k)$ is assigned to each pair as follows:

- if s_k is a single residue and t_k a string of length 3 (codon), then $p(s_k, t_k) = p_{\mathrm{MAP}}(s_k, t_k)$ is a *match/mismatch* penalty:

$$p_{\mathrm{MAP}}(s_k, t_k) = \begin{cases} 0, & \text{if } t_k \text{ translates to } s_k \\ p_{\mathrm{MISM}}, & \text{if not} \end{cases}$$

- an *insertion* penalty $p(s_k, t_k) = p_{\mathrm{INS}}$ is assigned to them if s_k is a single residue and t_k is empty
- a *gap* penalty $p(s_k, t_k) = p_{\mathrm{GAP}}$ is assigned to them if t_k is a codon and s_k is empty
- a *frameshift* penalty $p(s_k, t_k) = p_{\mathrm{FS}}$ is assigned to them if t_k consists of 1 or 2 nucleotides, and s_k is empty or a single residue

To cover the case of introns, in addition we define *intron* penalties based on the donor and acceptor splice sites:

$$p_{\mathrm{INTRON}}\left(n_1 \dots n_\ell\right) = p_{\mathrm{DSS}}\left(n_1 n_2\right) + p_{\mathrm{INT}} + p_{\mathrm{ASS}}\left(n_{\ell-1} n_\ell\right)$$

with a constant value p_{INT} for any sequence of nucleotides $n_1 \dots n_\ell$ and zero splice site penalties if $n_1 n_2$ = “GT”, and $n_{\ell-1} n_\ell$ = “AG”. We distinguish two cases: in-frame introns, and introns splitting codons:

- if s_k is empty and t_k exceeds the minimum intron length, then $p(s_k, t_k) = p_{\mathrm{INTRON}}(t_k)$ is the intron *penalty*
- if s_k is a single residue $n_1 n_2 n_3$, and $t_k = n_1 \omega n_2 n_3$, or $t_k = n_1 n_2 \omega n_3$ with single residues and ω a string exceeding the minimum intron length, then the penalty is a combined match/intron penalty: $p(s_k, t_k) = p_{\mathrm{INTRON}}(\omega) + p_{\mathrm{MAP}}(s_k, n_1 n_2 n_3)$. Here, two different penalties are defined (depending on the frame of the intron), and thus the minimum of them is taken.

If (s_k, t_k) does not satisfy any of these conditions, no penalty is defined resulting in an invalid parse. By combining insertions, deletions, and frameshifts, there is always some valid parse for any given pair of sequences. The cost of a parse Φ is the sum of the penalties: $p(\Phi) = p(s_1, t_1) + \dots + p(s_r, t_r)$, and we calculat

$$p(s,t) = \min\left\{p(\Phi) \mid \text{is a valid parse aligning } s \text{ and } t\right\}$$

by computing the DP matrix (M_{ij}) containing the minimal score for an alignment of the subsequences $s_{[0..j\text{-}1]}$ and $t_{[0..i\text{-}1]}$, using the following recursions:

$$M_{ij} = \min \begin{cases} M_{(i-3)(j-1)} + p_{\mathrm{MAP}}\left(s_{[j-1]}, t_{[i-3,i-2,i-1]}\right) \\ M_{i(j-1)} + p_{\mathrm{INS}}, \\ M_{(i-3)j} + p_{\mathrm{GAP}}, \\ \min\left\{M_{(i-1)j}, M_{(i-2)j}, M_{(i-1)(j-1)}, M_{(i-2)(j-1)}\right\} + p_{\mathrm{FS}}, \\ \min\limits_{i' \le i-\ell_{\min}} \left\{M_{i'j} + p_{\mathrm{INTRON}}\left(t_{[i'..i-1]}\right)\right\} \\ \min\limits_{i' \le i-\ell_{\min}-3} \left\{M_{i'(j-1)} + p_{\mathrm{MAP}}\left(s_{[j-1]}, t_{[i']}t_{[i-2,i-1]}\right)\right\} + p_{\mathrm{INTRON}}\left(t_{[i'+1..i-3]}\right) \\ \min\limits_{i' \le i-\ell_{\min}-3} \left\{M_{i'(j-1)} + p_{\mathrm{MAP}}\left(s_{[j-1]}, t_{[i',i'+1]}t_{[i-1]}\right)\right\} + p_{\mathrm{INTRON}}\left(t_{[i'+2..i-2]}\right) \end{cases}$$

where each of these expressions corresponds to one of the possible penalty types for the last segment of the parse.

The last three lines cover introns, one for each reading frame, with $\ell_{\min}$ denoting the minimum intron length. To avoid having to iterate over all values for i' in these cases, we precompute nine variants of the score matrix with partial intron penalties added (indexed by a nucleotide n if it splits a codon) as follows:

$$
\begin{aligned}
M^{(0)}_{i,j} &= \min_{i' \le i - \mathrm{l}_{\min}} \left\{ M_{i'j} + p_{\mathrm{DSS}}\left(t_{[i',i'+1]}\right)\right\} + p_{\mathrm{INT}} \\
M^{(1)}_{i,j,n} &= \min_{\substack{i' \le i - \mathrm{l}_{\min} - 3 \\ t_{[i']} = n}} \left\{ M_{i'(j-1)} + p_{\mathrm{DSS}}\left(t_{[i'+1,i'+2]}\right)\right\} + p_{\mathrm{INT}} \\
M^{(2)}_{i,j,n} &= \min_{i' \le i - \mathrm{l}_{\min} - 3} \left\{ M_{i'(j-1)} + p_{\mathrm{MAP}}\left(s_{[j-1]}, t_{[i,'i'+1]}n\right) + p_{\mathrm{DSS}}\left(t_{[i'+2,i'+3]}\right)\right\} + p_{\mathrm{INT}}
\end{aligned}
$$

Note that n denotes the nucleotide before the intron in $M^{(1)}$, and the nucleotide after it in $M^{(2)}$. The latter contains already the mismatch penalty, while the former does not. With i' the latest segment start allowed ($i' = i - \ell_{\min}$ for an intron scored by $M^{(0)}$, and $i' = i - \ell_{\min} - 3$ for a codon split by an intron), the intron variables are given recursively by

$$
\begin{aligned}
M^{(0)}_{i,j} &= \min \left\{ M_{i'j} + p_{\mathrm{DSS}}\left(t_{[i',i'+1]}\right) + p_{\mathrm{INT}}, M^{(0)}_{i-1,j} \right\} \\
M^{(1)}_{i,j,n} &= \min \left\{ M_{i'(j-1)} + p_{\mathrm{DSS}}\left(t_{[i'+1,i'+2]}\right) + p_{\mathrm{INT}}, M^{(1)}_{i-1,j,n} \right\} && (n = t_{[i']}) \\
M^{(1)}_{i,j,n} &= M^{(1)}_{i-1,j,n} && (n \ne t_{[i']}) \\
M^{(2)}_{i,j,n} &= \min \left\{ M_{i'(j-1)} + p_{\mathrm{MAP}}\left(s_{[j-1]}, t_{[i,'i'+1]}n\right) + p_{\mathrm{DSS}}\left(t_{[i'+2,i'+3]}\right) + p_{\mathrm{INT}}, M^{(2)}_{i-1,j,n} \right\}
\end{aligned}
$$

and then replace the last three lines in the recursion for Mij:

$$
M_{ij} = \min \begin{cases} \dots, \\ M^{(0)}_{i,j} + p_{\mathrm{ASS}}\left(t_{[i-2,i-1]}\right) \\ \min\limits_{n=a,c,g,t} \left\{ M^{(1)}_{i,j,n} + p_{\mathrm{MAP}}\left(s_{[j-1]}, n\, t_{[i-2,i-1]}\right)\right\} + p_{\mathrm{ASS}}\left(t_{[i-4,i-3]}\right) \\ M^{(2)}_{i,j,n} + p_{\mathrm{ASS}}\left(t_{[i-3,i-2]}\right) \end{cases}
$$

The penalties for the Needleman-Wunsch algorithm can be adjusted manually in the Scipio command-line version but not via the WebScipio web-interface. The penalties need to be well balanced so that the Needleman-Wunsch search does not result for example in a number of artificial short exons where a long exon is missing due to a gap in the genome assembly. Based on extensive tests with in-house test data we set the following values as default: mismatch-penalty: 1.0; insertion-penalty: 1.5; gap-penalty: 1.1; frameshift-penalty: 2.5; intron-penalty: 2.0 + the respective penalties for donor and acceptor splice sites.

WebScipio

At present, the web interface offers 2272 genome files of 643 eukaryotic organisms. Metadata corresponding to the species, like assembly versions, sequencing centers, and assembly coverage, is available from the diArk database (210). WebScipio reads the metadata out of a periodically updated text file generated from diArk, or queries the diArk database directly with SQL.

The gene structure schemes resulting from the Scipio run are generated and displayed in the Scalable Vector Graphics (SVG) format (259). This allows scaling the graphics while retaining their resolution and to show tooltips generated with JavaScript and HTML for each element of the gene structure schemes. For browsers not supporting SVG, a fall back solution is implemented, which uses the Portable Network Graphics (PNG) format. The PNG files are generated by Inkscape (289).

Internally, the sequence data is processed with the help of BioRuby (260). Results are saved in the YAML format (290), but are also available for download in the GFF format. The web application runs the Blat and Scipio jobs in the background, which was implemented using the Rails plug-in Workling in combination with Spawn (291,292). The server-side stored session data is increasing with every extension of WebScipio. To make the session storage fast, flexible, and scalable we use a database backend called Tokyo Cabinet (293). It offers a simple key-value store, also called hash store, for accessing different data objects with the help of a unique key for each object. Tokyo Tyrant is the network interface to Tokyo Cabinet and allows storing data across the network on several servers. It is used in WebScipio for scalability reasons.

External Tools

We use Hoptoad for error reporting (294). It is a web application that collects errors generated by WebScipio, aggregates them to the detailed error reports for developer review, and sends email notifications. We use a behaviour-driven testing strategy to validate the functionality and behaviour of WebScipio. For the automation of these tests we use RSpec (295), which is a behaviour-driven development framework for the Ruby programming language. Our intention for this test implementation was the need of reliability and accuracy within the continuously extended software. Application tests are run with Selenium, a test system for web applications (296). This offers the opportunity to test the web-interface as a whole. Selenium integrates into the Mozilla Firefox browser as a plug-in that records the user interaction in the form of a Ruby script. To run the test scripts without user-interaction, Selenium starts and controls the browser automatically. We integrated the user-interface tests into our automated test environment as additional RSpec test cases.

2.4.4 Results and Discussion

Scipio and WebScipio workflow and general parameters for fine-tuning gene predictions

The general workflow of Scipio and WebScipio is similar to that described previously (13,212). Scipio provides some general search parameters that filter the Blat output for further post-processing, and offers several expert options that influence the post-processing steps. In the new Scipio version, especially the part of the gap-closing (mapping the parts of the query sequence to the target sequence that Blat failed to recognize) and hit extension (modelling the regions at exon borders, including terminal exons, where homology was too low to be identified by Blat) has been improved (Figure 2.4-1, see also Additional file 2.4.11.1). This has been done by implementing the Needleman-Wunsch algorithm for the search of unmapped query sequence in respective target regions and by introducing parameters that allow a higher divergence from the exon border regions predicted by Blat. All new parameters are adjustable by the user although the default values should be good enough for most cases. However, especially when searching for very divergent homologs or when searching for homologs of very divergent species, these parameters might need manual adaptation. Figure 2.4-1 shows a detailed scheme of the Scipio workflow including all parameters that can manually be adjusted. Also, some of the most important decisions are outlined that Scipio makes to provide the best possible result. The detailed scheme should allow the experienced user to fine-tune the search in especially difficult cases. The rationale for implementing each of the parameters and its consequences are explained below.

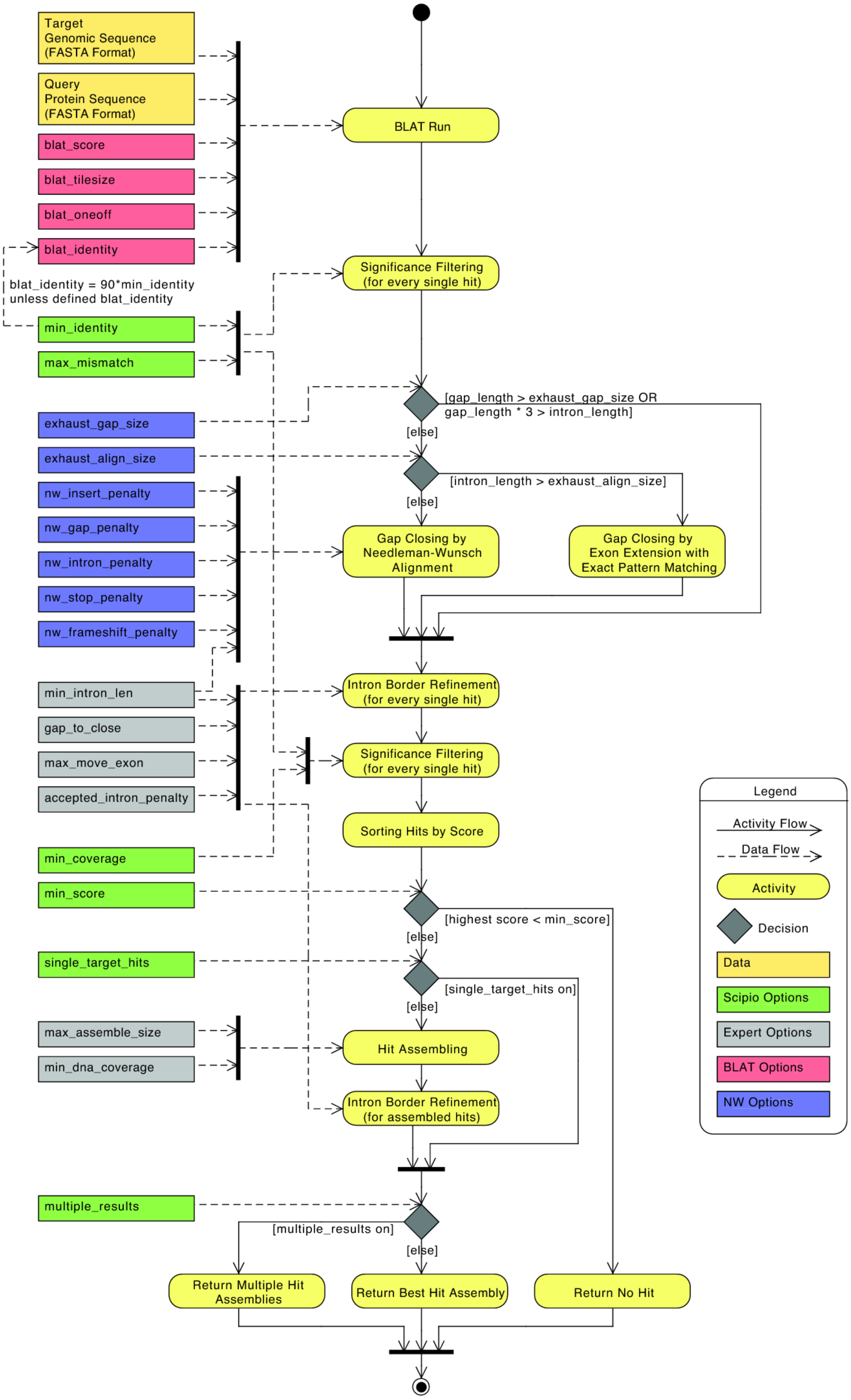


*Figure 2.4-1: **The extended Scipio workflow.** This diagram depicts the activity and data flow of a Scipio run. Scipio needs a protein and a target genome sequence, both in FASTA format, as input to start a Blat run. Every single Blat hit is subsequently processed and filtered, and assembled in the case of hits on multiple targets. The gap_length describes the number of amino acids of an unmatched query subsequence. The intron_length is the corresponding length of the unmatched target subsequence in nucleotides.*

The new web-interface

Because we wanted to offer most of the new parameters to the experienced user via the web-interface WebScipio, and we planned to introduce searches for alternatively spliced exons, we had to redesign the WebScipio workflow. The goal was to keep it well structured, intuitive and clear. We have also improved the usability for new and less experienced users by providing more examples, help pages, and documentation. The general design of selecting one target sequence for the search for multiple query sequences has been retained. Next, the experienced user can adjust many of the Scipio variables, and, also at this stage, many of the parameters for searches for alternative exons (those parameters are described elsewhere). We provide some default values for cross-species searches that are based on our experience in working with and knowledge about eukaryotic genomes (125). For example, some genomes are known to contain only small numbers of introns while others are known to contain only short introns. Special settings for cross-species searches are provided for several specific taxa but the default cross-species parameters should be applicable for most genomes. Having selected a specific set of parameters every single parameter can still be adjusted individually.

As before, the most important result view is the scheme of the exon-intron structure of the search result. In this scheme, all information regarding the quality of the result (complete versus incomplete, containing gaps, i.e. unmatched parts of the query sequence, questionable introns, mismatches, frame-shifts, in-frame stop-codons, etc.) is included. Opening the “Search details” box provides further information concerning the search parameters, and additional data regarding the aligned query sequence is available from the different result views.

Due to gene and whole genome duplications during eukaryotic evolution there are often two or more closely related homologs of a certain protein per genome. This might cause some problems for cross-species searches if the paralogs in the target genome are about equally homologous to the query sequence. Therefore, we implemented a --multiple_results parameter. Switching --multiple_results off is the best way to get the exact gene structure for an intra-species search. Switching --multiple_results on (default setting in cross-species searches) allows retrieving all possible results depending on the general search parameters (like --min_score or --min_identity, Figure 2.4-2). If multiple hits are found they will be listed separately and can be analysed using the various result views. In addition, we implemented a quick view showing the gene structure schemes as a fast overview. As example for the benefit and limitation of this parameter, we searched for class-II myosin heavy chain homologs in humans (Figure 2.4-2). It is known that vertebrate genomes contain several muscle myosin heavy chain genes (belonging to the class-II myosin heavy chains) that are specialised for certain tissues like heart muscles or skeletal muscles (125). Six of these genes are encoded in a cluster (297). The example search shows the gene structure corresponding to the query sequence (*Hs*Mhc1_fl) and the gene structures of six homologs of varying degree of divergence. While the closest homolog (*Hs*Mhc1_fl_(1)) only contains mismatches compared to the query sequence, the three next closest homologs have severely deviating gene structures. They contain very long introns in the middle of the genes indicating that they are mixed genes assembled from the N-terminal half from one gene of the muscle myosin heavy chain cluster and the C-terminal half taken from the following gene of the cluster. The next two homologs are already very divergent so that parts of the genes cannot be reconstructed leaving many and long gaps.

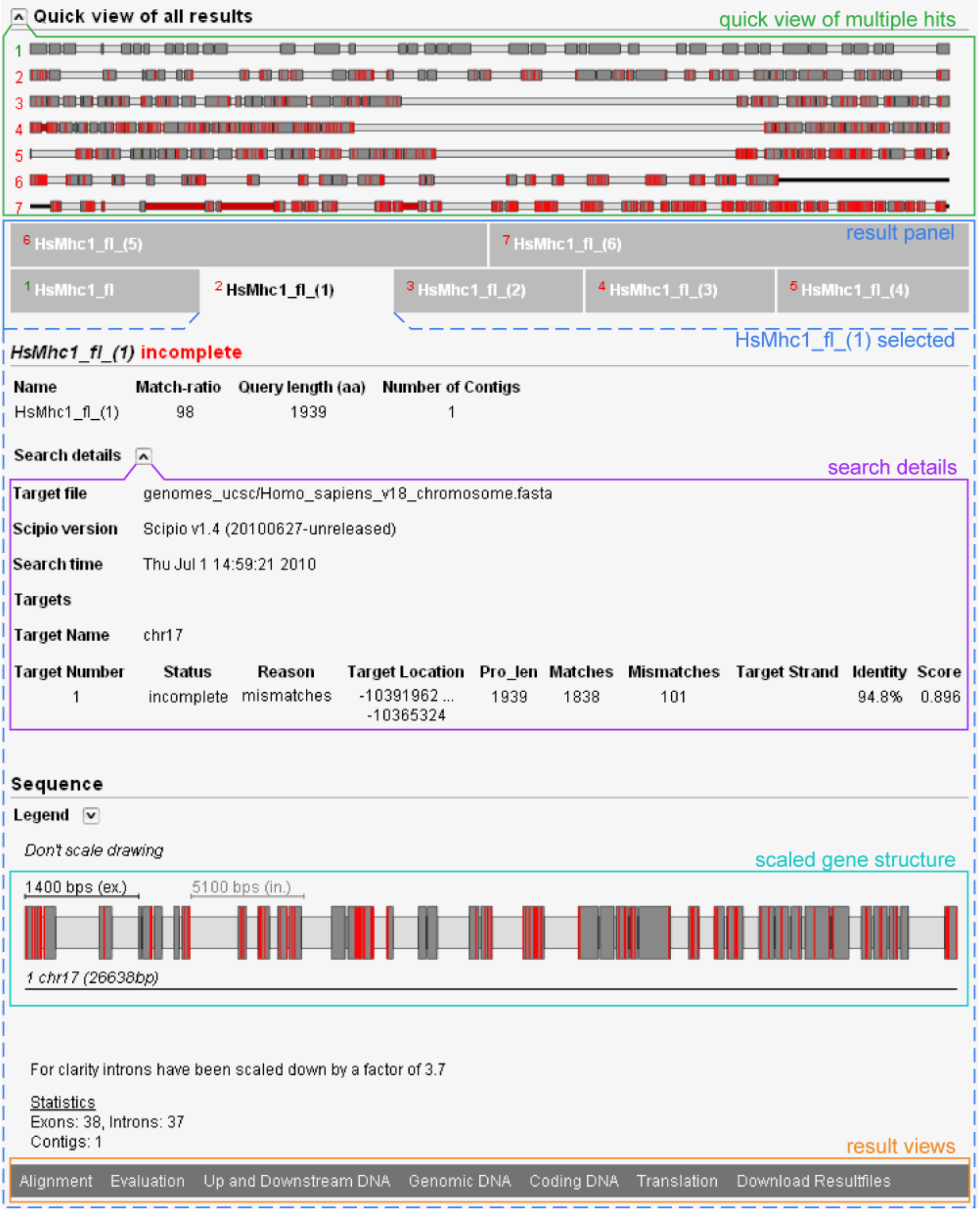
Use of WebScipio to produce publication-quality figures of gene structures

WebScipio can be used to easily produce publication-quality figures of gene structures. Either, these figures can be produced in the described way, or the user can upload an own genomic DNA sequence for use as target sequence. This is interesting in the case that the whole genome sequence is not known but only the genomic sequence of a certain region. SVGs can be downloaded and further processed in many graphics programs.

New general and expert search parameters

The parameters --min_score (previously: --best_size), --min_identity, and --max_mismatch have already been described(13) and define the threshold for the Blat hits to be processed by Scipio. To reduce or even abolish the artificial assembly of contigs that by chance contain some identical residues we have introduced the parameter --min_coverage that applies to every single Blat hit. The coverage is the number of mapped residues (as match or mismatch) divided by the query length of the (possibly partial) hit. By default, Scipio rejects hits with coverage of less than 60%.

In addition to these general parameters we have introduced several expert options most of which will be described in detail below. One of the parameters is --transtable that allows the user to specify a non-standard translation table, for the use with species like Candida species, *Tetrahymena thermophila* and others that would otherwise lead to mismatches. Another parameter called --accepted_intron_penalty is used to define valid splice sites. By default, GT---AG and GC---AG are accepted, whereas, for example, introns with the pattern AT---AC would be classified as doubtful (“intron?”). By adjusting the --accepted_intron_penalty parameter those introns will also be accepted instead of defining those introns as “intron?”.



HsMhc1_fl_(1) **incomplete**

Name	Match-ratio	Query length (aa)	Number of Contigs
HsMhc1_fl_(1)	98	1939	1

Search details

Target file	genomes_ucsc/Homo_sapiens_v18_chromosome.fasta
Scipio version	Scipio v1.4 (20100627-unreleased)
Search time	Thu Jul 1 14:59:21 2010
Targets	
Target Name	chr17

Target Number	Status	Reason	Target Location	Pro_len	Matches	Mismatches	Target Strand	Identity	Score
1	incomplete	mismatches	-10391962 ... -10365324	1939	1838	101		94.8%	0.896

Sequence

For clarity introns have been scaled down by a factor of 3.7

Statistics
Exons: 38, Introns: 37
Contigs: 1

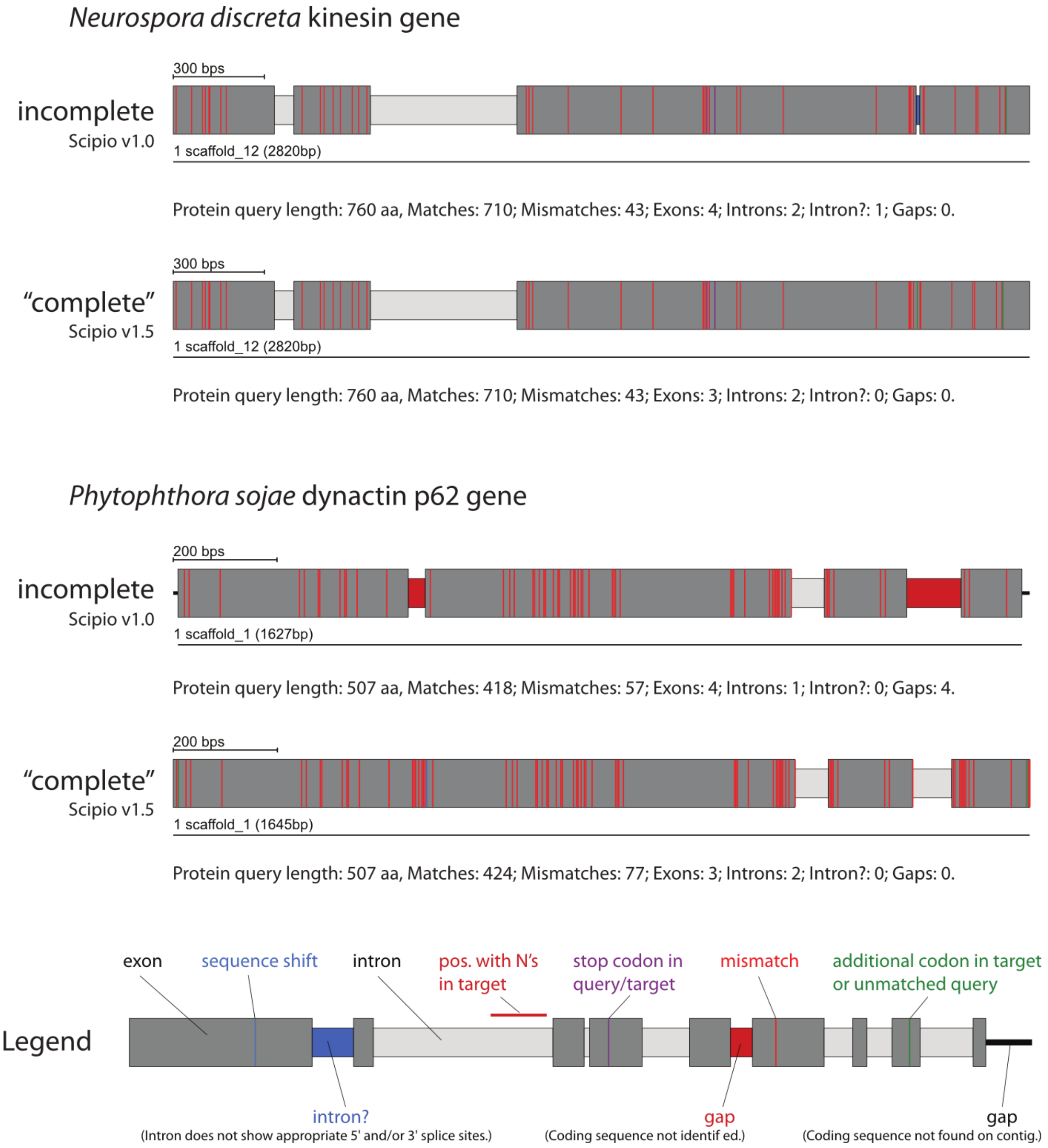
Figure 2.4-2: ***Screenshot of the multiple results view of WebScipio.*** *The screenshot shows the result of the search for multiple homologs of one of the muscle class-II myosins from human in the human genome. The search parameters were --min_identity=60%,--max_mismatch=∞, and --multiple_results=yes to get as many homologs as possible. On top, the opened quick view of all reconstructed gene structures is shown. Next, a panel with the different results is shown. Green numbers mark complete results (100% of the query sequence reconstructed) while red numbers mark incomplete results (might contain gaps, mismatches, frameshifts, etc.). Result hit number 2 was selected and shows the result for the closest homolog to the query sequence with no gaps (unmapped query sequence) but 101 mismatches.*

Parameters to account for additional/missing bases in predicted exons

Gene homologs even from very closely related species are often too divergent to be completely identified by Blat. While the core building block of the proteins and the functional sites are often strongly conserved, low homology is especially found at the surface of the proteins. Thus, loop regions are often sites of amino acid substitutions, insertions of long stretches of residues, and deletions. In addition, since the terminal regions of most proteins are at the surface, they are also often very divergent. Short stretches of nucleotides whose lengths are multiples of three and whose translations do not result in any in-frame stop codons are most likely to be insertions rather than true introns.

A parameter --min_intron_len has been implemented to distinguish introns from insertions, with a default minimum intron length of 22 nucleotides. A minimum intron length of 22 nucleotides is a rather conservative estimate given the minimum intron length of 35-40 nucleotides based on a test set of about 17,000 introns of genes of 10 model organisms (298). Thus, by default additional coding sequence for up to seven amino acids (= 21 nucleotides) will be treated as exon sequence and joined with the surrounding exons into a single exon.

The opposite case of extra amino acids in the query is dealt with by the parameter --gap_to_close. By default, a mapping of up to six additional amino acids from the query sequence to the exon borders will be enforced at the cost of further mismatches, in order to eliminate a gap (of unmatched query sequence). This parameter also effects the modelling of the intron borders (see below). Figure 2.4-3 shows two examples of cross-species searches in which the target sequence contains additional or less amino acids in conserved exons. Case A shows the results of a search for a kinesin homolog from *Neurospora crassa* (query sequence) in the closely related organism *Neurospora discreta* (target sequence, see also Additional file 2.4.11.2). Because of the relatively high homology of the two sequences, Blat has already retained the additional residues of the query sequence so that they are included in the result of the old Scipio version. However, a questionable intron (called intron? in Scipio) was introduced in the region that contained additional nucleotides in the target sequence leading to missing residues in the target translation. With the new parameter --min_intron_len these additional nucleotides are correctly treated as exonic sequence. Case B shows an example of two divergent homologs of the dynactin p62 gene of *Phytophthora ramorum* (query sequence) and *Phytophthora sojae* (target sequence, see also Additional file 2.4.11.2). These two homologs contain a long divergent region with many consecutive mismatches in the first exon that is not identified by Blat and introduces a long gap of unmatched residues. In addition, the N- and C-termini have divergent sequences and different lengths. With the new parameters, Scipio can correctly model the target gene.


Neurospora discreta kinesin gene

Protein query length: 760 aa, Matches: 710; Mismatches: 43; Exons: 4; Introns: 2; Intron?: 1; Gaps: 0.

Protein query length: 760 aa, Matches: 710; Mismatches: 43; Exons: 3; Introns: 2; Intron?: 0; Gaps: 0.

Phytophthora sojae dynactin p62 gene

Protein query length: 507 aa, Matches: 418; Mismatches: 57; Exons: 4; Introns: 1; Intron?: 0; Gaps: 4.

Protein query length: 507 aa, Matches: 424; Mismatches: 77; Exons: 3; Introns: 2; Intron?: 0; Gaps: 0.

Figure 2.4-3: ***Modelling of additional/missing bases in gene predictions.*** *Case A shows the result of the search of a kinesin from Neurospora crassa (query sequence) in Neusrospora discreta (target sequence) using the old and the new Scipio version. The --min_intron_len parameter has been set to 22. Case B shows the result of a search of the dynactin p62 homolog from Phytophthora ramorum (query sequence) in Phytophthora sojae (target sequence). To get the correct gene prediction the following Scipio parameters have been used: --min_identity=60%, --min_score=0.3, --max_mismatch=∞, --gap_to_close=15, --min_intron_length=22. The colour coding is explained in the legend and applies to all gene structure figures. For further information see Additional file 2.4.11.2.*

Parameters to identify divergent exons and very short exons ignored by Blat

To identify exons that contain too many mismatches to be identified by Blat, and to correctly annotate very short exons, the Needleman-Wunsch algorithm described above forces an alignment of unmatched query sequence to spare target sequence. Very short exons of one to four amino acids are only reconstructed if they are identical to the query sequence and contain valid splice sites while short exons of five to seven amino acids are also often correctly reconstructed if they contain mismatches between query and target sequence (e.g. in cross-species searches). The maximal lengths of query and target sequence fragments to be aligned with Needleman-Wunsch are controlled by the parameters --exhaust_align_size and --exhaust_gap_size, respectively. By default, the exhaustive search is restricted to query gaps of 21 amino acids (three times the default Blat tilesize), since we expect Blat to successfully discover at least parts of any longer exons, and to a target subsequence of 15,000 bps. The restriction of the latter value is caused by the exponentially increased run time with increased target subsequence so that for example the potentially very long introns in mammalian genomes are only searched after manual increase of this value. Other parameters affecting the Needleman-Wunsch algorithm, such as the penalties mentioned above, can be adjusted by the command line version only, and not via WebScipio. However, the default values have extensively been tested with in-house data and should not require changes in most if not all cases.

The effect of the new parameters on the search results is demonstrated with the examples shown Figure 2.4-4 (see also Additional file 2.4.11.2). In case A, the human dynactin p50 gene contains two very short exons of 3 and 2 amino acids. These two short exons are conserved in all vertebrates (1). Case B shows the coronin gene from the basidiomycote fungi *Puccinia graminis* encoding a short 3 amino acid exon (Figure 2.4-4, see also Additional file 2.4.11.2). In addition, the codons at the exon/intron junctions of this short exon are split. In most of the other basidiomycotes sequenced so far, this short exon is part of one of the neighbouring exons, or part of a longer exon that includes both neighbouring exons. However, it also exists in the basidiomycote *Melampsora laricis-populina*. Thus, this short exon is not an artificial creation but a true exon. Case C presents the dynactin p150 gene that contains three short exons of 7, 6, and 7 amino acids at the beginning of the gene (Figure 2.4-4, see also Additional file 2.4.11.2). Even with the Blat-tilesize set to 5 those exons are not recognized in the search against the chromosome assembly. This example best demonstrates the effect of the --exhaust_align_size (default setting 15,000 bps) and the --exhaust_gap_size (default setting 21aa) parameters to completely reconstruct the respective part of the gene. At the 3'-end of the p150 gene, there is another very short exon that shows some homology to the beginning of the preceding intron and is therefore added to the 3’-end of the preceding exon although this results in some mismatches. This behaviour has also been corrected in the new Scipio version by some other parameters (see below).

Genes might not only contain very short exons between other exons but also at gene termini. Scipio uses an exact pattern search for N-terminal and C-terminal exons. Terminal exons will only be accepted if they match the query sequence and if the resulting intron borders agree with the two most common splice site patterns (GT---AG and GC---AG). The length of the terminal exons searched for is limited by the --gap_to_close parameter that is by default six residues.

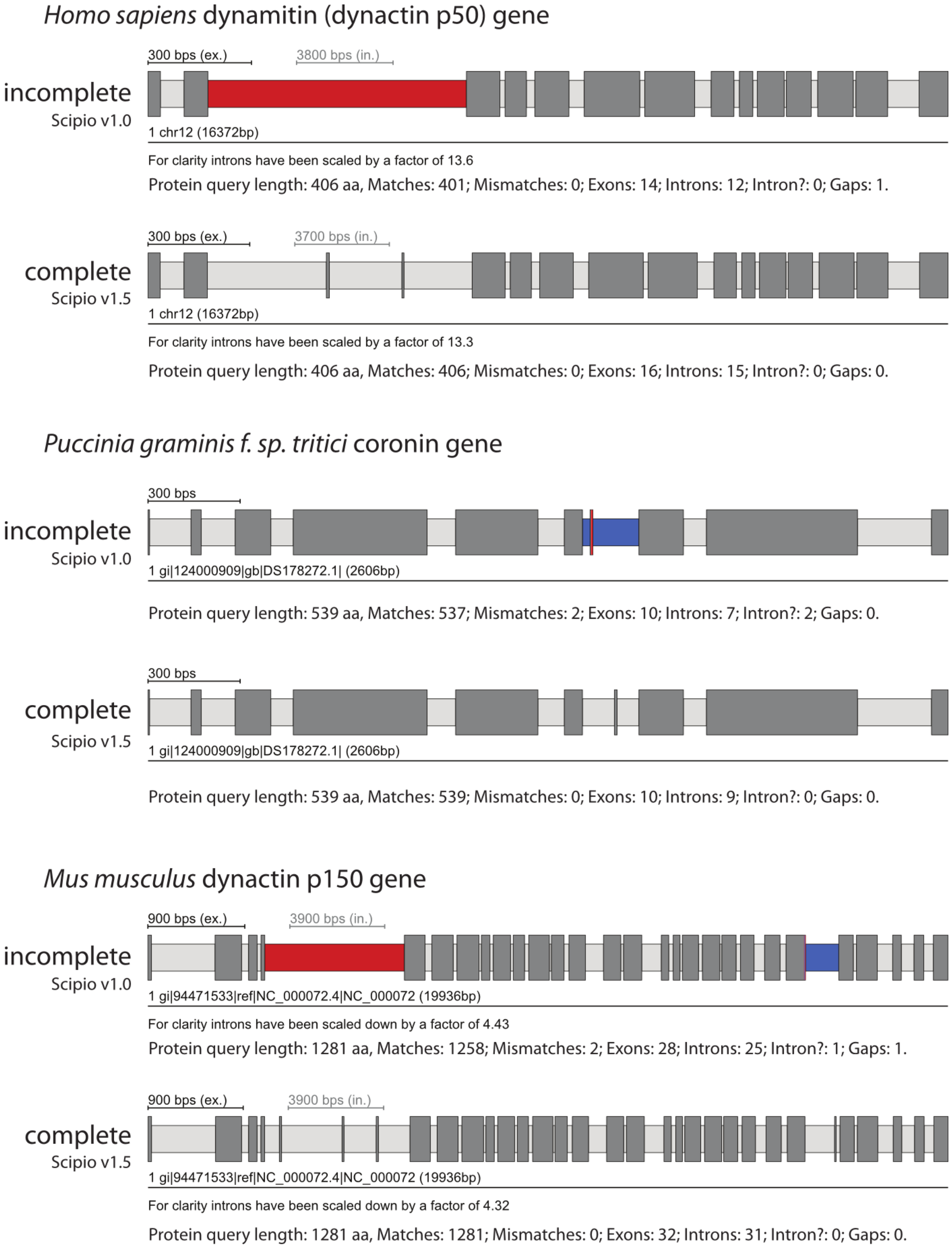

Homo sapiens dynamitin (dynactin p50) gene
300 bps (ex.)
3800 bps (in.)
incomplete
Scipio v1.0
1 chr12 (16372bp)
For clarity introns have been scaled by a factor of 13.6
Protein query length: 406 aa, Matches: 401; Mismatches: 0; Exons: 14; Introns: 12; Intron?: 0; Gaps: 1.
300 bps (ex.)
3700 bps (in.)
complete
Scipio v1.5
1 chr12 (16372bp)
For clarity introns have been scaled by a factor of 13.3
Protein query length: 406 aa, Matches: 406; Mismatches: 0; Exons: 16; Introns: 15; Intron?: 0; Gaps: 0.
Puccinia graminis f. sp. tritici coronin gene
300 bps
incomplete
Scipio v1.0
1 gi|124000909|gb|DS178272.1| (2606bp)
Protein query length: 539 aa, Matches: 537; Mismatches: 2; Exons: 10; Introns: 7; Intron?: 2; Gaps: 0.
300 bps
complete
Scipio v1.5
1 gi|124000909|gb|DS178272.1| (2606bp)
Protein query length: 539 aa, Matches: 539; Mismatches: 0; Exons: 10; Introns: 9; Intron?: 0; Gaps: 0.
Mus musculus dynactin p150 gene
900 bps (ex.)
3900 bps (in.)
incomplete
Scipio v1.0
1 gi|94471533|ref|NC_000072.4|NC_000072 (19936bp)
For clarity introns have been scaled down by a factor of 4.43
Protein query length: 1281 aa, Matches: 1258; Mismatches: 2; Exons: 28; Introns: 25; Intron?: 1; Gaps: 1.
900 bps (ex.)
3900 bps (in.)
complete
Scipio v1.5
1 gi|94471533|ref|NC_000072.4|NC_000072 (19936bp)
For clarity introns have been scaled down by a factor of 4.32
Protein query length: 1281 aa, Matches: 1281; Mismatches: 0; Exons: 32; Introns: 31; Intron?: 0; Gaps: 0.

Figure 2.4-4: ***Reconstruction of very short exons.*** *Case A shows the result for the reconstruction of the human dynamitin (dynactin p50) gene, that contains a 3 amino acid exon and a following 2 amino acid exon that are differentially included in the final transcript. These exons could not be reconstructed with Blat and the old Scipio version, but using the new Scipio version that enables Needlman-Wunsch searches. The --exhaust_align_size parameter has been set to 15,000bp because of the length of the intron. Case B shows the result of the reconstruction of the coronin gene from Puccinia graminis f. sp. tritici. The small but evolutionarily conserved exon 7 can now correctly be reconstructed. Case C shows the result of the reconstruction of the mouse dynactin p150 gene that contains three short exons of 7, 6, and 7 amino acids close to the 5'-end of the gene. For the correct reconstruction, the --exhaust_align_size parameter has been increased to 10,000 bp, because of the length of the intron, and the --exhaust_gap_size has been set to 21 because of the length of the query that could not be mapped. The colour coding of the scheme is the same as in Figure 2.4-3. For further information see Additional file 2.4.11.2.*

Parameters to account for low homology at intron borders

The correct prediction of exact intron borders is one of the most difficult tasks in protein-based gene-prediction, especially those intron borders next to small exons, because their residues might be falsely assigned to neighbouring exons, or when homology is low, as in cross-species applications. Here, divergent residues at intron borders are often not recognized by Blat, or conversely, intronic sequence is falsely assigned to the exon. To deal with the latter case, Scipio cuts off the marginal parts of Blat matches and realigns them. The parameter --max_move_exon allows increasing the default value of six residues that are cut off from the marginal parts. Figure 2.4-5 shows the effect of this parameter in some representative examples (see also Additional file 2.4.11.2). In the case of the human class-19 myosin gene, Blat and the old Scipio version were not able to reconstruct the 5’-end of the gene correctly, because the intron in front of the second exon of the gene ends with the translated sequence LFQ that is very homologous to the real sequence LQQ. Blat added these residues to exon 2 albeit introducing a mismatch. With the new parameter --max_move_exon (default setting is 6), Scipio is now able to resolve this misalignment and to subsequently identify the correct exon 1. Case B shows the reconstruction of the actin capping protein α from *Theileria heterothallica* (Figure 2.4-5, see also Additional file 2.4.11.2). Here, by chance the intergenic region before exon 3 shows some homology to exon 2 (3 matches and 3 mismatches) and thus the exon 2 sequence was erroneously joined to exon 3. This happened irrespectively of lowering the Blat tilesize or adjusting any of the other Scipio parameters. By setting the --max_move_exon to 6 (default setting), the new version of Scipio is now able to correctly reconstruct the CAPα gene.

Parameters to adjust searches on chromosomes or highly fragmented data

Scipio is able to reconstruct genes that are spread on several contigs or supercontigs of highly fragmented genomes. As we have shown, this feature is one of the most important strengths of Scipio (212) that other programs do not offer. However, this feature is not needed in chromosomal assemblies, and might lead, especially in the case of cross-species searches, to composed hits that stretch across multiple chromosomes, one of them being false positive (Figure 2.4-6). Hence, it can be switched off with the parameter --single_target_hits (or --chromosome), which is the default setting when selecting a chromosome assembly as genome target file in WebScipio.

For highly fragmented genomes it is still useful to allow gene reconstructions across several contigs. But also in this case one would want to exclude the assembly of hits that would introduce extremely long introns between exons on different contigs. To accomplish for those cases we have introduced the --max_assemble_size parameter that adjusts the maximum size of intron parts at target boundaries. If an intron would have to be created between two partial hits across two contigs that exceeds the given size (default: 75000 nucleotides), the two hits will not appear together as parts of one composed hit; rather, the lower-scoring contig will be discarded unless --multiple_results is enabled. Alternatively, the parameter --min_dna_coverage can be used to limit the length of introns stretching across contig boundaries, by specifying a minimum query/target length ratio for composed hits, in per cent.

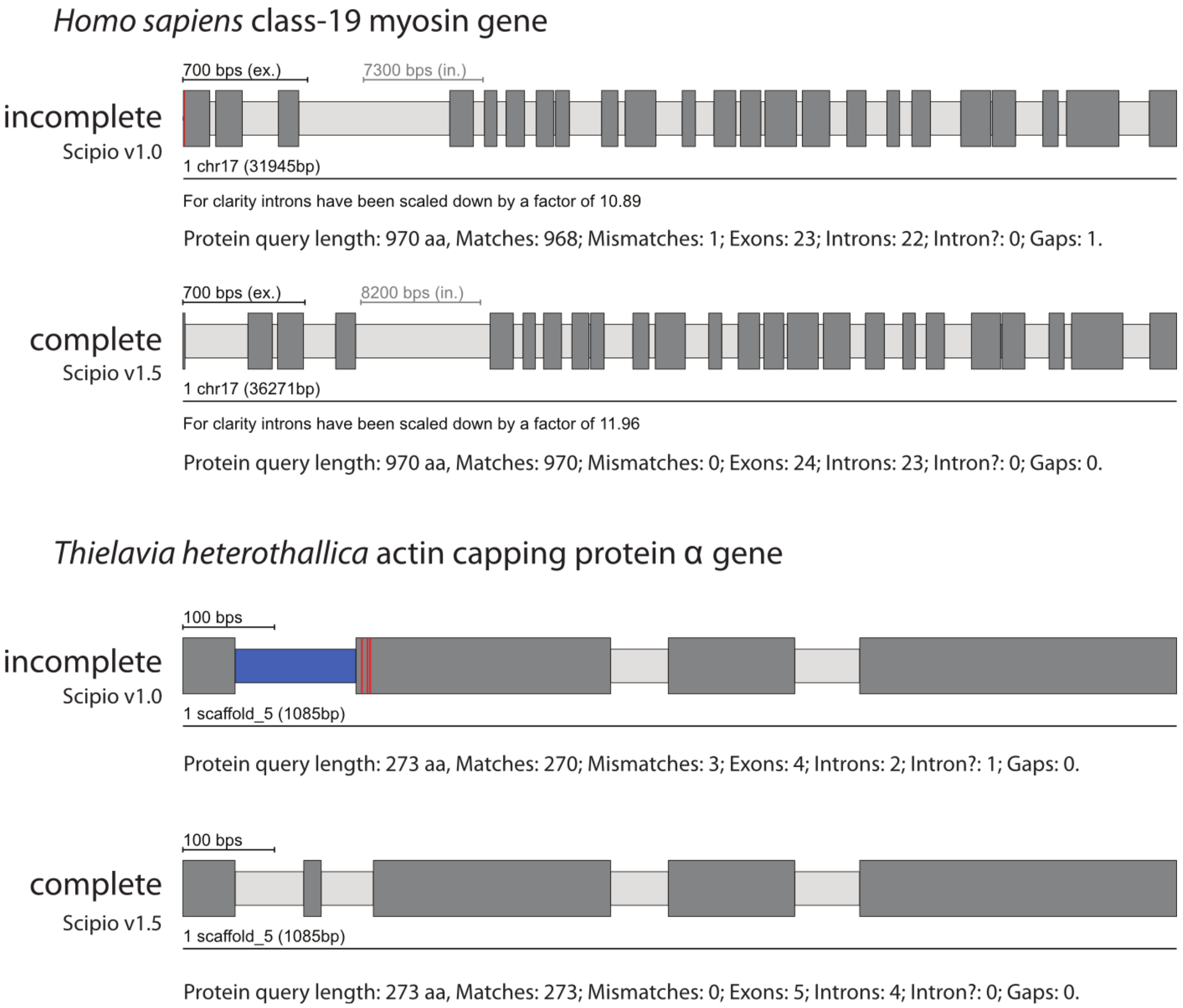

Homo sapiens class-19 myosin gene
700 bps (ex.)
7300 bps (in.)
incomplete
Scipio v1.0
1 chr17 (31945bp)
For clarity introns have been scaled down by a factor of 10.89
Protein query length: 970 aa, Matches: 968; Mismatches: 1; Exons: 23; Introns: 22; Intron?: 0; Gaps: 1.
700 bps (ex.)
8200 bps (in.)
complete
Scipio v1.5
1 chr17 (36271bp)
For clarity introns have been scaled down by a factor of 11.96
Protein query length: 970 aa, Matches: 970; Mismatches: 0; Exons: 24; Introns: 23; Intron?: 0; Gaps: 0.
Thielavia heterothallica actin capping protein α gene
100 bps
incomplete
Scipio v1.0
1 scaffold_5 (1085bp)
Protein query length: 273 aa, Matches: 270; Mismatches: 3; Exons: 4; Introns: 2; Intron?: 1; Gaps: 0.
100 bps
complete
Scipio v1.5
1 scaffold_5 (1085bp)
Protein query length: 273 aa, Matches: 273; Mismatches: 0; Exons: 5; Introns: 4; Intron?: 0; Gaps: 0.

Figure 2.4-5: ***Reconstructing short exons at low homology intron borders.*** *The scheme shows two examples for the reconstruction of short exons in regions where the intron borders of the neighbouring exons show some homology to the unmatched query sequence. The value for the --max_move_exon parameter has been set to 3 (case A) and 6 (case B), respectively. The colour coding of the scheme is the same as in Figure 2.4-3. For further information see Additional file 2.4.11.2.*

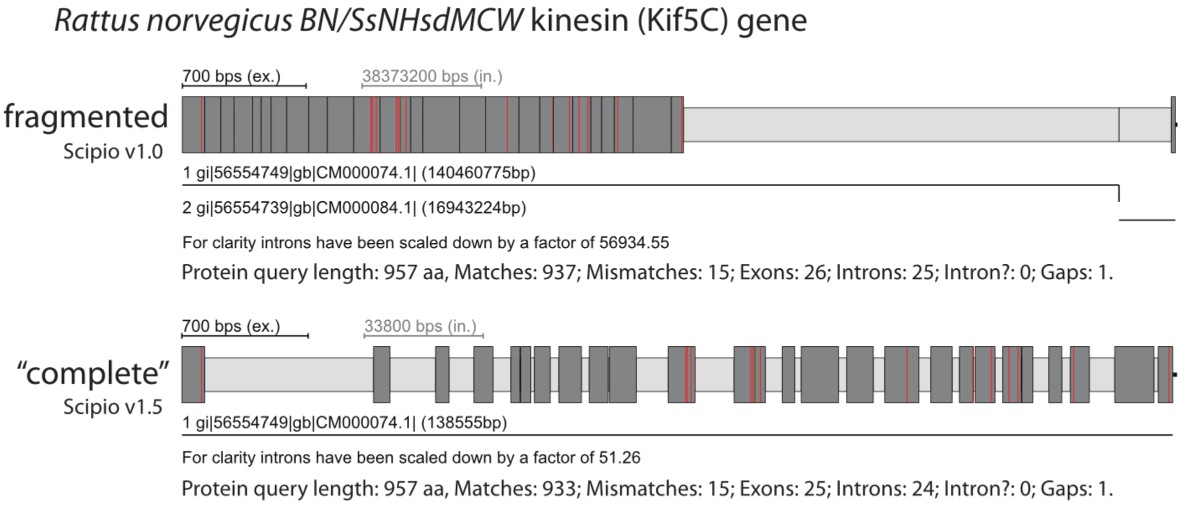

Rattus norvegicus BN/SsNHsdMCW kinesin (Kif5C) gene
fragmented
Scipio v1.0
700 bps (ex.)
38373200 bps (in.)
1 gi|56554749|gb|CM000074.1| (140460775bp)
2 gi|56554739|gb|CM000084.1| (16943224bp)
For clarity introns have been scaled down by a factor of 56934.55
Protein query length: 957 aa, Matches: 937; Mismatches: 15; Exons: 26; Introns: 25; Intron?: 0; Gaps: 1.
"complete"
Scipio v1.5
700 bps (ex.)
33800 bps (in.)
1 gi|56554749|gb|CM000074.1| (138555bp)
For clarity introns have been scaled down by a factor of 51.26
Protein query length: 957 aa, Matches: 933; Mismatches: 15; Exons: 25; Introns: 24; Intron?: 0; Gaps: 1.

Figure 2.4-6: ***Reconstructing genes on chromosome assemblies.*** *The scheme shows an example of the search for the rat homolog (target sequence) of the human Kif5C kinesin motor protein. The C-terminal about 25 amino acids of the rat Kif5C homolog is missing in the respective chromosome assembly. Using Scipio v1.0 a very short identical stretch of four amino acids, found on a different chromosome, has artificially been added to the 3'-end of the gene generating an "intron" of millions of base pairs (Note the scale of the introns!). The new parameter --single_target_hits now prevents this mis-assembly. The colour coding of the scheme is the same as in Figure 2.4-3.*

Improved gene structure reconstruction in cross-species searches

To test the sensitivity and specificity of the new Scipio version we performed a cross-species search of the dynein heavy chain (DHC) genes of *Homo sapiens* in *Loxodonta africana*. The dynein heavy chain genes have been chosen because they belong to the longest genes in eukaryotic genomes and thus contain many exons spread on several hundred thousands of base pairs (Table 2.4.1). In addition, the dynein heavy chain family members show different degrees of identity in mammals and are therefore very suitable to test the limits of Scipio. Afrotheria (to which the elephants belong) and the Euarchontoglires separated about 100 million years ago (299). The DHC query sequence test set and the longer time the species have split up should be a better test for the cross-species search capabilities of Scipio compared to the cross-species search of human myosin heavy chain genes in the mouse genome that we performed earlier (13).

Figure 2.4-7 shows some example results of the cross-species search with genes of decreasing identity. The class-1 dynein heavy chain genes (DHC1) are very conserved between mammals, and the *Loxodonta* DHC1 could perfectly be reconstructed (except for the N-terminus that is not covered in the genome assembly). The DHC4A protein of *Loxodonta* has about 88 per cent identity to the human homolog, and could also completely be reconstructed. In contrast, the DHC9B protein has only about 78 per cent identity to the human homolog and the reconstructed gene still contains several gaps. The figure shows the result of the search using the old Scipio version compared to the result of the search with the new Scipio version. As reference, the result of the manual annotation of the gene is shown. It is very obvious that the new Scipio version provides a dramatically improved reconstruction of the *Loxodonta* DHC9B gene. More than 1,000 additional residues could be mapped corresponding to an increase in completeness by about 25 per cent. The number of reconstructed exons increased from 62 to 80, which is close to the optimally reconstructed number of 85.

Table 2.4.1: Details of the dynein heavy chain genes used for the cross-species search

Protein Name	*Homo* Length [aa]	*Loxodonta* Length [aa]	status*	*Loxodonta* Length [bp]	*Loxodonta* Exons
DHC1	4646	4561	P	62248	77
DHC2	4307	4234	P	413085	89
DHC3A	4707	4690	✓	358204	92
DHC3B	4624	4582	P	298827	78
DHC4A	4507	4508	✓	361115	82
DHC4B	4462	4428	P	115251	79
DHC4C	4486	4339	P	486267	69
DHC5	4589	4584	✓	140290	79
DHC6	4509	4457	P	134640	86
DHC7A	4024	4019	✓	253605	62
DHC7B	4070	3966	P	187156	60
DHC7C	3960	3960	✓	201577	73
DHC8	4265	4064	F	80895	73
DHC9A	4158	4062	P	302568	75
DHC9B	4612	4597	P	369531	85
DHC11	4779	4779	✓	81205	43

* Status: P = partial sequence (short part of the sequence missing); F = sequence fragment (large region of the gene missing in the genome); ✓ = sequence complete.

The diagrams in Figure 2.4-8 show the improvements in gene reconstruction of the new Scipio version compared to the old version for the complete DHC dataset (see also Additional file 2.4.11.3). The reference for the perfectly reconstructed gene is the manual annotation based on the comparative annotation of more than 2,000 dynein heavy chain genes. The basis in diagram A is the reconstruction with Scipio v1.0, and shown are the improvements in the completeness of the annotation with Scipio v1.5 using different search parameters. In general, with the new Scipio version the reconstructions in these cross-species searches could considerably be improved. Lowering the tilesize, a Blat parameter to search with smaller fragments, further improved the results in only two cases. This corresponds to improvements independent of Scipio. However, extending the search frame for the exon search with the Needleman-Wunsch algorithm (parameter --exhaust_align_size) further completed the reconstruction in almost all cases demonstrating the effect of the newly introduced Needleman-Wunsch search for short or divergent exons.

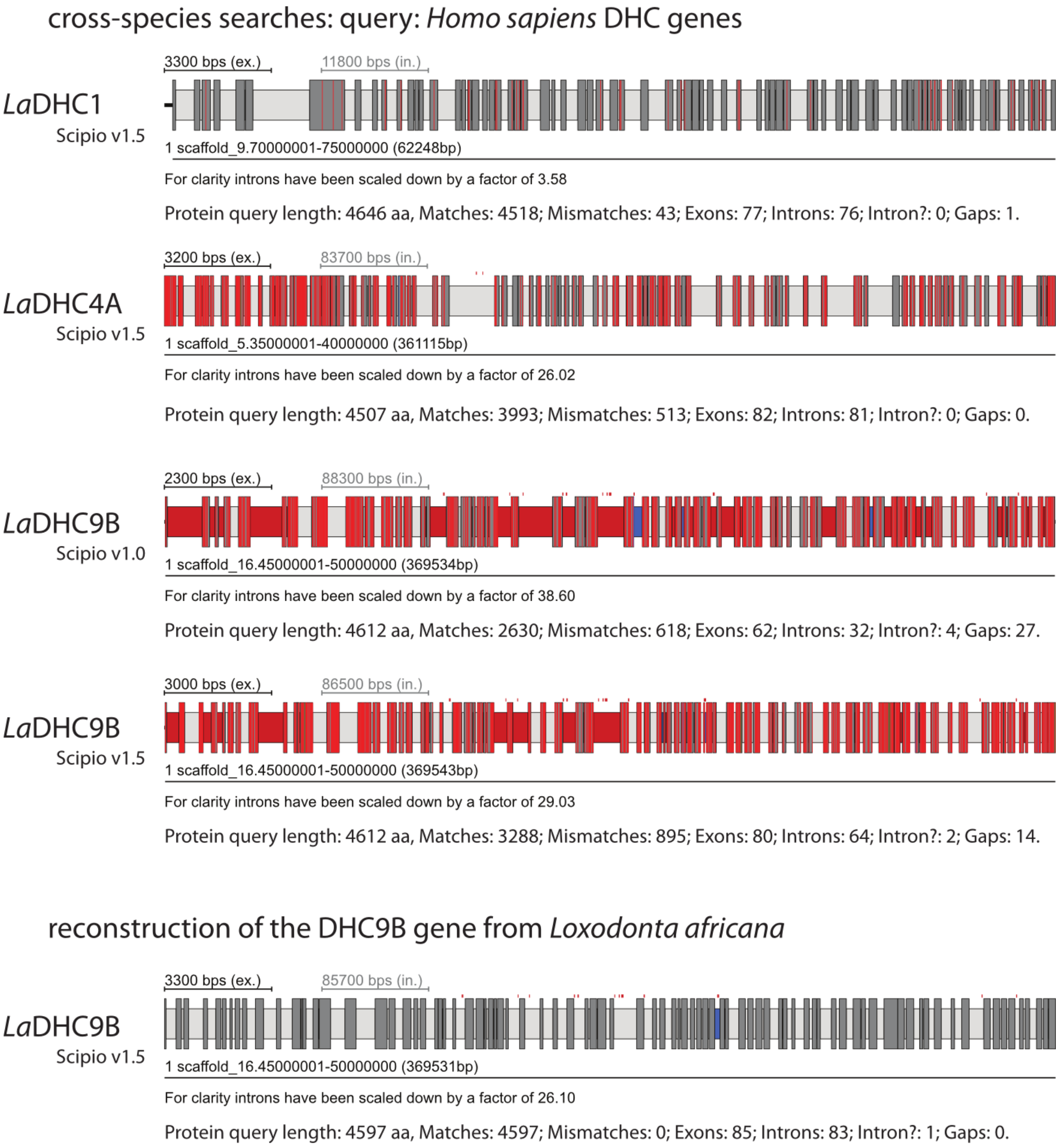

cross-species searches: query: *Homo sapiens* DHC genes
*La*DHC1
Scipio v1.5
3300 bps (ex.)
11800 bps (in.)
1 scaffold_9.70000001-75000000 (62248bp)
For clarity introns have been scaled down by a factor of 3.58
Protein query length: 4646 aa, Matches: 4518; Mismatches: 43; Exons: 77; Introns: 76; Intron?: 0; Gaps: 1.
*La*DHC4A
Scipio v1.5
3200 bps (ex.)
83700 bps (in.)
1 scaffold_5.35000001-40000000 (361115bp)
For clarity introns have been scaled down by a factor of 26.02
Protein query length: 4507 aa, Matches: 3993; Mismatches: 513; Exons: 82; Introns: 81; Intron?: 0; Gaps: 0.
*La*DHC9B
Scipio v1.0
2300 bps (ex.)
88300 bps (in.)
1 scaffold_16.45000001-50000000 (369534bp)
For clarity introns have been scaled down by a factor of 38.60
Protein query length: 4612 aa, Matches: 2630; Mismatches: 618; Exons: 62; Introns: 32; Intron?: 4; Gaps: 27.
*La*DHC9B
Scipio v1.5
3000 bps (ex.)
86500 bps (in.)
1 scaffold_16.45000001-50000000 (369543bp)
For clarity introns have been scaled down by a factor of 29.03
Protein query length: 4612 aa, Matches: 3288; Mismatches: 895; Exons: 80; Introns: 64; Intron?: 2; Gaps: 14.
reconstruction of the DHC9B gene from *Loxodonta africana*
*La*DHC9B
Scipio v1.5
3300 bps (ex.)
85700 bps (in.)
1 scaffold_16.45000001-50000000 (369531bp)
For clarity introns have been scaled down by a factor of 26.10
Protein query length: 4597 aa, Matches: 4597; Mismatches: 0; Exons: 85; Introns: 83; Intron?: 1; Gaps: 0.

Figure 2.4-7: ***Example cross-species searches.*** *The results of four searches with dynein heavy chain sequences from Homo sapiens in the elephant (Loxodonta africana) genome are shown. All genes are spread on several hundred thousands of base pairs. Statistics to the sequence results are given below the gene structure cartoons. An "intron?" is an intron for which the borders do not correspond to the standard splice sites GT---AG or GC---AG. The colour coding of the scheme is the same as in Figure 2.4-3.*

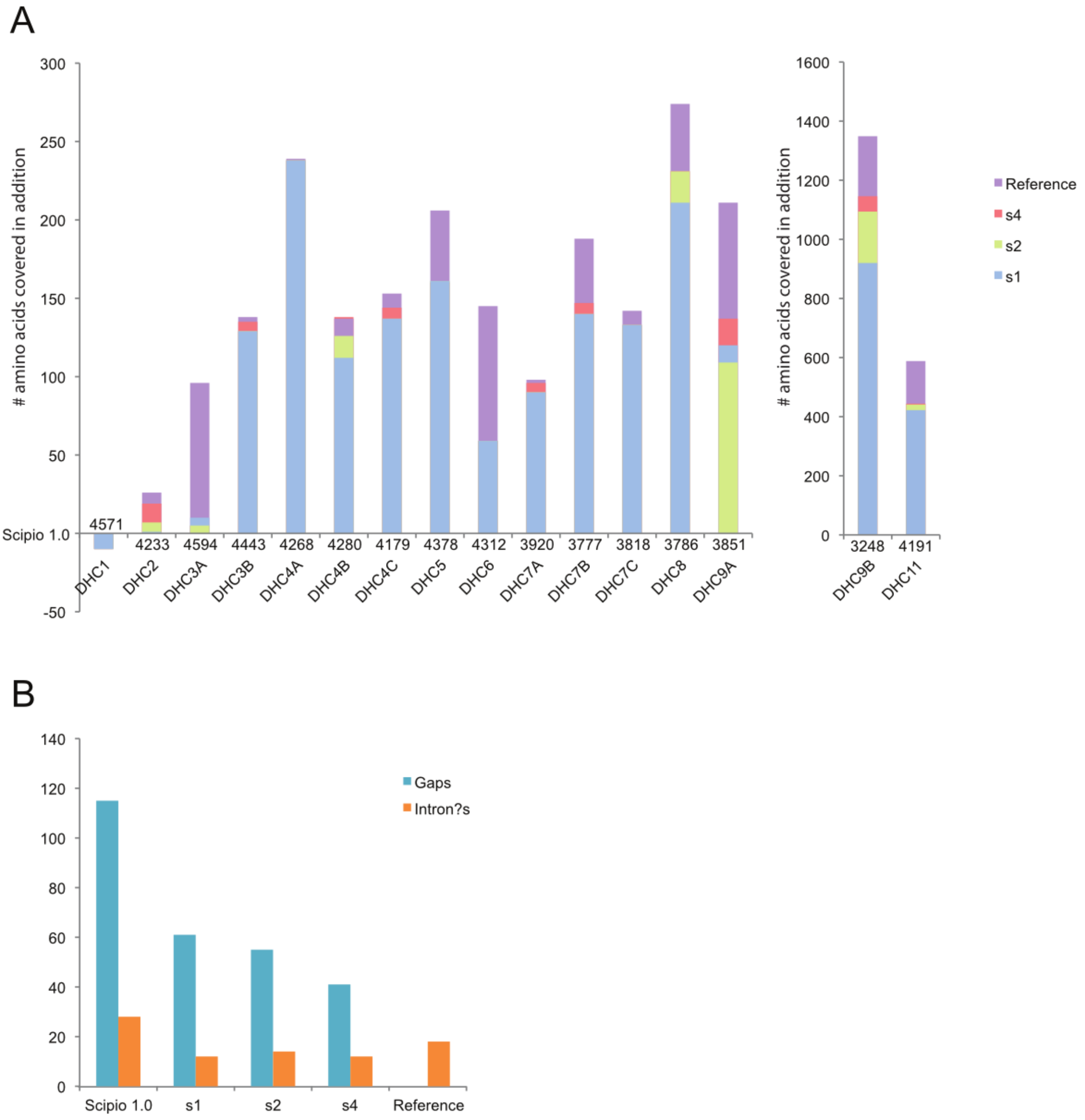


Figure 2.4-8: ***Diagrams of the improvements introduced with the new Scipio version.*** *The diagrams describe the improvement of the gene reconstructions of the DHC genes in the cross-species search of the human homologs (query sequences) in elephant (target sequence) using different Scipio versions and parameters. (A) The base-line is the result of the search using the old Scipio v1.0. The maximal possible annotation is represented by the gene reconstructions based on the manually annotated elephant DHC genes (reference dataset, purple). The blue bars show the reconstruction with Scipio v1.5 using --blat_tilesize=7, --exhaust_align_size=500 and --exhaust_gap_size=21 (dataset s1). Green bars are results from the second search (dataset s2) with same parameters as for the first search, except for --blat_tilesize=6 and --exhaust_gap_size=18 (three times the tilesize). This dataset represents improvements independent of Scipio. The red bars represent searches with same parameters as for dataset s1, except for the increased parameters --exhaust_align_size=5,000 and --exhaust_gap_size=25 (dataset s4). This data takes far longer to compute compared to the first search, because of the Needleman-Wunsch search in longer regions. For the DHC1 gene Scipio v1.0 maps too many amino acids of the human query sequence to the elephant genome. So the negative bar representing the other datasets shows that these datasets cover the right number of 4561 amino acids. (B) This diagram depicts the number of gaps (human query sequence not matched in the elephant genome) and questionable introns (intron?; introns with uncommon splice sites) for the searches with the old Scipio version and the new version applying different parameters as in (A). The detailed values of the diagrams are shown in tables in Additional file 2.4.11.3.*

Comparison of gene reconstruction and prediction tools

We compared Scipio to other tools that reconstruct and predict genes based on a protein sequence, and to general gene prediction tools. The tools can be ordered in three categories. The tools of the first category reconstruct the exon-intron structure of the protein-coding genes based on a genomic sequence and a provided protein sequence. Scipio (212), Prosplign (249), Exonerate (300), and Prot_map (301) belong to this category. The second category includes the tools Fgenesh+ (302), GeneWise/Wise2 (303), and GenomeScan (304), that combine homology based gene reconstructions taking advantage of given protein sequences, and *ab initio* gene prediction approaches. The third group of software packages consists of *ab initio* gene prediction tools like Augustus (305), Fgenesh (302), and Genscan (306). The latter tools are not really comparable with the other ones in the task of reconstructing single genes, but the comparison illustrates the differences of *ab initio* and homology based gene predictions. In addition to Blat, we tested Blast (77), which can also be used as an initial search for the Prosplign tool. However, for our test cases this approach did not improve the results of Prosplign (see Additional file 2.4.11.4).

To evaluate the performance of Scipio in comparison to the other tools, four test scenarios have been designed. The DHC proteins have been chosen as a large general test set, while the other examples used for the explanation of the new Scipio parameters have been used as a test set for genes difficult to reconstruct. Both test data sets have been explored in reconstructions/predictions out of whole genome assemblies and respective gene regions. This differentiation has been done because only a few of the above-mentioned tools could be used in searches against whole genomes due to the limited upload possibilities of the respective web-interfaces while command-line versions of the tools were not available for every software. Thus we tested the performance of all tools against the gene regions of the test data that correspond to the nucleotide sequence of the reference annotation plus 2,000 additional base pairs up- and downstream. To make the execution times comparable, the genome wide runs were performed on a dedicated server, which contains four 2.2Ghz AMD Opteron 6174 processors, with 12 cores each, and 128GByte of memory.

Scenario 1

In the first scenario, the tools had to reconstruct the dynein heavy chain genes in the whole *Loxodonta africana* genome assembly based on the human protein sequences (Table 2.4.2).

Table 2.4.2: Test scenario 1: Reconstruction of the Loxodonta africana dynein heavy chain genes in the whole genome sequence based on human protein sequences

Tool	Predicted genes	Missing exons[1]	Wrong exons[2]	Exon sens. %	Exon sens. (ov.)[3] %	Exon spec. %	Nucl. sens. %	Nucl. spec. %	Execution time per prot. seq.
Scipio 1.5[4]	16	11	6	93.4	99.1	93.3	98.7	99.8	70m 46s
Exonerate[5]	2145	6	5669	94.8	99.5	16.5	99.7	18.5	123m 23s
Exonerate[6]	16	287	62	73.4	76.1	90.2	76.1	94.3	121m 27s
Augustus[7]	1374928	0	3909434	47.9	100.0	0.0	100.0	0.3	> 10 days
BLAT[8]	-	9	264228	19.6	99.3	0.1	97.4	2.6	7m 24s
Scipio 1.0[4]	16	14	46	86.1	98.8	83.2	97.9	99.4	8m 24s

[1] Number of annotated exons, which are not overlapped by any predicted exon

[2] Number of predicted exons, which are not overlapped by any annotated exon

[3] Number annotated exons, which are overlapped by at least one predicted exon divided by the number of annotated exons

[4] Mammalia cross species default options (for detailed parameters see Additional file 2.4.11.5)

[5] Parameters: --model protein2genome

[6] Parameters: --model protein2genome --bestn 1

[7] Parameters: --species=human --genemodel=exactlyone (for more parameters see Additional file 2.4.11.5)

[8] Parameters as in [4]: -tileSize=7 -minIdentity=54 -minScore=15 -oneOff=1

Besides Scipio, only Exonerate and Augustus were able to produce reasonable results. Prot_map, Fgenesh, and Fgenesh+ could not be tested in this scenario because the command-line versions are proprietary and it is not possible to upload whole genome sequences via their web-interfaces. WebScipio is the only tool available, which already provides genome sequences. The dynein heavy chain genes contain 1,202 annotated exons including 209,486 nucleotides. The *Loxodonta africana* genome contains 3,271,792,967 nucleotides including N’s. For the DHC1 gene the N-terminus cannot be found in the genome sequence because of a gap in the genome assembly. We adjusted the start of the first known exon in the reference annotation to the predicted exon for each tool, because the start depends on whether a tool found an exon in front of the first known exon. The results of the first test scenario are presented in Table 2.4.2 (for more data see Additional file 4). Both Scipio and Exonerate in the standard mode are comparable in exon sensitivity (93.4% and 94.8%, respectively) and missed a similar amount of exons (11 exons and 6 exons, respectively). However, Exonerate predicted many wrong exons (5669 exons) resulting in a low specificity (16.5%, compared to 93.3% exon specificity by Scipio). Exonerate can be configured to report only the best hit by setting the --bestn option to 1. While this option increased the specificity (from 16.5% to 90.2%), the sensitivity decreased (from 94.8% to 73.4%). Also, the number of missing exons increased to 287.

Comparing the results of Scipio and Blat illustrates that Blat found almost all exons, but that Scipio is needed to refine the exon borders as well as to exclude hits not related to the query sequence. Using the new Needleman-Wunsch algorithm Scipio v1.5 closes many gaps by adding and extending exons to the hits found by Blat. The number of missing exons is lower in Blat (9 exons missing) than in Scipio (11 exons missing), because Blat maps parts of the protein sequence to the genomic sequence, although these hits are not in the same order as in the protein sequence. Scipio excludes these hits. The results also show the great improvement of Scipio v1.5 compared to Scipio v1.0 in sensitivity (93.4% and 86.1%, respectively) and specificity, (93.3% and 83.2%, respectively). Altogether, these results show that Scipio v1.5 is the only free tool that is able to reconstruct the genes nearly complete in this scenario.

Scenario 2

The results of the second scenario are shown in Table 2.4.3.

All above-mentioned tools were compared except for GenomeScan. Although GenomeScan produced results with the data provided on the respective webpage it did not work with our protein examples. The data show that Scipio performed in the same range as the other tools with respect to sensitivity and specificity. Scipio, Prospligm, and Exonerate revealed the highest sensitivity (94.7%, 95.7%, and 94.8%, respectively). Although Prosplign missed only one exon it also mis-predicted 41 exons. The homology based *ab initio* tools Fgenesh+ and Wise2 also provided almost complete reconstructions. Especially Fgenesh+ achieved high and balanced values for sensitivity (94.9%) and specificity (94.8%). The number of predicted genes illustrates that Exonerate without the --bestn option and Wise2 tend to divide long genes (32 and 39 genes predicted, respectively, instead of 16). The *ab initio* tools did not show comparable performance to the other tools in this scenario resulting in sensitivities of 76 – 82% and specificities of 58 – 83%. Augustus outperforms Fgenesh and Genscan with (Table 2.4.3) or without (see Additional file 2.4.11.4) the option to predict exactly one gene. Augusuts with the restriction to predict exactly one gene resulted in more accurate reconstructions. As in the whole genome scenario, the new Scipio v1.5 (93.1% sensitivity and 93.1% specificity) provides far better gene predictions than Blat and Scipio v1.0 (sensitivity of 19.9% and 86.2%, and specificity of 19.4% and 85.9%, respectively).

Table 2.4.3: Test scenario 2: Reconstruction of the Loxodonta africana dynein heavy chain gene structures in the respective gene regions based on human protein sequences

Tool	Pred. genes	Missing exons[1]	Wrong exons[2]	Exon sens. %	Exon sens. (ov.)[3] %	Exon spec. %	Nucl. sens. %	Nucl. spec. %
Scipio 1.5[4]	16	13	6	93.1	98.9	93.1	98.6	99.8
Scipio 1.5[5]	16	4	7	94.7	99.7	93.7	99.2	99.8
Prosplign[6]	16	1	41	95.7	99.9	92.6	99.9	98.7
Exonerate[7]	32	7	6	94.8	99.4	94.6	99.6	99.5
Exonerate[8]	16	255	4	75.7	78.8	95.6	79.2	99.7
Prot_map[9]	16	4	27	91.7	99.7	86.2	99.3	99.7
Fgenesh+[10]	16	10	10	94.9	99.2	94.8	99.0	99.7
Wise2[11]	39	3	16	93.3	99.8	91.2	99.7	98.9
Augustus[12]	16	132	111	81.9	89.0	83.2	89.9	88.7
Fgenesh[10]	161	111	342	80.2	90.8	67.3	91.8	62.3
Genscan[13]	194	138	520	76.3	88.5	57.9	90.4	55.3
BLAT[14]	-	16	19	19.9	98.7	19.4	97.0	98.9
Scipio 1.0[4]	16	16	10	86.2	98.7	85.9	97.8	99.8

[1] Number of annotated exons, which are not overlapped by any predicted exon

[2] Number of predicted exons, which are not overlapped by any annotated exon

[3] Number annotated exons, which are overlapped by at least one predicted exon divided by the number of annotated exons

[4] Mammalia cross species default options (for detailed parameters see Additional file 2.4.11.5)

[5] Mammalia cross species default options; -tileSize=6 (for detailed parameters see Additional file 2.4.11.5)

[6] Parameters: -full -two_stages

[7] Parameters: --model protein2genome

[8] Parameters: --model protein2genome --bestn 1

[9] Similarity: Weak; Search for one best alignment only (for more parameters see Additional file 2.4.11.5)

[10] Organism: Human

[11] Parameters: -both

[12] Parameters: --species=human --genemodel=exactlyone (for more parameters see Additional file 2.4.11.5)

[13] Organism: Vertebrate; Suboptimal exon cutoff: 1.00

[14] Parameters as in [4]: -tileSize=7 -minIdentity=54 -minScore=15 -oneOff=1

Scenario 3 and 4

In the third and fourth scenario we compared the tools in their performance to reconstruct the difficult cases, which we introduced above by describing the new parameters of Scipio v1.5. In scenario 3 a search in the whole genome and in scenario 4 a search in the respective gene regions (as in scenario 2) was performed. Table 2.4.4 summarizes the results of the third and fourth scenario. Only when using the latest version of Scipio the genes of the test data set could correctly be reconstructed and predicted in the whole genome assemblies as well as in the gene region. None of the other tools was able to reconstruct all genes correctly, even if the gene region was given as in the fourth scenario.

Table 2.4.4: Test scenario 3 and 4: Difficult cases for reconstruction of gene structures

Tool	Ned kinesin	Phs dynactin p62	Hs dynactin p50	Pug coronin	Mm dynactin p150	Hs myosin	Th CAPα
Scipio 1.5[1]	✓	✓	✓	✓	✓	✓	✓
Prosplign[2]	O	o	✓	–	✓	–	✓
Exonerate[3]	O	o	–	–	✓	–	–
Prot_map[4]	O	o	–	–	–	o	–
Fgenesh+[5]	O	o	–	–	–	o	o
Wise2[6]	O	o	–	–	–	–	–
Augustus[7]	O	o	–	–	–	–	–
Fgenesh[5]	O	o	–	–	–	–	–
Genscan[8]	O	o	–	–	–	–	–
BLAT[9]	O	o	–	–	–	–	–
Scipio 1.0[1]	O	o	–	–	–	–	–

Ned: *Neurospora discreta*, Phs: *Phytophthora sojae*, Hs: *Homo sapiens*, Pug: *Puccinia graminis*, Mm: *Mus musculus*, Th: *Thielavia heterothallica*

✓	All exons are reconstructed correctly
o	All annotated exons are matched by or overlap with predicted exons
–	Exons are missing

[1] Ned, Phs: cross species default options; Hs, Mm: default options, exhaust_align_size=15000; Pug, Th: default options (for detailed parameters see Figure 2.4-3, Figure 2.4-4, and Figure 2.4-5 and Additional file 2.4.11.5)

[2] Parameters: -full -two_stages

[3] Parameters: --model protein2genome

[4] Similarity: Weak; Search for one best alignment only (for more parameters see Additional file 5)

[5] Organisms: Ned, Th: Neurospora crassa; Phs: Phytophtora; Hs: Human; Pug: Puccina; Mm: Mouse

[6] Parameters: -both

[7] Parameters: --genemodel=exactlyone; Organisms: Ned: --species=neurospora; Phs, Pug, Th: --species=generic; Hs, Mm: --species=human (for more parameters see Additional file 2.4.11.5)

[8] Organism: Vertebrate; Suboptimal exon cutoff: 1.00

[9] Parameters: -minScore=15; Ned, Phs: -tileSize=5 -minIdentity=54 -oneOff=1; Hs, Pug, Mm, Th: -tileSize=7 -minIdentity=81

2.4.5 Conclusions

Scipio and its graphical web-interface WebScipio are tools for the reconstruction of gene structures in eukaryotes. Scipio is based on the widely used program Blat that has been developed for aligning sequences of very high similarity. However, for the correct reconstruction of intron splice sites, very short exons, genes spread on several contigs, and the handling of sequencing errors a lot of post-processing is required. This is done by Scipio. Here, we present the fundamentally updated versions of Scipio and WebScipio, with an improved reconstruction of very short exons and intron splice sites, especially for the case of cross-species searches. To this end, we introduced a version of the Needleman-Wunsch algorithm that was shown to find a higher number of short exons previously missed, and to correct intron boundaries, especially in cases of lower sequence similarity. Furthermore, gaps in the mapping are now more frequently explained by divergent sequences, allowing for longer regions of insertions or deletions predicted on the same exon. Several parameters were introduced that can be used to fine-tune this behaviour if necessary. The sequence similarity between query and target sequence decreases with increasing evolutionary distance. While Blat is in principle able to locate hits for more distant species, the results become more and more incomplete, raising the importance of the post-processing. We could show that Scipio is now able to almost completely reconstruct genes from species whose ancestors separated more than 100 Myr ago. WebScipio allows easy access to Scipio and genome assemblies of about 640 eukaryotic species. This is unique to all gene reconstruction/prediction tools available and allows easy identification and reconstruction of protein homologs in related organisms. We compared the performance of Scipio to many other tools using our test data. While there are only minor differences in the reconstruction of the mammalian dynein heavy chain genes between Scipio, Exonerate, Prosplign, and Fgenesh+, the other software tools were not able to correctly reconstruct the more difficult cases encoding very short exons and showing strong sequence divergence at intron borders or inside of exons. Also unique to Scipio, this is the only tool available that is able to correctly reconstruct and predict genes that are spread on several contigs.

2.4.6 Availability and requirements

Project name: WebScipio, Scipio
Project home page: http://www.webscipio.org
Operating system: Platform independent
Programming languages: Ruby, Perl

Software requirements: Installation of Blat and BioPerl for using Scipio as command-line tool. WebScipio has been tested with InternetExplorer, Firefox, Chrome, Safari, and Opera.

License: WebScipio and Scipio may be obtained upon request and used under a GNU General Public License.

Any restrictions to use by non-academics: Using WebScipio and Scipio by non-academics requires permission.

2.4.7 List of abbreviations

Blat: BLAST like alignment tool; FTP: File transfer protocol; HTML: Hypertext markup language; SVG: Scalable vector graphics; PNG: Portable network graphics; YAML: YAML ain’t markup language

2.4.8 Competing interests

The authors declare that they have no competing interests.

2.4.9 Authors' contributions

KH and MK set the requirements for the system and wrote the manuscript. KH and BH wrote the WebScipio software. HP implemented the test environment. OK wrote the Scipio source code and assisted in writing the manuscript. SW supervised the implementation of Scipio. KH, HP, OK, and MK performed extensive testing. KH performed the comparative software analysis. All authors read and approved the final version of the manuscript.

2.4.10 Acknowledgements and Funding

MK has been funded by grants KO 2251/3-1, KO 2251/3-2, and KO 2251/6-1, and SW by grant WA 766/6-1 of the Deutsche Forschungsgemeinschaft. We thank Florian Odronitz for helpful discussions and support, and all the known and unknown users of WebScipio for their testing and feedback.

2.4.11 Additional files

2.4.11.1 Additional file 1 – Activity flow of the hit processing step

The scheme shows a detailed activity flow of the hit processing step. Here, the experienced user can see, where and how the various expert parameters modulate Scipio’s hit processing, and can thus adjust these parameters to get the best result possible.

The file can be found in the corresponding publication.

2.4.11.2 Additional file 2 – Protein – DNA alignments corresponding to the example searches

Here, additional data corresponding to the example searches is provided.

The file can be found in the corresponding publication.

2.4.11.3 Additional file 3 – Table with detailed data of the results of the cross-species search of the human DHC genes in the elephant genome.

The table provides detailed data to the cross-species searches including numbers of matches and mismatches, gaps and intron?’s, for the searches with different parameters.

The file can be found in the corresponding publication.

2.4.11.4 Additional file 4 – Detailed evaluation values used for Tables 2, 3, and 4

This file provides a description of each evaluation parameter and the values obtained with each software tool for all sequence predictions. The values highlighted in yellow were used for Table 2.4.2, Table 2.4.3, and Table 2.4.4.

The file can be found in the corresponding publication.

2.4.11.5 Additional file 5 – Software versions and run parameters of the gene reconstruction and prediction tools

The tables shows the exact versions and run parameters, which were used for the comparison, for each scenario.

The file can be found in the corresponding publication.

2.5 Predicting mutually exclusive spliced exons based on exon length, splice site and reading frame conservation, and exon sequence homology

Holger Pillmann[*], Klas Hatje[*], Florian Odronitz, Björn Hammesfahr, and Martin Kollmar[1§]

Abteilung NMR basierte Strukturbiologie, Max-Planck-Institut für Biophysikalische Chemie, Am Fassberg 11, D-37077 Göttingen, Germany

[*] These authors contributed equally to the work.
[§] Corresponding author

BMC Bioinformatics Highly accessed
Published: 30 June 2011
BMC Bioinformatics 2011 12:270 doi:10.1186/1471-2105-12-270 This article is available from http://www.biomedcentral.com/1471-2105/12/270

2.5.1 Abstract

Background

Alternative splicing of pre-mature RNA is an important process eukaryotes utilize to increase their repertoire of different protein products. Several types of different alternative splice forms exist including exon skipping, differential splicing of exons at their 3'- or 5'-end, intron retention, and mutually exclusive splicing. The latter term is used for clusters of internal exons that are spliced in a mutually exclusive manner.

Results

We have implemented an extension to the WebScipio software to search for mutually exclusive exons. Here, the search is based on the precondition that mutually exclusive exons encode regions of the same structural part of the protein product. This precondition provides restrictions to the search for candidate exons concerning their length, splice site conservation and reading frame preservation, and overall homology. Mutually exclusive exons that are not homologous and not of about the same length will not be found. Using the new algorithm, mutually exclusive exons in several example genes, a dynein heavy chain, a muscle myosin heavy chain, and Dscam were correctly identified. In addition, the algorithm was applied to the whole *Drosophila melanogaster* X chromosome and the results were compared to the Flybase annotation and an *ab initio* prediction. Clusters of mutually exclusive exons might be subsequent to each other and might encode dozens of exons.

Conclusions

This is the first implementation of an automatic search for mutually exclusive exons in eukaryotes. Exons are predicted and reconstructed in the same run providing the complete gene structure for the protein query of interest. WebScipio offers high quality gene structure figures with the clusters of mutually exclusive exons colour-coded, and several analysis tools for further manual inspection. The genome scale analysis of all genes of the *Drosophila melanogaster* X chromosome showed that WebScipio is able to find all but two of the 28 annotated mutually exclusive spliced exons and predicts 39 new candidate exons. Thus, WebScipio should be able to identify mutually exclusive spliced exons in any query sequence from any species with a very high probability. WebScipio is freely available to academics at http://www.webscipio.org.

2.5.2 Background

Eukaryotes can enhance their repertoire of different protein products by alternative splicing of the corresponding genes (307). Since the first description of alternative splicing of precursor mRNA almost 30 years ago (308,309) the suggested and verified percentage of human genes that are spliced into alternative transcripts has steadily risen (for reviews see for example (22,310)). Very recently, two studies using high-throughput sequencing indicate that every single human gene containing more than one exon is transcribed and processed to yield multiple mRNAs (15,311).

Mainly, five different types of alternative splicing affect the resulting translated protein product (14,312,313): The first type is exon skipping, in which an exon, also called cassette exon, is spliced out of the transcript together with its flanking introns. The second and third types are the alternative splicing of the 3' splice site and 5' splice site, respectively. Here, two or more splice sites are recognized at one end of the exon. The fourth type is intron retention in which part of an exon is either spliced (like a regular intron) or retained in the mature mRNA transcript. While exon skipping and alternative 3' splice site selection account for most alternative splicing events in higher eukaryotes (16,17), the most prevalent type of alternative splicing in plants, fungi, and protozoa is intron retention (18). The fifths type is called mutually exclusive splicing and is used for clusters of internal exons that are spliced in a mutually exclusive manner. It is important to note that the term mutually exclusive splicing is only used for these specific clusters of exons. Mutually exclusive splicing demands a specific mechanism for the regulated splicing of exactly one of the exons of such a cluster. Recent analyses have shown that this mechanism might be based on intra-intronic RNA pairings that are conserved at the secondary structure level (314–316). These alternatively spliced exons must not be mixed up with exons that seem to be spliced in a mutually exclusive manner based on their annotation. This especially accounts for terminal exons that are alternatively spliced in conjunction with the use of alternative promoters or 3’-end processing sites (for a review see for example (317)). The regulation of the splicing of these types need not be at the level of splicing.

To our knowledge, the only study to identify and predict regions *in silico* that might contain mutually exclusive spliced exons used a method of local similarity of genomic regions at the nucleotide level (318). Assuming that clusters of mutually exclusive exons evolved by one or several rounds of single-exon duplications, given gene locations were self-aligned using a pairwise local alignment algorithm to derive similar regions. Those regions were regarded as candidate regions, and mutually exclusive exons were only predicted by verification through EST and cDNA data. The method itself cannot determine exons including intron splice sites, and is not able to identify mutually exclusive exons whose DNA sequences have diverged considerably. False positive candidates are detected in regions that contain clusters of duplicated genes, and in regions containing pseudo-exons (e.g. exons that are in the process of being lost containing frame-shifts and in-frame stop codons, and missing correct splice sites).

Here, we propose a different approach that is based on the knowledge of creating meaningful transcripts. We presume that most mutually exclusive exons encode the same region of the resulting protein structure. These regions are embedded in the surrounding three-dimensional structure and thus alternative exons must preserve all structurally important contacts between the corresponding local structure elements. A demonstrative example is the alternatively spliced motor domain of the muscle myosin heavy chain in arthropods (24). In *Drosophila*, four clusters of mutually exclusive spliced exons encode regions of the motor domain, and the variability of creating different transcripts and further fine-tune the motor domain function is even enhanced in the waterflea *Daphnia magna* by four additional clusters. One of the clusters contains exons encoding the so-called relay helix and subsequent relay loop, a structural element that starts at switch-2 embedded in the middle of the motor domain and ends at the connection to the converter domain. This whole relay element converts small conformational changes at the ATP-binding site to large movements of the lever arm (32). Retaining structural integrity is therefore indispensable for mutually exclusive exons. Of course, parts of the exons might also encode loop regions, but also those parts must at least partly be conserved to retain their general function.

Based on these preconditions we apply the following constrains to our search for mutually exclusive exons: A) Mutually exclusive exons must have about the same length (allowing some length difference for e.g. parts encoding loop regions). B) They must have conserved splice site patterns (e.g. a GT 5’ intron splice site cannot be combined with an AC 3’ splice site) and the reading frame of the exon must be conserved. C) They must show sequence similarity. These features have been implemented in an extension to the WebScipio software. The application of the algorithm to various genes from several eukaryotes, and to all genes of the X chromosome of *Drosophila melanogaster* is demonstrated.

2.5.3 Methods

The search algorithm has been implemented as an extension to the WebScipio web application (13). It is based on the exon-intron gene structure reconstructed by Scipio (212). The extension is written in the Ruby programming language (78) and fully integrated into WebScipio to facilitate user interaction, and visualization and analysis of the results. WebScipio uses the web framework Ruby on Rails (79). To make the session storage fast, flexible, and scalable a database backend consisting of Tokyo Cabinet and Tokyo Tyrant (293) is used. To run jobs in background the Rails plug-in Workling in combination with Spawn (291,292) is applied.

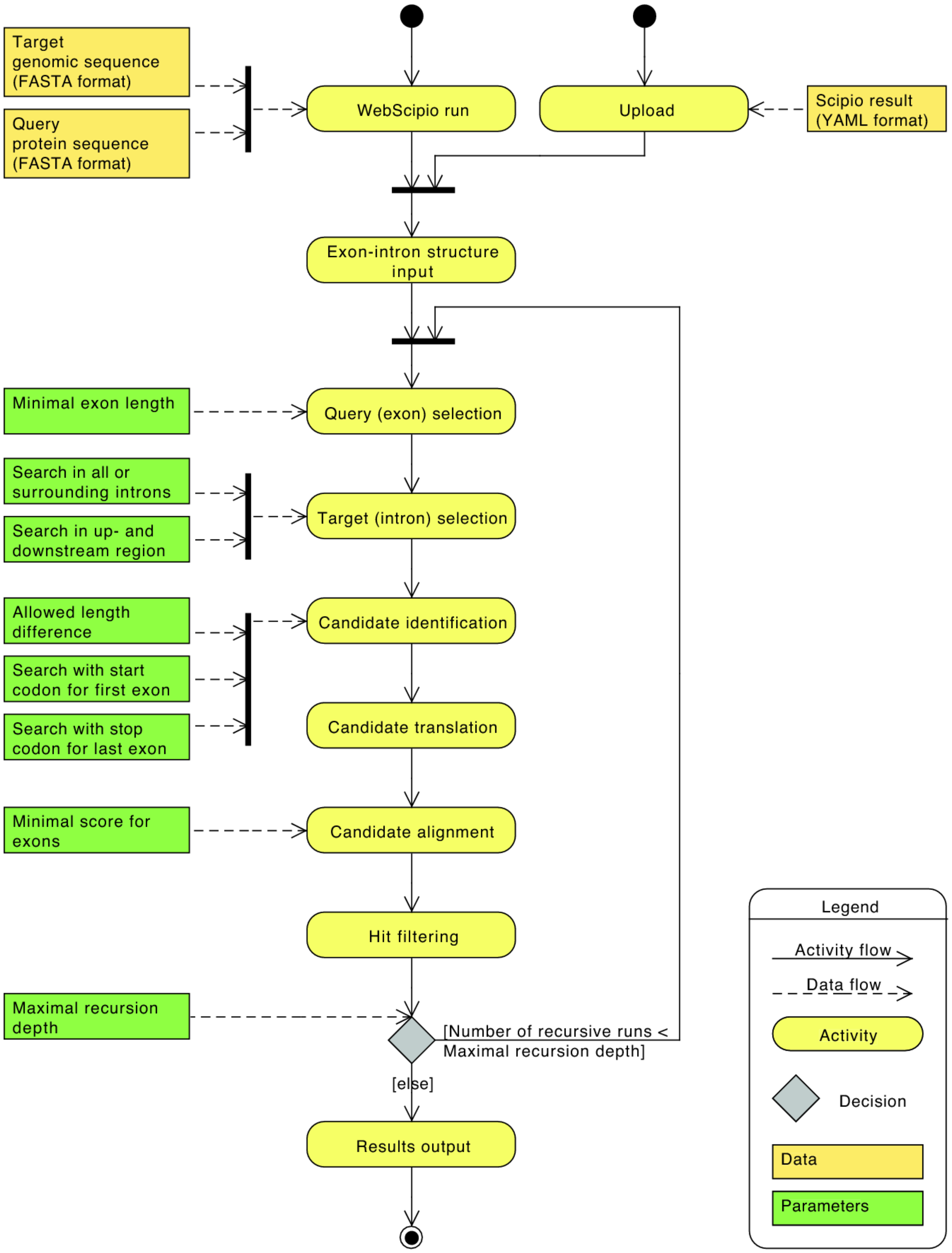

Target genomic sequence (FASTA format)
Query protein sequence (FASTA format)
WebScipio run
Upload
Scipio result (YAML format)
Exon-intron structure input
Minimal exon length
Query (exon) selection
Search in all or surrounding introns
Search in up- and downstream region
Target (intron) selection
Allowed length difference
Search with start codon for first exon
Search with stop codon for last exon
Candidate identification
Candidate translation
Minimal score for exons
Candidate alignment
Hit filtering
Maximal recursion depth
[Number of recursive runs < Maximal recursion depth]
[else]
Results output
Legend
Activity flow
Data flow
Activity
Decision
Data
Parameters

Figure 2.5-1: ***Activity diagram of the search algorithm.*** *The activity diagram shows the processing steps of the search algorithm and the influence of the parameters on each step. The run starts with an exon-intron gene structure determined by Scipio. Based on the chosen parameters the exons and corresponding introns are selected and searched for mutually exclusive spliced exon candidates. The candidates are processed and filtered. These steps are repeated in the case of a recursive run. In the end, the algorithm outputs the exon-intron structure including mutually exclusive spliced exons.*

Search algorithm

The new algorithm divides into several steps, which are executed for each original exon (Figure 2.5-1, a detailed activity diagram is available as Additional file 2.5.9.1). It assumes that mutually exclusive spliced exons share the following features: Firstly, mutually exclusive spliced exons have a similar length; secondly, their splice sites and reading frames are conserved; thirdly, they are homologous.

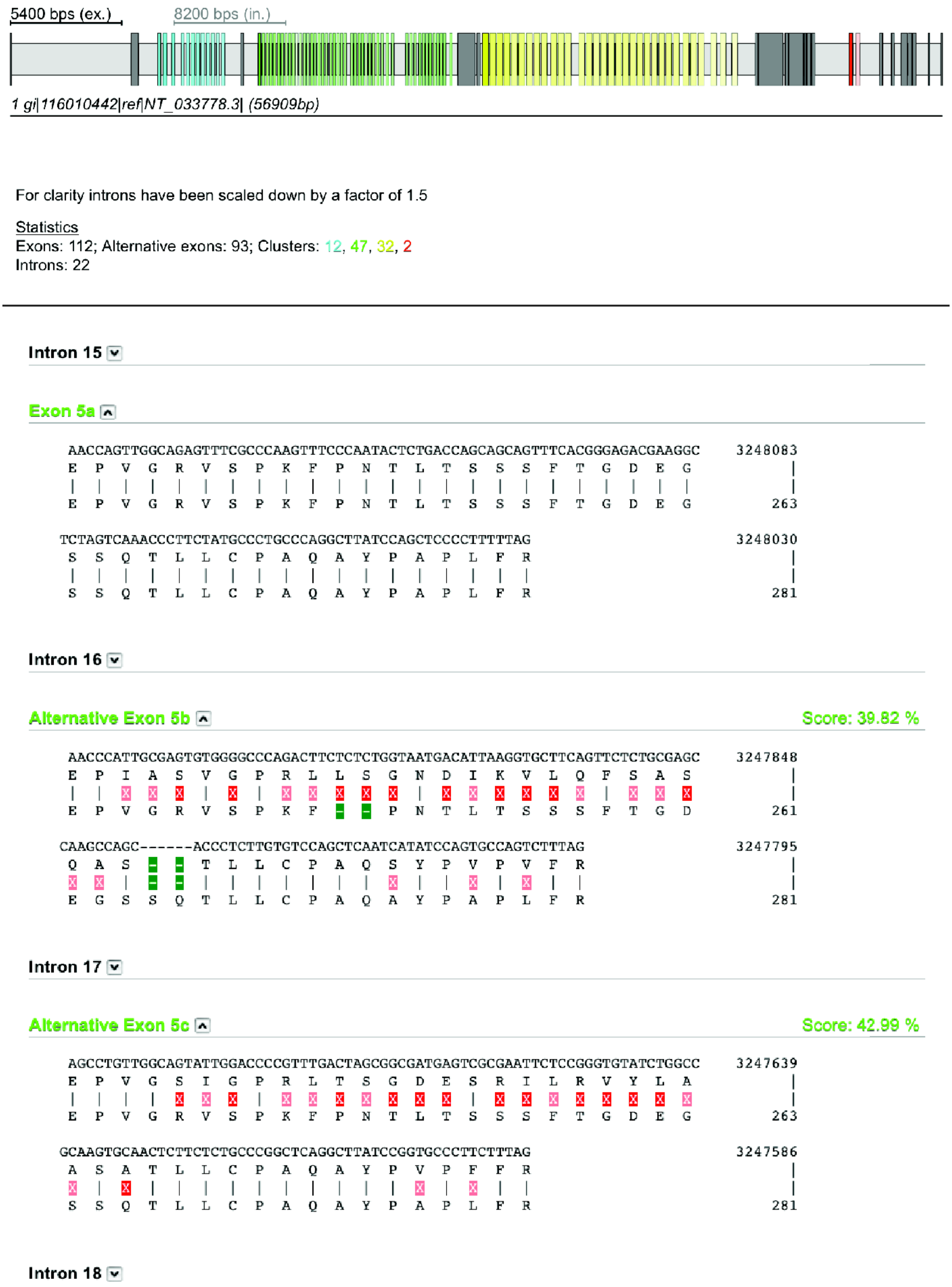
For each internal exon ("original exon") the two surrounding introns (or optionally all introns of the gene) are scanned for exon candidates that have a similar length. These exon candidates must introduce introns with the following splice site pattern: GT---AG, GC---AG, GG---AG, and AT---AC. Firstly, the algorithm looks for the nucleotide pairs AG or AC in the intron sequence, which define start sites of exon candidates and 3’ splice sites of the proposed intron. If the intron in front of the original exon starts with GT, GC or GG the algorithm searches for AG, if it starts with an AT the algorithm searches for AC. Secondly, the algorithm looks for the nucleotide pairs GT, GC, GG and AT in the intron sequence, which define ends of exon candidates and 5’ splice sites of the proposed intron. If the intron following the original exon ends with AG the algorithm searches for GT, GC and GG, if it ends with AC the algorithm searches for AT. The nucleotide sequences between two possible 3’ and 5’ splice sites of the scanned intron that have a length similar to the length of the original exon are considered as exon candidates. The maximum length difference between an exon and its candidate can be adjusted by the *allowed length difference* parameter in number of amino acids. The default value of this parameter is 20aa.

For terminal exons, the algorithm is able to scan the up- and downstream regions of the gene for exon candidates. The first exon of a protein-coding gene has to start with the start codon ATG. Thus, for the first exon, alternative candidates must start with ATG instead of sharing a theoretical splice site pattern with the first exon. The last exon is followed by a stop codon (TAG, TAA, or TGA) and all exon candidates must be followed by a stop codon instead of sharing a splice site pattern with the last exon. The use of the start codon and stop codon instead of the splice sites can be adjusted by the *search with start codon for first exon* and *search with stop codon for last exon* parameters. For example it would be useful to release this restriction in the case where the algorithm searches for alternative exons in a protein fragment. The default of these parameters is to search with a start codon if the first amino acid of the user-provided protein query sequence starts with methionine, and to search with stop codons if the last exon is followed by a stop codon. To reduce the number of candidates it is possible to set the *minimal exon length* parameter. Original exons, which are shorter than this length, are not considered in the candidate search. The default value for this parameter is 15aa.

The nucleotide sequences of the exon candidates are translated into amino acid sequences using the BioRuby library (260). The candidates are translated in the same reading frame as the original exon, because their nucleotide sequences appear mutually exclusive in the resulting mRNA and thus share the same reading frame. If the translation results in an in-frame stop codon, the candidate is rejected.

Each candidate sequence is aligned to the original exon sequence. If the alignment score is high, the probability that the two exons are homologous is high as well. The optimal global alignment of the two amino acid sequences is calculated with the Gotoh algorithm, which extends the Needleman-Wunsch algorithm by affine gap costs (319,320). For this task, the pair_align program of the SeqAn package (321) is used. The gap penalties are set to -10 for initial gaps and -2 for extending gaps. The Blosum62 matrix is used as substitution matrix (322,323). Because of differences in length and amino acid composition of the clusters of mutually exclusive exons the resulting global alignment scores are not directly comparable. To normalise the alignment scores each score is divided by the score of the alignment of the original exon sequence to itself. This relative score shows the similarity of the two sequences on a scale from zero to one. Candidates, which have a low alignment score, are rejected. The threshold for rejection can be adjusted in per cent by the *minimal score for exons* parameter (default: 15%). If candidate regions overlap the highest scoring candidates are retained or, if scores are identical, the longest candidates.

An optional recursive search was implemented to find less similar alternative exons. If this option is selected, the search is repeated with the found alternatively spliced exons as query exons. The number of recursive runs can be adjusted with the *maximal recursion depth* parameter up to three rounds of recursion (default: recursive search disabled).



For clarity introns have been scaled down by a factor of 1.5

Statistics
Exons: 112; Alternative exons: 93; Clusters: 12, 47, 32, 2
Introns: 22

Intron 15

Exon 5a

```
 AACCAGTTGGCAGAGTTTCGCCCAAGTTTCCCAATACTCTGACCAGCAGCAGTTTCACGGGAGACGAAGGC    3248083
 E  P  V  G  R  V  S  P  K  F  P  N  T  L  T  S  S  S  F  T  G  D  E  G          |
 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |          |
 E  P  V  G  R  V  S  P  K  F  P  N  T  L  T  S  S  S  F  T  G  D  E  G        263

TCTAGTCAAACCCTTCTATGCCCTGCCCAGGCTTATCCAGCTCCCCTTTTTAG                        3248030
 S  S  Q  T  L  L  C  P  A  Q  A  Y  P  A  P  L  F  R                            |
 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |                            |
 S  S  Q  T  L  L  C  P  A  Q  A  Y  P  A  P  L  F  R                          281
```

Intron 16

Alternative Exon 5b Score: 39.82 %

```
 AACCCATTGCGAGTGTGGGGCCCAGACTTCTCTCTGGTAATGACATTAAGGTGCTTCAGTTCTCTGCGAGC    3247848
 E  P  I  A  S  V  G  P  R  L  L  S  G  N  D  I  K  V  L  Q  F  S  A  S          |
 |  |  X  X  X  |  X  |  X  X  X  X  X  |  X  X  X  X  X  X  |  X  X  X          |
 E  P  V  G  R  V  S  P  K  F  -  -  P  N  T  L  T  S  S  S  F  T  G  D        261

CAAGCCAGC------ACCCTCTTGTGTCCAGCTCAATCATATCCAGTGCCAGTCTTTAG                  3247795
 Q  A  S  -  -  T  L  L  C  P  A  Q  S  Y  P  V  P  V  F  R                      |
 X  X  |  -  -  |  |  |  |  |  |  |  X  |  |  X  |  X  |  |                      |
 E  G  S  S  Q  T  L  L  C  P  A  Q  A  Y  P  A  P  L  F  R                    281
```

Intron 17

Alternative Exon 5c Score: 42.99 %

```
 AGCCTGTTGGCAGTATTGGACCCCGTTTGACTAGCGGCGATGAGTCGCGAATTCTCCGGGTGTATCTGGCC    3247639
 E  P  V  G  S  I  G  P  R  L  T  S  G  D  E  S  R  I  L  R  V  Y  L  A          |
 |  |  |  |  X  X  X  |  X  X  X  X  X  X  X  |  X  X  X  X  X  X  X  X          |
 E  P  V  G  R  V  S  P  K  F  P  N  T  L  T  S  S  S  F  T  G  D  E  G        263

GCAAGTGCAACTCTTCTCTGCCCGGCTCAGGCTTATCCGGTGCCCTTCTTTAG                        3247586
 A  S  A  T  L  L  C  P  A  Q  A  Y  P  V  P  F  F  R                            |
 X  |  X  |  |  |  |  |  |  |  |  |  |  X  |  X  |  |                            |
 S  S  Q  T  L  L  C  P  A  Q  A  Y  P  A  P  L  F  R                          281
```

Intron 18

Figure 2.5-2: ***Gene structure representation and detailed alignment view.*** *The figure shows the WebScipio gene structure representation of the Drosophila melanogaster Dscam gene with mutually exclusive spliced exons and a section of the alignment view including exon 5 and the first two identified alternative exon candidates. The colours in the gene structure figure are the same as the colours of the exon identifiers in the text alignment. The opacity of the colours of each alternative exon corresponds to the alignment score of the alternative exon to the original one. This score is shown in the detailed alignment view next to the exon identifier. For each exon the genomic sequence, its translation, and the translation of the original exon is shown. Identical residues are illustrated as dashes and mismatches as red highlighted crosses. The crosses are highlighted in light red for amino acids, which are chemically similar. Gaps are marked as green hyphens.*

WebScipio integration

The WebScipio tool allows reconstructing an exon-intron gene structure based on a protein sequence query. This reconstruction step is the basis for the mutually exclusive spliced exon search. The user can enable the search and adjust several parameters in the Advanced Options section of WebScipio. The search will run subsequently to the gene structure reconstruction step. In addition, the user can enable the search after uploading a previously calculated and downloaded Scipio result.

The result of the search is displayed in the Result section of the WebScipio interface (Figure 2.5-2, top). The standard gene structure picture is extended by the predicted mutually exclusive spliced exons. The alternative exons corresponding to the same original exon constitute a cluster. Exons of a cluster get the same colour. The original exon is dark coloured and the corresponding predicted ones are lighter coloured depending on their similarity with respect to the original exon. In the Statistics section the number of exons in each cluster is shown in colour.

The Alignment view (Figure 2.5-2, bottom) offers a detailed analysis at the sequence level. For each alternative exon the genomic sequence, its translation, and the alignment to the original translated exon are shown. The alignment score is given in per cent. The alternative exons are also marked in the Genomic DNA result view. In the Coding DNA and Translation result view the user can choose the alternative exons that should build the alternative coding DNAs or protein sequences. The results can be downloaded in several data formats. The YAML file contains all corresponding information and can later be uploaded and used for future analysis (290). Additionally, the results can be downloaded as General Feature Format (GFF) file (324). The figures can be downloaded in the Scalable Vector Graphics (SVG) format for further high quality processing (259). Example searches as well as further descriptions of the search parameters are provided on the help pages of WebScipio.

2.5.4 Results and Discussion

Identification of mutually exclusive spliced exons

The search for mutually exclusive spliced exons is based on three criteria: (1) The lengths of the mutually exclusive exons must be very similar, because these exons are supposed to code for the same part in the resulting protein structure, including identical secondary structural elements. (2) To be spliced in a mutually exclusive way, the exons must have similar splice sites and reading frames to be compatible with the previous and following exons. (3) The exons must encode homologous protein sequences, because their inclusion into the protein structure must be compatible with the corresponding local structural environment. The search implemented in WebScipio is based on the availability of the gene structure. Firstly, mutually exclusive exon candidates are searched for using corresponding splice sites to the query exons and restricting the candidate length to similar reading frames (e.g. split codons in the query exon must result in split codons in the candidate exons). Total length difference is less restricted allowing length differences between query and candidate exons at the DNA-level in multiples of three for each additional or missing codon. These candidate exons are then filtered and scored based on the Blosum62 matrix. The best scoring, non-overlapping candidates are proposed to be alternative exons to the respective query exon, resulting in a cluster of mutually exclusive exons. With this approach, the absolute necessary constraints at the DNA-level that can be obtained by bioinformatics means are combined with biological information. Based on these criteria several cases can be distinguished: (A) alternative exons found in the surrounding introns of single internal exons should form true clusters of mutually exclusive exons, (B) alternative exons found for terminal exons most probably constitute multiple promoters or multiple poly(A) sites, (C) clusters of several exons in combination, which can be found by searching for candidates for all exons in all introns and up- and downstream regions, most probably represent cases of tandemly arrayed gene duplications or *trans*-spliced genes.

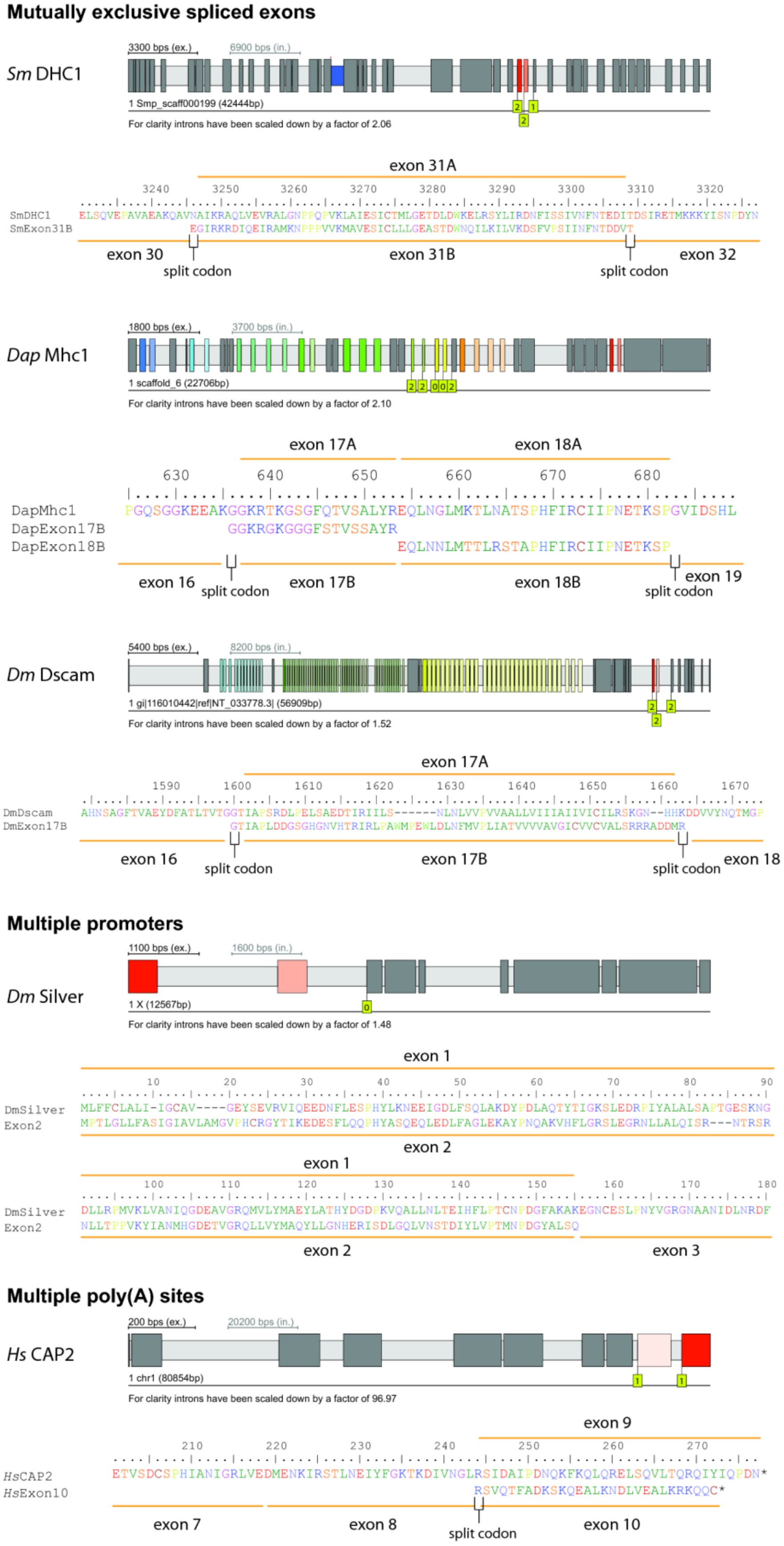

Mutually exclusive spliced exons
Sm DHC1
3300 bps (ex.)
6900 bps (in.)
1 Smp_scaff000199 (42444bp)
For clarity introns have been scaled down by a factor of 2.06
2
1
2
exon 31A
3240
3250
3260
3270
3280
3290
3300
3310
3320
SmDHC1 ELSQVEPAVAEAKQAVNAIKRAQLVEVRALGNPPQPVKLAIESICTMLGETDLDWKELRSYLIRDNFISSIVNFNTEDITDSIRETMKKKYISNPDYN
SmExon31B EGIRKRDIQEIRAMKNPPPVVKMAVESICLLLGEASTDWNQILKILVKDSFVPSIINFNTDDVT
exon 30
split codon
exon 31B
split codon
exon 32
Dap Mhc1
1800 bps (ex.)
3700 bps (in.)
1 scaffold_6 (22706bp)
2
2
0
0
2
For clarity introns have been scaled down by a factor of 2.10
exon 17A
exon 18A
630
640
650
660
670
680
DapMhc1 PGQSGGKEEAKGGKRTKGSGFQTVSALYREQLNGLMKTLNATSPHFIRCIIPNETKSPGVIDSHL
DapExon17B GGKRGKGGGFSTVSSAYR
DapExon18B EQLNNLMTTLRSTAPHFIRCIIPNETKSP
exon 16
split codon
exon 17B
exon 18B
exon 19
split codon
Dm Dscam
5400 bps (ex.)
8200 bps (in.)
1 gi|116010442|ref|NT_033778.3| (56909bp)
2
2
2
For clarity introns have been scaled down by a factor of 1.52
exon 17A
1590
1600
1610
1620
1630
1640
1650
1660
1670
DmDscam AHNSAGFTVAEYDFATLTVTGGTIAPSRDLPELSAEDTIRIILS------NLNLVVPVVAALLVIIIAIIVICILRSKGN--HHKDDVVYNQTMGP
DmExon17B GTIAPLDDGSGHGNVHTRIRLPAWMPEWLDLNFMVPLIATVVVVAVGICVVCVALSRRRADDMR
exon 16
split codon
exon 17B
split codon
exon 18
Multiple promoters
Dm Silver
1100 bps (ex.)
1600 bps (in.)
1 X (12567bp)
0
For clarity introns have been scaled down by a factor of 1.48
exon 1
10
20
30
40
50
60
70
80
90
DmSilver MLFFCLALI-IGCAV----GEYSEVRVIQEEDNFLESPHYLKNEEIGDLFSQLAKDYPDLAQTYTIGKSLEDRPIYALALSAPTGESKNG
Exon2 MPTLGLLFASIGIAVLAMGVPHCRGYTIKEDESFLQQPHYASQEQLEDLFAGLEKAYPNQAKVHFLGRSLEGRNLLALQISR---NTRSR
exon 2
exon 1
100
110
120
130
140
150
160
170
180
DmSilver DLLRPMVKLVANIQGDEAVGRQMVLYMAEYLATHYDGDPKVQALLNLTEIHFLPTCNPDGFAKAKEGNCESLPNYVGRGNAANIDLNRDF
Exon2 NLLTPPVKYIANMHGDETVGRQLLVYMAQYLLGNHERISDLGQLVNSTDIYLVPTMNPDGYALSQ
exon 2
exon 3
Multiple poly(A) sites
Hs CAP2
200 bps (ex.)
20200 bps (in.)
1 chr1 (80854bp)
1
1
For clarity introns have been scaled down by a factor of 96.97
exon 9
210
220
230
240
250
260
270
HsCAP2 ETVSDCSPHIANIGRLVEDMENKIRSTLNEIYFGKTKDIVNGLRSIDAIPDNQKFKQLQRELSQVLTQRQIYIQPDN*
HsExon10 RSVQTFADKSKQEALKNDLVEALKRKQQC*
exon 7
exon 8
split codon
exon 10

Figure 2.5-3: ***Example cases of mutually exclusive spliced exons, multiple promoters and multiple poly(A) sites.*** *The figure illustrates three examples of genes containing mutually exclusive spliced exons, one example containing multiple promoters, and one containing multiple poly(A) sites. Dark grey bars and light grey bars mark exons and introns, respectively. The small blue bar represents an "intron?" that does not have canonical splice sites because an exon is missing in the assembly. Coloured big bars represent mutually exclusive exons found by the new algorithm. The darkest coloured bar is the exon that was included in the query sequence, while the lighter coloured bars represent identified mutually exclusive exons. The higher the similarity between the candidate and the query exon the darker will be the colour of the candidate (100% identity would result in the same colour). Yellow boxes with numbers indicate the reading frame of the corresponding exon.*

Table 2.5.1: Mutually exclusive exons in the Drosophila species Dscam genes

exon	Dm	AF260 530	Dse[a]	Dy	Der	Da	Dp	Drp[a]	Dw	Dmo	Dv	Dg
4	12	12	12	12	12	12	12	10	12	12	12	12
6[b]	46/47	47/48	46[c]	39/40	44	47	49	49	48/49	50	52	53
9	32	33	29	32	33	33	32	29	29	32	32	32
17	2	2	2	2	2	2	2	2	2	2	2	2
total	92/93	94/95	90	85/86	91	94	95	90	91/92	96	98	99
[16, 46]	95	95	95	87	94	93	94	95	95	95	98	94
[15]	95	95		88		93	94			95	98	

Dm = *Drosophila melanogaster*; Dse = *Drosophila sechellia Rob3c*; Dy = *Drosophila yakuba Tai18E2*; Der = *Drosophila erecta TSC#14021-0224.01*; Da = *Drosophila ananassae TSC#14024-0371.13*; Dp = *Drosophila pseudoobscura MV2-25*; Drp = *Drosophila persimilis MSH-3*; Dw = *Drosophila willistoni TSC#14030-0811.24*;

Dmo = *Drosophila mojavensis TSC#15081-1352.22*; Dv = *Drosophila virilis TSC#15010-1051.87*; Dg = *Drosophila grimshawi TSC#15287-2541.00*.

[a] Genomes are fragmented and contain gaps in the Dscam genes.

[b] The first number corresponds to a search with standard parameters, the second to searches with one round of recursion.

[c] One of the exons is a pseudo-exon because it misses the last forth of the exon because of an in-frame stop codon.

Example genes with clusters of mutually exclusive exons

To test the quality of the new algorithm, several well-known genes with clusters of mutually exclusive exons with different characteristics were analysed (Figure 2.5-3). The first test case is the cytoplasmic dynein heavy chain from *Schistosoma mansoni* (*Sm*DHC1). Dynein heavy chains belong to the longest genes in eukaryotes encoding 4000 – 5000 residues and are spread over several dozens of exons. The mutually exclusive exon is clearly identified in the middle of the gene, encoding split codons at the 3'- and 5'-end of the exon. The query exon and the candidate exon have identical lengths and show strong homology. Based on the multiple sequence alignment of more than 2000 DHCs these exons are mutually exclusive and not constitutive or differentially included. The second case represents the muscle myosin heavy chain gene from the waterflea *Daphnia magna* (24). The arthropod muscle myosin heavy chain genes contain several clusters of mutually exclusive exons to fine tune the mechanochemical characteristics of the motor domain that are needed to accomplish the different tasks in the various muscle types (325). The *Dap*Mhc1 is an example with nine clusters of mutually exclusive exons of which several are adjacent and not interrupted by constitutive exons. The new algorithm found all mutually exclusive exons that have manually been identified previously (24). The two example alignments show that the new algorithm is able to correctly identify even short exons with limited complexity, and subsequent clusters of mutually exclusive exons encoded in different reading frames. The third example shows the prediction of the mutually exclusive exons in Dscam (Down syndrome cell adhesion molecule) from *Drosophila melanogaster*, which is known to encode the largest set of mutually exclusive exons of any gene analysed so far (326,327). The potentially 95 mutually exclusive exons of the Dscam gene are organized into four clusters that are separated by constitutive exons. The exon 4, 6, 9, and 17 clusters are supposed to contain 12, 48, 33, and 2 exons, respectively (327). In the publicly available *Drosophila melanogaster* reference genome sequence (chromosome assembly version 4.1 as provided by Flybase (328,329)) mutually exclusive exons were searched using a gene translation containing the first exons of each of the clusters as query sequence. If clusters contain that many exons as are found in the Dscam genes it might be possible that the exon, that has been included in the query sequence, is the most divergent of the exons of the cluster. Therefore, a parameter to the search algorithm that enforces recursive searches in all introns with the newly identified exon candidates was introduced. Exons that might not be identified in the first round might then be found in the second, third, or later round. Of course, the recursive depth should not be too large to avoid the inclusion of false positive exons because of the decreasing stringency of the query exons. Including every first exon of the Dscam mutually exclusive exon clusters in the query sequence, all twelve exons of the exon 4 cluster were identified, both exons of the exon 17 cluster, and 46 and 32 exons for the exon 6 and exon 9 cluster, respectively (Figure 2.5-3,Table 2.5.1). Increasing the recursive depth to one also revealed exon 6.11, which is the most divergent exon of the cluster, and which has not been detected in transcriptome studies yet (330–332). Exon 6.47 was not identified because the intron before exon 6.47 does not have an "AG" at the 3'-end and is therefore not compatible with the "GT" at the 5'-end of the intron succeeding exon 5. The supposed 5'-end sequence of exon 6.47 is different to the published sequence (327) but is supported by many genomic DNA reads available from GenBank (a genomic DNA read identical to the published sequence was not found). Exon 9.13 was also not identified because it contains a frame shift in the *Drosophila* reference genome assembly, supported by many genomic DNA reads. Therefore, the translations of the predicted transcripts containing exon 9.13 all stop shortly behind this frame shift (e.g. NM_001043054.1, NM_001043034.1, and NM_001043065.1). However, both exon 6.47 and exon 9.13 were identified in many transcripts (330–332). Thus, either the genome assembly based on the many genomic DNA reads is wrong, which is unlikely, or the many EST/cDNA-reads are wrong, which is also unlikely, or the genomic DNA has been obtained from a different strain than the one that has been used in the transcriptome studies. WebScipio is, however, not able to identify mutually exclusive exons if those do not correspond to the exon length (e.g. frame shifts will result in other reading frames and exon lengths) and corresponding splice site restrictions. The strength of the new algorithm is illustrated at the exon 17 cluster that encodes two highly divergent but mutually exclusive spliced exons (Figure 2.5-3). When applying the search for mutually exclusive exons in the Dscam gene against the published genomic sequence (NCBI accession number AF260530 (327)) all proposed 95 mutually exclusive exons were identified (Table 2.5.1). Less mutually exclusive exons in the search against the *Drosophila melanogaster* reference genome sequence compared to the search against the published sequence are therefore not due to problems with the search algorithm.

In addition, mutually exclusive exons in the Dscam genes of the other sequenced *Drosophila* species were searched ((276);Table 2.5.1). Here, all mutually exclusive exons were found immediately, and only three further exons were identified by a second recursive round of exon search. As found for the *Drosophila melanogaster* gene, WebScipio identified sometimes more sometimes less exons compared to the published analyses (315,316,333). However, the WebScipio searches were performed against the official reference genome assemblies, while the published analyses were based on manually performed genomic clone assemblies of the Dscam gene regions. Therefore, the differences in exon numbers do not result from shortcomings of the search algorithm, but from differences in the assembly of the reference genome data and the manually assembled genomic regions.

Example genes encoding 5'- and 3'-terminal exons with features of mutually exclusive spliced exons

Terminal exons are often not selected at the level of splicing. Instead, initial (5’-terminal) exons are most probably selected at the level of transcription that starts at different promoters. Terminal exons (or better alternative exons encoding for the terminal stop codon) might either be spliced as differentially included exons, like in the case of the *Drosophila* muscle myosin heavy chain gene (24), or as multiple poly(A) sites. Nevertheless, these terminal exons might contain an important structural part of the encoded protein and thus often have similar length and show sequence similarity. Figure 2.5-3 shows two examples of genes that contain 5’- and 3’-terminal exons sharing the described features of mutually exclusive exons, but are spliced as multiple promoters or multiple poly(A) sites. The silver protein of *Drosophila melanogaster* illustrates a case where two initial exons, which are transcribed/spliced as multiple promoters, share the features of mutually exclusive exons. The capping protein beta (Capβ) from *Homo sapiens* represents a case where homologous 3’-terminal exons containing multiple poly(A) sites are found. The detection of these cases can be suppressed by disabling the search for mutually exclusive exons for 5'- and 3'-terminal exons. By default, WebScipio enables the search for homologous exons for all exons, because it is not known whether the user is searching with a complete, partial or fragmented query sequence. In the case of partial and fragmented sequences the search would provide significant results. Also, genes sometimes contain untranslated 5'- and/or 3'-terminal exons whereby the first translated exon could well be part of a cluster of mutually exclusive spliced exons. In addition, alternative terminal exons by themselves might provide interesting perspectives to the corresponding genes independently of whether they are mutually exclusively spliced or not. WebScipio cannot distinguish between the described cases and thus the user has to be careful when alternative terminal exons are proposed.

Detection of trans-spliced genes and arrays of tandem gene duplications

The *trans*-splicing of separate pre-mRNAs involving coding exons to reveal a joined transcript is a relatively uncommon event (334). In general, *trans*-spliced genes in *Drosophila melanogaster* can be distinguished into those with multiple first exons or multiple 3'-terminal exons, or those with very large introns. Many of the *trans*-spliced genes contain variable single terminal exons (e.g. *mod(mdg4)* (335,336) or *lola* (337)) or alternative terminal exon groups (e.g. CG42235 (334)). When searching for mutually exclusive spliced exons based on one of the annotated isoforms of a *trans*-spliced gene potentially alternative exons of internal exons might be identified. An example of the *trans*-spliced *Drosophila melanogaster* gene CG1637 is shown in Figure 2.5-4. Three isoforms of the CG1637 gene exist (Isoform A, B, and C) that result in transcripts of a common 5’ exon spliced to isoform-specific sets of three 3’ exons. The sequences of the isoform-specific sets are homologous although the intron positions are different between the isoform A/B exons and the isoform C exons. When searching with the isoform A exons for mutually exclusive exons in surrounding introns the homologous exon of isoform B is found for the first of the three isoform A-specific exons (Figure 2.5-4-I). When only searching in surrounding introns (search in up- and downstream regions disabled) further exons are not found for isoform B (homologous exons would only exist in the downstream region, Figure 2.5-4-II) and for isoform C (the introns are at different positions so that the similar-length condition does not apply anymore, Figure 2.5-4-III). Thus, if only isoform A were known a mutually exclusive exon would have been proposed. To avoid the misannotation of exons of *trans*-spliced clusters a parameter was introduced that allows searching for candidate exons not only in the neighbouring introns but also in all introns. In Figure 2.5-4-IV the exons of isoform B were identified by searching with the exons of isoform A in all introns revealing the *trans*-spliced nature of the cluster.

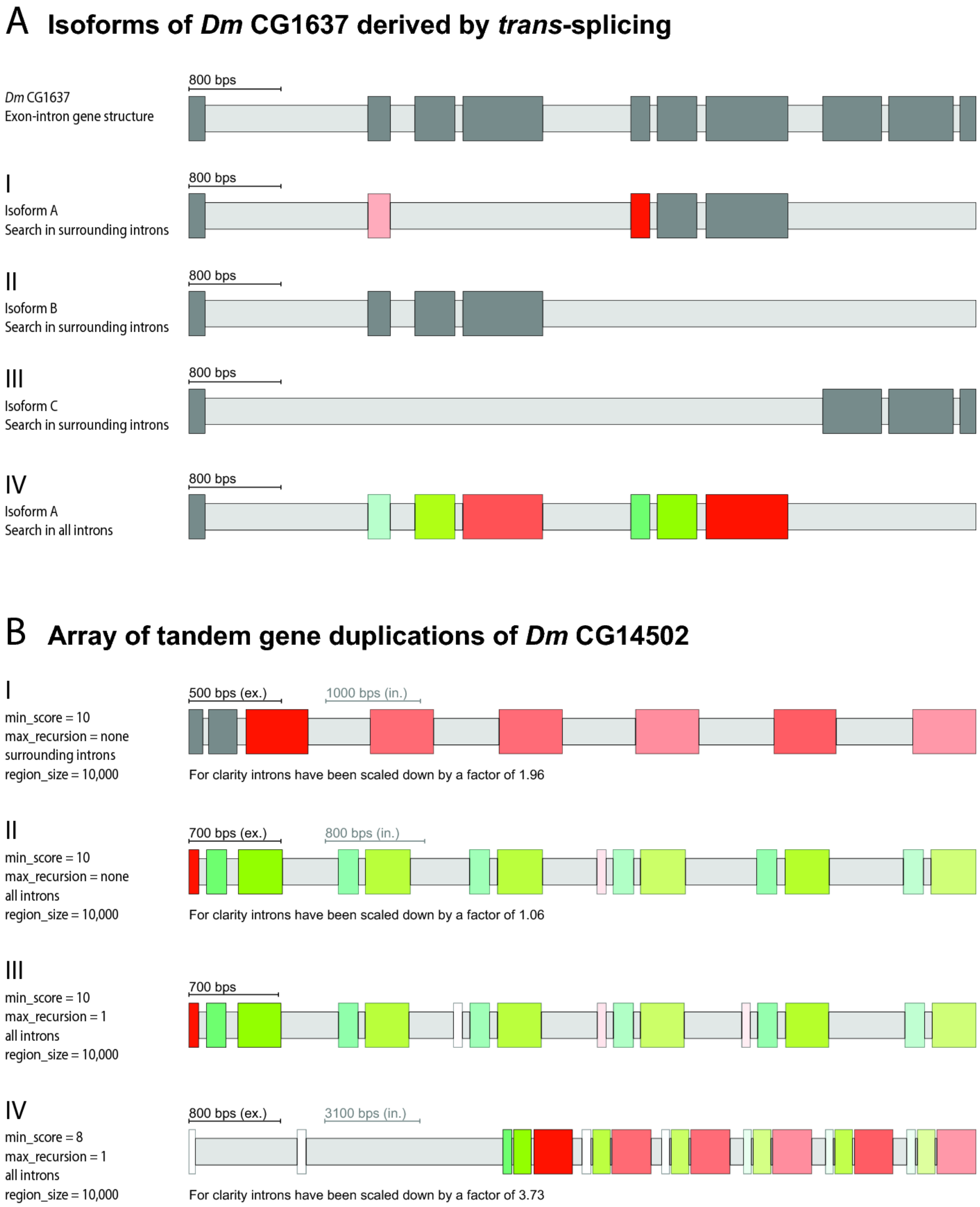


Figure 2.5-4: ***Examples of a trans-spliced gene and an array of tandem gene duplications.*** *A) Schematic representation of the trans-spliced Drosophila melanogaster CG1637 gene. The three annotated isoforms A-C are shown consisting of the common 3'-terminal start exon and different groups of alternative exons. If only isoform A were known a potentially mutually exclusive exon would have been found by a search for candidates in surrounding introns (case I). However, a search for candidates of all exons in all introns reveals the two groups of homologous exons that are trans-spliced in isoform A and B (case IV). Isoform C also encodes a cluster of trans-spliced exons whose sequence is homologous to that of isoform A/B. However, the exonic sequence is interrupted at different intron positions (case III). Note, that the gene structure annotated by Flybase (shown here) is different to the published one ((333), supplementary Figure 2.5-3). B) Gene duplications of the Drosophila melanogaster CG14502 gene. The figure shows the tandem arrangement of the duplicated genes of the Drosophila melanogaster CG14502 gene as found by WebScipio. The parameters minimal score for exons, maximal recursion depth, search in all introns and region size were adjusted for each search. With less restrict parameters less similar exons are found.*

If searching in up- and downstream regions for alternatively spliced exons, it is possible that candidate exons belong to gene duplicates (Figure 2.5-4B). In this case, the WebScipio option to search for candidates in all introns including up- and downstream regions and not only in surrounding introns helps identifying exons of gene duplications. In many cases, gene duplications result in genes arranged in tandem. Those gene duplicates often share the complete gene structure meaning that for every exon there is a corresponding exon in the duplicated gene. Figure 2.5-4B illustrates this behaviour and provides means by which users can judge between a true cluster of mutually exclusive exons belonging to one gene and a set of duplicated genes. If the search for candidate exons is only allowed in surrounding introns, a set of six homologous exons is found for the *Drosophila melanogaster* gene CG14502 (Figure 2.5-4B, I). Performing the search in all introns results in five homologous exons also for the second exon of the CG14502 gene, and shows one homologous exon for exon 1 (Figure 2.5-4B, II). The first exons of the genes seem to be very divergent. Allowing one additional recursive round of candidate search reveals the first exons for two additional gene homologs (Figure 2.5-4B, III). In addition lowering the score reveals the exon 1 candidates of the remaining two gene homologs, although two further regions with very low homology to exon 1 appear in the upstream region of the CG14502 gene (Figure 2.5-4B, IV). This example illustrates the use of the search parameters so that gene duplications can be identified. Gene duplicates that are not arranged in tandem but are distributed in the genome do not provide problems in evaluating exon candidates, because the search is restricted to a certain size of the up- and downstream regions. If needed, these gene duplicates can be identified with WebScipio using the general *multiple results* option.

Application of the search algorithm for mutually exclusive exons to genome scale data

The described search algorithm identifies three types of exons as described above: (A) mutually exclusive exons, (B) terminal exons that are spliced as multiple promoters or multiple poly(A) sites but share similar length, reading frame, and sequence homology, and (C) exons with the characteristics of mutually exclusive exons that are actually part of tandemly arrayed gene duplicates or groups of alternative exons in *trans*-spliced genes. Type B and type C exons are false positives, when looking for mutually exclusive exons. In addition, false positive exons are those exons that show all characteristics of type A exons but are constitutively or differentially included spliced. False negatives exons, which are not identified by WebScipio, are those mutually exclusive exons that do not have similar length and sequence homology. To quantify the amount of each of these exon types we searched the complete X chromosome of the fruit fly *Drosophila melanogaster* for mutually exclusive spliced exons with WebScipio and compared the results to the Flybase annotation.

Protein sequences for the search were obtained from the Flybase annotation (version 5.27) and mapped to the genomic sequence of the X chromosome using Scipio. 2,967 transcripts containing more than one exon were derived from 1,705 genes. For each exon mutually exclusive alternative splice variants have been searched for in the surrounding introns. The search parameters were set to 20 amino acids for the *allowed length difference*, to 15% for the *minimal score for exons*, and to 15 amino acids for the *minimal exon length*. We did not search for alternative exons in up- and downstream regions of genes, and we did not apply the recursive search, which means the repeated search for further alternative exons with the newly identified exons that we demonstrated for Dscam (see above). Three genes (lethal (1) G0193, CG1637, and CG42249), in which mutually exclusive exons were found, were excluded from the analysis, because the respective exons are spliced in a mutually exclusive manner in groups of two, three, and four exons, instead of single exons within a cluster. Those genes are probably *trans*-spliced (e.g. see Figure 2.5-4A).

Search for non-mutually exclusive exons sharing similar length, same reading frame, and sequence homology

It could well be possible that internal exons with similar length, same reading frame, and showing sequence homology are not mutually exclusive spliced exons, but constitutive exons or exons spliced by one of the other types of alternative splicing. To get a statistically relevant number of these types of exons we collected all genes of the *Drosophila melanogaster* X chromosome containing at least two exons based on the Flybase annotation version 5.27. The transcripts of each gene were analysed independently because alternative splicing produces different exon neighbours. Thus exons are counted for each transcript (not each gene) even if the transcripts have the same start and end points in the genomic sequence. In total, the 2,967 transcripts of the *Drosophila melanogaster* X chromosome include 16,180 exons. All neighbouring exons were compared with respect to having similar length (*allowed length difference* 20aa), sharing high similarity (*minimal score for exons* 15%), coding for at least fifteen amino acids (*minimal exon length* 15aa), and encoding the same reading frame.

Table 2.5.2: Search for exons annotated as constitutively spliced or differentially included sharing similar length, same reading frame and sequence homology in the Drosophila melanogaster X chromosome

	Total	Hits[a]	Percentage
Exons	16180	90	0.56%
Transcripts	2967	20	0.67%
Genes	1705	6	0.35%

[a] Exons (or transcripts/genes containing exons) which share similar length, same reading frame and sequence homology

The results are summarized in Table 2.5.2(for detailed information see Additional file 2.5.9.2). Only 0.56% of the non-mutually exclusive exons (90 out of 16180) share the features of mutually exclusive exons. These exons are located in only six genes out of 1705 (0.35%). In one of the six genes (*Ciboulot*) the two homologous exons are terminal exons and would represent a case of multiple poly(A) sites if alternatively spliced. This analysis shows that the chance that the exons predicted by WebScipio as mutually exclusive exons will later (e.g. after obtaining cDNAs) be reannotated as constitutive or differentially included exons, is very low.

Search for mutually exclusive spliced exons in the Drosophila melanogaster X chromosome

Some categories have to be defined to separate true (annotated) mutually exclusive spliced exons from predicted ones and false positives and false negatives. As real mutually exclusive exons we regard those with the following criteria: An exon is part of a cluster of mutually exclusive spliced exons if each transcript of the gene contains exactly one exon of the cluster (not none or more than one), the cluster contains at least two exons, the exons of the cluster are neighbouring exons, and the cluster is surrounded by further exons. The latter criterion distinguishes the mutually exclusive spliced exons from clusters of initial exons (5'-terminal exons) and 3'-terminal exons that are spliced in a mutually exclusive manner and share sequence similarity, similar length, and splice site conservation. In contrast to real mutually exclusive spliced exons the exons of these clusters appear mutually exclusive in the transcripts but their transcription and splicing is regulated in a different way. These clusters are therefore regarded as types of multiple promoters and types of multiple poly(A) sites, and are false positives. Other types of false positives are those exons that are predicted by WebScipio but overlap with already annotated exons and do not match exactly the positions of these exons. False negatives are those exons that do not meet the preconditions of similar length and sequence homology. However, if those exons are mutually exclusive spliced they must have conserved splice sites and reading frames.

In total, 94 exons of similar length, same splice sites and reading frames, and sequence homology have been identified by WebScipio, of which 65 are potentially in clusters of mutually exclusive exons, 21 in clusters of multiple promoters, and 8 in clusters of multiple poly(A) sites (Figure 2.5-5). Of the 65 exons predicted to belong to internal clusters of mutually exclusive spliced exons, 26 exons are already annotated in Flybase. 39 exons are predictions by WebScipio that have not been described before. These 39 exons are distributed into 18 clusters that belong to 17 genes. Thus, there are several clusters with more than two alternative exons, and one gene with two clusters. If the Flybase based annotation is assumed to represent true mutually exclusive exons, the chart represents the specificity of our method. The 26 already known mutually exons divided by 65 predicted exons result in 40% specificity.

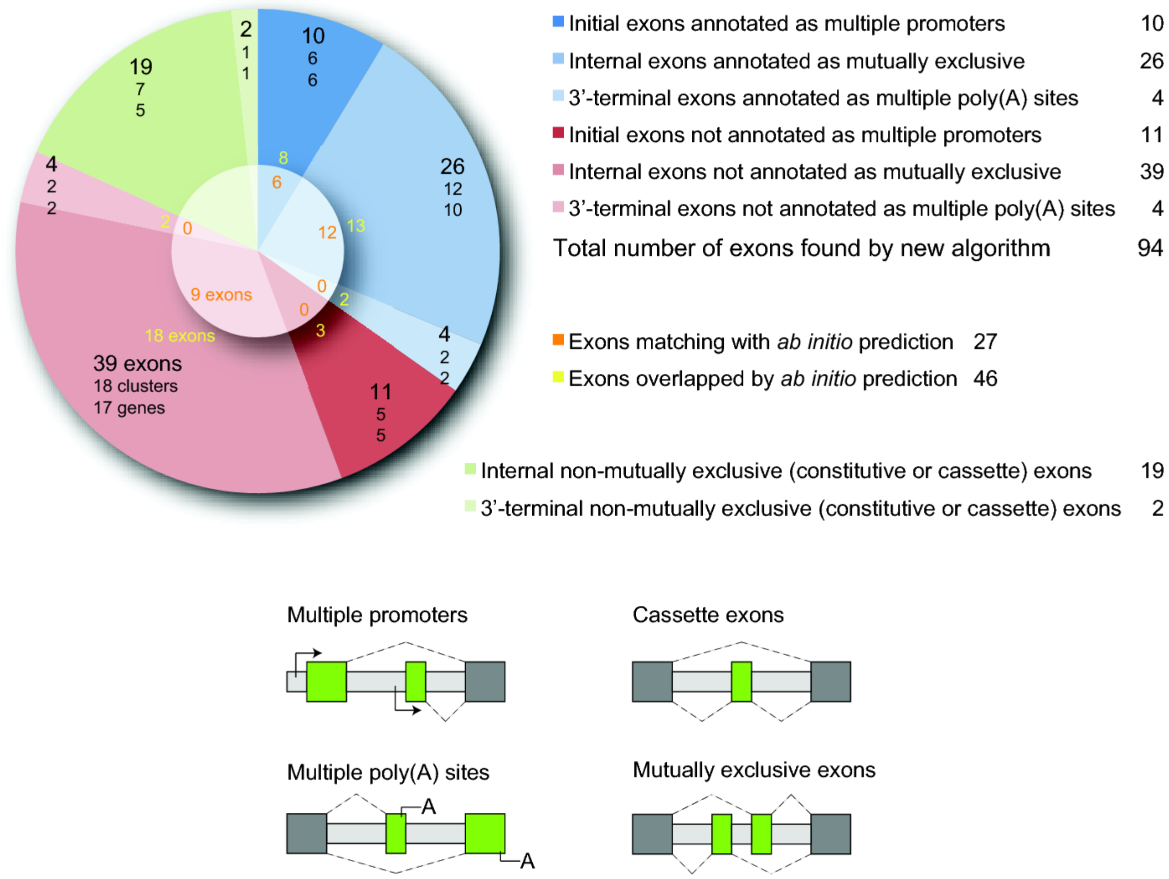
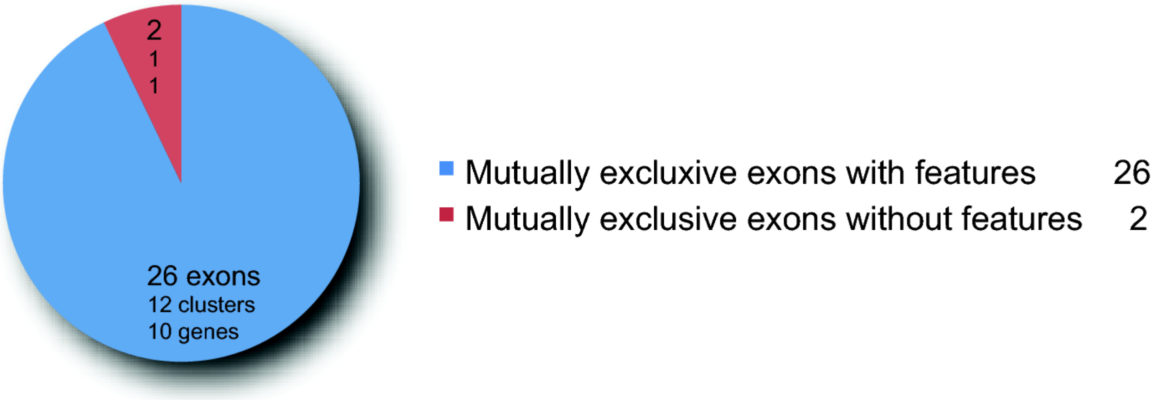


Figure 2.5-5: ***Exons located on the Drosophila melanogaster X chromosome sharing similar length, same splice sites and reading frames, and sequence similarity.*** *The pie chart shows the total number of exons of the Drosohpila melanogaster X chromosome, which share the features used by the new search algorithm. The blue and red slices represent the number of exons found by the new algorithm compared to existing annotations and to the ab initio prediction by AUGUSTUS, shown in the middle. The blue part illustrates the exons already annotated by Flybase, in contrast to the exons in clusters additionally predicted by WebScipio in red. The pie is divided in slices for initial, internal, and 3'-terminal exons. In addition to the number of exons, the chart indicates the number of clusters and genes, in which these exons were found. The orange numbers in the middle part of the pie indicate how many of the respective exons are found and reconstructed with correct exon borders by the ab initio prediction with AUGUSTUS, while the yellow numbers reveal the number of exons to which exons predicted by AUGUSTUS at least overlap. The green slices indicate constitutive exons, which share the features of mutually exclusive exons. These are the same exons, clusters, and genes as listed in Table 2.5.2 and Additional file 2.5.9.2. At the bottom, the figure illustrates the different types of alternatively spliced exons (multiple promoters, multiple poly(A) sites, mutually exclusive exons) in comparison with the cassette exon type.*

However, the value for the specificity is misleading because it depends on the "known" mutually exclusive exons. We expect that many of the additional exons predicted by WebScipio will be experimentally confirmed in the future and thus will become "known" mutually exclusive exons. The true specificity will therefore be much higher than the value of 40% suggests. To analyse whether the additional exons predicted by WebScipio contain general features of exons (for example a higher GC content than the surrounding region), the found exons were compared to those of an *ab initio* prediction performed by AUGUSTUS (305) (Figure 2.5-5). In many cases the WebScipio predictions are supported by the *ab initio* prediction, which is based on the genomic sequence alone. The AUGUSTUS prediction matches 27 of the 94 exons with exact exon borders (Figure 2.5-4, orange numbers) and overlaps with 46 of them (Figure 2.5-4, yellow numbers).

The results show that about 70% of all predicted exons (65 out of 94) comprise clusters of internal mutually exclusive exons. The false positive prediction of 5'- and 3'-terminal exons as mutually exclusive exons, which comprise the remaining 30% of predicted exons, could even be suppressed by a WebScipio option. We can also conclude that WebScipio correctly identifies all but one (see following section) of the annotated mutually exclusive exons. This suggests that most of the WebScipio predictions of new mutually exclusive exon candidates will also be real mutually exclusive exons. This is supported by the *ab initio* exon prediction by AUGUSTUS that showed exon probability for about 50% of the newly predicted exons, which is comparable to the *ab initio* prediction of the already annotated exons. However, we cannot completely exclude the possibility that some of the newly predicted exons might in truth be constitutive or differentially included exons (see previous section).

False negatives would be those mutually exclusive spliced exons that do not share a similar length and sequence similarity. To figure out how often clusters of mutually exclusive exons with such characteristics exist in comparison to mutually exclusive exons with similar length and sequence similarity, all internal clusters of exons on the X chromosome that were annotated as mutually exclusive based on Flybase transcripts were manually analysed (Figure 2.5-6). Of the annotated genes only the Phosphorylase kinase γ gene contains two mutually exclusive spliced exons that do not have similar length and sequence (Figure 2.5-6, bottom). If the Flybase annotation is assumed as true, the chart in Figure 2.5-6 represents the sensitivity of the algorithm. 26 predicted mutually exclusive spliced exons divided by 28 annotated exons results in 93% sensitivity for internal exons. These data likely indicate that not many mutually exclusive spliced exons will be missed given the constraints of similar length and sequence similarity as implemented in WebScipio. Mutually exclusive exons predicted for 5’- and 3’-terminal exons were regarded as false positives because these rather present cases of multiple promoters, multiple poly(A) sites, and differentially included exons. However, additional untranslated terminal exons might exist that were not analysed here, and in those cases the exons, based on the translation predicted as terminal, become internal and thus true mutually exclusive exons. For comparison all terminal exons annotated as transcribed or spliced in a mutually exclusive manner have been analysed (Figure 2.5-7). Of the 101 terminal exons only 14 terminal exons share the features of mutually exclusive spliced exons. A reason for the sequence and length variability of terminal exons is that the N- and C-termini of proteins are not as restricted in their structure as internal parts. Thus, the number of false positives predicted by WebScipio is rather low.



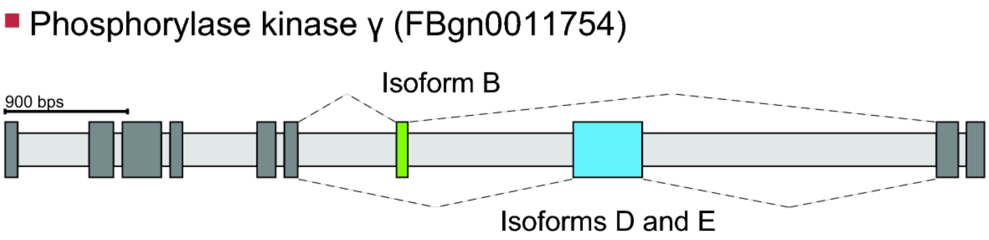


Figure 2.5-6: ***Mutually exclusive exons in genes of the Drosophila melanogaster X chromosome.*** *The figure illustrates how many of the mutually exclusive exons, which were annotated based on Flybase transcripts, share the following features: high sequence similarity, similar length, same reading frame, and a minimal exon length (15 residues). The blue slice indicates exons characterised by these features and found by the new algorithm. The red slice indicates exons not sharing these features. At the bottom, the figure shows the exon-intron structure of the Phosphorylase kinase γ gene, which includes the only cluster of mutually exclusive exons that was not found by the new algorithm.*

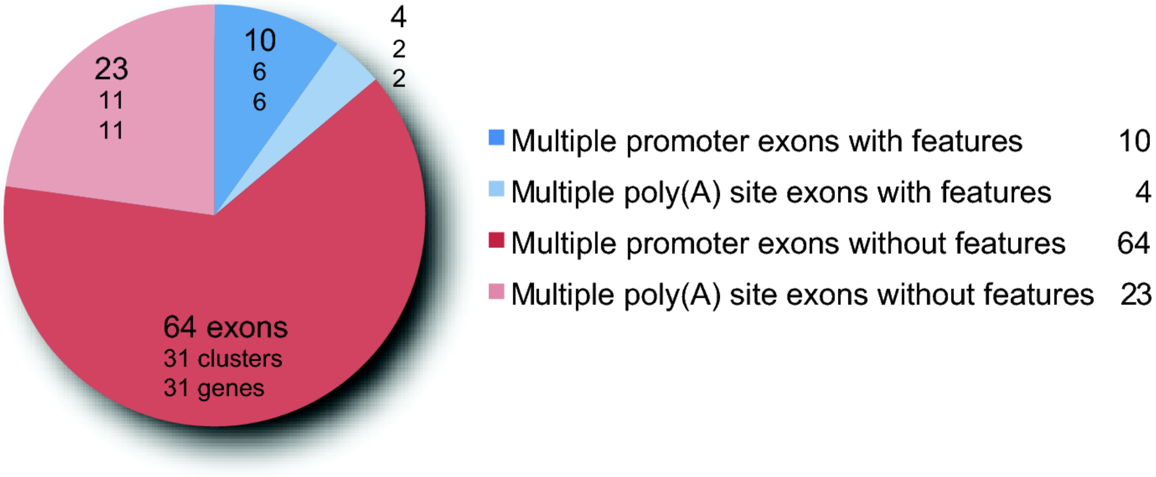


Figure 2.5-7: ***Exons belonging to clusters of multiple promoters and multiple poly(A) sites in the Drosophila melanogaster X chromosome.*** *The figure shows the number of multiple promoter exons and multiple poly(A) sites exons based on the Flybase annotation and illustrates how many of these exons share the following features: high sequence similarity, similar length, same reading frame, and a minimal exon length (15 residues). Blue slices indicate exons characterised by these features, and red slices indicate exons not sharing these features.*

Future developments and applications

Due to the precondition that mutually exclusive exons encode the same part of the protein product, we also want to include the comparison of the prediction of secondary structural elements for the query and the candidate exons as an additional scoring, analysis, and validation parameter. Also, other substitution matrices might be offered for the scoring of the aligned query and candidate exons. Scipio and WebScipio have been shown to be suitable for the prediction of genes in cross-species searches (13,212). Of course, both approaches can be combined and users can search, for example, with a human protein query sequence in other mammals to identify homologous genes and simultaneously predict mutually exclusive exons in the target sequence. Because the search for mutually exclusive exons relies on the translation of the exons as found in the genomic DNA, it does not depend on the initial query sequence but on the quality of the exons identified in the cross-species search. Another application would be to search for mutually exclusive spliced genes in the complete genomes of sequenced eukaryotes.

2.5.5 Conclusions

The extension of WebScipio to search for mutually exclusive exons is based on the precondition that these exons encode regions of the same structural part of the protein product. This precondition provides restrictions to the search for candidate exons concerning their length, splice site conservation and reading frame preservation, and overall homology. The implemented algorithm has been shown to identify all known mutually exclusive spliced exons in many example genes from various species, like the muscle myosin heavy chain gene of *Daphnia pulex* or the Dscam gene of *Drosophila melanogaster*. The search for homologs of terminal exons might, however, result in the prediction of multiple promoters, multiple poly(A) sites, groups of *trans*-spliced exons, or tandemly arrayed gene duplicates, and can therefore optionally be disabled. To quantify the quality of WebScipio to correctly predict already annotated mutually exclusive exons and to predict so far unrecognized exon candidates, an analysis of the whole X chromosome of *Drosophila melanogaster* has been performed. All but two of the 28 annotated mutually exclusive exons were found by WebScipio. In addition, WebScipio predicts 39 new mutually exclusive exon candidates of which about 50% are supported by an *ab initio* exon prediction by AUGUSTUS. In conclusion, WebScipio should be able to identify mutually exclusive spliced exons in any query sequence from any species with a very high probability.

2.5.6 Abbreviations

DHC: Dynein heavy chain; Dscam: Down Syndrome Cell Adhesion Molecule; GFF: General Feature Format; Mhc: Myosin heavy chain; SVG: Scalable Vector Graphics; YAML: YAML ain’t markup language

2.5.7 Authors' contributions

HP, FO, and MK set the requirements for the system. HP and KH wrote the software. FO assisted in and supervised the implementation of the software. BH implemented improvements to the software, and KH committed the final version. KH performed the whole chromosome search and evaluation. HP, KH, BH, and MK performed extensive testing. KH and MK wrote the manuscript. All authors read and approved the final version of the manuscript.

2.5.8 Acknowledgements

MK has been funded by grants KO 2251/3-1, KO 2251/3-2, and KO 2251/6-1 of the Deutsche Forschungsgemeinschaft.

2.5.9 Additional files

2.5.9.1 Additional file 1 - Detailed activity diagram

The detailed activity diagram shows each step of the search algorithm including points of decision and loops.

The file can be found in the corresponding publication.

2.5.9.2 Additional file 2 – Search for non-mutually exclusive exons sharing similar length, same reading frame and sequence homology

The file provides detailed information of the found genes and their gene structures.

The file can be found in the corresponding publication.

3 Manuscripts in revision

3.1 Peakr: Predicting solid-state NMR spectra of proteins

Robert Schneider[1,2,*,†], Florian Odronitz[1,†], Björn Hammesfahr[1,†], Marcel Hellkamp[1], and Martin Kollmar[1*]

[1] Department of NMR-based Structural Biology, Max-Planck-Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany.
[2] Protein Dynamics and Flexibility by NMR Group, Institut de Biologie Structurale J.P. Ebel, 41 rue Jules Horo-witz, F-38027 Grenoble Cedex 1, France.
[*] To whom correspondence should be addressed.
[†] These authors contributed equally to the work and should be regarded as joint First Authors.

Bioinformatics

3.1.1 Abstract

Motivation

When analysing solid-state NMR spectra of proteins, assignment of resonances to nuclei and derivation of restraints for 3D structure calculations are challenging and time-consuming processes. Predicted spectra that have been calculated based on, e.g., chemical shift predictions and structural models can be of considerable help. Existing solutions are typically limited in the type of experiment they can consider and difficult to adapt to different settings.

Results

Here, we present Peakr, a software to predict solid-state NMR spectra of proteins. It can take into account all types of correlations between nuclei usually relevant for assignment and structure elucidation and is able to produce lists and visualizations that can be useful when analysing measured spectra. Compared to other solutions it is fast, versatile and user friendly.

Availability and Implementation

Peakr is maintained under the GPL license and can be accessed at http://www.peakr.org. The source code can be obtained upon request from the authors.

3.1.2 Introduction

In recent years, solid-state NMR has made significant progress in studying structure and function of biomolecules such as membrane proteins and protein fibrils (338–341). However, especially for larger proteins, resonance assignment and determination of restraints for 3D structure calculations are still difficult and time-consuming processes due to often limited spectral resolution and chemical shift ambiguity, as well as complex relationships between internuclear distances and signal intensities. These problems especially apply to through-space correlations that cannot be traced along the chemical bond network (342).

For the assignment of resonances and the extraction of restraints, it has proven helpful to have predictions for spectra based on amino acid sequences, known or modelled 3D structures, chemical shift assignments or predictions from tools such as ShiftX (343), and the type of correlation probed in the respective experiment. This way, crosspeak assignments can be suggested and, for example, it can be investigated whether an experimental spectrum can be explained by a given structural model (344,345). Spectral predictions can also be used in an iterative process of obtaining shift assignments and refining the molecular structure at the same time (346).

Existing software for the prediction of NMR spectra is usually tailored to calculating spectra of small molecules in solution (347–349). Some NMR data analysis software packages such as Sparky (350), NMRPipe (351), and CcpNmr (352) contain routines for prediction of protein NMR spectra, but these are often not straightforward to use; moreover, they are usually more adapted to solution-state NMR correlation types. At present, prediction of solid-state NMR spectra of proteins is typically carried out using general-purpose tools like spread sheet software or custom-made programs limited in flexibility and usability.

Thus, it would be desirable to be able to predict a wide range of solid-state NMR experiments commonly used for resonance assignment and structure elucidation with a single software tool that is flexible enough to swiftly handle changes in input data such as chemical shift values, structural models, or labelling patterns. We implemented Peakr, a software package that fulfils these requirements. Spectra for 2D (^{15}N,^{13}C) and (^{13}C,^{13}C) intra- and inter-residue as well as through-space correlations can be computed quickly and can easily be adapted to specific needs. Using the calculated spectra, visual and numerical comparisons between predicted and measured data are possible. Peakr is available through a web interface (www.peakr.org).

3.1.3 Methods

The higher abstracted parts of Peakr are implemented in the object oriented programming language Ruby (78) using the BioRuby library (260). The more data intensive bookkeeping is done using a PostgreSQL database (73). The Ruby on Rails framework (79) is used for the web application, which drives the web interface. Calculated and experimental spectra are visualized by a Peakr-plugin to the PyBiomaps library (353). The workflow of the software is outlined in Figure 3.1-1. Briefly, measured or predicted chemical shifts are assigned to the nuclei of a user-provided protein sequence. Based on these shifts and an optionally provided protein structure, various 2-dimensional spectra can be calculated. For visualization, either all or a subset of the residues and nuclei of the protein can be selected. Predicted spectra can be superimposed on and compared with experimental spectra.

3.1.3.1 Protein sequences

Protein object models represent the amino acid sequence and the NMR chemical shifts of nuclei. They are either created from a user-provided protein sequence or from a Protein Data Bank (PDB) structure file (354).

3.1.3.2 Chemical shifts

Chemical shifts are added to nuclei from user-provided shift lists, computer-generated shift estimations, database values, or from a combination of these methods. User-provided lists of shifts are currently accepted in Sparky (350), ShiftX (343), and CSV (comma-separated values) formats. Output from other software can be converted to one of these formats using e.g. WeNMR (352). If a PDB file is provided, chemical shifts can directly be estimated using either ShiftX (343), ShiftX2 (85), Shifts (355), Sparta (356), Sparta+ (86), shAIC (357), or CamShift (358). Alternatively, average chemical shifts from the Biological Magnetic Resonance Database (BMRB, http://www.bmrb.wisc.edu) (359,360) can be assigned to nuclei. Here as well, other options such as using RefDB database values (361) could be added to Peakr if desired. These methods can be combined such that, for example, user-provided shifts from a Sparky-format list are used where available, while remaining nuclei are assigned either ShiftX prediction values or, for nuclei whose chemical shifts are not predicted by ShiftX, BMRB database values. Each nucleus can have multiple shifts to account for possible cases of conformational polymorphism sometimes seen in solid-state preparations of proteins (362).

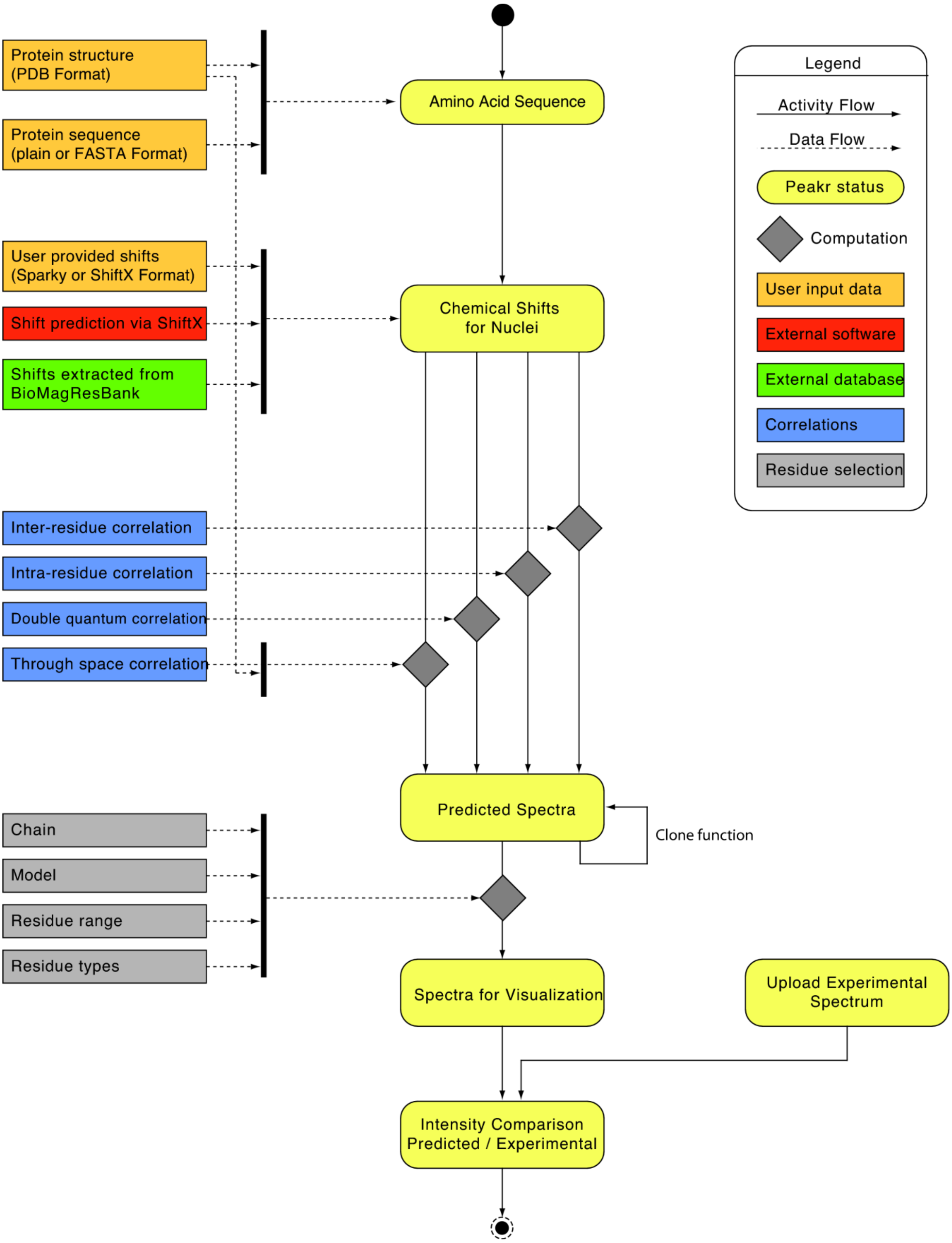

Protein structure (PDB Format)
Protein sequence (plain or FASTA Format)
Amino Acid Sequence
User provided shifts (Sparky or ShiftX Format)
Shift prediction via ShiftX
Shifts extracted from BioMagResBank
Chemical Shifts for Nuclei
Inter-residue correlation
Intra-residue correlation
Double quantum correlation
Through space correlation
Predicted Spectra
Clone function
Chain
Model
Residue range
Residue types
Spectra for Visualization
Upload Experimental Spectrum
Intensity Comparison Predicted / Experimental
Legend
Activity Flow
Data Flow
Peakr status
Computation
User input data
External software
External database
Correlations
Residue selection

Figure 3.1-1: Workflow of the Peakr web application.

3.1.3.3 Conformations

Conformations represent a set of coordinates of the atoms of a given protein. Each protein can have several conformations. This way, an ensemble of structures or different conformations of the same protein under different experimental conditions can be represented. Conformations are obtained from models as included in PDB files. Protons have to be already present in the PDB file if they are required for the spectrum to be predicted (see through-space correlations below). Protons can be added to a structure using standard software such as WHATIF (363) or PyMOL (The PyMOL Molecular Graphics System).

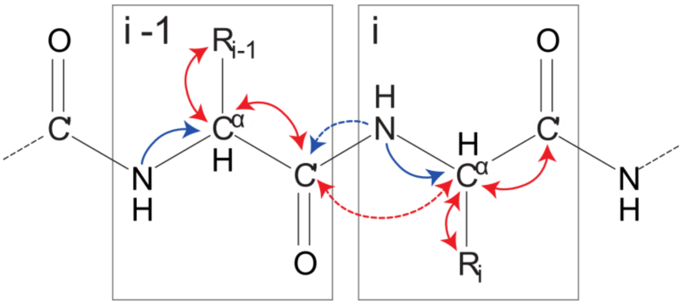


Figure 3.1-2: Scheme of the protein backbone. Two amino acid residues i and i-1 with sidechains symbolized by R_i *and* R_{i-1} *are shown. Possible (solid-state) NMR correlations for sequential resonance assignment are indicated by arrows. Red, (*$^{13}C,^{13}C$*) correlations; blue, (*$^{15}N,^{13}C$*) correlations; solid lines denote intra-, dashed lines interresidue transfer.*

3.1.3.4 Correlations

Correlations represent the type of experiment conducted to obtain a specific spectrum (see Figure 3.1-2 for typical correlations used in resonance assignment). They are applied to all or selected residues of the protein sequence to yield a list of crosspeaks. Six types of correlations are available:

Intra-residue ($^{13}C,^{13}C$) correlations

For defining ($^{13}C,^{13}C$) correlations within residues, the user can select which nuclei should be considered depending on their distance from the protein backbone (e.g. all carbons or only CO, CA and CB nuclei). It can also be specified by how many bonds two carbon nuclei may be separated to be included in the predicted spectrum (yielding, e.g., only one-bond correlations or correlations between all carbons separated by up to three bonds). This way, the user can select correlations of interest and avoid overcrowding of the predicted spectrum with peaks that might not be present in a measured spectrum due to, for example, increased side chain mobility or short mixing time.

Double quantum correlations

($^{13}C,^{13}C$) double quantum correlations represent intra-residue correlations seen in 2D double quantum–single quantum correlation spectra where the shift of a crosspeak in the indirect dimension corresponds to the sum of the chemical shifts of the two interacting nuclei. Double quantum correlations are created in the same way as regular intra-residue correlations; however, only one-bond correlations are considered, since only these are normally observed in experimental spectra.

Inter-residue (^{13}C,^{13}C) correlations

For (^{13}C,^{13}C) correlations between neighbouring residues, again, it can be specified which nuclei to consider depending on their distance from the backbone. In addition, the desired unique or maximum residue number difference for nuclei to be correlated can be chosen.

(^{15}N,^{13}C) correlations

(^{15}N,^{13}C) correlations can be defined as intra-residue N(i)-CA(i)-CX(i) or sequential N(i)-CO(i-1)-CX(i-1) correlations (with CX representing any other carbon nucleus in the respective residue). It can be specified whether all or only a subset of the 13C nuclei should be included in the prediction, depending on their distance from the backbone (yielding, e.g., only N(i)-CA(i) or also N(i)-CA(i)-CB(i) correlations etc.).

Through-space correlations

Through-space correlations are created by specifying a set of residues, a set of conformations with atom coordinates, and a pairwise distance cut-off up to which correlations should be taken into account. This type of correlation can act in different modes, allowing either direct distances between heteronuclei to be considered (yielding C-C or N-C through-space correlations) or the distances between protons directly attached to heteronuclei, yielding NHHC and CHHC correlations (364,365). A minimum distance threshold and a minimal residue number difference can also be specified.

Peakr can also predict intermolecular through-space correlations if a PDB file with multiple protein chains is provided. For example, symmetry equivalents can be generated from known crystal structures using pro-grams such as SwissPDBViewer or PyMol. These can then be analyzed as different chains in Peakr. In this way, intermolecular crosspeaks arising due to crystal packing interactions can be identified. In the context of multimeric proteins or protein fibrils, the prediction of intermolecular correlations can be particularly useful (344). Currently, the prediction of intermolecular through-space correlations requires the presence of two or more copies of the same protein in the PDB file; however, the generalization to heteromultimeric protein complexes could be implemented as well.

3.1.3.5 Calculated Spectra

Spectra represent a set of crosspeaks that are generated by applying a correlation object to a protein based on the selected lists of experimental or estimated chemical shifts. By default, spectra are calculated for all amino acids in the protein. If a provided PDB file contains several chains and/or models, calculated spectra can be displayed for each of these separately. In addition, the user can select subsets of residues to be displayed in a spectrum. One option is to select a certain sequence range based on residue number. Another possibility is to select residues within particular secondary structure types. If a PDB file is provided, secondary structure is assigned to individual residues using Stride (366); otherwise, it is predicted from the sequence using PsiPred v3.0 (367). Furthermore, any subset of amino acid types can be selected. This option can be used to simulate spectra of proteins that were expressed using forward or reverse labelling of certain amino acids (368,369).

In addition, the user can select one of several more complex isotope labelling schemes. In the current implementation, Peakr provides ^{13}C labelling patterns as obtained from protein expression using 1,3-^{13}C-glycerol, 2-^{13}C-glycerol, 1-^{13}C-glucose, and 2-^{13}C-glucose as sole carbon sources (370–372). The probabilities of individual carbon nuclei to be isotope-labelled as listed in these publications are translated into opacity values in Peakr’s spectrum display. Thus, a correlation between two nuclei that are less likely to be ^{13}C-labelled in the selected labelling scheme appears correspondingly less intense in the spectrum.

It is particularly helpful for analysis and interpretation of spectra if subsets of the same protein sequence are displayed at the same time, but with different colours or markers. Therefore, we provide a "clone"-button to copy every spectrum as often as needed. For each of these identical spectra, the user can then, e.g., select different sets of residues and update the displayed spectra accordingly.

3.1.3.6 Experimental Spectra

Processed experimental spectra can be read in and displayed, alone or overlaid with predicted spectra. Currently, Peakr accepts processed spectra in the format of Bruker XWinNMR or Topspin software (Bruker Biospin, Karlsruhe, Germany).

3.1.3.7 Output

Data storage

Visualized spectra (covering all predicted peaks or a region selected by the user) can be downloaded in PNG, PDF, or SVG format. The generated lists of predicted and experimental spectra can be downloaded as file archives referenced by a checksum string for further external investigation. Unmodified file archives can again be uploaded to Peakr for further analysis, including the generation of additional predicted spectra based on the previous settings. The individual checksum provides for security of the user's data. In addition, all data is automatically deleted from the server 24 hours after creation, to avoid long-term storage of potentially sensitive research data.

Lists

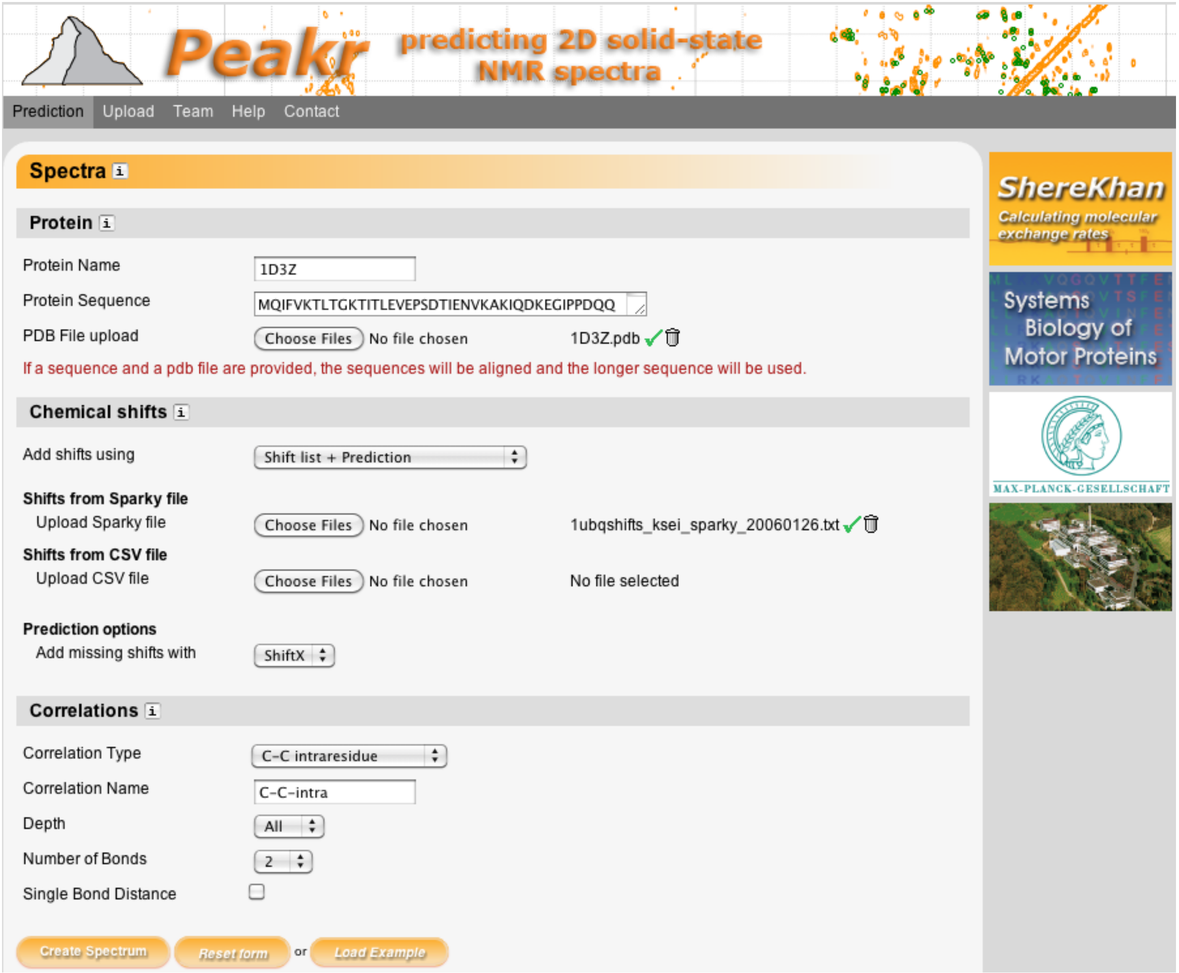
The crosspeak lists that are stored in a spectrum object can be retrieved as tab-delimited files. When comparing with an experimental spectrum, the intensity of the measured spectrum at the positions of the predicted cross-peaks or in a defined region around them can be included. This provides for a straightforward numerical comparison between prediction and experiment, which can be useful for model validation or generation of restraint lists for structure calculation.

Graphics

Peakr generates spectra with crosspeaks as PNG files that can be zoomed and browsed providing the functionality known from applications like Google Maps. Predicted and experimental spectra can be combined in one plot. Crosspeaks from different spectra are distinguished by colour. If a specific ^{13}C-labelling scheme is chosen, the probabilities of the correlations are displayed as opacities. Tooltips on every peak indicate the contributing nuclei and corresponding chemical shifts.

3.1.4 Results and Discussion

Peakr is available via a web interface (Figure 3.1-4). The workflow has been designed to provide the highest possible flexibility in terms of correlation types as well as protein regions and conformations chosen for comparison and analysis. The user can provide a protein sequence or a structural model, can choose between providing chemical shift lists, estimating shifts or combining these data, and can select various types of correlations that are used in most experimental setups. During spectrum prediction, Peakr first calculates correlations for all residues. Subsequently, any combination of residues can be selected for display, also from different chains or models that may be present in a PDB file (Figure 3.1-3). Here, the “clone” function provides an easy way to visualize various selections of residues while using the same settings for shifts and correlations.



Spectra

Protein

Protein Name: 1D3Z
Protein Sequence: MQIFVKTLTGKTITLEVEPSDTIENVKAKIQDKEGIPPDQQ
PDB File upload: Choose Files No file chosen 1D3Z.pdb ✓

If a sequence and a pdb file are provided, the sequences will be aligned and the longer sequence will be used.

Chemical shifts

Add shifts using: Shift list + Prediction

Shifts from Sparky file
Upload Sparky file: Choose Files No file chosen 1ubqshifts_ksei_sparky_20060126.txt ✓

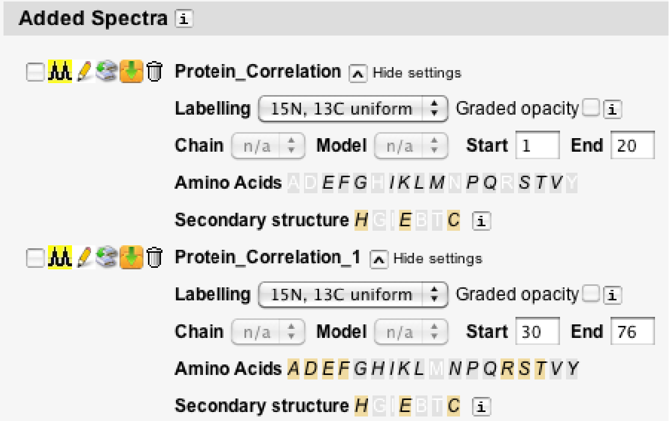
Shifts from CSV file
Upload CSV file: Choose Files No file chosen No file selected

Prediction options
Add missing shifts with: ShiftX

Correlations

Correlation Type: C-C intraresidue
Correlation Name: C-C-intra
Depth: All
Number of Bonds: 2
Single Bond Distance: ☐

Figure 3.1-4: Screenshot of the Peakr web application with the example data as input. The part in which spectra can be predicted is shown divided into the sections Protein, Chemical shifts, and Correlations.



Added Spectra

☐ Protein_Correlation
Labelling: 15N, 13C uniform; Graded opacity ☐
Chain: n/a; Model: n/a; Start: 1; End: 20
Amino Acids: A D E F G H I K L M N P Q R S T V Y
Secondary structure: H G I E B T C

☐ Protein_Correlation_1
Labelling: 15N, 13C uniform; Graded opacity ☐
Chain: n/a; Model: n/a; Start: 30; End: 76
Amino Acids: A D E F G H I K L M N P Q R S T V Y
Secondary structure: H G I E B T C

Figure 3.1-3: The screenshot shows the list of predicted spectra after cloning the first spectrum using the Clone function (third icon from left). Different parts of the protein sequence were selected (see Start and End values). The spectra show resonances from the first 20 residues and from residues 30 to 76, respectively. For the second spectrum, only some of the amino acid types present in the sequence were selected, which are thus highlighted in orange.

As an example for using Peakr, we demonstrate the prediction of intra-residue (^{13}C,^{13}C) correlation spectra of solid ubiquitin and their comparison with experimental data. The experimental dataset consisted of a (^{13}C,^{13}C) correlation spectrum of microcrystalline, uniformly [^{13}C,^{15}N] isotope-labelled ubiquitin prepared as described (362). It was recorded on a 700 MHz spectrometer (Bruker Biospin, Karlsruhe, Germany) using 7.8ms of DARR (dipolar assisted rotational resonance) ^{13}C-^{13}C mixing (373). Spectrum predictions were generated in Peakr based on the ubiquitin amino acid sequence in plain one-letter code format and using chemical shift assignments for this solid-phase ubiquitin preparation as reported (362) in Sparky list format. We generated three different (^{13}C,^{13}C) spectrum predictions based on these data using Peakr's intra-residue (^{13}C,^{13}C) correlation option, including sidechain nuclei up to Cδ and allowing for one-, two-, or three-bond correlations. Then, we compared these predictions to the experimental data by listing the intensities at the positions of the predicted peaks in the experimental spectrum. Based on visual inspection of the experimental spectrum, an intensity cut-off of 3000 was chosen to differentiate between signal and noise. 53.4% of one-bond, 17.9% of two-bond, and 2.4% of three-bond predicted correlations were found to be present in the spectrum based on this cut-off value, showing that the experimental spectrum yielded mainly one-bond and to some extent two-bond correlations under the conditions chosen.

With this approach for comparing prediction to experiment, spectral intensities at exactly the predicted peak position are returned. It can be useful to allow for more variability in peak positions to account for, e.g., small variations in sample conditions such as temperature or pH. To do so, Peakr can return the maximum intensity in a defined region of the experimental spectrum around each predicted peak. Allowing for ±0.2 p.p.m. (parts per million) variation in peak position when comparing the above one-bond (^{13}C,^{13}C) spectrum prediction with experiment, 74.5% of predicted one-bond correlations are found above the selected threshold, indicating good agreement between prediction and experiment (Figure 3.1-5, green dots).

For comparison, we predicted the same one-bond (^{13}C,^{13}C) correlation spectrum of ubiquitin using assignments reported for a different microcrystalline ubiquitin preparation (374), adjusted for the referencing offset of 2.01 p.p.m. between these assignments and the values used above. For this ubiquitin preparation employing 2-methyl-2,4-pentanediol as precipitant, rather than poly-(ethylene glycol) (362), significant chemical-shift differences were reported in some regions of the protein (362,375). Correspondingly, Peakr only finds 66.0% of all predicted one-bond peaks within a range of ±0.2 p.p.m. of a spectral intensity above the selected threshold Figure 3.1-5, blue dots). This example illustrates the use of Peakr to quickly assess the quality and state of a protein sample. Expected peaks that are absent from an experimental spectrum, weaker than expected, or shifted may hint at conformational differences or local motion (375). Thus, Peakr can directly point the user to spectral regions that merit further investigation.

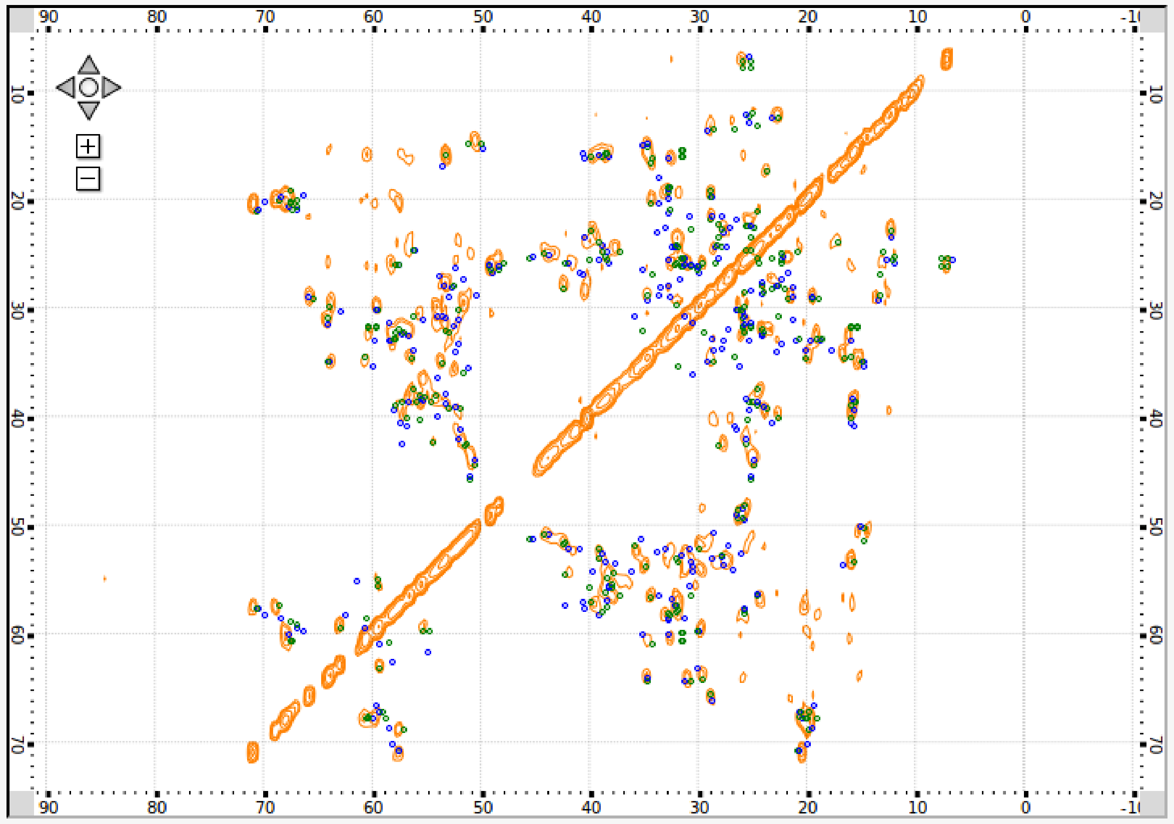


Figure 3.1-5: Screenshot of the Peakr spectrum display window showing the example discussed in the Results and Discussion section. Orange: experimental DARR (^{13}C,^{13}C) spectrum of microcrystalline ubiquitin. Green and blue: predicted one-bond (^{13}C,^{13}C) correlations based on assignments reported in (362) (green) and based on assignments from (374) (blue).

Several specific 13C labelling schemes that have been used in solid-state NMR studies in recent years are also implemented in Peakr. Labelling patterns obtained from using 1,3-13C- or 2-13C-glycerol (370) as well as 1-13C- or 2-13C-glucose as sole carbon sources (371,372) can be selected for spectrum prediction. This feature is demonstrated in Figure 3.1-6 for the same one-bond (13C,13C) correlation spectrum of ubiquitin as shown in green in Figure 3.1-5, using the 1,3-13C-glycerol labelling scheme. Peakr calculates opacity values of individual peaks according to the 13C labelling probabilities of the nuclei that give rise to the correlation. For the glycerol-based schemes, detailed labelling probabilities and isotopomer patterns are available (370) and implemented in Peakr, while the simplified scheme as shown in (371) is used for predicting spectra with 1-13C- and 2-13C-glucose-based labelling. Such predictions should be very helpful in assigning spectra of proteins expressed with one of these labelling patterns, effectively reducing the need for the user to consult tables of labelling schemes manually. In addition to the option to select only certain amino acid types for spectrum prediction, the selective 13C-labelling option in Peakr allows to assess which labelling method would best reduce spectral crowding for larger proteins with sizable spectral overlap, offering a fast and convenient way to guide protein expression strategies for further experiments.

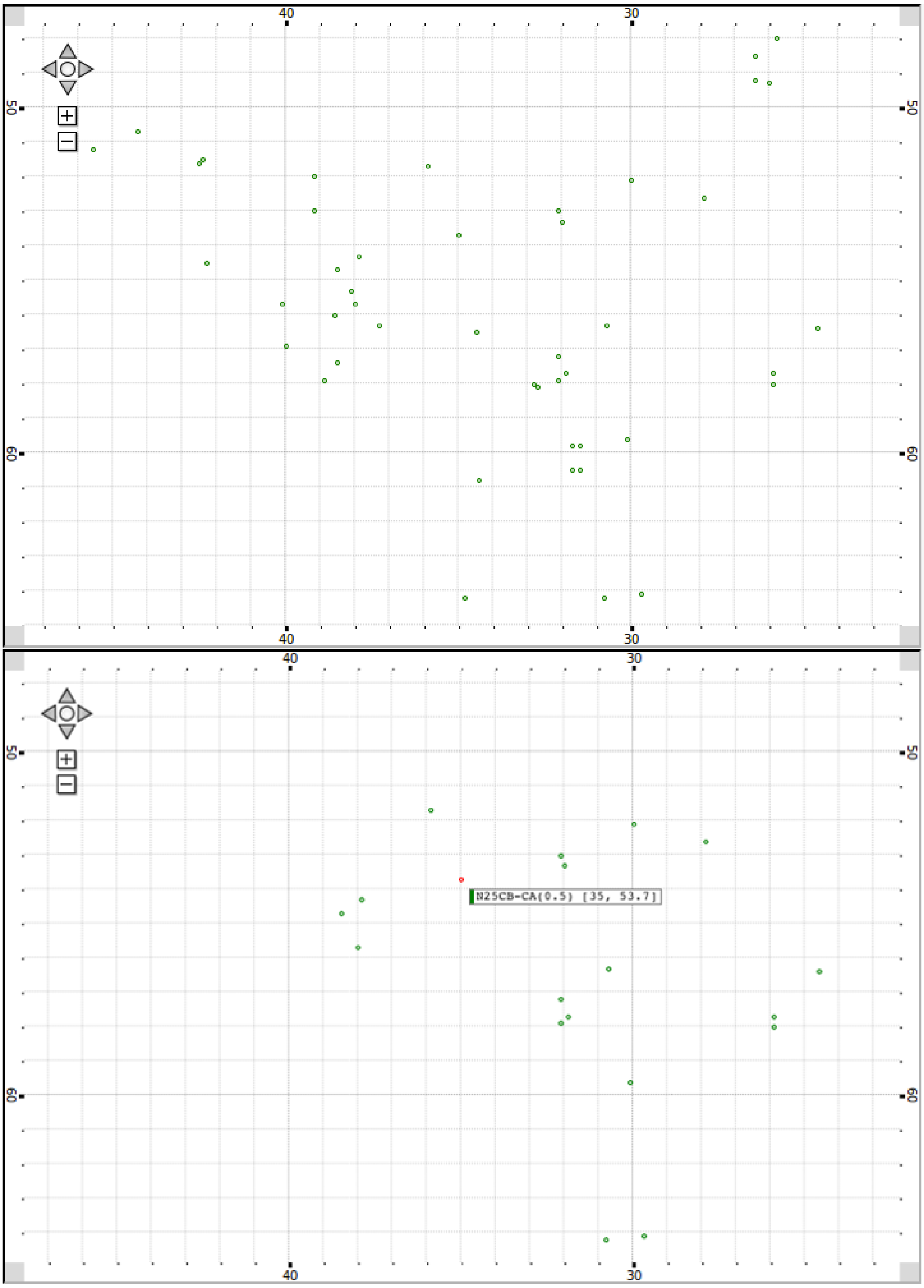


Figure 3.1-6: Screenshots of the Peakr spectrum display window demonstrating different labelling schemes. Shown is a region from the green spectrum of Figure 3.1-5, with predicted one-bond (^{13}C,^{13}C) correlations based on the assignments reported in (362). A) Predicted spectrum based on a uniformly ^{13}C labelled sample. B) Predicted spectrum based on the labelling scheme expected from using 1,3-^{13}C-glycerol as sole carbon source during protein expression. For one crosspeak, a tooltip shows assignment, chemical shifts and opacity value (corresponding to the probability that the corresponding nuclei are both isotope-labelled).

3.1.5 Conclusions

The Peakr software presented here can be of considerable help when analysing solid-state NMR spectra of proteins. It can predict 2D spectra for most of the common experimental setups. The predicted spectra can be helpful for suggesting resonance assignments and for deriving restraints for 3D structure calculations. As demonstrated in the case study, basic assumptions about a measured spectrum can be made in a matter of seconds, which can be useful in quality control of samples. In contrast to existing solutions, Peakr is very flexible and can use subsets of residues or nuclei to define spectra. This is especially valuable when reverse or selective labelling methods are used or when only a portion of the protein, e.g. the N-terminus, is of interest. Here, Peakr predictions can, for example, be used to assess which isotope labelling patterns would be optimal for a given protein in order to reduce spectral crowding. Peakr's ability to rapidly predict intra- and intermolecular through-space correlation spectra, with the same flexibility in choosing protein regions as well as upper distance limits to be considered, should be especially valuable in solid-state NMR structural studies. The option to compare predicted with measured spectra allows for estimating the degree of agreement between prediction and measurement. In this context, the percentage of predicted crosspeaks with a measured intensity above a given threshold can be seen as a simple figure of merit.

The Peakr framework is itself highly flexible and can easily accommodate extensions desired by its users. Future versions may thus, for example, be extended to predict 3D correlation spectra or proton-detected experiments, which are increasingly used in solid-state NMR (376,377), as well as to incorporate solution-state NMR correlation types.

In summary, Peakr has the power and flexibility to become a useful tool for routine analysis of solid-state NMR spectra. It is thus hoped that the community will adopt it and provide active feedback for further improvement and extension.

3.1.6 Acknowledgements

The ubiquitin spectrum was kindly provided by Dr. Hans Förster and Dr. Stefan Steuernagel (Bruker Biospin, Karlsruhe). We cordially thank Prof. Christian Griesinger for discussions and continuous support.

Funding: We thank the Max-Planck Society and especially the Department of NMR based Structural Biology, headed by Prof. Christian Griesinger, at the MPI for Biophysical Chemistry for generous financial support.

Conflict of interest: none declared.

3.2 ShereKhan – Calculating exchange parameters in relaxation dispersions data from CPMG experiments

Adam Mazur[1,†], Björn Hammesfahr[1,†], Christian Griesinger[1], Donghan Lee[1,*] and Martin Kollmar[1,*]

[1] Department of NMR-based Structural Biology, Max-Planck-Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany.
[*] To whom correspondence should be addressed.
[†] These authors contributed equally to the work and should be regarded as joint First Authors.

Bioinformatics

3.2.1 Abstract

Summary
Dynamics governing the function of biomolecule is usually described as exchange processes and can be monitored at atomic resolution with nuclear magnetic resonance (NMR) relaxation dispersion data. Here, we present a new tool for the analysis of CPMG relaxation dispersion profiles (ShereKhan). The web-interface of ShereKhan provides a user-friendly environment for the analysis.

Availability
A stable version of ShereKhan, the web application, and documentation are available at http://sherekhan.bionmr.org.

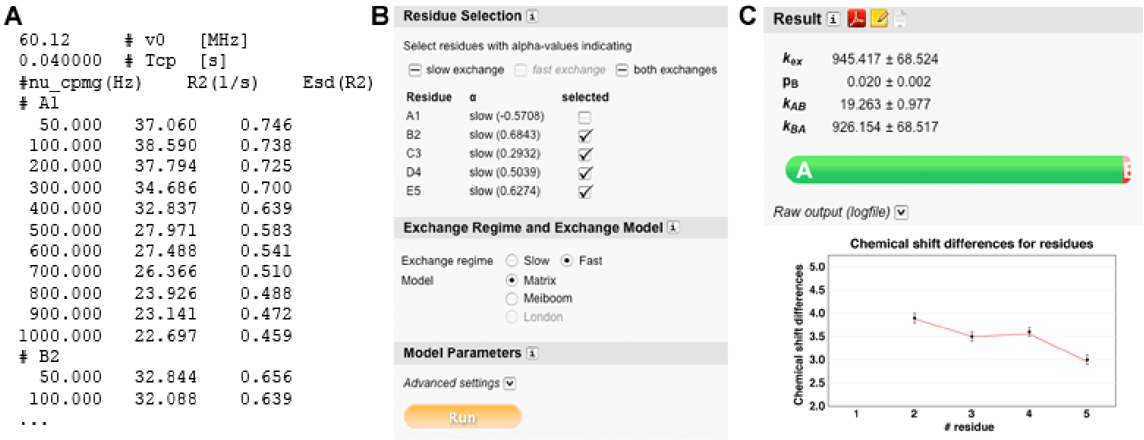
3.2.2 Introduction

Functions of biomolecules are governed by their dynamics of conformational interconversions sometimes induced by binding of a second partner. The kinetics of these processes needs to be explored. Normally, kinetics of the mentioned processes can be described with exchange processes. For example, proteins can be in conformational equilibrium, which may be characterized by two different conformations of the same molecule, or in chemical equilibrium, which may represent the bound and un-bound state of molecules to a binding partner. The environments of the nuclei may differ in each state and thus their nuclear magnetic resonance (NMR) parameters (e.g. chemical shifts, scalar or dipolar couplings, and relaxation) may also be different. We focus here on processes, in which at least one of the magnetically active nuclei exchanges between these states by chemical exchange.

The study of chemical exchange processes by NMR is already well established (378,379). Depending on the time scale at which the exchange process occurs, various NMR techniques such as line shape analysis, measurements of the spin-spin, or spin-lattice relation rates, the off-resonance saturation method, and pulse spin-echo techniques such as the Carr-Purcell-Meiboom-Gill (CPMG) experiment can be applied. In particular, the exchange processes occurring within the micro- to millisecond time window can be detected by the NMR relaxation dispersion experiments using CPMG or off-resonance irradiation approaches. Using the CPMG experiment, thermodynamic (relative populations of the species), kinetic (rates of exchange), and structural information (in form of chemical shifts) at atomic resolution can be obtained. Although CPMG relaxation dispersion experiments are in general well-established, their analysis is not a straightforward process. Currently available programs can be obtained upon request from the authors (e.g. GPMGFit), demand a number of software libraries (380), which are often not commonly available on personal computers and result in compatibility problems, or require proprietary software (GUARDD)(381). Therefore we developed ShereKhan, which is accessible through a web interface allowing a user-friendly selection of residues and suggesting models for the calculation of kinetic parameters like the relative populations of the species, the exchange rate, and chemical shift information from the CPMG relaxation dispersion data. ShereKhan assumes a global two state exchange process to fit the data with models for the slow or fast exchange.

3.2.3 Features

The ShereKhan web application provides an easy way to calculate exchange parameters (rates, relative populations, and their structural information) of molecules (Figure 3.2-1). The workflow has been designed to guide the user through the process from uploading data to the calculation of the exchange rates and populations of the states. The input file for a dataset is a simple tab-delimited text file containing R_2 relaxation rate values including error estimates at various ν_{CPMG} values for each residue (Figure 3.2-1A). In addition it must be specified with which resonance frequency and constant-time relaxation delay Tcp the data had been recorded. The residues must not be consecutive but should be numbered sequentially for calculating the chemical shift difference plot. ShereKhan accepts any number of datasets, e.g. data recorded at a pair or multiple field strengths. If the experimental datasets include relaxation dispersion data measured at two (or more) different magnetic fields, ShereKhan suggests the exchange regime (slow or fast exchange regime) for each given residue to facilitate the selection of an appropriate model (382). Subsequently, specific sets of residues (e.g. with a certain exchange regime) or any combination of residues can be selected. In these way single residues containing ambiguous data can be deselected before starting the calculations.



```
60.12      # v0    [MHz]
0.040000   # Tcp   [s]
#nu_cpmg(Hz)     R2(1/s)     Esd(R2)
# A1
  50.000    37.060     0.746
 100.000    38.590     0.738
 200.000    37.794     0.725
 300.000    34.686     0.700
 400.000    32.837     0.639
 500.000    27.971     0.583
 600.000    27.488     0.541
 700.000    26.366     0.510
 800.000    23.926     0.488
 900.000    23.141     0.472
1000.000    22.697     0.459
# B2
  50.000    32.844     0.656
 100.000    32.088     0.639
...
```

Select residues with alpha-values indicating

☐ slow exchange ☐ fast exchange ☐ both exchanges

Residue	α	selected
A1	slow (-0.5708)	☐
B2	slow (0.6843)	☑
C3	slow (0.2932)	☑
D4	slow (0.5039)	☑
E5	slow (0.6274)	☑

Exchange regime ○ Slow ● Fast

Model ● Matrix ○ Meiboom ○ London

k_{ex}	945.417 ± 68.524
p_B	0.020 ± 0.002
k_{AB}	19.263 ± 0.977
k_{BA}	926.154 ± 68.517

Figure 3.2-1: A) The input file for ShereKhan containing R_2 relaxation rate values including error estimates Esd(R_2) at various ν_{CPMG} values for each residue. B) Part of ShereKhan's user interface showing the residue selection and parameter options. C) As output SherKhan provides exchange regime dependent kinetic parameters, a chemical shift difference plot, and graphs with the fits (not shown).

3.2.3.1 Applying different exchange regimes

In the current implementation the user can choose between two exchange regimes (on the NMR time scale), slow or fast exchange (Figure 3.2-1B). After assigning the regime, the user can select a fitting model, namely the London model (383) or the Meiboom model (384) for the slow or the fast exchange regimes, respectively. Alternatively, users can use the Block-McConnell equation (385), which works for both regimes. Chemical shift variances and intrinsic R_2 relaxation rates are residue-specific but kinetic parameters can be fitted globally. Starting values for the parameters of the model can either be suggested by ShereKhan or be specified by the user.

3.2.3.2 Displaying and exporting results

For residues in the fast exchange regime, only the exchange rates and their population weighted chemical shift differences can be extracted from the fitting of NMR relaxation dispersion data whereas for residues in the slow exchange regime, all parameters such as rate constants, populations, and chemical shift differences can be obtained from the fitting. Fits of the model to the data can be browsed interactively or downloaded in various graphics formats.

3.2.4 Implementation

ShereKhan consists of a command-line program and a WWW-based tool, accessible using any modern web browser. The program was written in Python using the scipy (386) and matplotlib libraries (387) and the JSON parser. The web application framework is Ruby on Rails (79). In order to present the user with a feature rich interface the site makes extensive use of modern Web 2.0 techniques like Ajax (Asynchronous JavaScript and XML) using jQuery (388) and FancyBox (389). Interactive graphs are drawn using the graphical toolkit Protovis (256). User-uploaded data is stored temporary on the server and deleted when leaving the application.

3.2.5 Acknowledgements

We would like to thank David Ban, Marta Giao Carneiro, Dr. Thomas Michael Sabo and the members of the Griesinger department for helpful suggestions and discussions.

Funding:

Conflict of interest: none declared.

3.3 GenePainter: Aligning gene structures for phylogenetic analyses

Björn Hammesfahr[1,†], Florian Odronitz[1,†], Stefanie Mühlhausen[1], Stephan Waack[2] and Martin Kollmar[1,§]

[1] Department of NMR-based Structural Biology, Max-Planck-Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany.
[2] Institute of Computer Science, University of Göttingen, Goldschmidtstr. 7, 37077 Göttingen, Germany.
[§] Corresponding author
[†] These authors contributed equally to the work and should be regarded as joint First Authors.

BMC Bioinformatics

3.3.1 Abstract

Background

All sequenced eukaryotic genomes have been shown to possess at least a few introns. This includes those unicellular organisms, which were previously suspected to be intron-less. Therefore, gene splicing must have been present at least in the last common ancestor of the eukaryotes. To explain the evolution of introns, basically two mutually exclusive concepts have been developed. The introns-early hypothesis says that already the very first protein-coding genes contained introns while the introns-late concept asserts that eukaryotic genes gained introns only after the emergence of the eukaryotic lineage. A very important aspect in this respect is the conservation of intron positions within homologous genes of different taxa.

Results

GenePainter is a standalone application for mapping gene structure information onto protein multiple sequence alignments. Based on the multiple sequence alignments the gene structures are aligned down to single nucleotides. GenePainter accounts for variable lengths in exons and introns, respects split codons at intron junctions and is able to handle sequencing and assembly errors resulting in frame-shifts in exons and gaps in genome assemblies. Thus, even gene structures of considerably divergent proteins can properly be compared, as it is needed in phylogenetic analyses. Conserved intron positions can also be mapped to user-provided protein structures. For their visualization GenePainter provides scripts for the molecular graphics system PyMol.

Conclusion
GenePainter is a tool to analyse gene structure conservation providing various visualization options. A stable version of GenePainter for all operating systems as well as documentation and example data are available at http://www.motorprotein.de/genepainter.html.

Keywords
exon, intron, gene structure, evolution

3.3.2 Background

All eukaryotic genomes that have been sequenced so far have been shown to possess at least a few introns including the unicellular organisms that were previously suspected to be intron-less (390,391). These data has fuelled the lively debate between the introns-early and introns-late concepts that is on-going since the discovery of splicing (392). The introns-early hypothesis says that already the very first protein-coding genes contained introns while the introns-late concept asserts that eukaryotic genes gained introns only after the emergence of the eukaryotic lineage. Support for either of the concepts has been revealed by modelling the rates of intron gain and loss in eukaryotic genomes (393), by analysing the conservation of intron positions of example genes from a selection of genomes (394), or by population-genetic considerations (395). Intron position conservation has also been used to improve gene predictions (396) and multiple protein sequence alignments (397), and to reconstruct ancient genes (24).

A few software packages are available with which the conservation of intron positions can be analysed. Exalign is a software using only gene structure information to reveal intron conservation (398). Exalign uses gene structures from RefSeq gene annotations and from user definitions and calculates an alignment based on exon lengths and reading frames. However, Exalign fails if exon lengths between genes do not match. This is normally the case if genes encode less conserved proteins (e.g. differing in the lengths of surface loop regions) or if genes from more divergent species, which have been subject to complex intron loss and gain events, are compared. Thus, it is beneficial to use the information contained in protein sequence alignments. Tools that combine protein multiple sequence alignment (MSA) data and gene structures are CIDA/CIWOG (399), Malin (400), and scripts developed for large-scale analyses (401,402). CIDA/CIWOG comes with a web interface coupled to a database while Malin requires a species phylogeny as starting point. XdomView is the only software combining protein structures with intron and domain positions (403). In XdomView the user can specify a PDB-ID. Domain definitions from SCOP, CATH, DALI, 3DEE, and MMDB are mapped to the specified structure. In addition, the protein sequence from the PDB file is used to identify eukaryotic homologs in the ExInt database to map intron positions and phase. However, XdomView is strongly limited by accepting only PDB codes as input, and the reference databases are far out of date (PDB of June 2003, SCOP release of March 2003, ExInt based on Genbank 122 of February 2001).

With GenePainter we developed a tool that combines protein MSA data, gene and protein structures. GenePainter maps the intron positions obtained from the gene structures to the MSA taking reading frames into account. Additionally, conserved intron positions can be displayed in provided protein structures. The output can be used to compare gene structures from the exon/intron level down to the nucleotide sequences and to resolve and improve potentially ambiguous regions in the MSAs. GenePainter does not require any additional software/database to be installed and is unique compared to previous tools in its output options and the possible application in small- as well as large-scale analyses.

3.3.3 Implementation

GenePainter was written in Ruby (78), does not require any additional library, and can be used on any operating system (Additional File 3.3.10.1). The text and SVG output can be processed with any appropriate software. The output to visualize intron positions mapped to structures is scripts for PyMOL (404). The software as well as a comprehensive documentation can be found at http://www.motorprotein.de/genepainter.html.

Needleman-Wunsch

The mapping of intron positions and phases onto a PDB file is based on an alignment of the PDB sequence with one of the sequences from the protein MSA as reference. Thus, both the reference sequence and the chain of interest from the PDB file need to be specified. The alignment is calculated as described in (320). However, gaps at the end of the alignment are not penalized. This adaptation is of particular importance, as reference and protein sequence may vary greatly in length, possibly leading to an inappropriate alignment. Reasons for length differences can be full-length sequence in the alignment versus sequence of a single domain in the crystal structure, protein sequence in the alignment versus sequence joined to an expression/purification tag in the structure, and missing parts in the structure due to missing electron density.

3.3.4 Results and Discussion

To demonstrate GenePainter we use part of the coronin dataset published recently (126). Coronins are a family of actin remodelling proteins consisting of a conserved β-propeller domain that comprises the N-terminal two thirds of the sequences, a unique region, which is only conserved within closely related species, and a short C-terminal coiled-coil region that mediates trimerization. Also, a protein structure is available comprising the β-propeller domain and part of the unique region (204). GenePainter needs Fasta formatted protein MSAs and gene structure information stored in YAML files as input (Figure 3.3-1, Additional File 3.3.10.1). Optionally, a protein structure can be provided in PDB format. The gene structures can most easily be obtained by using the WebScipio (13,182) web interface or via the WebScipio web service by using the provided `gene_scan.rb` script. The latter option requires the user to specify species names and genome assemblies, which are easier to select via the WebScipio web interface. The advantage of using WebScipio is its ability to predict protein sequences and reconstruct gene structures in cross-species searches (182) and thus the possibility to easily extend the input data by adding genes from related species. Also, WebScipio can cope with genome assembly problems like assembly gaps and sequencing errors leading to frame shifts and in-frame stop codons in exons. In the current implementation, other file formats describing gene structures like GFF (405) cannot be used as alternative input files for GenePainter. This is due to the fact that GFF files normally do not contain DNA sequence and therefore do not provide all necessary information. Optionally, alignment limits can be defined in GenePainter. This is particularly useful when comparing specific regions and domains of multi-domain proteins separately.

Because GenePainter compares gene structures based on multiple sequence alignments of proteins it can be used to analyse proteins of any degree of similarity. The coronins from the sample data comprise sequences from apicomplexans, fungi and mammals. Accordingly, the similarity of the gene structures is not obvious at first glance (Figure 3.3-1A). By scaling the exons and introns the similarity of exon lengths between homologs of closely related organisms becomes suggestive (Figure 3.3-1B. Note the different scaling of exons and introns in this figure.). Exons and introns were scaled up and down, respectively, so that the average length of the exons equals the average length of the introns. GenePainter maps intron positions including phase to the sequences as provided in the multiple sequence alignments.

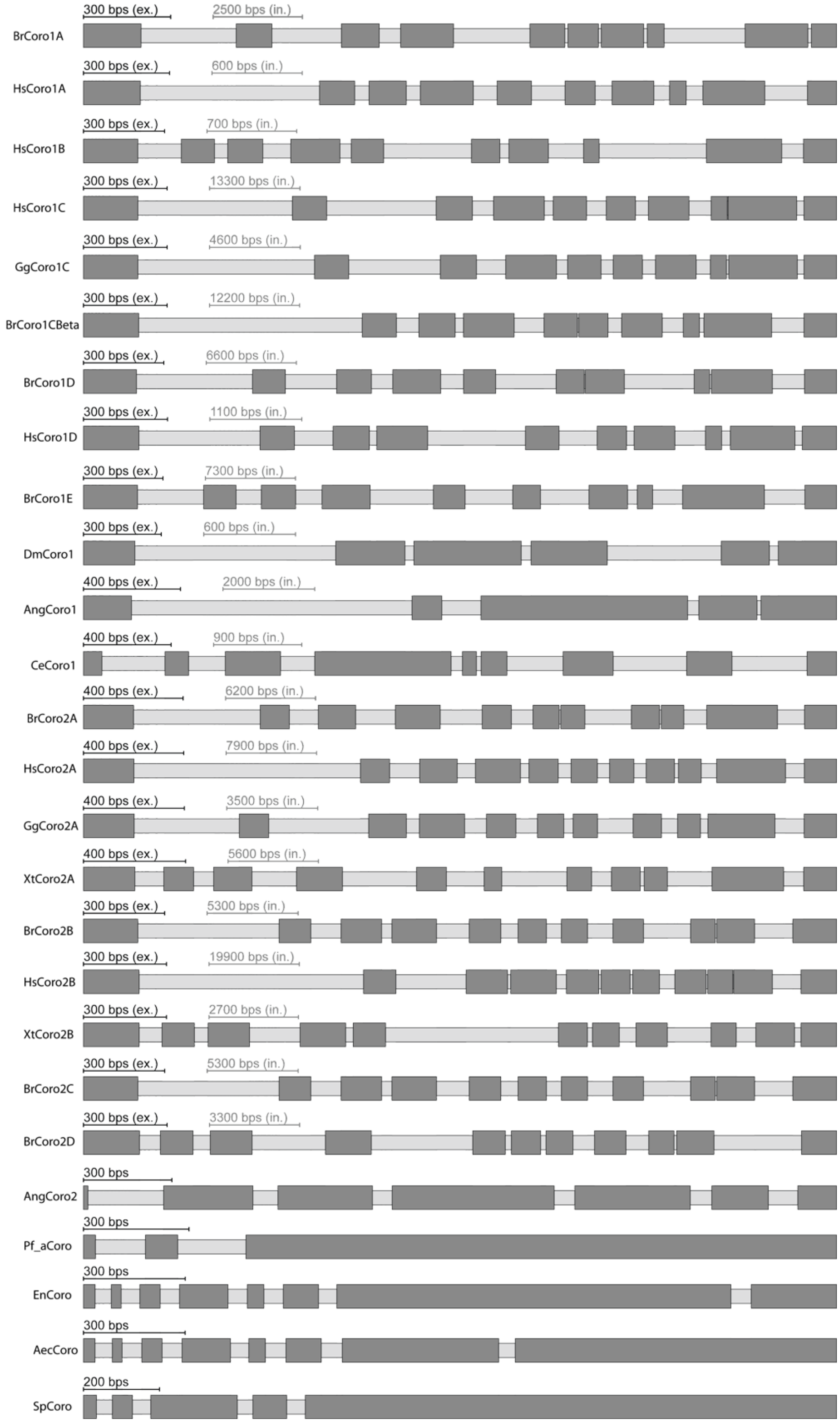


Figure 3.3-1: ***Gene structure schemes of coronins*** *A) The schemes illustrate examples of coronin genes from human (Hs = Homo sapiens) and zebrafish (Br = Brachydanio rerio) Dark grey bars and light grey bars mark exons and introns, respectively. B) This figure illustrates examples of coronin genes from vertebrates (Hs = Homo sapiens, Br = Brachydanio rerio, Gg = Gallus gallus, Xt = Xenopus tropicalis), arthropods (Dm = Drosophila melanogaster, Ang = Anopheles gambiae), nematods (Ce = Caenorhabditis elegans) and the protozoan parasite Plasmodium falciparum (Pf_a). In order not to make small exons vanish when very large intronic stretches are present, the scaling of introns and exons is automatically balanced to make the picture visually meaningful (scale bars for exons and introns are given, respectively). Here, except for AngCoro2 and Pf_aCoro in all schemes the introns were scaled down and the exons scaled up so that the average length of the introns equals the average length of the exons.*

Gene structure alignments

The aligned gene structures can be analysed in various formats. The basic output format displays intron position in plain text, where exons and introns are represented by hyphen-minuses “-” and vertical bars “|”, respectively. This format is particularly useful for large-scale data (many positions and sequences in the MSAs, Figure 3.3-2A) and is independent of exon and intron lengths. Additional information can be added by using the `-n` option of GenePainter by which intron positions are represented by phase numbers instead of vertical bars “|” (Figure 3.3-2B). A gene structure alignment including exon and intron lengths is shown in Figure 3.3-3. It is immediately obvious that intron lengths vary considerably (compare human HsCoro2B and frog XtCoro2B for example). In addition, the scheme shows that the N-terminal β-propeller domain of coronin is highly conserved while the unique regions are variable and contain many gaps in the multiple sequence alignment (blue bars).

The gene structure information can also be incorporated into the protein MSA as additional lines (option `-a`) where intron positions are either displayed as vertical bars “|” or as numbers defining the phase of the respective introns (Figure 3.3-3B). This format is most useful if the MSA will be re-evaluated to identify miss-aligned positions and regions.

Visualizing conserved intron positions on protein structures

Gene structure conservation derived by GenePainter can further be mapped on protein structures (option `-pdb <file> [chain]`). Therefore, one of the proteins from the MSA (set by `-pdb_prot`) is taken as reference and aligned with the protein sequence from a PDB file. Based on this alignment, intron positions and phases are projected on the protein structure. If no reference gene and no chain are specified, the first sequence in the alignment and chain A will be used by default. GenePainter supplies two python scripts for execution within PyMol comprising all necessary steps (including loading the PDB) in order to display the mapped intron positions and phases. While the script `color_exons.py` colours residues based on the underlying gene structure (Figure 3.3-4A), the other highlights only intron phases (`color_splicesites.py`; Figure 3.3-4C). In this visualization, both the last and first residues of succeeding exons are coloured by a three-color scheme denoting the phases of the respective introns. In order to elucidate the conservation of the respective intron positions, by default only those positions that are conserved in more than 80% of all genes (parameter can be changed via `-consensus`) are considered for visualization (Figure 3.3-4B and Figure 3.3-4D). In both visualizations attention is focused on those parts of the structure, on which intron data are mapped. Unused chains and regions not mapped to the reference sequence like cloning artefacts and protein purification tags are displayed in grey.

A

```
>BrCoro1A        ---------------|-----|---|---|---|-|-------|---|-----------------------------------|-
>HsCoro1A        ---------------|-----|---|---|---|-|-------|---|---------------------------------|---
>HsCoro1B        ---------------|-----|---|---|---|-|-------|---|-----------------------------|-------
>HsCoro1C        ---------------|-----|---|---|---|-|-------|---|-----------------|-------------------
>GgCoro1C        ---------------|-----|---|---|---|-|-------|---|-----------------|-------------------
>BrCoro1Cbeta    ---------------|-----|---|---|---|-|-------|---|-----------------|-------------------
>BrCoro1D        ---------------|-----|---|---|---|-|-------|---|-------------------------------|-----
>HsCoro1D        ---------------|-----|---|---|---|-|-------|---|-----------------------------|-------
>BrCoro1E        ---------------|-----|---|---|---|-|-------|---|-------------------------------|-----
>DmCoro1         ---------------|-----------|-----------|---------------|-----------------------|-----
>AngCoro1        ---------------|-----|---------------------------------|-----------------|-----------
>CeCoro1         ---------|-----|---------|-----------------------|-|-------|-------|-------|---------
>BrCoro2A        ---------------|---|-------|---|-|-|-----|-----|-----|---------|---------------------
>HsCoro2A        ---------------|---|-------|---|-|-|-----|-----|-----|---------|---------------------
>GgCoro2A        ---------------|---|-------|---|-|-|-----|-----|-----|---------|---------------------
>XtCoro2A        ---------------|---|-------|---|-|-|-----|-----|-----|---------|---------------------
>BrCoro2B        ---------------|---|-------|---|-|-|-----|-----|-----|---------|---------------------
>HsCoro2B        ---------------|---|-------|---|-|-|-----|-----|-----|---------|---------------------
>XtCoro2B        ---------------|---|-------|---|-|-|-----|-----|-----|---------|---------------------
>BrCoro2C        ---------------|---|-------|---|-|-|-----|---|-------|-------|-----------------------
>BrCoro2D        ---------------|---|-------|---|-|-|-----|-----|-----|---------|---------------------
>AngCoro2        -|-----------------|-----------|-------------------------|-------------|-----|-------
>Pf_aCoro        -----|-------|-----------------------------------------------------------------------
>EnCoro          ---|---|---|-----|-|---|---------------------------------------------|---------------
>AecCoro         ---|---|---|-----|-|---|-------------|-----------------------------------------------
>SpCoro          ---|-----|---------|---|-------------------------------------------------------------
```

B

```
>BrCoro1A        ---------------0-----0---1---0---0-0-------2---0-----------------------------------0-
>HsCoro1A        ---------------0-----0---1---0---0-0-------2---0---------------------------------0---
>HsCoro1B        ---------------0-----0---1---0---0-0-------2---0-----------------------------0-------
>HsCoro1C        ---------------0-----0---1---0---0-0-------2---0-----------------0-------------------
>GgCoro1C        ---------------0-----0---1---0---0-0-------2---0-----------------0-------------------
>BrCoro1Cbeta    ---------------0-----0---1---0---0-0-------2---0-----------------0-------------------
>BrCoro1D        ---------------0-----0---1---0---0-0-------2---0-------------------------------0-----
>HsCoro1D        ---------------0-----0---1---0---0-0-------2---0-----------------------------0-------
>BrCoro1E        ---------------0-----0---1---0---0-0-------2---0-------------------------------0-----
>DmCoro1         ---------------0-----------0-----------0---------------0-----------------------0-----
>AngCoro1        ---------------0-----0---------------------------------0-----------------0-----------
>CeCoro1         ---------0-----0---------1-----------------------0-1-------2-------0-------0---------
>BrCoro2A        ---------------0---0-------0---0-0-0-----1-----0-----1---------0---------------------
>HsCoro2A        ---------------0---0-------0---0-0-0-----1-----0-----1---------0---------------------
>GgCoro2A        ---------------0---0-------0---0-0-0-----1-----0-----1---------0---------------------
>XtCoro2A        ---------------0---0-------0---0-0-0-----1-----0-----1---------0---------------------
>BrCoro2B        ---------------0---0-------0---0-0-0-----1-----0-----1---------0---------------------
>HsCoro2B        ---------------0---0-------0---0-0-0-----1-----0-----1---------0---------------------
>XtCoro2B        ---------------0---0-------0---0-0-0-----1-----0-----1---------0---------------------
>BrCoro2C        ---------------0---0-------0---0-0-0-----1---0-------1-------0-----------------------
>BrCoro2D        ---------------0---0-------0---0-0-0-----1-----0-----1---------0---------------------
>AngCoro2        -0-----------------0-----------0-------------------------1-------------0-----0-------
>Pf_aCoro        -----1-------0-----------------------------------------------------------------------
>EnCoro          ---1---0---0-----2-0---2---------------------------------------------0---------------
>AecCoro         ---1---0---0-----2-0---2-------------2-----------------------------------------------
>SpCoro          ---1-----0---------0---2-------------------------------------------------------------
```

Figure 3.3-2: ***Gene structure alignments.*** *The gene structures shown in Figure 3.3-1 were aligned with GenePainter. A) Gene structure alignment representing exons ("-") and introns ("|") independent of their lengths. "|" underneath each other indicate intron positions with the same phase at the same position in the multiple sequence alignment. B) Similar visualization as in A) except that the intron phases are given instead of the intron indicator "|". The outputs shown in A) and B) are plain text files and can be analysed with any text editor.*

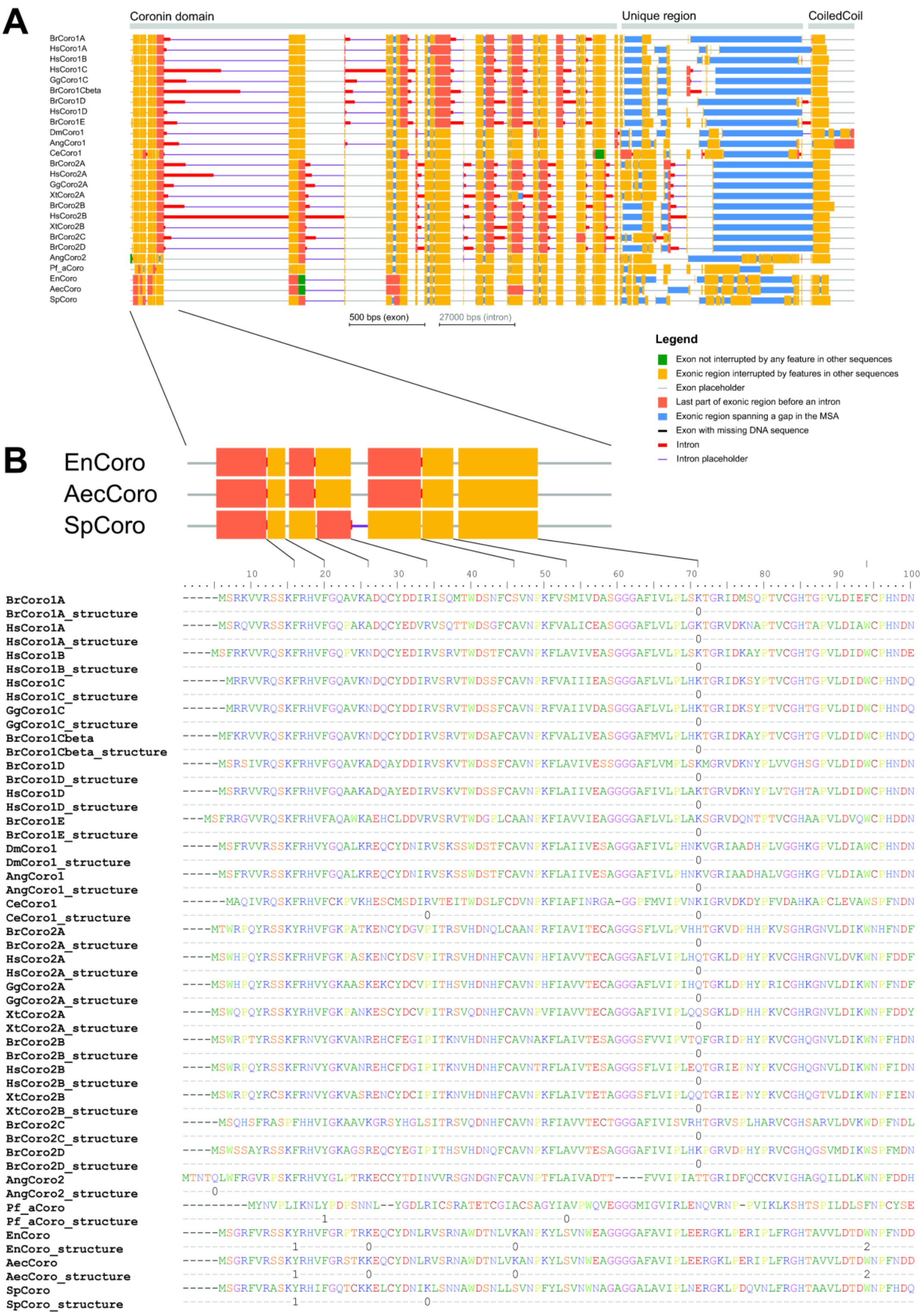


Figure 3.3-3: ***Nucleotide level alignment of gene structures.*** *A) Representation of the gene structures at the nucleotide level. Exons and introns are scaled that both represent 50% of the width of the figure. Without scaling, the introns would dominate the schemes. Red and magenta lines represent introns and intron gaps, respectively. Intron gaps are placeholders to fill the space of shorter introns compared to the longest intron at that position up to the next exon. The thick bars denote sequence within exons (green, orange and coral bars) and gap positions within exons (smaller blue bars) that were inserted into the protein sequences to adjust the multiple sequence alignment. Different colours for exonic sequences have been introduced to emphasize particular aspects like exons, which are not interrupted by sequence alignment gaps or introns in any of the other sequences (green bars) and the last uninterrupted parts of exonic sequence before introns (orange bars). The last option is particularly useful to identify the ends of exonic sequence before very short introns, or to identify introns in very huge alignments. Coral bars denote all other exonic sequence. Light grey lines symbolize placeholders within exonic sequences that are interrupted by introns in other sequences. All placeholders and markers for alignment gaps are added to optically align the corresponding exonic sequences beneath each other. B) Section of the gene structure alignment of A) with respect to the multiple sequence alignment to highlight the exon and intron features.*

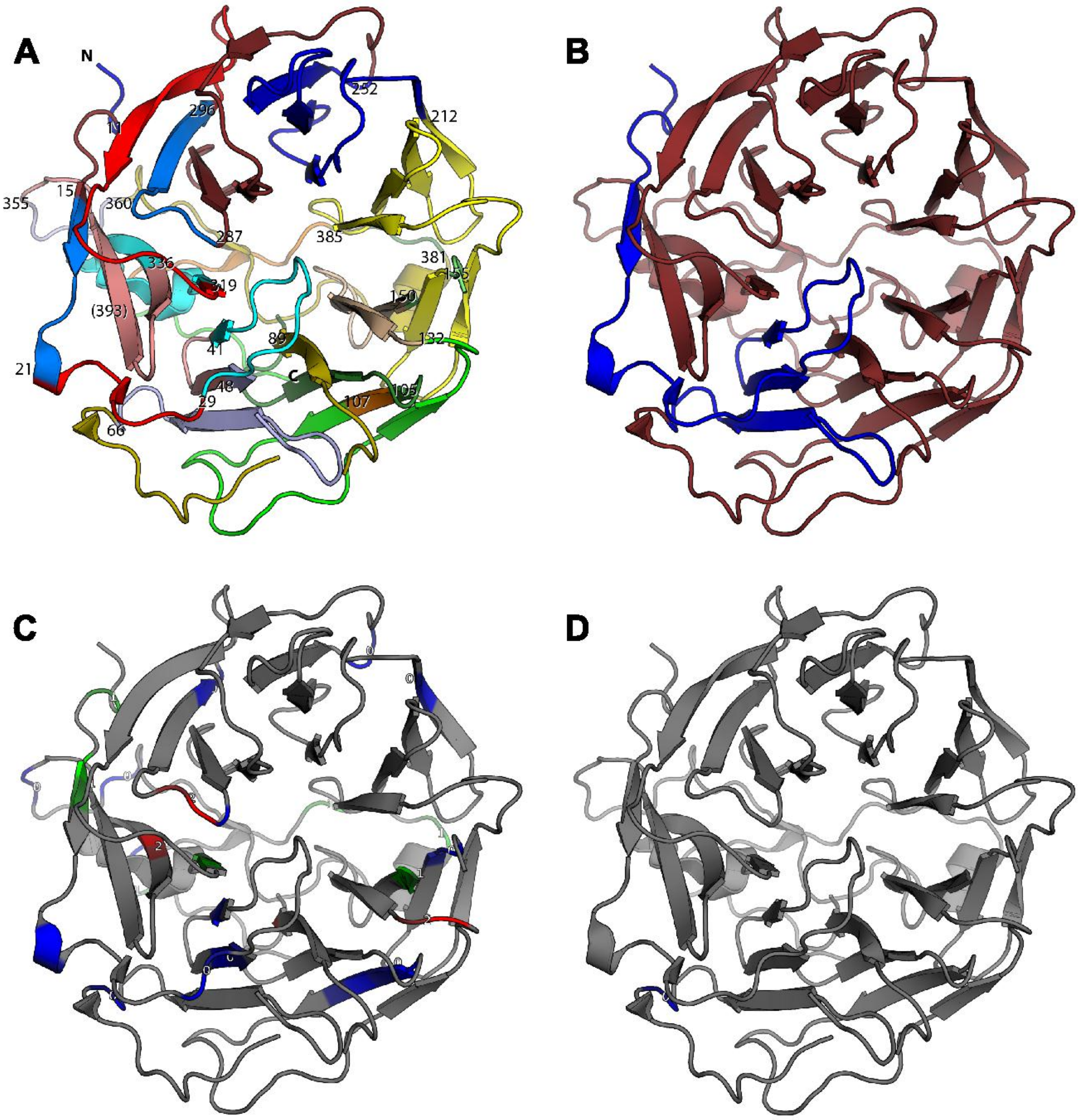


Figure 3.3-4: ***Visualization of gene structures on protein structures.*** *In this figure, intron positions and phases are visualized in a protein structure. The gene structure of human HsCoro1A (see Figure 3.3-1) was mapped onto the Coro1A structure from mouse (204). The two different output formats are shown for comparison. A) illustrates colouring of exons mapped to the protein structure (`color_exons.py`). For better orientation, the N- and C-termini and the positions of the last residues in each putative exon are given. The number in brackets denotes an intron position covered by another structural element. C) displays the phases of the introns (`color_splicesites.py`). Numbers indicate the respective intron phases. In both figures, introns occurring in any of the sequences within the MSA are shown (`-consenus 0`). Contrary, introns conserved in 80% of all proteins (default value) are shown in B) and D). Analogous to A) and C), B) refers to `color_exons` and D) to `color_splicesites`.*

3.3.5 Conclusion

GenePainter is a tool to analyse the conservation of gene structures of eukaryotic proteins. It aligns the gene structures to the respective protein sequences in a multiple sequence alignment. Gene structure conservation can be displayed in a binary format (exons and introns) and based on the nucleotide sequences. GenePainter can map gene structure conservation on protein structures and provides scripts for visualization in PyMol.

Therefore, GenePainter will be a valuable tool for gene structure guided improvements of multiple sequence alignments and for phylogenetic analyses including or focusing on the conservation of intron positions within eukaryotic genes.

3.3.6 Competing interests

The authors declare that they have no competing interests.

3.3.7 Availability and requirements

Project name: GenePainter
Project home page: http://www.motorprotein.de/genepainter
Operating system: Platform independent
Programming languages: Ruby
Software requirements: Ruby
License: GenePainter can be downloaded and used under a GNU General Public License.
Any restrictions to use by non-academics: Using GenePainter by non-academics requires permission.

3.3.8 Authors' contributions

MK, FO, and SW designed the project and set the requirements of the tool. BH and FO wrote the source code. SM added the mapping of the gene structure patterns to PDB files. MK, BH and SM wrote the manuscript. All authors read and approved the final manuscript.

3.3.9 Acknowledgements

We would like to thank Klas Hatje, Oliver Keller, Malte Hübner, and other members of the Kollmar and Waack groups for helpful suggestions and discussions. MK has been funded by grant KO 2251/6-1 and SW by grant WA 766/6-1 of the Deutsche Forschungsgemeinschaft. This work was partly supported by the Göttingen Graduate School of Neurosciences and Molecular Biosciences (DFG Grant GSC 226/1).

3.3.10 Additional files

3.3.10.1 Additional file 1

gene_painter.zip

This file contains the GenePainter software (`gene_painter.rb`) and a script to reconstruct genes via the web service of WebScipio (`gene_scan.rb`). It also includes example data (MSA, gene and protein structures) used to create the figures. This file is also available from the project homepage.

The zip file can be found in the corresponding publication or on the group homepage: http://www.motorprotein.de/genepainter/gene_painter.zip.

4 Conclusions

To understand a protein in detail, one possibility is to do a protein family analysis. Therefore, one starting protein sequence is used to search in one available related organism for homologues sequences. Afterwards, the starting sequence and the found homologues sequences have to be aligned. At this point, first similar sequence parts can be obtained. To add more sequences to the alignment, homologues sequences have to be search in many more related organisms. Adding and aligning more and more sequences, the existing alignment can be improved. Furthermore, biological information like specific domains of the protein helps to enhance the sequence alignment, too. The alignment should be corrected so that the domain sequences of each homolog are aligned to each other. A protein family analysis helps to determine if the sequence of the protein of interest is correct. Additionally, having only one sequence of one organism, it is not possible to make a statement when this protein occurred in the tree of life the first time. Doing such an analysis manually is time consuming. Therefore, it is faster and easier to use available sequences from databases. However, many automated annotated sequences found at NCBI are error prone. Unfortunately, many analyses are based on these wrong sequences. This is one reason why sequences from complete protein family analyses should be used, if available. The benefit of such an analysis is that all sequences of the analysis are compared to each other. Annotation errors are easier to recognize and to fix. Furthermore, for instance statistics of the number of amino acids, pI, molecular weight, and the level of conservation can be made for all sequences of the protein. With this knowledge, it is easier to plan mutation experiments of e.g. highly conserved amino acid positions. Such analyses were made for this dissertation.

Dynein is one of the three motor proteins. However, it is not able to bind and transport the corresponding cargo through the cell on its own. Therefore, the dynactin protein complex, composed of eleven different proteins (dynactin1 – dynactin6, Arp1, Arp11, actin, Capα, and Capβ), is necessary. To understand the evolution of this complex in eukaryotes, a protein family analysis for each of the protein was done. 3061 sequences from 478 organisms were collected. This analysis shows that the dynactin complex, like dynein, already existed in the last common ancestor of the eukaryotic branch. Furthermore, the loss of dynein, like in plants, accompanies with the loss of the dynactin complex.

At least seven proteins of this complex, dynactin1, dynactin2, dynactin4, dynactin5, the capping proteins, and Arp1, are necessary for the function of dynactin. These proteins were found in nearly every branch of the eukaryotic tree, where dynein is present. One exception is dynactin1, which was not found in Apicomplexa, Heterolobosea, and Apusozoa. The reason might be that the dynactin1 homologs of these branches do not have a CAP-Gly domain. This domain was used to find homologues sequences, because its core structure is highly conserved (Figure 2.1-2D). Furthermore, dynactin1 is the least conserved protein of the dynactin complex (Figure 2.1-1) and the TBLASTN and PSI-BLAST searches did not find the homologs in the three branches. However, this result does not mean that these branches do not have the possibility to produce a dynactin complex that interacts with dynein. The CAP-Gly domain interacts with the microtubule, but is not involved in the dynactin-dynein interaction.

Dynactin6 builds a sub-complex with dynactin5. But dynactin6 is not present in all branches of the eukaryotic tree, like in the yeasts. Comparing the found sequences of dynactin5 and dynactin6 from other organism, it is still possible to align them. This leads to the possibility that yeasts might build up the sub-complex of two dynactin5 proteins.

During the study, different duplication events were found. For dynactin1, independent duplication events were found in the Brachycera branch (including the Drosophila branch), Actinopterygii branch, and in some nematodes. Furthermore, duplications of dynactin1 were found in the fungis *Rhizopus arrhizus* and *Mucor cirinelloides*. The respective homologs grouped together in the phylogenetic tree that suggests that the duplication event happened before the separation of these two organisms. One further gene duplication event was found for Arp1. It happened at the origin of Vertebrates. During evolution, the Actinopterygii branch lost variant A, whereas the Amphibia and the Archosauria branch lost variant B. These two homologs are very similar, Arp1 mixing in the dynactin complex would not change the function at all.

Furthermore, we found that e.g. in vertebrates, the dynactin complex includes Arp11, whereas in yeasts, Arp10 is included. No eukaryotic genome was found including both homologs. We collected about 2300 Arp sequences from all eukaryotic branches and calculated phylogenetic trees. It was found that Arp10 and Arp11 grouped together. This leads to the result that Arp10 and Arp11 are orthologous proteins. According to HUGO we suggested to name both, Arp10 and Arp11, Arp10.

The sequence alignments of the dynactin proteins help to better understand the correlation between point mutations and diseases. For dynactin1, the point mutations G59S, G71R/E/A, T72P, and Q74P are related to different diseases. As illustrated in Figure 2.1-2D, G59 and G71 are highly conserved in all organisms. Furthermore, no sequence was found where a proline replaced T72, or Q74. All these amino acid positions are found in the CAP-Gly domain of dynactin1. Mutation of one of these positions destabilizes this domain and has direct influence to microtubule interaction, whereas the dynein-dynactin interaction is not interrupted.

Two proposed statements were revised. First, it has been proposed that dynactin3 is the least conserved protein of the complex (121). Therefore, sequences from four organisms were compared with the mouse homolog. Our analyses show that the least conserved protein of the dynactin complex is dynactin1, followed by dynactin3 and dynactin6 (Figure 2.1-1). Second, it has been proposed that dynactin4, dynactin5, and Arp11 have conserved alkaline pIs (97). For this statement, subunits for each mentioned protein were analysed from mouse, *Drosophila*, and *C. elegans*. Our analysis shows that this statement is true for dynactin5 (Figure 2.1-4). But for Arp10 (Arp11) and dynactin4, the pI's mainly distribute between a pH of about 6 to 7.5 and 6 to 8.5, respectively. With alkaline pI, a protein could interact electrostatically with negatively charged biomolecules like membrane lipids or acid cargoes. The recently reported interaction of dynactin5 with early endosomes is consistent with our result (150). The other pointed-end subunits might have electrostatic independent functions.

Like the analysis of the dynactin complex, the analysis of the coronin protein family gives an indication for the coronin inventory of the last common ancestor of the eukaryotic branch. The LCA must have contained a short coronin that was the origin of coronin class-1 and class-2. Furthermore, the LCA must have contained a class-3, and most probably a class-4 coronin. This result shows that the coronin introduced regulation of the actin cytoskeleton is a rather old function that already exists in the beginning of eukaryotes.

For a protein family analysis, as many organisms as possible should be analysed to find the homologues sequences. New homologs can improve the quality of already collected sequence alignment. However, if the found sequence of an organism might be wrong and the analysis of the corresponding genome helps to correct it, a different assembly should be analysed, if available. Assemblies might include errors, like missing genomic sequence. In order not to lose the overview of correlations between the sequences, proteins, organism, genomes and assemblies, all collected data should be stored in a database. To offer easy access to the collected data and the corresponding analyses, a web application is helpful. The necessary web browser is usually installed on every computer and no further program with its dependencies needs to be installed.

diArk is the most up to date database for eukaryotic genomes, organism and the corresponding meta data and analyses. It is the only database/web application that complies with the rapidly increasing genome data. Assembly related information about assembly version, assembly release date, completeness of the assembly, GC-content, assembly size, number of contigs, N50-value, accession numbers of contigs, genome assembly files, sequencing methods, and assembly method is stored. Knowing the used sequencing method and corresponding information like coverage, it is possible to recognize the quality of the genome. This information helps to select the best fitting assembly for upcoming analyses, if more than one is available.

diArk offers different search modules. They help users to select the data subset of interest. To keep track of the results, they are ordered by their types in seven different result tabs. Furthermore, different statistics, for instance the genome size/GC-content plot (Figure 2.3-7A), are calculated on the fly based on the search result. diArk generates more than 1200 unique visits per month that implies the importance of this application.

To provide access to the huge number of published manually annotated protein sequences of our group, CyMoBase was developed. For each sequence, different types of analyses are available. It is possible to compare biochemical characteristics of each protein sequence to all other sequences of the same protein and to the subset of sequences based on the search. The results are visualised directly in the web browser. Analyses about number of amino acids, isoelectric point, molecular weight, negatively/positively charged residues, extinction coefficients, atomic composition, instability index, aliphatic index, and grand average of hydropathicity (GRAVY) can be displayed. These analyses give different kinds of biochemical information that help to understand the protein sequence of interest. Furthermore, information about domain composition and gene structures computed with WebScipio are offered. One additional feature of CyMoBase is the possibility to display the sequence alignment directly in the web browser. Besides, it is possible to display a sequence and gene structure alignment that provides information about conserved intron positions. Like diArk, it is easy to search through the huge number of data provided by the database and to reach the data subset of interest. This is managed by different search modules that can be combined in any order. This allows users to get the sequences, analyses, and other information of interest fast and easy.

It does not suffice to provide access to this kind of data. Tools are necessary to search for new and to correct existing sequences. Furthermore, in depth analyses should be possible. WebScipio was developed for this reason. As above-mentioned, it reconstructs the exon intron gene structure using a genome and a protein sequence. WebScipio can reconstruct intron splice sites, very short exons, and exons spread over several contigs, correctly and is robust to sequencing errors. The possibility to change every parameter used during the search, the algorithm can be fine-tuned. This allows users not only to search in the corresponding genome of the protein sequence. It is shown that a cross species search is possible even if the last common ancestor is more than 100Myo years ago. Additionally, an algorithm is implemented in WebScipio to find and examine mutually exclusive exons that encode for the same structural part of the protein. This algorithm searches for exons next to each other that have roughly the same length, identical splice sites, are on the same reading frame, and have homologous sequence.

To analyse the gene structures of all sequences of a protein alignment, GenePainter was designed. It uses the gene structures calculated by WebScipio and the corresponding sequence alignment to produce an exon-intron alignment. The intron conservation level of all or a subset of sequences can be represented in different ways. Aligning errors in the multiple sequence alignment are directly visible. This information helps to improve the sequence alignment. Additionally, the intron conservation information assists to analyse the evolution of all introns of a gene. Moreover, with GenePainter, it is possible to map the intron information and their conservation level to a protein structure. It is possible to locate the part of the structure where exons with conserved introns can be found.

To understand the molecular function of proteins in detail, the protein sequence and the corresponding protein family analysis does not suffice alone. With a 3D protein structure, molecular functions might be explained at atomic resolution. One way to get the structure of a protein is to use the NMR technique. While the resonance lines and peaks produced by liquid state NMR are mostly sharp, the peaks produced by solid state NMR are not. Hence, the peaks displayed in a 2D spectrum produced by solid state are overlying and the following assignment of atoms is difficult. To improve the assignment process, predicted peaks based on a PDB, or a protein sequence can be used. Therefore, Peakr and the corresponding web application Webpeakr were designed. Peakr predicts 2D spectra for the common experimental settings in matter of seconds. It is possible to predict C-C inter- and intraresidue, C-C double quantum, N-C intraresidue, N-C interresidue, and through space correlations. For the prediction of the underlying chemical shifts, e.g. different prediction tools using different prediction methods can be selected. Furthermore, with Webpeakr it is possible to predict peaks of only e.g. the N-terminal part of the protein sequence, specific amino acid types, or structure elements. Additionally, the labelling pattern can be changed and its influence to the resulting peaks analysed. With this knowledge, a noise reduced experiment can be planed.

One additional benefit of Webpeakr is that only a modern web browser is required. No further program and its dependencies have to be installed on the local computer. It is possible to gain the results of interest fast and visually formatted. The complete session, including all uploaded files, settings, and selections can be downloaded and uploaded later on for further analysis.

The molecular function of proteins is often based on conformational changes induced by binding of another protein, or other biomolecule. To describe the kinetics, exchange processes can be analysed. NMR can be used to study the thermodynamic (relative populations of the species), kinetics (rates of exchange), and the structure (in form of chemical shifts) at atomic resolution of these processes, e.g. with CPMG experiments. The analysis of the CPMG results is difficult, especially for beginners. Therefore, ShereKhan was developed. It is accessible through a web interface. The user-friendly guided workflow allows first to upload the data. Second, a selection of every residue is available, as well as the selection of different models for the calculation of kinetics parameters. With these models and optional kinetic parameters, ShereKhan calculates the relative populations of the species, the exchange rate, and chemical shift information, based on the uploaded CPMG relaxation dispersion data. Third, the resulting values are listed directly in the browser. Furthermore, interactive plots providing exchange regime for each residue and chemical shift differences are visualized and can be downloaded in different graphical formats.

5 Acknowledgements

First, I thank my supervisor Dr. Martin Kollmar for the possibility to work in his group. Furthermore, I thank him for the pleasant and productive working atmosphere and mentoring.

Next, I thank Prof. Dr. Christian Griesinger for giving me the opportunity to work in his department and for providing coffee.

I also thank my theses committee members Prof. Dr. Burhard Morgenstern and Prof. Dr. Dirk Fasshauer for their support and helpful discussions.

I thank my colleagues Klas Hatje, Peggy Findeisen, Marcel Hellkamp and Stefanie Mühlhausen for all scientific and less scientific discussions during coffee breaks.

Especially, I thank my former colleague Dr. Christian Eckert for being himself and for his help. I will never forget Alpbach.

I thank my parents for their support during my educational development.

A special thank goes to my wonderful wife Janine. Without her, I would not be where I am. I thank her for all her love, her understanding, for everything.

Last but not least, I would like to thank my beautiful little daughter Lina-Adriana. Days and nights would be boring without her. She shows me every day what biology, evolution, and the meaning of life really means.

A Appendix

A.1 References

1. Hammesfahr B, Kollmar M. Evolution of the eukaryotic dynactin complex, the activator of cytoplasmic dynein. BMC Evol. Biol. 2012 Jun 22;12(1):95.

2. Goffeau A, Barrell BG, Bussey H, Davis RW, Dujon B, Feldmann H, et al. Life with 6000 Genes. Science. 1996 Oct 25;274(5287):546–67.

3. Sanger F, Nicklen S, Coulson AR. DNA sequencing with chain-terminating inhibitors. Proc. Natl. Acad. Sci. U.S.A. 1977 Dec;74(12):5463–7.

4. Dehal P, Satou Y, Campbell RK, Chapman J, Degnan B, De Tomaso A, et al. The draft genome of Ciona intestinalis: insights into chordate and vertebrate origins. Science. 2002 Dec 13;298(5601):2157–67.

5. Waterston RH, Lindblad-Toh K, Birney E, Rogers J, Abril JF, Agarwal P, et al. Initial sequencing and comparative analysis of the mouse genome. Nature. 2002 Dec 5;420(6915):520–62.

6. Aparicio S, Chapman J, Stupka E, Putnam N, Chia J-M, Dehal P, et al. Whole-genome shotgun assembly and analysis of the genome of Fugu rubripes. Science. 2002 Aug 23;297(5585):1301–10.

7. Hattori M, Fujiyama A, Taylor TD, Watanabe H, Yada T, Park HS, et al. The DNA sequence of human chromosome 21. Nature. 2000 May 18;405(6784):311–9.

8. Dunham I, Shimizu N, Roe BA, Chissoe S, Hunt AR, Collins JE, et al. The DNA sequence of human chromosome 22. Nature. 1999 Dec 2;402(6761):489–95.

9. Venter JC, Adams MD, Myers EW, Li PW, Mural RJ, Sutton GG, et al. The sequence of the human genome. Science. 2001;291:1304 – 1351.

10. Lander ES, Linton LM, Birren B, Nusbaum C, Zody MC, Baldwin J, et al. Initial sequencing and analysis of the human genome. Nature. 2001;409:860 – 921.

11. Luo C, Tsementzi D, Kyrpides N, Read T, Konstantinidis KT. Direct Comparisons of Illumina vs. Roche 454 Sequencing Technologies on the Same Microbial Community DNA Sample. PLoS ONE. 2012 Feb 10;7(2):e30087.

12. Smith DJ, Query CC, Konarska MM. “Nought may endure but mutability”: spliceosome dynamics and the regulation of splicing. Mol. Cell. 2008 Jun 20;30(6):657–66.

13. Odronitz F, Pillmann H, Keller O, Waack S, Kollmar M. WebScipio: an online tool for the determination of gene structures using protein sequences. BMC Genomics. 2008;9:422.

14. Keren H, Lev-Maor G, Ast G. Alternative splicing and evolution: diversification, exon definition and function. Nat Rev Genet. 2010;11:345 – 355.

15. Pan Q, Shai O, Lee LJ, Frey BJ, Blencowe BJ. Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. Nature Genetics. 2008;40(12):1413–5.

16. Alekseyenko AV, Kim N, Lee CJ. Global analysis of exon creation versus loss and the role of alternative splicing in 17 vertebrate genomes. RNA. 2007;13:661 – 670.

17. Sugnet CW, Kent WJ, Ares M, Haussler D. Transcriptome and genome conservation of alternative splicing events in humans and mice. Pac Symp Biocomput. 2004;66 – 77.

18. Kim E, Goren A, Ast G. Alternative splicing: current perspectives. Bioessays. 2008;30:38 – 47.

19. Fox-Walsh KL, Dou Y, Lam BJ, Hung S, Baldi PF, Hertel KJ. The architecture of pre-mRNAs affects mechanisms of splice-site pairing. PNAS. 2005 Nov 8;102(45):16176–81.

20. Koren E, Lev-Maor G, Ast G. The Emergence of Alternative 3′ and 5′ Splice Site Exons from Constitutive Exons. PLoS Comput Biol. 2007 May 25;3(5):e95.

21. Sakabe NJ, Souza SJ de. Sequence features responsible for intron retention in human. BMC Genomics. 2007 Feb 26;8(1):59.

22. Black DL. Mechanisms of alternative pre-messenger RNA splicing. Annu Rev Biochem. 2003;72:291 – 336.

23. Ast G. How did alternative splicing evolve? Nature Reviews Genetics. 2004 Oct 1;5(10):773–82.

24. Odronitz F, Kollmar M. Comparative genomic analysis of the arthropod muscle myosin heavy chain genes allows ancestral gene reconstruction and reveals a new type of “partially” processed pseudogene. BMC Mol. Biol. 2008;9:21.

25. Doherty GJ, McMahon HT. Mediation, Modulation, and Consequences of Membrane-Cytoskeleton Interactions. Annual Review of Biophysics. 2008;37(1):65–95.

26. Fuchs E, Cleveland DW. A Structural Scaffolding of Intermediate Filaments in Health and Disease. Science. 1998 Jan 23;279(5350):514–9.

27. de Hostos EL, Rehfuess C, Bradtke B, Waddell DR, Albrecht R, Murphy J, et al. Dictyostelium mutants lacking the cytoskeletal protein coronin are defective in cytokinesis and cell motility. J Cell Biol. 1993;120:163 – 173.

28. Cai L, Holoweckyj N, Schaller MD, Bear JE. Phosphorylation of coronin 1B by protein kinase C regulates interaction with Arp2/3 and cell motility. J Biol Chem. 2005;280:31913 – 31923.

29. Gloss A, Rivero F, Khaire N, Muller R, Loomis WF, Schleicher M, et al. Villidin, a novel WD-repeat and villin-related protein from Dictyostelium, is associated with membranes and the cytoskeleton. Mol Biol Cell. 2003;14:2716 – 2727.

30. Vale RD, Milligan RA. The way things move: looking under the hood of molecular motor proteins. Science. 2000 Apr 7;288(5463):88–95.

31. Schliwa M, Woehlke G. Molecular motors. Nature. 2003;422(6933):759–65.

32. Geeves MA, Holmes KC. The molecular mechanism of muscle contraction. Advances in protein chemistry. 2005;71:161–93.

33. Vale RD. The molecular motor toolbox for intracellular transport. Cell. 2003;112(4):467–80.

34. Finn RD, Mistry J, Tate J, Coggill P, Heger A, Pollington JE, et al. The Pfam protein families database. Nucleic Acids Res. 2010 Jan;38(Database issue):D211–222.

35. CyMoBase - a database for cytoskeletal and motor proteins. Available from: http://www.cymobase.org

36. Scholey JM, Brust-Mascher I, Mogilner A. Cell division. Nature. 2003 Apr 17;422(6933):746–52.

37. Yumura S, Uyeda TQP. Myosins and cell dynamics in cellular slime molds. Int. Rev. Cytol. 2003;224:173–225.

38. Taylor RS, Weeds AG. The magnesium-ion-dependent adenosine triphosphatase of bovine cardiac Myosin and its subfragment-1. Biochem. J. 1976 Nov;159(2):301–15.

39. Stone D, Perry SV. Studies on the heterogeneity of subfragment-1 preparations. Isolation of a new proteolytic fragment of the heavy chain of myosin. Biochem. J. 1973 Jan;131(1):127–37.

40. Vale RD, Reese TS, Sheetz MP. Identification of a novel force-generating protein, kinesin, involved in microtubule-based motility. Cell. 1985 Aug;42(1):39–50.

41. Hirokawa N. Kinesin and Dynein Superfamily Proteins and the Mechanism of Organelle Transport. Science. 1998 Jan 23;279(5350):519–26.

42. Zhu C, Zhao J, Bibikova M, Leverson JD, Bossy-Wetzel E, Fan J-B, et al. Functional Analysis of Human Microtubule-based Motor Proteins, the Kinesins and Dyneins, in Mitosis/Cytokinesis Using RNA Interference. Mol. Biol. Cell. 2005 Jul 1;16(7):3187–99.

43. Hirokawa N, Noda Y, Tanaka Y, Niwa S. Kinesin superfamily motor proteins and intracellular transport. Nat. Rev. Mol. Cell Biol. 2009 Oct;10(10):682–96.

44. Kanai Y, Dohmae N, Hirokawa N. Kinesin Transports RNA: Isolation and Characterization of an RNA-Transporting Granule. Neuron. 2004 Aug 19;43(4):513–25.

45. Höök P, Vallee RB. The dynein family at a glance. J. Cell. Sci. 2006 Nov 1;119(Pt 21):4369–71.

46. Karki S, Holzbaur EL. Cytoplasmic dynein and dynactin in cell division and intracellular transport. Current Opinion in Cell Biology. 1999 Feb 1;11(1):45–53.

47. Gibbons IR. Cilia and flagella of eukaryotes. J Cell Biol. 1981 Dec 1;91(3):107–24.

48. Schroer TA, Sheetz MP. Two activators of microtubule-based vesicle transport. J. Cell Biol. 1991 Dec;115(5):1309–18.

49. Steffen W, Karki S, Vaughan KT, Vallee RB, Holzbaur EL, Weiss DG, et al. The involvement of the intermediate chain of cytoplasmic dynein in binding the motor complex to membranous organelles of Xenopus oocytes. Mol Biol Cell. 1997;8(10):2077–88.

50. Holleran EA, Ligon LA, Tokito M, Stankewich MC, Morrow JS, Holzbaur ELF. βIII Spectrin Binds to the Arp1 Subunit of Dynactin. J Biol Chem. 2001;276(39):36598–605.

51. King SJ, Schroer TA. Dynactin increases the processivity of the cytoplasmic dynein motor. Nat Cell Biol. 2000 Jan;2(1):20–4.

52. Ross JL, Wallace K, Shuman H, Goldman YE, Holzbaur EL. Processive bidirectional motion of dynein-dynactin complexes in vitro. Nat Cell Biol. 2006;8(6):562–70.

53. Culver-Hanlon TL, Lex SA, Stephens AD, Quintyne NJ, King SJ. A microtubule-binding domain in dynactin increases dynein processivity by skating along microtubules. Nat Cell Biol. 2006 März;8(3):264–70.

54. Kardon JR, Reck-Peterson SL, Vale RD. Regulation of the processivity and intracellular localization of Saccharomyces cerevisiae dynein by dynactin. Proceedings of the National Academy of Sciences of the United States of America. 2009;

55. Deacon SW, Serpinskaya AS, Vaughan PS, Fanarraga ML, Vernos I, Vaughan KT, et al. Dynactin is required for bidirectional organelle transport. J Cell Biol. 2003 Feb 3;160(3):297–301.

56. Berezuk MA, Schroer TA. Dynactin enhances the processivity of kinesin-2. Traffic. 2007;8(2):124–9.

57. Blangy A, Arnaud L, Nigg EA. Phosphorylation by p34cdc2 protein kinase regulates binding of the kinesin-related motor HsEg5 to the dynactin subunit p150. J Biol Chem. 1997;272(31):19418–24.

58. Crooks GE, Hon G, Chandonia JM, Brenner SE. WebLogo: a sequence logo generator. Genome Res. 2004;14:1188 – 1190.

59. Hammesfahr B, Odronitz F, Hellkamp M, Kollmar M. diArk 2.0 provides detailed analyses of the ever increasing eukaryotic genome sequencing data. BMC Res. Notes. 2011;4:338.

60. Johnson M, Zaretskaya I, Raytselis Y, Merezhuk Y, McGinnis S, Madden TL. NCBI BLAST: a better web interface. Nucleic Acids Res. 2008;36:W5–9.

61. BioEdit. Available from: http://www.mbio.ncsu.edu/bioedit/bioedit.html

62. Chenna R, Sugawara H, Koike T, Lopez R, Gibson TJ, Higgins DG, et al. Multiple sequence alignment with the Clustal series of programs. Nucleic Acids Res. 2003;31(13):3497–500.

63. Odronitz F, Becker S, Kollmar M. Reconstructing the phylogeny of 21 completely sequenced arthropod species based on their motor proteins. BMC Genomics. 2009;10:173.

64. Van de Peer Y, Maere S, Meyer A. 2R or not 2R is not the question anymore. Nat Rev Genet. 2010;11(2):166.

65. Koonin EV. Orthologs, paralogs, and evolutionary genomics. Annu. Rev. Genet. 2005;39:309–38.

66. Thompson JD, Gibson TJ, Higgins DG. Multiple sequence alignment using ClustalW and ClustalX. Curr Protoc Bioinformatics. 2002;Chapter 2:Unit 2 3.

67. Stamatakis A, Hoover P, Rougemont J. A rapid bootstrap algorithm for the RAxML Web servers. Syst. Biol. 2008 Oct;57(5):758–71.

68. Li W, Godzik A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics. 2006 Jul 1;22(13):1658–9.

69. Talavera G, Castresana J. Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. Syst Biol. 2007 Aug;56(4):564–77.

70. Darriba D, Taboada GL, Doallo R, Posada D. ProtTest 3: fast selection of best-fit models of protein evolution. Bioinformatics. 2011 Apr 15;27(8):1164–5.

71. Akaike H. A new look at the statistical model identification. IEEE Transactions on Automatic Control. 1974 Dec;19(6):716 – 723.

72. Ronquist F, Huelsenbeck JP. MrBayes 3: Bayesian phylogenetic inference under mixed models. Bioinformatics. 2003 Aug 12;19(12):1572–4.

73. PostgreSQL. Available from: http://www.postgresql.org

74. Finn RD, Clements J, Eddy SR. HMMER web server: interactive sequence similarity searching. Nucleic Acids Research. 2011 May 18;39(Web Server):W29–W37.

75. NCBI LinkOut. Available from: http://www.ncbi.nlm.nih.gov/projects/linkout/

76. Encyclopedia of Life. Available from: http://eol.org/

77. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, et al. Gapped BLAST and PSI-BLAST: A New Generation of Protein Database Search Programs. Nucleic Acids Res. 1997 Sep 1;25(17):3389–402.

78. Ruby Programming Language. Available from: http://www.ruby-lang.org/

79. Ruby on Rails. Available from: http://rubyonrails.org

80. Apache Subversion. Available from: http://subversion.apache.org/

81. git. Available from: http://git-scm.com/

82. Ruby Version Manager (RVM). Available from: https://rvm.io/

83. Capistrano. Available from: https://github.com/capistrano/capistrano/wiki

84. Slony - a replication system for PostgreSQL. Available from: http://slony.info/

85. Han B, Liu Y, Ginzinger SW, Wishart DS. SHIFTX2: significantly improved protein chemical shift prediction. J. Biomol. NMR. 2011 May;50(1):43–57.

86. Shen Y, Bax A. SPARTA+: a modest improvement in empirical NMR chemical shift prediction by means of an artificial neural network. J. Biomol. NMR. 2010 Sep;48(1):13–22.

87. Gill SR, Schroer TA, Szilak I, Steuer ER, Sheetz MP, Cleveland DW. Dynactin, a conserved, ubiquitously expressed component of an activator of vesicle motility mediated by cytoplasmic dynein. J Cell Biol. 1991 Dec 15;115(6):1639–50.

88. Quintyne NJ, Schroer TA. Distinct cell cycle–dependent roles for dynactin and dynein at centrosomes. J Cell Biol. 2002 Oct 28;159(2):245–54.

89. Quintyne NJ, Gill SR, Eckley DM, Crego CL, Compton DA, Schroer TA. Dynactin Is Required for Microtubule Anchoring at Centrosomes. J Cell Biol. 1999 Oct 18;147(2):321–34.

90. Eaton BA, Fetter RD, Davis GW. Dynactin is necessary for synapse stabilization. Neuron. 2002;34(5):729–41.

91. Howell BJ, McEwen BF, Canman JC, Hoffman DB, Farrar EM, Rieder CL, et al. Cytoplasmic dynein/dynactin drives kinetochore protein transport to the spindle poles and has a role in mitotic spindle checkpoint inactivation. J Cell Biol. 2001;155(7):1159–72.

92. Lee WL, Oberle JR, Cooper JA. The role of the lissencephaly protein Pac1 during nuclear migration in budding yeast. J Cell Biol. 2003;160(3):355–64.

93. Merdes A, Ramyar K, Vechio JD, Cleveland DW. A Complex of NuMA and Cytoplasmic Dynein Is Essential for Mitotic Spindle Assembly. Cell. 1996 Nov 1;87(3):447–58.

94. Kim H, Ling S-C, Rogers GC, Kural C, Selvin PR, Rogers SL, et al. Microtubule binding by dynactin is required for microtubule organization but not cargo transport. J Cell Biol. 2007 Feb 26;176(5):641–51.

95. Schroer TA. Dynactin. Annu. Rev. Cell Dev. Biol. 2004;20:759–79.

96. Seal RL, Gordon SM, Lush MJ, Wright MW, Bruford EA. genenames.org: the HGNC resources in 2011. Nucleic Acids Res. 2011 Jan;39(Database issue):D514–519.

97. Eckley DM, Gill SR, Melkonian KA, Bingham JB, Goodson HV, Heuser JE, et al. Analysis of Dynactin Subcomplexes Reveals a Novel Actin-Related Protein Associated with the Arp1 Minifilament Pointed End. J Cell Biol. 1999 Oct 18;147(2):307–20.

98. Schafer DA, Gill SR, Cooper JA, Heuser JE, Schroer TA. Ultrastructural analysis of the dynactin complex: an actin-related protein is a component of a filament that resembles F-actin. J Cell Biol. 1994 Jul;126(2):403–12.

99. Waterman-Storer CM, Karki S, Holzbaur EL. The p150Glued component of the dynactin complex binds to both microtubules and the actin-related protein centractin (Arp-1). Proc Natl Acad Sci U S A. 1995;92(5):1634–8.

100. Akhmanova A, Hoogenraad CC. Microtubule plus-end-tracking proteins: mechanisms and functions. Curr Opin Cell Biol. 2005;17(1):47–54.

101. Weisbrich A, Honnappa S, Jaussi R, Okhrimenko O, Frey D, Jelesarov I, et al. Structure-function relationship of CAP-Gly domains. Nat Struct Mol Biol. 2007;14(10):959–67.

102. Vaughan KT, Vallee RB. Cytoplasmic dynein binds dynactin through a direct interaction between the intermediate chains and p150Glued. The Journal of Cell Biology. 1995 Dec 15;131(6):1507 –1516.

103. Karki S, Holzbaur EL. Affinity chromatography demonstrates a direct binding between cytoplasmic dynein and the dynactin complex. J. Biol. Chem. 1995 Dec 1;270(48):28806–11.

104. Echeverri CJ, Paschal BM, Vaughan KT, Vallee RB. Molecular characterization of the 50-kD subunit of dynactin reveals function for the complex in chromosome alignment and spindle organization during mitosis. J. Cell Biol. 1996 Feb;132(4):617–33.

105. Jacquot G, Maidou-Peindara P, Benichou S. Molecular and functional basis for the scaffolding role of the p50/dynamitin subunit of the microtubule-associated dynactin complex. J Biol Chem. 2010;285(30):23019–31.

106. Melkonian KA, Maier KC, Godfrey JE, Rodgers M, Schroer TA. Mechanism of Dynamitin-mediated Disruption of Dynactin. J Biol Chem. 2007 Jul 6;282(27):19355–64.

107. Terasawa M, Toya M, Motegi F, Mana M, Nakamura K, Sugimoto A. Caenorhabditis elegans ortholog of the p24/p22 subunit, DNC-3, is essential for the formation of the dynactin complex by bridging DNC-1/p150Glued and DNC-2/dynamitin. Genes to Cells. 2010 Nov 1;15(11):1145–57.

108. Holleran EA, Tokito MK, Karki S, Holzbaur EL. Centractin (ARP1) associates with spectrin revealing a potential mechanism to link dynactin to intracellular organelles. J Cell Biol. 1996;135(6 Pt 2):1815–29.

109. Caviston JP, Holzbaur ELF. Microtubule motors at the intersection of trafficking and transport. Trends Cell Biol. 2006 Oct;16(10):530–7.

110. Lee IH, Kumar S, Plamann M. Null Mutants of the Neurospora Actin-related Protein 1 Pointed-End Complex Show Distinct Phenotypes. Mol Biol Cell. 2001 Jul 1;12(7):2195 –2206.

111. Plamann M, Minke PF, Tinsley JH, Bruno KS. Cytoplasmic dynein and actin-related protein Arp1 are required for normal nuclear distribution in filamentous fungi. J Cell Biol. 1994;127(1):139–49.

112. Robb MJ, Wilson MA, Vierula PJ. A fungal actin-related protein involved in nuclear migration. Mol Gen Genet. 1995;247(5):583–90.

113. Vierula PJ, Mais JM. A gene required for nuclear migration in Neurospora crassa codes for a protein with cysteine-rich, LIM/RING-like domains. Mol Microbiol. 1997 Apr 1;24(2):331–40.

114. Tinsley JH, Minke PF, Bruno KS, Plamann M. p150Glued, the largest subunit of the dynactin complex, is nonessential in Neurospora but required for nuclear distribution. Mol Biol Cell. 1996;7(5):731–42.

115. Bruno KS, Tinsley JH, Minke PF, Plamann M. Genetic interactions among cytoplasmic dynein, dynactin, and nuclear distribution mutants of Neurospora crassa. Proc Natl Acad Sci U S A. 1996;93(10):4775–80.

116. Amaro IA, Costanzo M, Boone C, Huffaker TC. The Saccharomyces cerevisiae Homolog of p24 Is Essential for Maintaining the Association of p150Glued With the Dynactin Complex. Genetics. 2008 Feb;178(2):703–9.

117. Clark SW, Rose MD. Arp10p Is a Pointed-End-associated Component of Yeast Dynactin. Mol Biol Cell. 2006 Feb;17(2):738–48.

118. Kahana JA, Schlenstedt G, Evanchuk DM, Geiser JR, Hoyt MA, Silver PA. The yeast dynactin complex is involved in partitioning the mitotic spindle between mother and daughter cells during anaphase B. Mol Biol Cell. 1998;9(7):1741–56.

119. Muhua L, Karpova TS, Cooper JA. A yeast actin-related protein homologous to that in vertebrate dynactin complex is important for spindle orientation and nuclear migration. Cell. 1994;78(4):669–79.

120. Holzbaur ELF, Hammarback JA, Paschal BM, Kravit NG, Pfister KK, Vallee RB. Homology of a 150K cytoplasmic dynein-associated polypeptide with the Drosophila gene Glued. Nature. 1991 Jun 13;351(6327):579–83.

121. Eckley DM, Schroer TA. Interactions between the Evolutionarily Conserved, Actin-related Protein, Arp11, Actin, and Arp1. Molecular Biology of the Cell. 2003 Jul 1;14(7):2645–54.

122. Skop AR, White JG. The dynactin complex is required for cleavage plane specification in early Caenorhabditis elegans embryos. Current Biology. 1998 Oct 8;8(20):1110–7.

123. Gönczy P, Pichler S, Kirkham M, Hyman AA. Cytoplasmic Dynein Is Required for Distinct Aspects of Mtoc Positioning, Including Centrosome Separation, in the One Cell Stage Caenorhabditis elegans Embryo. J Cell Biol. 1999 Oct 4;147(1):135–50.

124. Yandell M, Ence D. A beginner’s guide to eukaryotic genome annotation. Nat. Rev. Genet. 2012;13(5):329–42.

125. Odronitz F, Kollmar M. Drawing the tree of eukaryotic life based on the analysis of 2,269 manually annotated myosins from 328 species. Genome Biol. 2007;8(9):R196.

126. Eckert C, Hammesfahr B, Kollmar M. A holistic phylogeny of the coronin gene family reveals an ancient origin of the tandem-coronin, defines a new subfamily, and predicts protein function. BMC Evol. Biol. 2011;11:268.

127. Kollmar M, Lbik D, Enge S. Evolution of the eukaryotic ARP2/3 activators of the WASP family: WASP, WAVE, WASH, and WHAMM, and the proposed new family members WAWH and WAML. BMC Res Notes. 2012;5:88.

128. Gupta KK, Joyce MV, Slabbekoorn AR, Zhu ZC, Paulson BA, Boggess B, et al. Probing interactions between CLIP-170, EB1, and microtubules. J. Mol. Biol. 2010 Feb 5;395(5):1049–62.

129. Tian G, Lewis SA, Feierbach B, Stearns T, Rommelaere H, Ampe C, et al. Tubulin subunits exist in an activated conformational state generated and maintained by protein cofactors. J. Cell Biol. 1997 Aug 25;138(4):821–32.

130. Voloshin O, Gocheva Y, Gutnick M, Movshovich N, Bakhrat A, Baranes-Bachar K, et al. Tubulin chaperone E binds microtubules and proteasomes and protects against misfolded protein stress. Cell. Mol. Life Sci. 2010 Jun;67(12):2025–38.

131. Tokito MK, Holzbaur EL. The genomic structure of DCTN1, a candidate gene for limb-girdle muscular dystrophy (LGMD2B). Biochim Biophys Acta. 1998;1442(2-3):432–6.

132. Dixit R, Levy JR, Tokito M, Ligon LA, Holzbaur ELF. Regulation of Dynactin through the Differential Expression of p150Glued Isoforms. J Biol Chem. 2008 Nov 28;283(48):33611–9.

133. Honnappa S, Okhrimenko O, Jaussi R, Jawhari H, Jelesarov I, Winkler FK, et al. Key interaction modes of dynamic +TIP networks. Mol Cell. 2006;23(5):663–71.

134. Hayashi I, Wilde A, Mal TK, Ikura M. Structural Basis for the Activation of Microtubule Assembly by the EB1 and p150Glued Complex. Mol Cell. 2005 Aug 19;19(4):449–60.

135. Hayashi I, Plevin MJ, Ikura M. CLIP170 autoinhibition mimics intermolecular interactions with p150Glued or EB1. Nat Struct Mol Biol. 2007 Oktober;14(10):980–1.

136. McGrail M, Gepner J, Silvanovich A, Ludmann S, Serr M, Hays TS. Regulation of cytoplasmic dynein function in vivo by the Drosophila Glued complex. J Cell Biol. 1995;131(2):411–25.

137. Puls I, Jonnakuty C, LaMonte BH, Holzbaur EL, Tokito M, Mann E, et al. Mutant dynactin in motor neuron disease. Nat Genet. 2003;33(4):455–6.

138. Farrer MJ, Hulihan MM, Kachergus JM, Dachsel JC, Stoessl AJ, Grantier LL, et al. DCTN1 mutations in Perry syndrome. Nat Genet. 2009 Feb;41(2):163–5.

139. Ahmed S, Sun S, Siglin AE, Polenova T, Williams JC. Disease-Associated Mutations in the p150Glued Subunit Destabilize the CAP-gly Domain. Biochemistry. 2010;49(25):5083–5.

140. Yue L, Lu S, Garces J, Jin T, Li J. Protein Kinase C-regulated Dynamitin-Macrophage-enriched Myristoylated Alanine-Rice C Kinase Substrate Interaction Is Involved in Macrophage Cell Spreading. J Biol Chem. 2000;275(31):23948–56.

141. Maier KC, Godfrey JE, Echeverri CJ, Cheong FKY, Schroer TA. Dynamitin Mutagenesis Reveals Protein–Protein Interactions Important for Dynactin Structure. Traffic. 2008 Apr 1;9(4):481–91.

142. Garces JA, Clark IB, Meyer DI, Vallee RB. Interaction of the p62 subunit of dynactin with Arp1 and the cortical actin cytoskeleton. Curr Biol. 1999;9(24):1497–500.

143. Karki S, Tokito MK, Holzbaur ELF. A Dynactin Subunit with a Highly Conserved Cysteine-rich Motif Interacts Directly with Arp1. J Biol Chem. 2000 Feb 18;275(7):4834–9.

144. Eisenhaber B, Chumak N, Eisenhaber F, Hauser M-T. The ring between ring fingers (RBR) protein family. Genome Biol. 2007;8(3):209.

145. Kadrmas JL, Beckerle MC. The LIM domain: from the cytoskeleton to the nucleus. Nat Rev Mol Cell Biol. 2004 Nov;5(11):920–31.

146. Krishna SS, Majumdar I, Grishin NV. Structural classification of zinc fingers. Nucleic Acids Research. 2003 Jan 15;31(2):532–50.

147. Schmeichel KL, Beckerle MC. Molecular dissection of a LIM domain. Mol Biol Cell. 1997 Feb 1;8(2):219–30.

148. Cole C, Barber JD, Barton GJ. The Jpred 3 secondary structure prediction server. Nucleic Acids Res. 2008 May 19;36(Web Server):W197–W201.

149. Scannell DR, Frank AC, Conant GC, Byrne KP, Woolfit M, Wolfe KH. Independent sorting-out of thousands of duplicated gene pairs in two yeast species descended from a whole-genome duplication. Proc. Natl. Acad. Sci. U.S.A. 2007 May 15;104(20):8397–402.

150. Zhang J, Yao X, Fischer L, Abenza JF, Peñalva MA, Xiang X. The p25 subunit of the dynactin complex is required for dynein–early endosome interaction. The Journal of Cell Biology. 2011 Jun 27;193(7):1245 –1255.

151. Lees-Miller JP, Helfman DM, Schroer TA. A vertebrate actin-related protein is a component of a multisubunit complex involved in microtubule-based vesicle motility. Nature. 1992;359(6392):244–6.

152. Clark SW, Meyer DI. Centractin is an actin homologue associated with the centrosome. Nature. 1992;359(6392):246–50.

153. Clark SW, Staub O, Clark IB, Holzbaur EL, Paschal BM, Vallee RB, et al. Beta-centractin: characterization and distribution of a new member of the centractin family of actin-related proteins. Mol Biol Cell. 1994 Dec 1;5(12):1301–10.

154. Kozielski F, Riaz T, DeBonis S, Koehler CJ, Kroening M, Panse I, et al. Proteome analysis of microtubule-associated proteins and their interacting partners from mammalian brain. Amino Acids. 2010 Jun 22;41(2):363–85.

155. Poch O, Winsor B. Who’s Who among the Saccharomyces cerevisiae Actin-Related Proteins? A Classification and Nomenclature Proposal for a Large Family. Yeast. 1998 Dec 4;13(11):1053–8.

156. Muller J, Oma Y, Vallar L, Friederich E, Poch O, Winsor B. Sequence and Comparative Genomic Analysis of Actin-related Proteins. Molecular Biology of the Cell. 2005 Dec 1;16(12):5736 –5748.

157. Simpson AGB, Inagaki Y, Roger AJ. Comprehensive multigene phylogenies of excavate protists reveal the evolutionary positions of “primitive” eukaryotes. Mol Biol Evol. 2006 Mar;23(3):615–25.

158. Parfrey LW, Grant J, Tekle YI, Lasek-Nesselquist E, Morrison HG, Sogin ML, et al. Broadly sampled multigene analyses yield a well-resolved eukaryotic tree of life. Syst Biol. 2010 Oct;59(5):518–33.

159. Burki F, Shalchian-Tabrizi K, Minge M, Skjaeveland A, Nikolaev SI, Jakobsen KS, et al. Phylogenomics reshuffles the eukaryotic supergroups. PLoS ONE. 2007;2(8):e790.

160. Nozaki H, Maruyama S, Matsuzaki M, Nakada T, Kato S, Misawa K. Phylogenetic positions of Glaucophyta, green plants (Archaeplastida) and Haptophyta (Chromalveolata) as deduced from slowly evolving nuclear genes. Mol Phylogenet Evol. 2009 Dec;53(3):872–80.

161. Keeling PJ. Chromalveolates and the evolution of plastids by secondary endosymbiosis. J Eukaryot Microbiol. 2009 Feb;56(1):1–8.

162. Hackett JD, Yoon HS, Li S, Reyes-Prieto A, Rümmele SE, Bhattacharya D. Phylogenomic analysis supports the monophyly of cryptophytes and haptophytes and the association of rhizaria with chromalveolates. Mol Biol Evol. 2007 Aug;24(8):1702–13.

163. Burki F, Shalchian-Tabrizi K, Pawlowski J. Phylogenomics reveals a new “megagroup” including most photosynthetic eukaryotes. Biol Lett. 2008 Aug 23;4(4):366–9.

164. Reeb VC, Peglar MT, Yoon HS, Bai JR, Wu M, Shiu P, et al. Interrelationships of chromalveolates within a broadly sampled tree of photosynthetic protists. Mol Phylogenet Evol. 2009 Oct;53(1):202–11.

165. Hampl V, Hug L, Leigh JW, Dacks JB, Lang BF, Simpson AGB, et al. Phylogenomic analyses support the monophyly of Excavata and resolve relationships among eukaryotic “supergroups.”Proc Natl Acad Sci U S A. 2009 Mar 10;106(10):3859–64.

166. Derelle R, Lang BF. Rooting the eukaryotic tree with mitochondrial and bacterial proteins. Mol. Biol. Evol. 2012 Apr;29(4):1277–89.

167. Keeling PJ, Burger G, Durnford DG, Lang BF, Lee RW, Pearlman RE, et al. The tree of eukaryotes. Trends Ecol. Evol. (Amst.). 2005 Dec;20(12):670–6.

168. Stechmann A, Cavalier-Smith T. Phylogenetic analysis of eukaryotes using heat-shock protein Hsp90. J. Mol. Evol. 2003 Oct;57(4):408–19.

169. Koonin EV. The origin and early evolution of eukaryotes in the light of phylogenomics. Genome Biology. 2010 May 5;11(5):209.

170. Rogozin IB, Basu MK, Csürös M, Koonin EV. Analysis of rare genomic changes does not support the unikont-bikont phylogeny and suggests cyanobacterial symbiosis as the point of primary radiation of eukaryotes. Genome Biol Evol. 2009;1:99–113.

171. Eme L, Trilles A, Moreira D, Brochier-Armanet C. The phylogenomic analysis of the anaphase promoting complex and its targets points to complex and modern-like control of the cell cycle in the last common ancestor of eukaryotes. BMC Evol. Biol. 2011;11:265.

172. Moparthi VK, Hägerhäll C. The evolution of respiratory chain complex I from a smaller last common ancestor consisting of 11 protein subunits. J. Mol. Evol. 2011 Jun;72(5-6):484–97.

173. Tokito MK, Howland DS, Lee VM, Holzbaur EL. Functionally distinct isoforms of dynactin are expressed in human neurons. Mol Biol Cell. 1996;7(8):1167–80.

174. Zhapparova ON, Bryantseva SA, Dergunova LV, Raevskaya NM, Burakov AV, Bantysh OB, et al. Dynactin subunit p150Glued isoforms notable for differential interaction with microtubules. Traffic. 2009;10(11):1635–46.

175. Vaughan PS, Miura P, Henderson M, Byrne B, Vaughan KT. A role for regulated binding of p150(Glued) to microtubule plus ends in organelle transport. J Cell Biol. 2002;158(2):305–19.

176. Parisi G, Fornasari MS, Echave J. Dynactins p25 and p27 are predicted to adopt the L[beta]H fold. FEBS Letters. 2004 Mar 26;562(1-3):1–4.

177. Moore JK, Li J, Cooper JA. Dynactin function in mitotic spindle positioning. Traffic. 2008;9(4):510–27.

178. Yeh E, Skibbens RV, Cheng JW, Salmon ED, Bloom K. Spindle dynamics and cell cycle regulation of dynein in the budding yeast, Saccharomyces cerevisiae. J Cell Biol. 1995 Aug;130(3):687–700.

179. Kurtzman CP. Phylogeny of the ascomycetous yeasts and the renaming of Pichia anomala to Wickerhamomyces anomalus. Antonie Van Leeuwenhoek. 2011 Jan;99(1):13–23.

180. Curtin CD, Borneman AR, Chambers PJ, Pretorius IS. De-Novo Assembly and Analysis of the Heterozygous Triploid Genome of the Wine Spoilage Yeast Dekkera bruxellensis AWRI1499. PLoS ONE. 2012;7(3):e33840.

181. Odronitz F, Kollmar M. Pfarao: a web application for protein family analysis customized for cytoskeletal and motor proteins (CyMoBase). BMC Genomics. 2006;7:300.

182. Hatje K, Keller O, Hammesfahr B, Pillmann H, Waack S, Kollmar M. Cross-species protein sequence and gene structure prediction with fine-tuned Webscipio 2.0 and Scipio. BMC Res. Notes. 2011;4:265.

183. Campanella JJ, Bitincka L, Smalley J. MatGAT: an application that generates similarity/identity matrices using protein or DNA sequences. BMC Bioinformatics. 2003 Jul 10;4:29.

184. Le SQ, Gascuel O. An improved general amino acid replacement matrix. Mol. Biol. Evol. 2008 Jul;25(7):1307–20.

185. Jones DT, Taylor WR, Thornton JM. The rapid generation of mutation data matrices from protein sequences. Comput. Appl. Biosci. 1992;8(3):275–82.

186. Altekar G, Dwarkadas S, Huelsenbeck JP, Ronquist F. Parallel Metropolis coupled Markov chain Monte Carlo for Bayesian phylogenetic inference. Bioinformatics. 2004 Feb 12;20(3):407–15.

187. Whelan S, Goldman N. A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. Mol. Biol. Evol. 2001 May;18(5):691–9.

188. Pillmann H, Hatje K, Odronitz F, Hammesfahr B, Kollmar M. Predicting mutually exclusive spliced exons based on exon length, splice site and reading frame conservation, and exon sequence homology. BMC Bioinformatics. 2011;12:270.

189. Schafer DA, Korshunova YO, Schroer TA, Cooper JA. Differential localization and sequence analysis of capping protein beta-subunit isoforms of vertebrates. J. Cell Biol. 1994 Oct;127(2):453–65.

190. de Hostos EL, Bradtke B, Lottspeich F, Guggenheim R, Gerisch G. Coronin, an actin binding protein of Dictyostelium discoideum localized to cell surface projections, has sequence similarities to G protein beta subunits. EMBO J. 1991 Dec;10(13):4097–104.

191. Tardieux I, Liu X, Poupel O, Parzy D, Dehoux P, Langsley G. A Plasmodium falciparum novel gene encoding a coronin-like protein which associates with actin filaments. FEBS Letters. 1998 Dec 18;441(2):251–6.

192. Figueroa JV, Precigout E, Carcy B, Gorenflot A. Identification of a Coronin-Like Protein in Babesia Species. Annals of the New York Academy of Sciences. 2004;1026(1):125–38.

193. Heil-Chapdelaine RA, Tran NK, Cooper JA. The role of Saccharomyces cerevisiae coronin in the actin and microtubule cytoskeletons. Current Biology. 1998 Nov 19;8(23):1281–S7.

194. Suzuki K, Nishihata J, Arai Y, Honma N, Yamamoto K, Irimura T, et al. Molecular cloning of a novel actin-binding protein, p57, with a WD repeat and a leucine zipper motif. FEBS Letters. 1995 May 15;364(3):283–8.

195. de Hostos EL. A brief history of the coronin family. Subcell Biochem. 2008;48:31 – 40.

196. Clemen CS, Rybakin V, Eichinger L. The coronin family of proteins. Subcell Biochem. 2008;48:1 – 5.

197. Uetrecht AC, Bear JE. Coronins: the return of the crown. Trends Cell Biol. 2006;16:421 – 426.

198. Smith TF. Diversity of WD-repeat proteins. Subcell Biochem. 2008;48:20 – 30.

199. Neer EJ, Schmidt CJ, Nambudripad R, Smith TF. The ancient regulatory-protein family of WD-repeat proteins. Nature. 1994;371:297 – 300.

200. Maniak M, Rauchenberger R, Albrecht R, Murphy J, Gerisch G. Coronin involved in phagocytosis: dynamics of particle-induced relocalization visualized by a green fluorescent protein Tag. Cell. 1995;83:915 – 924.

201. Ferrari G, Langen H, Naito M, Pieters J. A coat protein on phagosomes involved in the intracellular survival of mycobacteria. Cell. 1999;97:435 – 447.

202. Rybakin V, Stumpf M, Schulze A, Majoul IV, Noegel AA, Hasse A. Coronin 7, the mammalian POD-1 homologue, localizes to the Golgi apparatus. FEBS Lett. 2004;573:161 – 167.

203. de Hostos EL. The coronin family of actin-associated proteins. Trends Cell Biol. 1999;9:345 – 350.

204. Appleton BA, Wu P, Wiesmann C. The crystal structure of murine coronin-1: a regulator of actin cytoskeletal dynamics in lymphocytes. Structure. 2006;14:87 – 96.

205. Oku T, Itoh S, Ishii R, Suzuki K, Nauseef WM, Toyoshima S, et al. Homotypic dimerization of the actin-binding protein p57/coronin-1 mediated by a leucine zipper motif in the C-terminal region. Biochem J. 2005;387:325 – 331.

206. Spoerl Z, Stumpf M, Noegel AA, Hasse A. Oligomerization, F-actin interaction, and membrane association of the ubiquitous mammalian coronin 3 are mediated by its carboxyl terminus. J Biol Chem. 2002;277:48858 – 48867.

207. Kammerer RA, Kostrewa D, Progias P, Honnappa S, Avila D, Lustig A, et al. A conserved trimerization motif controls the topology of short coiled coils. Proceedings of the National Academy of Sciences of the United States of America. 2005;102(39):13891 – 13896.

208. Rybakin V, Clemen CS. Coronin proteins as multifunctional regulators of the cytoskeleton and membrane trafficking. Bioessays. 2005;27:625 – 632.

209. Morgan RO, Fernandez MP. Molecular phylogeny and evolution of the coronin gene family. Subcell Biochem. 2008;48:41 – 55.

210. Odronitz F, Hellkamp M, Kollmar M. diArk--a resource for eukaryotic genome research. BMC Genomics. 2007;8:103.

211. Breathnach R, Chambon P. Organization and expression of eucaryotic split genes coding for proteins. Annu Rev Biochem. 1981;50:349 – 383.

212. Keller O, Odronitz F, Stanke M, Kollmar M, Waack S. Scipio: using protein sequences to determine the precise exon/intron structures of genes and their orthologs in closely related species. BMC Bioinformatics. 2008;9:278.

213. Letunic I, Bork P. Interactive Tree Of Life (iTOL): an online tool for phylogenetic tree display and annotation. Bioinformatics. 2007;23:127 – 128.

214. Goode BL, Wong JJ, Butty AC, Peter M, McCormack AL, Yates JR, et al. Coronin promotes the rapid assembly and cross-linking of actin filaments and may link the actin and microtubule cytoskeletons in yeast. J Cell Biol. 1999;144:83 – 98.

215. Liu S-L, Needham KM, May JR, Nolen BJ. Mechanism of a Concentration-dependent Switch between Activation and Inhibition of Arp2/3 Complex by Coronin. Journal of Biological Chemistry. 2011 Mai;286(19):17039 –17046.

216. Veltman DM, Insall RH. WASP family proteins: their evolution and its physiological implications. Mol Biol Cell. 2010;21:2880 – 2893.

217. Vertessy BG, Toth J. Keeping uracil out of DNA: physiological role, structure and catalytic mechanism of dUTPases. Acc Chem Res. 2009;42:97 – 106.

218. Yonemura I, Mabuchi I. Heterogeneity of mRNA coding for Caenorhabditis elegans coronin-like protein. Gene. 2001;271:255 – 259.

219. Xavier CP, Rastetter RH, Stumpf M, Rosentreter A, Muller R, Reimann J, et al. Structural and functional diversity of novel coronin 1C (CRN2) isoforms in muscle. J Mol Biol. 2009;393:287 – 299.

220. Asano S, Mishima M, Nishida E. Coronin forms a stable dimer through its C-terminal coiled coil region: an implicated role in its localization to cell periphery. Genes Cells. 2001;6:225 – 235.

221. Beck K, Gambee JE, Kamawal A, Bachinger HP. A single amino acid can switch the oligomerization state of the alpha-helical coiled-coil domain of cartilage matrix protein. EMBO J. 1997;16:3767 – 3777.

222. Oku T, Itoh S, Okano M, Suzuki A, Suzuki K, Nakajin S, et al. Two regions responsible for the actin binding of p57, a mammalian coronin family actin-binding protein. Biol Pharm Bull. 2003;26:409 – 416.

223. Cai L, Makhov AM, Bear JE. F-actin binding is essential for coronin 1B function in vivo. J Cell Sci. 2007;120:1779 – 1790.

224. Gandhi M, Jangi M, Goode BL. Functional surfaces on the actin-binding protein coronin revealed by systematic mutagenesis. J Biol Chem. 2010;285:34899 – 34908.

225. Steinke D, Hoegg S, Brinkmann H, Meyer A. Three rounds (1R/2R/3R) of genome duplications and the evolution of the glycolytic pathway in vertebrates. BMC Biol. 2006;4:16.

226. Jaillon O, Aury JM, Brunet F, Petit JL, Stange-Thomann N, Mauceli E, et al. Genome duplication in the teleost fish Tetraodon nigroviridis reveals the early vertebrate proto-karyotype. Nature. 2004;431:946 – 957.

227. Chan KT, Creed SJ, Bear JE. Unraveling the enigma: progress towards understanding the coronin family of actin regulators. Trends Cell Biol. 2011;21:481 – 488.

228. Berg JS, Powell BC, Cheney RE. A millennial myosin census. Mol Biol Cell. 2001;12:780 – 794.

229. Archer SK, Claudianos C, Campbell HD. Evolution of the gelsolin family of actin-binding proteins as novel transcriptional coactivators. Bioessays. 2005;27:388 – 396.

230. Xavier CP, Eichinger L, Fernandez MP, Morgan RO, Clemen CS. Evolutionary and functional diversity of coronin proteins. Subcell Biochem. 2008;48:98 – 109.

231. Khurana S, George SP. Regulation of cell structure and function by actin-binding proteins: villin’s perspective. FEBS Lett. 2008;582:2128 – 2139.

232. Keeling PJ. The endosymbiotic origin, diversification and fate of plastids. Philos Trans R Soc Lond B Biol Sci. 2010;365:729 – 748.

233. diArk - a resource for eukaryotic genome research. Available from: http://www.diark.org

234. Letunic I, Doerks T, Bork P. SMART 6: recent updates and new developments. Nucleic Acids Res. 2009;37:D229–32.

235. Sigrist CJ, Cerutti L, de Castro E, Langendijk-Genevaux PS, Bulliard V, Bairoch A, et al. PROSITE, a protein domain database for functional characterization and annotation. Nucleic Acids Res. 2010;38:D161–6.

236. Metzker ML. Sequencing technologies - the next generation. Nat Rev Genet. 2010;11:31 – 46.

237. McPherson JD. Next-generation gap. Nat Methods. 2009;6:S2–5.

238. Petty NK. Genome annotation: man versus machine. Nat Rev Microbiol. 2010;8:762.

239. Human genome: Genomes by the thousand. Nature. 2010;467:1026 – 1027.

240. Durbin RM, Abecasis GR, Altshuler DL, Auton A, Brooks LD, Gibbs RA, et al. A map of human genome variation from population-scale sequencing. Nature. 2010;467:1061 – 1073.

241. Weigel D, Mott R. The 1001 genomes project for Arabidopsis thaliana. Genome Biol. 2009;10:107.

242. 1,000 Plant & Animal reference genomes project. Available from: http://www.ldl.genomics.cn/page/pa-research.jsp

243. Genome 10K Community of Scientists. Genome 10K: A Proposal to Obtain Whole-Genome Sequence for 10 000 Vertebrate Species. Journal of Heredity. 2009 Nov 1;100(6):659 –674.

244. Tangphatsornruang S, Somta P, Uthaipaisanwong P, Chanprasert J, Sangsrakru D, Seehalak W, et al. Characterization of microsatellites and gene contents from genome shotgun sequences of mungbean (Vigna radiata (L.) Wilczek). BMC Plant Biol. 2009;9:137.

245. Xu J, Saunders CW, Hu P, Grant RA, Boekhout T, Kuramae EE, et al. Dandruff-associated Malassezia genomes reveal convergent and divergent virulence traits shared with plant and human fungal pathogens. Proc Natl Acad Sci USA. 2007;104:18730 – 18735.

246. Xia Q, Zhou Z, Lu C, Cheng D, Dai F, Li B, et al. A draft sequence for the genome of the domesticated silkworm (Bombyx mori). Science. 2004;306:1937 – 1940.

247. Guerrero FD, Moolhuijzen P, Peterson DG, Bidwell S, Caler E, Bellgard M, et al. Reassociation kinetics-based approach for partial genome sequencing of the cattle tick, Rhipicephalus (Boophilus) microplus. BMC Genomics. 2010;11:374.

248. Liolios K, Chen IM, Mavromatis K, Tavernarakis N, Hugenholtz P, Markowitz VM, et al. The Genomes On Line Database (GOLD) in 2009: status of genomic and metagenomic projects and their associated metadata. Nucleic Acids Res. 2010;38:D346–54.

249. Sayers EW, Barrett T, Benson DA, Bolton E, Bryant SH, Canese K, et al. Database resources of the National Center for Biotechnology Information. Nucleic Acids Res. 2011;39:D38–51.

250. The Large-Scale Genome Sequencing Program. Available from: http://www.genome.gov/10001691

251. International Sequencing Consortium. Available from: http://www.intlgenome.org/

252. Pennisi E. Scientific publishing. Genomics researchers upset by rivals' publicity. Science. 2010;329:1585.

253. pgpool. Available from: http://pgpool.projects.postgresql.org/

254. Prototype JavaScript framework: Easy Ajax and DOM manipulation for dynamic web applications. Available from: http://www.prototypejs.org/

255. Lightwindow. Available from: http://www.p51labs.com/lightwindow/

256. Bostock M, Heer J. Protovis: a graphical toolkit for visualization. IEEE Trans Vis Comput Graph. 2009 Dec;15(6):1121–8.

257. Heer J, Bostock M. Declarative language design for interactive visualization. IEEE Trans Vis Comput Graph. 2010;16:1149 – 1156.

258. The R Project for Statistical Computing. Available from: http://www.r-project.org/

259. W3C SVG Working Group. Available from: http://www.w3.org/Graphics/SVG/

260. Goto N, Prins P, Nakao M, Bonnal R, Aerts J, Katayama T. BioRuby: bioinformatics software for the Ruby programming language. Bioinformatics. 2010;26:2617 – 2619.

261. Richards S, Liu Y, Bettencourt BR, Hradecky P, Letovsky S, Nielsen R, et al. Comparative genome sequencing of Drosophila pseudoobscura: chromosomal, gene, and cis-element evolution. Genome Res. 2005;15:1 – 18.

262. Elsik CG, Tellam RL, Worley KC, Gibbs RA, Muzny DM, Weinstock GM, et al. The genome sequence of taurine cattle: a window to ruminant biology and evolution. Science. 2009;324:522 – 528.

263. Zimin AV, Delcher AL, Florea L, Kelley DR, Schatz MC, Puiu D, et al. A whole-genome assembly of the domestic cow, Bos taurus. Genome Biol. 2009;10:R42.

264. Reaffirmation and Extension of NHGRI Rapid Data Release Policies: Large-scale Sequencing and Other Community Resource Projects. Available from: http://www.genome.gov/10506537

265. Liti G, Carter DM, Moses AM, Warringer J, Parts L, James SA, et al. Population genomics of domestic and wild yeasts. Nature. 2009;458:337 – 341.

266. Wheeler DA, Srinivasan M, Egholm M, Shen Y, Chen L, McGuire A, et al. The complete genome of an individual by massively parallel DNA sequencing. Nature. 2008;452:872 – 876.

267. Genome sequence of the pea aphid Acyrthosiphon pisum. PLoS Biol. 2010;8:e1000313.

268. Li R, Fan W, Tian G, Zhu H, He L, Cai J, et al. The sequence and de novo assembly of the giant panda genome. Nature. 2010;463:311 – 317.

269. Sharpton TJ, Stajich JE, Rounsley SD, Gardner MJ, Wortman JR, Jordar VS, et al. Comparative genomic analyses of the human fungal pathogens Coccidioides and their relatives. Genome Res. 2009;19:1722 – 1731.

270. Mendes ND, Freitas AT, Vasconcelos AT, Sagot MF. Combination of measures distinguishes pre-miRNAs from other stem-loops in the genome of the newly sequenced Anopheles darlingi. BMC Genomics. 2010;11:529.

271. Qin X, Evans JD, Aronstein KA, Murray KD, Weinstock GM. Genome sequences of the honey bee pathogens Paenibacillus larvae and Ascosphaera apis. Insect Mol Biol. 2006;15:715 – 718.

272. Diguistini S, Liao NY, Platt D, Robertson G, Seidel M, Chan SK, et al. De novo genome sequence assembly of a filamentous fungus using Sanger, 454 and Illumina sequence data. Genome Biol. 2009;10:R94.

273. Mardis ER. A decade’s perspective on DNA sequencing technology. Nature. 2011;470(7333):198 – 203.

274. Gardner MJ, Hall N, Fung E, White O, Berriman M, Hyman RW, et al. Genome sequence of the human malaria parasite Plasmodium falciparum. Nature. 2002;419(6906):498 – 511.

275. Tyler BM, Tripathy S, Zhang X, Dehal P, Jiang RH, Aerts A, et al. Phytophthora genome sequences uncover evolutionary origins and mechanisms of pathogenesis. Science. 2006;313(5791):1261 – 1266.

276. Clark AG, Eisen MB, Smith DR, Bergman CM, Oliver B, Markow TA, et al. Evolution of genes and genomes on the Drosophila phylogeny. Nature. 2007 Nov 8;450(7167):203–18.

277. Butler G, Rasmussen MD, Lin MF, Santos MAS, Sakthikumar S, Munro CA, et al. Evolution of pathogenicity and sexual reproduction in eight Candida genomes. Nature. 2009 Jun 4;459(7247):657–62.

278. Srivastava M, Begovic E, Chapman J, Putnam NH, Hellsten U, Kawashima T, et al. The Trichoplax genome and the nature of placozoans. Nature. 2008;454(7207):955 – 960.

279. Prochnik SE, Umen J, Nedelcu AM, Hallmann A, Miller SM, Nishii I, et al. Genomic analysis of organismal complexity in the multicellular green alga Volvox carteri. Science. 2010;329(5988):223 – 226.

280. Picardi E, Pesole G. Computational methods for ab initio and comparative gene finding. Methods Mol Biol. 2010;609:269 – 284.

281. Wei C, Brent MR. Using ESTs to improve the accuracy of de novo gene prediction. BMC Bioinformatics. 2006;7:327.

282. Stanke M, Tzvetkova A, Morgenstern B. AUGUSTUS at EGASP: using EST, protein and genomic alignments for improved gene prediction in the human genome. Genome Biol. 2006;7(Suppl 1):S11 11–8.

283. Babenko VN, Rogozin IB, Mekhedov SL, Koonin EV. Prevalence of intron gain over intron loss in the evolution of paralogous gene families. Nucleic Acids Res. 2004;32(12):3724 – 3733.

284. Roy SW, Gilbert W. Rates of intron loss and gain: implications for early eukaryotic evolution. Proc Natl Acad Sci USA. 2005;102(16):5773 – 5778.

285. Kent WJ. BLAT—The BLAST-Like Alignment Tool. Genome Res. 2002 Apr 1;12(4):656–64.

286. van Nimwegen E, Paul N, Sheridan R, Zavolan M. SPA: a probabilistic algorithm for spliced alignment. PLoS Genet. 2006;2(4):e24.

287. The Perl Programming Language. Available from: http://www.perl.org

288. script.aculo.us - web 2.0 javascript. Available from: http://script.aculo.us

289. Inkscape. Draw Freely. Available from: http://inkscape.org

290. The Official YAML Web Site. Available from: http://www.yaml.org/

291. purzelrakete’s workling at master - GitHub. Available from: http://github.com/purzelrakete/workling

292. tra’s spawn at master - GitHub. Available from: http://github.com/tra/spawn

293. Tokyo Cabinet: a modern implementation of DBM. Available from: http://fallabs.com/tokyocabinet/

294. Hoptoad: The app error app. Available from: http://hoptoadapp.com

295. RSpec.info: Home. Available from: http://rspec.info

296. Selenium web application testing system. Available from: http://seleniumhq.org

297. Yoon SJ, Seiler SH, Kucherlapati R, Leinwand L. Organization of the human skeletal myosin heavy chain gene cluster. Proc Natl Acad Sci USA. 1992;89(24):12078 – 12082.

298. Deutsch M, Long M. Intron-exon structures of eukaryotic model organisms. Nucleic Acids Res. 1999;27(15):3219 – 3228.

299. Benton MJ, Donoghue PC. Paleontological evidence to date the tree of life. Mol Biol Evol. 2007;24(1):26 – 53.

300. Slater GS, Birney E. Automated generation of heuristics for biological sequence comparison. BMC Bioinformatics. 2005;6:31.

301. Solovyev V, Kosarev P, Seledsov I, Vorobyev D. Automatic annotation of eukaryotic genes, pseudogenes and promoters. Genome Biol. 2006;7(Suppl 1):S10 11–2.

302. Salamov AA, Solovyev VV. Ab initio gene finding in Drosophila genomic DNA. Genome Res. 2000;10(4):516 – 522.

303. Birney E, Clamp M, Durbin R. GeneWise and Genomewise. Genome Res. 2004;14(5):988 – 995.

304. Yeh RF, Lim LP, Burge CB. Computational inference of homologous gene structures in the human genome. Genome Res. 2001;11(5):803 – 816.

305. Stanke M, Waack S. Gene prediction with a hidden Markov model and a new intron submodel. Bioinformatics. 2003;19(Suppl 2):ii215–25.

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

307. Nilsen TW, Graveley BR. Expansion of the eukaryotic proteome by alternative splicing. Nature. 2010;463:457 – 463.

308. Early P, Rogers J, Davis M, Calame K, Bond M, Wall R, et al. Two mRNAs can be produced from a single immunoglobulin mu gene by alternative RNA processing pathways. Cell. 1980;20:313 – 319.

309. Alt FW, Bothwell AL, Knapp M, Siden E, Mather E, Koshland M, et al. Synthesis of secreted and membrane-bound immunoglobulin mu heavy chains is directed by mRNAs that differ at their 3’ ends. Cell. 1980;20:293 – 301.

310. Mendes Soares LM, Valcarcel J. The expanding transcriptome: the genome as the “Book of Sand.”EMBO J. 2006;25:923 – 931.

311. Wang ET, Sandberg R, Luo S, Khrebtukova I, Zhang L, Mayr C, et al. Alternative isoform regulation in human tissue transcriptomes. Nature. 2008;456:470 – 476.

312. Zavolan M, van Nimwegen E. The types and prevalence of alternative splice forms. Curr Opin Struct Biol. 2006;16:362 – 367.

313. Blencowe BJ. Alternative splicing: new insights from global analyses. Cell. 2006;126:37 – 47.

314. Yang Y, Zhan L, Zhang W, Sun F, Wang W, Tian N, et al. RNA secondary structure in mutually exclusive splicing. Nat Struct Mol Biol. 2011;18:159 – 168.

315. Anastassiou D, Liu H, Varadan V. Variable window binding for mutually exclusive alternative splicing. Genome Biol. 2006 Jan 13;7(1):R2.

316. Graveley BR. Mutually exclusive splicing of the insect Dscam pre-mRNA directed by competing intronic RNA secondary structures. Cell. 2005;123:65 – 73.

317. Matlin AJ, Clark F, Smith CWJ. Understanding alternative splicing: towards a cellular code. Nat. Rev. Mol. Cell Biol. 2005 May 1;6(5):386–98.

318. Stephan M, Moller F, Wiehe T, Kleffe J. Self-alignments to detect mutually exclusive exon usage. Silico Biol. 2007;7:613 – 621.

319. Gotoh O. An improved algorithm for matching biological sequences. J Mol Biol. 1982;162:705 – 708.

320. Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. Journal of Molecular Biology. 1970 Mar 28;48(3):443–53.

321. Doring A, Weese D, Rausch T, Reinert K. SeqAn an efficient, generic C++ library for sequence analysis. BMC Bioinformatics. 2008;9:11.

322. Eddy SR. Where did the BLOSUM62 alignment score matrix come from? Nat Biotechnol. 2004;22:1035 – 1036.

323. Henikoff S, Henikoff JG. Amino acid substitution matrices from protein blocks. Proc Natl Acad Sci USA. 1992;89:10915 – 10919.

324. GFF (General Feature Format) Specifications Document - Wellcome Trust Sanger Institute. Available from: http://www.sanger.ac.uk/resources/software/gff/spec.html

325. George EL, Ober MB, Emerson CP. Functional domains of the Drosophila melanogaster muscle myosin heavy-chain gene are encoded by alternatively spliced exons. Mol Cell Biol. 1989;9:2957 – 2974.

326. Graveley BR, Kaur A, Gunning D, Zipursky SL, Rowen L, Clemens JC. The organization and evolution of the dipteran and hymenopteran Down syndrome cell adhesion molecule (Dscam) genes. RNA. 2004;10:1499 – 1506.

327. Schmucker D, Clemens JC, Shu H, Worby CA, Xiao J, Muda M, et al. Drosophila Dscam is an axon guidance receptor exhibiting extraordinary molecular diversity. Cell. 2000;101:671 – 684.

328. Tweedie S, Ashburner M, Falls K, Leyland P, McQuilton P, Marygold S, et al. FlyBase: enhancing Drosophila Gene Ontology annotations. Nucleic Acids Res. 2009;37:D555–9.

329. Adams MD, Celniker SE, Holt RA, Evans CA, Gocayne JD, Amanatides PG, et al. The genome sequence of Drosophila melanogaster. Science. 2000;287:2185 – 2195.

330. Zhan XL, Clemens JC, Neves G, Hattori D, Flanagan JJ, Hummel T, et al. Analysis of Dscam diversity in regulating axon guidance in Drosophila mushroom bodies. Neuron. 2004;43:673 – 686.

331. Neves G, Zucker J, Daly M, Chess A. Stochastic yet biased expression of multiple Dscam splice variants by individual cells. Nat Genet. 2004;36:240 – 246.

332. Hummel T, Vasconcelos ML, Clemens JC, Fishilevich Y, Vosshall LB, Zipursky SL. Axonal targeting of olfactory receptor neurons in Drosophila is controlled by Dscam. Neuron. 2003;37:221 – 231.

333. Lee C, Kim N, Roy M, Graveley BR. Massive expansions of Dscam splicing diversity via staggered homologous recombination during arthropod evolution. RNA. 2010;16:91 – 105.

334. McManus CJ, Duff MO, Eipper-Mains J, Graveley BR. Global analysis of trans-splicing in Drosophila. Proc Natl Acad Sci USA. 2010;107:12975 – 12979.

335. Labrador M, Mongelard F, Plata-Rengifo P, Baxter EM, Corces VG, Gerasimova TI. Protein encoding by both DNA strands. Nature. 2001;409:1000.

336. Dorn R, Reuter G, Loewendorf A. Transgene analysis proves mRNA trans-splicing at the complex mod(mdg4) locus in Drosophila. Proc Natl Acad Sci USA. 2001;98:9724 – 9729.

337. Horiuchi T, Giniger E, Aigaki T. Alternative trans-splicing of constant and variable exons of a Drosophila axon guidance gene, lola. Genes Dev. 2003;17:2496 – 2501.

338. McDermott A. Structure and dynamics of membrane proteins by magic angle spinning solid-state NMR. Annu. Rev. Biophys. 2009;38:385–403.

339. Renault M, Cukkemane A, Baldus M. Solid-State NMR Spectroscopy on Complex Biomolecules. Angew. Chem. Int. Ed. Engl. 2010 Nov 2;49(45):8346–57.

340. Judge PJ, Watts A. Recent contributions from solid-state NMR to the understanding of membrane protein structure and function. Curr. Opin. Chem. Biol. 2011 Oktober;15(5):690–5.

341. Tycko R. Solid-State NMR Studies of Amyloid Fibril Structure. Annu. Rev. Phys. Chem. 2011;62(1):279–99.

342. Manolikas T, Herrmann T, Meier BH. Protein structure determination from 13C spin-diffusion solid-state NMR spectroscopy. J. Am. Chem. Soc. 2008 Mar 26;130(12):3959–66.

343. Neal S, Nip AM, Zhang H, Wishart DS. Rapid and accurate calculation of protein ^{1}H, ^{13}C and ^{15}N chemical shifts. J. Biomol. NMR. 2003 Jul 1;26(3):215–40.

344. Wasmer C, Lange A, Van Melckebeke H, Siemer AB, Riek R, Meier BH. Amyloid fibrils of the HET-s(218-289) prion form a beta solenoid with a triangular hydrophobic core. Science. 2008 Mar 14;319(5869):1523–6.

345. Schneider R, Etzkorn M, Giller K, Daebel V, Eisfeld J, Zweckstetter M, et al. The native conformation of the human VDAC1 N terminus. Angew. Chem. Int. Ed. Engl. 2010 Mar 1;49(10):1882–5.

346. Matsuki Y, Akutsu H, Fujiwara T. Spectral fitting for signal assignment and structural analysis of uniformly ^{13}C-labeled solid proteins by simulated annealing based on chemical shifts and spin dynamics. J. Biomol. NMR. 2007;38(4):325–39.

347. Golotvin SS, Vodopianov E, Pol R, Lefebvre BA, Williams AJ, Rutkowske RD, et al. Automated structure verification based on a combination of 1D (1)H NMR and 2D (1)H - (13)C HSQC spectra. Magn. Reson. Chem. 2007 Oct;45(10):803–13.

348. Binev Y, Aires-de-Sousa J. Structure-based predictions of 1H NMR chemical shifts using feed-forward neural networks. J. Chem. Inf. Comput. Sci. 2004 Jun;44(3):940–5.

349. ACD/Labs NMR Predictors [Internet]. Toronto, ON, Canada: Advanced Chemistry Development; 2007. Available from: http://www.acdlabs.com/products/adh/nmr/nmr_pred/

350. Goddard TD, Kneller DG. SPARKY3. San Franscisco: University of California;

351. Delaglio F, Grzesiek S, Vuister GW, Zhu G, Pfeifer J, Bax A. NMRPipe: a multidimensional spectral processing system based on UNIX pipes. J. Biomol. NMR. 1995 Nov;6(3):277–93.

352. Vranken WF, Boucher W, Stevens TJ, Fogh RH, Pajon A, Llinas M, et al. The CCPN data model for NMR spectroscopy: development of a software pipeline. Proteins. 2005 Jun 1;59(4):687–96.

353. PyBioMaps is a framework to manage and visualize scientific data in a browser. Available from: http://pypi.python.org/pypi/PyBioMaps

354. Berman H, Henrick K, Nakamura H, Markley JL. The worldwide Protein Data Bank (wwPDB): ensuring a single, uniform archive of PDB data. Nucl. Acids Res. 2007 Jan 3;35(Database):D301–D303.

355. Xu XP, Case DA. Automated prediction of 15N, 13Calpha, 13Cbeta and 13C’ chemical shifts in proteins using a density functional database. J. Biomol. NMR. 2001 Dec;21(4):321–33.

356. Shen Y, Bax A. Protein backbone chemical shifts predicted from searching a database for torsion angle and sequence homology. J. Biomol. NMR. 2007 Aug;38(4):289–302.

357. Nielsen JT, Eghbalnia HR, Nielsen NC. Chemical shift prediction for protein structure calculation and quality assessment using an optimally parameterized force field. Prog Nucl Magn Reson Spectrosc. 2012 Jan;60:1–28.

358. Kohlhoff KJ, Robustelli P, Cavalli A, Salvatella X, Vendruscolo M. Fast and accurate predictions of protein NMR chemical shifts from interatomic distances. J. Am. Chem. Soc. 2009 Oct 7;131(39):13894–5.

359. Markley J, Ulrich E, Berman H, Henrick K, Nakamura H, Akutsu H. BioMagResBank (BMRB) as a partner in the Worldwide Protein Data Bank (wwPDB): new policies affecting biomolecular NMR depositions. J. Biomol. NMR. 2008 März;40(3):153–5.

360. Ulrich EL, Akutsu H, Doreleijers JF, Harano Y, Ioannidis YE, Lin J, et al. BioMagResBank. Nucl. Acids Res. 2007 Dec 23;36(Database):D402–D408.

361. Zhang H, Neal S, Wishart DS. RefDB: a database of uniformly referenced protein chemical shifts. J. Biomol. NMR. 2003 Mar;25(3):173–95.

362. Seidel K, Etzkorn M, Heise H, Becker S, Baldus M. High-Resolution Solid-State NMR Studies on Uniformly [13C,15N]-Labeled Ubiquitin. ChemBioChem. 2005 Sep 5;6(9):1638–47.

363. Vriend G. WHAT IF: a molecular modeling and drug design program. J. Mol. Graph. 1990 Mar;8(1):52–6, 29.

364. Lange A, Luca S, Baldus M. Structural constraints from proton-mediated rare-spin correlation spectroscopy in rotating solids. J. Am. Chem. Soc. 2002 Aug 21;124(33):9704–5.

365. Lange A, Seidel K, Verdier L, Luca S, Baldus M. Analysis of proton-proton transfer dynamics in rotating solids and their use for 3D structure determination. J. Am. Chem. Soc. 2003 Oct 15;125(41):12640–8.

366. Frishman D, Argos P. Knowledge-based protein secondary structure assignment. Proteins. 1995 Dec;23(4):566–79.

367. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. J. Mol. Biol. 1999 Sep 17;292(2):195–202.

368. Vuister GW, Kim S-J, Wu C, Bax A. 2D and 3D NMR Study of Phenylalanine Residues in Proteins by Reverse Isotopic Labeling. J. Am. Chem. Soc. 1994;116(20):9206–10.

369. Heise H, Hoyer W, Becker S, Andronesi OC, Riedel D, Baldus M. Molecular-level secondary structure, polymorphism, and dynamics of full-length alpha-synuclein fibrils studied by solid-state NMR. Proc. Natl. Acad. Sci. U.S.A. 2005 Nov 1;102(44):15871–6.

370. Castellani F, van Rossum B, Diehl A, Schubert M, Rehbein K, Oschkinat H. Structure of a protein determined by solid-state magic-angle-spinning NMR spectroscopy. Nature. 2002 Nov 7;420(6911):98–102.

371. Lundström P, Teilum K, Carstensen T, Bezsonova I, Wiesner S, Hansen DF, et al. Fractional 13C enrichment of isolated carbons using [1-13C]- or [2-13C]-glucose facilitates the accurate measurement of dynamics at backbone Calpha and side-chain methyl positions in proteins. J. Biomol. NMR. 2007 Jul;38(3):199–212.

372. Hong M. Determination of Multiple φ-Torsion Angles in Proteins by Selective and Extensive 13C Labeling and Two-Dimensional Solid-State NMR. Journal of Magnetic Resonance. 1999 Aug;139(2):389–401.

373. Takegoshi K, Nakamura S, Terao T. 13C–1H dipolar-assisted rotational resonance in magic-angle spinning NMR. Chem. Phys. Lett. 2001 Aug 31;344(5–6):631–7.

374. Igumenova TI, McDermott AE, Zilm KW, Martin RW, Paulson EK, Wand AJ. Assignments of carbon NMR resonances for microcrystalline ubiquitin. J. Am. Chem. Soc. 2004 Jun 2;126(21):6720–7.

375. Schneider R, Seidel K, Etzkorn M, Lange A, Becker S, Baldus M. Probing molecular motion by double-quantum (13C,13C) solid-state NMR spectroscopy: application to ubiquitin. J. Am. Chem. Soc. 2010 Jan 13;132(1):223–33.

376. Habenstein B, Wasmer C, Bousset L, Sourigues Y, Schütz A, Loquet A, et al. Extensive de novo solid-state NMR assignments of the 33 kDa C-terminal domain of the Ure2 prion. J. Biomol. NMR. 2011 Nov;51(3):235–43.

377. Linser R, Dasari M, Hiller M, Higman V, Fink U, Lopez del Amo J-M, et al. Proton-detected solid-state NMR spectroscopy of fibrillar and membrane proteins. Angew. Chem. Int. Ed. Engl. 2011 May 2;50(19):4508–12.

378. Wang C, Grey MJ, Palmer AG 3rd. CPMG sequences with enhanced sensitivity to chemical exchange. J. Biomol. NMR. 2001 Dec;21(4):361–6.

379. Mittermaier A, Kay LE. New tools provide new insights in NMR studies of protein dynamics. Science. 2006 Apr 14;312(5771):224–8.

380. Bieri M, Gooley PR. Automated NMR relaxation dispersion data analysis using NESSY. BMC Bioinformatics. 2011;12:421.

381. Kleckner IR, Foster MP. GUARDD: user-friendly MATLAB software for rigorous analysis of CPMG RD NMR data. J. Biomol. NMR. 2012 Jan;52(1):11–22.

382. Millet O, Loria JP, Kroenke CD, Pons M, Palmer AG. The static magnetic field dependence of chemical exchange linebroadening defines the NMR chemical shift time scale. J. Am. Chem. Soc. 2000;122(12):2867–77.

383. Davis DG, Perlman ME, London RE. Direct measurements of the dissociation-rate constant for inhibitor-enzyme complexes via the T1 rho and T2 (CPMG) methods. J Magn Reson B. 1994 Jul;104(3):266–75.

384. Luz Z, Meiboom S. Nuclear Magnetic Resonance study of the protolysis of trimethylammonium ion in aqueous solution—order of the reaction with respect to solvent. The Journal of Chemical Physics. 1963 Jul 15;39(2):366–70.

385. Cavanagh J. Protein NMR spectroscopy: principles and practice. Academic Press; 2007.

386. SciPy - Scientific Tools for Python. Available from: http://www.scipy.org/

387. Matplotlib - a python 2D plotting library. Available from: http://matplotlib.sourceforge.net/

388. jQuery. Available from: http://jquery.com/

389. FancyBox. Available from: http://fancybox.net/

390. Nixon JEJ, Wang A, Morrison HG, McArthur AG, Sogin ML, Loftus BJ, et al. A spliceosomal intron in Giardia lamblia. Proceedings of the National Academy of Sciences. 2002 Mar 19;99(6):3701–5.

391. Vaňáčová Š, Yan W, Carlton JM, Johnson PJ. Spliceosomal introns in the deep-branching eukaryote Trichomonas vaginalis. Proceedings of the National Academy of Sciences of the United States of America. 2005 Mar 22;102(12):4430–5.

392. Koonin EV. The origin of introns and their role in eukaryogenesis: a compromise solution to the introns-early versus introns-late debate? Biology Direct. 2006 Aug 14;1(1):22.

393. Carmel L, Wolf YI, Rogozin IB, Koonin EV. Three distinct modes of intron dynamics in the evolution of eukaryotes. Genome Research. 2007 May 1;17(7):000–000.

394. de Roos AD. Conserved intron positions in ancient protein modules. Biology Direct. 2007 Feb 8;2(1):7.

395. Lynch M. Intron evolution as a population-genetic process. Proceedings of the National Academy of Sciences. 2002 Apr 30;99(9):6118–23.

396. Hsieh SJ, Lin CY, Liu NH, Chow WY, Tang CY. GeneAlign: a coding exon prediction tool based on phylogenetical comparisons. Nucleic Acids Research. 2006 Jul 1;34(Web Server):W280–W284.

397. Csűrös M, Holey JA, Rogozin IB. In search of lost introns. Bioinformatics. 2007 Jul 1;23(13):i87–i96.

398. Pavesi G, Zambelli F, Caggese C, Pesole G. Exalign: a new method for comparative analysis of exon–intron gene structures. Nucleic Acids Research. 2008 May 1;36(8):e47–e47.

399. Wilkerson MD, Ru Y, Brendel VP. Common introns within orthologous genes: software and application to plants. Briefings in Bioinformatics. 2009 Nov 1;10(6):631–44.

400. Csűrös M. Malin: maximum likelihood analysis of intron evolution in eukaryotes. Bioinformatics. 2008 Jul 1;24(13):1538–9.

401. Fedorov A, Merican AF, Gilbert W. Large-scale comparison of intron positions among animal, plant, and fungal genes. PNAS. 2002 Dec 10;99(25):16128–33.

402. Rogozin IB, Wolf YI, Sorokin AV, Mirkin BG, Koonin EV. Remarkable Interkingdom Conservation of Intron Positions and Massive, Lineage-Specific Intron Loss and Gain in Eukaryotic Evolution. Current Biology. 2003 Sep 2;13(17):1512–7.

403. Vivek G, Tan TW, Ranganathan S. XdomView: protein domain and exon position visualization. Bioinformatics. 2003 Jan;19(1):159–60.

404. The PyMOL Molecular Graphics System [Internet]. Schrödinger, LLC; Available from: www.pymol.org

405. GFF - GMOD [Internet]. [cited 2012 Jul 30]. Available from: http://gmod.org/wiki/GFF

A.2 Curriculum vitae

Name: Björn Hammesfahr

Date of Birth: 13 June 1982

Place of Birth: Ludwigsburg

Education: 1989-1993
Josef-Helmer-Schule Waldenburg

1993-1994
Hohenlohe Gymnasium Öhringen

1994-1997
Deutschorden-Gymnasium Bad Mergentheim

1997-2003
Schloßgymnasium Künzelsau

Study: 2003-2009
Biology at the Julius-Maximilians-University Würzburg

PhD Study: 2009-2012
Kollmar Group at the Max Planck Institute for Biophysical Chemistry, Göttingen