INFERENCE OF DEGREE OF SIGNIFICANCE OF SINGLE AMINO ACIDS FROM THE LITERATURE

BY

ANTHONY E BECKER

DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Molecular and Integrative Physiology in the Graduate College of the University of Illinois at Urbana-Champaign, 2011

Urbana, Illinois

Doctoral Committee:
Professor Eric Jakobsson, Chair
Professor Mark E Nelson
Assistant Professor Hee Jung Chung
Associate Professor Thomas J Anastasio
Associate Professor Claudio F Grosman
ABSTRACT

Several subfamilies of potassium channels are highly conserved along the vast majority of the protein sequence among a wide array of very distantly related animals. We call this characteristic “hyperconservation”. In this work we create a quantitative definition and explore the degree of hyperconservation and characterization in each of the well-known potassium channel subfamilies. In general the potassium channels seem to exhibit a large degree of hyperconservation within the subfamilies but a wide diversity (to the point of confounding alignment) between subfamilies. Here we examine the literature of one potassium channel subfamily (KCNA2) to determine whether or not all of the completely conserved residues have been noted and considered for functional inference. Out of several thousand papers, we find four residues that are completely conserved but unmentioned in any article; F85, E112, P156, S159. F85 and E112 are in fact completely conserved within and across different K channel subfamilies. The challenges encountered during this search, plus the fact that some completely conserved residues have been overlooked, make it clear that there needs to be a more automated method for extracting sequence-related information from literature articles. The work in this thesis emerged from considering the problem of how to intensively study a protein family based on the sequences for the family. In the first part of the thesis, we consider the issue of studying a family of potassium channels, residue-by-residue. This involves accounting for a history in which residue numbering systems and protein nomenclature are variable throughout the literature on this family. Discovering information in literature about single residues in any protein family can be daunting considering that the residues have a different number placement in each sequence. Then one must consider the change in numbers for each isoform or if an author renumbers them from a sequence section. This problem is greatly compounded when one wishes to consider orthologs and paralogs to these orthologs (homologs) in all species. This involves accounting for a history in which residue numbering systems and protein nomenclature are variable throughout the literature on any family. This has resulted in the creation of a program called FiSHAAL-Finding Single Homologous Amino Acids. It is offered as a prototype literature amino acid location determination program for partial automation of identifying homologous residues and linking any corresponding residues in an alignment column to their PubMed IDs. Ultimate Hypothesis: Can accurate homologous amino acid residue mention information be linked effectively to all PubMed articles in a semi-automated fashion?
ACKNOWLEDGEMENTS

I would like to thank all in the Jakobsson lab that aided me with understanding biological ion channels. I would especially like to thank Ashok Palaniappan, Shreedhar Natarajan and Hui Liu for their algorithms and valuable insights into some of the reasons for potassium channel characteristics. In addition, I would like to thank Bill Baker of NCSA on helping me set up the Python Django web server and Daniel Winski who sat with me to create, rapidly type the code modules and quickly find Google problem solving solutions for FiSHAAL. I am also indebted to Illinois Department of Human Services/Rehabilitation Services for providing support during the thesis and for supporting the assistance of Daniel to work with me on the portions of the work that required rapid code typing. I would also like to thank the University of Illinois, the division of Disability Resources and Educational Services and the Beckman Institute for providing location and accessibility support for this work.
Table of Contents

CHAPTER 1: POTASSIUM CHANNELS BACKGROUND AND SIGNIFICANCE OF INITIAL STUDIES ............................................................ 1
CHAPTER 2: SELECTIVITY FILTER POSITION ................................................ 3
CHAPTER 3: HYPERCONSERVATION ........................................................... 5
CHAPTER 4: RELATIVE CONSERVATION OF DIFFERENT HUMAN POTASSIUM CHANNEL SUBFAMILIES ..............................12
CHAPTER 5: AUTOMATION UTILIZING FiSHAAL .........................................16
CHAPTER 6: RESULTS; WORKFLOW THROUGH WEB SITE PROGRAM INTERFACE ..............................................................................24
CHAPTER 7: CONCLUSION ........................................................................25
CHAPTER 8: POSSIBLE FUTURE FUNCTIONALITY AND WORK ....................28
CHAPTER 9: REFERENCES ........................................................................32
CHAPTER 10: FIGURES AND TABLES ........................................................ 39
APPENDIX A: SFFINDERV2.0 ..................................................................... 83
APPENDIX B: KCNA2VERTICLEALIGNMENT.XLS .........................................84
APPENDIX C: SUPPLEMENT REFERENCES ............................................... 85
APPENDIX D: PDBDISTBF85.XLS ...............................................................86
APPENDIX E: PDBDISTBE112.XLS .............................................................87
APPENDIX F: PERLPROGRAM .................................................................. 88
APPENDIX G: PYTHON X, Y; BARPLOT.PY ................................................. 89
APPENDIX H: EXTRA FIGURES ..................................................................90
APPENDIX I: EXAMPLE SNAPSHOTS FROM WEB SITE ..............................128
APPENDIX J: FISHAAL CODE ....................................................................130
APPENDIX K: THE COMPLETE QUERY LIST .............................................131
CHAPTER 1: POTASSIUM CHANNELS BACKGROUND AND SIGNIFICANCE OF INITIAL STUDIES

Ever since the landmark studies of Hodgkin and Huxley in 1952 and the collective framework in Bertil Hille’s master reference (2001), membrane excitable ion channels have been extensively characterized. Potassium channels (Table 1.1) are vital components in the function for the propagation of neurophysiological communication processes, acting as highly specific potassium ion flux regulators. They contain two, four, six, or eight transmembrane helices. The pore helix and selectivity filter that is highly specific for potassium ions (Jiang et al. 02) is before the last transmembrane helix in two or six transmembrane channels that usually exist as homo-tetramers (Nelson et al. 99)(Choe, 02). They are regulated by a multitude of factors such as: voltage via S4 domain as the sensor, pH, or ligands like calcium ions, cyclic nucleotides and ATP.

The use of bioinformatic tools to analyze different segments of sequence data may draw insight into the function of particular sections of the potassium channel. This may lead to pharmacological or other advances in how to manipulate the function of all or specific types of potassium channels. It may also lead to remedies that block effects from various poisons produced by organisms that are designed to shut down or incapacitate the potassium channels located in neurons. Creating algorithms using these collective tools may be useful in quickly analyzing data from other ion channels or other proteins.

Creating blocks for potassium channel sequences

Potassium channel sequences for main Sequence Logos (Schneider and Stephens, 90) were acquired by BLASTing (Altschul, et al. 97) under the default conditions of the Biology Workbench with the Salmonella potassium channel fragment. Redundancies were removed at the level of five amino acid difference in the entire sequence, or approximately 99%. To achieve a rough initial alignment, CLUSTALW (Higgins, D. et al. 92) was used for alignments. Alignments were done separately for different taxa; i.e., Genbank groups. For these short Logos the area surrounding the selectivity filter (SF) was cut out (see below). The criteria for a correct alignment were: 1) the TM domains should align, and 2) the SF should align. Alignments were visualized using Sequence Logo in figure 1.1 for a portion of the permeation pathway including the selectivity filter. Alignments of the SFs must reflect the G[X]G mandatory motif for potassium selectivity (Heginbotham et al. 94). TMHMM www.cbs.dtu.dk/services/TMHMM/
was used to reference transmembrane area motifs in order to achieve an accurate topology of the protein. Channels with multiple selectivity filters were split into segments in which single selectivity filters were in their own sequence. The initial use of Sequence Logo on potassium channels was to observe the region of the selectivity filter (Figure 1.1a-g) in an alignment of +15 to -15 relative to the initial G of the GxG of the selectivity filter (except invertebrates were -17 to +6). Figures a-f reveal that, aside from the two G’s of the selectivity filter which are absolutely conserved, the other most widely conserved residues are two T’s shortly preceding the selectivity filter. Other residues are strongly conserved in some taxa but not others, such as the D immediately following the GxG which is very strongly conserved in vertebrates and plants, but not in invertebrates. Figure 1.1g shows the overall conservation pattern as represented by the sequences in the Transport Classification Database (TCDB) (Saier et al). A block represented by Figure 1.1g was used to identify the position of K channel selectivity filters, as described below and shown in Figure 1.2.
CHAPTER 2: SELECTIVITY FILTER POSITION

The potassium channel's defining feature is the selectivity filter, which for most members of this family imparts high specificity for potassium ions over all other types of physiologically significant ions (Doyle et al. 98) (Jiang et al. 02). The iconic sequence for the selectivity filter is TTVGYGD. The only two residues that are conserved for all potassium channels are the two glycines, since they uniquely provide the ability to form two carbonyl cages selective for potassium. If one of the G’s is missing, the strong potassium selectivity is gone (Heginbotham et al. 94, Zagotta, 06, Shi, et al. 06). Therefore, focusing on common variations of the location of the selectivity filter in reference to the entire sequence of the gene may show significant evolutionary, structural or functional reasons of how this single motif has persisted over the eons (Jiang et al. 02). A program for finding the position of the selectivity filter is shown in supplement SFFinderv2.0 (Appendix A). The program (written first by lab mate Ashok Palaniappan and modified by me) searches for a sequence block characteristic of the selectivity filter. The block was created from the potassium channel sequences in the Transport Classification Data Base (TCDB http://www.tcdb.org/) The global potassium channel block derived from the TCDB sequences is

[YWFLIS][FWLYC][SACTGVIL][IVFLTAMSCP][VITESAML][TSVLCl][LMIFQVAHEGSY][TSALVC][TSC][VITL]G[YFL]G[DNFHGVLYT] and is represented by the Sequence Logo in Figure 1.1g. The template above was used to find the position of the selectivity filter in all potassium channels returned from the UniProt database by InterPro (Hunter et al. 2009) domain IPR003091 which is the archetypal voltage-gated potassium channel domain. Figure 1.2 shows the positions of the selectivity filters within the complete protein sequences for the taxa Invertebrates, Chordates, Plastids, Cyanobacteria, Eubacteria, and Archaea. The taxa show a wide range of differences with the Cyanobacteria sequences varying very little and the Invertebrates sequences having much diversity. The TCDB-derived block retrieves the filters from 86 % of the hits from the canonical K channel domain IPR03091. Use of the same template against the refseq nonmammalian vertebrate database and checking all the returns revealed 1.4% of them were false positives. Increasing the number of alternative residues in possible positions, by building the block from a broader set of training sequences would reduce the number of false negatives, but at the probable expense of increasing false positives. This might be balanced by increasing the length of the block, possibly to include the inner helix as
well as the P region. The complete balance between selectivity (minimize false positives) and sensitivity (minimize false negatives) is yet to be completely worked out. It may be that some of the false positives are in fact significant; i.e., they may be remnants of a selectivity motif that is no longer functional. The SF is a highly conserved but a small part of the potassium channel sequence. There are subfamilies that are highly conserved along almost the entire length of their sequence; we call this hyperconservation.
CHAPTER 3: HYPERCONSERVATION

The terms “hyperconservation” and “ultraconservation” have been used in the context of referring to relatively short motifs of protein (Doyle et al. 03) or coding and non-coding (Richler et al. 06) DNA sequences. In the context of this paper we define “hyperconservation” as a consensus conserved sequence along the entire length of the alignment, across a wide reach of evolutionary distance (at least several hundred million years, or near the last common ancestor of the metazoa). We will show below that some subfamilies of potassium channels exhibit this phenomenon. Examples of this in potassium channels are seen in (Figure 3.1). Critical functional areas such as the permeation pathway are obvious in their strong conservation but there are several other areas that show hyperconservation, for no obvious functional reason.

Hyperconservation in KCNA and Acquiring Orthologs

In order to define a representative Shaker-like (KCNA) sequence to explore hyperconservation, all eight human Shaker-like potassium channels as per HUGO (HUman Genome Organization) (http://www.genenames.org/genefamily/vgic.php#KV) (PMID: 16382104) KCNA1-7, KCNA10 were collected and aligned with each other. The alignment was submitted to WebLogo in block form to create a Sequence Logo. The top line in the Logo of residues reveals the consensus sequence of KCNA (Figure 3.2). BLASTing this sequence against the human database reveals KCNA2 as the one with the closest correspondence to the KCNA consensus sequence. Using human KCNA2 as the authoritative sequence, orthologs were obtained manually by BLASTing against the nonredundant database. The top hit of each new species was then extracted from the results. This was performed until sequence positives dropped to less than 50%. True orthologs were observed if they reverse BLASTed to human KCNA2 in the human genome (Reciprocal Best Hit (RBH)). In some species there were no KCNA2 RBH ortholog equivalents but they reverse BLASTed to the KCNA family. Sequences were renamed with the English name if a species acquired one, then the e-value, a "y" or "n" on whether or not they reverse BLASTed, and if they did not then the number of sequences before KCNA2 was written (Table 3.1). Nineteen species reverse BLASTed human KCNA2 as the top hit (Table 3.2), (Figure 3.1) and went as deep into the phylogenetic tree (Figure 3.3) as roundworms and jellyfish. Three species (Apis mellifera, Loligo opalescens, and Schistosoma mansoni) found human KCNA1 as the top hit, but the differential score between KCNA1 and KCNA2 was so slight that these sequences were included.
as well (Table 3.3). These sequences were also used in the hyperconservation study in order to include distantly related organisms. Two species that have a significant number of top hit sequences prior to KCNA2 that were not used in the alignment were *Notoplana atomata* and *Ciona intestinalis*. Figure 3.3 shows two major groups of the vertebrates and invertebrates that are close to 50% or more identical in sequence. A more detailed description of which proteins were returned from the BLAST prior to KCNA2 is shown in Table 3.2. Sequences were aligned using the defaults in CLUSTALW from the original Biology Workbench. A shortened visual diagram of the alignment (Figure 3.1) was obtained using the defaults of BOXSHADE in the Biology Workbench (Hofmann and Baron [http://www.isrec.isb-sib.ch/software/BOX_faq.html](http://www.isrec.isb-sib.ch/software/BOX_faq.html)).

The standard horizontal alignment was converted into a vertical alignment and placed into an Excel worksheet for a more efficient system of tracking corresponding residues to specific journals. (Excel raw KCNA2 vertical alignment work sheets. See Supplement kcna2vertalignment.xls (Appendix B).)

Hyperconservation was observed deep inside the phylogenetic tree (Figure 3.3). In order to analyze why KCNA2 has persisted so long with so little evolutionary change, the entire PubMed literature database was searched for all residues in the human KCNA family, all aliases of KCNA, orthologs and ortholog aliases to learn what is already known about this protein in order to analyze what is unknown about hyperconserved residues. We present the manually acquired information extraction technique here and later present a program that can semi-automatically perform this for any protein.

**Journal Database Acquisition, Disparities and Nuances**

In the early stage of the project, we obtained all the different names or aliases of the KCNA2 protein through different resources, including the different names and aliases of the orthologs to the KCNA protein (Table 3.4). An automated method has been developed to retrieve all articles pertaining to a protein or group of proteins. Individual residue information was extracted from over 900 articles (see supplementreferences.doc (Appendix C)) out of approximately 4000 Shaker K channel articles whose abstracts indicated that they might contain individual residue information. Information was recorded if the article gave a residue number. At this point only the amino acid residue and number were recorded in the database in 2006 from the article in the orthologous group of proteins. Amino acid information was not taken if the sequence in the article study was a chimeric protein and did not talk of individual residues. In order to record information efficiently into an Excel document, the alignment was taken from the standard
horizontal plane and converted to a vertical plane. Numbers taken from articles that referred to the authoritative sequence that is human Kv1.2 were taken as the actual number of reference whereas; all others were recorded relative to that sequence number in the alignment. For most articles it was easy to determine which residue number was the point of reference. Articles that contained the updated authoritative sequence were recorded in the vertical alignment at the appropriate residue number as a notation of the residue in the article for example Baker et al. 98 (Table 3.5) and Visan et al. 04 (Table 3.6). On the other hand, more difficult articles were saved for further analysis to normalize the numbering to the authoritative sequence, KCNA2. These are articles that renumbered their own sequence, were describing sequences in which the name has been changed, or otherwise used nonstandard, outdated, or unspecified numbering and/or naming conventions. Visan et al. 04 also shows a variant amino acid name and a different protein name. Another example can be seen in an article section taken directly from Wrisch and Grissmer, 2000 in Figure 3.4. Other articles were referring to proteins that have been given a different name in the database. These sequences were identified in several different ways. In some cases, all of the amino acids in the paper were collected and put in order according to the residue number with X's placed in positions not mentioned (See Data Mining). A CLUSTALW sequence alignment between human Kv1.2, the aforementioned protein, and the newly formed sequence was performed. Then the correctly extrapolated sequence numbers were recorded. Bowlbry et al. 97 shows a -2 shift in residue numbers relative to the database sequence (Table 3.7). Some cases involved papers in which a certain motif sequence section was cut out of a potassium channel and renumbered starting with one (see below). This method is now performed differently in order to make it more easily implemented by computational scripts (see below).

**Un-mentioned completely conserved residues**

Upon further research the Table was scanned through to discover that four hyperconserved residues, F85, E112, P156, S159, have never been mentioned individually in any scientific article searched through. Only two, F85 and E112, were contained within the x-ray structure of the rat shaker Kv1.2; 2A79 (Jiang et al. 02), making them more suitable for further analysis. F85 and E112 are not only completely conserved among species going back to a common ancestor with the roundworm but are also conserved in all human KCNA family channels.
A nice visual system of the x-ray structure 2A79 or 3LUT (Chen et al. 10) and the KCNA alignment can be seen with the ConSurf server at http://consurf.tau.ac.il/ developed by (Landau et al. 05). A sequence alignment including a PDB structure can be entered to produce a color-coded visual system of conserved residues. Visual inspection of conservation patterns (Figure 3.5) can help to design mutagenesis experiments. Shown are F85 and E112 as well as other characteristics of KCNA2 being pointed out at various angles and zoom levels using the ConSurf program. In (Figures 3.6-3.7) a stick model X-ray structure of the tetramer rat KCNA2 shows only F85 and E112 as the space-filled amino acids, with nearby residues in stick representation at various angles. VMD software (http://www.ks.uiuc.edu/Research/vmd/) and PyMol (http://www.pymol.org/) were used for representations. These figures show the conservation in the KCNA2 subfamily so we performed further analysis in other human potassium channel families to observe how far the conservation continues.

Alignments with other potassium channel subfamilies were performed and it has been further discovered that these two residues have been conserved in several other potassium channel subfamilies in humans; KCNB, KCNC, KCND, and KCNG (Figure 3.8). There is 100% conservation for these two residues in KCNA orthologs and human KCNB, KCNC, KCND, and KCNG families. The same research techniques as described above using the Excel worksheet were recorded for those subfamilies, and their orthologs. We then went on to consider the possible function of these two amino acids by considering what other residues are in close proximity to them. In a protein, other amino acid residues within approximately 8 Å of each other affect amino acid behavior (Fodor et al. 04). A program devised by Shreedhar Natarajan that calculates distances from amino acids was performed on the tetramer PDB x-ray structure to determine which amino acids could affect the behavior of F85 or E112, which would direct the analysis of why these two proteins are so conserved. The journal descriptions of surrounding residues to these two amino acids were reviewed for analysis into their possible function. A Table of proximate residues was produced to research the function of these nearby residues and only residues found within an acute triangle F85 and E112 or within 8 Å of both to limit the number of journal results to residues that possibly affect both residues.

Proximity Summary
We focus on residues that are near enough and appropriately situated to have significant interactions with both F85 and E112. For this study we looked at subject residues that are in one of two classes:

1) within 8 Å of both F85 and E112 (S84, I88, N81, and R80) for ease of following the discussion, in the rest of the text references to these four residues will be highlighted with yellow.

2) are within 8 Å of either F85 or E112 and form with F85 and E112 an acute triangle (Table 3.8). (Y116, A87, P83, and F78) For ease of following the discussion, in the rest of the text references to these four residues will be highlighted with aqua.

All of the residues listed above are in the tetramerization (T1) domain of Kv1.2, as shown in the x-ray structure of the Shaker ortholog (Kreusch et al. 98), and subsequently in the x-ray structure of the rat Kv1.2 (Jiang et al. 02). Their positions relative to F85 and E112 are shown in Figure 3.9 based on the tetramer Kv1.2 x-ray structure PDB file 2A79 at various angles using PyMol software. The red are E112, the blue are F85 and then the yellow are the subject residues. The order of the figures and summary start with the smallest value to the sum of the distances from the two query residues F85 and E112 and is still in the cutoff value of 8 Å. F85 is 7.41 Å from E112, so they may influence each other.

In the following descriptions below, if a residue position has a lower case letter other than an “s” directly in front of it, is the first initial of a species name. If the lower case letter is an “s” then that residue is referring to a Shaker channel residue from Drosophila. No lower case letters before the amino acid indicates a human protein.

S84 is the most proximal residue to E112 at a little under 4.24 Å (Figure 3.9a). A couple of studies have shown that it may be a phosphorylation site by kinases (Tao et al. 05). If it is changed to an alanine in its counterpart of S229 in Kv1.4 it has no significant effects on activation, inactivation, conductance but only some difference in current density (Tao et al. 05). If two other phosphorylation sites in addition to S84 were modified in Kv1.3-S105=S84, C-type inactivation was reduced (Kupper et al. 95). This suggests that E112 or F85 at 5.38 Å may play a role in phosphorylation sensitivity of this proximal residue. This is one characteristic to investigate while performing experiments on E112 or F85 even though it is 70% conserved in our studied species.

Located at 4.75 Å from F85 and 6.34 Å from E112 seen in Figure 3.9b, I88 is not conserved but does have experimental data (Strang et al. 01). If the Aplysia channel is perturbed
there is a loss of polar tetramerization formation as aI121=I88 interacts with aR115=R82, aQ126=Q93, aD119=D86 and aS73=S40. Mutations likely produce either steric blocks to subunit tetramerization or alter the packing between layers of secondary structure within the monomer.

N81 interacts with Y90 but has no significant effect on channel gating if mutated to an alanine (Minor et al. 00). The residue in shawH58=N81 only forms Shaw specific interactions in this paper, as it is not conserved (Bixby et al. 99). It is 7.25 Å from F85 and 5.82 angstroms from E112 seen in Figure 3.9c. The x-ray structure from rat PDB 1NN7 rKv4.2 T1 region shows 4.2-rD88=N81 along with 4.2-rE110=V104 create an acidic pocket with 4.2-rR93=D86 (Nanao et al. 03).

A 90% conserved residue R80, is in Figure 3.9d at 7.32 Å from F85 and 6.77 Å from E112. It interacts with Q93 but has no significant affects if changed (Minor et al. 00). Seen in Shaker, sQ126(sQ157)=Q93 has interactions with sR113(sR144)=R80 (Bixby et al. 99). It is a possible candidate in 1.10-R133R=R80 for long QT syndrome mutation as a nucleotide change (Iwasa et al. 01). (Jerng et al. 02) refers to (Liu et al. 01) as a protein residue used in experiments in delayed rectifier K+ conductance. That article shows that SqKv1A-G87=SqKv1B-R76=SqKv1D-R53=R80 is part of the highly conserved FFDR motif, in which SqKv1A-G87 is the only one that is a glycine in potassium channels Kv1-6, Shab, Shaw and Shal. Therefore if SqKv1A-G87is changed to an Arginine, functional expression amounts of SqKv1A dramatically goes up but does not affect channel conductance. If the SqKv1D-R53 is changed to a Glycine functional expression amounts are almost the same as SqKv1A levels in which it is a Glycine. This is located in the T1 region showing if these residues are perturbed like this there is a suggestion of a loss of tetramerization thus reducing functional expression (Liu et al. 01). A change in SqKv1.1A-R87G=sR144=R80 causes an almost 50-fold reduction in tetramer formation and reduces functional expression levels in SqKv1.1A channels (Rosenthal and Bezanilla 02).

Figure 3.9e show Y116 which is not close to the cutoff from F85 but is 6.10 Å from E112. It is 70% conserved among species. The mutation 1.3-Y137F(1.3-Y135)=Y116 causes loss of its phosphorylation site and causes a decrease in peak current with a quicker inactivation state (Bowlbry et al. 97; Colley et al. 04; Fadool et al. 97, 98, 00). When mutated Kv1.1-Y149=Y116 there was inter-domain destabilization of layers one and three (Strang et al. 01).
Like F115, 1.5-Y194A(1.5-Y203)=Y116 reduces Kvβ1 hetero-oligo binding (Sewing et al. 96).

A87 is not conserved and is in Figure 3.9f. (Bixby et al. 99) shows that sA151=shawQ64=A87 has Shaw specific channel polar interactions. It is 7.31 Å from F85 and is not within the cutoff of E112.

The farthest residue within the 8 Å cutoff of only F85 at 7.70 Å is P83 and is conserved among 70% of the species. It is in Figure 3.9g and has had no research done on that particular residue.

F78 is also only within the cutoff of F85 at 7.65 Å. In Figure 3.9h and is a conserved residue in all species and other subfamilies. In the case of (Liu et al. 01) F78 is mentioned because it is part of the FFDR sequence, a highly conserved motif which corresponds to SqKv1A-F85=SqKv1B-F74=SqKv1D-F51=F78. When seen as aF111=sF142=F78 conserved end of layer 1 (Liu et al. 05). Additionally aF111=sF142=F78 is a conserved residue at the end of layer one that is part of the hydrophobic core of T1 (Kreusch et al. 98).

With the proximity of various amino acids calculated in relation to our two residues we can visit the journals that mention these nearby residues to discover what function(s) are located in this particular motif (Excel Distances from F85 & E112 see Supplements (pdbdistBF85.xls(Appendix D)) and (pdbdistBE112.xls (Appendix E)(Figure 3.9). Then we can draw an inference into the possible experimental designs that could discover the functions of our residues. Using the various data mining and bioinformatic techniques to pull out relevant sections of journals referring to our residue(s) for experimental design which would reduce the time and cost of wet lab work. One such example presented itself when R80 and E112 appeared suspiciously close enough for a possible polar bond. Asking PyMol to show polar contacts in the structure had shown this was true as predicted (Figure 3.10). FiSHAAL (see below) is a semi-automated system that offers any researcher working on any protein the ability to quickly and cheaply decide what experiments to perform and discover which experiments have either been completed or which have nearby amino acids that were studied. Initial work of doing these techniques manually took much time to perform in a highly studied protein. It may take many months up to a year to extract all desired information and 100% automation may take only a few hours of search through computation, but if accomplished other orthologs/homologs of proteins can be analyzed.
Ortholog Background

Orthologs are pairs of homologous proteins or genes in different genomes, derived from common ancestry, that have the same function. If two homologous proteins or genes exist in the same genome, they are in-paralogs. (Note that for purposes of this discussion, we consider the nuclear genome and the mitochondrial or chloroplast genomes as separate entities. However, we do not often consider orthologous/paralogous relationships across such a vast evolutionary distance as that between the nuclear genome and the mitochondrial genome of the same species.) If two sequences a and b are orthologous to each other, from genomes “A” and “B” respectively, then the in-paralogs to ‘a’ are out-paralogs to ‘b’, and vice versa. Although these definitions are straightforward, the fact that these relationships require the reconstruction of evolutionary history from present sequences, it is not clear yet what the optimum methods are for identifying orthology/paralogy relationships (Fang et al. 10). Searching for orthologs using a query sequence to BLAST other species genomes, the ortholog is typically the top hit for each species. Then take this top hit and BLAST back to the subject genome, the top hit should be the query sequence Reciprocal Best Hit. However, this is not always the case. Some reasons for this may be incomplete genomes, deletions or additions to a species genome, splice variants, or even more distantly related species orthologous sequences that have diverged significantly since they shared a common ancestor. In order to analyze hyperconservational characteristics several orthologs were needed.

Orthologs were needed in as many organisms as possible to study hyperconservation. Automation would be a key component to obtaining orthologs of any given protein in order to save a large amount of time. I wrote a Perl-based program to be controlled through a browser in order to gather orthologs by the reciprocal best hits criterion according to bit score from a query sequence. The program interface was written to appear similar to the biology workbench, but to function only to locate orthologs. It would sometimes happen that two or more sequences from the same organism would share the highest bit score, in which case the choice between them was made arbitrarily. Since in these cases the differences were very slight, we judge that this did not affect any of the conclusions of the study. (See supplemental Perlprogram.pl (Appendix F))

There are several orthology-calculating programs available (Fang et al. 10). Some web-
based interfaces of calculated orthologs can be obtained from OMA (http://omabrowser.org/ (Schneider et al. 07)) used in hyperconservation of other subfamilies (see below), OrthoMCL (http://www.orthomcl.org (Chen et al. 06)) or InParanoid (http://inparanoid.sbc.su.se (Schmitt et al. 09). Orthologs can also be acquired from a workflow devised by Shreedhar Natarajan called MoLFunCs or Most Likely Functional Counterparts (Natarajan and Jakobsson, 09). MoLFunCs invokes “authority” in the form of prior annotation in combination with sequence based scoring in order to combine automation with human expertise in the definition of orthology. For analyzing other subfamilies including hyperconserved subfamilies the OMA browser was used for obtaining orthologs from the reference database in the TCDB.

**Obtaining newer sequences for hyperconservation comparison**

Original sequences were obtained from the TCDB website http://www.tcdb.org/tcdb/. The sequences obtained were all potassium channels from humans, if available, or other eukaryotes. A section of the sequence was then taken to the OMA browser at http://omabrowser.org/cgi-bin/gateway.pl. The human sequence was then taken as the reference entry for each representative subfamily through all the eukaryotic species available at the time on 11SEP09. The fasta file was then downloaded and put into text form. Orthologs are listed in OMA groups as: 1:1 Orthologs, 1:many Orthologs, many:1 Orthologs, and many:many Orthologs. If there was more than one ortholog entry from a species the sequences were BLASTed back to the human genome and the sequence with the highest percent identity or bit score and reverse BLASTed was retained. The sequences were put into the biology workbench and submitted to the program MVIEW (Brown et al. 98) which returned the percent identity in relation to the top human reference sequence for that representative subfamily. This was also done for the proteins of cytochrome C, mitochondrial DNA polymerase, myosin and myosinviib for comparison. These proteins are commonly used throughout the field for phylogenetic analysis. The color-coding, order and grouping were derived from a large phylogenetic study by Holton et al. 10. The species were placed in a tree (Figure 4.1) in the order used in figures 4.2-4.519-22. The percent identity values were used in the software Python(x, y) http://www.pythonxy.com/ and its programming language. A script written by lab mate Hui Liu was used to produce 86 graphs in the IDE program of the Python package (Supplement for Python x, y; barplot.py Appendix G).

Comparisons of the different example graphs show that most are not highly conserved. A
select few, are strikingly conserved along almost the entire length of the sequence KCNA2, CNGA3, KCND2, KCNMA1 and the hyperconserved sequence cytochrome C (Figure 4.2) have conserved sequences among more than 30 distantly related species from C. elegans to H. sapiens, that are very near 50% or greater identity to the human corresponding ortholog. From the survey of 86 potassium channel subfamilies, KCNMA1 has the highest conservation among all the potassium channels subfamilies. The graphs show that the organisms that are more distantly related to humans do not always have the least percent identity. Others appear only in chordates, such as CNGA4, KCNB2, and KCNH7 (Figure 4.3) and are examples of non-hyperconserved subfamilies. Still others have orthologs by the RBH criterion in invertebrates but only reach half identity in vertebrates such as KCNQ1, KCNF1, KCNJ1 and KCNV1 (Figure 4.4). These sequences seem to be undergoing typical evolution with most motifs not being important enough to keep alike or have other forces acting on making them different such as toxins affecting the variability of the turret region (Liu and Lin, 04), which can be seen in the ConSurf pictures 3.5. Unlike some other channel types, potassium channels seemed to have duplicated in the genome many times and then evolved other properties while the original channel remains highly conserved. This might explain why some subfamily members are variable. Hyperconservation among the select few potassium channels is apparent when compared to other proteins such as mitochondrial DNA polymerase, myosin, and myosinviib plus the ion channel CatSper 4 (Figure 4.5). The rest of the subfamily graphs can be seen in the supplement (extra figures Appendix H). The list of where the different subfamilies lie can be seen in Table 4.1. Some of the subfamilies are very close to the 50% identity criteria and may or may not be considered hyperconserved. Other graphs were also produced from all ortholog information seen above.

Orthologs of subfamilies and species comparisons can be seen in (Figures 4.6-8). In (Figure 4.6) there is a distinctive difference between the vertebrates and invertebrates with respect to the number of human subfamilies that are represented in each species. The number is lower in invertebrates than in vertebrates. This does not necessarily imply that there are fewer potassium channels or channel families in invertebrates. The number of orthologs to the channels does not show any significant variation (Figure 4.7). However, there are some non-chordate groups that have no corresponding subfamily members to humans (Figure 4.8) and may have other orthologs that humans do not.
3) Bioinformatics and Data Mining Issues, an introduction to the FiSHAAL process

The quick extraction of amino acid information from a vast array of authors and different journal requirements would make an elaborate integration of data mining and bioinformatics necessary. However, data mining problems are extensive. Deciding on which proteins to be included in an alignment is just one problem that requires a scientific basis of decision. Writing a program that probes the natural language of scientific articles and extract all the accurate and exact desired information is a major barrier of a fully automated data-mining project. Once a researcher decides on a particular protein, there are many different directions that need to be exhaustively searched in order to pull out all desired information with no false positives. Depending on the species a researcher is starting with, they need to search using all known aliases of the studied protein, all orthologs’ names from all other related species, including all their aliases of ortholog names.
CHAPTER 5: AUTOMATION UTILIZING FiSHAAL

There are several programs and databases that study proteins at the short domain/motif level such as InterPro or MEME (Bailey and Elkan, 94) / MAST (Bailey and Gribskov, 98). Some mutated residue databases are also available: http://www.receptors.org/NR/mutation/allmut_ID.html but are devoted to a specific group of proteins. There are challenges that the BioCreative (http://www.biocreative.org/) organization has posed to automatically extract, interpret and extrapolate journal information at the mutated residue level. From this, there are a number of labs that address this type of language processing dealing with mutated residues (Horn et al. 04; Rebholz-Schuhmann et al. 04; Baker and Witte 06; Gabdoulline et al. 06; McDonald et al. 06; Erdogmus and Sezerman 07; Kanagasabai et al. 07; Lee et al. 07; Yip et al. 07; Furlong et al. 08; Saunders and Perkins 08; Winnenburg et al. 09; Wong et al. 09; Yeniterzi and Sezerman 09; Laurila et al. 10) and many extensive comparison papers of their work to others can be seen at (Cohen and Hersh 05; Skusa et al. 05; Krallinger et al. 09; Harmston et al. 10). One web implemented site is similar in the presentation to our approach of an alignment of similar proteins, but only deal with proteins that they have selected for their mutated residues (http://3dmcsis.systemsbiology.nl/ Kuipers 10). Many others are no longer maintaining their projects like PASTA (Gaizauskas, et al 03). But little other work except (Nagel et al. 09) has been proposed to find articles that discuss protein residues regardless of whether or not they have been mutated. Many of these systems are limited by data mining techniques that require extensive natural language programming and complete automation, some include only analyzing abstracts, ignoring text in pictures or requiring key words to be within the same sentence to be valid. Many also focus on particular proteins or do not go far enough into the homology (ortholog/paralog) of protein residue functions. We propose FiSHAAL in which automation performs the majority of the work with the more difficult cases performed manually. Our end product is the identifications of all residue positions in every article of interest. In order to do that we need to identify all proteins in each article and match them to the alluded species to identify very specific sequences to match those residues mentioned. Instead of preparing programs that perform this from scratch we selected three programs to independently recall three parts and integrate the information to produce probable matches. A concise short flow chart with the natural language code can be seen in figure 5.1.

A note on terminology: a normalized term is the human lexical canonical name (HUGO
assigned) i.e. articles that call a human protein HBK5, HK4 HUKIV or any other alias is normalized to human KCNA2. Whereas the grounded term for that protein is the database specific string assigned to that object i.e. in UniProt it is P16389 or in RefSeq it is ID NP_004965.1 which is often called an accession number. For species we would take kid, patient, girl or any other human reference and normalize to Homo sapiens and ground that species to NCBI’s taxonomical reference of 9606. If we are discussing article identification the normalized name may be the title of the article or “Long et al. 05” and is grounded to PubMed ID 16002581. A detailed description of these terms can be seen at Witte and Baker 07. Residues are normalized from an example like “glycine at position 376” becomes G376 and do not have a ground per se but are linked to a grounded ID of a protein sequence. Most of the programs we use automatically output either the normalized and/or grounded term. Mapping is done through UniProt by taking one grounded ID in a particular database and identifying its equivalent in another database. Within FiSHAAAL we integrate three string identification specific programs that have been produced or used by the BioCreative challenges and other general programs/packages. For each PubMed article a modified 1) MutationFinder (Caporaso et al. 07) was used to list all residue mentions, 2) GeneTUKit (Huang et al. 11) to extract normalized protein names and 3) Linnaeus (Gerner et al. 10) to pull out species names. GeneTUKit and Linnaeus were used as written. MutationFinder was modified to find all residue mentions rather than just mutations, as follows: The scanning regular expression was modified to not require the “m” in the wNm output, where “w” is wild residue, “N” is its numbered position and “m” is the mutated residue. We found that occasionally situations in which an amino acid number appears to be mentioned are actually nomenclatures for solutions, compounds, equipment or some other characterization (Figure 5.2). These were later filtered out manually by verifying it as a false positive at the post processing stage of article analysis along with any unlocated residues. (Example snapshots from web site http://fishaal.beckman.illinois.edu/ see Appendix I, for the entire code used see Appendix J). With CLUSTALW as the alignment program, the alignment was sent for FiSHAAAL processing to present links to publications that mention any residue in that column.

In order to avoid some natural language analysis problems, the solution has been separated into three major groups of processes. 1. Obtaining all articles that may have homologous residues mentioned. 2. Processing all articles as automated as possible to place all
plausible residue mention positions with their sequences for each journal article, with a human manually locating the remaining anomalous residues. 3. Picking the reference sequence and all desired homologs for an alignment in which each column residues has links to PubMed articles that mention any one of them. Some articles give some type of accession number such as the SI tag in Medline, making finding information on a particular residue rather easy. Others give the protein name that is the usual and common name of the protein found in the database see (Table 3.4). Difficulty begins to arise when a given residue number in a particular named protein does not match the named protein in the database. The different names a protein can obtain can vary over the years as different nomenclature standards change. The exact protein for the exact name may have been changed in the database, but they cannot change in older articles. The database changes because of newer scientific discoveries and classifications within the databases. We overcome most of the naming problems by using GeneTUKit to find and normalize all potential protein names in each article so that further processing is then performed with all programs using the same name for each particular protein. We may add/update the dictionary synonym name list in the future to improve upon recall. Other problems arise when an author uses a particular motif or segment of a protein and re-numbers the residues starting with the beginning of their desired sequence motif. We can use a CLUSTALW alignment match of residues to renumber them back to the original sequence numbers such as; (Figure 5.3) Bright et al. 02 which shows S6 Shaker section as; (G4 K5 V7 G8 C11 V13 A14 G15 V16 L17 T18 I19 A20 P22 V23 P24)= (+452 amino acid number shift to G455 K456 V458 G459 C462 I464 A465 G466 V467 L468 T469 I470 A471 P473 V474 P475). In addition, as sequences become more and more divergent a correct and true alignment needs to be obtained to verify a correct residue number. In some articles more than one protein is described. (Table 3.6) Visan et al. 04 has Kv1.2 R354 E355 D363 Y377 V381 T383 and Kv1.3 H399 His399 or Val381. Writing code to distinguish which protein in which species an author is describing may be difficult. The difficulty is amplified when there is more than one species and/or more than one type of protein. Sometimes this information can be obtained within the same sentence, but it gets more difficult if it is in the same paragraph or only mentioned in one section of the article such as in the very beginning or in the research design and methods. So instead of computationally discerning what the language is describing we collect all protein names, species names and match mentioned residues to specific sequences from a UniProt protein query of all combinations of species and protein
names retrieved. The probability of residue matching not just by chance was calculated using the formula (Table 5.1) where p is the probability of a particular amino acid, as some are more likely than others, by the chart from http://cbrg.ethz.ch/Server/ServerBooklet/section2_11.html (Table 5.2) and N is the number of amino acids appearing in their respective position by chance. In simplest terms it is the average probability of a residue raised to the number of matching residues (see example Table 5.1).

Sometimes sequence information may only be given in the literature as just a few different amino acids and their number. With this limited information a sequence can be built and BLASTed or aligned using the given residue numbers and X's to fill in the undescribed residues. Creation of this Pseudo Filler Sequence (PFS) is performed by aligning just the mentioned protein with a sequence created with Xs between all the given numbered residues for a 100% match except Xed residues (PFS) (Table 3.7).

The process of collecting all journal articles that mention residues to a particular protein sequence using the canonical name and all its aliases requires an elaborate Boolean search string. This requires a dictionary of all synonyms of the protein name and all the synonyms of different species if desired. In future work we will produce a small script that creates a string that can be pasted into PubMed's search box. A synonym dictionary can be found at gpsdb.expasy.org/. To obtain an updated alias name list of KCNA homologs, several key names were queried to this database: KCNA*, KCNB*, KCNC*, KCND*, KCNF*, KCNG*, KCNV*, KCNS*, KQT, Kv1*, Kv2*, Kv3*, Kv4*, Kv6*, Kv7*, Kv9*, shaker, shab, shal and shaw (* is a wildcard). Redundancies were removed from resulting list. Then “+” replaced all spaces and 3 different field tags were added to each term(s): [mh] (mesh headings), [tiab] (title abstract), [tw] (text words) to limit false positive search results from other fields such as an author’s last name which may be “Shaker”. In between each term(s) is an “OR”. Here is an example: “+OR+voltage-gated+potassium+channel+HBK5[mh]+OR+voltage-gated+potassium+channel+HBK5[tiab]+OR+voltage-gated+potassium+channel+HBK5[tw]”. The complete query list (Appendix K plus instructions) was submitted in parts as the PubMed web site would not except 1515 sets of terms. Only PMIDs were collected, reduced to a unique list and then articles already in our database were removed. The resulting 7824 abstract listings as of 06SEP11 were scanned through manually to obtain articles that may contain residue mentions of our desired homologs. From that list an additional novel 580 articles were added to
our database of 870 from the 11SEP06 search results. These will be used for FiSHAAL processing only once the program is fully optimized. As work progresses on FiSHAAL automation of this process will be added. The majority of false positive resulting articles were about proteins that have official names that were once an alias name of a potassium channel protein such as; “HK4” or “F5”. Not much can be done to automatically remove this from the query results, but the Medline files can be filtered using date tags and keywords such as: “mutation”, “residue” or ”structure” so that only potential target articles are presented to the user.

Processing the journal articles will take up the most time computationally. In order to comprehensively collect all possible mentions of amino acid residues, full text and preferably tagged simple text needs to be provided to the program for optimal results. In this case, tagged text refers to clearly marked sections of the journal similar to Medline, html or XML format. This is preferred to reduce many false positives in which the program will then try to strip out most of the non-body objects i.e. references, acknowledgments, pre-title text etc, which unlike many other formats are clearly marked in this format. For all journal article file names, a PubMed ID number is required anywhere in the name in the format of PMID immediately followed by the number i.e. PMID12037559. If there is only have a hard copy of an article or there is only a picture embedded in the .pdf, a reader can write a correctly named simple text file that contains only residues, proteins and species. As long as the numbering scheme matches the mentioned protein/species our program should have no problem adding this article to the FiSHAAL alignment. In addition, foreign language papers (not scanned picture based) typically mention this information (with some exceptions of specie names) in characters that are readable by our program. Depending on the speed of the computer and the amount of manual curation needed this may require several days or more. The PubMed IDs will be checked against our own database to verify whether or not they have already been processed and novel files are prepared for the next stage. Journal articles are obtained in four basic file types that need to be formatted into a consistent simple single file type. The four types are .pdf, text images imbedded in a .pdf, .html and simple text. All of them need to be converted into a simple preprocessed text type format. To perform this we utilize the Linux programs pdftotext (http://linuxappfinder.com/package/poppler-utils), htmltotext (http://code.google.com/p/flaxcode/wiki/HtmToText), and in the future an OCR (optical character recognition) reader to extract text from figures and from entire journal article in which the document has been saved as an image. The next task is to
prepare the text to be formatted into the form that is required by MutationFinder. This is a simple format in which each article starts with the PubMed ID plus a tab character followed by the full text of a journal all on a single line with a new line character at the end. The end input format is a list of PubMed IDs, one per line in a single file. These are then fed into our modified MutationFinder program which outputs that same list of journal IDs, one per line, plus a tab followed by all normalized amino acid mentions including repeat mentions. We then reduce all redundant mentions to one. Residue mentions then need to be matched to their corresponding grounded fasta sequence. Occasionally authors provide some grounded database number information embedded in the Medline tagged output field SI [Secondary Source ID](Example Table 5.3). If all sequence identifying information pertaining to that article is given, then the matching residue information to those sequences is done by retrieving the protein name, species and fasta sequence of that entry using that grounded ID via command line services through the Internet by Python Django. If all mentions match, that PMID is permanently stored in our MySQL (mysql.com) database and tagged with the residue that mentions their corresponding grounded ID, normalized protein names, normalized and grounded species name which are also linked to their corresponding stored fasta. The PMID is marked as analyzed and removed from further processing. If residues are still unidentified they are sent to GeneTUKit and Linnaeus, which can both handle the same format of a directory full of text files. We use GeneTUKit to extract normalized protein names from each PubMed article and even though they offer species in their analysis we have found that it is often incorrect; therefore we use Linnaeus to produce our species list. We then use the output to create a Boolean search string to send to the UniProtKB (See URL example table 5.4) database to obtain the fasta sequences including variants of all possible combinations of species and protein names that were produced by both programs. The initial search combination list may be large but many combinations will not produce results, in addition, we will be checking and storing previous search combinations in order to save time and lessen Internet traffic. These sequences are then searched for matching residue mentions. A probability is calculated using the formula for probability seen earlier. Accuracy depends on the number of unique mentions an article gives; i.e. a match of 10 residues in a single sequence has a much lower probability of occurring just by chance whereas a single residue has the probability of occurring just by chance at its use frequency in any protein. If all residues mentioned by MutationFinder in each article are matched to a corresponding fasta
sequence then these are linked to that PMID. If a residue is matched to more than one fasta sequence then it will be assigned to a sequence that has the lowest probability of occurring just by chance. If both sequences have the same probability and same locations they may be highly conserved and both deserve to be linked to that PubMed ID, because they probably have the same function as homologous residues. In the future if any residue is ever shown to be a false positive there will be an option for a user to manually assign (through a caution window) a residue to a particular grounded fasta sequence. This step will also be available as a last step after all computationally automated steps are exhausted in which residue(s) could not be matched to a sequence. In future work for non-matching residues a pseudo-filler sequence will be produced to either search our own fasta database or to BLAST a database to find a sequence that matches the exact distance calculations between the residues mentioned in an article. As more functionality is added to the program the need for the manually curated last step for article processing will be reduced.

This last part will produce the FiSHAAL output. After the user has provided articles/PMID's or has knowledge that all articles had been processed pertaining to their studied protein and its homologs they will be asked to provide a normalized protein name and species or a grounded UniProtKB ID. This will invoke a series of automated steps that will send out various requests to Internet ortholog database sites. At this stage we are utilizing the results via command line (see URL examples table 5.5) from Omabrowser.org and InParanoid (inparanoid.sbc.su.se). Future implementation will include more databases. For the OMA browser we map the grounded gene ID using UniProt's mapping system to OMA's grounded ID. InParanoid can be given the UniProtKB directly. We retrieve the entire group and close group’s protein names and species via a UniProtKB ID if given. If only an Ensembl ID is available the script is directed to get the species name and protein name as well. Once all redundancies are removed the fasta sequences are sent to allow the user to choose which sequences, including any user added, to send to the program CLUSTALW to produce an alignment which is then processed to link to a list of PubMed titles/links that correspond to all amino acids that are mentioned somewhere in that alignment column. If a user hovers over individual amino acids, their number and position are displayed.

As mentioned earlier there are other difficulties in using a computer algorithm search for finding amino acid residues as seen in (Figure 5.4). Most authors give amino acid sequences as
single letters or the exact location in a sequence as a letter immediately followed by the number location in the sequence. In this example case by (Aguilar et al. 98), the author gives no number but a three-letter code for each amino acid in a sequence. Mutation Finder resolves the language nuances of describing a numbered mutated residue by using 759 different regular expressions to filter them out. In order to collect the additional residue non-mutation mentions we copied and then removed the required mutation section of the expressions and added them back to the expression list.
CHAPTER 6: RESULTS; WORKFLOW THROUGH WEB SITE PROGRAM INTERFACE

After registering/logging in, new folders may be added or the default will be used. In order to create a FiSHAAL alignment there are several required pre-processing steps to perform. The first step is to locate and upload the articles to be used in the analysis, as described in chapter 5 above, and submit them for processing. After all automated computation has been performed; the program will have stored a MySQL database containing article identifiers coupled with residues mentioned in those articles, and the residues placed in connection with a fasta protein sequence. There will also be a set of apparent residues which the automation was not able to place in a sequence. Each of these cases will be presented to the user as page of the article, which the user can inspect and make a decision as to where the residue belongs, or confirm that it will not be possible to identify from the literature the user may skip it. They may also deem the mention as a false positive (for example an “S4” which is not a residue but an identifier of the voltage sensor in a K+ channel). If the user clicks the “Skip” button for any reason, then this residue will not be accounted for in the final alignment (Figure 6.1). After all articles are processed the user needs to pick a grounded reference sequence (Figure 6.2), the user may also add sequences. This will cause an automated retrieval of non-redundant homolog sequence names from several ortholog databases in which a user can checkbox all sequences they wish to include (Figure 6.3) then click “Produce FiSHAAL” for the alignment. The next page is the FiSHAAL alignment (Figure 6.4) in which, if any residue was mentioned in any of our processed articles there would be a link for that entire column bringing the researcher to a clickable list of PubMed articles pertaining to any/all of those residues (Figure 6.5). A studied protein residue typically behaves similar to those other related and nearby residues (Fodor et al. 04) mentioned in this article list depending on the conservation of that column. Therefore, as in any software program there are many opportunities to improve it and aid in obtaining known results in-silico before planning in-vitro/vivo type of research. The current version of the software is a beta version, just now being made generally available. It has useful functionality and is an improvement over manual searching a body of literature but is fragile; i.e., it can freeze or crash operations that might seem reasonable. The possible future work (Chapter 8) is aimed in three fundamental directions. One is to improve the robustness of the existing functionalities, a second is to improve ease of use, and the third is to add functionality.
CHAPTER 7: CONCLUSION

Aligning sequences can be rather difficult. It seems that in some proteins motifs that should align do not align properly. A method that seems to show a correct alignment is cut the protein into segments of one or two motifs and then aligns the sequences. Showing these sequences in Sequence Logo displays important conserved residues as large letters. In lesser-studied proteins this technique would readily point out residues that need further study. Many motifs of known functionality appear in these hyperconserved clusters, such as the mechanism for opening and closing potassium channels. In recent times important residues such as a particular glycine with an alanine a few residues down from the selectivity filter were found to be the hinge of the gating mechanism and have been pointed out to be in conjunction with the function of the potassium channel (Jiang et al. 02). These residues have been clearly seen while viewing the logos, but the significance to their conservation could not be attached until the crystal structure of the closed and open channel were solved. In hindsight, these conserved areas were quite prominent but the reasons for their conservation have eluded the prospect of their function. In the future this type of conservation of residues may be a clue as to other functions of the potassium channel. Other hyperconserved proteins, CNGA3, KCND2, KCND3 and KCNMA1 might contain interesting characteristics that may be more conspicuous because of their hyperconservation. Other subfamilies are more variable in distantly related species. This displays a trend of duplication and variability of some potassium channels while an original remains conserved unlike several other types of channels. Some future work may be to investigate if there is a tendency of a reciprocal relationship between proliferation of paralogs and conservation of orthologs. Using the techniques described to obtaining residue specific ortholog research articles may enhance the efficiency of the wet lab research. By pointing out sections of sequence areas that have been heavily studied, such as the selectivity filter, the user can focus on highly conserved residues that do not appear to be studied at all, bringing a greater chance of new, intriguing and possibly highly useful discoveries.

The well-characterized permeation pathway is also highly conserved. This shows intuitively that other highly conserved areas of unknown function or characterization can be analyzed as an important structure in the physiology of all potassium channels that retain these conserved motifs. Further research is needed for hyperconserved areas of some highly conserved motifs, which are not well characterized. Studies have seen some areas of hyperconservation
that do not have a known function. Using bioinformatic techniques two residues in the X-ray structure of KCNA2; F85 and E112 have jumped out as highly conserved, yet undescribed residues. The function of these residues can be postulated by analyzing what is described about nearby residues and tested in-vitro. For these residues there is a strong case that they are involved in tetramerization of the animal (humans to roundworms) KCNA channel but also the human KCNB, KCNC, KCND and KCNG channels. These two appear to be especially important because they are the only two residues that are 100% conserved in the tetramerization domain among all these proteins. Other residue discoveries can be obtained by utilizing the same techniques.

Quickly finding out what is known about particular residues can be extremely difficult especially if it is a well-characterized protein. There are many problems with residue numbers being off for a variety of reasons. If a researcher could find all known information on a particular residue in Google Scholar, the time saved may be months for highly researched proteins. A program to pull out all the journals that contain this information would be highly useful. The time to accomplish this has been drastically reduced by employing the work flow of FiSHAAL. After acquiring a folder of journal articles the files can be provided to the program via a browser interface in which the end result is an alignment of protein sequences with clickable links to PubMed articles that discuss homologous residues in each column. A researcher may then discover what may be known or unknown of their own particular residue or nearby residues that they are interested in studying. Ideally, complete automation for all tasks are desired, but for optimal accuracy a user needs to manually analyze and answer final questions where the computer code calculations have not been able to determine the locations of residue mentions in an article. Journals may need to undergo a more standardized writing style, use tables or be required to provide information to specific text fields during submission to PubMed to alleviate this and other comparable problems. Journals that are already published cannot be changed, but may have these data added to their Medline format. Another problem is that older articles need to be converted to text searchable word processing format, because they are just pictures of the article. OCR programs may perform this but are not always accurate opening possibilities for false positives. The aliases of names can sometimes be a problem even if a complete database is used because of the chance that two or more proteins were once described with the exact same name. This may be overcome by residue number matching. Occasionally
articles contain characters that appear to be an amino acid residue and number but are referring to some other type of information. This type of false positive may never be fully processed automatically but will always require manual input. This leaves out calculations of the f-score from the BioCreative group, in which one requirement is full automation in processing. Although some manual input is required, the final in-silico output provides a sound foundation for a research to discover what has been observed about a particular area of their protein and decide on what in-vivo, in-vitro experiments to pursue.
CHAPTER 8: POSSIBLE FUTURE FUNCTIONALITY AND WORK

Investigate the orthologs of KCNB, KCNC, KCND, KCNG and possibly other channels need to be researched to view just how far F85 and E112 are conserved in other metazoans. More structure cluster analysis of these two residues needs to be performed as well. Another protein family to look up would be KCNMA as it is the most hyperconserved family studied so far.

Investigate the following hypothesis: Members of orthologous groups that have a large number of paralogs within individual genomes have a higher degree of conservation than members of orthologous groups that have few or no paralogs. This hypothesis would predict, for example, that CatSper channels would have less conservation across the metazoa than most other orthologous groups within the voltage-gated ion channel family. The underlying possible mechanism is that some proteins create necessary variation in evolutionary history by variation in the primary sequence, while others create necessary variation by duplicating, with the duplicates varying independently.

Analysis using techniques of FiSHAAL decreases the time needed to discover what is known about individual homologous residues. There is a logarithmic increase in the availability of genomic data and a substantial increase in the amount of publication articles that study them. This data enhances our ability to recognize and manipulate in silico different aspects of proteins as an adjunct to wet lab work.

The working FiSHAAL can have many new functions and ease of use capabilities added. Here is a list of items that can be done to improve the system:

1) Try to correct text flow if a .pdf is in text form but does not convert properly due to a poorly formed pdf (different character encoding, etc.) and add this function to include OCR reading of pictures embedded in the PDFs.

2) Enable a user to query KCNA2 AND Drosophila to search PubMed using our code KCNA2 AND fly would be mapped to our dictionary and produce all possible combinations of KCNA2 AND Drosophila synonyms e.g. (KCNA2, kv1.2, shaker, shaker-like, shaker potassium channel, Kv1., KCNA, HUK, HK, RBK, HAK, etc.). A decision would need to be made as how inclusive (or exclusive) the scientist wants the homologs to go. Do they want only orthologs or include paralogs and decide which species to include. We would also like to use this dictionary to provide a Boolean expression to copy/paste into the PubMed search box for a complete list of
all articles discussing those protein(s) for them to acquire.

3) MutationFinder - May require a few fixes to regular expressions due to possible false negative results.

4) Have a crash log and email sent to us in the event of a major error.

5) Improve GeneTUKit with the larger dictionary. Possibly find or improve on the dictionary in Linnaeus.

6) Give a species name and protein to UniProt and it returns everything that mentions that protein including (possible) interacting subunits.

7) More processing of converted pdf to text, cut out more possible false positive strings such as: author names being confused for protein names or species.

8) After converting a pdf to text, compare the file size to determine if they are proportionate to pdf file size. If it’s entirely blank, fetch abstract from Medline.

9) Adapt MutationFinder to prevent any previous marked false negatives.

10) Increase recall from the Linnaeus program by increasing its synonym library.

11) Search for any downloadable full text (ideally in a tagged format, as xml or html or even text is preferred over pdf) to improve speed and accuracy of document processing.

12) Currently using retmode text for one official symbol per gene to be extracted from the database. We can use other types to extract all (many variants etc.) genes pertaining to this grounded gene (maybe we can look at other retmodes and rettypes) and other tools offered to be used for possible unverifiable residues.

13) Currently we use exact name(gene_exact:76 AND human (in UniProt)), we can later backtrack when our sequences don't match, and use just gene name (not exact) and species (gene:76 AND organism:"Human [9606]")(obtain all variants).

14) Give options for the code to run starting from each module of the code, and give it the option of completing the run (so that in case it failed in the middle, a fix can be made, and can continue from the middle of the code instead of rerunning the entire block).

15) Run GeneTUKit on abstracts only to help to choose between proteins (maybe on Linnaeus also) for an increased probability of selecting the correct residue. In other words, the correct protein name will more likely be in the abstract. More contemporary Medline pages are increasingly vigilant in describing the normalized protein and organism (possibly grounded sequence) and we would suggest that certain properties of a study be required when depositing to
16) Reduce redundancies from Linnaeus and GeneTUKit output.

17) Create a second run that checks for equivalent distances between residues (PFSs). If only some match (if there was an insertion, deletion, or substitution) flag it when it is displayed for human processing.

18) When displaying a FiSHAAL alignment add an 'I think this is wrong' button that gives the user the option (after they verify they are sure they want to make changes and know what they are doing) to correct a grounded residue to a sequence, and the previous information will be stored as a backup from a probably false positive or falsely located residue.

19) If we have two proteins A, B and residues 1,2,3,4 (1, 2, 3 belong to A and 3 seems to also belong to B) with 4, then our program asterisks 3 and lets the end user know that this residue may or may not belong to that protein. We can confirm from user input whether or not 3 actually belongs in A and/or B.

20) Some grounded protein IDs do not have results in UniProt (but are in NCBI). So we will set up other database searches for an alternate ground ID.

21) Separate orthologs from paralogs (because of the difficulties of certain database tags, we presently can only decipher homologs).

22) In our list of homologs, if a different gene name appears from our ortholog group, we can search homologs for that UniProt name as well (in order to gather a larger list of equivalent homolog names). i.e.: kcna2 human = kcnaw chicken = Sh fly. We will only use the extended list to search our tagged PubMed id list of names. [Alternatively, we can find and use a better synonym dictionary]

23) Create alignment conservation highlighting using Boxshade or TEXTSHADE colors and/or use a Sequence Logo output.

24) Produce a script to dissolve multiple/partial articles to focus on main article in a pdf. This is to remove other article end or beginning pieces that run on to the same page as desired article as they typically do in a physical journal (this may only be an older article problem).

25) Have residue numbers at the top of PubMed results page as per original clickable results, but it isn't easily implemented at the moment. Example:
https://netfiles.uiuc.edu/aebecker/www/figures/y90.htm (but with species as well, instead of just the gene name)
26) Use mentioned residues in an alignment or BLAST to discover the unknown name of a protein. Most of the time they are very closely related proteins, often splice variants or alleles that are usually only off by plus or minus five residues.

27) Offer uploading only a list of PMIDs to then display two lists: one presenting the articles we have already processed in our database and the other showing the articles may wish to be added to the investigation.

28) Offer a user input box for suggesting improvements as this list is by no means complete.
CHAPTER 9: REFERENCES


binding site of M3: evidence that M3 is an old eutherian molecule with conserved recognition of 
a pathogen-associated molecular pattern." Journal of immunology (Baltimore, Md. : 1950) 

molecular basis of K+ conduction and selectivity." Science 280(5360): 69-77.


Fang, Gang et al. (2010). “Getting started in gene orthology and functional analysis.” 
PLoS computational biology 6 (3), e1000703.

amplitude and kinetics of a neuronal voltage-gated potassium channel." Journal of 
Neurophysiology 78(3): 1563-1573.

rat olfactory bulb neurons and a cloned Kv1.3 channel." Annals of the New York Academy of 
Sciences 855: 529-532.

current suppression in the olfactory bulb through multiple phosphorylation of Kv1.3." Journal of 
Neurophysiology 83(4): 2332-2348.


amino acid covariance in multiple sequence alignments." Proteins 56(2): 211-221.

recognition system for sequence variants of genes in biomedical literature.” BMC 
Bioinformatics. Feb 5;9:84.


Hofmann, Kay and Michael D. Baron., Boxshade version 3.3.1.


### CHAPTER 10: FIGURES AND TABLES

CNG and Potassium Channel Subfamilies α-Subunits

<table>
<thead>
<tr>
<th>HUGO name</th>
<th>Descriptive name</th>
<th>Some aliases or alias beginnings</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNG -</td>
<td>cyclic nucleotide gated channel</td>
<td>RCNC, OCNC, CCNC</td>
</tr>
<tr>
<td>HCN -</td>
<td>hyperpolarization cyclic nucleotide gated</td>
<td>BCNG, HAC</td>
</tr>
<tr>
<td>KCNA -</td>
<td>voltage-gated channel, shaker-related</td>
<td>Kv1. HUK, HK, RBK, HAK</td>
</tr>
<tr>
<td>KCNB -</td>
<td>voltage-gated channel, Shab-related</td>
<td>Kv2</td>
</tr>
<tr>
<td>KCNC -</td>
<td>voltage-gated channel, Shaw-related</td>
<td>Kv3</td>
</tr>
<tr>
<td>KCND -</td>
<td>voltage-gated channel, Shal-related</td>
<td>Kv4, RK</td>
</tr>
<tr>
<td>KCNF -</td>
<td>voltage-gated channel, subfamily F</td>
<td>Kv5, hH, IK</td>
</tr>
<tr>
<td>KCNG -</td>
<td>voltage-gated channel, subfamily G</td>
<td>Kv6, hH</td>
</tr>
<tr>
<td>KCNH -</td>
<td>voltage-gated channel, H</td>
<td>Kv10, Kv11, Kv12, HERG, eag, elk, erg</td>
</tr>
<tr>
<td>KCNJ -</td>
<td>inwardly-rectifying, J</td>
<td>Kir, ROMK, IRK, GIRK, HIR, CIR, BIR</td>
</tr>
<tr>
<td>KCNK –</td>
<td>two pore K</td>
<td>K2p, TWIK, TREK, TASK, TRAAK, THIK, TALK, TRESK</td>
</tr>
<tr>
<td>KCNMA -</td>
<td>large conductance calcium-activated-mSLO</td>
<td>KCa1</td>
</tr>
<tr>
<td>KCNN -</td>
<td>intermediate/sm conductance Ca++activated</td>
<td>KCa2, hSK</td>
</tr>
<tr>
<td>KCNQ -</td>
<td>voltage-gated channel, KQT-like subfamily</td>
<td>Kv7, KVLQT, ENB</td>
</tr>
<tr>
<td>KCNS -</td>
<td>voltage-gated channel, delayed-rectifier, S</td>
<td>Kv9</td>
</tr>
<tr>
<td>KCNT -</td>
<td>calcium-activated subfamily T</td>
<td>KCa4.1, KIAA1422, SLICK, SLO2.1</td>
</tr>
<tr>
<td>KCNU -</td>
<td>calcium-activated subfamily U</td>
<td>KCa5.1, Slo3, KCNMC1, Kcnma3</td>
</tr>
<tr>
<td>KCNV -</td>
<td>voltage-gated channel, delayed-rectifier, V</td>
<td>Kv8</td>
</tr>
</tbody>
</table>

Table 1.1. This is the list of the major subfamily groups of the potassium channel superfamily in humans from HUGO (Human Genome Organization)


![K+ chan SF region of Fishes, Frog and Chicken](image_url)

Figure 1.1a Fishes Frog and Chicken, S5, pore helix and S6 (or S1, S2 depending on the channel).
Figure 1.1b Human, S5, pore helix and S6 (or S1, S2 depending on the channel).

Invertebrates

Figure 1.1c Invertebrates, S5, pore helix and S6 (or S1, S2 depending on the channel).

Other Mammals Sequence Logo

Generated by http://www.bio.cam.ac.uk/seqlogo/logocgi

Figure 1.1d other Mammals, S5, pore helix and S6 (or S1, S2 depending on the channel).
Figure 1.1e Plants, S5, pore helix and S6 (or S1, S2 depending on the channel).

Figure 1.1f Rodents, S5, pore helix and S6 (or S1, S2 depending on the channel).

Figure 1.1g The TCDB logo, S5, pore helix and S6 (or S1, S2 depending on the channel) used to obtain residue positions to place in the SF Finder program.
Figure 1.2. position of the selectivity filter versus length of sequence. Using Ashok Palaniappan's algorithm modified from initial intentions the distance to the selectivity filter in reference to the entire length of the sequence was quickly calculated. The resulting values could then be entered into an Excel worksheet in which these graphs can be used to visualize the data. The area below the first line represents the number of amino acids before the selectivity filter. The top line is the total length of the sequence. They are a representation of all the sequences from the Chordates, Invertebrates, Plastids to the prokaryotes Cyanobacteria, Eubacteria and Archaea.
Figure 3.1. an alignment tool called BOXSHADE (Hofmann and Baron) using the defaults in the biology workbench to produce this conservation graphic of the species used in this study. There is very high conservation of KCNA2 orthologs (by criterion of reverse best hits) among species that had a last common ancestor more than 550 million years ago. Green represents total identity, yellow almost-total conservation of identity, blue conservation of similarity. This is a small section of the alignment, which consists mostly of gaps which are not conserved and are not shown, are represented by red diagonal lines. Human and cow are identical.
Figure 3.2. This is the representative basal sequence of the KCNA subfamily for humans. It was BLASTed to the human potassium channels in order to obtain the most common or similar sequence of all eight of the members from the KCNA subfamily. KCNA2 was obtained as the representative. Rat KCNA2 is also the sequence in which the x-ray structure has been solved.

<table>
<thead>
<tr>
<th>Species</th>
<th>Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. seahorse 5e-146_69yfish</td>
<td>50.7%</td>
</tr>
<tr>
<td>2. mosquito 7e-141_72yaphaenops</td>
<td>65.5%</td>
</tr>
<tr>
<td>3. fruitfly 7e-145_69yfrosophila</td>
<td>40.2%</td>
</tr>
<tr>
<td>4. lobster 2e-150_57yfamiliaris</td>
<td>60.1%</td>
</tr>
<tr>
<td>5. horseshoe crab 1e-194_64ylymulus</td>
<td>57.9%</td>
</tr>
<tr>
<td>6. pufffish 0.99ytetradon_nigro</td>
<td>69.0%</td>
</tr>
<tr>
<td>7. zebrafish 0.99ydanio_rerio</td>
<td>92.0%</td>
</tr>
<tr>
<td>8. human_KCNA2_1345413sp_P1 100.0%</td>
<td></td>
</tr>
<tr>
<td>9. cow 0.100_bos_taurus_g1 6384 100.0%</td>
<td></td>
</tr>
<tr>
<td>10. chimp 0.91ypan_troglodytes</td>
<td>91.8%</td>
</tr>
<tr>
<td>11. dog 0.99ycanis_familiaris</td>
<td>90.9%</td>
</tr>
<tr>
<td>12. rat 0.99yRattus_norvegicus</td>
<td>92.4%</td>
</tr>
<tr>
<td>13. mouse 0.99yMusculus_musculus</td>
<td>99.4%</td>
</tr>
<tr>
<td>14. rabbit 0.99yOryctolagus_cunia</td>
<td>99.3%</td>
</tr>
<tr>
<td>15. chicken 0.97yGallus_gallus</td>
<td>97.8%</td>
</tr>
<tr>
<td>16. frog 0.92yRana_catesbeiana</td>
<td>92.2%</td>
</tr>
<tr>
<td>17. zebra 1e-137_65yzebra</td>
<td>52.0%</td>
</tr>
<tr>
<td>18. roundworm 1e-87_50yCaenorhabditis</td>
<td>43.5%</td>
</tr>
<tr>
<td>19. roundworm 1e-82_47yCaenorhabditis</td>
<td>46.5%</td>
</tr>
<tr>
<td>20. flatworm 3e-123_66enythraeid</td>
<td>53.6%</td>
</tr>
<tr>
<td>21. squid 6e-132_62n-hole1geo_cpa</td>
<td>57.3%</td>
</tr>
<tr>
<td>22. bee 1e-145_69n_helix_mellifera</td>
<td>61.7%</td>
</tr>
<tr>
<td>23. flatworm 7e-42_33n_columnella</td>
<td>55.2%</td>
</tr>
<tr>
<td>24. jellyfish 1e-78_45yjellyfish</td>
<td>36.8%</td>
</tr>
<tr>
<td>25. zebrafish 4e-49_36n_zebra</td>
<td>27.7%</td>
</tr>
</tbody>
</table>

Table 3.1. 3 - 5, 7 - 22 are the species that are fairly well conserved and reverse BLASTed to human KCNA2. One, two and six reverse BLASTed as the second top hit from KCNA2. 23 and 25 are very low in conservation and do not reverse BLAST very close to human KCNA2. 24 does reverse BLAST to human KCNA2 but is too low in conservation of the sequence to be considered part of the hyperconserved subfamily.
<table>
<thead>
<tr>
<th>Name</th>
<th>Rev BLAST #</th>
<th>% Id</th>
<th>E Value</th>
<th>Bit Score</th>
<th>Which</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aplysia californica</td>
<td>y</td>
<td>69</td>
<td>5.00E-148</td>
<td>514</td>
<td>kcna2</td>
</tr>
<tr>
<td>Anopheles gambiae</td>
<td>y</td>
<td>73</td>
<td>7.00E-141</td>
<td>496</td>
<td>kcna2</td>
</tr>
<tr>
<td>Drosophila melanogaster</td>
<td>y</td>
<td>68</td>
<td>7.00E-145</td>
<td>511</td>
<td>kcna2</td>
</tr>
<tr>
<td>Pamphilus interruptus</td>
<td>y</td>
<td>67</td>
<td>9.00E-150</td>
<td>517</td>
<td>kcna2</td>
</tr>
<tr>
<td>Limulus polyphemus</td>
<td>y</td>
<td>69</td>
<td>1.00E-144</td>
<td>501</td>
<td>kcna2</td>
</tr>
<tr>
<td>Tetraodon nigroviridis</td>
<td>y</td>
<td>89</td>
<td>0</td>
<td>765</td>
<td>kcna2</td>
</tr>
<tr>
<td>Danio rerio</td>
<td>y</td>
<td>91</td>
<td>0</td>
<td>790</td>
<td>kcna2</td>
</tr>
<tr>
<td>HUMAN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>kcna2</td>
</tr>
<tr>
<td>Bos taurus</td>
<td>y</td>
<td>100</td>
<td>0</td>
<td>884</td>
<td>kcna2</td>
</tr>
<tr>
<td>Pan troglodytes</td>
<td>y</td>
<td>91</td>
<td>0</td>
<td>779</td>
<td>kcna2</td>
</tr>
<tr>
<td>Canis familiaris</td>
<td>y</td>
<td>98</td>
<td>0</td>
<td>499</td>
<td>kcna2</td>
</tr>
<tr>
<td>Rattus norvegicus</td>
<td>y</td>
<td>99</td>
<td>0</td>
<td>880</td>
<td>kcna2</td>
</tr>
<tr>
<td>Mus musculus</td>
<td>y</td>
<td>99</td>
<td>0</td>
<td>880</td>
<td>kcna2</td>
</tr>
<tr>
<td>Oryctolagus cuniculus</td>
<td>y</td>
<td>99</td>
<td>0</td>
<td>880</td>
<td>kcna2</td>
</tr>
<tr>
<td>Galbus galus</td>
<td>y</td>
<td>97</td>
<td>0</td>
<td>863</td>
<td>kcna2</td>
</tr>
<tr>
<td>Xenopus laevis</td>
<td>y</td>
<td>92</td>
<td>0</td>
<td>822</td>
<td>kcna2</td>
</tr>
<tr>
<td>Halocynthia rosetti</td>
<td>y</td>
<td>65</td>
<td>1.00E-137</td>
<td>479</td>
<td>kcna2</td>
</tr>
<tr>
<td>Caenorhabditis briggsae</td>
<td>y</td>
<td>50</td>
<td>2.00E-67</td>
<td>318</td>
<td>kcna2</td>
</tr>
<tr>
<td>Caenorhabditis elegans</td>
<td>y</td>
<td>47</td>
<td>1.00E-82</td>
<td>338</td>
<td>kcna2</td>
</tr>
<tr>
<td>Polyorchis penicillatus</td>
<td>y</td>
<td>45</td>
<td>1.00E-78</td>
<td>279</td>
<td>kcna2</td>
</tr>
</tbody>
</table>

Table 3.2. detailed results of each species is given. In each column, the name of the species, whether or not it reverse BLASTed, the percent identity, E Value, the bit score, and which human protein was the top hit in the reverse BLAST.
Figure 3.3. CLUSTALW from the biology workbench was used to produce this tree. There are two distinct groups, vertebrates and invertebrates. In the tree, each entry starts with the English name, and then the E-value, with the percent identity to human KCNA2 has given from the BLAST search. Then a "y" for yes and a "n" for no in regards to whether or not this sequence reverse BLASTed to human KCNA2, and if it did not, the number following is the amount of top hits before human KCNA2. And the amount of characters left in the name is the scientific name.

<table>
<thead>
<tr>
<th>Species</th>
<th>% Identity</th>
<th>e-value</th>
<th>Bit Score</th>
<th>Protein</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Schistosoma mansoni</td>
<td>70</td>
<td>1.00E-126</td>
<td>484</td>
<td>kcna1</td>
</tr>
<tr>
<td></td>
<td>68</td>
<td>3.00E-125</td>
<td>479</td>
<td>kcna2</td>
</tr>
<tr>
<td>2 Loligo opalescens</td>
<td>67</td>
<td>6.00E-135</td>
<td>482</td>
<td>kcna1</td>
</tr>
<tr>
<td></td>
<td>62</td>
<td>6.00E-132</td>
<td>473</td>
<td>kcna2</td>
</tr>
<tr>
<td>6 Apis mellifera</td>
<td>75</td>
<td>4.00E-145</td>
<td>494</td>
<td>kcna1</td>
</tr>
<tr>
<td></td>
<td>75</td>
<td>1.00E-145</td>
<td>493</td>
<td>kcna2</td>
</tr>
</tbody>
</table>

Table 3.3. The three species in this list is a continuation of table 4, but also describes which human protein is the top hit in their perspective reverse BLAST.
1. **Table 3.4.** is the list of the various aliases obtained from three different websites of the human KCNA subfamily and their orthologs. The first name in most of the numbered list is the HUGO applied nomenclature followed by all possible aliases that were found for that protein. Not all proteins have resources in which alias names can be easily obtained, so a method for data mining different names for a protein is being researched. (Xu Ling, personal communication)

2. **KCNA1** human, (HK1) (HUK1) (HK1) (MK1), mouse; (MK1) (MK1), rat; (IA) (IBK1) (RBK1) (CK1), dog; (CSMK1), blood fluke; (SKv1.1), trout; (Tsha1), sea squirt; (TuKv1) other; (KCPV) (EA7), (MGC126782) (MGC138385) (B4) (MCEPH) (KCA1-1) (A1804627) (MGC124402)

3. **KCNA2** human, (HUK5) (NGK1) (HUK4) (HK4), mouse; (MK2), rat; (RBK2) (RC5) (RAK), frog; xenopus; (KGHA2), trout; (Tsha2), other; (RK2) (B4) (KG2) (MGC50217) (B4) (AKR64A) (KCA1-2)

4. **KCNA3** human, (HPCN3) (HGK5) (HUK3) (HK3), mouse; (MK3), rat; (RGK5) (RC5) (KV3), cow; (BAK4), rabbit; (Kv1.3-g-bGlibenclamide-sensitive voltage-gated potassium channel), frog; (Kv1.3E), other; (MBK3) (n-channel) (PCN3) (MK-3) (KCA1-3)

5. **KCNA4** human, (HK1) (HPNC2) (HUK4) (HUK3), ferret; (FK1), rat; (RCR4) (RHK1) (RK3), pigeon; (oKv1.4), chicken; (oKv1.4), other; (RK4) (RK3) (MK4) (PCN2) (KCNA3) (KCA4) (KVA4) (MK1) (KCHAN) (MGC124445)

6. **KCNA5** human, (HK2) (HPCN1), mouse; (Kv1.5), rat; (RK7) (RK4), dog; (Delayed rectifier K+ channel), cow; (BAK5-DELATED rectifier potassium channel), OTHER; (Kv1) (RK3) (HC1) (HHR) (RK2) (HUK1) (PCN1) (MGC117058) (MG117059) (MGC5248)

7. **KCNA6** human, (EBK2), mouse; (MK1.5), rat; (RC2) (Kv2)

8. **KCNA7** human, (HAK6), other; (MK6)

9. **KCNA10** human, (Cyclic GMP gated potassium channel) (Kv1.8) (KCN1) (GM1962)

10. **SHAKER** fruit fly; (Sh) (Shaker), squid; (SquKv1A)

11. **CG12348** fruit fly; (PB, isoform B (Cg12348-pg, isoform g))

12. **KCNI** rabbit (CGMP-gated potassium channel)
**Table 3.5.** The highlighted residues and their numbers are taken directly from the journal (Baker et al., 98). They are an example of ideal conditions that an algorithm mining this journal would have minimal problems extracting the desired information (: E293 M356 A359 R362 R365 R368 F370). The single protein and the residue numbers align correctly to what is given in the database.

<table>
<thead>
<tr>
<th>A</th>
<th>285</th>
<th>A</th>
<th>287</th>
<th>A</th>
<th>305</th>
<th>A</th>
<th>437</th>
<th>N</th>
<th>338</th>
<th>A</th>
<th>393</th>
<th>A</th>
<th>335</th>
<th>A</th>
<th>355</th>
<th>A</th>
<th>271</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>286</td>
<td>M</td>
<td>288</td>
<td>M</td>
<td>305</td>
<td>M</td>
<td>438</td>
<td>M</td>
<td>339</td>
<td>M</td>
<td>394</td>
<td>M</td>
<td>336</td>
<td>M</td>
<td>356</td>
<td>M</td>
<td>272</td>
<td>H</td>
</tr>
<tr>
<td>S</td>
<td>287</td>
<td>S</td>
<td>289</td>
<td>S</td>
<td>307</td>
<td>S</td>
<td>439</td>
<td>S</td>
<td>340</td>
<td>S</td>
<td>395</td>
<td>S</td>
<td>337</td>
<td>S</td>
<td>357</td>
<td>S</td>
<td>273</td>
<td>S</td>
</tr>
<tr>
<td>L</td>
<td>288</td>
<td>L</td>
<td>290</td>
<td>L</td>
<td>308</td>
<td>F</td>
<td>440</td>
<td>L</td>
<td>341</td>
<td>L</td>
<td>396</td>
<td>L</td>
<td>338</td>
<td>L</td>
<td>358</td>
<td>L</td>
<td>274</td>
<td>H</td>
</tr>
<tr>
<td>A</td>
<td>289</td>
<td>A</td>
<td>291</td>
<td>A</td>
<td>309</td>
<td>A</td>
<td>441</td>
<td>A</td>
<td>342</td>
<td>A</td>
<td>397</td>
<td>A</td>
<td>339</td>
<td>A</td>
<td>359</td>
<td>A</td>
<td>275</td>
<td>A</td>
</tr>
<tr>
<td>I</td>
<td>290</td>
<td>I</td>
<td>292</td>
<td>I</td>
<td>310</td>
<td>I</td>
<td>442</td>
<td>I</td>
<td>343</td>
<td>I</td>
<td>398</td>
<td>I</td>
<td>340</td>
<td>I</td>
<td>360</td>
<td>I</td>
<td>276</td>
<td>I</td>
</tr>
<tr>
<td>L</td>
<td>291</td>
<td>L</td>
<td>293</td>
<td>L</td>
<td>311</td>
<td>L</td>
<td>443</td>
<td>L</td>
<td>344</td>
<td>L</td>
<td>399</td>
<td>L</td>
<td>341</td>
<td>L</td>
<td>361</td>
<td>L</td>
<td>277</td>
<td>L</td>
</tr>
<tr>
<td>R</td>
<td>292</td>
<td>R</td>
<td>294</td>
<td>R</td>
<td>312</td>
<td>R</td>
<td>444</td>
<td>R</td>
<td>345</td>
<td>R</td>
<td>400</td>
<td>R</td>
<td>342</td>
<td>R</td>
<td>362</td>
<td>R</td>
<td>278</td>
<td>R</td>
</tr>
<tr>
<td>V</td>
<td>293</td>
<td>V</td>
<td>295</td>
<td>V</td>
<td>313</td>
<td>I</td>
<td>445</td>
<td>I</td>
<td>346</td>
<td>V</td>
<td>401</td>
<td>V</td>
<td>343</td>
<td>V</td>
<td>363</td>
<td>V</td>
<td>279</td>
<td>L</td>
</tr>
<tr>
<td>I</td>
<td>294</td>
<td>I</td>
<td>296</td>
<td>I</td>
<td>314</td>
<td>I</td>
<td>446</td>
<td>I</td>
<td>347</td>
<td>I</td>
<td>402</td>
<td>I</td>
<td>344</td>
<td>I</td>
<td>364</td>
<td>I</td>
<td>280</td>
<td>I</td>
</tr>
<tr>
<td>R</td>
<td>295</td>
<td>R</td>
<td>297</td>
<td>R</td>
<td>315</td>
<td>R</td>
<td>447</td>
<td>R</td>
<td>348</td>
<td>R</td>
<td>403</td>
<td>R</td>
<td>345</td>
<td>R</td>
<td>365</td>
<td>R</td>
<td>281</td>
<td>R</td>
</tr>
<tr>
<td>L</td>
<td>296</td>
<td>L</td>
<td>298</td>
<td>L</td>
<td>316</td>
<td>L</td>
<td>448</td>
<td>L</td>
<td>349</td>
<td>L</td>
<td>404</td>
<td>L</td>
<td>346</td>
<td>L</td>
<td>366</td>
<td>L</td>
<td>282</td>
<td>L</td>
</tr>
<tr>
<td>V</td>
<td>297</td>
<td>V</td>
<td>299</td>
<td>V</td>
<td>317</td>
<td>V</td>
<td>449</td>
<td>V</td>
<td>350</td>
<td>V</td>
<td>405</td>
<td>V</td>
<td>347</td>
<td>V</td>
<td>367</td>
<td>V</td>
<td>283</td>
<td>V</td>
</tr>
<tr>
<td>R</td>
<td>298</td>
<td>R</td>
<td>300</td>
<td>R</td>
<td>318</td>
<td>R</td>
<td>450</td>
<td>R</td>
<td>351</td>
<td>R</td>
<td>406</td>
<td>R</td>
<td>348</td>
<td>R</td>
<td>368</td>
<td>R</td>
<td>284</td>
<td>R</td>
</tr>
<tr>
<td>V</td>
<td>299</td>
<td>V</td>
<td>301</td>
<td>V</td>
<td>319</td>
<td>V</td>
<td>451</td>
<td>V</td>
<td>352</td>
<td>V</td>
<td>407</td>
<td>V</td>
<td>349</td>
<td>V</td>
<td>369</td>
<td>V</td>
<td>285</td>
<td>V</td>
</tr>
<tr>
<td>F</td>
<td>300</td>
<td>F</td>
<td>302</td>
<td>F</td>
<td>320</td>
<td>F</td>
<td>452</td>
<td>F</td>
<td>353</td>
<td>F</td>
<td>408</td>
<td>F</td>
<td>350</td>
<td>F</td>
<td>370</td>
<td>F</td>
<td>286</td>
<td>F</td>
</tr>
<tr>
<td>R</td>
<td>301</td>
<td>R</td>
<td>303</td>
<td>R</td>
<td>321</td>
<td>R</td>
<td>453</td>
<td>R</td>
<td>354</td>
<td>R</td>
<td>409</td>
<td>R</td>
<td>351</td>
<td>R</td>
<td>371</td>
<td>R</td>
<td>287</td>
<td>R</td>
</tr>
</tbody>
</table>

48
Table 3.6. The vertical alignment from residues in the journal from (Visan et al., 04) employs the correct residue and sequence number but there is more than one way a residue is described and more than one protein is being described (Kv1.2 R354 E355 D363 Y377 V381 T383 and Kv1.3 H399 His399 or Val381).

Figure 3.4. This is a typical example of a section in a journal from (Wrisch and Grissmer, 2000) in which a data mining algorithm would need to be able to read and accurately extract information on a selected group of orthologous or MoLFunCs proteins in order to determine if this journal describes any residues relevant to a desired study. From this paragraph six different proteins are discussed and six different amino acids are described. The proteins are from six different species, one of which is written in more than one way. It describes residues only from three of the different proteins. It also represents the residue numbers in three different fashions.
Table 3.7. An example journal taken from (Bowlbry et al., 97) shows how extensive Bioinformatic techniques are required to verify exactly which protein sequence the author studied. The numbers were from an older outdated sequence reference and did not match the sequence numbers in the present day database. Only a few amino acids and their numbers were mentioned in the article (Kv1.3 Y111 Y112 Y113 Y137 Y449 Y479). This information was used to create a sequence with the referenced number of gaps represented as Xs to align to the present day sequence for appropriate insertion into the vertical alignment article reference numbers.
S3 = I254 – A275
S4 = L290 – K312
Pore = I361 – S371
S5 = M325 – A349
S6 = I385 – N412
S2 = F222 – A243
S1 = A162 – E183

Turret = E350 – P362
GYG

F85, E112

3.5a
Figure 3.5 (cont.)
Figure 3.5 (cont.)

S1 = A162 – E183
S2 = F222 – A243
S3 = I254 – A275
S4 = L290 – K312
S5 = M325 – A349
S6 = I385 – N412

Pore = I361 – S371

GYG

Turret = E350 – P362

F85, E112
Figure 3.5 (cont.)

S1 = A162 – E183
S2 = F222 – A243
S3 = I254 – A275
S4 = L290 – K312
Turret = E350 – P362
Pore = I361 – S371
GYG
S5 = M325 – A349
S6 = I385 – N412
F85, E112
Figure 3.5a-f. This is a series of cartoons produced at different angles with the ConSurf program in which an x-ray structure is color-coded depending on how conserved is each specific residue. A deep rich maroon red represents fully conserved residues. While white represents medium conservation and blue is completely un-conserved. Each figure shows a new angle or zoom level and a line pointing to either F85 or E112 is visible from each particular angle in addition to other characteristics of KCNA2.
Figure 3.6. This is a pictograph of the x-ray structure in tetramer form looking through the channel opening. It is in tube form in which the only completely visible residues are F85 and E112.
Figure 3.7a-b. These show the x-ray structure and an angle similar to figure 13 but use different colors and the space filling model for our 2 studied residues.
Figure 3.8 displays a CLUSTALW alignment of the highly conserved F85 and E112 among all the species in KCNA and the human subfamily channels KCNB, KCNC, KCND, and KCNG.

<table>
<thead>
<tr>
<th>Residue/ %</th>
<th>Dist(Å)F85/E112</th>
<th>&lt;F85-S</th>
<th>&lt;E112-S</th>
<th>&lt;Ss-Qs</th>
<th>Characteristical Function (experimental)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S84 / 70%</td>
<td>5.38/4.24</td>
<td>34.30</td>
<td>45.60</td>
<td>100.00</td>
<td>phosphorylation site &amp; C-type inactivation was reduced</td>
</tr>
<tr>
<td>I88 / Small</td>
<td>4.75/6.34</td>
<td>58.00</td>
<td>39.50</td>
<td>82.50</td>
<td>T1 interactions &amp; T1 interactions</td>
</tr>
<tr>
<td>N81 / Small</td>
<td>7.25/5.82</td>
<td>46.80</td>
<td>65.20</td>
<td>68.10</td>
<td>T1 interactions &amp; T1 interactions</td>
</tr>
<tr>
<td>R80 / 90%</td>
<td>7.32/6.77</td>
<td>54.70</td>
<td>62.00</td>
<td>63.30</td>
<td>T1 interactions &amp; candidate for long QT syndrome as a NT change and part of FFDR, motif, in which SqKv1A-G87 is a G change to R, functional expression increases</td>
</tr>
<tr>
<td>Y116 / 70%</td>
<td>8.37/6.10</td>
<td>45.00</td>
<td>75.90</td>
<td>59.20</td>
<td>loss of phosphorylation site; decrease in peak current with a quicker inactivation state, T1 interactions &amp; reduces Kvb1 hetero-oligo binding</td>
</tr>
<tr>
<td>A87 / Small</td>
<td>7.70/9.33</td>
<td>70.10</td>
<td>54.40</td>
<td>55.50</td>
<td>T1 interactions</td>
</tr>
<tr>
<td>P83 / 70%</td>
<td>7.65/9.68</td>
<td>80.00</td>
<td>51.10</td>
<td>48.90</td>
<td>T1 interactions &amp; part of FFDR, a highly conserved motif &amp; conserved end of layer I</td>
</tr>
</tbody>
</table>

Table 3.8. This table displays the proximal residues that are within 8 Å of both F85 or E112 or are in an acute triangular plane of all angles(F85:E112 = Query, S = subject) being less than 90 degrees to either show how the two highly conserved unmentioned residues may be influenced by these residues. The residue numbers are as specified in human KCNA2. A possible function of these residues may be quickly tested with the knowledge of the characteristics of nearby residues.
Figure 3.9 (cont.)

Figure 3.9a-h. as a result of the calculations these residues are made visible in the x-ray structure PDB file 2A79 at various angles using PyMol software with a detailed description of their proximity and conservation percentage in each figure. The blue residues are E112, the red residues are F85 and the yellow residues will each be the amino acid studied for the distance calculated per monomer within the monomer. The snapshot angle is at an optimal point where all three residues are at their farthest view from each other. The bond distance shown is the smallest integer in angstroms in which there is a bond distance, calculated by PyMol to the nearest pair of query/subject residues. The order of the figures start with the smallest value to the sum of the distances from the two query residues F85 and E112 and is still in the cutoff value of 8 Å.
Figure 3.10. is the calculated distance of the polar contacts between E112 (blue) and R80 (yellow) by PyMol in the x-ray structure 2A79. The red residue is F85.
Figure 4.1. is a phylogenetic tree in relation to humans according to a large study by Holton et al., 10. The order from the bottom corresponds to figures 19-22 below.
Preface of figures 4.2-4.5. are graphs in which the bars represent the percent identity in relation to the human ortholog of the subfamily. The legend colors are in phylogenetic groups ordered according to relative distance from humans. Within each group the species are numbered according to the evolutionary distance from humans, in accordance with figure 18 above (Holton et al., 10). The spectrum starts with Violet as the Mammalia, Aves (Amniota) 1-33. Followed by Frog, Fish (Amphibia, Sarcopterygii) Blue 34-39, Lancelet (Cephalochordata) Blue Green 40, Squirts (Urochordata) Green 41-42, Arthropoda Light Green 43-48, Leech (Annelida) Yellow 49, Limpet (Mollusca) Yellow orange 50, Nematoda Orange 51-54, and the last group being the Anemone (Cnidaria) in Red 55. Figures show the graph when the groups are ordered upward by distance from humans.
Figure 4.2 (cont.)
Figure 4.2 (cont.)

4.2c

4.2d
Figures 4.2a-e. are the main ortholog subfamilies that are hyperconserved. 19e is an example of hyperconservation from a shorter sequence.
Figures 4.3a-c. are ortholog subfamilies that had very few orthologs that are only in chordates and therefore not very highly conserved among a diverse set of species.
Figures 4.4a-d. are examples that have many orthologs among most species but are not particularly highly conserved among all of them.
Figures 4.5a-d. are examples of the same type of above graphs for comparison of proteins that are generally used in phylogenetic studies plus the cat ion selective channel CatSper 4.
<table>
<thead>
<tr>
<th>Hyperconserved sequences</th>
<th>Chordates only</th>
<th>Average conservation/few orthologs</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNGA2, 3</td>
<td>CNGA1</td>
<td>CNGA4</td>
</tr>
<tr>
<td>KCNA1, 2</td>
<td>HCN2, 4</td>
<td>HCN1, 3</td>
</tr>
<tr>
<td>KCND1-3</td>
<td>KCNB1, 2</td>
<td>KCNA3-7, 10</td>
</tr>
<tr>
<td>KCNH1, 5</td>
<td>KCNG1, 3, 4</td>
<td>KCNC1-4</td>
</tr>
<tr>
<td>KCNK9</td>
<td>KCNH7</td>
<td>KCNF1</td>
</tr>
<tr>
<td>KCNMA1</td>
<td>KCNK5, 7, 10</td>
<td>KCNG2</td>
</tr>
<tr>
<td></td>
<td>KCNQ2, 3, 5</td>
<td>KCNH2-4, 6, 8</td>
</tr>
<tr>
<td></td>
<td>KCNU1</td>
<td>KCNJ1-6, 8-16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KCNK1-4, 6, 12, 13, 15-18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KCNN1-4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KCNQ1,4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KCNS1-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KCNT1, 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KCNV1, 2</td>
</tr>
</tbody>
</table>

Table 4.1. are the categorization of where the different potassium subfamily channels belong in accordance with the type of conservation of their orthologs in the complete genomes of metazoa as of March 2011.
Figure 4.6. is the number of channels per species, which most species have at least half the number of the surveyed subfamilies until the vertebrate invertebrate line where there are much fewer channels per species (Red Arrow).

Figure 4.7. is the number of orthologs per channel, showing that some subfamilies have fewer orthologs but none are fewer than 12 (based on OMA orthology finder).

Figure 4.8. there are 6 groups of invertebrates. Here the number of groups that contain recognized human subfamilies are shown. Some have no channel members per group, which may mean that these channels have evolved after vertebrates evolved or these invertebrates have lost those channels.
Figure 5.1. After all articles are acquired, upload them to the program through its browser interface. It then processes them if they are new to the system. Only the body of the text are kept and is converted to MutationFinder(MF) format and run. The MedLine file of each article is checked to see if they gave an accession number. If so the residues from MF are compared for matching to the sequence, if they all match, the data is stored and the article is marked as complete. The remaining articles have all protein and organism names extracted by GeneTUKit and Linnaeus in which all combinations are used to identify all possible sequences that may match our residues. If remaining articles do not match they are run through a sequence number match corresponding to the numbers given in the residues. If these do not match then they are sent to the user via a series of web pages to decide where these residues belong. When the user completes this they can then offer an accession number to return a list of homologous sequences to be check-boxed and sent to FiSHAAL, which returns the alignment of links to PubMed articles that mention any one/all residues in that column.
changed with fresh IMDM containing 0.3 mg/ml of G418 and passed at 2–3 day intervals using a brief trypsin–EDTA treatment. The cells were seeded onto glass coverslips (diameter: 12 mm, Fisher Scien-

Figure 5.2. in this article by (Ahn et al., 05) a computer searching for residue numbers can be fooled by nomenclature for items other than residue numbers but have characteristics of an amino acid and number.

<table>
<thead>
<tr>
<th>Amino Acid</th>
<th>Residue Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>FWGKIVSLCVAGVLIALPVPIVSNF</td>
<td></td>
</tr>
</tbody>
</table>

Figure 5.3. example of renormalization. The program TEXSHADE (Beitz, 00) was used to create this alignment conservation picture. It shows how the sequence FWGKIVSLCVAGVLIALPVPIVSNF numbered by (Bright et al., 02) with the first “F” as number 1 is renormalized to correspond to a reference numbering for the original entire protein sequence, with the first “F” at number 393.
\[ p = \left( \sum_{a_1, a_2, \ldots, a_N} (p_1 + p_2 + \ldots + p_N) / N \right)^N \]

\[ p = ((0.0583 + 0.0407 + 0.0537) / 3)^3 = 1.3 \times 10^{-4} \]

Table 5.1 is the formula and an example calculation for one typical case the probability of these three mentions by chance is in an article F85, D86 and E111 to being a match to the mentioned human KCNA2. This is where \( p \) is probability of all of the \( N \) amino acids appearing in their respective positions by chance. \( a_1, a_2 \ldots \) are identifiers for the \( N \) amino acids and \( p_1, p_2, \ldots \) are the overall incidence for each of the amino acids.

<table>
<thead>
<tr>
<th>Amino acid</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alanine</td>
<td>Ala</td>
</tr>
<tr>
<td>Arginine</td>
<td>Arg</td>
</tr>
<tr>
<td>Asparagine</td>
<td>Asn</td>
</tr>
<tr>
<td>Aspartic acid</td>
<td>Asp</td>
</tr>
<tr>
<td>Cysteine</td>
<td>Cys</td>
</tr>
<tr>
<td>Glutamine</td>
<td>Gln</td>
</tr>
<tr>
<td>Glutamic acid</td>
<td>Glu</td>
</tr>
<tr>
<td>Glycine</td>
<td>Gly</td>
</tr>
<tr>
<td>Histidine</td>
<td>His</td>
</tr>
<tr>
<td>Isoleucine</td>
<td>Ile</td>
</tr>
<tr>
<td>Leucine</td>
<td>Leu</td>
</tr>
<tr>
<td>Lysine</td>
<td>Lys</td>
</tr>
<tr>
<td>Methionine</td>
<td>Met</td>
</tr>
<tr>
<td>Phenylalanine</td>
<td>Phe</td>
</tr>
<tr>
<td>Proline</td>
<td>Pro</td>
</tr>
<tr>
<td>Serine</td>
<td>Ser</td>
</tr>
<tr>
<td>Threonine</td>
<td>Thr</td>
</tr>
<tr>
<td>Tryptophan</td>
<td>Trp</td>
</tr>
<tr>
<td>Tyrosine</td>
<td>Tyr</td>
</tr>
<tr>
<td>Valine</td>
<td>Val</td>
</tr>
<tr>
<td>Unknown</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.2 from [http://cbrg.ethz.ch/Server/ServerBooklet/section2_11.html](http://cbrg.ethz.ch/Server/ServerBooklet/section2_11.html) contains the amino acids, their abbreviations and their frequency being used for all probabilistic computations. Unknown is \( X \), and others like \( B \), \( Z \) are not utilized.
Table 5.3. is an example of an SI tag in the MedLine page.

http://www.uniprot.org/uniprot/?query=%28gene_exact:KCNA1+AND+organism:9606%29+OR+%28gene_exact:KCNA3+AND+organism:9606%29+OR+%28gene_exact:GLY+AND+organism:8355%29+OR+%28gene_exact:hal.1+AND+organism:8355%29&format=fasta&include=yes

Table 5.4. is a sample of an example URL sent to UniProt to retrieve all possible fasta sequences from combinations, including variants, of species/protein names mentioned in the text.

http://omabrowser.org/cgi-bin/gateway.pl?f=GroupDownload&p1=MTFHTYS&p2=darwin and
http://inparanoid.sbc.su.se/cgi-bin/gene_search.cgi?id=p16389;idtype=geneid;all_or_selection=all;specieslist=190;scorelimit=0.05;rettype=xml;cgifields=specieslist;cgifields=idtype;cgifields=all_or_selection

Table 5.5. are two URLs sent to OMA and InParanoid to retrieve a list of homologs to the user entered accession number.

with a -GlyPheGly- sequence in place of the more common -GlyTyrGly- motif that contributes to the potassium selectivity of other potassium channels (27) (Fig. 1). Members

Figure 5.4. taken from (Aguilar et al., 98) this shows how residues in a motif can sometimes be described in three letter format. This would also be required to take into account when searching for residues in articles.
Figure 6.1. shows the options available for the user to select in the cases where automation could not successfully place a residue mention.
Figure 6.2. is where the user enters the accession number of their desired reference sequence.
Figure 6.3. is where the researcher decides which sequences they wish to include in the final alignment.
Figure 6.4. is an example cut-out of an alignment where the links in column 15 point to a list of articles that mention any one or all of these residues.

Somewhere in these articles there are mention(s) of the homologous amino acid you selected. In the future we will be giving you homologous (orthologous/paralogous) residues.

<table>
<thead>
<tr>
<th>PMID</th>
<th>Date</th>
<th>Authors</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>11222625</td>
<td>2001 Mar</td>
<td>Li</td>
<td>Multiple binding sites for melatonin on Kv1.3.</td>
</tr>
<tr>
<td>10828242</td>
<td>2000 Jun</td>
<td>Li</td>
<td>Localization and molecular determinants of the Hanatoxin receptors on the voltage</td>
</tr>
<tr>
<td>10719895</td>
<td>2000 Feb</td>
<td>Li</td>
<td>A localized interaction surface for voltage domain of a K+ channel.</td>
</tr>
<tr>
<td>10613917</td>
<td>2000 Jan</td>
<td>Li</td>
<td>alpha K(+) channel.</td>
</tr>
<tr>
<td>14642276</td>
<td>2003 Dec</td>
<td>Broomand, Amir</td>
<td>Conformational changes in the C terminus of Shaker K+ channel bound to the rat Kvbeta2</td>
</tr>
<tr>
<td>11055440</td>
<td>2000 Sep</td>
<td>Gandhi, C S</td>
<td>Reconstructing voltage sensor a voltage</td>
</tr>
<tr>
<td>9600245</td>
<td>1998 Apr</td>
<td>Scheffer, H</td>
<td>Three novel KCNA1 mutations in episodic ataxia type I families.</td>
</tr>
<tr>
<td>10206953</td>
<td>1999 Apr 23</td>
<td>Mathur, R</td>
<td>Ile channel function.</td>
</tr>
<tr>
<td>10928570</td>
<td>2000 Jul</td>
<td>Klockgether, T</td>
<td>The molecular biology of the autosomal</td>
</tr>
<tr>
<td>12799903</td>
<td>2003 Jun 17</td>
<td>Poveda, Jose A</td>
<td>Dihydropyridine Ca2+ channel antagonists and agonists block Kv4.2, Kv4.3 and Kv1.4 K+ channels expressed in HEK293 cells.</td>
</tr>
<tr>
<td>10613618</td>
<td>2000 Jan</td>
<td>Hong, K H</td>
<td>The limit.</td>
</tr>
</tbody>
</table>

Figure 6.5. is a cut-out example of the list of links to PubMed articles to the particular column a user clicked on. The links here are directly to http://www.ncbi.nlm.nih.gov/pubmed/PMID#
APPENDIX A: SFFINDERV2.0
This is for the file program SFFinderv2.0 used in chapter 2.
APPENDIX B: KCNA2VERTICLEALIGNMENT.XLS
This is for the file kcna2vertalignment.xls
APPENDIX C: SUPPLEMENT REFERENCES
This is for the file supplementreferences.doc
APPENDIX D: PDBDISTBF85.XLS
This is for the file kcna2vertalignment.xls pdbdistBF85.xls.
APPENDIX E: PDBDISTBE112.XLS
This is for the file kcna2vertalignment.xls pdbdistBE112.xls.
APPENDIX F: PERLPROGRAM
This is for the file Perlprogram.pl.
APPENDIX G: PYTHON X, Y; BARPLOT.PY
This is for the file barplot.py.
APPENDIX H: EXTRA FIGURES
APPENDIX I: EXAMPLE SNAPSHOTS FROM WEB SITE

Final product produces an alignment in which homologous residues in a column point to all PubMed articles that mention any one of them.
Article Upload

Only .pdf, .html, and .txt accepted. The file name must contain “PMID1234567” somewhere. (1234567 being the PubMed ID of that article)

Do not put a space between the two, and do not have any additional numbers following the PMID.

Example: MacKinnon.PMID16002581Kchannelstructure,science05.txt

Please click Full Article Preferences for a full description of desired format.

Upload Queue
0 Files Uploaded

Click here to Select File Cancel All Uploads

Click here when you are done

Active Folder: asdf

Upload a sequence file. (This will come later)

Only fasta Input Formats

Grounded ID: Protein Name: Species: Description:

Sequence:

Save Sequence

Copyright (C) 2011, University of Illinois.
APPENDIX J: FISHAAL CODE
This is for the files in the folder FiSHAALcode
APPENDIX K: THE COMPLETE QUERY LIST
This is for the file completequerylist.txt