Visualizing the Protein Sequence Universe

Larissa Stanberry[*]
Bioinformatics &
High-throughput Analysis
Laboratory, Seattle Children's
Research Institute (SCRI);
DELSA
larissa.stanberry@
seattlechildrens.org

Roger Higdon
Bioinformatics &
High-throughput Analysis
Laboratory, SCRI; DELSA
roger.higdon@
seattlechildrens.org

Winston Haynes
Bioinformatics &
High-throughput Analysis
Laboratory,SCRI; DELSA
winston.haynes@
seattlechildrens.org

Natali Kolker
High-Throughput Analysis
Core, SCRI; DELSA
natali.kolker@
seattlechildrens.org

William Broomall
High-Throughput Analysis
Core, SCRI; DELSA
bill@quantumlinux.com

Saliya Ekanayake
School of Informatics and
Computing and Pervasive
Technology Institute, Indiana
University
sekanaya@indiana.edu

ABSTRACT

Modern biology is experiencing a rapid increase in data volumes that challenges our analytical skills and existing cyberinfrastructure. Exponential expansion of the Protein Sequence Universe (PSU), the protein sequence space, together with the costs and complexities of manual curation creates a major bottleneck in life sciences research. Existing resources lack scalable visualization tools that are instrumental for functional annotation. Here, we describe a new visualization tool using multi-dimensional scaling (MDS) to create a 3D embedding of the protein space. The advantages of the proposed PSU method include the ability to scale to large numbers of sequences, integrate different similarity measures with other functional and experimental data, and facilitate protein annotation. We applied the method to visualize the prokaryotic PSU using sequence alignment scores. As an annotation example, we used the interpolation approach to map the set of annotated archaeal proteins into the prokaryotic PSU. Transdisciplinary approaches akin to the one described in this paper are urgently needed to quickly and efficiently translate the influx of new data into tangible innovations and groundbreaking discoveries.

Categories and Subject Descriptors

J.3 [**Computer Applications**]: Life and Medical Sciences—*Biology and genetics*; H.3.3 [**Information Systems**]: Information Storage and Retrieval—*Information search and retrieval*

Keywords

MapReduce, data-enabled life sciences, sequence similarity, computational bioinformatics, protein annotation, protein sequence universe, PSU, COG, UniProt, UniRef, DELSA, multidimensional scaling, data visualization, BLAST, Azure, Sammon, Twister, Hadoop, Needleman-Wunsch, Hive, MPI, EM.

1. INTRODUCTION

Functional annotation of newly sequenced genomes and meta-genomes is one of the principal challenges of modern biology. Rapidly advancing sequencing technologies generate peta- and even exabyte scale data, exponentially expanding the PSU (see Table 1) [41, 43, 10]. Assigning functions to this glut of newly sequenced proteins is an immense computational challenge that requires advanced analytical tools and scaling capabilities [47, 50, 40, 38, 31].

Protein functional annotation relies on expert knowledge along with sophisticated statistical and machine-learning methods including pairwise and multiple sequence alignment algorithms, structure prediction models, motif and domain finding algorithms, and clustering methods [2, 3, 44, 48, 53]. The size and complexity of data from high-throughput technologies require the methods that can cohesively integrate information on protein expression, pathways, structure and functional annotation across different experiments, organisms and conditions, and to put these data into context with sequence information [28].

Functional annotation is typically done on a protein-by-protein basis. While this 'manual' approach is feasible for a small group of proteins, it quickly becomes unsustainable as the volume of sequences expands [17, 6]. In functional and comparative genomics approximately 30% of proteins in any newly sequenced genome have unknown function [7, 18, 31]. This barrier remains relatively constant as more new organisms are sequenced. The influx of data from novel

[*] Corresponding author, 1900 9th Ave, C9S-9, Seattle, WA 98101, larissa.stanberry@seattlechildrens.org

ECMLS'12, June 18, 2012, Delft, The Netherlands.

Table 1: Definitions of keywords and abbreviations used in this paper.

Abbreviation/Keyword	Definition
ActiveMQ	Apache publish-subscribe environment; http://activemq.apache.org/.
Apache Hadoop	A software framework that supports data-intensive distributed applications and provides a distributed file system that stores data on the compute nodes, allowing for high aggregate bandwidth across the cluster; http://hadoop.apache.org/.
Apache Hive	An open source software designed to run data warehouse-styled operations against large datasets stored in Hadoop Distributed File System. Hive allows projecting an RDBMS-like structure onto the stored data and run queries against those structures using HiveQL language; http://hive.apache.org/.
Azure, Microsoft Windows	Provides on-demand compute and storage to host, scale, and manage applications on the internet through Microsoft datacenters. The NCBI BLAST on Windows Azure is a cloud-based implementation of the NCBI BLAST tool; http://research.microsoft.com/en-us/projects/azure/azureblast.aspx.
BLAST	A heuristic algorithm which is optimized to identify local alignments with high sequence similarity. After optimal alignments are determined, BLAST calculates a bit score and an e-value for each alignment where the latter considers both the bit score and additional information about search database size and the scoring system http://blast.ncbi.nlm.nih.gov/Blast.cgi [2, 3].
COG	Clusters of Orthologous Groups of proteins database developed by NCBI. The database is separated into COGs for prokaryotic genomes and KOGs for eukaryotic genomes; http://www.ncbi.nlm.nih.gov/COG/ [52, 53].
DELSA Global	The mission of the Data-Enabled Life Sciences Alliance is to accelerate the impact of data-enabled life sciences research on solutions to the pressing needs of our global society; http://delsaglobal.org/.
EM	Expectation Maximization is an iterative algorithm used to find maximum likelihood estimators of the underlying distribution for incomplete data or data with missing values.
KOG	Clusters of orthologous groups for eukaryotic genomes; http://www.ncbi.nlm.nih.gov/COG/ [53].
MapReduce	A computational paradigm, where the application is divided into many small fragments of work, each of which may be executed on any node in the compute cluster.
MDS	Multidimensional scaling finds a low-dimensional Euclidean representation of data given the matrix of pairwise similarities. The classical MDS estimates the projections so that the relation between the resulting interpoint distances and the original similarities is linear.
MPI	The Message Passing Interface designed for high performance on massively parallel machines and on workstation clusters; http://www.mcs.anl.gov/research/projects/mpi/.
NW	Needleman-Wunsh dynamic programming algorithm is used to find the highest-scoring global alignment of two sequences.
PlotViz	A visualization software developed by SALSA group at Indiana University; http://salsahpc.indiana.edu/plotviz/ [45].
PSU	Protein Sequence Universe is the totality, or the aggregate, of all the protein sequences that exists in nature. PSU is also an interactive visualization framework with scalable software architecture. When developed the framework will allow users to explore, browse, analyze, and visualize protein data; http://manxcatcogblog.blogspot.com/.
Sammon's loss	A cost function for nonlinear MDS with an emphasis on preserving small distances [46].
Sequence similarity	A score that gives the degree of matching between the two compared sequences. The examples include BLAST, NW and Smith-Waterman scores.
Twister	An open source implementation of Iterative MapReduce that supports more efficient and broader range of communication collectives (including reduce, gather, and broadcast in an MPI language) in the Reduce phase of MapReduce; http://www.iterativemapreduce.org/.
UniProt	The Universal Protein Resource for protein sequence and annotation data; http://www.uniprot.org/.
UniRef	The UniProt Reference Clusters database that groups members based on sequence similarity. UniRef is composed of the distinct databases UniRef100, UniRef90, and UniRef50, that have 100%, 90%, and 50% sequence similarity, respectively, within protein clusters and reduce the UniProt database size by approximately 10%, 40%, and 70%, respectively. Each cluster contains one reference sequence and all proteins within the similarity threshold to the reference. UniRef retains annotation from all members of the protein cluster to prevent information loss; http://www.ebi.ac.uk/uniref/.

sequencing technologies creates an ever expanding backlog of un-annotated proteins, or "hypothetical", proteins [7, 32, 29, 18]. In addition to this backlog, a growing number of databases can no longer sustain the expansion including some of the most popular resources like the Clusters of Orthologous Groups database (COG; see Table 1, [53]). Last updated in 2006, the COG database remains one of the most popular scientific resources (over 6K citations according to Google Scholar).

The first of a kind all-versus-all sequence alignment of 9.9 million UniRef100 [51] proteins demonstrated the computational complexity of functional annotation [31]. The alignment on Microsoft Windows Azure with 475 eight-core virtual machines took six days to run and produced over 3 billion records. Consequently, 5.1 million (68%) bacterial proteins were assigned into COG clusters. The remaining 2 millions were classified into functional groups using an innovative implementation of a single-linkage algorithm on a Hadoop compute cluster using Hive and the MapReduce paradigm (Table 1). Similarly, the eukaryotic database was expanded by over 1 million proteins and 100,000 new functional groups.

The UniRef clustering project showed both the promise and the challenge of protein annotation. Public annotation resources are struggling to cope with the influx of data and, as a result, are either no longer supported [53, 34, 33] or provide limited interactive and analytic capabilities [24, 26]. Comprehensive functional annotation of large scale data requires a wide range of skills and tools including expert knowledge, manual curation, compute power, and analytic methods with scaling capabilities.

Because functionally similar proteins tend to cluster together, visualizing proximity of hypothetical proteins to the existing functional groups can significantly simplify the task of functional annotation. One approach to PSU visualization is through low-dimensional embedding of sequence similarity data. Methods for low-dimensional embedding include MDS, principal- and independent component analyses, spring embedding, feature selection and others [9, 19, 23, 22].

Visualization methods for biological data proposed in the literature include BioLayout [13] and Large Graph Layout (LGL) [1]. Both methods implement graph layout algorithms to visualize the network. Large volumes of data may affect the performance and utility of the visualization methods. Indeed, the BioLayout rendering limit of 45,000 nodes and 5 million edges is only a quarter of the COG database size. The software also does not allow an iterative update and the layout has to be recomputed for the entire data set with each expansion. The LGL method appears to be no longer available.

In this paper, we propose a PSU, an exploratory tool to enable protein annotation. The tool provides a low-dimensional visualization of data using a parallel MDS implementation on cloud and HPC systems with Iterative MapReduce, the standard Message Passing Interface (MPI; see Table 1), and threading. The implementation allows for iterative expansion by interpolating the new experimental data into the existing universe. When fully developed, the PSU would provide interactive, exploratory tools to examine complex biological data both independently and in the context of the existing information. As an example, we apply the method to create a 3D projection of the prokaryotic PSU. Prokaryotes are one of the four major biological kingdoms. To demonstrate the utility of the method as a tool for functional annotation, we interpolate the positions of the archaeal proteins and discuss the implications of the result in the context of functional annotation.

2. MATERIALS AND METHODS

2.1 COG Database

A major principle of molecular evolution is that functionally important proteins tend to be conserved across species. The COG database was developed by the National Center for Biotechnology Information (NCBI) [53]. The project constructed clusters of proteins from 66 prokaryotic and seven eukaryotic genomes. For each protein, the best aligned protein in every other genome was determined using a sequence similarity search [2]. If three proteins from three organisms were mutual best hits, they created a triple. COGs are the result of exhaustive, successive merging of triples with two common members. Manual curation of the clusters was done by experts to ensure correct grouping and functional annotations. The COG database is separated into COGs for prokaryotic genomes and KOGs for eukaryotic genomes [52, 53]. The database was last updated in 2008 and is not currently maintained.

In this paper, we are using the COG database of prokaryotic genomes that we will refer to as COGs.

2.2 Archaeal Database

The archaeal clusters of orthologous genes (arCOGs) contains 120 archaeal genomes with over 250,000 protein-coding genes that are classified into 10,335 arCOGs. The expert annotation of arCOGs was based on the COGs, the Conserved Domains and Protein Classification and the homolog annotation in the nonredundant nucleotide database [56]. The archaeal proteins were classified into 10,335 archaeal functional groups (arCOGs) that were further assigned to COG clusters. The current version of the database covers 91% of 120 archaeal genomes.

2.3 Multi-Dimensional Scaling

The MDS algorithm was used to project the protein sequence data into a low-dimensional space [9]. The method uses a dissimilarity matrix to estimate the positions in the lower dimensional space while preserving the dissimilarity between the sequences. Here, we optimize Sammon's loss function [46] given by

$$H = \sum_{\substack{i,j=1 \\ i<j}}^{n} \frac{(f(\delta_{ij}) - d(x_i, x_j))^2}{f(\delta_{ij})}, \tag{1}$$

where δ_{ij} is the dissimilarity measure between sequences i and j and d is the Euclidean distance between the corresponding 3D projections x_i and x_j. Function f in equation (1) is a monotone transformation of dissimilarity measure. The transformation f is chosen heuristically to increase the range of dissimilarity measures. The denominator in (1) ensures a larger contribution from smaller dissimilarities thus making the clustering structure of the data more apparent.

Equation (1) shows that projections x_i are mutually dependent. Hence the parallel MDS implementaion is accomplished by splitting the data into parts, computing the projections for each part using the mapping results for other

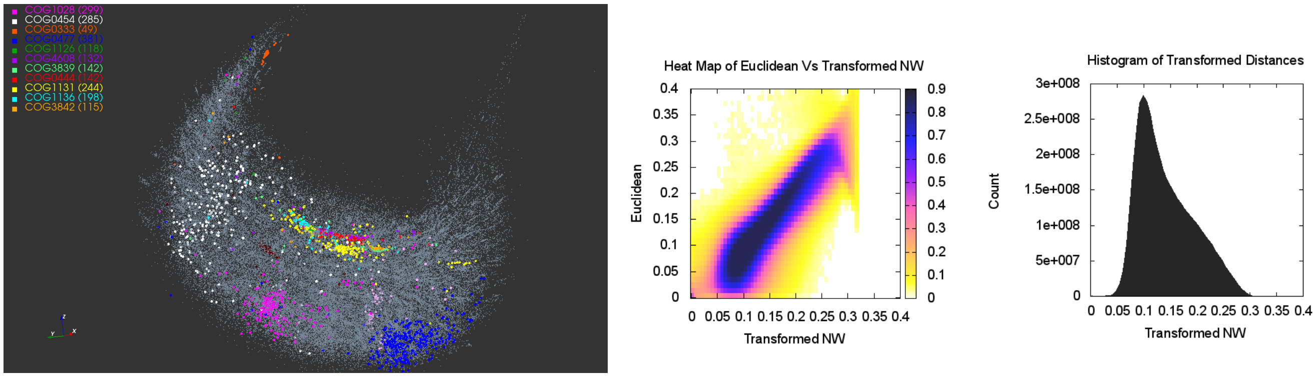

Figure 1: (left) MDS representation of the 100,000 sequences from well-characterized COGs in prokaryotic PSU. Each point represents a protein sequence. Eleven COG clusters were color-coded as marked in the legend. The number of proteins in each cluster is in parentheses; (center) the heatmap of the transformed NW distances versus the Euclidean distances between the MDS projections and (right) the histogram of transformed NW distances for 100,000 COG proteins.

parts and merging the mapping results. The iterations are stopped when the layout is stable, i.e. the projections do not change significantly after an iteration step. The MDS method has an $\mathcal{O}(n^2)$ computational complexity to map n sequences into 3D. Here, we used an expectation maximization (EM) approach to minimize the loss function [35, 8].

2.4 Interpolation

Large volumes of newly generated high-throughput data require efficient processing methods. To enable efficient mapping of newly sequenced proteins into the existing universe, we used an interpolation approach [4]. The approach uses pre-computed MDS projections for a sample of sequences to estimate the positions of new elements.

1. Map the initial set of n sequences using MDS and let $x_1, \ldots, x_n$ denote the corresponding projections.

2. For each new protein sequence, compute n dissimilarity measures δ_{ip}, where i and p index the original and new sequence, respectively.

3. For each new sequence, identify its K nearest neighbors among the original n proteins.

4. Estimate the projection x_p of the new protein by minimizing the loss function

$$H(x_p) = \sum_{i=1}^{n} (f(\delta_{pi}) - d(x_p, x_i))^2 / f(\delta_{pj}). \qquad (2)$$

Equation (2) shows that in interpolation, the objective function is optimized independently for each new protein p. Therefore, the computations can be easily parallelized and hence, the algorithm can be scaled to handle large data. The interpolation approach requires $\mathcal{O}(n)$ operations [4].

2.5 Implementation

We used a scaled, parallel traditional MPI with threading intranode for minimizing the loss function [15]. In the Reduce phase of MapReduce, we used Twister (see Table 1) [54, 12]. In Twister, all communication avoids using intermediate disk and is built around ActiveMQ (see Table 1) in Java Twister and around Azure primitives in the Microsoft cloud.

The method was applied to obtain a 3D projection of sequences in COG and archaeal databases. Initially, we applied MDS to create a low dimensional representation of COG consensus sequences. A consensus sequence was computed for each COG cluster separately and reflects the consensus of residues across the alignment columns. The consensus sequences were mapped into the 3D space using the MDS approach. The projections of consensus sequences were used to interpolate the coordinates of the COG protein sequences as described in Section 2.4.

We used sequence alignment scores as proximity measure. All pairwise distances were calculated using an MPI implementation of the Needleman-Wunsch (NW, see Table 1) alignment algorithm. The NW algorithm was realized by a parallel computation on the 24-core node system. The efficiency of the parallel distance computation was less than that of MDS due to saturation of memory bandwidth.

The distances were normalized to $2\delta_{ij}/(\delta_{ii} + \delta_{jj})$ to account for the sequence length effect. Then, we applied a monotone $\log(1 - \delta_{ij}^6)$ transformation to the normalized distances. This nonlinear transformation shortensl distances between similar sequences while magnifying distances between those with low alignment score. For MDS of consensus sequences, we used an MPI implementation of the nonlinear MDS with random initialization[27]. For interpolation, we set $K = 20$. The calculations were performed on a 768 core Microsoft HPC cluster. The resulting 3D projections were visualized in PlotViz (see Table 1) [45]

The NW distance calculation required one day to complete and the MDS job ran for three days. The parallel efficiency of the code was approximately 70% based on earlier studies that discuss both the inter-node and intra-node cases and find that it is essential to adopt a hybrid model with intra-node threading and MPI between nodes [42, 16]. All software used to analyze and visualize the data is open source. The results of the MDS analysis including estimated coordinates, parameters and captures are available at http://manxcatcogblog.blogspot.com/.

3. RESULTS

3.1 COG Database

Figure 1 (left) shows the 3D rendering of the prokaryotic PSU with each point representing a protein sequence. The figure shows the complexity of the PSU and the presence of distinct grouping structure. We color-coded eleven COG clusters in Figure 1 to illustrate the diversity of the underlying protein groups with respect to their location, shape, dispersion and size. While some clusters are rather tight, others are scattered throughout a sizeable domain. For example, compare the tight COG0333 cluster of ribosomal protein L32 with the diffuse COG0454 (HPA2) and COG0477 (Permeases of the major facilitator superfamily); see also Table 2.

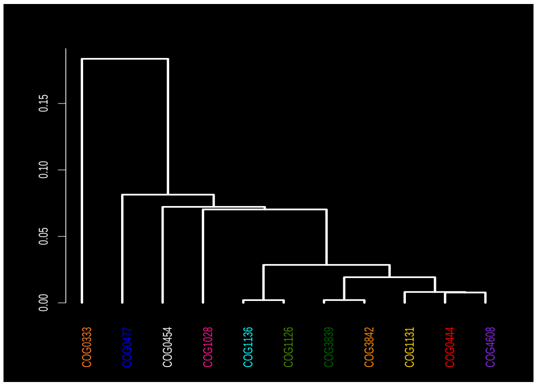

Figure 2: The dendrogram tree of the cluster centroids. The cluster labels are color-coded as in Figure 1.

Recall that in MDS, the goal is to create a low-dimensional representation of a high-dimensional space while preserving the similarity measures. Hence, given the choice of the similarity measure, the proximity of two points in the 3D representation in Figures 1 and 3 implies the similarity of the corresponding protein sequences as measured by the NW scores. High intensity values along the diagonal in Figure 1 (center) show a strong correlation between the NW distances and the distances based on MDS projections. The excess of points with mapped distances less than original values can be traced to equation (1) where the denominator depends on the original rather than mapped distances. Consequently, clusters that appear tight in 3D can be thought of as consisting of similar sequences, in NW sense. Similarly, scattered clusters imply greater variability of NW alignments between the proteins in the same cluster. Spatial proximity of clusters indicates the similarity of the sequences across these clusters. Note that the histogram of NW distances in Figure 1 also shows a lack of spatial separation between the clusters.

For the eleven color-coded COG clusters in Figure 1, we computed the centroids of their respective MDS projections. The dendrogram tree in Figure 2 shows the relative proximity of the cluster centroids to each other. Out of the eleven selected clusters, COG1131 (yellow) and COG1136 (cyan) are the tightest with respect to the mean intra-cluster distance. These two clusters are a part of a group that includes seven COGs in all; see right branch of the dendrogram. The other four COGs 1028, 0333, 0477, 0454 appear to be less similar to this group of seven or to each other.

The magnified view in Figure 3 (left) details the neighborhood structure of the COG1131 and COG1136 showing five more COGs lying in close proximity. Remarkably, all seven clusters are functionally similar and correspond to the ABC-type transport system, ATPase component (see Table 2). The heatmap shows a good agreement between the NW distances and MDS projections for the seven selected clusters; see Figure 3.

From the biological standpoint, the spatial features of the MDS projection of sequence alignment scores conform well to the clusters' functions. For example, a tight COG3839 cluster contains 142 protein sequences of the sugar transport systems that are similar both in function and composition. Similarly, COG1126 of the polar amino acid transport system proteins with very specific functions appears as a very tight cluster. In turn, the apparent diffusivity of COG1131 can be explained by the fact that the 244 multidrug transport system proteins that compose the cluster differ in amino acid composition and functional mechanisms. The inter-cluster distance of the 3D projections reflects the similarity between protein sequences in the corresponding clusters. For example, the two oligopeptide transport systems, COG4608 and COG0444, have similar shape and are located in close proximity to one another. The example of the COG data clearly demonstrates that MDS can effectively create a 3D projection of the PSU while preserving the fundamental grouping structure.

As mentioned, in our previous work we used all-versus-all alignment of 10 million UniRef100 proteins to populate the existing COG clusters [31]. The last column in Table 2 shows the number of UniRef100 proteins added to each of the eleven clusters from Figure 1. Notably the most diffuse clusters show the greatest expansion.

3.2 Comparison with BioLayout

We further compared the performance of the proposed method to BioLayout [13]. In BioLayout, the current limit for network rendering is 45,000 nodes and 5 million edges. This was only a quarter of the size of COG database. Hence, we decided to compute the layout only for data in seven selected clusters in Figure 3 that contain about 5.5 million edges. The projections resembled a large spherical cluster and did not reflect the underlying grouping structure (see Figure 4). Limited zooming capabilities did not allow exploring the results in more detail. The BioLayout approach does not have an interpolation option and hence the layout has to be recomputed every time the data set is expanded.

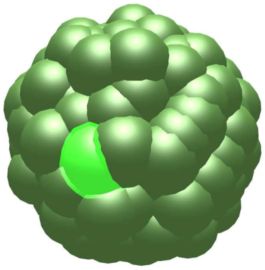

Figure 4: The 3D layout of data by BioLayout based on sequence similarity for seven clusters in Figure 3.

3.3 Archaeal Database

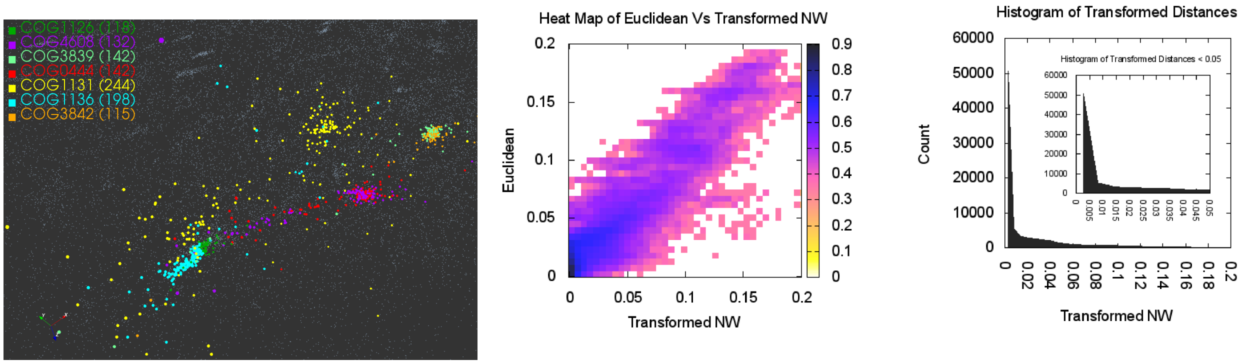

Figure 3: (left) Magnified version of the prokaryotic PSU showing seven functionally similar COGs from Figure 1; (center) the heatmap of the transformed NW distances versus the Euclidean distances between the MDS projections and (right) the histogram of transformed NW distances for the corersponding clusters. The inset in the top right corner shows the distribution for the distances below 0.05

Table 2: Annotations of COG clusters in Figures 1 and 3.

COG	Annotation	Size	UniRef
COG1131	ABC-type multidrug TS, ATPase comp.	244	14,406
COG1136	ABC-type antimicrobial peptide TS, ATPase comp.	198	7,306
COG1126	ABC-type polar amino acid TS, ATPase comp.	118	4,061
COG3839	ABC-type sugar TSs, ATPase comp.	142	4,121
COG0444	ABC-type di-/oligopeptide/nickel TS, ATPase comp.	142	3,520
COG4608	ABC-type oligopeptide TS, ATPase comp.	132	3,074
COG3842	ABC-type spermidine/putrescine TSs, ATPase comp.	115	3,665
COG0333	Ribosomal protein L32	49	1,148
COG0454	Histone acetyltransferase HPA2 & related acetyltransf.	285	14,085
COG0477	Permeases of the major facilitator superfamily	381	48,590
COG1028	Dehydrogenases with different specificities	299	37,461

Figure 5 shows an example of four COG clusters and the positions of the archaeal proteins classified into those clusters. The spread and shape of the projections is similar for bacterial and archaeal proteins. All four clusters have one common phenomenon: a tight core with extended, sparse scatter. The figure suggest that the proximity of the projections may be used to annotate new proteins by classifying them into existing clusters. However, the presence of outliers shows that projection information alone may not suffice for accurate classification at least for observation in the tails.

4. DISCUSSION

Functional protein annotation is one of the most important and resource-intensive challenges in biology [6]. The rapid influx of data from newly sequenced genomes together with high costs of expert annotaton create a major bottleneck, stalling scientific advances. The number of sequenced genomes is poised to increase in the next five years. The Earth Microbiome Project alone is expected to sequence 500,000 microbial genomes [10]. This is well over a 100-fold increase in the number of sequenced microbial genomes and proteins currently contained in GenBank. The i5K Insect and other Arthropod Genome Sequencing Initiative plans to sequence 5,000 insects and related species, yielding nearly 100 million new protein sequences [43]. Assigning functions to this glut of newly sequenced proteins is an immense scientific challenge.

Large-scale annotation projects require expert validaton, significant compute power, and a wide spectrum of analytic tools with scaling capabilities. Used here as an example, the COG database is one of the primary research tools in functional annotation and comparative genomics. However, rapid accumulation of data drastically raised the computational demands for COG update and enhancement. As a result, the database has not been updated since 2006. Sustaining resources like COG is essential to enable advances in functional annotation, comparative and evolutionary genomics.

In life sciences, efficient data exploration and analysis requires advanced visualization tools. However, existing methods neither address large-scale biological problems, nor offer sustainable, affordable means to cope with the influx of new information. Biological data are typically analyzed on the experiment level and in the context of known relationships, e.g. pathways, complexes. Tools for pathway and network visualization (e.g. Ingenuity or Biobase) consider neither sequence information nor extend to the entire PSU. Tools that would enable a low-dimensional representation of data and provide interactive visualization would substantially aid functional annotation.

The low-dimensional MDS implemented here allows dy-

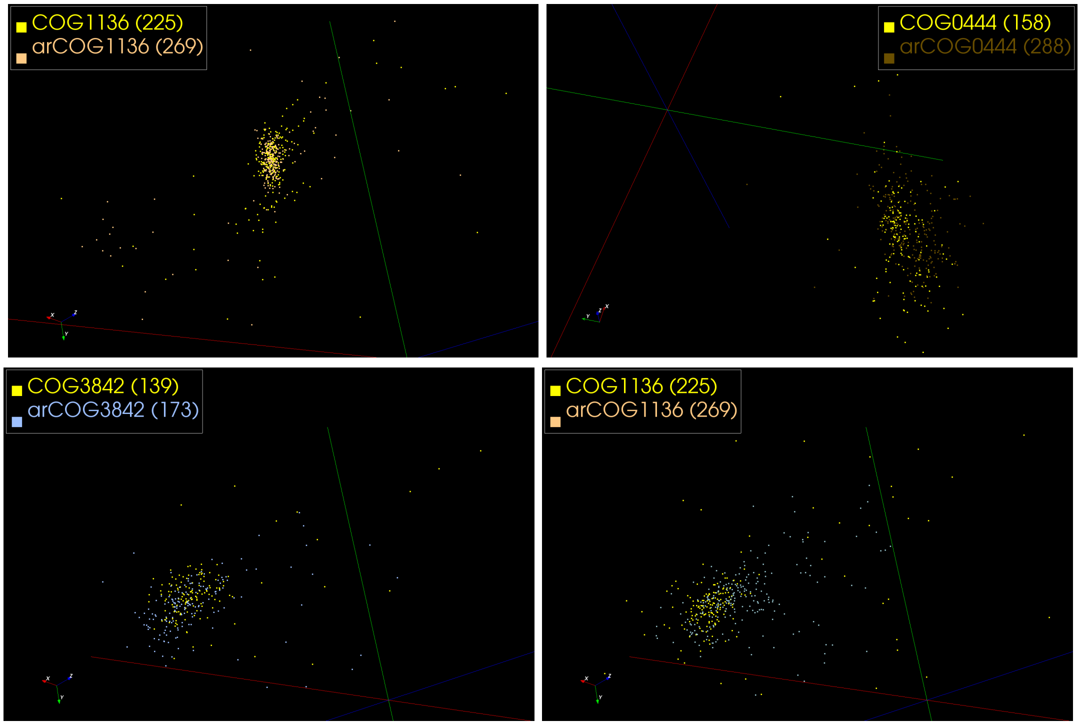

Figure 5: 3D view of selected COG clusters (yellow) and the archaeal proteins from those clusters. Orientation is shown in the bottom left corner. Big axis show the scale of the zoom. Numbers in parenthesis indicate the number of proteins in the given cluster.

namic, interactive exploration that is a mandatory precursor to statistical modeling. The MDS approach can be readily adapted to incorporate a composite similarity measure based on different types of proximities and biological information [2, 49, 20]. The parallel implementation employed here was developed specifically to handle large-scale data. Furthermore, the interpolation methods allow for quick mapping of new sequences into the existing projection space. The interpolation runs in $\mathcal{O}(n)$ time after an initial MDS embedding with the $\mathcal{O}(n^2)$ approach [4]. Given the ever increasing volumes of data from new sequencing technologies, this feature is essential as it facilitates prompt integration of large scale data while reducing computational costs. As a tool, PSU provides an interactive visualization of dependencies between a large number of proteins. The projection preserves the structure of data and can be integrated with information on function, pathways, structure, and environment, enabling analysis across domains of interest.

BioLayout platform provides an alternative visualization approach for biological data. Currently BioLayout has a 45,000 node rendering limit that is not enough even to visualize an example subset of well-characterized bacterial proteins. When applied to sequence data the selected seven clusters, BioLayout failed to preserve the distinct grouping structure. Furthemore, BioLayout has no interpolation mechanisms to iteratively update the results, so that an addition of a single sequence requires recomputing the layout of the entire set. In comparison, the MDS approach preserved the clustering structure and allowed for iterative expansion of the universe.

The mapping of archaeal proteins demonstrated the capabilities of the PSU as an annotation tool. The archaeal set was annonated by experts and hence provided a reliable standard. The interpolation allowed mapping a large number of archaeal proteins while effectively reducing computational complexity and memory requirement. The resulting projections were in good agreement with functional annotation of the corresponding proteins, i.e. the features and structure of archaeal proteins projections resembled those of the COG cluster they were classified into. Further, we intend to develop an accurate and efficient method for classifying new proteins into existing clusters based on the MDS layout. The two nearest neighbor rules, one based on the nearest annotated protein and the other based on the nearest consensus, did not achieve desirable accuracy attesting to the complexity of the problem.

In conclusion, we have illustrated the merits of low-dimensional embedding as a tool to explore the protein space and annotate new sequences. The method based on MDS retains the important grouping structure of the data, whereas the interpolation scheme allows for efficient expansion of the existing protein universe at reduced computational costs. The method outperformed the alternative graph layout approach implemented in BioLayout. The mapping of archaeal proteins illustrated both the advantages of the interpolation and the capacity of the proposed approach as an aid to functional annotation. The agreement between archaeal projections and the corresponding functional cluster suggested that an efficient classification scheme based on features of the projection space may enable an accurate functional annotation of new sequences.

Functional annotation of newly sequenced genomes cannot be solved by the life sciences community alone. The exascale of sequencing data requires a new, trans-disciplinary approach that would leverage and adopt the most prominent advances of modern sciences. This turn to collective innovation in data-enabled sciences is essential for truly groundbreaking medical discoveries. Scientific alliances like DELSA Global (Data-Enabled Life Sciences Alliance) stand to harness the essential diversity of skills and expertise, thus quickly and efficiently translating the influx of new data into tangible innovations and groundbreaking discoveries [39, 30]. Functional annotation represents one of the grand challenges in biology where communities like DELSA Global can help solve large-scale biological problems.

5. ACKNOWLEDGEMENTS

The authors thank the anonymous reviewers for their thoughful comments and feedback that helped improve the manuscript. We are greatful to Elizabeth Stewart and Christopher Moss for critical reading of the paper and insightful discussions. We also thank Courtney MacNealy-Koch, Maggie Lackey, Randy Salamon and Gregory Yandl. This work was made possible by the support from NSF (under DBI: 0969929 to E.K. and 0910818 to G.F.) and NIH (grant 5 RC2 HG 005806- 02 to G.F.; NIGMS grant R01 GM-076680-04 to E.K.; NIDDK grants U01-DK-089571 and U01-DK-072473 to E.K.).

6. ADDITIONAL AUTHORS

Additional authors: Adam Hughes (Pervasive Technology Institute, Indiana University), Yang Ruan (School of Informatics and Computing and Pervasive Technology Institute, Indiana University Bloomington, yangruan@indiana.edu); Judy Qiu (School of Informatics and Computing and Pervasive Technology Institute, Indiana University Bloomington; DELSA; xqiu@indiana.edu); Eugene Kolker (Bioinformatics & High-throughput Analysis Laboratory, SCRI; High-throughput Analysis Core, SCRI; Predicitive Analytics, Seattle Children's Hospital; Departments of Pediatrics and Biomedical Informatics & Medical Education, University of Washington; DELSA; eugene.kolker@seattlechildrens.org); and Geoffrey Fox (School of Informatics and Computing and Pervasive Technology Institute; DELSA; gcf@indiana.edu)

7. REFERENCES

[1] A. T. Adai, S. V. Date, S. Wieland, and E. M. Marcotte. LGL: creating a map of protein function with an algorithm for visualizing very large biological networks. *J. Mol. Biol.*, 340(1):179–190, Jun 2004.

[2] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215:403–410, Oct 1990.

[3] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, 25:3389–3402, Sep 1997.

[4] S.-H. Bae, J. Y. Choi, J. Qiu, and G. Fox. Dimension reduction and visualization of large high-dimensional data via interpolation. In Hariri and Keahey [21], pages 203–214.

[5] A. Bairoch, R. Apweiler, C. H. Wu, W. C. Barker, et al. The Universal Protein Resource (UniProt). *Nucleic Acids Res.*, 33:D154–159, Jan 2005.

[6] W. A. Baumgartner, K. B. Cohen, L. M. Fox, G. Acquaah-Mensah, and L. Hunter. Manual curation is not sufficient for annotation of genomic databases. *Bioinformatics*, 23:i41–48, Jul 2007.

[7] P. Bork. Powers and pitfalls in sequence analysis: the 70% hurdle. *Genome Res.*, 10:398–400, Apr 2000.

[8] J. Y. Choi, S.-H. Bae, X. Qiu, and G. Fox. High performance dimension reduction and visualization for large high-dimensional data analysis. In *CCGRID*, pages 331–340. IEEE, 2010.

[9] J. de Leeuw. Applications of convex analysis to multidimensional scaling. In J. Barra, F. Brodeau, G. Romier, and B. V. Cutsem, editors, *Recent Developments in Statistics*, pages 133–146. North Holland Publishing Company, Amsterdam, 1977.

[10] Earth Microbiome Project. http://www.earthmicrobiome.org/

[11] R. C. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32(5):1792–1797, 2004.

[12] J. Ekanayake, H. Li, B. Zhang, T. Gunarathne, S.-H. Bae, et al. Twister: a runtime for iterative mapreduce. In Hariri and Keahey [21], pages 810–818.

[13] A. J. Enright and C. A. Ouzounis. BioLayout–an automatic graph layout algorithm for similarity visualization. *Bioinformatics*, 17(9):853–854, Sep 2001.

[14] A. J. Enright, S. Van Dongen, and C. A. Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.*, 30(7):1575–1584, Apr 2002.

[15] G. Fox, S.-H. Bae, J. Ekanayake, X. Qiu, and H. Yuan. Parallel data mining from multicore to cloudy grids. In W. Gentzsch, L. Grandinetti, and G. R. Joubert, editors, *High Performance Computing Workshop*, volume 18 of *Advances in Parallel Computing*, pages 311–340. IOS Press, 2008.

[16] G. Fox, X. Qiu, S. Beason, J. Y. Choi, J. Ekanayake, et al. Biomedical case studies in data intensive computing. In M. G. Jaatun, G. Zhao, and C. Rong, editors, *CloudCom*, volume 5931 of *Lecture Notes in Computer Science*, pages 2–18. Springer, 2009.

[17] D. Frishman. Protein annotation at genomic scale: the current status. *Chem. Rev.*, 107:3448–3466, Aug 2007.

[18] M. Y. Galperin and E. Kolker. New metrics for comparative genomics. *Curr. Opin. Biotechnol.*, 17:440–447, Oct 2006.

[19] D. R. Gilbert, M. Schroeder, and J. van Helden. Interactive visualization and exploration of relationships between biological objects. *Trends Biotechnol*, 18(12):487–494, Dec. 2000.

[20] O. Gotoh. An improved algorithm for matching biological sequences. *J. Mol. Biol.*, 162:705–708, Dec 1982.

[21] S. Hariri and K. Keahey, editors. *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, HPDC 2010, Chicago, Illinois, USA, June 21-25, 2010.* ACM, 2010.

[22] T. Hastie, R. Tibshirani, and J. H. Friedman. *The*

Elements of Statistical Learning. Springer, corrected edition, July 2003.

[23] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[24] L. J. Jensen, P. Julien, M. Kuhn, C. von Mering, J. Muller, et al. eggNOG: automated construction and annotation of orthologous groups of genes. *Nucleic Acids Res.*, 36:D250–254, Jan 2008.

[25] A. J. Kearsley, R. A. Tapia, and M. W. Trosset. The solution of the metric STRESS and SSTRESS Problems in multidimensional scaling using Newton's method, 1995.

[26] W. Klimke, R. Agarwala, A. Badretdin, S. Chetvernin, S. Ciufo, et al. The National Center for Biotechnology Information's Protein Clusters Database. *Nucleic Acids Res.*, 37:D216–223, Jan 2009.

[27] H. Klock. Data visualization by multidimensional scaling: a deterministic annealing approach. *Pattern Recognition*, 33(4):651–669, 2000.

[28] E. Kolker, R. Higdon, W. Haynes, D. Welch, W. Broomall, et al. MOPED: Model Organism Protein Expression Database. *Nucleic Acids Res.*, 40:D1093–1099, Jan 2012.

[29] E. Kolker, K. S. Makarova, S. Shabalina, A. F. Picone, S. Purvine, et al. Identification and functional analysis of 'hypothetical' genes expressed in Haemophilus influenzae. *Nucleic Acids Res.*, 32:2353–2361, 2004.

[30] E. Kolker, E. Stewart, and V. Ozdemir. Opportunities and challenges for the life sciences community. *OMICS*, 16(3):138–147, Mar 2012.

[31] N. Kolker, R. Higdon, W. Broomall, L. Stanberry, D. Welch, et al. Classifying proteins into functional groups based on all-versus-all BLAST of 10 million proteins. *OMICS*, 15:513–521, 2011.

[32] E. Koonin and M. Galperin. *Sequence - evolution - function: computational approaches in comparative genomics*. Kluwer Academic, 2003.

[33] A. Krause, J. Stoye, and M. Vingron. The SYSTERS protein sequence cluster set. *Nucleic Acids Res.*, 28:270–272, Jan 2000.

[34] E. V. Kriventseva, W. Fleischmann, E. M. Zdobnov, and R. Apweiler. CluSTr: a database of clusters of SWISS-PROT+TrEMBL proteins. *Nucleic Acids Res.*, 29:33–36, Jan 2001.

[35] J. Leeuw. Convergence of the majorization method for multidimensional scaling. *Journal of Classification*, 5(2):163–180, September 1988.

[36] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Applied Mathmatics*, II(2):164–168, 1944.

[37] W. Li and A. Godzik. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22(13):1658–1659, Jul 2006.

[38] B. Louie, R. Higdon, and E. Kolker. A statistical model of protein sequence similarity and function similarity reveals overly-specific function predictions. *PLoS ONE*, 4:e7546, 2009.

[39] V. Ozdemir, D. Rosenblatt, L. Warnich, and S. e. a. Srivastava. Towards an ecology of collective innovation: Human Variome Project, Rare Disease Consortium for Autosomal Loci and Data-Enabled Life Sciences Alliance. *Current Pharmacogenomics and Personalized Medicine*, 9(4):1–9, 2011.

[40] E. Pennisi. Human genome 10th anniversary. Will computers crash genomics? *Science*, 331:666–668, Feb 2011.

[41] L. M. Proctor. The Human Microbiome Project in 2011 and beyond. *Cell Host Microbe*, 10:287–291, Oct 2011.

[42] J. Qiu and S.-H. Bae. Performance of Windows multicore systems on threading and MPI. *Concurrency and Computation: Practice and Experience*, 24(1):14–28, 2012.

[43] G. E. Robinson, K. J. Hackett, M. Purcell-Miramontes, S. J. Brown, J. D. Evans, et al. Creating a buzz about insect genomes. *Science*, 331:1386, Mar 2011.

[44] A. Roy, A. Kucukural, and Y. Zhang. I-TASSER: a unified platform for automated protein structure and function prediction. *Nat Protoc*, 5:725–738, 2010.

[45] SALSA group, Indiana University. PlotViz: a tool for visualizing large and high-dimensional data. `http://salsahpc.indiana.edu/plotviz/`

[46] J. W. Sammon. A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.*, 18:401–409, 1969.

[47] M. C. Schatz, B. Langmead, and S. L. Salzberg. Cloud computing and the DNA data race. *Nat. Biotechnol.*, 28:691–693, 2010.

[48] C. J. Sigrist, L. Cerutti, E. de Castro, P. S. Langendijk-Genevaux, V. Bulliard, et al. PROSITE, a protein domain database for functional characterization and annotation. *Nucleic Acids Res.*, 38:D161–166, Jan 2010.

[49] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147:195–197, Mar 1981.

[50] L. D. Stein. The case for cloud computing in genome informatics. *Genome Biol.*, 11:207, 2010.

[51] B. E. Suzek, H. Huang, P. McGarvey, R. Mazumder, and C. H. Wu. UniRef: comprehensive and non-redundant UniProt reference clusters. *Bioinformatics*, 23:1282–1288, May 2007.

[52] R. Tatusov, E. Koonin, and D. Lipman. A genomic perspective on protein families. *Science*, 278:631–637, 1997.

[53] R. L. Tatusov, N. D. Fedorova, J. D. Jackson, A. R. Jacobs, B. Kiryutin, E. V. et al. The COG database: an updated version includes eukaryotes. *BMC Bioinformatics*, 4:41, Sep 2003.

[54] J. D. Thompson, D. G. Higgins, and T. J. Gibson. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, 22:4673–4680, 1994.

[55] J. Vlasblom and S. Wodak. Markov clustering versus affinity propagation for the partitioning of protein interaction graphs. *BMC Bioinformatics*, 10(1):99, 2009.

[56] Y. I. Wolf, K. S. Makarova, N. Yutin, and E. V.

Koonin. Updated clusters of orthologous genes for Archaea: a complex ancestor of the Archaea and the byways of horizontal gene transfer. *Biol. Direct*, 7:46, 2012.

[57] G. Yona, N. Linial, and M. Linial. ProtoMap: automatic classification of protein sequences and hierarchy of protein families. *Nucleic Acids Res.*, 28:49–55, Jan 2000.