Journal of
Data Mining in Genomics & Proteomics

Foy and Wyckoff, J Data Mining Genomics Proteomics 2014, 5:1
DOI: 10.4172/2153-0602.1000152

Research Article Open Access

Units: Universal True SDSA (Structure-Dependent Sequence Alignment)

Scott Foy and Gerald Wyckoff *

Department of Molecular Biology and Biochemistry, University of Missouri-Kansas City, Kansas City, MO64110, USA

Abstract

Motivation: The Universal True SDSA (Structure-dependent Sequence Alignment), or UniTS, program calculates the most probable amino acid sequence alignment derived from multiple superimposed protein three-dimensional structures. Additionally, utilizing this newly generated SDSA, UniTS calculates improved quality assessment scores (e.g., RMSD, etc.) for the superimposed protein structures. Although other algorithms have been developed to derive an amino acid sequence alignment from aligned protein three-dimensional structures utilizing atomic proximity, none of these appropriately manages multiple residue matches, prevents the incorrect ordering of residues, and sequentially aligns structurally nonconserved regions. UniTS compensates for the weaknesses inherent in residue profile-based SDSA programs and structural alignment programs. Unlike the residue profile-based SDSA programs utilized as precursors to threading and homology modeling, UniTS is truly structure-dependent.

Results: The results presented herein demonstrate that UniTS calculates the universal sequence alignment for the complete protein compared to the partial sequence alignment derived from structural alignment programs. Furthermore, these results demonstrate the capability of UniTS to refine the sequence alignment input into a superpositioning program and utilize this refined alignment to calculate improved structural quality assessment scores. Finally, the quality score generation capabilities of UniTS allow it to compare numerous method of superimposing proteins utilizing consistently calculated scores.

Introduction

Although protein structure is a superior indicator of protein homology because it is general more evolutionarily conserved [1,2], utilization of protein sequences continues to be the primary method for determining protein homology due to sequence abundance, relatively inexpensive generation, and comparative algorithmic simplicity. However, as both computational power and the number of solved protein three-dimensional structures increase, the influence of structural information in protein biology is increasing. Evolutionary relationships and functional correlations between homologous proteins are increasingly determined utilizing protein structural alignment and super positioning software instead of an exclusive reliance on sequence alignment software. While protein structural alignment and super positioning software can calculate structural homology, the inability of this software to calculate an accurate consequent sequence alignment limits its utilization. Although some algorithms either directly (e.g., the Chimera Match=>Align function [3,4]) or indirectly (e.g., inverse folding and structural alignment) attempt to calculate protein sequence homology utilizing structural information, no algorithmic solution permits the derivation of an accurate alignment while utilizing the complete protein (including nonhomologous regions).

Current SDSA limitations

Inverse folding Structure-Dependent Sequence Alignment (SDSA) algorithms, precursors to protein structure prediction methods such as threading or homology modeling, attempt to align the amino acid sequence of one protein to the sequence of another with a known structure [5-7]. To align sequences possessing less than thirty percent sequence identity [8], these SDSA programs utilize profile-based sequencing techniques [5,6,9]. That is, they generate amino acid profiles based upon structural information to assist with the sequence alignment. Importantly, these profile-based SDSA algorithms continue to utilize conventional sequence alignment algorithms [9]. The amino acid profiles generated by structural information only supplement the conventional sequence alignment; they are not truly dependent on this structural information [5,10].

After superimposing protein structures, protein structural alignment algorithms must calculate a sequence alignment utilizing this structural alignment. Protein super-positioning algorithms require a preliminary sequence alignment to determine amino acid matching and function to superimpose the structures. Protein structure alignment algorithms do not require a preliminary amino acid sequence alignment and function to identify and align evolutionarily homologous regions of protein tertiary structures [11,12]. Although both types of algorithms ultimately superimpose protein structures, they require different starting inputs and utilize different methodologies. Unfortunately, the sequence alignments generated by structural alignment algorithms constitute only a fraction of the total protein. Consequent of utilizing a contact matrix, only those amino acids contained within matching submatrices are sequentially aligned [13,14]. Therefore, the homologous proteins are sequentially aligned intermittently rather than universally, resulting in an incomplete residue alignment. In addition to the decreased number of amino acid matches preventing the sequential alignment of structurally nonconserved regions, this also prevents the calculation of accurate structural alignment quality assessment scores (e.g., RMSD, etc.).

Neither inverse folding SDSAs nor structural alignment algorithms are designed to specifically calculate a sequence alignment from superimposed protein structures; however, the Match =>Align function of the University of California-San Francisco's Chimera protein structure visualization and modeling program is designed

***Corresponding author:** Gerald Wyckoff, Department of Molecular Biology and Biochemistry, University of Missouri-Kansas City, Kansas City, MO64110, USA, Tel: 8162352065; E-mail: wyckoffg@umkc.edu

Received December 23, 2013; **Accepted** February 23, 2014; **Published** February 26, 2014

Citation: Foy S, Wyckoff G (2014) Units: Universal True SDSA (Structure-Dependent Sequence Alignment). J Data Mining Genomics Proteomics 5: 152. doi:10.4172/2153-0602.1000152

for this purpose [4]. Unfortunately, the Match =>Align function is unsophisticated and possesses numerous algorithmic deficiencies. First, while matching amino acids from homologous chains, it maintains the residue order of the polypeptide chains (i.e., prevents the amino acids from becoming disordered) by serially matching them [3]. This heuristic serial matching method prevents the algorithm from calculating the best SDSA for the entirety of the proteins. Second, similar to many structural alignment programs [12,13], the Match =>Align function of Chimera is unable to directly match amino acids whose spatial distance exceeds a predetermined distance threshold. Instead, it utilizes arbitrary scores such as gap penalties and negative scores to match amino acids with a spatial distance greater than the threshold distance [3]. These arbitrary scores are inconsistent with the utilization of structure to calculate sequence matches and thus prevent the determination of an accurate sequential homology for structurally nonconserved regions.

The UniTS solution

The Universal True SDSA, or UniTS, program calculates the most probable sequence alignment derived from multiple superimposed protein structures. Although designed to neither resolve the inverse protein folding problem (as are residue profile-based SDSA algorithms) nor superimpose protein structures, UniTS compensates for the aforementioned limitations and deficiencies inherent in residue profile-based SDSA programs, structural alignment programs, and other spatial SDSA programs such as Chimera. If superimposed protein structures are available, UniTS is truly structure-dependent because it derives the SDSA utilizing spatial coordinates instead of residue profiles. Additionally, compared to the incomplete or partial sequence alignment generated by a structural alignment algorithm, UniTS calculates a universal amino acid sequence alignment constituting tertiary structure information from the entire protein.

Although the Match =>Align function of the Chimera program also derives a SDSA from superimposed protein structures utilizing atomic proximity [3], UniTS calculates the sequence homology of structurally nonconserved regions utilizing sequential information. Predicated on the evolutionary model, this method is biologically superior to the utilization of arbitrary scores. Furthermore, UniTS calculates residues matches comprehensively based upon the totality of the proteins instead of the heuristic serial matching performed by Match =>Align.

The consequent SDSA derived by UniTS permits the calculation of improved quality assessment scores (e.g., RMSD, etc.) for the superimposed proteins relative to those calculated exclusively by structural alignment and superpositioning algorithms. Unfortunately, as aforementioned, protein structure alignment algorithms derive a partial sequence alignment; additionally, superpositioning algorithms require an input sequence alignment that is derived utilizing a conventional sequence-based alignment program [15]. Therefore, neither the structure alignment nor superpositioning algorithms derive a sequence alignment utilizing structural information. Consequently, both algorithms utilize inadequate sequence alignments to calculate quality assessment scores for the superimposed protein structures. However, after these proteins have been superimposed, UniTS can modify and improve both the sequence alignment and the quality scores.

Methods

Pairwise SDSA

Given two structurally aligned proteins, the pairwise SDSA algorithm of the UniTS program will generate an amino acid sequence alignment based upon the tertiary protein structural alignment. The simplest and most intuitive SDSA algorithmic solution would calculate the distances between all opposing alpha carbons (i.e., alpha carbons located in different proteins). This algorithm would then consider two opposing amino acids to be a structural match if the distance between them is less than a predetermined distance threshold (four to five angstroms in many programs [3,4,12,13]). Unfortunately, this algorithmic solution is problematic despite being simple and intuitive.

The first problem with the aforementioned algorithm is the possibility of a single amino acid matching multiple opposing amino acids. If the multiple op-posing amino acids are adjacent to each other, any one of them can structurally match to the single amino acid. Furthermore, if the multiple amino acids are remote or nonadjacent, it is possible for the amino acids to match in an incorrect sequential order (Figure 1).

The second problem with the aforementioned algorithm regards the handling of singular omega loops. As detailed in Figure 2, exclusively utilizing spatial coordinates in structurally divergent regions possessing random insertion/deletion events prevents the determination of amino acid homology. Therefore, utilizing the aforementioned algorithm, determining the SDSA is impossible in protein regions possessing divergent structural alignments.

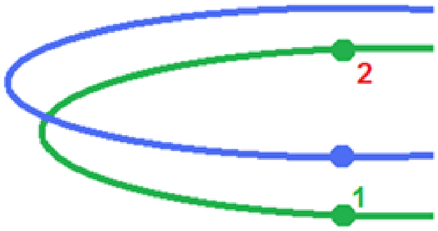

Figure 1: Disarranged amino acid matches illustrated utilizing the homologous loops of two protein chains (Green and Blue) with dots representing the alpha carbons of noteworthy amino acids. The Blue Amino Acid and the Green Amino Acid 1 are truly homologous and calculated to be a structural match. However, the Blue Amino Acid also structurally matches to Green Amino Acid 2 due to their close proximity. If the remaining amino acids in the loop are matched correctly, the Green Amino Acid 2 will be disarranged in the amino acid sequence.

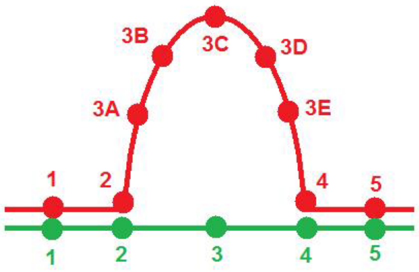

Figure 2: The omega loop problem illustrated utilizing two homologous protein chains with dots representing the alpha carbons of noteworthy amino acids. Amino Acids 1, 2, 4, and 5 for both proteins will be structurally matched. However, to which amino acid composing the omega loop (3A, 3B, 3C, 3D, or 3E) will Amino Acid 3 match? Assuming three of the four amino acids composing the omega loop were evolutionarily inserted (as opposed to the loop being deleted in the opposing protein), the original homologous amino acid can be any one of the loop amino acids. Conversely, utilizing only structural coordinates, the amino acid opposing the omega loop can be homologous to any of the amino acids composing the loop.

The pairwise SDSA algorithm of the UniTS program proposed herein determines pairwise structural matches similarly to the aforementioned algorithm. UniTS consider two opposing amino acids to be structurally matched if the distance between the alpha carbons of each amino acid is less than three ang-stroms. The algorithm uses the distance of three angstroms because it is approximately the maximum distance that prevents the frequent occurrence of an amino acid matching multiple opposing amino acids. Note, however, that multiple and disarranged matches can still emerge and their occurrence must be resolved. Therefore, following the calculation of structural matches, the pairwise SDSA algorithm utilizes a sorting algorithm to resolve multiple and disarranged matches (Figure 1).

The sorting algorithm orders a list of unordered amino acid positions by removing any positions that obstruct the correct order. Positions are removed based upon the distance from their ideal ordered index (Supplemental Material Section 1). Any position possessing multiple possible matches is input into the sorting algorithm as an element containing an "or" statement. However, the sorting algorithm continues to calculate an index distance for each possible match and removes them accordingly. Importantly, although the sorting algorithm can resolve multiple matches in which the opposing amino acids are not adjacent, it may be unable to resolve matches containing adjacent opposing amino acids. Therefore, if an amino acid position continues to match multiple opposing amino acids upon completion of the sorting algorithm, the algorithm will reject this position as a structural match. That is, UniTS is unable to structurally match this position based exclusively upon structural information. Instead, UniTS will resolve the match utilizing the same methodology it uses to align the remaining unmatched positions.

Although the sorting algorithm resolves matching multiple and disarranged residues, amino acids located in highly divergent regions of the structural alignment remain unmatched. A divergent region in a protein is an unmatched oligopeptide located between two structural matches and is composed of residues whose alpha carbons are greater than three angstroms from any opposing alpha carbon. As detailed in Figure 2, divergent regions of the structural alignment do not provide sufficient structural information to match homologous residues. Therefore, the UniTS program utilizes sequence information to align the amino acids of the divergent regions.

The pairwise SDSA algorithm utilizes the structurally matched amino acids to determine which divergent regions of each protein match. Divergent regions from opposing proteins that are located between the same structurally matched amino acids are matching divergent regions. The algorithm inputs the sequence from a divergent region of one protein and the sequence from the matching divergent region of the other protein into the MUSCLE sequence alignment program and sequentially aligns (i.e., utilize sequence information) the regions [16,17]. This process is repeated for all divergent regions of both proteins. The algorithm inserts gaps as necessary to compliment any unmatched divergent regions. Upon completion, all amino acids will be matched (either to another amino acid or to a gap) by either the structural matching algorithm or the MUSCLE sequence alignment.

The grid

Although deriving a pairwise alignment (conventional sequence, SDSA, or structure) is relatively straightforward, aligning multiple proteins introduces a fundamental difficulty in bioinformatics: How does one align multiple proteins at the same time? Many alignment programs (regardless of the specific type of information being aligned) solve this problem by subdividing the alignment into multiple pairwise-alignments. Inevitably, the program generates a multiple alignment by combining the results of these pairwise alignments [11,18].

Like many alignment programs, UniTS subdivides multiple alignments into numerous pairwise alignments. It subdivides and recombines pairwise alignments comparably to the Clustal algorithm [11]. Additionally, the way in which it dynamically modifies the results of each iteration in a data structure is analogous to the position specific scoring matrix (PSSM) in PSI-BLAST [18]. This PSSM equivalent data structure in the UniTS program, designated as the Grid, relates the amino acid positions of the overall multiple alignments (including gaps) to the amino acid positions of each input protein. Figure 3 is a visual representation of the Grid. UniTS designate the Grid to be an abstract protein (i.e., the Grid protein) and utilize it as the template protein to which the input proteins are aligned. The spatial coordinates of an alpha carbon at a certain position in the Grid protein are the mean of the alpha carbon spatial coordinates of any positions aligned to that Grid position. Note that because the aligned positions in the Grid are dynamic, UniTS continuously modifies the spatial coordinates of the Grid protein as it calculates new alignment iterations.

Residue determination

In the afore mentioned algorithm, the Grid protein is an abstract protein derived using the mean alpha carbon spatial coordinates of the input proteins. Deriving the mean of coordinates is possible because numbers are analog and capable of being averaged together. Conversely, divergent regions in the pairwise SDSA algorithm require the input of amino acid sequences into the MUSCLE program. Because the pairwise SDSA algorithm aligns the Grid protein to an input protein, the algorithm necessitates the sequence of the Grid protein. However, the digital nature of amino acid residues prevents the derivation of a mean sequence (e.g., how does one average a glycine and a phenylalanine?).

The pairwise SDSA algorithm designates the amino acid identity for a Grid position as the most frequently occurring residue for that respective Grid position. However, if the residues aligned to a Grid position occur with equal frequency, deriving the Grid residue requires a more complex solution. Before UniTS inputs a divergent region of the Grid into MUSCLE, any Grid position without an established residue identity (because no residue is the most frequently occurring) receives the designation of an unknown amino acid (i.e., assigned an IUPAC abbreviation of "X" [19]). For each Grid position featuring an unknown amino acid, UniTS substitutes the unknown amino acid with each of the possible amino acids available in the Grid position. MUSCLE then performs a sequence alignment for each of

Grid Position	1	2	3	4	5
Protein A Position	1	2	-	3	4
Protein B Position	-	1	-	2	3
Protein C Position	1	2	3	4	-
Protein D Position	1	-	2	3	-

Figure 3: Visual representation of the Grid. A gap is represented by a dash ("-"). The first amino acid in each of Proteins A, C, and D are aligned, while the first amino acid in Protein B aligns to the second amino acid in each of Proteins A and C. The spatial coordinates of the alpha carbon of the first Grid position are the mean of the coordinates of the first alpha carbon in each of Proteins A, C, and D.

these substitutions. Note that only one amino acid is substituted for each sequence alignment; the other positions retain the unknown designation. For each of these alignments, MUSCLE outputs a score file containing the average BLOSSOM62 score for each position aligned [11,20]. Therefore, each of the possible amino acids for each unknown Grid position receives a BLOSSOM62 score. The amino acid receiving the greatest BLOSSOM62 score for a given Grid position is selected to represent that Grid position. Importantly, the amino acid positional designations are utilized exclusively for the MUSCLE alignment; furthermore, because the Grid is dynamic, the designations change with each iteration of the multiple SDSA algorithms.

Multiple SDSA

Calculating the SDSA of multiple structures initiates by selecting one of the input proteins to be the initial template protein. The Supplemental Material (Section 2) describes the methodology UniTS employs to select the template protein. UniTS insert the selected template protein into the Grid. Because this is initially the only protein in the Grid, the alpha carbon spatial coordinates assumed by the Grid protein will equal those of the template protein (i.e., the mean of a single number is that number). The UniTS program then performs a pairwise SDSA of the Grid protein (initially only the template protein) and another input protein. Thereafter, UniTS will insert the input protein into the Grid based upon this pairwise alignment. Because the Grid now contains two proteins, the alpha carbon spatial coordinates of the Grid protein are recalculated by averaging the coordinates of both proteins. UniTS perform another pairwise SDSA of the newly calculated Grid protein and another input protein. This process is repeated until all input proteins have been inserted into the Grid (Figure 2).

Because the spatial coordinates of the Grid protein are updated as each input protein is iteratively inserted, the alignment of the initial template protein (or any of the early subsequent proteins) to the final Grid protein may now be inaccurate. Therefore, after the insertion of all input proteins into the Grid, the UniTS program will individually remove each protein (beginning with the original template protein) from the Grid. Upon removal of a protein, UniTS recalculates the spatial coordinates of the Grid protein utilizing those proteins remaining in the Grid. The removed protein will then be realigned to the recalculated Grid protein (via the pairwise SDSA algorithm) and reinserted into the Grid. This removal and realignment calculation is repeated for each protein. A single iteration is delineated by the removal and realignment of all the proteins. The afore mentioned iteration is repeated until the Grid stabilizes. That is, until all the amino acid positions of the input proteins remain in consistent Grid positions. Importantly, the UniTS program will not cease in the middle of iteration. Once the first protein is removed and realigned, the iteration must be completed by removing and realigning the remaining subsequent proteins. Only after the removal and realignment of the final protein will UniTS compare the current state of the Grid to its state at the conclusion of the previous iteration. If the positional state of the Grid in the current iteration equals that of the previous iteration, the iterations cease and UniTS achieves Grid position-al stabilization. Upon stabilization, the final state of the Grid is the final SDSA.

Multiple SDSA quality assessment: mean standard deviation

Traditionally, structural alignment and superpositioning algorithms utilize the RMSD score to quantitatively assess the quality of two superimposed proteins (i.e., a pairwise alignment). Unfortunately, superimposing more than two proteins (i.e., a multiple alignment) prevents the calculation of the RMSD for quantitative analysis. Therefore, UniTS performs the quantitative assessment of a multiple protein superposition or structural alignment utilizing the mean standard deviation. Calculation of the mean standard deviation consists of averaging the individual standard deviations of each Grid position. UniTS calculate each individual standard deviation utilizing the spatial coordinates of all the alpha carbons constituting each Grid position. Specifically, it calculates the mean and standard deviation for the coordinates of each axis separately. The three mean coordinates (one from each axis) combine to establish the three-dimensional spatial coordinates representing the mean. UniTS than derives the standard deviation coordinates by adding the calculated standard deviation distance for each axis to each respective mean coordinate. The positional standard deviation equals the spatial distance between the mean coordinates and the standard deviation coordinates.

Results

To determine the accuracy of the UniTS program, we utilized UniTS to calculate the SDSA results of four protein families. The two PDB files comprising each protein family are illustrated in Table 1 [21]. Because UniTS requires superimposed input proteins, we utilized the Theseus structural superpositioning program to superimpose the two proteins for each family [15,22,23] then compared the SDSAs, quality assessment scores, or generation parameters derived by UniTS to those results generated by the Theseus, DALI, and Chimera programs [4,22,24]. The subsequent results demonstrate that UniTS is currently the most capable and accurate algorithm for producing a SDSA if superimposed protein structures are available.

Because UniTS requires no input parameters (other than PDB files), we executed all comparison programs utilizing their default parameters. Additionally, although UniTS is capable of calculating a SDSA for multiple input protein structures (i.e., those involving more than two proteins), conducted comparisons utilize only pairwise alignments to reduce the complexity of manual analysis. Furthermore, all RMSD distances calculated herein incorporate only the alpha carbon atoms. Importantly, UniTS, Theseus, and DALI all perform distinctive primary functions. Therefore, UniTS does not replace these or other superpositioning and structural alignment programs; instead, UniTS supplements them by modifying their results. Finally, we performed no comparison of UniTS to a residue profile-based SDSA because UniTS requires superimposed protein structures. This requirement prevents the utilization of UniTS to solve the protein folding problem, thus a comparison is unwarranted.

UniTS compared to theseus

We performed the first UniTS comparison utilizing data output from the Theseus structural superpositioning program against the output data as subsequently refined by UniTS [22]. Importantly, Theseus does not generate a resultant sequence alignment; instead, Theseus requires the input of a preliminary sequence alignment. Therefore, the conventional sequence alignment program MUSCLE

Protein family	First protein PDB	Second protein PDB
Isocitrate Dehydrogenase	1T09[a]	1XGV[a]
Pectate Lyase	1PLU	2BSP
Polygalacturonase	1CZF[a]	1HG8
Hemopexin Repeats	1QHU[b]	1QHU[c]

[a]Chain A of the protein
[b]Residues 56-134.
[c]Residues 263-353

Table 1: PDB designations associated with each protein family.

Protein family	Theseus RMSD	UniTS RMSD
Isocitrate Dehydrogenase	15.23 Å	7.00 Å
Pectate Lyase	11.85 Å	6.19 Å
Polygalacturonase	1.99 Å	1.57 Å
Hemopexin Repeats	3.22 Å	1.74 Å

Table 2: Original RMSD reported by Theseus compared to the UniTS RMSD calculated from the proteins superimposed by Theseus for each protein family.

Protein family	Mean chain length[a]	DALI res matches[b]	UniTS res matches[b]	DALI RMSD
IDH[c]	422	299 (71%)	362 (86%)	2.57 Å
Pectate Lyase	375.5	261 (70%)	307 (82%)	1.59 Å
Polygalacturonase	342	325 (95%)	331 (97%)	1.13 Å
HPX Rep[d]	87.5	70 (80%)	70 (80%)	1.53 Å

[a] Mean number of amino acids comprising the two polypeptide chains representing each protein family.

[b] Number of amino acid residue matches. Parentheses contain the percentage of matched residues utilized out of the total (mean) number of amino acids. Due to length differences between proteins, it is not possible to reach 100% residue matches.

[c] Isocitrate Dehydrogenase.

[d] Hemopexin Repeats.

Table 3: Original RMSD reported by Theseus compared to the UniTS RMSD calculated from the proteins superimposed by Theseus for each protein family.

derived the input preliminary sequence alignment for all utilizations of Theseus presented herein [16,17]. Upon input of the MUSCLE sequence alignment, Theseus superimposed the input proteins and calculated the classical RMSD utilizing the amino acid matches established by the MUSCLE alignment. We then input the super positioned protein structures into UniTS to calculate a comparison RMSD and sequence alignment for each protein family.

The RMSD calculation is utilized to measure spatial similarity and requires the establishment of amino acid matching derived by various forms of homologous alignment. UniTS utilizes protein structure to derive a SDSA while Theseus utilizes a conventional sequence-based MUSCLE alignment; therefore, the amino acid matches established utilizing the SDSA of UniTS will more accurately represent the spatial homology of the two proteins. Table 2 illustrates a significantly decreased resultant RMSD calculated by the improved amino acid matches established by the UniTS SDSA (Table 2).

UniTS compared to DALI

We next compared UniTS to the DALI structural alignment program [24]. In addition to calculating the SDSA and RMSD for each protein family as detailed in the previous section, we also calculated the number of structurally matched residues UniTS utilized for these calculations. As displayed in Table 3, for three of the four protein families, UniTS utilized more residue matches than DALI when calculating the sequence alignment and RMSD (even the shorter hemopexin chains utilized the same percentage of matched residues for each methodology). The increased number of amino acid matches permits UniTS to produce a more complete and comprehensive SDSA and thus a more accurate RMSD calculation.

Notably, the RMSD returned from DALI is less than that calculated by UniTS. Although appearing favorable to DALI, this discrepancy is a product of the fewer matched residues DALI utilizes to calculate the RMSD. Specifically, the RMSD calculated by DALI is derived utilizing exclusively structurally conserved residues (i.e., those residue matches containing spatially proximate amino acids), thus resulting in a lower RMSD value [13] (Table 3).

UniTS compared to chimera

We performed the final comparison against the Match =>Align function of the Chimera protein structure visualization and modeling program [4]. Although relatively unsophisticated, the Match =>Align function is a SDSA algorithm similar to UniTS. After superimposing each protein family utilizing Theseus, we derived two SDSAs for each superimposed family utilizing UniTS and Chimera respectively (Supplementary Material Section 3).

To quantitatively determine which SDSA represents the more accurate evolutionary homology for each family, we calculated the log-odds score of each SDSA utilizing a similar methodology to that of a sequence alignment algorithm. Specifically, the log-odds score for each SDSA represents the significance of the similarity between the composing polypeptide sequences given the amino acids matching therein. That is, it represents the significance of a nonrandom homologous relationship existing, with a greater score indicating a more significant, nonrandom alignment [11]. We calculated the log-odds score utilizing the Gonnet substitution matrix to score individual amino acid matches [25]. The summation of these individual scores was then calculated to represent the sequential accuracy of the alignment.

We quantitatively compared the SDSAs for each protein family twice: The first comparison employed a gap opening penalty of 10.0 and a gap extension penalty of 0.1 because these constitute the standard default penalties in many sequence alignment programs [26,27]. However, the second comparison featured gap penalties of 5.0 and 0.1 respectively. The decreased gap opening penalty compensates for the increased number of gaps that will inevitably form when generating a SDSA as compared to a standard sequence alignment (Tables 4a and 4b).

Table 4a and 4b contain the log-odds scores derived by both UniTS and the Match =>Align function of Chimera for each protein family. The polypeptide sequences contained within the SDSAs generated by UniTS demonstrate superior significance of evolutionary homology relative to those generated utilizing the Match =>Align function of Chimera. The single exception to the aforementioned results is the log-odds score of the hemopexin repeats derived utilizing the 10.0 gap opening penalty. This inconsistent result can likely be attributed to the relatively short length of the repeats (Table 3 for a length comparison). The short length of the hemopexin repeats prevents an adequate number of amino acid matches that are required to counterweigh the large gap opening penalty. This explanation is reinforced by UniTS

Protein family	UniTS Alignment Score	Chimera Alignment Score
Isocitrate Dehydrogenase	-126.7	-437.9
Pectate Lyase	58.0	-108.8
Polygalacturonase	677.2	643.3
Hemopexin Repeats	14.6	16.0

Table 4a: Comparison of the UniTS and Chimera alignment scores utilizing a gap opening penalty of -10.

Protein family	UniTS Alignment Score	Chimera Alignment Score
Isocitrate Dehydrogenase	58.3	-192.9
Pectate Lyase	203.0	71.2
Polygalacturonase	742.2	713.3
Hemopexin Repeats	64.6	56.0

Table 4b: Comparison of the UniTS and Chimera alignment scores utilizing a gap opening penalty of -5.

producing the superior SDSA when utilizing the lesser 5.0 gap opening penalty (Figure 4).

Multiple PL/PG SDSA

To demonstrate the complete capability of UniTS, we calculated a multiple SDSA (Figure 4a) and quantitatively assessed the structural superpositioning of four homologous proteins (Figure 4b). The quad structural superpositioning was performed utilizing Theseus and consisted of two Pectate Lyase (PL) and two Polygalacturonase (PG) proteins (Table 1 for specific PDB designations). The mean standard deviation for the quad superposition is 3.36 angstroms.

In addition to calculating the total mean standard deviation for the entirety of the four protein structures super positioned, UniTS also exhibits the capability of outputting the standard deviation for each individual amino acid position (i.e., the Grid position as described in the Methods section 2.5). Furthermore, one can generate a graph correlating these individual standard deviations to their respective amino acid positions (the graph in Figure 4c demonstrates this capability utilizing the aforementioned quad superposition). This graph permits intelligible differentiation of those regions of the protein superposition that are structurally conserved from those that are nonconserved.

Discussion

Although only the Match =>Align function of the Chimera program directly calculates a sequence alignment utilizing spatial information from superimposed protein structures, other algorithms (e.g., inverse folding sequence alignments and structural alignments) are capable of performing this function indirectly. However, the aforementioned results indicate that UniTS is the most capable SDSA program to date. Furthermore, these results demonstrate the capability of UniTS to refine the sequence alignment input into a superpositioning program and utilize this refined alignment to calculate improved structural quality assessment scores. Importantly, because these quality assessment scores are consistently derived, they also provide the capability to compare different superpositioning and structural alignment algorithms [28].

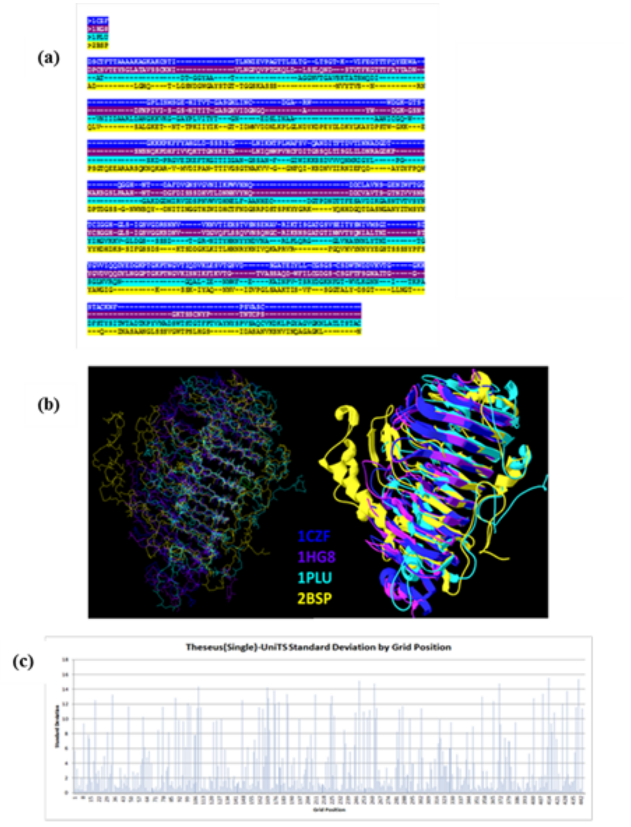

Figure 4: (a) PL/PG multiple SDSA derived utilizing UniTS. (b) Graphical representation of the PL/PG protein superposition in both atomic and ribbon formats. Protein super positioning was derived by Theseus and graphical imaging was performed utilizing Swiss-Pdb Viewer [27]. (c) Standard derivation for each Grid position of the PL/PG protein alignment.

Most significantly, implementation of the UniTS program requires a more formalized analysis of sequence alignments derived utilizing sequential information versus those derived utilizing structural information. That is, does amino acid sequence or protein structure primarily influence the evolutionary homology of proteins? Although the solution to this question is extraordinarily complex, the problem is reconcilable in many situations. However, consider the following question: Provided the results of a conventional sequence alignment and dissimilar results of an SDSA derived utilizing the same input proteins, which alignment most accurately represents the homology of the proteins? Unfortunately, this complex but reconcilable solution must now be simplistically reduced to two incompatible options. This inevitable problem substantiated by the UniTS program necessitates further research into protein sequential/structural correlation and robustness.

Acknowledgements

SDSA analysis was performed with the UCSF Chimera package. Chimera is developed by the Resource for Biocomputing, Visualization, and Informatics at the University of California, San Francisco (supported by NIGMS P41-GM103311).

Funding

Initial development work was funded in part by NIGMS R41 GM880633.

References

1. Kim C, Lee B (2007) Accuracy of structure-based sequence alignment of automatic methods. BMC Bioinformatics 8: 355.
2. Martí-Renom MA, Stuart AC, Fiser A, Sánchez R, Melo F, et al. (2000) Comparative protein structure modeling of genes and genomes. Annu Rev Biophys Biomol Struct 29: 291-325.
3. Meng EC, Pettersen EF, Couch GS, Huang CC, Ferrin TE (2006) Tools for integrated sequence-structure analysis with UCSF Chimera. BMC Bioinformatics 7: 339.
4. Pettersen EF, Goddard TD, Huang CC, Couch GS, Greenblatt DM, et al. (2004) UCSF Chimera--a visualization system for exploratory research and analysis. J Comput Chem 25: 1605-1612.
5. Bowie JU, Lüthy R, Eisenberg D (1991) A method to identify protein sequences that fold into a known three-dimensional structure. Science 253: 164-170.
6. Hong Y, Ko KD, Bharadwaj G, Zhang Z, van Rossum DB, et al. (2010) Towards solving the inverse protein folding problem.
7. Yang AS (2002) Structure-dependent sequence alignment for remotely related proteins. Bioinformatics 18: 1658-1665.
8. Rost B (1999) Twilight zone of protein sequence alignments. Protein Eng 12: 85-94.
9. Edgar RC, Sjölander K (2004) A comparison of scoring functions for protein sequence profile alignment. Bioinformatics 20: 1301-1308.
10. Kuziemko A, Honig B, Petrey D (2011) Using structure to explore the sequence alignment space of remote homologs. PLoS Comput Biol 7: e1002175.
11. Gibas C, Jambeck P (2001) Developing Bioinformatics Computer Skills. Yale J Biol Med 75: 117-118.

12. Ortiz AR, Strauss CE, Olmea O (2002) MAMMOTH (matching molecular models obtained from theory): an automated method for model comparison. Protein Sci 11: 2606-2621.

13. Holm L, Sander C (1993) Protein structure comparison by alignment of distance matrices. J Mol Biol 233: 123-138.

14. Konagurthu AS, Whisstock JC, Stuckey PJ, Lesk AM (2006) MUSTANG: a multiple structural alignment algorithm. Proteins 64: 559-574.

15. Theobald DL, Wuttke DS (2006) THESEUS: maximum likelihood superpositioning and analysis of macromolecular structures. Bioinformatics 22: 2171-2172.

16. Edgar RC (2004) MUSCLE: a multiple sequence alignment method with reduced time and space complexity. BMC Bioinformatics 5: 113.

17. Edgar RC (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res 32: 1792-1797.

18. Krane DE, Raymer ML (2003) Fundamental concepts of bioinformatics. Benjamin Cummings, San Francisco, USA.

19. Dixon HBF, Cornish-Bowden A, Liebecq C, Loening KL, Moss GP, et al. (1984) Nomenclature and symbolism for amino acids and peptides. European Journal of Biochemistry 138: 9-37.

20. Edgar RC (2010) MUSCLE user guide.

21. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, et al. (2000) The Protein Data Bank. Nucleic Acids Res 28: 235-242.

22. Theobald DL, Wuttke DS (2006) Empirical Bayes hierarchical models for regularizing maximum likelihood estimation in the matrix Gaussian Procrustes problem. Proc Natl Acad Sci U S A 103: 18521-18527.

23. Theobald DL, Wuttke DS (2008) Accurate structural correlations from maximum likelihood superpositions. PLoS Comput Biol 4: e43.

24. Holm L, Rosenstrom P (2010) Dali server: conservation mapping in 3D. Nucleic Acids Res 38: W545-549.

25. Gonnet GH, Cohen MA, Benner SA (1992) Exhaustive matching of the entire protein sequence database. Science 256: 1443-1445.

26. Tamura K, Peterson D, Peterson N, Stecher G, Nei M, et al. (2011) MEGA5: molecular evolutionary genetics analysis using maximum likelihood, evolutionary distance, and maximum parsimony methods. Mol Biol Evol 28: 2731-2739.

27. Guex N, Peitsch MC (1997) SWISS-MODEL and the Swiss-PdbViewer: an environment for comparative protein modeling. Electrophoresis 18: 2714-2723.

28. Isaev A (2006) Introduction to mathematical methods in bioinformatics. Springer Publications, Berlin.