On sufficient statistics of least-squares superposition of vector sets

Arun S. Konagurthu[1,*], Parthan Kasarapu[1], Lloyd Allison[1], James H. Collier[1], Arthur M. Lesk[2]

[1]Clayton School of Information Technology, Faculty of Information Technology, Monash University,
Clayton VIC 3800 Australia.

[2]The Huck Institute of Genomics, Proteomics and Bioinformatics and the Department of
Biochemistry and Molecular Biology, Pennsylvania State University, University Park PA 16802 USA

Corresponding Author: Arun S. Konagurthu

Address for correspondence: Clayton School of Computer Science and Information Technology,
Faculty of Information Technology, Monash University, Clayton VIC 3800, Australia.

Email: arun.konagurthu@monash.edu

Phone: +61-3-99053227

Published in Journal of Computational Biology, 20(6): 487-497 (2015)

Conference version of this paper appeared in the *Eighteenth Annual International Conference on Research in Computational Molecular Biology (RECOMB 2014)*, Apr 2-5 2014 Pittsburgh, USA. LNCS/LNBI 8394: 144-159

Abstract. The problem of superposition of two corresponding vector sets by minimizing their sum-of-squares error under orthogonal transformation is a fundamental task in many areas of science, notably structural molecular biology. This problem can be solved exactly using an algorithm whose time complexity grows linearly with the number of correspondences. This efficient solution has facilitated the widespread use of the superposition task, particularly in studies involving macromolecular structures. This paper formally derives a set of *sufficient statistics* for the least-squares superposition problem. These statistics are additive. This permits a highly efficient (*constant time*) computation of superpositions (and sufficient statistics) of vector sets that are composed from its constituent vector sets under addition or deletion operation, where the sufficient statistics of the constituent sets are already known (that is, the constituent vector sets have been previously superposed). This results in a drastic improvement in the run time of the methods that commonly superpose vector sets under addition or deletion operations, where previously these operations were carried out *ab initio* (ignoring the sufficient statistics). We experimentally demonstrate the improvement our work offers in the context of protein structural alignment programs that assemble a reliable structural alignment from well-fitting (substructural) fragment pairs. A `C++` library is available under an open-source license from `http://www.csse.monash.edu.au/~karun/superpose3D`.

1 Introduction

Optimal superposition of two corresponding vector sets is a commonly used method to measure spatial similarity of three-dimensional (3D) objects. In this method, treating both the vector sets as rigid bodies, one set is *rotated* and *translated* to fit on another. Such superposition of vector sets permits the evaluation of shape similarity both qualitatively and visually. A nearly universal *criterion of optimality* for the superposition problem is the one that *minimizes the sum of squared deviations* between the corresponding vectors after superposition (or least-squares superposition, in short). This yields a reliable quantitative measure of similarity, the *root mean square deviation* (or r.m.s.d.) between vector sets (Eidhammer *et al.*, 2004; Lesk, 2000, 2001).

Many approaches to address the least-squares superposition problem in three dimensions (3D) have been proposed over the years (Cohen, 1997; Coutsias *et al.*, 2004; Cox, 1967; Diamond, 1988; Kabsch, 1976, 1978; Kearsley, 1989; KenKnight, 1984; Koehl, 2001; Lesk, 1986; Mackay, 1984; McLachlan, 1972, 1982). It is indeed remarkable that this problem can be solved exactly and efficiently, and involves a computational effort that grows *linearly* with the number of correspondences in the vector sets. Noteworthy of the approaches to solve the superposition problem is the method by Kabsch (1976) that allows computing the optimal transformation via singular value decomposition of a covariance matrix derived from the coordinates of the corresponding vector sets. An equivalent, and more analytically elegant, approach for this problem proposed by Kearsley (1989) uses the algebra of *quaternions* (Hamilton and Hamilton, 1866). Quaternions are generalizations of complex numbers with direct applications to orthogonal transformations in three dimensional (3D) space. Specifically, the space group corresponding to unit quaternions is equivalent to the group of all possible proper 3D rotations defined about an arbitrary origin. Any pure rotation in 3D by an angle θ about some normalized axis $\hat{\boldsymbol{n}}$ passing through the origin can be represented using a unit quaternion denoted by $\left[\cos\left(\frac{\theta}{2}\right), \hat{\boldsymbol{n}}\sin\left(\frac{\theta}{2}\right)\right]$. Among the key advantages of using Kearsley's method to solve the least-squares superposition problem are:

1. the problem can be solved analytically as an eigenvalue and eigenvector problem in quaternion parameters, and
2. the method avoids problems with singularities (and rotary inversions) that can result from Kabsch's method, where these oddities are handled explicitly after the solution is found (Coutsias *et al.*, 2004; Kearsley, 1989)

Structural molecular biology employs least-squares superposition to support a wide variety of tasks. An example is the role of superposition in programs for aligning protein 3D structures (Kolodny *et al.*, 2005; Konagurthu *et al.*, 2006; Lackner *et al.*, 2000; Lesk, 1986; Shatsky *et al.*, 2002, 2004; Shindyalov and Bourne, 1998; Vriend and Sander, 1991; Ye and Godzik, 2003). A structural alignment is the assignment of amino acid residue-residue correspondences between proteins based on the similarity of their structural contexts (Konagurthu *et al.*, 2006). Among the commonly used heuristics to align protein structures are the ones that rely on identifying a library of well-fitting fragments (or contiguous substructures) within the protein structures being aligned. This library is then refined by *jointly superposing* the fragment pairs and determining the pairs that fit consistently, from which a structural alignment is finally *assembled*.

The joint superpositions in the current structural aligners are computed from scratch, even though previous superpositions of constituent fragment pairs provide a lot of information about the joint superposition. We can see that the *number* of joint superpositions grows quadratically in the size of the well-fitting fragment-pairs library, where each joint superposition takes a computational effort that is linear in the size of the combined vector sets. On average, when aligning a pair of homologous protein structures, there are well in excess of thousand well-fitting fragment pairs, exhaustively requiring millions of joint superpositions before a structural alignment can be assembled. This exhaustive step poses a significant computational bottleneck for structural aligners, and, therefore, this problem is commonly mitigated by invoking joint superpositions rather restrictively, trading off potential structural alignment quality for speed.

In this paper we explore the foundations of the orthogonal superposition problem and derive a set of statistics that are sufficient to compute the r.m.s.d of the best superposition, and its corresponding rotation and translation parameters. We demonstrate that these *sufficient statistics* (Hogg and Craig, 1994) are additive. Therefore, these statistics can be used to compute new superpositions in *constant time* using the sufficient statistics of superpositions of constituent vector sets under addition and deletion operations. Using sufficient statistics results in a drastic speed up of tasks like joint superposition described above, compared to the current approaches that recompute these superpositions from scratch.

Organization: Section 2 gives the background to the orthogonal superposition problem using the least-squares criterion. Section 3 introduces the notion of sufficient statistics, and derives the full set of sufficient statistics for the orthogonal superposition problem. Section 4 provides the update rules to update the superposition parameters in constant-time, building on the sufficient statistics of constituent superpositions. Section 5 describes an approach to speed up the diagonalization step used in the Kearsley approach. Finally, the paper ends with an experimental evaluation of performance gain achieved using the results of this work.

2 Orthogonal superposition

Formally let $\mathcal{U} = \{\boldsymbol{u}_1, \cdots, \boldsymbol{u}_n\}$ and $\mathcal{V} = \{\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n\}$ denote two vector sets with one-to-one correspondence. In this paper we consider vectors in three dimensions. Let the (x, y, z) components of each $\boldsymbol{u}_i$ be represented here as $(\boldsymbol{u}_i(x), \boldsymbol{u}_i(y), \boldsymbol{u}_i(z))$. (Similar representation holds for $\boldsymbol{v}_i$ or any other vector.)

The least-squares superposition problem is an optimization problem that involves finding the orthogonal rotation $\mathbf{R}$ and translation $\boldsymbol{t}$ with the optimality criterion defined as:

$$\mathcal{E} = \min \|\mathbf{R}\mathcal{U} + \boldsymbol{t} - \mathcal{V}\|^2 = \min \sum_{i=1}^{n} \|\mathbf{R}\boldsymbol{u}_i + \boldsymbol{t} - \boldsymbol{v}_i\|^2 = \min \sum_{i=1}^{n} \langle \mathbf{R}\boldsymbol{u}_i + \boldsymbol{t} - \boldsymbol{v}_i, \mathbf{R}\boldsymbol{u}_i + \boldsymbol{t} - \boldsymbol{v}_i \rangle$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product between the stated terms, $\mathbf{R}$ is a 3×3 pure rotation matrix, and $\boldsymbol{t}$ is a translation vector.

Under this least-squares criterion, the optimal translation can be made independent of the optimal rotation as follows. Differentiating $\mathcal{E}$ with respect to $\boldsymbol{t}$ and evaluating it at its extremum

$$\begin{aligned} \frac{\partial \mathcal{E}}{\partial \boldsymbol{t}} &= \frac{\partial}{\partial \boldsymbol{t}} \sum_{i=1}^{n} \langle \mathbf{R}\boldsymbol{u}_i + \boldsymbol{t} - \boldsymbol{v}_i, \mathbf{R}\boldsymbol{u}_i + \boldsymbol{t} - \boldsymbol{v}_i \rangle = 2 \sum_{i=1}^{n} \frac{\partial (\mathbf{R}\boldsymbol{u}_i + \boldsymbol{t} - \boldsymbol{v}_i)}{\partial \boldsymbol{t}} (\mathbf{R}\boldsymbol{u}_i + \boldsymbol{t} - \boldsymbol{v}_i) = 0 \\ &\implies \sum_{i=1}^{n} \mathbf{R}\boldsymbol{u}_i + \boldsymbol{t} - \boldsymbol{v}_i = 0 \implies \boldsymbol{t} = \frac{\sum_{i=1}^{n} \boldsymbol{v}_i}{n} - \mathbf{R}\frac{\sum_{i=1}^{n} \boldsymbol{u}_i}{n} = \mathbf{Centroid}(\mathcal{V}) - \mathbf{R}\ \mathbf{Centroid}(\mathcal{U}) \end{aligned}$$

It follows that moving each of the vector sets to an origin at its centroid, about which the rotation is defined, gives us a modified (but equivalent) objective which is independent of the translation $\boldsymbol{t}$: $\mathcal{E} = \min \sum_{i=1}^{n} |\mathbf{R}\boldsymbol{u}_i' - \boldsymbol{v}_i'|^2$, where $\boldsymbol{u_i}' = \boldsymbol{u_i} - \mathbf{Centroid}(\mathcal{U})$ and $\boldsymbol{v_i}' = \boldsymbol{v}_i - \mathbf{Centroid}(\mathcal{V})$.

Kearsley (1989) proposed an elegant method that removes the non-linear aspect to this least-squares problem and transforms it into an eigenvalue problem of the form $\mathbf{Q}\boldsymbol{q} = \lambda \boldsymbol{q}$, or in expanded terms

$$\begin{pmatrix} \sum(x_m^2 + y_m^2 + z_m^2) & \sum(y_p z_m - y_m z_p) & \sum(x_m z_p - x_p z_m) & \sum(x_p y_m - x_m y_p) \\ \sum(y_p z_m - y_m z_p) & \sum(x_m^2 + y_p^2 + z_p^2) & \sum(x_m y_m - x_p y_p) & \sum(x_m z_m - x_p z_p) \\ \sum(x_m z_p - x_p z_m) & \sum(x_m y_m - x_p y_p) & \sum(x_p^2 + y_m^2 + z_p^2) & \sum(y_m z_m - y_p z_p) \\ \sum(x_p y_m - x_m y_p) & \sum(x_m z_m - x_p z_p) & \sum(y_m z_m - y_p z_p) & \sum(x_p^2 + y_p^2 + z_m^2) \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} = \lambda \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} \quad (1)$$

As seen above, $\mathbf{Q}$ is a 4×4 square symmetric matrix and $\boldsymbol{q} = (q_1, q_2, q_3, q_4)^T$ are the unknown (to be solved) quaternion components associated with a 3D rotation, and λ is an (unknown) eigenvalue. In the eigenvalue problem defined in Equation 1, the notation x_m, a scalar quantity, denotes the component-wise difference $\boldsymbol{v}_i'(x) - \boldsymbol{u}_i'(x)$ (equivalent notations for y_m and z_m) and the scalar x_p denotes the component-wise sum $\boldsymbol{v}_i'(x) + \boldsymbol{u}_i'(x)$ (equivalently, y_p and z_p). From this point onwards, we use the term *quaternion matrix* to refer to the 4×4 matrix $\mathbf{Q}$ shown in Equation 1.

Diagonalizing $\mathbf{Q}$ yields four eigenvalues and (corresponding) eigenvectors. Kearsley (1989) shows that the eigenvector corresponding to the smallest eigenvalue, $\lambda_{\min}$, corresponds to the best rotation producing the least squares error, and the r.m.s.d is computed as $\sqrt{\dfrac{\lambda_{\min}}{n}}$

Time complexity The computational effort required to solve the rigid-body superposition problem using Kearsley's quaternion approach (or equivalently Kabsch's approach) grows linearly with the number of vectors being superimposed. In Kearsley's approach this is dominated by the computation of the quaternion matrix $\mathbf{Q}$ where each of 10 distinct terms in the matrix requires $O(n)$ effort. The diagonalization of $\mathbf{Q}$ is independent of n and shows a rapid convergence with numerical methods such as Jacobi's diagonalization algorithm (Jacobi, 1846).

3 Sufficient Statistics

We note that this rigid-body superposition problem is a geometric instance of the general regression problem using total least-squares, where a regression line has to be determined that minimizes the sum-of-squares error with respect to the observed data points.

The error terms of this regression problem are assumed to be normally distributed as $\mathcal{N}(0, \sigma)$, where the mean μ is 0 and σ is the standard deviation which is minimized by the problem. In fact, the least squares estimator of σ is also its maximum likelihood estimator.

More formally, consider the standard normal distribution of some random variable x:

$$\mathcal{N}(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

This normal density can be reparameterized into a general form denoting the family of exponential distributions: $f(x|\boldsymbol{\eta}) = h(x)g(\boldsymbol{\eta})\exp(\boldsymbol{\eta}^T\boldsymbol{U}(x))$ where $h(x) = \frac{1}{\sqrt{\pi}}$, $g(\eta_2) = \sqrt{-\eta_2}\exp\left(\frac{\eta_1^2}{4\eta_2}\right)$, $\boldsymbol{\eta}^T = (\frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2})$, $\boldsymbol{U}^T(x) = (x, x^2)$.

This transformation can be used to show certain important properties that allows efficient computation of maximum likelihood estimators of μ and σ.

Considering a sample set of observations that are normally distributed $\mathbf{X} = \{x_1, x_2, \cdots, x_n\}$. The likelihood for these samples is given by:

$$f(\mathbf{X}|\boldsymbol{\eta}) = \left(\prod_{i=1}^{n} h(x_i)\right)(g(\boldsymbol{\eta}))^n \exp(\boldsymbol{\eta}^T\sum_{i=1}^{n} u(x_i))$$

Taking natural logarithms on both sides gives us the log likelihood:

$$\log(f(\mathbf{X}|\boldsymbol{\eta})) = \kappa + n\log\left(g(\boldsymbol{\eta})\right) + \boldsymbol{\eta}^T\sum_{i=1}^{n}\boldsymbol{U}(x_i)$$

where $\kappa = \sum_{i=1}^{n}\log(h(x_i))$ is a term independent of $\boldsymbol{\eta}$.

To find the maximum likelihood estimators $\hat{\boldsymbol{\eta}}$, take the gradient with respect to $\boldsymbol{\eta}$ and set to 0. This results in:

$$n\nabla_{\hat{\boldsymbol{\eta}}}\left[\log\left(g(\hat{\boldsymbol{\eta}})\right)\right] + \sum_{i=1}^{n}\boldsymbol{U}(x_i) = 0 \implies -\nabla_{\hat{\boldsymbol{\eta}}}\left[\log\left(g(\hat{\boldsymbol{\eta}})\right)\right] = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{U}(x_i) = \frac{-1}{g(\hat{\boldsymbol{\eta}})}\nabla_{\hat{\boldsymbol{\eta}}}g(\hat{\boldsymbol{\eta}}) = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{U}(x_i)$$

Notice that maximum likelihood estimate $\hat{\boldsymbol{\eta}}$ depends on the statistic $\sum_{i=1}^{n} \boldsymbol{U}(x_i)$ rather than the individual data. This suggests that to obtain the maximum likelihood estimate we do not need the data explicitly as it can be derived from that statistic. This sufficiency to derive the maximum likelihood estimator without explicit consideration of data makes $\sum_{i=1}^{n} \boldsymbol{U}(x_i)$ a *sufficient statistic* for the exponential family of functions. For normal distribution, we saw earlier that $\boldsymbol{U}(x_i) = (x_i, x_i^2)$ gives the sufficient statistics of $\sum_{i=1}^{n} x_i$ and $\sum_{i=1}^{n} x_i^2$ (Hogg and Craig, 1994).

Sufficient statistics for orthogonal superposition

For the orthogonal superposition problem, each (x, y, z) component of the error vector, $\varepsilon_i = \mathbf{R}\boldsymbol{u_i}' - \boldsymbol{v_i}'$, is also assumed to be normally distributed. That is, $\varepsilon_i(x) \sim \mathcal{N}(0, \sigma)$, $\varepsilon_i(y) \sim \mathcal{N}(0, \sigma)$, and $\varepsilon_i(z) \sim \mathcal{N}(0, \sigma)$. We now derive the sufficient statistics for σ of $\{\varepsilon_i(x), \varepsilon_i(y), , \varepsilon_i(z)\}$ terms.[1] The likelihood of the observed normally distributed errors after superposition, $\mathbf{E} = \{\varepsilon_1(x), \varepsilon_1(y), \varepsilon_1(z) \ldots, \varepsilon_n(x), \varepsilon_n(y), \varepsilon_n(z), \}$, can be written as:

$$f(\mathbf{E}|\sigma) = \prod_{i=1}^{3n} (2\pi\sigma^2)^{-\frac{1}{2}} \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{R}\boldsymbol{u_i}' - \boldsymbol{v_i}'\|^2\right) = (2\pi\sigma^2)^{-\frac{3n}{2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\|\mathbf{R}\boldsymbol{u_i}' - \boldsymbol{v_i}'\|^2\right)$$

Let's examine the decomposition of any $\|\varepsilon_i\|$ term:

$$\|\varepsilon_i\|^2 = \|\mathbf{R}\boldsymbol{u_i}' - \boldsymbol{v_i}'\|^2 = \|\boldsymbol{u_i}'\|^2 + \|\boldsymbol{v_i}'\|^2 - 2{\boldsymbol{v_i}'}^{\mathrm{T}}\mathbf{R}\boldsymbol{u_i}'. \tag{2}$$

From Equation 1, the matrix $\mathbf{Q}$ is made up of terms of the form $A_m = v_i'(A) - u_i'(A)$ and $A_p = v_i'(A) + u_i'(A)$ where each A and B take the values x, y, or z denoting the vector components. Rewriting, we get $v_i'(A) = \frac{A_p + A_m}{2}$ and $u_i'(A) = \frac{A_p - A_m}{2}$. The first two terms on the right hand side of Equation 2 can be expanded as follows:

$$\begin{aligned}\|\boldsymbol{u_i}'\|^2 + \|\boldsymbol{v_i}'\|^2 &= (u_i'(x)^2 + u_i'(y)^2 + u_i'(z)^2) + (v_i'(x)^2 + v_i'(y)^2 + v_i'(z)^2) \\ &= \frac{1}{2}(x_m^2 + x_p^2 + y_m^2 + y_p^2 + z_m^2 + z_p^2) = \frac{1}{2}\sum_{A\in\{x,y,z\}} A_m^2 + \frac{1}{2}\sum_{A\in\{x,y,z\}} A_p^2\end{aligned} \tag{3}$$

The last term on the right hand side of Equation 2 can be expanded as ${\boldsymbol{v_i}'}^{\mathrm{T}}\mathbf{R}\boldsymbol{u_i}' = {\boldsymbol{v_i}'}^{\mathrm{T}}[\mathbf{r_1}\ \mathbf{r_2}\ \mathbf{r_3}]\boldsymbol{u_i}'$ where $\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}$ are column vectors of the 3×3 rotation matrix $\mathbf{R}$. Therefore,

$${\boldsymbol{v_i}'}^{\mathrm{T}}\mathbf{R}\boldsymbol{u_i}' = (\boldsymbol{v_i}'.\mathbf{r_1})u_i'(x) + (\boldsymbol{v_i}'.\mathbf{r_2})u_i'(y) + (\boldsymbol{v_i}'.\mathbf{r_3})u_i'(z) \tag{4}$$

[1] Note that, $\sqrt{3}\sigma$ is equal to the r.m.s.d. after least-squares superposition.

Take the first term on the right hand side of Equation 4. This can be expanded as:

$$\begin{aligned}(\boldsymbol{v_i}'.\mathbf{r_1})u_i'(x) &= r_{11}v_i'(x)u_i'(x) + r_{12}v_i'(y)u_i'(x) + r_{13}v_i'(z)u_i'(x)\\ &= \frac{r_{11}}{4}(x_p + x_m)(x_p - x_m) + \frac{r_{12}}{4}(y_p + y_m)(x_p - x_m) + \frac{r_{13}}{4}(z_p + z_m)(x_p - x_m)\\ &= \frac{r_{11}}{4}(x_p^2 - x_m^2) + \frac{r_{12}}{4}(y_p x_p - y_p x_m + y_m x_p - y_m x_m)\\ &\quad + \frac{r_{13}}{4}(z_p x_p - z_p x_m + z_m x_p - z_m x_m)\end{aligned}$$

where r_{11}, r_{12}, r_{13} are the terms in the $\boldsymbol{r}_1$ column vector in $\mathbf{R}$. More generally,

$$(\boldsymbol{v_i}'.\mathbf{r_1})u_i'(x) = c_1 A_p^2 + c_2 A_m^2 + c_3 A_p B_p + c_4 A_m B_m + c_5 A_m B_p \tag{5}$$

where c_k are constants in terms of components of $\boldsymbol{r}_1$.

Similarly, $(\boldsymbol{v_i}'.\mathbf{r_2})u_i'(y)$ and $(\boldsymbol{v_i}'.\mathbf{r_3})u_i'(z)$ can be expanded as above and will have the same form as Equation 5, but with different constants. Therefore, combining Equations 3-4, Equation 2 can expressed as

$$\|\varepsilon_i\|^2 = \zeta_1 \sum_A A_p^2 + \zeta_2 \sum_A A_m^2 + \zeta_3 \sum_{\forall A \neq B} A_p B_p + \zeta_4 \sum_{\forall A \neq B} A_m B_m + \zeta_5 \sum_{\forall A \neq B} A_m B_p$$

where ζ_k are constants. Hence, the likelihood function can be written as

$$f(\mathbf{E}|\sigma) = (2\pi\sigma^2)^{-\frac{3n}{2}} \exp\left(-\frac{1}{2\sigma^2}\mathbf{U}\right) \tag{6}$$

where $\mathbf{U} = \sum_{i=1}^{n} \left(\zeta_1 \sum_A A_p^2 + \zeta_2 \sum_A A_m^2 + \zeta_3 \sum_{\forall A \neq B} A_p B_p + \zeta_4 \sum_{\forall A \neq B} A_m B_m + \zeta_5 \sum_{\forall A \neq B} A_m B_p\right)$ and $A, B \in \{x, y, z\}$.

Using Equation 6, the negative log-likelihood is given as:

$$\mathcal{L}(\mathbf{E}|\sigma) = \frac{3n}{2}\log(2\pi) + 3n\log\sigma + \frac{1}{2\sigma^2}\mathbf{U} \tag{7}$$

The maximum likelihood estimate $\hat{\sigma}$ can be determined by minimising Equation 7 and evaluating the corresponding σ, *i.e.* $\frac{\partial \mathcal{L}}{\partial \sigma} = 0 \implies \hat{\sigma}^2 = \frac{\mathbf{U}}{3n}$.

$\mathbf{U}$ involves statistics that do not take into account the data (the coordinates of vector sets) explicitly, and are sufficient to estimate σ (and through it, r.m.s.d). Therefore the set of *sufficient statistics* for the least-squares superposition problem can be defined as:

$$\mathbf{\Psi} = \left\{\sum_{i=1}^{n} A_m B_m, \quad \sum_{i=1}^{n} A_m B_p, \quad \sum_{i=1}^{n} A_p B_p\right\} \tag{8}$$

where A and B take the values $\{x, y, z\}$, $A_m = \boldsymbol{v_i}'(A) - \boldsymbol{u_i}'(A)$ is the component-wise difference (similarly B_m), and $A_p = \boldsymbol{v_i}'(A) + \boldsymbol{u_i}'(A)$ is the component-wise sum (similarly B_p). Altogether the set $\boldsymbol{\Psi}$ consists of 18 distinct statistics. Since, we translated the original vector sets these terms are sufficient to compute the best rotation for the least-squares superposition problem. Further, the sufficient statistics to compute the translation are simply the component-wise sums for each of the two (untranslated) vector sets $\mathcal{U}$ and $\mathcal{V}$.

4 Updating sufficient statistics

4.1 Updating sufficient statistics under an addition operation

Consider two pairs of corresponding vector sets: $\mathcal{Q} \leftrightarrow \mathcal{R}$ containing n_1 correspondences and $\mathcal{S} \leftrightarrow \mathcal{T}$ containing n_2 correspondences. Let $\mathcal{U}$ be defined as a combination of vectors $\mathcal{Q}$ and $\mathcal{S}$) and similarly $\mathcal{V}$ as a combination of $\mathcal{R}$ and $\mathcal{T}$. Let $\boldsymbol{\Psi}_1$ denote the sufficient statistics of the first pair after superposition and $\boldsymbol{\Psi}_2$ denote the same for the second pair. Define these as:

$$\boldsymbol{\Psi}_1 = \left\{ \sum_{i=1}^{n_1} C_m D_m, \sum_{i=1}^{n_1} C_m D_p, \sum_{i=1}^{n_1} C_p D_p \right\}, \boldsymbol{\Psi}_2 = \left\{ \sum_{i=1}^{n_2} E_m F_m, \sum_{i=1}^{n_2} E_m F_p, \sum_{i=1}^{n_2} E_p F_p \right\} \tag{9}$$

Where C, D, E and F take the values x, y, or z, denoting the respective components of vectors in the sets. Consistent with the previous notation (see Equation 8), C_p and C_m (similarly D_p and D_m) are the component-wise sums and differences between corresponding vectors in $\mathcal{Q}$ and $\mathcal{R}$. The same definitions hold for E_m (and E_p) and F_m (and F_p), with respect to corresponding vectors in $\mathcal{S}$ and $\mathcal{T}$.

We want to use $\boldsymbol{\Psi_1}$ and $\boldsymbol{\Psi_2}$ to compute a new set of sufficient statistics $\boldsymbol{\Psi}$ (defined in Equation 8) for the superposition of vector sets $\mathcal{U} = \mathcal{Q} + \mathcal{S}$ with $\mathcal{V} = \mathcal{R} + \mathcal{T}$. Below we derive the construction of the new sufficient statistics. The statistics involved in computing the new centroids of the sets $\mathcal{U}$ and $\mathcal{V}$, $\sum_{i=1}^{n=n_1+n_2} \boldsymbol{u}_i(A)$ and $\sum_{i=1}^{n=n_1+n_2} \boldsymbol{v}_i(A)$, can be trivially updated using the statistics $\sum_{i=1}^{n_1} \boldsymbol{q}_i(C)$, $\sum_{i=1}^{n_1} \boldsymbol{r}_i(D)$, $\sum_{i=1}^{n_2} \boldsymbol{s}_i(E)$, and $\sum_{i=1}^{n_2} \boldsymbol{t}_i(F)$. To compute the remaining statistics in $\boldsymbol{\Psi}$, we define the following vectors:

$$\boldsymbol{\alpha}_1 = \textbf{Centroid}(\mathcal{U}) - \textbf{Centroid}(\mathcal{Q}) \qquad \boldsymbol{\beta}_1 = \textbf{Centroid}(\mathcal{V}) - \textbf{Centroid}(\mathcal{R})$$

$$\boldsymbol{\alpha}_2 = \textbf{Centroid}(\mathcal{U}) - \textbf{Centroid}(\mathcal{S}) \qquad \boldsymbol{\beta}_2 = \textbf{Centroid}(\mathcal{V}) - \textbf{Centroid}(\mathcal{T})$$

Also define $\Delta_m^C = \boldsymbol{\beta_1}(C) - \boldsymbol{\alpha_1}(C)$, $\Delta_m^D = \boldsymbol{\beta_1}(D) - \boldsymbol{\alpha_1}(D)$, $\Delta_m^E = \boldsymbol{\beta_2}(E) - \boldsymbol{\alpha_2}(E)$, and $\Delta_m^F = \boldsymbol{\beta_2}(F) - \boldsymbol{\alpha_2}(F)$, where $A = C = E \in \{x, y, z\}$ and $B = D = F \in \{x, y, z\}$. These vectors define the corrections (or updates) required to the constituent centroids so that the new centroid can be constructed. Using these definitions, the lemma and corollaries below show the computation of $\boldsymbol{\Psi}$ from $\boldsymbol{\Psi_1}$ and $\boldsymbol{\Psi_2}$.

Lemma 1. $\displaystyle\sum_{i=1}^{n=n_1+n_2} A_m B_m = \sum_{i=1}^{n_1} C_m D_m + n_1 \varDelta_m^C \varDelta_m^D + \sum_{i=1}^{n_2} E_m F_m + n_2 \varDelta_m^E \varDelta_m^F$

Proof.

$$
\begin{aligned}
\text{Updated}(\sum_{i=1}^{n_1} C_m D_m) &= \sum_{i=1}^{n_1} \left[(\boldsymbol{r}'(C) + \boldsymbol{\beta_1}(C)) - (\boldsymbol{q}'(C) + \boldsymbol{\alpha_1}(C))\right]\left[(\boldsymbol{r}'(D) + \boldsymbol{\beta_1}(D)) - (\boldsymbol{q}'(B) + \boldsymbol{\alpha_1}(D))\right] \\
&= \sum_{i=1}^{n_1} \left[(\boldsymbol{r}'(C)\boldsymbol{r}'(D) - \boldsymbol{q}'(C)\boldsymbol{q}'(D) - \boldsymbol{r}'(C)\boldsymbol{q}'(D) + \boldsymbol{q}'(C)\boldsymbol{r}'(D)\right] \\
&+ \sum_{i=1}^{n_1} \left[(\boldsymbol{\beta_1}(C)\boldsymbol{r}'(D) - \boldsymbol{\alpha_1}(C)\boldsymbol{r}'(D) - \boldsymbol{\beta_1}(C)\boldsymbol{q}'(D) + \boldsymbol{\alpha_1}(C)\boldsymbol{q}'(D)\right] \\
&+ \sum_{i=1}^{n_1} \left[(\boldsymbol{r}'(C)\boldsymbol{\beta_1}(D) - \boldsymbol{r}'(C)\boldsymbol{\alpha_1}(D) - \boldsymbol{q}'(C)\boldsymbol{\beta_1}(D) + \boldsymbol{q}'(C)\boldsymbol{\alpha_1}(D)\right] \\
&+ \sum_{i=1}^{n_1} \left[(\boldsymbol{\beta_1}(C)\boldsymbol{\beta_1}(D) - \boldsymbol{\alpha_1}(C)\boldsymbol{\beta_1}(D) - \boldsymbol{\beta_1}(C)\boldsymbol{\alpha_1}(D) + \boldsymbol{\alpha_1}(C)\boldsymbol{\alpha_1}(D)\right] \\
&= \sum_{i=1}^{n_1} C_m D_m + \sum_{i=1}^{n_1} \varDelta_m^C D_m + \sum_{i=1}^{n_1} \varDelta_m^D C_m + \sum_{i=1}^{n_1} \varDelta_m^C \varDelta_m^D \\
&= \sum_{i=1}^{n_1} C_m D_m + \varDelta_m^C \sum_{i=1}^{n_1} D_m + \varDelta_m^D \sum_{i=1}^{n_1} C_m + n_1 \varDelta_m^C \varDelta_m^D \\
&= \sum_{i=1}^{n_1} C_m D_m + n_1 \varDelta_m^C \varDelta_m^D \qquad (\because \sum_{i=1}^{n_1} C_m = \sum_{i=1}^{n1} D_m = 0)
\end{aligned}
$$

Similarly, we can show that Updated$(\sum_{i=1}^{n_2} E_m F_m) = \sum_{i=1}^{n_2} E_m F_m + n_2 \varDelta_m^E \varDelta_m^F$. Adding the two updated statistics, the lemma above follows. □

Corollary 1. $\displaystyle\sum_{i=1}^{n=n_1+n_2} A_p B_p = \left[\sum_{i=1}^{n_1} C_p D_p + n_1 \varDelta_p^C \varDelta_p^D\right] + \left[\sum_{i=1}^{n_2} E_p F_p + n_2 \varDelta_p^E \varDelta_p^F\right]$

Corollary 2. $\displaystyle\sum_{i=1}^{n=n_1+n_2} A_m B_p = \left[\sum_{i=1}^{n_1} C_m D_p + n_1 \varDelta_m^C \varDelta_p^D\right] + \left[\sum_{i=1}^{n_2} E_m F_p + n_2 \varDelta_m^E \varDelta_p^F\right]$

4.2 Updating sufficient statistics under a deletion operation

Let us consider the case where we want to find a superposition under a deletion operation. Let $\mathcal{U} \leftrightarrow \mathcal{V}$ and $\mathcal{Q} \subset \mathcal{U} \leftrightarrow \mathcal{R} \subset \mathcal{V}$ denote two pairs of vector sets that are in correspondence. We want to compute the superposition of the vector sets $\mathcal{S} = \mathcal{U} \setminus \mathcal{Q}$ and $\mathcal{T} = \mathcal{V} \setminus \mathcal{R}$.

Using the same notations as in the previous section, it is straightforward to rearrange Lemma 1 and Corollaries 1 and 2 to derive the sufficient statistics $\mathbf{\Psi_2}$ of the superposition of $\mathcal{S}$ with $\mathcal{T}$ using the sufficient statistics $\mathbf{\Psi}$ (of $\mathcal{U}$ with $\mathcal{V}$) and $\mathbf{\Psi_1}$ (of $\mathcal{Q}$ with $\mathcal{R}$) as follows.

Corollary 3. $\sum_{i=1}^{n_2} E_m F_m = \sum_{i=1}^{n=n_1+n_2} A_m B_m - \sum_{i=1}^{n_1} C_m D_m - n_1 \Delta_m^C \Delta_m^D - n_2 \Delta_m^E \Delta_m^F$

Corollary 4. $\sum_{i=1}^{n_2} E_p F_p = \sum_{i=1}^{n=n_1+n_2} A_p B_p - \sum_{i=1}^{n_1} C_p D_p - n_1 \Delta_p^C \Delta_p^D - n_2 \Delta_p^E \Delta_p^F$

Corollary 5. $\sum_{i=1}^{n_2} E_m F_p = \sum_{i=1}^{n=n_1+n_2} A_m B_p - \sum_{i=1}^{n_1} C_m D_p - n_1 \Delta_m^C \Delta_p^D - n_2 \Delta_m^E \Delta_p^F$

5 Computing the r.m.s.d. from updated sufficient statistics, and other optimizations

It is easy to see that Kearsley's 4 × 4 symmetric quaternion matrix $\mathbf{Q}$ given in Equation 1 can be constructed using the updated sufficient statistics derived from the constituent sufficient statistics of previously superposed vector sets under addition and deletion operations in constant time.

In practice, $\mathbf{Q}$ is diagonalized and its eigenvalues and eigenvectors are computed using the numerical approach of Jacobi, where rotations are applied to $\mathcal{Q}$ iteratively, where each rotation annihilates (that is, sets to zero) a symmetric pair of off-diagonal entries in $\mathcal{Q}$. This approach has a fast convergence and the smallest diagonal element (or eigenvalue) is used to derive the r.m.s.d term, as described in Section 2.

However, in some cases, for instance, when a current superposition is extended by just one correspondence, $\mathcal{Q}$ changes marginally. Therefore, diagonalisation of $\mathcal{Q}$ can build on the previous solution. Let $\mathbf{Q}$ denote the quaternion matrix corresponding to the superposition of corresponding vector sets $\mathcal{U}$ and $\mathcal{V}$. From the eigenvalue decomposition theorem, we have $\mathbf{Q} = S\Lambda S^{-1}$, where S is the matrix of eigenvectors and Λ is the diagonal matrix of eigenvalues. Also note that $\mathbf{Q}$ is a positive semidefinite matrix with the property $\mathbf{Q}^T\mathbf{Q} = \mathbf{Q}\mathbf{Q}^T$. This implies that all the eigenvectors are orthogonal to each other. This further simplifies the decomposition to $\mathbf{Q} = S\Lambda S^T$. Also, since S is an orthogonal matrix, $\mathbf{Q} = S\Lambda S^T \implies \Lambda = S^T\mathbf{Q}S$.

Now, assume that the corresponding vector sets are augmented from $\mathcal{U}$ and $\mathcal{V}$ to $\mathcal{U}'$ and $\mathcal{V}'$, resulting in an updated Kearsley's matrix $\mathbf{Q}'$. We want to diagonalize this matrix into $S'\Lambda'S'^T$. Instead of starting the Jacobi's iterative process from scratch, we will use the previously computed eigenvectors (before the vector sets were augmented), S, and compute $\tilde{\Lambda}$ as $S^T \mathcal{Q}' S$. Notice that if the augmentation does not include drastic changes, then $\tilde{\Lambda}$ is nearly diagonal (that is, $\tilde{\Lambda} \approx \Lambda'$), thus requiring very few iterations to fully diagonalize $\tilde{\Lambda}$. This provides a further optimization to the diagonalization step under update operations on vector sets.

6 Results

Source code

We wrote a C++ library that allows superposition of vector sets. Superpositions can be carried out from scratch (using the raw coordinates), or, alternatively, using the sufficient statistics of constituent vector sets, by operating on them using set addition or deletion operations. The source code is available freely under a GNU public license from `http://www.csse.monash.edu.au/~karun/superpose3D`.

Consistency of superpositions using sufficient statistics

To validate the consistency of superpositions generated using sufficient statistics (under both addition and deletion operations discussed in Section 4) we undertake the following experiment. 8992 ASTRAL SCOP (Chandonia *et al.*, 2004; Murzin *et al.*, 1995) domains were used as the source structures from which superposable fragments were randomly sampled. The general procedure of sampling is as follows. From the list of source structures, randomly choose any structure. Within this structure choose 2 random fragments of lengths l_1 and l_2, where each fragment has between 10 and 40 residues. These chosen fragments form the sets $\mathcal{Q}$ and $\mathcal{S}$. Yet another structure is again chosen randomly from our source list, and two fragments are randomly extracted from it with exactly the same length, l_1 and l_2. These form the sets $\mathcal{R}$ and $\mathcal{T}$. Assuming one-to-one correspondence between $\mathcal{Q} \leftrightarrow \mathcal{R}$ we compute the sufficient statistics $\boldsymbol{\Psi}_1$ of their orthogonal superposition. Similarly the sufficient statistics $\boldsymbol{\Psi}_2$ is computed for the orthogonal superposition between $\mathcal{S} \leftrightarrow \mathcal{T}$. Define $\mathcal{U} = \mathcal{Q} + \mathcal{S}$ and $\mathcal{V} = \mathcal{R} + \mathcal{T}$. Iterating this random sampling 100 million times, we computed:

1. the r.m.s.d of superposition of $\mathcal{U} \leftrightarrow \mathcal{V}$ from scratch (using the raw coordinates in the vector sets). Denote this r.m.s.d as ρ_1
2. the r.m.s.d of superposition of $\mathcal{U} \leftrightarrow \mathcal{V}$, but using the sufficient statistics $\boldsymbol{\Psi}_1$ and $\boldsymbol{\Psi}_2$ of superpositions of constituent vector sets $\mathcal{Q} \leftrightarrow \mathcal{R}$, and $\mathcal{S} \leftrightarrow \mathcal{T}$. Denote this r.m.s.d as ρ_2.

We measure the difference between the two r.m.s.d values, $\Delta\rho = \rho_1 - \rho_2$. Over the 100 million samples, the mean and standard deviation of $\Delta\rho$ was found to be zero to a very high precision ($< 10^{-17}$).

We repeat the same experiment to validate superpositions under deletion operation using sufficient statistics, where, in each iteration, we compute the superposition of vector sets $\mathcal{S} \leftrightarrow \mathcal{T}$, with $\mathcal{S} = \mathcal{U} \backslash \mathcal{Q}$ and $\mathcal{T} = \mathcal{V} \backslash \mathcal{R}$. This experiment again confirms the same consistency as observed in the test on addition operation.

Measuring the performance gain using sufficient statistics for superpositions

We demonstrated in Section 4 that superpositions under addition and deletion can be updated in constant time, building on the sufficient statistics of the constituent sets. This was also validated empirically using the experiments above. We will now measure the gain in performance using this approach by comparing it with superpositions built from scratch.

Figures 1(a)-(c) show the runtime plots of three sets of randomly chosen 10 million joint superpositions carried out from scratch (Blue line) and compared against the same superpositions updated using sufficient statistics (Green line). We note that these three sets vary in the (average) size of the joint superpositions being carried out, as indicated in the plot titles. Fig. 1

Notice that, when joint superpositions are carried out from scratch ignoring the sufficient statistics, the average size of the superpositions introduces a constant factor to the run time. This is expected as each superposition is linear in the size of the vector sets being superposed. Consequently, the slopes of those blue lines across the three plots in Figure 1 become steeper with the increase in the superposition size. On the other hand, when the joint superpositions are updated in constant time, the updates are independent of the superposition size. This is because, any update involves recomputing only a small (fixed) number of sufficient statistics. This is clearly reflected in the slopes of the green lines being unchanged across the three plots in Figure 1.

More formally, if $|J|$ is the number of joint superpositions and l is the (average) number of vectors being superposed, then the first method (Blue line in Figure 1) grows as $O(l|J|)$. Since $l \ll |J|$ we see a linear trend (with a steeper gradient accounting for the multiplier l in the complexity term). In comparison, the results with sufficient statistics (Green line in Figure 1) grow simply as $O(|J|)$, independent of the superposition size.

Using sufficient statistics of superposition in the setting of structural alignments

As discussed in the introduction, a common heuristic employed to compute a structural alignment between pairs of structures involves collecting a library of well-fitting fragments. This library is refined by jointly superposing pairs from this library, and a final structural alignment is assembled from these results.

Table 1

To test the potential performance gain by using sufficient statistics, we computed the time taken to undertake an exhaustive joint superposition on libraries of well-fitting fragments corresponding to a small collection of structural pairs. For each pair of structures chosen, the well-fitting fragments are identified as those that superpose *maximally* within an r.m.s.d threshold of 2 Å. By maximal, we mean those fragment pairs that cannot be extended any further without violating the r.m.s.d. threshold.

Table 7 shows the run times of joint superpositions performed exhaustively on a small set of protein structural pairs. As can be seen from the table, using sufficient statistics for superpositions results in up to an order of magnitude improvement in the run time to carry our these superpositions exhaustively. Since performing this task *without* sufficient statistics creates a computational bottleneck, existing structural alignment programs attempt to drastically restrict the number of superpositions, often trading off structural alignment quality for speed. We note that the improvements gained from using sufficient statistics for superpositions will allow these restrictions to be generously relaxed without any effect on the current run times, but potentially improving the structural alignment quality.

All the above experiments were carried out on a standard laptop with 2.2GHz Intel® CPU and 4GB RAM.

7 Conclusion

Optimal superpositions of vector sets are central to identify similarities and differences between spatial objects. We derived a set of sufficient statistics for the orthogonal superposition problem under the least squares criterion. These statistics provide an efficient way to operate (via addition and deletion of vectors) on previously computed superpositions. Our results demonstrate a drastic improvement in the computational effort required to compute r.m.s.d. based on sufficient statistics. These results are relevant to many analyses involving structural data. These include the plethora of algorithms to construct pairwise and multiple protein structural alignments by assembling fragment pairs. Source code (written in C++) to undertake superpositions implementing our work can be downloaded from `http://www.csse.monash.edu.au/~karun/superpose3D`.

Bibliography

Chandonia, J.-M., Hon, G., Walker, N. S., Conte, L. L., Koehl, P., Levitt, M., and Brenner, S. E., 2004. The ASTRAL compendium in 2004. *Nucleic acids research* 32, D189–D192.

Cohen, G., 1997. ALIGN: a program to superimpose protein coordinates, accounting for insertions and deletions. *Journal of applied crystallography* 30, 1160–1161.

Coutsias, E. A., Seok, C., and Dill, K. A., 2004. Using quaternions to calculate RMSD. *Journal of Computational Chemistry* 25, 1849–1857.

Cox, J. M., 1967. Mathematical methods used in the comparison of the quaternary structures. *Journal of Molecular Biology* 28, 151–156.

Diamond, R., 1988. A note on the rotational superposition problem. *Acta Crystallographica Section A: Foundations of Crystallography* 44, 211–216.

Eidhammer, I., Jonassen, I., and Taylor, W. R., 2004. *Protein Bioinformatics: An algorithmic approach to sequence and structure analysis.* J. Wiley & Sons.

Hamilton, W. R. and Hamilton, W. E., 1866. *Elements of quaternions.* Longmans, Green, & Company.

Hogg, R. V. and Craig, A., 1994. *Introduction to mathematical statistics.* Prentice Hall.

Jacobi, C. G. J., 1846. Über ein leichtes Verfahren, die in der Theorie der Säkularstörungen vorkommenden Gleichungen numerisch aufzulösen. *Journal für die Reine und Angewandte Mathematik* 30, 51–95.

Kabsch, W., 1976. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography* 32, 922–923.

Kabsch, W., 1978. A discussion of the solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography* 34, 827–828.

Kearsley, S. K., 1989. On the orthogonal transformation used for structural comparisons. *Acta Crystallographica Section A: Foundations of Crystallography* 45, 208–210.

KenKnight, C., 1984. Comparison of methods of matching protein structures. *Acta Crystallographica Section A: Foundations of Crystallography* 40, 708–712.

Koehl, P., 2001. Protein structure similarities. *Current opinion in structural biology* 11, 348–353.

Kolodny, R., Koehl, P., and Levitt, M., 2005. Comprehensive evaluation of protein structure alignment methods: scoring by geometric measures. *Journal of molecular biology* 346, 1173–1188.

Konagurthu, A. S., Whisstock, J. C., Stuckey, P. J., and Lesk, A. M., 2006. MUSTANG: a multiple structural alignment algorithm. *Proteins: Structure, Function, and Bioinformatics* 64, 559–574.

Lackner, P., Koppensteiner, W. A., Sippl, M. J., and Domingues, F. S., 2000. Prosup: a refined tool for protein structure alignment. *Protein Engineering* 13, 745–752.

Lesk, A., 1986. A toolkit for computational molecular biology. II. on the optimal superposition of two sets of coordinates. *Acta Crystallographica Section A: Foundations of Crystallography* 42, 110–113.

Lesk, A. M., 2000. The unreasonable effectiveness of mathematics in molecular biology. *The Mathematical Intelligencer* 22, 28–37.

Lesk, A. M., 2001. *Introduction to protein architecture: the structural biology of proteins.* Oxford University Press.

Mackay, A. L., 1984. Quaternion transformation of molecular orientation. *Acta Crystallographica Section A: Foundations of Crystallography* 40, 165–166.

McLachlan, A. D., 1972. A mathematical procedure for superimposing atomic coordinates of proteins. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography* 28, 656–657.

McLachlan, A. D., 1982. Rapid comparison of protein structures. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography* 38, 871–873.

Murzin, A. G., Brenner, S. E., Hubbard, T., and Chothia, C., 1995. SCOP: a structural classification of proteins database for the investigation of sequences and structures. *Journal of molecular biology* 247, 536–540.

Shatsky, M., Nussinov, R., and Wolfson, H. J., 2002. Flexible protein alignment and hinge detection. *Proteins: Structure, Function, and Bioinformatics* 48, 242–256.

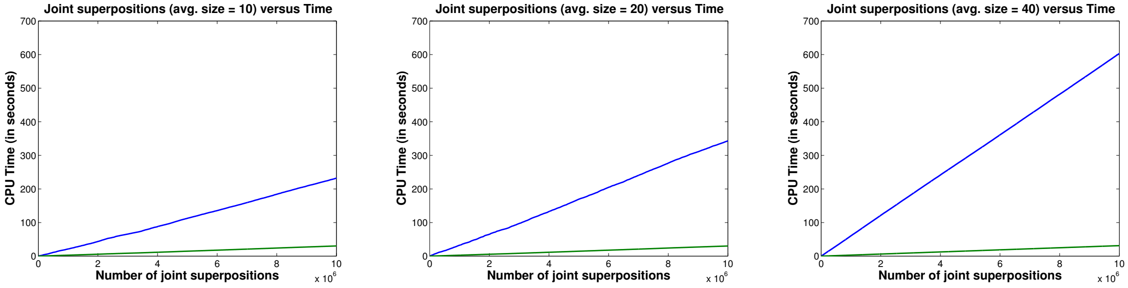
Shatsky, M., Nussinov, R., and Wolfson, H. J., 2004. A method for simultaneous alignment of multiple protein structures. *Proteins: Structure, Function, and Bioinformatics* 56, 143–156.

Shindyalov, I. N. and Bourne, P. E., 1998. Protein structure alignment by incremental combinatorial extension (ce) of the optimal path. *Protein engineering* 11, 739–747.

Vriend, G. and Sander, C., 1991. Detection of common three-dimensional substructures in proteins. *Proteins: Structure, Function, and Bioinformatics* 11, 52–58.

Ye, Y. and Godzik, A., 2003. Flexible structure alignment by chaining aligned fragment pairs allowing twists. *Bioinformatics* 19, ii246–ii255.

List of Figures



(a) (b) (c)

Fig. 1. The CPU times (in seconds) performing joint superpositions from scratch (Blue line) compared against the same using sufficient statistics (Green line) over 10 million random fragment data sets derived from ASTRAL SCOP domains. The three plots vary in the average superposition size as indicated in the title.

$\Uparrow$

Arun S. Konagurthu[1,*], Parthan Kasarapu[1], Lloyd Allison[1], James H. Collier[1], Arthur M. Lesk[2]

List of Tables

Table 1. Time taken to perform exhaustive joint superpositions on a library of well-fitting fragment pairs between two structures from different families.

Protein Family	structural pair `wwPDB` IDs	Number of joint superpositions	Average size of superpositions	Time in seconds (from scratch)	Time in seconds (sufficient stats)
Serine Proteinases	`3EST` vs. `2PKA`	18,486,240	14	419.6	56.0
Calmodulin-like	`1NCX` vs. `2SAS`	67,820,481	18	1618.4	178.2
Serine Proteinases	`3EST` vs. `2SNV`	71,025,321	16	1328.0	187.8
Globins	`1HHOA` vs. `1HHOB`	74,890,441	20	1923.3	194.6

$\Uparrow$

Arun S. Konagurthu[1,*], Parthan Kasarapu[1], Lloyd Allison[1], James H. Collier[1], Arthur M. Lesk[2]