INVITED ARTICLE

mtsslSuite: In silico spin labelling, trilateration and distance-constrained rigid body docking in PyMOL

Gregor Hageluekena,*, Dinar Abdullina, Richard Warden and Olav Schiemanna,b

aInstitute of Physical and Theoretical Chemistry, University of Bonn, Bonn, Germany; bBiomedical Sciences Research Complex, The University of St. Andrews, Fife, UK

(Received 8 March 2013; final version received 18 May 2013)

Nanometer distance measurements based on electron paramagnetic resonance methods in combination with site-directed spin labelling are powerful tools for the structural analysis of macromolecules. The software package mtsslSuite provides scientists with a set of tools for the translation of experimental distance distributions into structural information. The package is based on the previously published mtsslWizard software for in silico spin labelling. The mtsslSuite includes a new version of MtsslWizard that has improved performance and now includes additional types of spin labels. Moreover, it contains applications for the trilateration of paramagnetic centres in biomolecules and for rigid-body docking of subdomains of macromolecular complexes. The mtsslSuite is tested on a number of challenging test cases and its strengths and weaknesses are evaluated.

Keywords: DEER; PELDOR; triangulation; EPR; FRET; docking

Introduction

Electron paramagnetic resonance (EPR) based distance measurements in biological macromolecules are a valuable complement to crystallography or NMR studies [1–3]. The most commonly used EPR technique for distance measurements is pulsed electron–electron double resonance (PELDOR, also known as DEER) [4]. It can be used to accurately determine interspin distances in the range within 15–80 Å [5]. Since most biological macromolecules are not paramagnetic, site-directed spin labelling (SDSL) with nitroxides is often used to incorporate spins into proteins [6,7] or nucleic acids [8,9]. However, the translation of PELDOR-derived distances into structures is often affected by the flexibility and dimension of the spin labels used. Thus, an accurate model of the conformational distribution of the spin labels attached to the biomolecule is required to draw structural conclusions from the distance data. Since crystallographic studies of multiple spin-labelled mutants of a biomolecule are in general not feasible; this model will in most cases be based on computer simulations. The currently available simulation programs MMM, mtsslWizard and PRONOX that generate such label distributions predict experimental PELDOR distances between two nitroxide spin labels with errors in the range of ∼3 Å [10–12].

Once, a model of the macromolecule has been spin labelled in silico, PELDOR-derived distances can be used to localise individual spin centres such as paramagnetic metal ions or EPR-active substrates of enzymes relative to the model of the macromolecular structure [13,14]. Although the location of metal ions can in most cases be inferred from sequence alignments or crystal structures, in some cases this problem is far from trivial, even if high-resolution structures are available. An example of this is metallonucleases where multiple metal centres can be part of the active site and even movements of the metals during the reaction cycle are proposed [15]. Although crystal structures of these enzymes are available, the location and function of the metal ions are still under debate [15]. Similarly, the localisation of the active site of a novel enzyme (especially non-metallo enzymes) can be difficult, for example, when only a structure of the apo-enzyme is available. The trilateration of spin-labelled or intrinsically EPR-active substrates can in such cases provide a quick and cost-effective solution to locate the active site or a metal-binding site. The general idea of trilateration is the localisation of a certain object in three-dimensional (3D) space by measuring distances between this object and known reference objects [16]. In the context of EPR spectroscopy, spin labels introduced by SDSL would serve as reference objects for the trilateration and the object to be localised would be the intrinsic spin centre, such as a metal ion [14], the spin-labelled or intrinsically EPR-active substrate [13] or additional spin labels [17,18].

Another important application for PELDOR-derived distances is the in silico reconstruction of macromolecular complexes from individual domains. Often, structural information is only available for subunits of such complexes and PELDOR can be used to reconstruct the complex based on distance-constrained rigid-body docking or refinement.

*Corresponding author. Email: hagelueken@pc.uni-bonn.de

© 2013 The Author(s). Published by Taylor & Francis.
This is an Open Access article. Non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly attributed, cited, and is not altered, transformed, or built upon in any way, is permitted. The moral rights of the named author(s) have been asserted.
Results and discussion

All programs of the mtslsSuite are designed to work in conjunction with the freely available PyMOL molecular graphics program (www.pymol.org). MtslTrilaterate and mtslDock can use the models of spin-labelled macromolecules generated by means of mtslWizard or other packages as input data. In the following, the usage and the special features of each program will be discussed with the help of examples.

mtslWizard

The mtslWizard program allows the user to spin label macromolecules in silico. Its graphical user interface (GUI) is fully integrated into PyMOL. Generally, the user can choose the type of spin label from the mtslWizard menu (supplementary Figure 1) and then attach the chosen label to the selected residue of a biomolecule by clicking on the amino acid/nucleotide it should be positioned at [10]. The program rotates the spin label around its rotatable bonds in order to explore the volume, which is accessible to the label (‘tether-in-a-cone-model’, [23–27]). MtslWizard does not consider preferred rotamers of the spin label but only excludes conformations that produce clashes with the macromolecular surface or with a label itself, considering a user-defined cut-off value. The advantages and limitations of this approach are thoroughly discussed in a previous publication [10]. In the following, only the changes between versions (1.0) and (1.1) are discussed.

In the first version of MtslWizard (1.0), only the spin-label MTSSL was included into the program and it was shown that experimental data from a large test data-set (52 distances) can be predicted with an average accuracy of 3 Å, which is on par with other available software packages (MMM, PRONOX). Here, we present an improved version of mtslWizard (1.1), which contains several new features in comparison with the original version (1.0). Firstly, the algorithm has been optimised, so that the calculation speed has increased and is no longer dependent on the size of the macromolecular model. Secondly, the GUI has been slightly modified in order to make it more comprehensible (supplementary Figure 1). For example, the ‘allowed clashes’ criterion has been merged with the ‘vdW-cut-off’ criterion. The resulting ‘vdW-restraint’ setting can now be switched between two values: ‘tight’ (3.4 Å cut-off, 0 clashes) and ‘loose’ (2.5 Å cut-off, 5 clashes). The ‘loose’ setting allows closer contacts between label and protein, accounting
for flexibility of the system. This setting is useful if a site can be experimentally labelled but cannot be labelled \textit{in silico} using the default ‘tight’ setting, because the static molecular model is too crowded. As discussed previously [10], the value of 2.5 Å was chosen as a minimum cut-off value because it represents the distance between the donor and acceptor of a short hydrogen bond, whereas the 3.4 Å cut-off represents a short vdW interaction (ignoring the hydrogens). The mtsslWizard asks the user to switch to the ‘loose’ setting if less than 10 conformations of a label can be found during a run. Importantly, the new release of mtsslWizard includes several additional spin labels that were requested by users (Figure 1(A)): the PROXYL spin label, the gadolinium-based DOTA spin label [28] and two spin labels for nucleic acids, the C-label [29] and a urea-based TEMPO spin label (URIP) [30]. Figure 1(B) shows examples for use of the newly included spin labels. Since mtsslWizard is based on the accessible volume approach, any spin or FRET label with known structure can be easily added as a plug-in to the program, even if no rotamer libraries are available.

Version (1.1) of the program was benchmarked against the original test data-set [10] consisting of 52 MTSSL-derived distances (Figure 2(A)). The results confirm that the program can predict experimental data with an accuracy of 3 Å, similar to MMM or PRONOX (see also [31]). The slight deviations in the prediction of some of the interspin distances between this version of mtsslWizard and the original version are due to the simplified clash criterion (see above). For the newly added spin labels, no large experimental data-sets are available, and we therefore cannot provide a similarly extensive benchmark as for the MTSSL spin label. For the PROXYL label, we used mtsslWizard to predict experimental distance distributions from PROXYL-labelled light-harvesting complex II [12]. The overlay of the experimental distance distributions with the ones from mtsslWizard (Figure 2(B)) shows that the program can predict the PROXYL-based distributions with good accuracy. The DOTA label was tested using a doubly DOTA-labelled transmembrane peptide resulting in good fits of experimental distance distributions [32]. To test the C spin label, a model of a DNA duplex (make-NA server, http://structure.usc.edu/make-NA/server.html) was constructed using the sequence of ds-DNA1 (fwd: 5′-GATGC GGFCGGCGCATGAC-3′, rev: 3′-CTACGGCGCGCTAFTG-5′) from [33] and the C spin label was attached to the positions marked by ‘F’ in the sequence using mtsslWizard. The distance between the modelled C labels is 37.6 Å, close to the experimental value of 36.5 Å Figure 2(D). Note that due to the rigidity of the C label, it is only attached to the selected nucleotide by mtsslWizard and not rotated in any way. As a test for the URIP label, a double-stranded polyA:T 15mer was experimentally spin labelled using URIP at positions 3 and 8 of the T strand and PELDOR data were recorded (Figure 2(C)). A model of the labelled DNA strand was constructed using the make-NA server (http://structure.usc.edu/make-NA/server.html), PyMOL and mtsslWizard. Figure 2(C) shows that experimental data (mean distance 27.7 Å [Deer-Analysis]) and prediction (mean distance 27.0 Å) fit well.

\textbf{mtsslTrilaterate}

The mtsslTrilaterate program can be used to locate spin centres in the structure of biomolecules by means of trilateration, for example, to find metal- or substrate-binding sites (see above).

Once the trilateration program has been launched via the PyMOL menu, its GUI appears as shown in supplementary Figure 2(A). The trilateration process can be prepared by importing the attached spin labels and experimental distances into the program as explained in the software manual (www.pymolwiki.org). The distance data itself can be entered in two different ways: manually or by import of data files. The program can, for example, import DeerAnalysis [34] interspin distance distributions. The program solves the trilateration problem in two steps. In the first step, the coordinates of the unlocalised spin centre are estimated by means of singular value decomposition (SVD). The result of this step is needed as an ‘initial guess’ for the nonlinear least squares (NLS) optimisation. The goal of NLS algorithms is the minimisation of the commonly used \( \chi^2 \) merit function,

\[
\chi^2(x, y, z) = \sum_l \left[ \frac{r_l - \sqrt{(x - x_l)^2 + (y - y_l)^2 + (z - z_l)^2}}{\sigma_l} \right]^2,
\]

where \( l \) is the number of the spin label, \((x_l, y_l, z_l)\) are coordinates of the \( l \) spin label, \( r_l \) and \( \sigma_l \) are the mean value and the standard deviation of the distance between the \( l \) spin label and the unlocalised spin centre. Here, \((x, y, z)\) are the unknown coordinates of the spin centre. They are incremented in each iteration and the corresponding value of \( \chi^2 \) is calculated. Then, the program checks whether the obtained \( \chi^2 \) is decreased by a factor of less than \( 10^{-3} \) after the last iteration. If this criterium is satisfied, the problem is considered to be solved. The NLS algorithm uses either the inverse-Hessian method or the Levenberg–Marquardt method [35]. The user can choose the algorithm in the ‘Preferences’ menu (supplementary Figure 2(B) and (C)). In our hands, both algorithms produce exactly the same results. For each algorithm, a number of parameters can be set in the ‘Preferences’ menu (supplementary Figure 2(B) and (C)). These settings include the maximum number of iterations, the minimal \( \chi^2 \) value (below which the problem is considered solved) and the damping parameter lambda (used only for the Levenberg–Marquardt algorithm). The values used as default were found to be optimal for the test cases described below. Further description of these parameters can be found elsewhere [35]. The
solution of the trilateration problem can be interpreted as the most probable coordinates of the unlocalised spin centre and their standard errors. These data are presented as a table in the ‘Output’ panel (supplementary Figure 2(A)) and can optionally be visualised graphically inside PyMOL (Figures 3 and 4). The calculation statistics are presented in terms of the obtained $\chi^2$ value and the number of NLS iterations used to reach this $\chi^2$. The $\chi^2$ value shows how precisely the trilateration problem was solved for the given set of input data. A lower value of $\chi^2$ corresponds to a lower uncertainty in the coordinates obtained. The factors that affect the $\chi^2$ value are the number of spin labels used to localise the intrinsic spin centre, the precision and the dispersion of the distances evaluated from PELDOR time traces and the accuracy of the model of the spin-labelled biomolecule.

We tested the mtsslTrilaterate program for soybean seed lipoxygenase-1 (SBL1) using a data-set of
Figure 3. Trilateration of the LOPTC lipid in soybean seed lipooxygenase-1 (SBL1). (A) SBL1 is shown as green cartoon model. The MTSSL ensembles that were generated with mtsslWizard are shown as blue and red stick models. The trilateration spheres whose radii correspond to the experimental LOPTC-MTSSL distances are shown in cyan. The trilateration result ('target') is represented as an orange ellipsoid and marks the position of the LOPTC lipid. (B) Close up of the ‘target’ area. The protein is shown as green cartoon model; selected amino acid residues are shown as sticks. The target is represented by a translucent orange ellipsoid. The magenta spheres inside the ellipsoid are the results of 10 independent attempts of docking LOPTC into SBL1 with mtsslDock (marked by arrow).

PELDOR-derived distances published by Gaffney et al. [13]. The data-set consists of five distances measured between the TEMPO-labelled lipid (LOPTC) and five spin labels that were attached to lipooxygenase by SDSL. In the original paper, these data were used to localise the polar end of the LOPTC on the molecular surface of lipooxygenase by means of trilateration using multi-dimensional scaling and Procrustes analysis that are included in the MATLAB (http://www.mathworks.de) statistical toolbox. The mtsslTrilaterate program was run with the input parameters listed in Table 1. The coordinates of the spin labels were obtained by using mtsslWizard and the crystal structure of SBL1 (PDB:1YGE, [36]). The mean values and standard deviations of the interspin distances between LOPTC and five spin labels were taken from [13]. Calculations were performed with the default settings of mtsslTrilaterate (see supplementary Figure 2(B) and (C)). The results are shown in Figure 3 and Table 1. As can be seen from Figure 3, the program places the LOPTC spin centre next to helices 2 and 11 of SBL1. In this position, LOPTC is surrounded by amino acids Glu236, Lys260, Gln264 and Gln544 (Figure 3(B)). Within the error of calculations, the trilateration result coincides with those published in the original paper (Table 1). A slight deviation between these two solutions may be related to the usage of different models of the spin-labelled protein (either derived from mtsslWizard or PRONOX) and different calculation procedures. To estimate the influence of differences in the models on the solution, the mtsslTrilaterate program was run with the coordinates of the spin labels taken from [13]. It was found that the coordinates of the LOPTC spin derived in this way fit with high precision to the published coordinates. This fact indicates that the small deviations between the solutions are related to the differences in the models of spin-labelled protein. With mtsslTrilaterate, this localisation of LOPTC was done within 10 min, including the in silico spin labelling of the labelled positions with mtsslWizard.

As a second test case, we took an experimental data-set that was used to localise an inhibitory copper-binding site in EcoRI [14]. Position Ser180 in both monomers of the dimeric protein (PDB-ID: 1CKQ) was labelled with mtsslWizard and the experimental distances between the spin label and the copper site to be localised were used as input for mtsslTrilaterate. Although these input data are not sufficient for the program to solve the trilateration with a unique solution (at least four distances are needed for this), the visualisation of the trilateration spheres in PyMOL (Figure 4) shows that the...
His114, which was also identified as the Cu²⁺ point of closest approach of the two spheres coincides with the Cu²⁺ binding site in the original paper.

**mtsslDock**

The mtsslDock program enables the user to dock two macromolecules based on a set of experimental PELDOR distances. The program requires models of the spin-labelled macromolecules and the experimental distance data between them as input. The structural models can be spin labelled with mtsslWizard or other available software packages.

After launching mtsslDock from within PyMOL, its GUI appears as shown in supplementary Figure 3. The docking process is then prepared by importing the spin-labelled docking partners into the program as explained in the software manual (www.pymolwiki.org). In the following, the algorithm and the adjustable settings are explained: mtsslDock uses a mixed genetic and evolutionary algorithm [37] to find docking models between two macromolecules (e.g. proteins A and B) that agree to the experimental distance data. The principle of genetic and evolutionary algorithms is based on natural evolution where inheritance, mutation and genetic crossover between generations lead to a population with increased fitness for survival in the particular environment. Such algorithms have been used for a wide array of problems, including protein–protein docking based on surface complementarity [38] or molecular replacement in crystallographic phasing [39]. The algorithm works as follows: protein A is held fixed in space, while in the first step of the calculation, mtsslDock generates a population of 400 random orientations (each orientation is called a chromosome) of protein B. A chromosome is a set of six numbers (‘genes’), which determine the translation of protein B clashing into protein A (only Cα atoms are considered; a distance lower than 3.4 Å between two Cα atoms is counted as a clash). The 5% fittest trial orientations are then used as starting orientations for a second evolution (supplementary Figure 4) are introduced to the fittest chromosome – here, a low χ² value corresponds to a higher fitness than a high χ² value. In the next step, the 5% least-fit chromosomes of the population are discarded and replaced by ‘offspring’ of the surviving chromosomes (supplementary Figure 4). The offspring is generated by genetic operations: random mutation and single-point crossover (e.g. [38], supplementary Figure 4), forming a new generation of the population. This process is repeated 1000 times before a ‘packing-function’ adds a high fitness penalty to chromosomes that lead to protein B clashing into protein A (only Cα atoms are considered; a distance lower than 3.4 Å between two Cα atoms is counted as a clash). The 5% fittest trial orientations are then used as starting orientations for a second evolution with 50 generations in which only small creep mutations (supplementary Figure 4) are introduced to the fittest chromosome to replace the 95% unfittest chromosomes in each step. Here, the fitness is determined by scoring χ² as well as clashes between both proteins. This effectively resembles a least-squares rigid-body refinement. In some cases,
conformational changes of the docking partners between the bound- and unbound state might lead to ‘false negative’ docking results due to the clash criterion. To account for this, mtsslDock does not simply discard clashing solutions but displays them marked as ‘clashing’ inside the GUI and within PyMOL, so that ultimately the user decides whether a solution should be discarded or not.

By default, the program calculates 10 independent populations and exports the fittest solution of each population to PyMOL to be investigated by the user. A docking calculation takes between 0.5 and 5 min for a single population on a standard laptop, depending on the settings that were used and the size of the molecules (due to the clash scoring). We recommended to do the calculation with at least 10 independent populations to get an impression of how statistically significant the produced solutions are.

The robustness of the algorithm was analysed using the crystal structure of the protein–protein complex rubredoxin:rubredoxin-reductase (PDB-ID: 2vb3, [40]). The rubredoxin reductase was kept static (protein A), while rubredoxin was allowed to be moved by mtsslDock (protein B). Both proteins were spin labelled at 12 positions with mtsslWizard (supplementary Figure 5), and 4–12 of the 144 possible distances were randomly selected as constraints. Each label was only allowed to contribute to one distance constraint. Because of the statistic nature of the docking algorithm, the results of this analysis have to be regarded in terms of the ability of the algorithm to reliably find a correct docking solution with a given number of constraints. For the purpose of our benchmark, we arbitrarily define a correct solution as one that has an RMSD (all atoms) to the experimental position of protein B of less than 2.5 Å. Figure 5(A) (top) shows that in each docking run of our benchmark, the algorithm found potential solutions that fit to the ‘experimental’ distances with RMSD values <1 Å. However, only when six or more constraints were used, was the correct solution among the docked structures (Figure 5(A), bottom). With an increasing number of constraints, the fraction of wrong solutions decreased, simply because it becomes less likely for the algorithm to find an additional solution that satisfies all distance constraints (Figure 5(A), bottom). Figure 5 (B) and (C) shows the docking solutions when 7 or 10 constraints were used, respectively. Whereas all solutions were correct when 10 constraints were used (Figure 5(C)), the 10 solutions derived from the 7 constraints form 2 clusters (Figure 5(B), red and green), one of which is located at the correct position. For both the clusters, all solutions show a similar fit to the ‘experimental data’ with RMSD <1 Å (Figure 5(A), top). Importantly, this is much lower than the uncertainty that is on average introduced by in silico spin labelling (~3 Å) [10,31] and experimental errors. Figure 5(D) shows that interestingly, the ‘worst’ solution of the correct cluster has a χ² value that is higher than that of the incorrect solutions. Thus, in reality, right and wrong solutions cannot simply be distinguished by ranking the solutions. In our benchmark case, it would be straightforward to identify the correct cluster of protein B, since the wrong solutions do not make any contact with protein A. However, in reality and especially in cases, where, for example, a domain of the protein complex is missing from the structure, more constraints would have to be added to get a definite answer.

Proteins often form functional dimers and sometimes only the monomeric structure or competing dimer models can be derived from crystallography. PELDOR can in such cases be used to determine the native dimer structure [22]. We thus added an option to mtsslDock that only C2 symmetric docking models are constructed and evaluated during the docking run. In theory, less constraints (and therefore experimental time) should be needed in such cases, because the system has less degrees of freedom. We analysed this function of mtsslDock with the same benchmark, but used the structure of the dimeric Cap protein (Figure 1(A), PDB-ID: 1O3Q) as a test case. This time, six labels were attached to each monomer (at corresponding positions) of the protein (supplementary Figure 5) and six, five or four distances between monomers were randomly chosen as constraints. Figure 5(A) (blue dots) shows, that as expected, correct solutions can be found with down to four constraints. More symmetry restraints will be added in future releases of the software.

In summary, our algorithm can reliably dock two structures when enough distance constraints are available (six or more for unsymmetric structures, four or more for C2 symmetric structures). It cannot distinguish a correct from an incorrect solution when distances derived from both solutions show a similarly good fit to the experimental distances. This situation can arise when not enough constraints are present, when the algorithm gets trapped in local minima or from specific placements of labels. For example, when three labels were placed on protein A, they would form a mirror plane for any number of labels on protein B, and the algorithm would not be able to distinguish the solution from its mirror image.

We also tested mtsslDock with three published sets of experimental interspin distances. The first example is the complex between the cytoplasmic domain of erythrocyte band 3 and ankyrin-R repeats 13–24 [20]. This complex has been investigated by various biochemical and biophysical techniques, including a set of 20 PELDOR measurements. In the original paper, the PELDOR-derived distances were used as distance constraints for docking analysis with RosettaDock [41], resulting in a model of the protein complex that is in agreement with other biochemical and biophysical data [20]. To dock the complex with mtsslDock, we labelled the experimentally used sites in both proteins with mtsslWizard, imported the mean coordinates of the labels into mtsslDock and entered the 20 experimental distances and standard deviations [20]. The result of the
docking calculation (10 docking runs were used, calculation time: 30 min) is shown in Figure 6 and resembles the result that was obtained in the original publication (supplementary Figure 5). Interestingly, the program produces two clusters of possible solutions, which have similar fitness scores (Figure 6, supplementary Figure 5). Both of these clusters are contained in the ensemble of the published solutions. Since the number of constraints in this case greatly exceeds six (see above and Figure 5(A)), it seems surprising that our algorithm does not find a single solution. There are several possible reasons for this: (1) some of the distances might be incompatible with each other due to in silico spin-labelling uncertainties and/or experimental uncertainties, (2) the algorithm might get trapped in local minima or (3) some of the constraints are not linearly independent from each other. The latter reason could arise from the fact that in this case most labels were (understandably) repeatedly used for distance measurements. More linearly independent constraints would have to be introduced to resolve this ambiguity. In the original publication, the authors did this by introducing constraints from orthogonal methods, such as solvent accessibility and cross-linking data.

The mtsslDock approach can also be used to solve the trilateration problem (see above). For this purpose, we used the SBL-1 example (see mtsslTrilaterate above) and imported the mean coordinates of the five spin labels, the
Figure 6. Docking of the cytoplasmic domain of erythrocyte band 3, and ankyrin-R repeats 13–24 [20]. The docking solutions for the ankyrin structure based on 20 PELDOR distances are shown as orange loop models. The erythrocyte band 3 dimer is shown as cartoon model (green and blue). MtsslDock finds two solutions for the problem (solution I and solution II). In the bottom panel, the structure is rotated by 90°.

coordinates of an atom that was used as a surrogate for the position of the LOPTC lipid (termed ‘pseudoatom’) below) and the five experimental interspin distances into mtsslDock. The docking was performed using the default settings. Figure 3(B) (purple spheres) shows that the program places the pseudoatom into the centre of the ‘target’ ellipsoid that was generated by mtsslTrilaterate. Similarly mtsslDock was also used to compute the position of the copper-binding site in EcoRI (see above, [14]). The result of this calculation lies exactly between the two trilateration spheres that were output by mtsslTrilaterate and close to His114, which was identified as the copper-binding site in the original publication (Figure 4, red spheres). Note that the program would output a range of differing solutions spread on an intersection circle if the two triangulation spheres would intersect each other. Thus, this result has to be considered as a special case. These two examples represent a cross-check of both algorithms. The use of mtsslTrilaterate is, however, recommended in such cases since it is faster, calculates statistics and provides more informative output for the trilateration problem.

Conclusion

The mtsslSuite currently contains a set of three applications that allow in silico spin labelling of macromolecules, localising spin centres with respect to reference spin centres or docking macromolecules based on experimental spin-pair distance constraints.

The programs can import spin-labelled locations that were generated with mtsslWizard but do at the same time offer users the opportunity to use input data from other sources (e.g. MMM or PRONOX). We have shown that all the three programs achieve results that are comparable in precision to those obtained by means of other computational strategies. The source code of all the three programs is freely available at (www.pymolwiki.org).

Acknowledgements

We would like to thank Jason Vertrees and Thomas Holder for help concerning PyMOL programming and Damien Farrell for help with the tkinterTable class for mtsslDock. This work has been supported by the SFB813 projects C8 and A6 of the Deutsche Forschungsgemeinschaft (DFG) and the Wellcome Trust (091825/Z/10/Z).

Note added in proof

While this paper went into production, the K1 spin label [42] was added to mtsslWizard upon user request. Supplementary Figure 7 shows a comparison between predicted and experimental distance distributions derived from this spin label.

References

[1] G. Jeschke, Annu. Rev. Phys. Chem. 63, 419 (2012).
[2] G.W. Reginsson and O. Schiemann, Biochem. J. 434, 353 (2011).
[3] O. Schiemann and T.F. Prisner, Q. Rev. Biophys. 40, 1 (2007).
[4] A. Milov, K. Salikhoov, and M. Shirov, Fizika. Tverdogo. Tela. 23, 975 (1981).
[5] G. Jeschke, A. Bender, and H. Paulsen, J. Magn. Reson. 169, 1–12 (2004).
[6] C. Altenbach, T. Marti, H.G. Khorana, and W.L. Hubbell, Science 248, 1088 (1990).
[7] J.P. Klare and H.-J. Steinhoff, Photosyn. Res. 102, 377 (2009).
[8] S.A. Shelke and S.T. Sigurdsson, Eur. J. Org. Chem. 2012, 2291 (2012).
[9] R. Ward and O. Schiemann, Structural Information from Oligonucleotides (Springer, Berlin, 2012).
[10] G. Hageluukken, R. Ward, J.H. Naismith, and O. Schiemann, Appl. Magn. Reson. 42, 377 (2012).
[11] M.M. Hatmal, Y. Li, B.G. Hegde, P.B. Hegde, C.C. Jao, R. Langen, and I.S. Haworth, Biopolymers 97, 35 (2011).
[12] Y. Polylich, E. Bordignon, and G. Jeschke, Phys. Chem. Chem. Phys. 13, 2356 (2011).
[13] B.J. Gaffney, M.D. Bradshaw, S.D. Frausto, F. Wu, J.H. Freed, and P. Borbat, Biophys. J. 103, 2134 (2012).
[14] Z. Yang, M.R. Kurpiewski, M. Ji, J.E. Townsend, P. Mehta, L. Jen-Jacobson, and S. Saxena, Proc. Nat. Acad. Sci. U.S.A. 109, E993–1000 (2012).
[15] C.M. Dupureur, Curr. Opin. Chem. Biol. 12, 250 (2008).
[16] A. Muschielok, J. Andrecka, A. Jawhari, F. Brückner, P. Cramer, and J. Michaelis, Nat. Methods 5, 965 (2008).
[17] P.P. Borbat, H.S. Mchaourab, and J.H. Freed, J. Am. Chem. Soc. 124, 5304 (2002).
[18] G. Jeschke, A. Bender, T. Schweikardt, G. Panek, H. Decker, and H. Paulsen, J. Biol. Chem. 280, 18623–18630 (2005).
[19] D. Hilger, Y. Polyhach, E. Padan, H. Jung, and G. Jeschke, Biophys. J. 93, 3675 (2007).
[20] S. Kim, S. Brandon, Z. Zhou, C.E. Cobb, S.J. Edwards, C.W. Moth, C.S. Parry, J.A. Smith, T.P. Lybrand, E.J. Hustedt, and A.H. Beth, J. Biol. Chem. 286, 20746 (2011).
[21] R. Ward, M. Zoltner, L. Beer, H. El-Mkami, I.R. Henderson, T. Palmer, and D.G. Norman, Structure 17, 1187 (2009).
[22] M. Zoltner, D.G. Norman, P.K. Fyfe, H. El-Mkami, T. Palmer, and W.N. Hunter, Structure 21, 595–603 (2013).
[23] N. Alexander, M. Bortolus, A. Al-Mestarihi, H. Mchaourab, and J. Meiler, Structure 16, 181 (2008).
[24] Z. Guo, D. Cascio, K. Hideg, and W. Hubbell, Protein Sci. 17, 228 (2008).
[25] S.J. Hirst, N. Alexander, H.S. Mchaourab, and J. Meiler, J. Struct. Biol. 173, 506 (2011).
[26] E. Hustedt, R. Stein, L. Sethaphong, S. Brandon, Z. Zhou, and S. DeSensi, Biophys. J. 90, 340 (2006).
[27] Z. Zhou, S.C. DeSensi, R.A. Stein, S. Brandon, M. Dixit, E.J. McArdle, E.M. Warren, H.K. Kroh, L. Song, C.E. Cobb, E.J. Hustedt, and A.H. Beth, Biochemistry. 44, 15115 (2005).
[28] Y. Song, T.J. Meade, A.V. Astashkin, and E.L. Klein, J. Magn. Reson. 210, 59 (2011).
[29] P. Cekan, A.L. Smith, N. Barhate, B.H. Robinson, and S.T. Sigurdsson, Nucleic Acids Res. 36, 5946 (2008).
[30] T.E. Edwards, T.M. Okonogi, and B.H. Robinson, J. Am. Chem. Soc. 123, 1527 (2001).
[31] G. Jeschke, Prog. Nucl. Magn. Reson. Spectrosc. 72, 42–60 (2013).
[32] D. Goldfarb (private communication).
[33] O. Schiemann, P. Cekan, D. Margraf, T.F. Prisner, and S.T. Sigurdsson, Angew. Chem. Int. Edit. 48, 3292 (2009).
[34] G. Jeschke, V. Chechik, P. Ionita, A. Godt, H. Zimmermann, J. Banham, C. Timmel, D. Hilger, and H. Jung, Appl. Magn. Reson. 30, 473 (2006).
[35] B.P. Flannery, W.H. Press, and S.A. Teukolsky, Numerical Recipes in C. (Cambridge University Press, New York, 1992).
[36] W. Minor, J. Steczko, O.B. Stec, and Z. Otwinowski, Biochemistry 35, 10687 (1996).
[37] M. Srinivas and L.M. Patnaik, Computer 27, 17 (1994).
[38] E.J. Gardiner, P. Willett, and P.J. Artymiuk, Proteins 44, 44 (2001).
[39] C.R. Kissinger, D.K. Gehlhaar, and D.B. Fogel, Acta Crystallogr. D. Biol. Crystallogr. 55, 484 (1999).
[40] G. Hagelueken, L. Wiehlmann, T.M. Adams, H. Kolmar, D.W. Heinz, B. Tümmler, and W-D. Schubert, Proc. Nat. Acad. Sci. U.S.A. 104, 12276 (2007).
[41] S. Lyskov and J.J. Gray, Nucleic Acids Res. 36, W233 (2008).
[42] M.R. Fleissner, E.M. Brustad, T. Kállai, C. Altenbach, D. Cascio, F.B. Peters, K. Hideg, S. Peuker, P.G. Schultz, and W.L. Hubbell, Proc. Nat. Acad. Sci. U.S.A. 106, 21637–21642 (2009).