NEW UNRES FORCE FIELD PACKAGE WITH FORTRAN 90

EMILIA A. LUBECKA[1,2,3] AND ADAM LIWO[3]

[1] *Institute of Informatics, University of Gdansk*
Wita Stwosza 57, 80-308 Gdansk, Poland

[2] *Academic Computer Center in Gdansk TASK*
Narutowicza 11/12, 80-233 Gdansk, Poland

[3] *Faculty of Chemistry, University of Gdansk*
Wita Stwosza 63, 80-308 Gdansk, Poland

(received: 13 September 2016; revised: 12 October 2016;
accepted: 17 October 2016; published online: 28 October 2016)

Abstract: UNRES is a coarse-grained model of polypeptide chains. Until now, each version of UNRES (UNRESPACK v. 3.2 and earlier ones) has been written in Fortran 77. Due to the fact that Fortran 77 enables us to use only static arrays, the Fortran 77 version has significant memory problems, and consequently, UNRESPACK has had to be split into many programs. Our recent work was focused on creating a new UNRES package with Fortran 90 (UNRESPACK v. 4.0), based on the previous Fortran 77 versions. Fortran 90 provides dynamic memory allocation, user defined data types, and structuring the code into modules which encompass subroutines, functions, and variables. Moreover, Fortran 90 adds internal functions and subroutines, providing greater flexibility. The whole code of UNRES with Fortran 90 has been restructured, so that it now consists of modules that can be assembled to create the main simulation program and companion programs. This approach enabled us to eliminate the redundancy of the code, while keeping all functions of the package.

Keywords: UNRES, coarse-grained model, Fortran 90, modeling of protein structures
DOI: https://doi.org/10.17466/tq2016/20.4/n

1. Introduction

The united residue (UNRES) model that is being developed in our laboratory is designed to perform simulations of the protein structure and dynamics. In the UNRES model, each amino-acid residue is reduced to two interaction sites: a united peptide group and a united side chain. The α-carbon (Cα) atoms are also present; however, they serve to define the polypeptide backbone geometry and they are not interaction sites (Figure 1) [1, 2]. The UNRES force field that

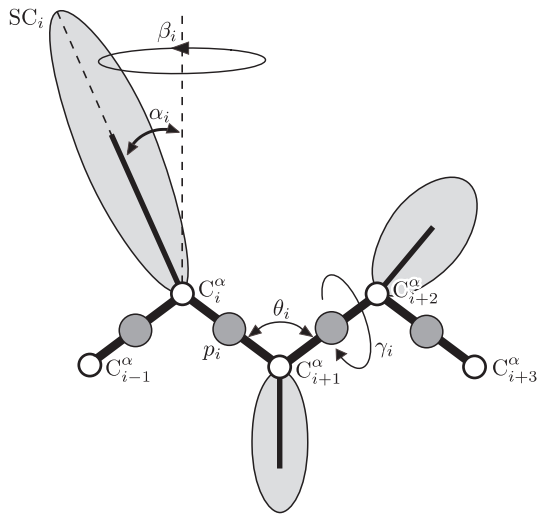

Figure 1. UNRES model of polypeptide chains. The interaction sites are united peptide groups located between the consecutive α-carbon atoms and united side chains attached to the α-carbon atoms; the backbone geometry of the simplified polypeptide chain is defined by the $C^\alpha\cdots C^\alpha\cdots C^\alpha$ virtual-bond angles θ (θ_i has the vertex at C^α_i) and the $C^\alpha\cdots C^\alpha\cdots C^\alpha\cdots C^\alpha$ virtual-bond-dihedral angles γ (γ_i has the axis passing trough C^α_i and C^α_{i+1}); the local geometry of the i-th side-chain center is defined by the spherical angles α_i (the angle between the bisector of the respective angle θ_i and the $C^\alpha_i\cdots SC_i$ vector) and β_i (the angle of counter-clockwise rotation of the $C^\alpha_i\cdots SC_i$ vector about the bisector from the $C^\alpha_{i-1}\cdots C^\alpha_i\cdots C^\alpha_{i+1}$ plane, starting from C^α_{i-1})

corresponds to this model is a physics-based force field that has been derived as a restricted free energy (RFE) function, which corresponds to averaging the energy over the degrees of freedom that are neglected in the united-residue model [3]. The ability of the UNRES force field to predict the protein structure has been continuously assessed since 1998 in the Critical Assessment of Techniques for Protein Structure Prediction (CASP) with very good results [4–6]. In particular, UNRES can lead to exceptionally good results when correct domain packing is an issue (*e.g.* for CASP12 target T0663) [6].

The effective energy function in UNRES is expressed by Equation (1).

$$\begin{aligned} U =& w_{\mathrm{SC}}\sum_{i<j}U_{\mathrm{SC}_i\mathrm{SC}_j}+w_{\mathrm{SC}p}\sum_{i\neq j}U_{\mathrm{SC}_i p_j}+w_{pp}^{\mathrm{VDW}}\sum_{i<j-1}U_{p_i p_j}^{\mathrm{VDW}}+w_{pp}^{\mathrm{el}}f_2(T)\sum_{i<j-1}U_{p_i p_j}^{\mathrm{el}}\\ &+w_{\mathrm{tor}}f_2(T)\sum_i U_{\mathrm{tor}}(\gamma_i)+w_{\mathrm{tord}}f_3(T)\sum_i U_{\mathrm{tord}}(\gamma_i,\gamma_{i+1})\\ &+w_b\sum_i U_b(\theta_i)+w_{\mathrm{rot}}\sum_i U_{\mathrm{rot}}(\alpha_{\mathrm{SC}_i},\beta_{\mathrm{SC}_i})+w_{\mathrm{bond}}\sum_i U_{\mathrm{bond}}(d_i)\\ &+w_{\mathrm{corr}}^{(3)}f_3(T)U_{\mathrm{corr}}^{(3)}+w_{\mathrm{corr}}^{(4)}f_4(T)U_{\mathrm{corr}}^{(4)}+w_{\mathrm{turn}}^{(3)}f_3(T)U_{\mathrm{turn}}^{(3)}+w_{\mathrm{turn}}^{(4)}f_4(T)U_{\mathrm{turn}}^{(4)}\\ &+w_{ss\mathrm{bond}}\sum_{nss}U_{ss\mathrm{bond}}(d_{ss})+w_{\mathrm{SC\text{-}corr}}f_2(T)\sum_{m=1}^{3}\sum_i U_{\mathrm{SC\text{-}corr}}\left(\tau_i^{(m)}\right) \end{aligned} \tag{1}$$

where the U's are energy terms, θ_i is the backbone virtual-bond angle between three consecutive C^α atoms, γ_i is the backbone virtual-bond-dihedral angle (defined by four consecutive C^αs), α_i and β_i are the angles defining the location of the center of the united side chain of residue i (Figure 1) with respect to the plane defined by the C^α_{i-1}, C^α_i and C^α_{i+1} atoms, d_i is the length of the ith virtual bond, which is either a $C^\alpha \cdots C^\alpha$ virtual bond or a $C^\alpha \cdots$SC virtual bond, d_{ss} is the distance between the side chains of two cystine residues forming a disulfide bond. The terms $U^{(m)}_{\mathrm{corr}}$ represent the correlation or multibody contributions from the coupling between backbone-local and backbone-electrostatic interactions, and the terms $U^{(m)}_{\mathrm{turn}}$ are correlation contributions involving m consecutive peptide groups. U_{bondss} is the energy of disulfide-bond formation and nss is the number of disulfide bonds. The $U_{\mathrm{SC-corr}}$ terms are new physics-based side-chain backbone correlation potentials [7].

The UNRES force field was developed for simulation of the protein structure and dynamics [1, 2]. Unlike most of the other coarse-grained force fields, UNRES has been derived [8, 9] and parametrized [8, 10, 11] as a potential of the mean force (PMF) of polypeptide chains immersed in water. UNRES was initially used in the prediction of a single-chain protein structure as a global minimum in the effective potential-energy surface [4, 5], with a very efficient genetic-type algorithm (the Conformational Space Annealing (CSA) method) [12], to locate the global minimum. Then, UNRES was extended to run coarse-grained molecular dynamics (Langevin dynamics) [13, 14] and to study oligomeric proteins [15, 16]. Moreover, five techniques of multicanonical simulations [17] in UNRES/MD were included [18, 19] to determine the thermodynamic characteristics of the UNRES force field [20] for efficient sampling at various temperatures, namely, replica-exchange molecular dynamics (REMD), multiplexed replica-exchange molecular dynamics (MREMD), multicanonical molecular dynamics (MUCAMD), as well as replica-exchange multicanonical (REMUCA) and multicanonical replica-exchange (MUCAREM) molecular dynamics. Of those, the MREMD method has turned out to be the most efficient.

Most of the coarse-grained force fields that are capable of predicting the structures of proteins are knowledge-based and, therefore, do not handle proteins that contain D-amino acid residues because of an insufficient number of D-amino-acid residues in protein structural databases to derive the respective statistical potentials. By contrast, UNRES, as a physics-based force field, has been easily extended to include D-amino-acid residues [21, 22]. Moreover, the *trans-cis* isomerization of peptide groups has been also introduced into UNRES [23]. Another extension allows dynamic formation and breaking of disulfide bonds during the simulations [24, 25].

Apart from protein-structure prediction and investigation of protein folding, UNRES has been also applied with success to study biological problems such as, *e.g.*, amyloid formation [26, 27], signaling [28], and chaperone dynamics [29].

2. Implementation

The current version of UNRES (UNRESPACK v. 3.2) has been written in Fortran 77 – the Fortran standard since 1978 [30] (and the most popular one). Due to the fact that only static arrays can be used in Fortran 77, memory problems often occur. Therefore, UNRESPACK has had to be split up into many programs, and each of these programs implements a different UNRES functionality, resulting in code redundancy. Many functions, subroutines and variables have had to be repeated several times. Moreover, the nature of Fortran 77 (only static arrays available, poor flexibility etc.), makes the source code unclear, with complicated dependencies.

The source code in the currently available UNRES package has the following directory structure (see the UNRES project web side: http://unres.pl for details):

- unres (UNRES source codes; various versions)
  - src_MIN (energy evaluation and minimization only) – 27,362 lines of code
  - src_CSA (all functions except MD, includes CSA) – 45,898
  - src_MD (all functions except CSA, includes MD, single chains) – 60,962
  - src_MD-M (all functions except CSA, includes MD, oligomeric proteins) – 105,524
- wham (weighted histogram analysis method source code)
  - src (single chains) – 33,804
  - src-M (treatment of oligomeric proteins) – 31,188
- cluster (cluster-analysis source code)
  - clust-unres
    - src (clustering the conformations from energy minimization/CSA/canonical MD searches) – 4,527
  - clust-wham (clustering the conformations from MREMD searches post-processed with WHAM)
    - src (for single-chain proteins) – 18,152
    - src-M (for oligomeric proteins) – 16,505

The UNRES package contains the source codes of the main simulation program and companion programs: WHAM, CLUSTER, and Cartesian-coordinate-format converters (xdrf2pdb, xdrf2pdb-m, and xdrf2x). The conformations produced by UNRES are used as inputs to the companion programs. WHAM (weighted histogram analysis method) processes the results of REMD or MREMD simulations with UNRES to compute temperature profiles of ensemble averages and probabilities of the obtained conformations to occur at particular temperatures [20, 31, 32]. CLUSTER performs cluster analysis of conformations that are obtained directly from UNRES runs (CSA, MCM, MD, (M)REMD, multiple-conformation energy minimization). The program incorporates the hierarchical-clustering subroutine, hc.f written by G. Murtagh. The subroutine contains seven methods of hierarchical clustering [33, 34]. xdrf2pdb and xdrf2pdb-m convert the compressed coordinate

files from molecular dynamics into the PDB format, for single (MD) and multiple (MREMD) trajectory capacity runs, respectively. In turn, xdrf2x converts the plain Cartesian coordinate files into the PDB format, and PHOENIX converts UNRES conformations into all-atom conformations. Developers can also use ZSCORE, for force field optimization.

In our current work, to create a new UNRES package, we selected Fortran 90 [35], which is the major revision to Fortran 77. We decided to use most of all dynamic memory allocation and organize the code into modules encompassing subroutines, functions, and variables, and also add internal functions and subroutines providing a greater flexibility [35]. These features of Fortran 90 enabled us to obtain one version of the UNRES code with all UNRES functionalities.

The basis of the new UNRES package were the source-code files from the following directories from UNRESPACK v. 3.2 (see www.unres.pl):

- the main simulation program: src_MD-M directory, 105,524 lines of code, 99 source-code and 52 COMMON block files
- the WHAM companion program: src-M directory, 31,188 lines of code, 54 source-code and 37 COMMON block files
- the CLUSTER companion program: unres/src directory, 4,527 lines of code, 22 source-code and 20 COMMON block files

The components of the source code listed above were adapted to Fortran 90 and assembled into the UNRESPACK v. 4.0 version of the package. The code takes advantage of dynamic memory allocation, this making memory usage more efficient, especially for 2-,3- and 4-dimensional arrays. The whole code was restructured so that it now consists of modules that can be assembled to create the main simulation program. This restructuring was carried out using multiple modules but without any additional crucial function or subroutine. Consequently, UNRESPACK v. 4.0 is a unified version of the UNRES code with all functionalities. This new package also contains all companion programs. Moreover, the companion programs (like CLUSTER, WHAM, etc.) share the modules of the main program (Figure 2). This approach enabled us to reduce the redundancy in the code to a great extent.

The source code in the new UNRES package with Fortran 90 has the following directory structure:

- unres (UNRES source codes with all functions – all functionalities) – 68,126 lines of code
  - data (modules containing arrays declarations that correspond to COMMON block files) – 1,080
- wham (weighted analysis method source code) – 9,498
- cluster (cluster analysis source code) – 2,721

3. Tests of UNRESPACK v. 4.0

A standard set of tests were run which included single-point energy evaluation, analytical gradient check, energy minimization, canonical molecular dy-

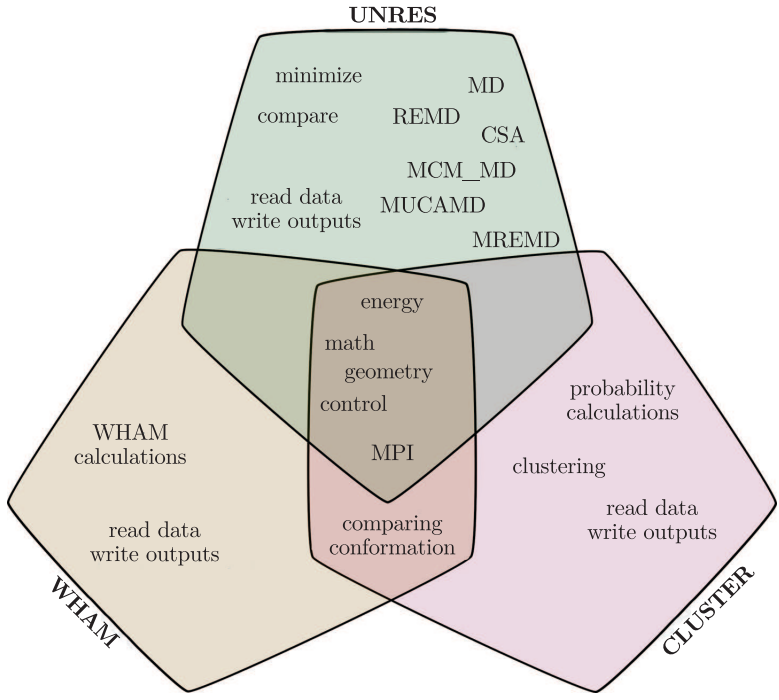

Figure 2. The blocks/modules dependence in the source code of the new UNRES package with Fortran 90

namics, and replica-exchange molecular dynamics. As an example of production calculations, MREMD calculations with the tryptophan cage (PDB code: 1L2Y) are presented. The simulations were started from a fully-extended polypeptide chain and no external information was included in the simulation process. 32 trajectories were run at 16 temperatures (2 trajectories per temperature). The temperatures ranged from 250 K to 400 K with a 10 K increment. Fifty million steps with a length of 4.89 fs [36] (0.24 μs of the total UNRES simulation time corresponding to about 0.24 ms because of the UNRES time-scale extension resulting, in turn, from averaging out the fast degrees of freedom [13, 37]) were run. The Berendsen thermostat [38] with the coupling parameter $\tau = 48.9$ fs was used to maintain constant temperature. The variable time step (VTS) algorithm [37] was used to integrate the equations of motion.

The test MREMD simulations confirmed proper functioning of all UNRES programs, including the main simulation program and the companion programs (WHAM, CLUSTER, and Cartesian coordinate format converters). MREMD simulation results obtained with the previous UNRESPACK v. 3.2.1 and the new UNRESPACK v. 4.0 were in good agreement (Figure 3) as well as in good agreement with with the experimental data [39] (Figure 4). It should be noted that, due to the fact that MREMD simulations are stochastic, perfect agreement cannot be achieved. The mean structures of the simulated conformational families of the tryptophan cage calculated with UNRESPACK v. 4.0 have a root-mean-square

deviation (RMSD) over all non-hydrogen atoms of 1.34, 1.61 and 1.83 Å from those calculated with UNRESPACK v. 3.2.1., for clusters 1, 2 and 3, respectively. The calculated representative structures of the most probable cluster have RMSD from the experimental structures of 3.45 and 3.26 Å for UNRESPACK v. 3.2.1 and UNRESPACK v. 4.0, respectively.

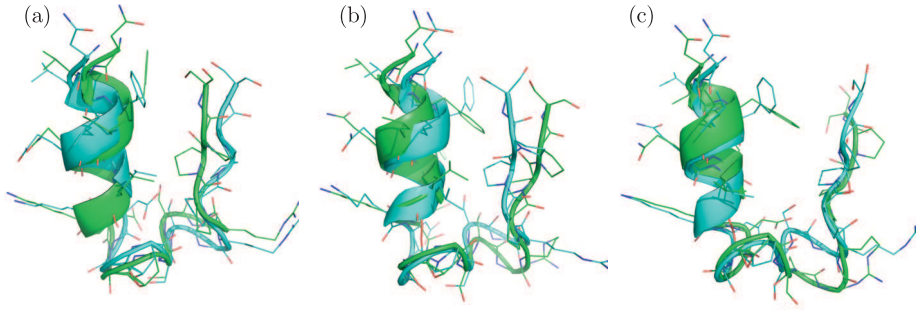

Figure 3. Cartoon representation of the superposition of the non-hydrogen atoms of the average structures of the most probable clusters of conformations of the tryptophan cage obtained in MREMD simulations with the Fortran 77 version of UNRES (cyan) on those of the conformation obtained with the Fortran 90 version of UNRES (green); (a) clusters1, (b) clusters2, (c) clusters3; the RMSDs are 1.34 Å for panel (a), 1.61 Å for panel (b) and 1.83 Å for panel (c), respectively

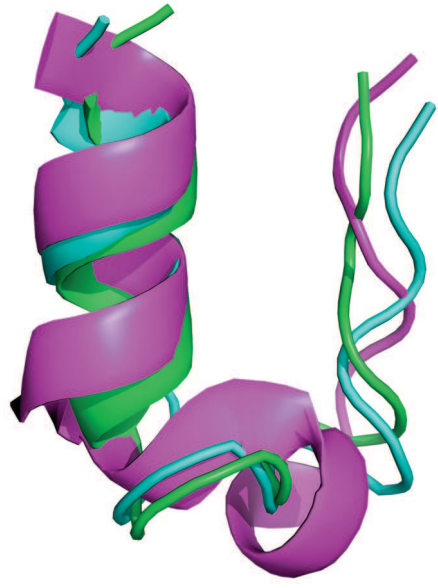

Figure 4. Cartoon representation of superposition of the non-hydrogen atoms of the representative structures of the most probable cluster of the conformations of the tryptophan cage obtained in MREMD simulations with the Fortran 77 version of UNRES (cyan) and the Fortran 90 version of UNRES (green) on those of the experimental structure (magenta); the RMSDs from the experimental structure are 3.45 Å for the Fortran 77 and 3.26 Å for Fortran 90 version of UNRES, respectively

4. Conclusions and Outlook

The new version of the UNRES package, UNRESPACK v. 4.0, reported in this paper has greatly improved code logic. Subroutines, functions, and data blocks are organized into modules, each performing well-defined tasks (*e.g.*, geometry processing, energy and force calculation, *etc.*), which makes the dependencies easy to follow. The new UNRES package provides the complete functionality of the previous version. Moreover, memory occupation has been optimized owing to dynamic memory allocation.

The modularization of the code provides means to extend the UNRES package to incorporate the UNRES-like coarse-grained models of nucleic acids (NARES-2P) and polysaccharides (SUGRES-1P), which are the components of the Unified Coarse Grained model developed in our laboratory [2].

Acknowledgements

This work was supported by the PLGrid NG project ("New generation domain-specific services in the PL-Grid Infrastructure for Polish Science", 2014–2015), which is co-funded by the European Regional Development Fund as part of the Innovative Economy program. Partial funding was also provided by the University of Gdansk (D498).

References

[1] Liwo A, Czaplewski C, Ołdziej S, Rojas A V, Kaźmierkiewicz R, Makowski M, Murarka R K, Scheraga H A 2008 *Simulation of protein structure and dynamics with the coarse-grained UNRES force field*, in Coarse-Graining of Condensed Phase and Biomolecular Systems (ed. Voth G), CRC Press 1391
[2] Liwo A *et al.* 2014 *J. Mol. Model.* **20** (8) 2306
[3] Czaplewski C, Liwo A, Makowski M, Ołdziej S, Scheraga H A 2010 *Coarse-grained models of proteins: theory and applications*, in Multiscale approaches to protein modeling (ed. Koliński A), Springer-Verlag, Berlin, **3** 35
[4] Liwo A, Lee J, Ripoll D R, Pillardy J, Scheraga H A 1999 *Proc. Natl. Acad. Sci., U.S.A.* **96** 5482
[5] Ołdziej S *et al.* 2005 *Proc. Natl. Acad. Sci. U.S.A.* **102** 7547
[6] He Y *et al.* 2013 *Proc. Natl. Acad. Sci. U.S.A.* **110** (37) 14936
[7] Krupa P, Sieradzan A K, Rackovsky S, Baranowski M, Ołdziej S, Scheraga H A, Liwo A, Czaplewski C 2013 *J. Chem. Theory Comput.* **9** (10) 4620
[8] Liwo A, Czaplewski C, Pillardy J, Scheraga H A 2001 *J. Chem. Phys.* **115** 2323
[9] Liwo A *et al.* 1998 *J. Comput. Chem.* **19** 259
[10] Liwo A, Ołdziej S, Czaplewski C, Kozłowska U, Scheraga H A 2004 *J. Phys. Chem. B* **108** 9421
[11] Kozłowska U, Maisuradze G G, Liwo A, Scheraga H A 2010 *J. Comput. Chem.* **31** 1154
[12] Lee J, Scheraga H A 1999 *Int. J. Quant. Chem.* **75** 255
[13] Liwo A, Khalili M, Scheraga H A 2005 *Proc. Natl. Acad. Sci. U.S.A.* **102** 2362
[14] Rakowski F, Grochowski P, Lesyng B, Liwo A, Scheraga H A 2006 *J. Chem. Phys.* **125** 204107
[15] Saunders J A, Scheraga H A 2003 *Biopolymers* **68** (3) 300
[16] Rojas A V, Liwo A, Scheraga H A 2007 *J. Phys. Chem. B* **111** (1) 293
[17] Mitsutake A, Sugita Y, Okamoto Y 2003 *J. Chem. Phys.* **118** 6664
[18] Nanias M, Czaplewski C, Scheraga H A 2006 *J. Chem. Theor. Comput.* **2** 513

[19] Czaplewski C, Kalinowski S, Liwo A, Scheraga H A 2009 *J. Chem. Theor. Comput.* **5** 627
[20] Ołdziej S, Łągiewka J, Liwo A, Czaplewski C, Chinchio M, Nanias M, Scheraga H A 2004 *J. Phys. Chem. B* **108** 16950
[21] Sieradzan A K, Hansmann U H E, Scheraga H A, Liwo A 2012 *J. Chem. Theory Comput.* **8** (11) 4746
[22] Sieradzan A K, Niadzvedtski A, Scheraga H A, Liwo A 2014 *J. Chem. Theory Comput.* **10** 2194
[23] Sieradzan A K, Scheraga H A, Liwo A 2012 *J. Chem. Theor. Comput.* **8** (4) 1334
[24] Czaplewski C, Oldziej S, Liwo A, Scheraga H A 2004 *PEDS* **17** 29
[25] Chinchio M, Czaplewski C, Liwo A, Oldziej S, Scheraga H A 2007 *J. Chem. Theor. Comput.* **3** 1236
[26] Rojas A, Liwo A, Browne D, Scheraga H A 2010 *J. Mol. Biol.* **404** 537
[27] Rojas A, Liwo A, Scheraga H A 2011 *J. Phys. Chem. B* **115** 12978
[28] He Y, Liwo A, Weinstein H, Scheraga H A 2011 *J. Mol. Biol.* **405** 298
[29] Golas E I, Maisuradze G G, Senet P, Ołdziej S, Czaplewski C, Scheraga H A, Liwo A 2012 *J. Chem. Theor. Comput.* **8** 1334
[30] American National Standards Institute 1978 *Ansi x3.9-1978. American National Standard – Programming Language FORTRAN*, *ISO 1539-1980*
[31] Kumar S, Bouzida D, Swendsen R H, Kollman P A, Rosenberg J M 1992 *J. Comput. Chem.* **13** 1011
[32] Liwo A, Khalili M, Czaplewski C, Kalinowski S, Ołdziej S, Wachucik K, Scheraga H A 2007 *J. Phys. Chem. B* **111** 260
[33] Murtagh F 1985 *Multidimensional clustering algorithms*, Physica-Verlag
[34] Murtagh F, Heck A 1987 *Multivariate data analysis*, Kluwer Academic Publishers
[35] American National Standards Institute 1991 *Ansi x3.198-1992. American National Standard – Programming Language Fortran Extended*, *ISO/IEC 1539:1991*
[36] Khalili M, Liwo A, Rakowski F, Grochowski P, Scheraga H A 2005 *J. Phys. Chem. B* **109** 13785
[37] Khalili M, Liwo A, Jagielska A, Scheraga H A 2005 *J. Phys. Chem. B* **109** 13798
[38] Berendsen H J C, Postma J P M, van Gunsteren W F, DiNola A, Haak J R 1984 *J. Chem. Phys.* **81** 3684
[39] Neidigh J W, Fesinmeyer R M, Andersen N H 2002 *Nat. Struct. Biol.* **9** 425