P-found: The Protein Folding and Unfolding Simulation Repository

Cândida G. Silva*, Vitaliy Ostropytskyy†, Nuno Loureiro-Ferreira*,
Daniel Berrar†, Martin Swain†, Werner Dubitzky† and Rui M. M. Brito*
*Chemistry Department, Faculty of Science and Technology, and Center for Neuroscience and Cell Biology
University of Coimbra, 3004-535 Coimbra, Portugal
Email: brito@ci.uc.pt
†School of Biomedical Sciences, University of Ulster
Cromore Road, BT52 1SA Coleraine, Northern Ireland
Email: w.dubitzky@ulster.ac.uk

***Abstract*— One of the central challenges in structural molecular biology today is the protein folding problem, *i.e.* the acquisition of the 3D structure of a protein from its linear sequence of amino-acids. Different computational approaches to study protein folding and protein unfolding have recently become common tools available to the researcher. However, due to the lack of appropriate infrastructures, it is very difficult to directly compare simulations performed by different groups, with different methods, in different experimental conditions or for different proteins. Thus, we set out to create a public data repository with the goal of addressing the problem of comparison, analysis and sharing of information and data on protein folding and protein unfolding simulations. The P-found system for protein folding and protein unfolding simulations is presented. At the moment, the data repository allows uploading of molecular dynamics (MD) protein folding and unfolding simulations, calculates and stores several time series with the variation over time of pre-defined molecular properties, and allows searching and downloading of these data. In the near future, simulations performed by other than MD methods may be uploaded, and data mining techniques for analysis and comparison of multiple simulations will be implemented. The home page for the P-found system is accessible at `http://www.p-found.org`.**

I. Introduction

One of the unsolved paradigms in molecular biology is the protein folding problem, *i.e.* the acquisition of the functional three-dimensional structure of a protein from its linear sequence of amino-acids, ultimately determined by the sequence of bases in a gene. In recent years, with the identification of several diseases as protein folding disorders and the advent of many genomics projects, protein folding has become a central issue in molecular sciences research. The detailed understanding of the forces and molecular mechanisms driving the protein folding process *in vivo* and *in vitro* is essential in areas as diverse as therapeutics of neurodegenerative diseases or bio-catalysis in organic solvents.

Many experimental and computational approaches have been used to tackle the protein folding problem (for reviews, see for example: [1]–[3]). However, more recently and fuelled by the ever increasing computational power available to a wider range of scientists, the computational approaches to study protein folding and unfolding have gained increasing importance [2], [4], [5].

Although computational protein folding and unfolding simulations are expensive (in time and resources), today these simulations are not generally available outside the groups that perform them, preventing detailed comparisons between simulations and hampering the development of new analysis tools. Thus, it is becoming apparent that the creation of a platform where protein folding and unfolding computer simulations may be compared and analyzed would be of the utmost importance in the progress of the field [6].

To address the goal of comparison, analysis and sharing of information and data on protein folding and unfolding simulations, the project P-found has been developed. The overall aim of this project is to create a public data and information repository that will enable researchers around the globe to share, analyze and compare protein folding and unfolding simulation data:

- from different simulation methods, such as molecular dynamics (MD), methods based on Monte Carlo techniques, structure-based force fields, etc;
- using simplified or all-atom protein representations;
- implicit or explicit solvent descriptions;
- in aqueous or organic solution, with or without co-solutes;
- for different proteins (wild type *vs.* mutant; different structural classes or different topologies);
- mimicking different experimental conditions (different temperature, pressure, pH, ionic strength, etc).

The main functional requirements of the P-found system are to facilitate (i) the sharing of simulation data (raw simulation data, calculated molecular property data, provenance and metadata), (ii) the analysis and data mining of molecular property data, and (iii) the dynamic deployment and application of proprietary programs for calculating molecular properties and for analyzing molecular property data. At the present stage of project development, sharing of the data arising from protein folding and unfolding simulations means that users around the globe can contribute and store (*i.e.* upload) data

1–4244–0623–4/06/$20.00

from simulation experiments, and they can search, select and access data from the repository. Additionally, the repository calculates and stores a series of molecular properties from each submitted simulation, which may be visualized by any user and accessed (*i.e.* downloaded) by registered users. A second important part of the project is to develop suitable data mining techniques to analyze and compare large sets of multiple protein folding and unfolding simulations stored in the repository. This will allow users to apply a range of different data processing and analysis methods (classification, clustering, time-series modeling, etc.) to the data in order to address their scientific questions. A third aspect of the repository is to allow the application of different property calculation methods. Currently, Visual Molecular Dynamics (VMD) [7], a molecular visualization program for displaying, animating, and analyzing large biomolecular systems using 3-D graphics and built-in scripting, has been integrated in the repository. It is envisaged that future versions will facilitate the dynamic deployment and application of proprietary methods.

Should the P-found system be widely accepted, we anticipate that different groups would desire their own methods and programs to be applied to the data, both to the raw trajectory data (to calculate molecular properties) and to the molecular property data (to analyze the data to answer the underlying scientific questions). The calculation of molecular properties involves potentially very large volumes of data and the analysis of property data involves potentially highly compute-intensive procedures. To cope with these requirements, we anticipate a future development which would leverage grid technology to facilitate the deployment and execution of proprietary programs on powerful (CPU, network connection) application servers near the P-found system. We have already investigated these technologies in the context of the OpenMolGRID [8] and DataMiningGrid [9] projects.

In this paper, we present the architecture and functionality of the first version of this protein folding and unfolding simulation data warehouse – the P-found system. In this version, the system is prepared to accept data and metadata from simulations using molecular dynamics (MD) methods, and to calculate global molecular properties such as root mean square deviation (RMSD), root mean square fluctuation (RMSF), gyration radius (R_g), secondary structure, native contacts and solvent accessible surface area (SASA).

We believe that the development and implementation of a publicly available repository of protein folding and unfolding simulations will allow a more efficient dissemination of the scientific results provided by these simulations. Additionally, and even more importantly, the P-found system will allow the development of effective analysis tools to compare and characterize protein simulations, which may have a very positive impact on the understanding of the molecular mechanisms of protein folding, misfolding and aggregation, in protein structure prediction, and even in areas such as *de novo* protein design in aqueous or organic solution.

Ultimately, the P-found resource will contain large amounts of complex data describing the dynamic behavior of folding and unfolding proteins. Effective and efficient analysis and interpretation of this data is a considerable challenge. We therefore anticipate that a wide range of sophisticated techniques from artificial and computational intelligence, pattern recognition and machine learning will be needed to tackle the analysis and interpretation problem. We have previously explored association mining [10] and are currently exploring the deployment and application of other machine learning techniques. In order to facilitate the development of AI and related techniques for folding/unfolding simulation data, a first pre-requisite is to provide a suitable date warehouse which will facilitate the sharing and pooling of these data. This work mainly focuses on this first comprehensive but necessary step and it is hoped that the AI and related communities will be keen to try their methodologies on the P-found resource.

II. Data Warehousing of Protein Folding and Unfolding Simulations

The main goal of the P-found system is to provide a global repository for sharing and analysis of protein folding and unfolding simulation data. To realize this aim and to facilitate a practical evolution of the system, we have decided to develop the following key system components (see Fig. 1).

First, a centralized data storage (drum symbol labeled Data Warehouse in Fig. 1) which stores

- *Simulation raw data.* These data record the atomic positions of all atoms in the protein along the simulated trajectory;
- *Derived molecular property data.* These data represent different local (amino-acid-specific) and global (entire protein) properties of the folding/unfolding protein. The properties are computed from the simulation raw data by suitable property calculation programs.
- *Provenance data.* These data record critical parameters of the processes, tools and other aspect which led to the creation of the simulation raw data and derived molecular property data.
- *Metadata.* Metadata is used to convey the content and structure of the repository to users so that they can efficiently navigate and use P-found.

Second, a component (box labeled Property Calculation in Fig. 1) which will compute molecular properties from the raw simulation data based on commonly used calculation methods. Third, a set of commonly used data analysis components (box labeled Analysis in Fig. 1) for analyzing the molecular property data in the warehouse. Fourth, a web portal (box labeled Web Portal in Fig. 1) providing users with simple tools for uploading, downloading and analyzing data. Fifth, grid-enabled mechanisms for dynamically deploying and applying proprietary property computation and data analysis programs (box with dotted line at the bottom-right corner of the diagram in Fig. 1). The dynamic deployment and application of property calculation and data analysis becomes necessary for two reasons: first, if many users were to download raw trajectory data frequently to apply their proprietary property calculation programs locally, a considerable bottleneck would

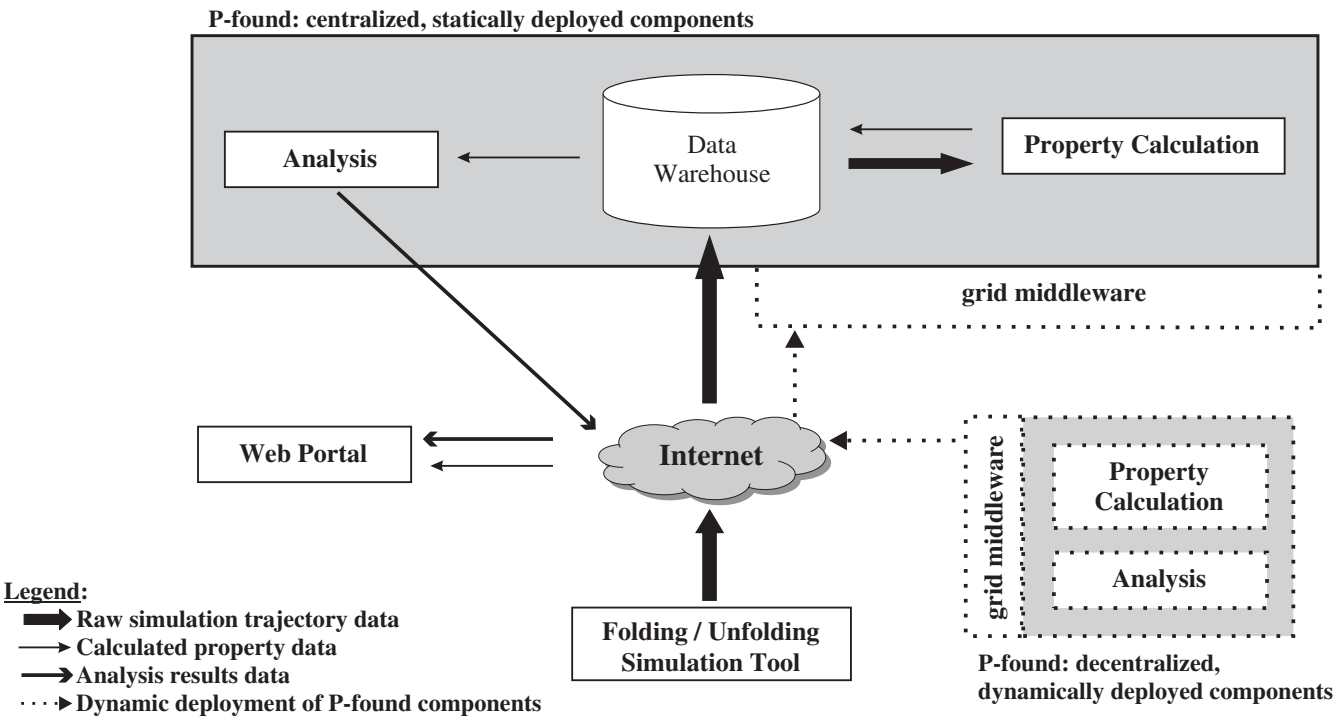

Fig. 1. Overview of the high-level system architecture of P-found. The solid-lined arrows represent the flow of some of the data (raw simulation, derived property, and analysis result data) in the system. The thick arrow indicates large data volume. The dotted arrows indicate the dynamic deployment and application of proprietary programs which may also be controlled via the web portal.

arise because of the significant data volume of trajectory data; second, many users who would like to apply their proprietary data analysis programs may not have the computer power to efficiently run these programs locally. A grid-enabled dynamic deployment and application will allow these users to run their programs elsewhere. The grid-enabled aspects of the P-found system would be restricted to users joining the P-found Grid. Many aspects of the grid-enabled components of the system are based on investigations reported in [12] and currently not fully implemented.

A. System Requirements

The creation of a public repository of protein folding and unfolding simulation data allows the community to mutually produce folding and unfolding trajectories of a wide variety of proteins, using different methods, under a large range of experimental conditions and compare and contrast these data. The data generated in protein folding and unfolding simulations are massive. For example, for just one MD unfolding simulation of 8 ns on a transthyretin monomer [11], a protein with 127 amino-acid residues, the data volume of the trajectory binary file capturing the coordinates of all involved atoms (protein and solvent explicitly represented) is in the order of 4 GB. When capturing just the information on the protein, the data volume is in the order of 180 MB. If multiple simulations in the same or different experimental conditions are required, the data volume increases proportionally. Sharing such data and facilitating their analysis among globally dispersed research groups is a considerable challenge.

The general system architecture of the P-found system is depicted in Fig. 1. The data warehouse component takes a central role in this architecture. Generally, a data warehouse is used to integrate data from underlying sources in a way that is readily processed by data mining methods [13]. In the P-found system the data warehouse provides (i) storage, access, and retrieval functionalities for the raw simulation data, (ii) several different 'views' of the same 'raw' data to facilitate fast access to this data, (iii) several different pre-processed versions of the 'raw' data, and (iv) metadata allowing users to navigate and understand the content and structure of the data warehouse and its different 'views'. Additionally, associated to each simulation, provenance data and molecular property data will also be stored in the data warehouse. Thus, four main types of data are stored in the data warehouse: raw simulation trajectory data, calculated molecular property data, provenance data and metadata. Molecular property data is computed on a compute server operating near (short latency, large bandwidth) the data warehouse. In the current implementation a commonly used program, VMD, is implemented on the compute server to calculate molecular properties which are then stored in the data warehouse. As indicated above, future grid-enabled versions of the system will facilitate a dynamic deployment and application of proprietary property calculation programs. In this case, the result of these calculations will be fed back directly to the user and not stored in the warehouse.

At the present stage, a *web portal* to the data warehouse and

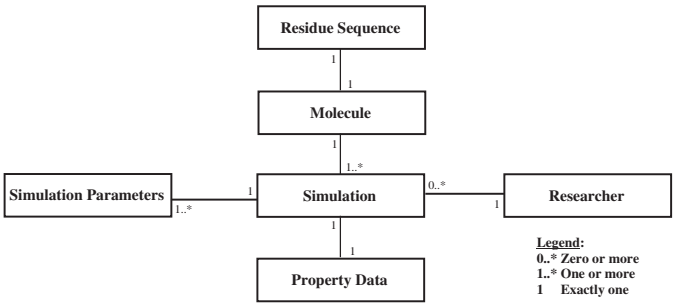

Fig. 2. The conceptual data model of the warehouse.

its analysis components provides global access to a publicly shared repository for storage and access to protein folding and unfolding simulation data, provenance data and molecular property data. Bulk download of simulation raw data will initially be restricted for performance reasons and authorship protection. However, these downloads will be allowed in a context of scientific collaborations, and with the agreement of all parties involved. In the short run, the Data Repository will provide methods that implement data mining algorithms to be run on user-selected subsets of the data warehouse.

B. Conceptual Data Model

The conceptual data model of the data warehouse describes on an abstract level the conceptual entities and their relationships. It is independent of the actual database model. The model presented here (Fig. 2) has evolved from a previous version presented in [12]. We can distinguish between three levels of data:

1) Simulation raw data (trajectory and topology files),
2) Simulation parameters data, and
3) Molecular properties data.

Fig. 2 depicts the high-level entity-relationship model of the involved conceptual entities and their relationships. For example, a researcher may conduct zero to several simulation experiments. One particular simulation, on the other hand, is carried out by exactly one researcher. One specific experiment is carried out with a set of clearly defined parameter settings, and the same experimental settings may apply to multiple simulations.

The conceptual data model (Fig. 2) is mapped onto a logical database schema shown in Fig. 3. The central entity in the warehouse data model is the Simulation. It brings together all the data, metadata and provenance data. Each simulation is carried out on a molecule and accompanied by provenance data, describing the experimental parameter settings, information on the software used, who conducted the experiment, where, when, etc. Dictionary tables (Fig. 3), which contain entries for repeated use, are a particular type of metadata. For example, the dictionary table SolventDict contains entries describing possible solvents used in the simulation. At this point, entries in the dictionaries related with simulation parameters reflect the most common values that researchers use when performing MD simulations. However, most of the dictionaries have a value (*Other*) if the researcher wants to specify a different value than the ones existing in the dictionaries. Upon selection of the value *Other*, a text box is provided for the researcher to specify the value he or she used for a particular simulation parameter. On a periodic basis, the curator of the P-found system will analyze those values, and add them to the respective dictionary if they are found to be of interest to the community.

The molecular properties data refer to data automatically calculated by a compute server from the trajectory file, whenever a new simulation is deposited in the warehouse. At this stage, the molecular properties being calculated are: root mean square deviation (RMSD), root mean square fluctuation (RMSF), radius of gyration (R_g), native contacts, secondary structure, and global, polar and non-polar solvent accessible surface area (SASA).

Future versions of the data warehouse will accommodate new tables reflecting the ability to accept folding and unfolding simulations other than MD simulations.

C. Hardware and Software Infrastructure

Currently, the hardware supporting the data warehouse is a 64-bit Intel Xeon dual-processor server (up to 3.2GHz, 1MB L2 Cache, 800 FSB) with 4GB DDR2 memory, 500GB hard drive, running the Linux Suse Operating System, and hosting Oracle Database 10g. For the calculation of molecular properties, the software VMD runs on a 64-way Itanium Altix 3700 high-performance computer with 128GB of addressable memory and 8 TB of storage space. Oracle 10g has unique security features that address requirements in the areas of privacy, regulatory compliance, and data consolidation, including row level security, fine grained auditing, and data encryption, and it is likely to be the database of choice for a grid-enabled environment.

III. User Interface

Three different user profiles for accessing the Data Repository are implemented:

- *Information users*. These users may browse the data stored in P-found, perform searches and visualize graphical representations of the molecular properties calculated for each simulation.
- *Data consumer users*. These users inherit the user privileges from *information users* but have also the ability to actually download data from the repository to their local computing environments.
- *Data provider users*. Data provider users inherit all privileges from *information users* and *data consumer users* but have also the privileges to upload simulation data to the repository.

Users who wish to have privileges associated with *Data consumer* or *Data provider* user classes are required to go through a registration process. In the registration process a researcher provides information like the name, e-mail address (mandatory), affiliation, etc. The researcher also supplies information about the access profile he wishes to have. Once

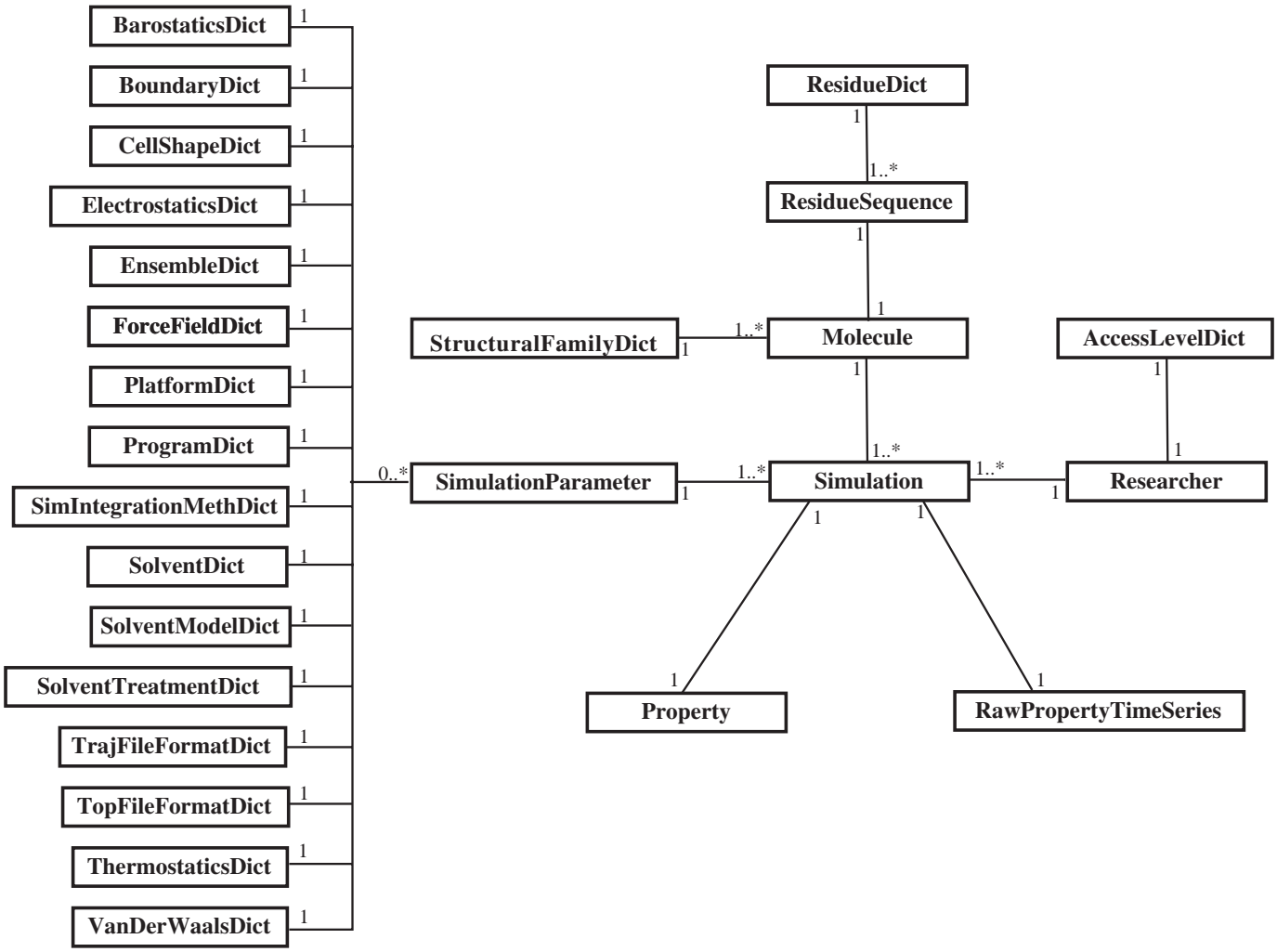

Fig. 3. The logical data model of the warehouse.

the researcher's identification data is submitted to the server, the system curator starts a validation process after which login and password are attributed.

The following sections describe the conceptual workflow of user interactions with the data warehouse. The user can (i) upload data, (ii) display data in a web browser, (iii) download query results to his or her client machine, (iv) download molecular property data and (v) download raw trajectory files. The five basic ways the P-found system can be used are described below.

A. Data Deposition and Processing

A key component in creating a public repository of information is the efficient capture and curation of the data. The system stores three types of data: trajectory data generated by simulation programs, provenance data associated with each simulation, and molecular properties data calculated for each trajectory. The trajectory file contains the coordinates over time of the molecule being studied. Due to the large size of the trajectory files, the coordinates of solvent molecules must be removed before upload by the reseacher. At the moment, the system is prepared to accept all types of trajectory and structure or topology files generated by the most popular MD simulation packages, namely NAMD [14], CHARMM [15], Gromacs [16] and AMBER [17].

The data deposition process itself is divided in three major steps: (i) researcher identification; (ii) entry of simulation parameters; and (iii) raw data upload (Fig. 4). Only researchers belonging to *Data provider users* class can deposit data into the system. When uploading new data a researcher may update some identification details, like e-mail, but other changes to a previously accepted researcher's profile, like affiliation, leads to a new registration process.

Simulation parameter data are organized in four different information levels. The researcher starts by supplying information on the molecule, such as the name of the molecule, the structural family, and cross-references to PDB [18] if available. The second level of information is a general description of the simulation details, like the method and software package used. The last two levels of information refer to details on the environment and configuration of the simulation experiment. The

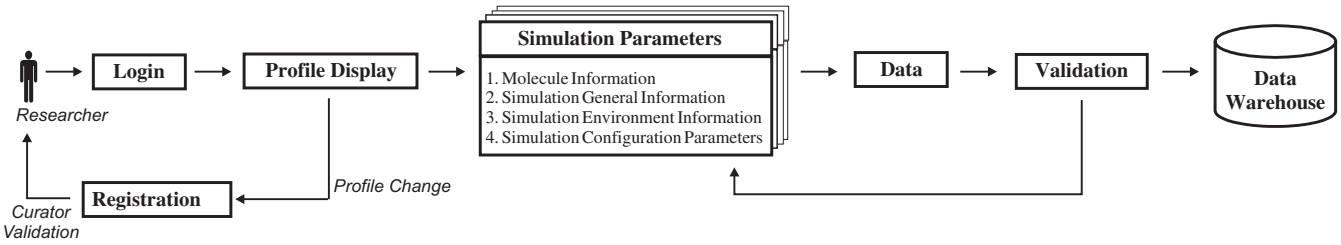

Fig. 4. Steps for data deposition in the warehouse.

TABLE I
MD SIMULATION PARAMETER DATA

Molecule Information
Name, function and structural family (CATH classification)
PDB identifier (if available)
Number of residues and atoms
Molecule primary sequence (obtained from the topology file)
Simulation General Information
Type of simulation: Folding or Unfolding
Simulation integration method, e.g. Langevin, Brownian
Software package and platform
Forcefield
Simulation Environment Information
Cell shape, boundary condition, solvent type and salt concentration
Temperature, energy, pressure, pH
Protonation state of particular residues
Simulation Configuration Information
Thermostatics, barostatics and ensemble
Integration time step and recording frequency
Electrostatics and Van der Waals settings

simulation environment is described by such items as boundary conditions, solvent type, temperature, pH, among others. The simulation configuration includes more specific configuration parameters like integration time step, electrostatics and Van der Waals interactions settings, etc. A more comprehensive description of the simulation parameter data is presented in Table I. Due to the large amount of data, the system supports asynchronous data upload, *i.e.*, the user can upload the data in different sessions. After this information is completed the upload process takes place. The trajectory data will be stored in the original format uploaded by the user. Every trajectory file must be accompanied by a structure or topology file, which is necessary to fully describe the molecule being uploaded.

A set of analysis tools is already implemented for standard and generic analysis of each simulation deposited. A general view of the steps involved in the computation of this analysis is depicted in Fig. 5. At the moment, the following properties are being calculated: root mean square deviation (RMSD) (backbone atoms only), root mean square fluctuation (RMSF) (Cα atoms only), radius of gyration (R_g) (backbone only), native contacts, secondary structure, and global, polar and non-polar solvent accessible surface area (SASA). A Tcl [19] package (`pfoundBasic`) brings together all procedures to calculate these properties. This modularity approach enables the analysis to be easily extensible. `pfoundBasic` runs under VMD's Tcl interpreter making use of the major facilities VMD provides for handling MD simulation data. After all molecular properties have been calculated, the resulting time series data are stored in the data warehouse. Additionally, a graphical representation of each property time series is also stored as an image file in `png` format.

B. Data Retrieval

The data retrieval interface provides the point of entry for all the trajectories stored in the repository. Two distinct query interfaces are available for the query of data within the P-found system. These two interfaces are customizable query forms that allow the searching over different items in the repository. Fig. 6 illustrates how the different query options are organized. The simplest query that can be performed is based on the simulation identification code, a unique ASCII string automatically assigned by the system to each simulation, at the moment of uploading. In this case, a single simulation is retrieved. The Simulation Explorer interface provides information about a single simulation. A summary containing information on the molecule and on the simulation general parameters is presented to the user. All the properties calculated for the simulation are summarized. For each property, *Data consumer users* and *Data provider users* can retrieve both the entire time series and the graphical representation for further analysis; *Information users* can only visualize the property graphical representation .

Multiple simulations might be retrieved when the researcher uses either the Quick Search Mode or the Advanced Search Mode. The Quick Search Mode is based on a small predetermined set of criteria: molecule name, type of simulation, simulation method, simulation program, structural family and deposition date. The Advanced Search Mode allows a broader scope of the search and allows Boolean operators. Apart from the criteria available in the Quick Search Mode, the Advanced Search Mode also includes attributes related with the origin of the data as well as much of the attributes described in Table I. The Query Result Browser interface allows for access to general information on the results retrieved from the search query. More detailed information for a single result can be

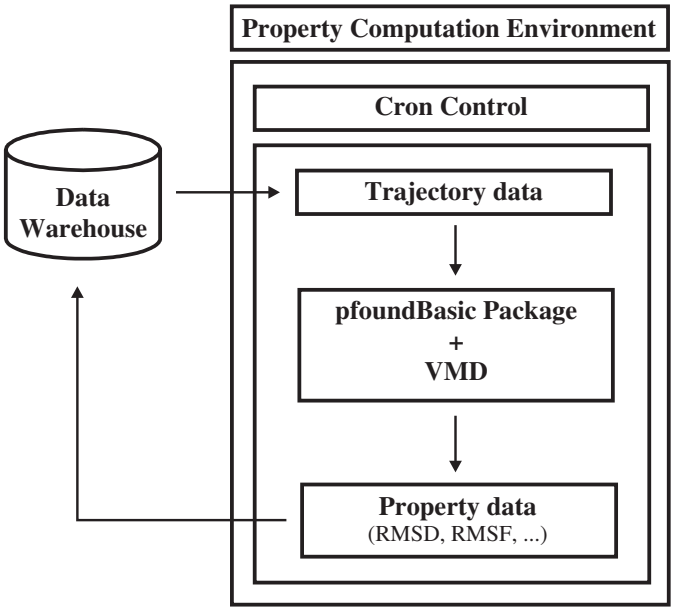

Fig. 5. Steps in the computation of molecular properties following uploading of simulation raw data.

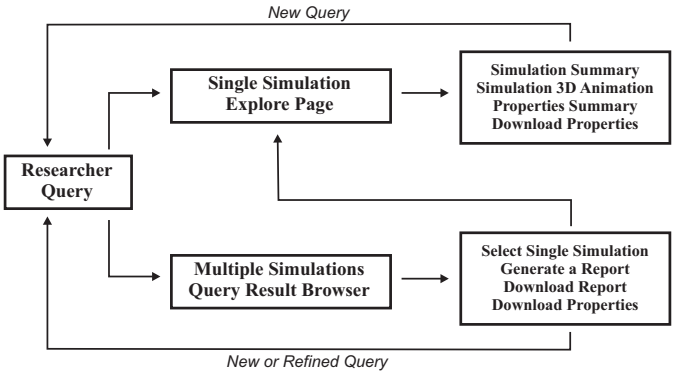

Fig. 6. Several query options are available at the P-found system.

viewed upon selection.

At the moment, the trajectory file may be made available for download in a case-by-case scenario, for example, if scientific collaborations requiring data exchange are being carried out. In these cases, trajectory files may be downloaded in any of the formats associated with the simulation packages previously referred (NAMD, CHARMM, Gromacs and AMBER). To perform this operation the researcher must have download privileges and the trajectory is made available through an FTP server.

C. Case Studies

The P-found system is being tested with several MD simulations performed with the programs NAMD and Gromacs. Molecular dynamics unfolding simulations of monomeric forms of the amyloidogenic protein transthyretin (TTR), [11] and MD folding simulations of the peptide δ-toxin [20] were uploaded to the data warehouse in order to test compatibility and performance issues.

Molecular dynamics unfolding simulations were performed on two different monomeric forms of TTR: wildtype (WT-TTR) and an amyloidogenic form (L55P-TTR). The TTR unfolding simulations were carried out with the program NAMD, using periodic boundary conditions, at high temperature (approximately 500 K), in a box of explicit water with salt, comprising a total of 44556 atoms. These simulations were carried out for 8 ns and are composed of 8001 frames. Only the information on the protein is uploaded into P-found. TTR monomers are composed of 127 amino-acids for a total of around 1915 atoms. The size of the trajectory files is in the order of 176 MB and the size of the topology files in the order of 440 KB. Uploading these files from Coimbra to Ulster took about 60 minutes. The time spent for calculating all molecular properties was about 140 minutes. Most of this time was used in the calculation of the variation of native contacts (25 minutes) and SASA (110 minutes).

The program Gromacs was used to study δ-toxin folding at 298 K in several explicit solvents: dimethylsulfoxide, methanol and water. These simulations were carried out for 20 ns and are compose by 20,001 frames. The peptide δ-toxin is constituted by 26 amino-acids, comprising a total of 268 atoms. The size of trajectory and topology files of these systems is considerably smaller than the TTR system – 22 MB and 64 KB, respectively. The uploading time was less than 2 minutes. The time needed to compute all molecular properties was about 180 minutes. As in the case of TTR systems, SASA was the molecular property involving more computation time: 160 minutes.

Uploading and molecular property calculation times are high. Molecular property computation time values can be greatly improved and ways of doing that are being studied at the moment. Nevertheless, the data from a new simulation upload takes about four hours to be publicly available.

IV. Conclusion

The search and identification of the rules driving the acquisition of the functional three-dimensional structure of a protein from its linear sequence of amino-acids is one of the central challenges that researchers face in molecular biology. In recent years, helped by the ever increasing computer power available to a wider range of researchers, many computational approaches have been taken to address the problem of how a protein folds, *i.e.* how it finds its functional 3D structure. The detailed knowledge of the molecular mechanism driving protein folding and unfolding are essential in the understanding of such diverse problems as the development of amyloid diseases, protein structure prediction from gene sequences, or *de novo* design of artificial enzymes.

Although an increasing number of research groups worldwide communicate results on computer simulations of protein folding and protein unfolding, sometimes even for the same or very similar proteins or peptides, it is almost impossible to compare in detail simulations from different groups or to apply alternative analysis methods to a wide range of

simulations. With this in mind we initiated development of a project to address the problem of sharing, analysis and comparison of protein folding and unfolding simulations – the P-found system. Our initiative is innovative in several ways: (i) the object of study is focused on protein folding and protein unfolding, using multiple computational approaches and allowing comparisons between different computational methods; (ii) development of new data mining tools for study and comparison of multiple simulations is an important part of the project; and (iii) global accessibility to the data repository is encouraged.

In previous papers [6], [12] we put forward the main ideas and requirements for such a data repository. Here, we present the first functional version of the P-found system. With this project, the research community will have a platform to compare multiple protein folding and unfolding simulations, using different conceptual approaches, different algorithms, for different or similar proteins, in a multitude of experimental conditions. This will open unprecedented opportunities for testing, for example, simplified *vs.* all-atom protein representations, implicit *vs.* explicit solvent effects, normal *vs.* pathological forms of a protein, among many other issues. Additionally, the availability of this resource will open new opportunities to develop and test novel analysis tools to explore the massive amounts of data generated by the simulations on protein folding and unfolding. Just as an example, we may think of comparing multiple simulations of two variants of one protein: one benign and one pathological. We know today that, due to the need of exploring different regions of the conformational folding/unfolding funnel, we cannot compare just one simulation from each one of the two protein variants. Thus, we face the challenge of comparing and finding similarities and differences, common or uncommon structural intermediates, among multiple trajectories and thousands of structures. This is a task calling for automation using data mining algorithms working on large data sets. The development of these novel analysis tools, the need to test them in a large number of simulations, and the lessons we may learn from these novel ways of looking into large groups of simulations, are additional points in favor of the need for a global repository of protein folding and unfolding simulations.

The full implementation, maintenance, and further development of the project presented here, ultimately will depend on the acceptance feedback from the research community, but we believe the time has come for such an effort to be successful. An important next step in the development of this project is the incorporation of suggestions made by the research community, which should be triggered by the presentation of this first version of the P-found system.

ACKNOWLEDGMENT

We acknowledge the cooperation of John Stone at the VMD Team, Kirby Vandivort at BioCoRE - Biological Collaborative Environment, and Olivier Riche for technical support. This work was supported in part by grant POCTI/BME/49583/2002 (FCT and FEDER, Portugal) to RMMB, Doctoral Fellowships SFRH/BD/16888/2004 to CGS and SFRH/BD/1354/2000 to NLF, and by FP6 DataMiningGrid Contract No. 004475 to WD.

REFERENCES

[1] C. M. Dobson, "Experimental investigation of protein folding and misfolding," *Methods*, vol. 34, pp. 4-14, 2004.

[2] C. D. Snow, E. J. Sorin, Y. M. Rhee and V. S. Pande, "How well can simulation predict protein folding kinetics and thermodynamics?," *Annu Rev Biophys Biomol Struct.*, vol. 34, pp. 43-69, 2005.

[3] J. N. Onuchic and P. G. Wolynes, "Theory of protein folding," *Curr Opin Struct Biol.*, vol. 14, pp. 70-75, 2004.

[4] R. Day and V. Daggett, "All-atom simulations of protein folding and unfolding," *Adv Protein Chem.*, vol. 66, pp. 373-403, 2003.

[5] J. W. Ponder and D. A. Case, "Force fields for protein simulations," *Adv Protein Chem.*, vol. 66, pp. 27-85, 2003.

[6] R. M. M. Brito, W. Dubitzky and J. R. Rodrigues, "Protein folding and unfolding simulations: A new challenge for data mining," *OMICS: A Journal of Integrative Biology*, vol. 8, pp. 153166, 2004.

[7] W. Humphrey, A. Dalke and K. Schulten, "VMD - Visual Molecular Dynamics," *J. Molec. Graphics*, vol. 14, pp. 33-38, 1996.

[8] W. Dubitzky, D. McCourt, M. Galushka, M. Romberg and B. Schuller, "Grid-enabled Data Warehousing for Molecular Engineering," in Special Issue on High-performance and Parallel Bio-computing in Parallel Computing, vol. 30, pp 1019-1035, 2004.

[9] Data mining tools and services for grid computing environments (DataMiningGrid) at `www.DataMiningGrid.org`.

[10] P. J. Azevedo, C. G. Silva, J. R. Rodrigues, N. Loureiro-Ferreira and R. M. M. Brito, "Detection of Hydrophobic Clusters in Molecular Dynamics Protein Unfolding Simulations Using Association Rules," *Proc. 6th International Symposium ISBMDA 2005, Lect. Notes Comput. Sc.*, vol. 3745, pp. 329-337, 2005.

[11] J. R. Rodrigues and R. M. M. Brito, "How important is the role of compact denatured states on amyloid formation by transthyretin?," *Amyloid and Amyloidosis*, CRC Press, pp. 323325, 2004b.

[12] D. Berrar, F. Stahl, C. G. Silva, J. R. Rodrigues, R. M. M. Brito and W. Dubitzky, "Towards data warehousing and mining of protein unfolding simulation data," *J. Clin. Mon. Comp.*, vol. 19, pp. 307-317, 2005.

[13] L. Moss and A. Adelman, "Data warehousing methodology," *J. Data Warehousing*, vol. 5, pp. 23-31, 2000.

[14] L. Kalé, R. Skeel, M. Bhandarkar, R. Brunner, A. Gursoy, N. Krawetz, J. Phillips, A. Shinozaki, K. Varadarajan and K. Schulten, "NAMD2: Greater scalability for parallel molecular dynamics," *J Comp.Physics*, vol. 151, pp. 283-312, 1999.

[15] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan and M. Karplus, "CHARMM: A program for macromolecular energy, minimization, and dynamics calculations," *J. Comput. Chem.*, vol. 4, pp. 187-217, 1983.

[16] H. J. C. Berendsen, D. van der Spoel and R. van Drunen "GROMACS: A message-passing parallel molecular dynamics implementation," *Comp. Phys. Comm.*, vol. 91, pp. 43-56, 1995.

[17] D. A. Pearlman, D. A. Case, J. W. Caldwell, W. R. Ross, T. E. Cheatham III,, S. DeBolt, D. Ferguson, G. Seibel and P. Kollman, "AMBER, a computer program for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to elucidate the structures and energies of molecules," *Comp. Phys. Commun.*, vol. 91, pp. 1-41, 1995.

[18] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N.Bhat, H. Weissig, I. N. Shindyalov and P. E. Bourne, "The Protein Data Bank," *Nucleic Acids Res.*, vol. 28, pp. 235-242, 2000.

[19] J. Ousterhout, *Tcl and the Tk Toolkit*, Addison-Wesley, 1994.

[20] N. Loureiro-Ferreira, J. R. Rodrigues and R. M. Brito, "Conformational Plasticity in δ-toxin," *6th European Symposium of The Protein Society*, 133a, 2005.