BioSuite: A comprehensive bioinformatics software package (A unique industry–academia collaboration)

*The NMITLI-BioSuite Team**

Keywords: Bioinformatics, BioSuite, industry–academia collaboration, software.

THE last decade has witnessed an exponential growth of information in the field of biological macromolecules such as proteins and nucleic acids and their interactions with other molecules. Computational analysis and predictions based on such information are increasingly becoming an essential and integral part of modern biology. With rapid advances in the area, there is a growing need to develop versatile bioinformatics software packages, which are efficient and incorporate the latest developments in this field. In view of this, the Council of Scientific and Industrial Research, India, undertook an initiative to promote a unique industry–academia collaboration, to develop a comprehensive bioinformatics software package, under its New Millennium Initiative for Technology Leadership in India programme. BioSuite, a product of that effort, has been developed by Tata Consultancy Services who took the primary coding responsibility with significant backing from a large academic community who participated on advisory roles through the project period.

BioSuite integrates the functions of macromolecular sequence and structural analysis, chemoinformatics and algorithms for aiding drug discovery. The suite organized into four major modules, contains 79 different programs, making it one of the few comprehensive suites that caters to a major part of the spectrum of bioinformatics applications. The four major modules, (a) Genome and proteome sequence analysis, (b) 3D modelling and structural analysis, (c) Molecular dynamics simulations and (d) Drug design, are made available through a convenient graphics-user interface along with adequate documentation and tutorials. The unique partnership with academia has also ensured that the best available methodology has been adopted for each of the 79 programs, which has been thoroughly evaluated in several stages, leading to high scientific value of the suite. The software, apart from having the advantage of running on a Linux platform on a personal computer, is also flexible, modular, and allows for newer algorithms to be plugged into the overall framework. The package will be valuable for high quality academic research, industrial research and development and for teaching purposes, both locally within the country as well as in the international arena. A full list of the programs as well as their example usage can be found at http://www.atc.tcs.co.in/bioinfo/publications/biosuite_paper.pdf.

Background

Genesis of BioSuite

The Council of Scientific and Industrial Research (CSIR), Government of India, proposed a new millennium initiative for technology leadership in India (NMITLI), in 2000, wherein India could acquire leadership positions in key technology areas (NMITLI). Development of versatile,

The team consists of *Tata Consultancy Services*: M. Vidyasagar, S. Mande, S. Rajgopal, B. Gopalkrishnan, S. T. P. T. Srinivas, C. Uma Maheswara Rao, T. Kathiravan, K. Mastanarao, S. Narendranath, S. Rohini, A. Irshad, T. Murali, C. Subrahmanyam, T. Mona, S. Sankha, V. Priya, D. Suman, V. V. Raja Rao, P. Nageswara Rao, R. Issaac, H. Yashodeep, B. Arundhoti, G. Nishant, S Jignesh, K. S. Chaitanya, S. P. V. Prasad Reddy; *Bose Institute*: P. Chakraborty; *Centre for DNA Fingerprinting and Diagnosis*: S. E. Hasnain, S. Mande, A. Nagarajaram, A. Ranjan, M. S. Acharya, M. Anwaruddin, S. K. Arun, Gyanrajkumar, D. Kumar, S. Priya, S. Ranjan, B. R. Reddi, J. Seshadri, P. Sravan Kumar, S. Swaminathan, P. Umadevi, V. Vindal, S. Vijaykrishnan; *Central Drug Research Institute*: A. K. Saxena, A. Dixit, P. Prathipati, S. K. Kashaw; *Indian Institute of Chemical Biology*: C. Mandal, S. Bag; *Indian Institute of Science*: N. Balakrishnan, M. Bansal, N. R. Chandra*, M. R. N. Murthy, S. Ramakumar, K. Sekar, N. Srinivasan, K. Suguna, S. Vishveshwara*, R. Anandhi, Bhadra, S. Das, P. Hansia, S. Hariharaputran, J. Jeyakani, R. Karthikeyan, R. K. Pandey, C. S. Swamy, B. Vasanthakumar; *Indian Institute of Technology Bombay*: P. V. Balaji, R. Y. Patel; *Indian Institute of Technology Delhi*: B. Jayaram, S. A. Shaikh; *Indian Institute of Technology Kharagpur*: P. P. Chakrabarti, A. Banerjee, A. Chakrabarti; *Indian Statistical Institute*: R. L. Karandikar, Delhi and P. Chaudhuri, Kolkata; *Institute of Microbial Technology*: G. P. S. Raghava, A. Ghosh; *Institute of Bioinformatics and Applied Biotechnology*: M. Bansal, N. Paramsivam; *Institute of Genomics and Integrative Biology*: S. K. Brahmachari, D. Dash, C. Balasubramaniam, A. Basu, P. Biswas, M. Hariharan, R. Mathur, K. S. Sandhu, V. Scaria, R. Shankar; *International Institute of Information Technology*: P. J. Narayanan, V. Jain, Nirnimesh; *Madurai Kamaraj University*: S. Krishnaswamy, V. Alaguraj, R. Marikkannu, A. V. S. K. Mohan Katta, N. Krishnan, K. V. Srividhya, P. J. Eswari; *National Institute of Pharmaceutical Education and Research*: P. V. Bharatam, P. Iqbal; *Saha Institute of Nuclear Physics*: D. Bhattacharyya; *University of Hyderabad*: G. R. Desiraju, J. J. Kumar, M. Ravikumar; *University of Madras*: M. Gautham, P. A. Prasad and D. Bharanidharan. *For correspondence. (nchandra@physics.iisc.ernet.in; sv@mbu.iisc.ernet.in)

Table 1. Roles played by different groups for ensuring successful development of BioSuite

Algorithm design, Code writing, Coding quality checks, Graphic-user interfaces and performance benchmarking	Tata Consultancy Services, team led by M. Vidyasagar Sharmila Mande and Rajagopal Srinivasan
Algorithm/module design suggestions and scientific evaluations	Academic partners
Project monitoring committee	R. Narasimha, G. Padmanaban, G. R. Desiraju, D. Balasubramanian
Project co-ordination	Yogeswara Rao and Vibha Sawhney, CSIR
Project funding	CSIR, NMITLI Scheme, Govt of India
Manuscript preparation	Coordinated by Nagasuma Chandra and Saraswathi Vishveshwara, IISc

portable bioinformatics software was recognized as one such area, taking into account the expertise available in the Indian academic community. Such a project, promoted by CSIR, was therefore flagged off in partnership with the industry, where Tata Consultancy Services (TCS) took the major responsibility of developing the BioSuite software with significant scientific support from the major academic institutions in the country (Table 1). The objectives of the project have been to develop indigenously, a set of software tools, that would assist the academic research, R&D and applications in industry, in the rapidly emerging field of bioinformatics and rational drug design.

The need for such a software suite is exemplified by two main factors: (a) increase in bioinformatics activities at all levels – education, research, industry, rapid growth of primary data and methods in computational biology and (b) limitations of existing suites – such as very high cost and not being comprehensive under a single framework, as discussed later. A team of 35 members from TCS worked on this project.

Mode of operation

To ensure the smooth functioning of the project, the following management structure was put in place: (a) *A Monitoring Committee*, monitored the progress of the project through periodic meetings with TCS and the academic partners providing timely focus, (b) *A Steering Committee*, consisting of scientists from academic institutions and TCS, coordinated the activities of the group, (c) *Domain experts and consultants*, consisting of all academic partners, helped in arriving at a basic structure for the suite. Given the large size of the group and the involvement of 18 institutions, the efforts from CSIR and the monitoring committees have played a significant role in fostering the unique partnership to ensure success of this project. The domain experts have advised TCS on the individual modules and individual programs required in each module, identified appropriate algorithms at each step, as also the features required for each program, as per the current research trends and requirements. Further, (d) *a team of pseudo-code developers of six people at* TCS, have interacted with domain experts and directed their (e) in-house *team of code developers*, consisting of 27 software engineers, who have written the actual code. The (f) *Software Project Management Committee* from TCS has ensured the overall activities at that end and ensured appropriate benchmarking and in-house quality checks from the software perspective. The scientific performance of the codes developed has been further evaluated by the academic partners, who have tested and reported bugs to Project Management Committee, after which the codes have been improved/modified where required. Further, an autonomous assessment of the suite has been obtained by an independent expert in the area.

Operational schedules

A glimpse of the schedules and the various milestones reached are given below: (a) Identification of the modules, the required programs in each module and the appropriate algorithm(s) for each program, was completed in the first four months, following which a (b) Software Requirement Specification (SRS) document was developed and reviewed in the next two months. Next, the pseudo-codes were developed in about five months and converted into final code in the next 12 months. In parallel with alpha-testing that was carried out simultaneously with code development, the documentation and creation of a user guide took about seven months. Bug reporting and bug fixes were carried out in iterations through the testing phases and a beta-version was produced by June 2004, taking a total of 24 months. Evaluation and bug fixing of this version was carried out in five months, leading to the first full version, soft-launched in July 2004 and product released in December 2004.

Overview of the organization of the suite

The entire package, consisting of 79 different programs is organized into four major modules, all linked through three common graphics-user interface (GUI) workbenches, as illustrated in Figure 1. The four modules are: (a) Genome and sequence analysis, (b) 3D modelling and structure analysis, (c) Molecular dynamics simulations and (d) Drug design. They are accessible through central GUIs for file handling, sequence and structure windows.

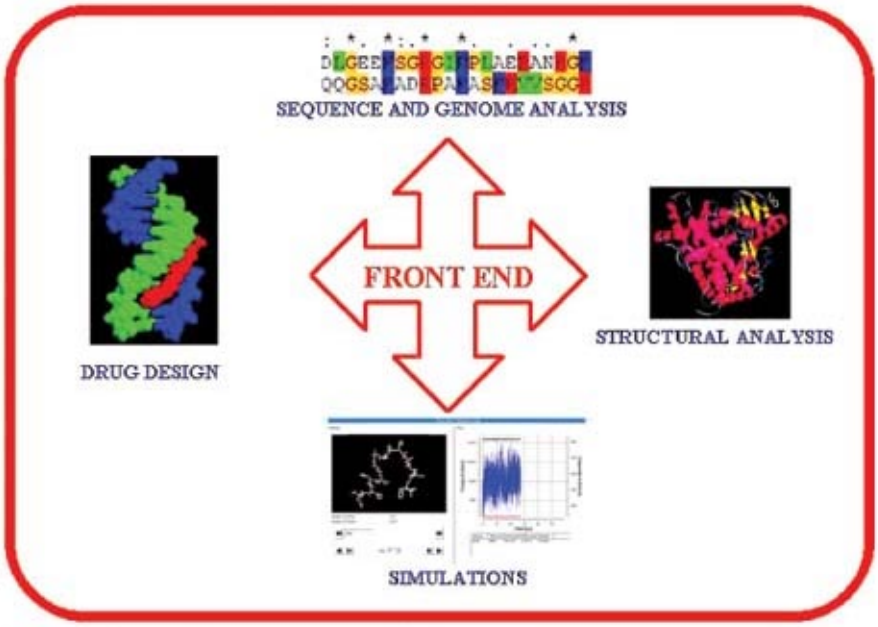

Figure 1. Modular organization of BioSuite.

Table 2. Examples of programs contained in the modules

Sequence and genome analysis

Genome sequence assembly and EST mapping[1], ePCR[2], ORF prediction[3], Intron–exon boundary[4], Database search[5] and sequence alignments (pairwise[6,7]; multiple[8]; whole genome alignment[9]); Motifs and patterns (restriction sites[10], motif building and searching[11]; primer and probe design[12]); RNA and protein secondary structure and transmembrane prediction[13–15]; Domain building and searching[16], gene order[17], unique genes[18]; Phylogenetic analysis, tree construction, evolutionary distance estimation and profiling[19–21].

Structural analysis

Nucleic acid analysis[22], protein structure quality check[23], symmetry-related molecules, structural superposition[24], interactions[25], homology modelling and threading[26]; Fold classification[27]; Molecular surface area, solvent accessible surface area and volume[28]; Binding site detection (PASS[29]; ET[30]).

Simulations

Energy minimizations (steepest descent[31] and conjugate gradient minimizers[32]; forcefields[33]); Electrostatic potential maps[34,35]; Molecular dynamics[36,37]; MD analysis of various trajectories, RMSD, average position and plots of system properties.

Drug design

Structure-based design using protein–ligand docking[38]; Conformation search[39]; Steric and electrostatic ligand alignment[40]; QSAR with over 80 descriptors and regression analysis; Pharmacophore identification and pharmacophore-based search[41,42].

Table 2 lists the important programs in each module. A full list of the modules as well as example outputs of the individual programs can be found at http://www.atc.tcs.co.in/bioinfo/publications/biosuite_paper.pdf. Combination of the four modules makes BioSuite a comprehensive package, covering much of the activities of the bioinformatics spectrum, starting from genome sequences to individual and multiple protein sequences, different levels of structure prediction, analysis of the structures, molecular mechanics calculations, molecular dynamics simulations, chemoinformatics and finally integration with the application of the sequence and structural analyses in rational drug design through algorithms for QSAR, pharmacophore identification and docking processes, for facilitating rational drug design.

Choice of algorithms and coding methods

Choice of algorithms was discussed extensively with academic partners and the latest concepts available in the literature have been adopted wherever possible. For some programs, more than one algorithm has also been implemented, to suit the current research trends of using multiple methods and studying consensus predictions. In general, about two scientists have analysed and chosen a particular algorithm for a particular purpose. Table 2 indicates the algorithms chosen for each of the programs. The knowledge and description of each of the algorithms have been captured into detailed SRS documents by the pseudo-code development team at TCS through extensive interactions with the academic partners as well as with a detailed study

of the appropriate literature. The pseudo-code generated for each algorithm and its linkages have been developed using formal software engineering methods, so as to guarantee correctness. The pseudo-code was then converted into actual code by another set of programers who have ensured strict adherence to well-established quality processes such as CMMi Level 5.

All codes have been written in C^{++}. A total of 170 algorithms and about 100 QSAR descriptor calculators have been implemented in 79 programs, with about 700,000 lines of code. The suite is modular, which not only facilitates seamless updation of the modules but also enables integration of new programs by the end users.

Description of the modules

The functionalities of the programs contained within the four major modules are briefly described below.

Genome and proteome sequence analysis

This module deals with the applications relating to the analysis of the nucleic acid and protein sequences, not only of individual molecules, but also of complete genome and proteome sequences. This module would enable researchers to annotate genomes, predict protein secondary structures, derive a phylogenetic relationship among organisms and compare two genomes for similarities at the gene or protein level, along with a range of other applications. This module is further divided into four sub-modules: Sequence analysis, genome analysis, Comparative genomics and Utilities.

Sequence analysis of individual molecules is enabled through the sequence analysis modules, while the programs in the 'Genome analysis' sub-module enable comparison and analysis of full genomes and proteomes. Two database searching tools, BLAST and PSI-BLAST are interfaced with the suite, that will enable searching databases to identify a given sequence or find conserved domains or even find distantly related homologues from some other species. An option of building custom-made databases is also provided. Alignment of sequences, a crucial task in sequence analysis, is provided for, through two well-established algorithms for global and local alignments using dynamic programing algorithms (Needleman–Wunsch and Smith–Waterman). Further, a hierarchical clustering-based multiple alignment algorithm (ClustalW) is included for aligning a set of sequences. Besides, pattern identification and matching tasks such as finding composition, inverted repeats, DNA structure motifs, restriction site analysis and repeat analysis, are part of this module.

Algorithms for secondary structure prediction including transmembrane region detection, RNA structure prediction and analysis are also part of this module. The secondary structure prediction algorithms were trained (or retrained as appropriate) using a comprehensive dataset containing 731 high resolution protein structures (with resolutions ≤ 2 Å) that comprise a non-redundant dataset (redundancy has been removed through sequence comparisons, using a similarity cut-off of 25% with the Blosum62 substitution matrix). Use of a large dataset in training the prediction algorithms ensures high prediction accuracy. A comprehensive biophysical parameter computation ability has also been built into BioSuite, by extracting 36 different physico-chemical properties for protein molecules from the dataset and subsequently using them as training-sets in the prediction algorithms. Algorithms for predicting isoelectric point, peptide cleavage patterns, B-cell antigenicity from protein sequences are also included in this module. Yet another useful feature of this module is the domain building and analysing functionality. Programs are available for identifying domains, building consensus domain sequences, calibrating them and searching across a database. Hidden Markov models using sequence profiles are used for these purposes. In addition, the module has programs for studying molecular evolution, to cluster groups of sequences based on several criteria and to compute phylogenetic trees as well as to calculate evolutionary distances. Finally, algorithms for gene finding, gene assembly, probe and primer design, vector trimming and EST analysis are also part of this module. Two examples of using the programs of this module are illustrated in Figure 2 *a* and *b*.

3D Modelling and analysis

The 3D modelling and analysis module has capabilities to build, analyse and predict three-dimensional structures of macromolecules and macromolecular complexes. This module is further subdivided into the following submodules: (a) Homology modelling, (b) Threading, (c) Building proteins, (d) Building nucleic acids, (e) Building carbohydrates, (f) Generation of symmetry-related molecules, (g) Structural superposition, (h) Surfaces and volumes, (i) Binding site analysis, (j) Nucleic acid analysis, (k) Interactions, (l) Quality check, and (m) Fold detection. Example snapshots are shown in Figure 2 *c* and *d*.

Building the models of protein molecules by predicting their three-dimensional structures by comparative modelling techniques are enabled through the first two sub-modules, for which six algorithms are available that incorporate the latest concepts in these areas. Building nucleic acids and carbohydrates using geometric information is enabled through the building modules. A notable feature of the builder programs is the incorporation of 17 geometrical templates for nucleic acids and 12 templates for carbohydrates providing a handle to address the stereo-chemical variability in a large number of sugars. Several programs that can address visualization and analysis of crystallographically derived structures are also included in this

module. For example, a lattice assembly of a protein molecule, as seen in its crystal structure can be generated easily. Structure validation tools for proteins and nucleic acids are enabled through the quality check programs. Extensive analysis is possible through the analysis and interactions functions, that can be used for analysing integral features of protein structure, protein–protein interactions as well as protein–ligand interactions. Finally, algorithms for classifying protein structures, in relation to the other protein structures known in the literature, are also included in this module through the fold detection routines. Here too, the unique integration of building,

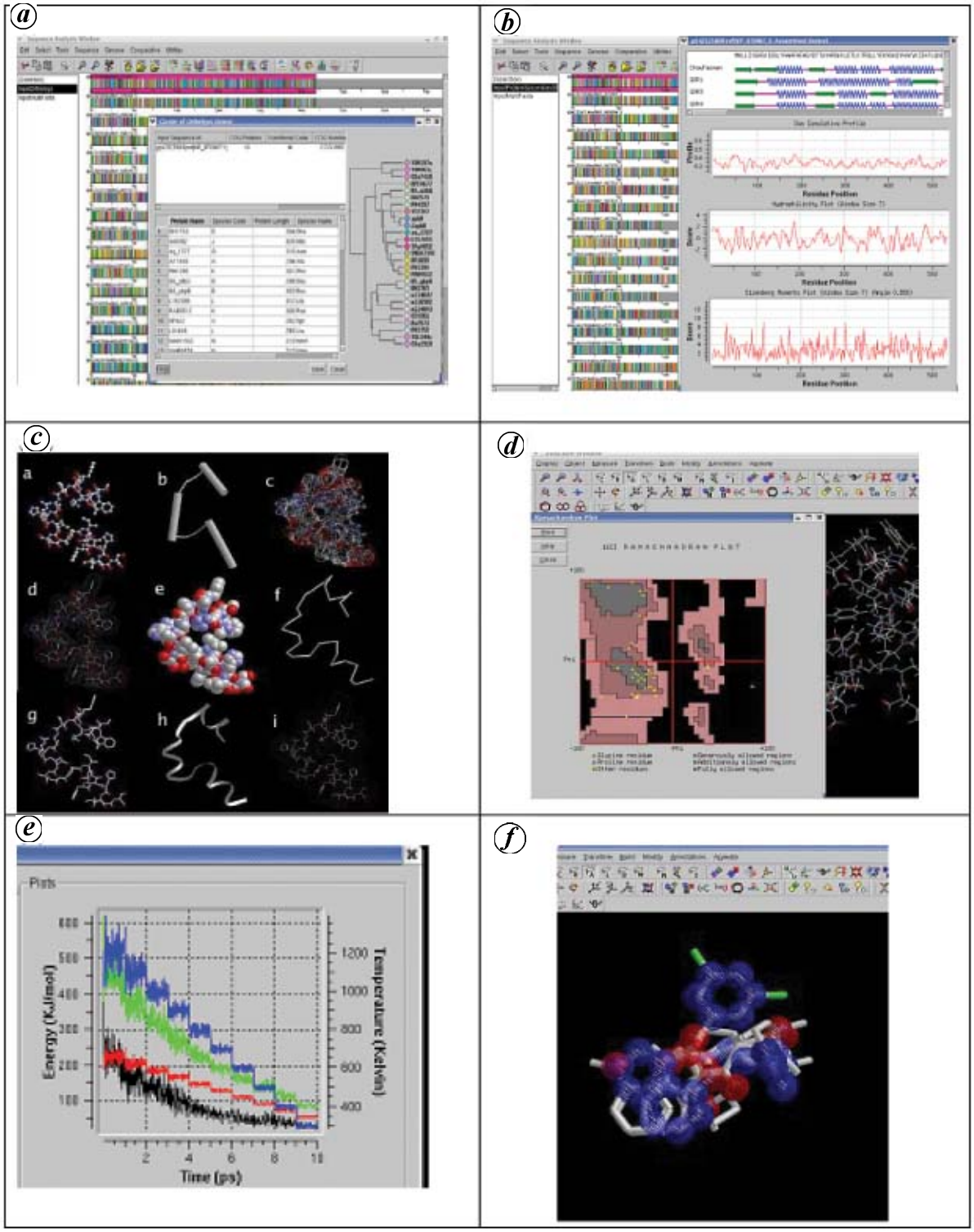

Figure 2. (*Contd…*)

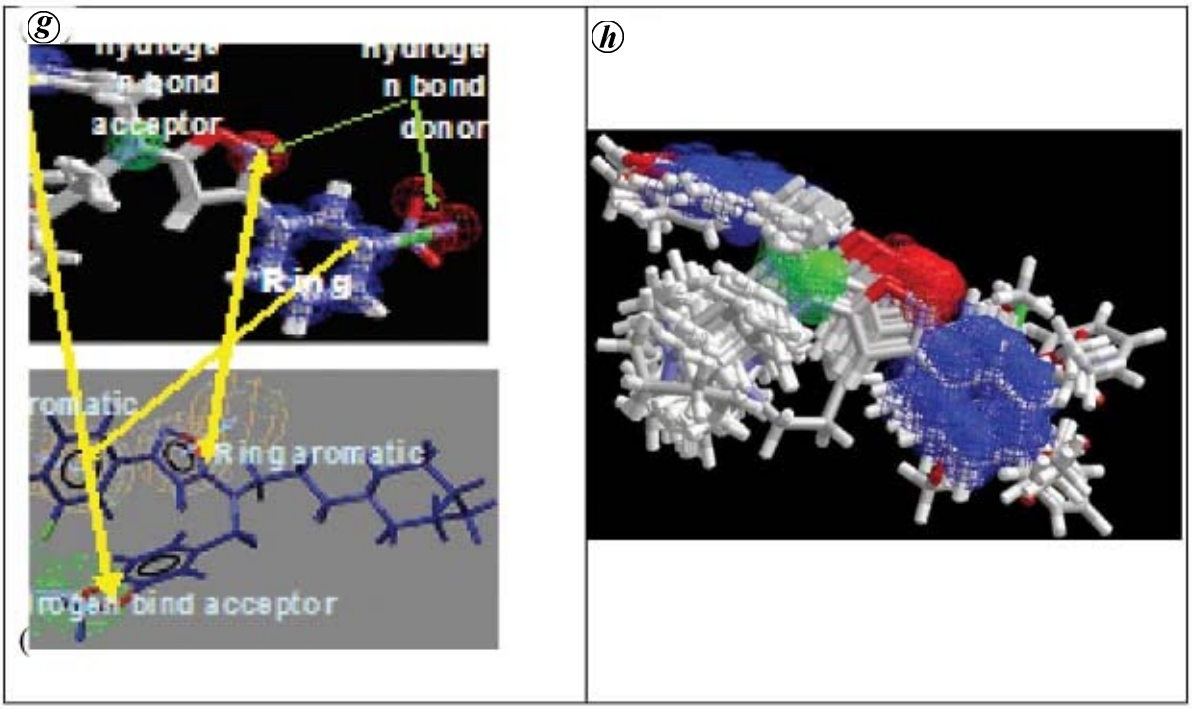

Figure 2 ***a–h.*** Example snapshots from various modules of BioSuite: ***a***, Genome comparison: Mapping Protein gi|42525869, from *Bacillus halorudians* to Clusters of Orthologous Groups (COG no. 1893), by using orthologues. A homologue of a lipase from Treponema denticola gi|42525869 was identified from *Bacillus halorudians*; ***b***, Protein secondary structure prediction using different methods and property profiles derived for the lipase protein sequence; ***c***, Different molecular representations in BioSuite – (a) ball-and-stick, (b) cartoons, (c) molecular surface, (d) van der Waals surface, (e) space fill, (f) C-alpha trace, (g) sticks, (h) ribbons, (i) solvent accessible surface; ***d***, Protein structure quality check using a Ramachandran plot; ***e***, An example of MD-analysis, variation in kinetic energy, potential energy, total energy, temperature during simulation; ***f***, An example of pharmacophore fitting, ***g***, Alignments produced by BioSuite derived pharmacophore model, and ***h***, An example of a field fit alignment: Molecular similarity between a pair of molecules is calculated by using the Gaussian function in BioSuite.

analysis and structural bioinformatics tools such as structure classification, all within one framework, significantly enhances the technical value of BioSuite.

Simulations

The 'simulations' module essentially simulates the behaviour of a molecule, in terms of its three-dimensional structure. The different submodules covered are, Forcefield, Energy minimization, Molecular dynamics, Monte Carlo simulations and Electrostatics. The molecular simulation of a system can conceptually be broken into three components: (a) generating a computational description of a biological/chemical system typically in terms of atoms, molecules and associated force field parameters, (b) the numerical solution of the equations which govern their evolution, and (c) the application of statistical mechanics to relate the behaviour of a few individual atoms/molecules to the collective behaviour of the very many. BioSuite is compatible both with the AMBER and the CHARMM force fields for macromolecules (proteins, nucleic acids and carbohydrates) and uses GAFF for small molecules (for e.g. natural substrates, drugs and drug-like substances). For each of the force fields, both treatments of the type of dielectric: either constant or distant dependent, are provided.

Several algorithms for first-order unconstrained energy minimization are contained in this module, providing a wide range of line search options. Thus, the coordinates of the molecular system can be adjusted so as to lower its energy, relative to the starting conformation, by using one of the following minimizers: Steepest descent algorithm, Conjugate gradient methods, Fletcher–Reeves algorithm, Polak–Ribiere algorithm, Polak–Ribiere plus algorithm and Shanno's algorithm.

Further, to carry out molecular dynamics (MD) simulations, BioSuite provides NVE (micro-canonical), NVT (canonical), and NPT (isobaric–isothermal) ensembles for MD simulations with the choice of using velocity–verlet or leapfrog integrator. BioSuite also provides options for using SHAKE and RATTLE constraints.

MD being a deterministic approach, where the state of the system at any future time can be predicted from its current state, the tools provided in the suite can be used for solving Newton's equations of motion for a given initial conformation, to study how the system evolves over

time. Several intuitive and user-friendly tools are provided to analyse the resulting trajectories or time series of conformations. For example (Figure 2 *e*), plots at various energy levels along with the temperature, can be obtained. Plots generated with defined parameters show the structure and position at various energy levels, both of them present in two adjacent panels that can help to view the position of the molecule at a given temperature. The Monte Carlo method that generates configurations randomly and uses a special set of criteria to decide whether or not to accept each new configuration, is also part of this module.

In the electrostatics sub-module, BioSuite provides a solution for the linear Poisson–Boltzmann equation, to enable modelling of contributions of solvent, counterions and protein charges to electrostatic fields in molecules. Four choices for boundary conditions namely, zero, partial coulombic, full coulombic and focusing, are provided. For charge distribution, there are two options: trilinear and uniform. BioSuite has a very fast SOR solver, which utilizes spectral radius calculations to speed up convergence.

Drug design

This module provides the following functionalities: (a) Prediction of biological activities of unknown chemical entities using QSAR, (b) Identification of pharmacophores in biologically active molecules, (c) Superimposition of a set of molecules in 3D space by alignment, (d) Identification of the ligand poses in 3D space when it binds to a target using docking. Using the functionalities provided in the drug design module, one can identify lead-like molecules from a set of molecules, redesign them and predict their activities. Thus, lead optimization can be achieved iteratively. If the target structure is known, then the lead optimization can be done using the structure-based method, such as by docking.

The process of aligning a set of molecules in three-dimensional space, to find the superimposable regions of a group of molecules or to estimate molecular similarity can be performed by using either the 'Field Fitting' or the 'RMS Fitting' approach. The field fitting is done by aligning molecules using their electrostatic potentials and steric shapes, starting from their atomic coordinates and charges computed from Gaussian functions, while the 'RMS fitting' is done by minimizing the distances between specified atoms in the molecules. Flexible superposition can also be achieved by allowing rotations about single bonds.

For deriving and matching '3D-pharmacophores', the following features are extracted/used: (a) Hydrogen bond donor, (b) Hydrogen bond acceptor, (c) Aliphatic hydrophobic group, (d) Aromatic ring, (e) Negatively charged group, and (f) Positively charged group. Pharmacophores are identified by using configurations of features common to a set of molecules. The pharmacophoric configurations are identified by a pruned exhaustive search, starting with small sets of features and extending them until no larger common configuration exists.

To carry out QSAR, where consistent relationships between the variations in the values of molecular properties and the biological activity for a series of compounds are sought, so that these 'rules' can be used to evaluate new chemical entities, a series of widely accepted feature extraction and statistical tools are provided within BioSuite. For example, a 2D-QSAR calculation uses either one or combinations of (a) Electronic, (b) Spatial, (c) Structural, (d) Thermodynamic and (e) Topological descriptors. BioSuite has the ability to compute 89 different descriptors. a few representative descriptors from different classes, e.g. Polarizability, HOMO and LUMO (electronic), Hf and Log P from (thermodynamic), log P, MR (thermodynamic), etc. and were compared with those computed from standard softwaers, using a dataset of 33 isooxazoles as potential thrombin receptor antagonists and in general, a high correlation (>0.9) was observed for the descriptor values.

Creating and refining a training set required for QSAR predictions are aided by (a) K-means, (b) K-nearest neighbours or (c) UPGMA hierarchical clustering algorithms. Tools are also provided for building user-defined data sets/training sets as well as for searching chemical databases. The QSAR model can be generated using regression techniques such as Multiple Linear Regression or Partial Least Squares. If the linearly independent descriptors for the molecules have to be eliminated while generating the model, then a dimensionality reduction can be performed by using either (a) Principal component analysis or (b) Discriminant analysis. Validation of the model to check the accuracy of the generated model can be performed by the K-fold cross validation technique

The structure-based drug design sub-module contains algorithms and utilities required for carrying out molecular docking. Using either simulated annealing or genetic algorithms (GA) based technique, the ligand conformations are searched and docked into the binding site of the macromolecule. In a simulated annealing-based method, the ligand's current position, orientation and conformation are changed during each cycle, to reach the most energetically favourable conformation of the ligand bound to the target macromolecule. Thus these algorithms predict both the lowest energy conformation of the bound ligand as well as the best position and orientation for its binding to the target molecule, within the realm of the scientific capabilities of the approach.

A second popular algorithm is provided for this, the one based on genetic algorithms. The conformations of the ligand are encoded as a chromosome. The crossover and mutation operators are used to bring about random changes in the conformations of the ligand. A fitness function is defined for calculating the energy of the conformations generated. Through a number of runs of the GA cycle, a conformation having minimum energy is obtained.

Conformation search functionality generates the conformations for an input molecule, clusters the conformations and displays energy and torsion angle values of low energy conformations. This application generates conformations using two different methods, namely random conformation search and systematic conformation search. The random conformation search uses the simulated annealing algorithm. An option is provided to the user to select the rotatable bonds in the molecule. A few sample results from the drug-design modules are presented in Figure 2 *f–h*.

Performance evaluation

Evaluation has been an integral part of the entire development process. To start with, the choice of modules and the choice of algorithms themselves were evaluated, both at TCS and by the academic partners. The pseudo-codes and the SRS documents were then verified, followed by verification of the software codes by the TCS team. The scientific performance of the algorithms at various stages (versions 0.3, 0.7, 1.0a and 1.0) was evaluated independently by the academic partners at their institutions and any bugs reported or improvements suggested were subsequently considered and implemented into the suite, where appropriate. The outputs of each program were compared with those of other established academic codes/ commercial packages, to verify the scientific performance. They were also compared with the latest implementations of the chosen algorithms in the public domain, where available. The performance has been found to be comparable in all cases. While the utilities of many of the individual programs have been enhanced while implementing in BioSuite, the scientific capabilities and limitations of each of the programs are bounded by those of the corresponding original algorithms cited in Table 2.

An example of the manner in which the scientific performance was evaluated, is cited below. For testing the drug design module, 42 thymidine monophosphate kinase inhibitors were taken and minimization performed using both AMBER and CHARMM force fields with the conjugate gradient algorithm method. Conformational searches were tested with both systematic and randomized search methods. Alignments were satisfactory and we obtained low RMSD values for similar molecules, comparable to those obtained in Cerius. The time for computation was found to be good and comparable to other competitor software. The docking procedure is simple and user-friendly.

Prominent features of the package

For the most part, the existing software packages evolved out of academia, and were implementations of algorithms developed at different places and different times by different persons. As such, often there is no single 'superstructure' into which the algorithms fit seamlessly. To overcome these issues, BioSuite has been written in a modular fashion, which would permit the easy implementation of new algorithms as and when they are discovered. The unique partnership of the industry with academia harnesses the strengths of both communities, thus leading to a superior product both scientifically as well as according to software engineering standards. Some of the unique features of BioSuite are: (a) It is comprehensive, contains programs for carrying out sequence, whole genome and structure analysis, drug design, all under a common framework. (b) The software runs on simple personal computers on a Linux platform. (c) Domain identification and domain searching tools also available. (d) Transmembrane beta strand prediction, enhanced capability in building molecules in terms of the number of secondary structure templates available. (e) Enhanced capability in building larger carbohydrate structures, and (f) Code written fresh with CMMi-5 standards and consistency in coding methods to incorporate versatility in each program making up the entire suite, keeping in view of the genome-scale operations in bioinformatics.

Roadmap for the future

Going forward, several features are planned to be added to BioSuite to make it an even more useful platform for scientific research. Some developments in the pipeline are described below:

ADME

The Absorption, Distribution, Metabolism and Excretion profile (ADME) of a drug is an important determinant of its therapeutic efficacy. Accurately modelling the ADME properties of a candidate drug molecule is a necessary step to increase the chances that it will eventually become a successful drug. In the recent past, models have been developed for estimating various ADME-related properties such as blood-brain barrier penetration, human intestinal absorption, binding affinity to human serum albumin and $CaCO_2$ cell permeability. These will be integrated into the existing QSAR module of BioSuite.

Flexible docking

Docking, in BioSuite 1.0, explores the energetically optimal fit of a flexible small molecule with a rigid protein molecule. In subsequent releases, an improved version of the docking algorithm will be implemented that allows restricted flexibility in the protein molecule as well. This has been shown to be useful in improving the accuracy in prediction of the optimal binding conformation.

De novo drug design

An important requirement for drug design is the ability to generate novel molecules that bind to a known active site. Implementation of an algorithm is underway for the generation of novel binding candidates using a strategy of fragment docking followed by elaboration of selected fragments.

tRNA identification

A procedure for identifying tRNA genes in a genome will be included in the next version of BioSuite. The program identifies tRNAs based on the recognition of two intragenic control regions known as A and B boxes, a highly conserved part of B box, a transcription termination signal, and the evaluation of the spacing between these elements.

Improved whole genome comparison

MUMmer is an open source software package for the rapid alignment of very large DNA and amino acid sequences. A newer version of the MUMmer package has been integrated in BioSuite to find maximal unique matches between two genomes. The MUMmer output can also be viewed in the dot-plot format.

Improved graphics

Several techniques are being implemented to enhance the quality of the 3D graphics display in BioSuite while speeding up the display.

Scripting interface

While BioSuite provides a number of features and a vast array of functionality, users might want to implement their own procedures and programs. For this purpose, a scripting interface that exposes the functionality in BioSuite will be provided so that users can create their own workflows, develop and test new ideas and automate several tasks.

Sketcher

The next version of Bio-Suite will include a 2D sketcher for drawing molecules in a manner that chemists are familiar with and to automatically generate 3D structures for the molecules.

A high-performance version called Bio-Cluster for some of the memory intensive applications is also planned.

Hardware requirements and documentation

The minimum hardware requirements for BioSuite are as follows: Intel compatible ×86 Processor, 1.5 GHz, 256 MB RAM, 3 GB Free Hard Disk Space, Display capable of 1280×1024 pixel resolution, High end graphics card with 3D support for better viewing, Red-Hat Linux 8.0 or 9.0 or Fedora-Core 1/2 operating systems. BioSuite comes with its own set of documentation. The entire package is well documented and comes with easy to use tutorials, which reduce the learning curve and increase efficiency. Detailed documentation is available at the BioSuite website: http://www.atc.tcs.co.in/BioSuite/.

1. Huang, X., A contig assembly program based on sensitive detection of fragment overlaps. *Genomics*, 1992, **14**, 18–25.
2. Schuler, G. D., Sequence mapping by electronic PCR. *Genome Res*., 1997, **7**, 541–550.
3. Delcher, A. L., Harmon, D., Kasif, S., White, O. and Salzberg, S. L., Improved microbial gene identification with GLIMMER. *Nucleic Acids Res.*, 1999, **27**, 4636–4641.
4. Kleffe, J., Hermann, K., Vahrson, W., Wittig, B. and Brendel, V., Logitlinear models for the prediction of splice sites in plant pre-mRNA sequences. *Nucleic Acids Res.*, 1996, **24**, 4709–4717.
5. Altschul, S. F., Gish, W., Miller, W., Myers, E. W. and Lipman, D. J., Basic local alignment search tool. *J. Mol. Biol.*, 1990, **215**, 403–410.
6. Needleman, S. B. and Wunsch, C. D., A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, 1970, **48**, 443–453.
7. Smith, T. F. and Waterman, M. S., Identification of common molecular subsequences. *J. Mol. Biol.*, 1981, **147**, 195–197.
8. Thompson, J. D., Higgins, D. G. and Gibson, T. J., CLUSTALW improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, 1994, **22**, 4673–4680.
9. Arthur, L. D., Kasif, S., Fleschmann, R. D., Peterson, J., White, O. and Salzberg, S. L., Alignment of whole genomes. *Nucleic Acids Res*., 1999, **27**, 2369–2376.
10. Knuth, D. E., Morris, J. H. and Pratt, V. R., Fast pattern matching in strings. *SIAM J. Computing* 1977, **6**, 323–350.
11. Bailey, T. L. and Elkan, C., Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning J.*, 1995, **21**, 51–83.
12. SantaLucia, J. Jr., Allawi, H. T. and Seneviratne, P. A., Improved nearest-neighbor parameters for predicting DNA duplex stability. *Biochemistry*, 1996, **35**, 3555–3562.
13. Zuker, M., On finding all suboptimal foldings of an RNA molecule. *Science*, 1989, **244**, 48–52.
14. Jones, D. T., Protein secondary structure prediction based on position-specific scoring matrices. *J. Mol. Biol.*, 1999, **292**, 195–202.
15. Gromiha, M. M., Majumdar, R. and Ponnuswamy, P. K., Identification of membrane spanning beta strands in bacterial porins. *Protein Eng.*, 1997, **10**, 497–500.
16. Durbin, R., Eddy, S., Krogh, A. and Mitchison, G., *Biological Sequence Analysis*: *Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, UK, 1998.
17. Mazumdar, A., Kolaskar, A. and Donald, S., GeneOrder: Comparing the order of genes in small genomes. *Bioinformatics*, 2001, **17**, 162–166.
18. Enright, A. J. and Ouzounis, C. A., GeneRAGE: a robust algorithm for sequence clustering and domain detection. *Bioinformatics*, 2000, **16**, 451–457.
19. Marcotte, E. M., Pellegrini, M., Thompson, M. J., Yeates, T. and Eisenberg, D., A combined algorithm for genome-wide prediction of protein function. *Nature*, 1999, **402**, 83–86.
20. Tamura, K. and Nei, M., Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. *Mol. Biol. Evolut*., 1993, **10**, 512–526.

21. Fitch, W. M. and Margoliash, E., Construction of phylogenetic trees. *Science*, 1967, **155**, 279–284.
22. Bansal, M., Bhattacharyya, D. and Ravi, B., NUPARM and NUCGEN: software for analysis and generation of sequence dependent nucleic acid structures. *Comput. Appl. Biosci.*, 1995, **11**, 281–287.
23. Laskowski, R. A., MacArthur, M. W., Moss, D. S. and Thornton, J. M., PROCHECK: a program to check the stereochemical quality of protein structures. *J. Appl. Cryst.*, 1993, **26**, 283–291.
24. Sutcliffe, M. J., Haneef, I., Carney, D. and Blundell, T. L., Knowledge based modeling of homologous proteins, Part I: Three dimensional frameworks derived from the simultaneous superposition of multiple structures. *Protein Eng*., 1987, **1**, 377–384.
25. Baker, E. N. and Hubbard, R. E., Hydrogen bonding in globular proteins. *Progr. Biophys. Mol. Biol.*, 1984, **44**, 97–179.
26. Zhang, C. and Kim, S., Environment-dependent residue contact energies for proteins. *Proc. Nat. Acad. Sci.*, 2000, **97**, 2550–2555.
27. Orengo, C. A. and Taylor, W. R., SSAP: Sequential structure alignment program for protein structure comparison. *Methods Enzymol*., 1996, **266**, 617–635.
28. Connolly, M. L., Computation of molecular volume. *J. Am. Chem. Soc*., 1985, **107**, 1118–1124.
29. Brady, G. P. and Stouten, F. W. P., Fast prediction and visualization of protein binding pockets with PASS. *J. Computer-Aided Mol. Design*, 2000, **14**, 383–401.
30. Lichtarge, O., Bourne, H. R. and Cohen, F. E., An evolutionary trace method defines binding surfaces common to protein families. *J. Mol. Biol.*, 1996, **257**, 342–358.
31. Gilbert, J. C. and Nocedal, J., Global convergence properties of conjugate gradient methods for optimization. *SIAM J. Optimization*, 1992, **2**, 21–42.
32. Watowich, S. J., Meyer, E. S., Hagstrom, R. and Josephs, R., A stable, rapidly converging conjugate gradient method for energy minimization. *J. Computat. Chem.*, 1988, **9**, 650–661.
33. Weiner, S. J., Kollman, P. A., Case, D. A., Singh, U. C., Ghio, C., Alagona, G., Profeta, S. Jr. and Weiner, P. K., A new force field for molecular mechanical simulation of nucleic acids and proteins. *J. Am. Chem. Soc*., 1984, **106**, 765–784.
34. Jayaram, B., Sharp, K. A. and Honig, B., The electrostatic potential of B-DNA. *Biopolymers*, 1989, **28**, 975–993.
35. Nicholls, A. and Honig, B., A rapid finite difference algorithm, utilizing successive over-relaxation to solve the Poisson–Boltzmann equation. *J. Computat. Chem.*, 1991, **12**, 435–445.
36. Andersen, H. C., Molecular dynamics simulations at constant pressure and/or temperature. *J. Chem. Phys.*, 1980, **72**, 2384–2393.
37. Berendsen, H. J. C., Postma, J. P. M., van Gunsteren, W. F., DiNola, A. and Haak, J. R., Molecular dynamics with coupling to an external bath. *J. Chem. Phys.*, 1984, **81**, 3684–3690.
38. Morris, G. M., Goodsell, D. S., Halliday, R. S., Huey, R., Hart, W. E., Belew, R. K. and Olson, A. J., Automated docking using a lamarckian genetic algorithm and empirical binding free energy function. *J. Computat. Chem*., 1998, **19**, 1639–1662.
39. Goodman, J. M., *Chemical Applications of Molecular Modelling*, The Royal Society of Chemistry, London, 1998, pp. 61–69.
40. Good, A. C., Hodgkin, E. E. and Richards, W. G., Utilization of Gaussian functions for the rapid evaluation of molecular similarity. *J. Chem. Inf. Comput. Sci.*, 1992, **32**, 188.
41. Jones, G., Willet, P. and Glen, R. C., A genetic algorithm for flexible molecular overlay and pharmacophore elucidation. *J. Comput.-Aided Mol. Des*., 1995, **9**, 532.
42. Kurogi, Y. and Guner, O. F., Pharmacophore modeling and three-dimensional database searching for drug design using catalyst. *Curr. Med. Chem*., 2001, **8**, 1035–1055.

Received 17 September 2005; revised accepted 26 October 2006