SIMULATION OF DILUTE-SOLUTION PROPERTIES OF BIOLOGICAL MACROMOLECULES WITH THE AID OF HIGH-PERFORMANCE COMPUTING

R. Rodríguez Schmidt[(a)], D. Amorós Cerdán[(b)], J.G. Hernández Cifre[(c)], F.G. Díaz Baños[(d)], J. García de la Torre[(e)]

[(a,b,c,d,e)] Dep. Química Física, Universidad de Murcia, 30100 Murcia, Spain

[(a)]ricrogz@um.es, [(b)]damoros@um.es, [(c)]jghc@um.es, [(d)]fgb@um.es, [(e)]jgt@um.es

ABSTRACT

The determination of dilute solution properties of macromolecules (hydrodynamic coefficients, radius of gyration and scattering-related properties, NMR and viscoelastic relaxation, etc) is of interest to characterize their conformation (size and shape) and dynamics in usual working environment (e.g. physiological conditions in case of biomacromolecules). Over the years, the Polymer Group at the University of Murcia has made computational developments intended for the prediction of dilute solution properties of synthetic polymers and biological macromolecules. Most of these developments are of public domain (see our web site http://leonardo.inf.um.es/macromol).

In this work, we present some improvements in our methodology aimed to achieve high-performance computing. The strategy is based on using parallelized versions of the LAPACK and similar mathematical libraries, and implementing in-house written codes. We show some results obtained by applying that methodology to some macromolecular models.

Keywords: bead model, biomacromolecule, conformation, hydrodynamic, parallel computing

1. INTRODUCTION

The calculation of conformational and hydrodynamic properties (like radius of gyration, sedimentation and diffusion coefficients, or intrinsic viscosities) of rigid and flexible macromolecules (like globular and denatured proteins respectively) is frequently based on constructing bead models that represent the size and shape of the particle (see Fig. 1) and then apply some fundamental hydrodynamic equations for the sphere to get the global particle properties. Those calculations involve linear-algebra problems, essentially the solutions of sets of linear (hydrodynamic interaction) equations (Carrasco and García de la Torre 1999), in which the number of unknowns is proportional to the number of elements in the model. This number has to be sufficiently large as to describe the complex structural details typical of biological macromolecules, particularly when modelling is done at atomic level (García de la Torre, Huertas, and Carrasco 2000), which quite often entails a large computation time.

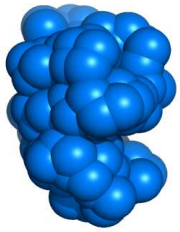

Figure 1: Example of bead model for lysozyme

The increasing number of cores in a processor and the availability of computer clusters, open the possibility of parallelizing some stages of the calculus and speeding up the global simulation process by running several simulations simultaneously. That is achieved by dividing the global problem into independent pieces that can be processed at the same time and then collect the results. Thus, an update in the software developed by our group is in order to embody some of those features and improve its efficiency.

We present in this work some strategies leading to easily implement parallelization with minor changes in our codes, mainly focused to run our programs under multi-core processor computers. On the one hand, we have replaced some of our homemade functions and subroutines by some others included in public domain mathematical libraries like LAPACK that are optimized to carry out parallel calculation if required. On the other hand we have built a set of ancillary tools intended to run a number of generations (cases) of some model in several cores simultaneously.

2. METHODOLOGY FOR RIGID PARTICLES

Conformational and hydrodynamic properties of rigid structures like many globular proteins can be calculated using our HYDRO programs suite (Carrasco and García de la Torre 1999). Nowadays, there is a big amount of accessible experimental information about the atomic structure of proteins, nucleic acids and many other biopolymers, which can be used to model those molecules and subsequently calculate solution properties. Our program HYDROPRO (García de la

Torre, Huertas, and Carrasco 2000; Ortega, Amorós, and García de la Torre 2011) is able to construct bead models based upon the atomic-level information contained in the Protein Data Bank that is a public domain data base for proteins and other related complex structures (www.rcsb.org/pdb/home/home.do).

The program places a sphere per atom (except hydrogen) with a radius that includes the hydration layer. Then, and after some model simplifications, it calculates basic solution properties of the structure: radius of gyration, diffusion and sedimentation coefficients, intrinsic viscosity, and relaxation times. Those calculations involve time consuming matrix and vector operations that can be parallelized by using public domain mathematical libraries. In particular, the most time consuming mathematical operation is the inversion of the hydrodynamic interaction tensor, the dimensions of which, for a model made of N beads, is 3Nx3N. Because the CPU time of such an operation scales as N^3, the calculation becomes slow quite soon.

Other aspect to take into account is the memory required to store such a big tensor. Because of the symmetry of the hydrodynamic tensor, it is possible to store just the elements in the upper or lower triangle of the matrix (what is called a packed matrix). Sometimes that strategy slows down the calculation and it is preferred to work with the whole square matrix. The program HYDROPRO has been rewritten to improve both memory management and calculation speed. For that purpose, subroutines to perform LU and Cholesky matrix factorization included in the LAPACK mathematical library (www.netlib.org) have been employed. On the other hand, bead models of complex structures like that in Fig. 1 for the protein lysozyme (so-called shell models) were generated starting from atomic-level structures taken from the Protein Data Bank.

Fig. 2 is a histogram showing the CPU time consuming by different versions of HYDROPRO in calculating a set of solution properties when running on a processor Intel Xeon x5660 6 cores (x2 CPU).

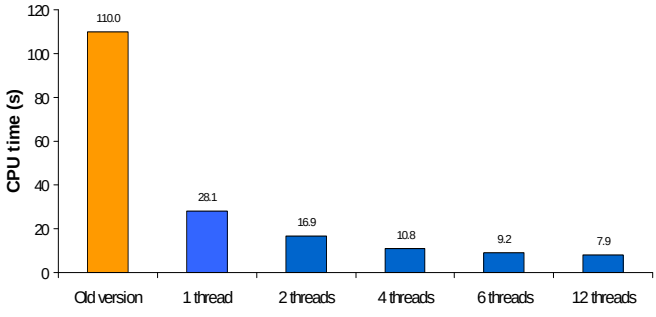

Figure 2: Gain in CPU time due to parallelization

The first bar on the left, corresponding to our old version of HYDROPRO, is exceedingly large in comparison with the other bars that measure the CPU time consumed by the parallel version of HYDROPRO that includes the possibility of using several calculation threads at a time. As appreciated, we can diminish the duration of the process more than ten times.

Experimental and calculated values for the radius of gyration, Rg, and the sedimentation coefficient, s, of several complex structures are compared in Table 1. As observed, the agreement is quite good (as it already occurred with our old version).

Table 1: Comparison of solution properties of complex structures calculated by HYDROPRO with their experimental values: [a]Tomonao et al, Biophys J 2008, 94, 1392-1402; [b]Behlke et al, Biochemistry 1997, 36, 5149-5156; [c]Armstrong et al, Biophys J 2004, 87, 4259-4270; [d]Hill et al, J Mol Biol 1969, 44, 263-277.

Structure	**Rg / Å**		**sx10^{13}/ s**	
	Calc.	Exp.	Calc.	Exp.
Chaperone GroEL	66.1	67.0[a]	21.5	22.13[b]
IgM antibody	127	121[c]	18.4	17.5[c]
Ribosme 30S	68.3	69[d]	36.9	31.8[d]
Ribosme 50S	74.6	77[d]	53.6	50.2[d]
Ribosme 70S	86.0	91.5[d]	69.3	70.5[d]

3. METHODOLOGY FOR FLEXIBLE CHAINS

The skeleton of the long and (usually) linear chains that constitutes biological macromolecules consists of chemical bonds arranged in such a manner that bond lengths and bond angles are nearly constant, but there is an important degree of freedom in the internal rotation around all or some of the bonds in the chain. The consequence of such conformational variability about each of so many bonds along the chain skeleton is that it could adopt multiple conformations like it is the case of denatured proteins. With a dynamic point of view, a single macromolecule is continuously changing its own conformation. Thus macromolecular chains are, in principle, essentially flexible entities that can be represented by bead and connector models like that in Fig. 3.

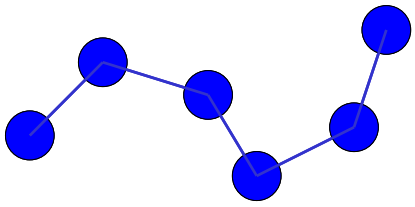

Figure 3: Example of bead and connector model

The theoretical foundation for the description of flexible chain macromolecules has its origin in the pioneering works of Kirkwood and Zimm (Kirwood and Riseman 1948; Zimm 1956).

As in the bead models used for rigid-body modeling, in the mechanic model of flexible particles, the parts composing the particle are represented by spherical beads, which is a convenient representation for hydrodynamic calculations. Instead of having a unique shape, the beads in the array are linked by a series of internal interactions that determine the conformational variability of the ensemble.

Firstly, there must be some connectors between (neighbor) beads, like the bonds in a chemical molecule. They behave as springs of some degree of flexibility, and are the primary interactions in the mechanic model, which -- regardless of the instantaneous conformation -- determine its topology (linear, branched, etc). Additionally, there can be other short-term interactions, mainly bending interactions between two neighbor connectors, and perhaps torsional or internal-rotation restrictions. Also, it may be present long-term interactions between non-connected beads (so-called excluded volume interactions) which are properly represented by several potential expressions depending on the features of the system (hard sphere potential, Lennard-Jones potential…)

The conformational statistics of flexible macromolecules can be simulated by a standard Monte Carlo (MC) procedure. However, the full, rigorous simulation of the macromolecular dynamic behavior has to be done by means of Brownian dynamics (BD).

3.1. Monte Carlo

Generically speaking, the MC simulation methods are intended to generate possible states of a system, which in our case are conformations of a flexible macromolecule, in order to obtain observable properties that correspond to averages over all the accessible conformations or properties. The various states have different probabilities, which are proportional to the Boltzmann factor, exp(-V/kT), associated to the potential energy of the system in such conformation, V (being k the Boltzmann constant and T the absolute temperature). If the possible conformations are generated in an absolutely random manner, this factor should be used as a statistical weight in the evaluation of the averages.

A usual choice to carry out a MC simulation is the importance-sampling procedure which is quite simple. From a previous conformation of the chain, a new one is generated making small random displacements of the beads. The potential energy of the new conformation, V, is evaluated as a sum of the several contributions from the various kinds of intramolecular or external interactions that the mechanic model may include. The new potential is compared to that of the previous conformation, Vprev. The new conformation is accepted if V < Vprev. Otherwise, if exp[(Vprev-V)/kT] > u, where u is a random number with uniform distribution in (0,1) generated each time that this decision is to be made, then the new conformation is accepted. If exp[(Vprev-V)/kT] < u, the conformation is rejected and the resulting conformation after the MC step is a copy of the previous one.

For MC simulations of bead-and-connector models, we have developed the public domain MONTEHYDRO program (García de la Torre, Ortega, Pérez Sánchez, and Hernández Cifre 2005) that implements an importance-sampling Monte Carlo simulation coupled to rigid-body hydrodynamics which is based on the procedures of the above mentioned HYDRO programs suite. Thus, the hydrodynamic coefficients are calculated using the "Monte-Carlo rigid-body" approach (Zimm 1980): the macromolecular conformations generated by MC are considered as instantaneously rigid and the rigid body hydrodynamic equations are applied to each one. Then a conformational average is performed in order to obtain global equilibrium properties.

Fig. 4 shows the excellent agreement obtained between experimental and simulation data when comparing the radius of gyration of a set of flexible proteins with different number of residues, N, (Amorós 2012). Simulations were carried out using the MONTEHYDRO program including parallelization. The model included excluded volume interactions represented by a hard sphere potential where the hard sphere radius was r_{HS}=2.25 Å.

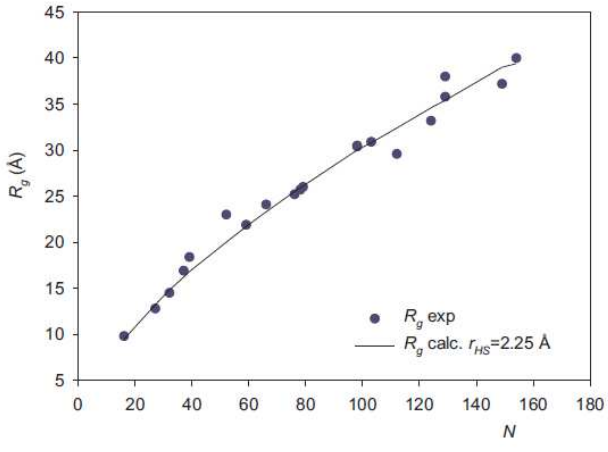

Figure 4: Rg of proteins: experimental (dots) and calculated by MONTEHYDRO (line)

3.2. Brownian dynamics

To study dynamic aspects of flexible macromolecules in solution, such as relaxation processes and non-equilibrium behavior, it is necessary to solve the

equation of motion that governs the macromolecular dynamics. This can be done by using molecular dynamics (MD) or Brownian dynamics (BD). MD is not adequate for long time and size scales. BD is a numerical technique to solve the stochastic equation of motion that arises from considering the solvent as a continuum, thus eliminating the solvent degrees of freedom. In other words, BD simulations describe the Brownian motion of a collective of frictional elements, beads in our model, which can interact with each other through different potentials.

An essential aspect in the BD simulation is the inclusion of the so-called hydrodynamic interaction (HI) effect, which determines the solvent-mediated influence of the motion of every element of the model on the others. A first order algorithm to solve the stochastic equation of motion and performed Brownian dynamics simulations including HI is that of Ermak and MacCammon (Ermak and McCammon 1977).

We have developed a BD simulation scheme that enables for the calculation of solution properties of flexible macromolecules with arbitrary complexity. Our procedures take into account fluctuating (non-preaveraged) hydrodynamic interaction as well as the possibility of including different types of intra-molecular potentials to represent excluded volume conditions and electrostatic interactions. That computational scheme is implemented in a suite of public domain programs, named SIMUFLEX. The suite consists mainly of two programs: BROWFLEX and ANAFLEX. The program BROWFLEX generates a Brownian trajectory of a flexible bead-and-connector model with arbitrary connectivity, and the program ANAFLEX analyses that trajectory to obtain several steady and time-dependent macromolecular quantities (García de la Torre, Hernández Cifre, Ortega, Rodríguez Schmidt, Fernandes, Pérez Sánchez, and Pamies 2009).

However, BD algorithms with fluctuating HI demands long simulation (CPU) time, which increases dramatically with the number of elements forming the chain model, N. The main reason is that the BD-HI methodology requires the calculation of the square root of the 3Nx3N HI tensor, a very time-consuming operation carried out every time that tensor is updated during the simulation. An exact mathematical procedure to get the square root of the symmetric matrix is the already mentioned Cholesky factorization. Some approximations have been proposed to speed up BD-HI simulations. Those mathematical approaches are able to increase enormously the efficiency of BD-HI simulations (Rodríguez Schmidt, Hernández Cifre, and García de la Torre 2011), moreover when they are used in combination with parallelization strategies for the matrix and vector operations.

It is worth to see how we can gain in CPU time due to the mathematical approaches themselves without using parallel computing. Fig. 5 shows the CPU time needed for the BD-HI simulations of linear Gaussian chains with excluded volume with increasing number of beads. For that purpose, we have measured the duration of 10^6 Brownian steps on a 3.0 GHz Intel Xeon Quad-Core L5450. As appreciated, the use of Geyer (Geyer and Winter 2009) and Kröger (Kröger, Alba Pérez, Laso, and Öttinger 2000) algorithms, when applicable, entails a fabulous gaining in CPU time respect to those of Jendrejack (Jendrejack, Graham, and de Pablo 2000) and Cholesky (Ermak and McCammon 1977). (Note: algorithms are named here after the first author of the paper where they were presented, except that of Cholesky that refers to algorithms that use the rigorous Cholesky factorization).

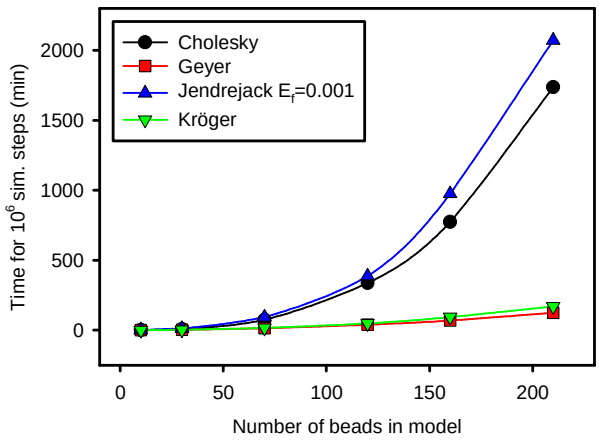

Figure 5: Runtime behavior of different BD-HI algorithms

4. DISTRIBUTION OF A GLOBAL TASK

Nowadays, even processors of modest PCs are multi-core (e.g. Dual, or Quad) and it is common to work with multiprocessor units, or even clusters of many units. Then, if the kind of problem under simulation study meets some requirements, one can make use of these resources in order to split a large task, like the generation of a macromolecular trajectory, into many shorter independent trajectories that will run in parallel using all of the available processors. The final results will be produced after a proper combination of the partial results obtained from those short independent trajectories. Obviously, that simulation strategy reduces the time needed to solve the global task approximately so many times as the number of available processors.

With the aim of dividing a global task into smaller independent ones, we are designing the programs suite MULTISIMUFLEX, a set of ancillary tools to be used along with MONTEHYDRO and SIMUFLEX. Thus, one can take advantage of multi-core computers and distribute through the available computing cores the task of generating conformations of bead-and-connector models either via Monte Carlo or via Brownian dynamics. In order to generate independent trajectories we resort to two procedures: a) the use of different initial conformations for each sub-simulation, b) the use of different sequence of random numbers for each sub-simulation. The MULTISIMUFLEX suite is not available in our web site yet.

We have carried out several tests to verify the right work of MULTISIMUFLEX. Fig. 6 is a sketch of a 'homemade' tetra-block A-B-C-B copolymer that represents a chimerical protein. It consists of a globular domain plus a hanging chain having an internal helix. We ran a large Brownian dynamics simulation by dividing it into independent simulations. Thus, the CPU time needed for the whole dynamics and then getting accurate values of the steady-state properties is reduced by an order of magnitude. Table 2 shows the values of the analyzed properties.

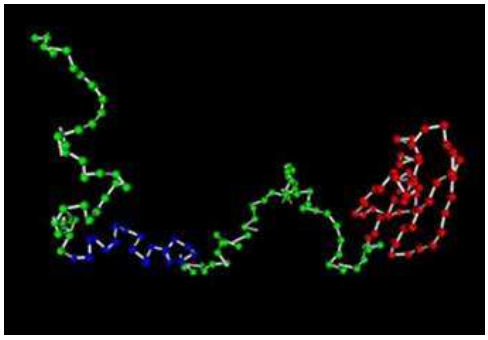

Figure 6: Bead-connector model of chimerical protein

Table 2: properties obtained for the chimerical protein after running a simulation using MULTISIMUFLEX.

Radius gyration (cm)	Diffusion coefficient (cm^2/s)	Intrinsic viscosity (cm^3/g)
2.73×10^{-7}	9.622×10^{-7}	4.726

5. CONCLUSIONS

It is clear that the use of multiprocessor units speed up the calculation of solution macromolecular properties without loss of precision. We have presented here some simple parallelization strategies used to improve the efficiency of our public-domain programs available at our web site http://leonardo.inf.um.es/macromol.

One strategy is the substitution of parts of our code by equivalent parallelized functions obtained from public-domain mathematical libraries like LAPACK. Thus, a faster version of our program HYDROPRO to calculate solution properties of rigid structures is now available. Other possibility is distributing efficiently the calculation among every available processor. In order to help in the task of splitting a large simulation into shorter ones that will run in parallel, we have designed the suite MULTISIMUFLEX. That program must be used in combination with our MONTEHYDRO and SIMUFLEX programs which are used to run Monte Carlo and Brownian dynamics simulations of flexible structures and then get their solution properties.

The above commented parallelization strategies along with the improvement of mathematical aspects of the simulation algorithms were tested by applying them to several bead models of proteins. It was verified that those strategies are able to yield excellent results for the solution properties and diminish the computational time more than one order of magnitude, thus increasing enormously the performance of our algorithms.

ACKNOWLEDGMENTS

This work was performed within a Grupo de Excelencia de la Región de Murcia (grant 04531/GERM/06). Support also provided by grant CTQ-2009-06831 from Ministerio de Educación y Ciencia, including FEDER funds.

REFERENCES

Amorós, D., 2012, *Metodologías para la predicción de propiedades conformacionales y dinámicas de macromoléculas biológicas en disolución*. PhD. Thesis, University of Murcia.

Carrasco, B., García de la Torre, J., 1999, Hydrodynamic Properties of Rigid Particles. Comparison of Different Modelling and Computational Procedures. *Biophysical Journal* 76, 3044–3057.

Ermak, D..L., McCammon, J.A., 1978, Brownian dynamics with hydrodynamic interactions. *Journal of Chemical Physics* 69, 1352–1360.

García de la Torre, J., Huertas, M.L., Carrasco, B., 2000, Calculation of Hydrodynamic Properties of Globular Proteins from their Atomic-Level Structure. *Biophysical Journal* 78, 719–730.

García de la Torre, J., Ortega, A., Pérez Sánchez, H.E., Hernández Cifre, J.G., 2005, MULTIHYDRO and MONTEHYDRO: Conformational search and Monte Carlo calculation of solution properties of rigid and flexible macromolecular models. *Biophysical Chemistry* 116, 12–128.

García de la Torre, J., Hernández Cifre, J.G., Ortega, A., Rodríguez Schmidt, R., Fernandes, M.X., Pérez Sánchez, H.E., Pamies, R., 2009, SIMUFLEX: Algorithms and tools for simulation of the conformation and dynamics of flexible molecules and nanoparticles in dilute solution. *Journal of Chemical Theory and Computation* 5, 2606-2618.

Geyer, T., Winter, U., 2009, A O(N^2) approximation for hydrodynamic interactions in Brownian dynamics simulations. *Journal of Chemical Physics* 130, 1149051 (8 pages).

Jendrejack, R.M., Graham, M.D., de Pablo, J.J.., 2000, Hydrodynamic interactions in long chain polymers: Application of the Chebyshev polynomial approximation in stochastic simulations. *Journal of Chemical Physics* 113, 2894-2900.

Kirkwood, J.G.., Riseman, J., 1948, The intrinsic viscosities and diffusion constants of flexible macromolecules in solution. *Journal of Chemical Physics* 16, 565–573.

Kröger, M., Alba Pérez, A., Laso, M., Öttikger, H.C., 2000, Variance reduced Brownian simulation of a bead-spring chain under steady shear flow considering hydrodynamic interaction effects. *Journal of Chemical Physics* 113, 4767-4773.

Ortega, A., Amorós, D., García de la Torre, J., 2011, Prediction of hydrodynamic and other solution properties of rigid proteins from atomic- and residue-level models. *Biophysical Journal* 101, 892–898.

Rodríguez Schmidt, R., Hernández Cifre, J.G., García de la Torre, J., 2011, Comparison of Brownian dynamics algorithms with hydrodynamic interaction. *Journal of Chemical Physics* 135, 084116 (10 pages).

Zimm, B.H., 1956, Dynamics of polymer molecules in dilute solution: viscoelasticity, flow birefringence and dielectric loss. *Journal of Chemical Physics* 24, 269–277.

Zimm, B.H., 1980, Chain molecule hydrodynamics by the Monte-Carlo method and the validity of the Kirkwood-Riseman approximati. M*acromolecules* 13, 592–602.