Representing of Problem, Solution and Implementation Spaces with Interrelated Attributes for Developing Knowledge Management Base in Computational Chemistry Area

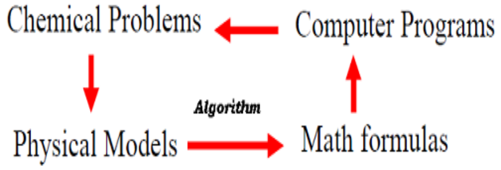
Sunil Khilari
IMED Research Center, Pune

Dr. Sachin Kadam
Bharti Vidyapeeth University, Pune

Abstract: The representation of solution, problem and implementation spaces with interrelated attributes which will help to the chemical and related scientific research and software development community. Scientific computing will help to address some of the challenges in computational chemistry and process systems particularly computational tasks that scale exponentially with various computational problems from chemistry domain. To design Knowledge Management (KM) base therefore has a lot to offer to a chemistry lab. Our main purpose is to identify computational problems of high priority to progress in chemical knowledge management initiatives that should be undertaken with support provided in the development of open source ICT tools for the computational chemistry domain. This paper presents algorithms, software development and computational complexity analysis for problems arising in the Computational Chemistry domain.

I. Computational Chemistry Computations

Use of methodical approximation and computer programs to obtain results relative to chemical problems. Computational chemistry is simply the application of chemical, mathematical and computing skills to the solution of interesting chemical problems. It uses computers to produce information such as properties of molecules and/or simulated experimental results. Some common computer software used for computational chemistry includes,Nlopt , MATLAB, TINKER , Gaussian etc.(8). Also computational chemistry is based on an approximations and assumptions. Computational Chemistry Calculate Energy, Structure, and Properties. Computations of this type are derived directly from theoretical principles, with no inclusion of experimental data. Mathematical approximations are usually a simple functional form for an approximate solution to a differential equation. A mathematical method that is sufficiently well developed that it can be automated for implementation on a computer.



The input data for these computational problems are laboratory experiments, where some lead compounds were identified. The problem is to generate new laboratory experiments that will accelerate the likelihood of discovering new, more powerful compounds/substances. In order to do so we have to solve inverse problems based on specific indices. One wants several solutions for the inverse problem that are as diverse (i.e. different chemical structure) as possible. Based on them, a new combinatorial library is created, and new lead compounds are discovered. (1).

II. Challenges in Computational Chemistry Domain

- Invent new computer tools and logistics methods to reduce significantly the time needed for commercializing new drugs.
- Invent new algorithms to globally optimize at the worldwide level the use of raw materials, energy, and environmental impact of chemical processes.

- Develop computer methods that will accurately predict the properties of unknown compounds.
- Invent computer methods to predict the three-dimensional folded structure of a protein—and the pathway by which folding occurs—from its amino acid sequence, so information from the human genome can be translated into the encoded protein structures.
- Develop reliable computer methods to calculate the detailed pathways by which reactions occur in both ground states and excited states, taking full account of molecular dynamics as well as quantum and statistical mechanics(8)
- Develop reliable force fields for molecular mechanics calculations on complex systems, including those with metallic elements.

III. ALGORITHMIC SOLUTIONS FOR COMPUTATIONAL PROBLEMS

Important features of algorithms are fitness, definiteness, input, output and effectiveness

- ✓ Finiteness -it must terminate after finite number of steps.
- ✓ Definiteness -It must have each and every step of procedure to be precisely defined.
- ✓ Input/output- Algorithm must communicate to the environment in which it operate
- ✓ Effectiveness - Algorithms must be practical .i.e. must be capable of implementation
- ✓ Degree of goodness -algorithms is of its speed of execution/generates correct result

Faster and cheaper computers will extend the range of high-level methods .To begins; we must understand that programs and algorithms are not the same thing. One is an implementation; the other is a mathematical description. One is associated with a machine, the other with a computational model. Why do we care about analysis of algorithms? We care about having efficient algorithms in terms of time and space. The computational models at the present time reflect quite well the behavior of an algorithm when it is implemented. Remember, the interest is in algorithms typically. We prove theorems and analyze algorithms, not their implementations.

Algorithmic understanding is defined as the ability to match up or recall an appropriate mathematical formula and a Strategy to compute a numerical answer (5) .We begin with an overview of the algorithm.

The entire computational chemistry problems are solved with the help of algorithmic solutions. Some of these solutions are very crude and others are expected to be more accurate than any experiment that has yet been conducted. There are several implications of this situation.

1. Computational chemistry end-users require knowledge of each algorithm being used and how accurate the results are expected to be.
2. Obtaining very accurate results requires extremely powerful computers.
3. If the equations can be solved analytically, much of the work now done on supercomputers could be performed faster and more accurately on a PC.

Computational chemistry has become a useful way to investigate materials that are too difficult to find or too expensive to purchase. It also helps chemists make predictions before running the actual experiments so that they can be better prepared for making observations.

Following are the list of computational chemistry algorithms identified and used for solving various computational problems. Some of algorithms are implemented with the help of ICT tools.

SrNo	Computational Chemistry Algorithm	SrNo	Computational Chemistry Algorithm	SrNo	Computational Chemistry Algorithm
1	ab initio Algorithm	8	EMBED Algorithm	15	Molecular Recognition and Docking Algorithms
2	Accelerated Random Search (ARS) Algorithm	9	Euclidean Algorithm	16	Monte Carlo Algorithm
3	ACRB algorithm	10	Fletcher±Powell (FP) Algorithm	17	Morgan Algorithm
4	ACRN Algorithm	11	Floyd’s Algorithm	18	Morgan's Algorithm
5	Adaptive Substituent Reordering Algorithm (ASRA)	12	Floyd-Warshall Algorithm	19	MTL Algorithm
6	AHM Algorithm	13	Fruchterman and Reingold Algorithm	20	Nelder-Mead (NM) Algorithm
7	Arvis-Patrick Algorithm	14	Gear Algorithm	21	NEO Algorithm

Table No.2 List of Algorithms used in Computational Chemistry domain

IV. ICT SOLUTION OF COMPUTATIONAL PROBLEMS OF CHEMISTRY

There are new potential applications from software and Internet-based computing. Many computational chemistry techniques are extremely computer-intensive. Depending on the type of calculation desired, it could take anywhere from seconds to weeks to do a single calculation. There are many calculations, such as ab initio analysis of biomolecules that cannot be done on the largest computers in existence. Likewise, calculations can take very large amounts of computer memory and hard disk space. In order to complete work in a reasonable amount of time, it is necessary to understand what factors contribute to the computer resource requirements. Ideally, the user should be able to predict in advance how much computing power will be needed.(8).

The demand for software development may increase as chemistry and chemical engineering move to new areas in which there are no standard software packages. For Internet-based computing an exciting possibility will be to share more readily new software developments directly from the developers, bypassing the commercial software vendors. Another area of sharing leading to powerful new computational opportunities in the chemical sciences is the use of peer-to-peer computing in the form of sharing unused cycles on small computers. We can do very large-scale computations on networks of personal computers, as is being done in studies of protein folding9 and molecular docking. The potential here is to tackle computational problems of unprecedented size and complexity, with a relatively low investment in the actual computational resources.

The description of the advantages and limitations of each software package is again a generalization for which there are bound to be exceptions. Some software packages can be run on a networked cluster of workstations as though they were a multiple-processor machine. However, the speed of data transfer across a network is not as fast as the speed of data transfer between them.

Sr.No.	Algorithm	ICT Tool	ICT Tool Properties
1	ab initio Algorithm	Nlopt 2.4.2	Unix Open Source Fortran
2	Beeman's Algorithm	TINKER 2.0	DOS Open Source Fortran77
3	Berny Algorithm	Gaussian 3.0	Windows Open Source C
4	NOVA Algorithm	YASARA	Linux Open Source Android

Table No.4 Computational Chemistry algorithms-ICT implementation software's and software properties

The most user-friendly software packages require little more work than a molecular mechanics calculation. The price for this ease of use is that the program uses many defaults, which may not be the most appropriate for the needs of a given research project.

V. REPRESENTATION OF PROBLEM ,SOLUTION AND IMPLEMENTATION OF ALGORITHMS

Advances in scientific computing will help to address some of the challenges in computational chemistry and process systems engineering, particularly computational tasks that scale exponentially with size. While single-threaded execution speed is important and needed, coordination of multiple instruction multiple data (MIMD) computer systems is rapidly becoming the major challenge in scientific computing.

Holistic Analysis of Problem spaces corresponding algorithmic solution and ICT Implementation.

SrNo	Computational Problem Category	Problem Spaces	Algorithmic Solution	ICT Implementation	ICT Tool Properties		
1	Reaction Mechanism	chemical Reaction Balancing	ACRB algorithm	MATLAB	Empirical	trial-and-error	Unix
2	Chemical Reactivity	Valance Calculation	ab initio Algorithm	Nlopt 2.4.2	open-source	Fortran	Unix
3	Molecular Simulation	designed to allow high numbers of particles in simulations of molecular dynamics	Beeman's Algorithm	TINKER 2.0	open-source	Fortran77	DOS
4	Geometry optimization	identify linear connection between gradient and coordinate changes	Berny Algorithm	Gaussian 3.0	open-source		Windows
5	Chemical Modeling	Studying the macroscopic and experimental influences on microscopic structure chemical kinetics and thermal decomposition	NOVA Algorithm	YASARA	open-source	Android	Linux

Table No.5 –Holistic analysis of CC Problems, Algorithms and ICT software with their properties

This study may helpful for changing the nature of computational chemistry software development.

We have to stop learning how to make molecules or materials. There’s plenty more to do in the area, even without considering flow chemistry, cascade reactions, and other such dynamic synthetic strategies (10).

VI. CONCLUSION

Computational chemistry and process systems engineering play a foremost role in providing new understanding and development of computational procedures for the simulation, design, and operation of systems ranging from atoms and molecules to industrial-scale processes. Knowledge management (KM) aims to maximize efficiency, nurture creativity, and even enhance coincidence in computational chemistry.

The applications of standard software engineering approaches like automatic code generation, simplified the code, reduce the development, testing and debugging time. These are may be the challenge's to specific development of computational chemistry software. It is now so easy to do computational chemistry that calculations can be performed with the knowledge of the underlying principles. As a result, many people do not understand even the most basic concepts involved in a calculation. To date, the field has neither sufficient tools nor enough trained people to pursue computational chemistry and chemical engineering across all these scales. The field will qualitatively change—in new insights, in what experiments are done and how chemical products and processes are designed—when this is achievable.

REFERENCES

[1] Algorithmic Strategies in Combinatorial Chemistry, Deborah Goldman,Sorin Istrail

[2] R. Leach, Molecular Modelling, 2nd Ed.,Prentice Hall, 2001

[3] J. Cramer, Essentials of Computational Chemistry, 2nd Ed., Wiley, 2004

[4] Bemis, Guy W. and Kuntz, Irwin D., A fast and efficient method for 2D and 3D molecular shape description, J. Computer-Aided Molecular Design, Vol.6 (1992) 607-628

[5] BAYRAM COSTU,Algorithmic, Conceptual and Graphical Chemistry Problems: A Revisited Study, Asian Journal of Chemistry, Vol. 22, No. 8 (2010), 6013-6025

[6] F. Jensen, Introduction to Computational Chemistry John Wiley & Sons, New York (1999).

[7] David C. Young,Computational Chemistry: A Practical Guide for Applying Techniques to Real-World Problems.Copyright 2001 John Wiley & Sons, Inc.ISBNs: 0-471-33368-9 (Hardback); 0-471-22065-5 (Electronic)

[8] H.S. Chan, and K.A. Dill, Physics Today, 46 (2): 24-32 Feb. 1993

[9] Robinson, Daniel D.; Lyne, Paul D.; Richards, W. Graham, J. Chem. Inf. Comput. Sci., 40(2), 503-512, 2000.

[10] Bruce C. Gibb is in the Department of Chemistry at Tulane University, New Orleans, Louisiana 70118, USA. Nature Chemistry | VOL 4 | October-2012 | www.nature.com/naturechemistry