

Introduction to Computational Chemistry

Laboratory

Table of Contents

- 1. Introduction. Overview of computational chemistry.**
- 2. Theoretical background of computational chemistry.**
 - **Ab-initio methods for electronic structure calculations**
 - **Semiempirical calculations**
 - **Molecular mechanics approach**
 - **Molecular dynamics method**
 - **Statistical mechanics and Thermodynamics**
 - **Structure-Property Relationships**
 - **Symbolic calculations**
 - **Artificial Intelligence**
- 3. How to do a computational research project (lab).**
- 4. TAU Computational Chemistry Laboratory.**
 - **Hardware and Software for computations and visualization available in TAU Computational Chemistry Laboratory.**
- 5. Summary**
- 6. Further information and references**

1. Introduction.

Overview of Computational Chemistry.

The term **theoretical chemistry** may be defined as the mathematical description of chemistry.

Currently, there are two ways to approach theoretical chemistry problems: computational theoretical chemistry and non-computational theoretical chemistry.

Computational theoretical chemistry is primarily concerned with the numerical computation of molecular electronic structures and molecular interactions and non-computational quantum chemistry deals with the formulation of analytical expressions for the properties of molecules and their reactions.

The term **computational chemistry** is usually used when a mathematical method is sufficiently well developed that it can be automated for implementation on a computer. Computational chemistry is the application of chemical, mathematical and computing skills to the solution of interesting chemical problems. It uses computers to generate information such as properties of molecules or simulated experimental results. Very few aspects of chemistry can be computed exactly, but almost every aspect of chemistry has been described in a qualitative or approximate quantitative computational scheme. **The biggest mistake** that computational chemists can make is to assume that any computed number is exact. However, just as not all spectra are perfectly resolved, often a qualitative or approximate computation can give useful insight into chemistry if you understand what it tells you and what it doesn't.

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.

The quantum and classical mechanics as well as statistical physics and thermodynamics are the foundation for most of the computational chemistry theory and computer programs. This is because they model the atoms and molecules with mathematics. Using computational chemistry software you can in particular perform:

- electronic structure determinations,
- geometry optimizations,
- frequency calculations,
- definition of transition structures and reaction paths,
- protein calculations, i.e. docking,
- electron and charge distributions calculations,
- calculations of potential energy surfaces (PES),
- calculations of rate constants for chemical reactions (kinetics)
- thermodynamic calculations- heat of reactions, energy of activation, etc
- calculation of many other molecular and bulk physical and chemical properties.

The most important numerical techniques are ab-initio, semi-empirical and molecular mechanics. Definitions of these terms are helpful in understanding the use of computational techniques for chemistry:

- ab-initio, (Latin for "from scratch") a group of methods in which molecular structures can be calculated using nothing but the Schrödinger equation, the values of the fundamental constants and the atomic numbers of the atoms present.
- Semi-empirical techniques use approximations from empirical (experimental) data to provide the input into the mathematical models.
- Molecular mechanics uses classical physics and empirical or semi-empirical (pre-determined) force fields to explain and interpret the behavior of atoms and molecules.

The table below attempts to capture the main specifics of each of these three methods:

Method Type	Features	Advantages	Disadvantages	Best for
Molecular Mechanics	<ul style="list-style-type: none"> • Uses classical physics • Relies on force-field with embedded empirical parameters • Computationally least intensive - fast and useful with limited computer resources • Can be used for molecules as large as enzymes 	<ul style="list-style-type: none"> • Relies on potentials that have to be somehow supplied • Sometimes inaccurate because the supplied potentials are used beyond their proven range of validity 	<ul style="list-style-type: none"> • Particular force field, applicable only for a limited class of molecules • Does not calculate electronic properties • Requires experimental data (or data from <i>ab initio</i> calculations) 	<ul style="list-style-type: none"> • Large systems (~1000 of atoms) • Systems or processes with no breaking or forming of bonds
Semi-Empirical	<ul style="list-style-type: none"> • Uses quantum physics • Uses experimentally derived empirical parameters • Uses many approximation 	<ul style="list-style-type: none"> • Less demanding computationally than <i>ab initio</i> methods • Capable of calculating transition states and excited states 	<ul style="list-style-type: none"> • Requires experimental data (or data from <i>ab initio</i>) for parameters • Less rigorous than <i>ab initio</i>) methods 	<ul style="list-style-type: none"> • Medium-sized systems (hundreds of atoms) • Systems involving electronic transition
<i>Ab Initio</i>	<ul style="list-style-type: none"> • Uses quantum physics • Mathematically rigorous, no empirical parameters • Uses approximation extensively 	<ul style="list-style-type: none"> • Useful for a broad range of systems • does not depend on experimental data • Capable of calculating transition states and excited states 	<ul style="list-style-type: none"> • Computationally expensive 	<ul style="list-style-type: none"> • Small systems (tens of atoms) • Systems involving electronic transition • Molecules without available experimental data • Systems requiring rigorous accuracy

In the next chapter these and some other basic theoretical methods, both numerical and analytical, will be described in more details.

2. Theoretical background of Computational Chemistry

- **Ab-initio methods for electronic structure calculations**

The most common type of ab-initio calculation is called Hartree-Fock calculation (abbreviated HF), in which the primary approximation is called the mean field approximation. This means that the Coulombic electron-electron repulsion is not explicitly taken into account, however, its average effect is included in the calculation. This is a variational calculation, which implies that the approximate energies calculated are all equal to or greater than the exact energy. The accuracy of the calculation depends on the size of the basis set used, however because of the mean field approximation, the energies from HF calculations are always greater than the exact energy and tend, with increasing basis size, to a limiting value called the Hartree-Fock limit.

An additional issue that affects the accuracy of the computed results is the form chosen for the basis functions. The actual form of the single electronic molecular wave function (molecular orbital) is of course not known. The forms, used for the basis functions, can provide a better or worse approximation to the exact numerical single electron solution of the HF equation. The basis functions used most often are combinations of either Slater type orbitals ($\exp(-ax)$) or Gaussian type orbitals ($\exp(-ax^2)$), abbreviated STO and GTO. Molecular orbital is formed from linear combinations of atomic orbitals, which are nothing more than linear combinations of basis functions with coefficients founded from the appropriate atomic HF calculations. Because of this approximation, most HF calculations give a computed energy greater than the Hartree-Fock limit. The exact set of basis functions used is often specified by an abbreviation, such as STO-3G or 6-311++g**. In the Appendix 1 to this manual you can read about structures and features of some popular basis sets.

More advanced calculations begin with a HF calculation then correct for correlations that result from the electron-electron repulsion. Some of these methods are Many-Body Perturbation Theory (MBPT- n , where n is the order of correction), the Generalized Valence Bond (GVB) method, Multi-Configurations Self Consistent Field (MCSCF), Configuration Interaction (CI) and Coupled Cluster theory (CC). As a group, these methods are referred to as correlated calculations.

Another method, which avoids making the HF mistakes in the first place is called Quantum Monte Carlo (QMC). There are several flavors of QMC, variational, diffusion and Green's functions. These methods work with an explicitly correlated wave function and evaluate integrals numerically using a Monte Carlo integration. These calculations can be very time consuming, but they could yield extremely accurate results.

An alternative ab-initio method is Density Functional Theory (DFT), in which the total energy is expressed in terms of the total electron density, rather than the wavefunction. This type of calculation leads to an approximate effective or model Hamiltonian and to an approximate expression for the total electron density. Often the electron density approximations are made using some empirical corrections. Such DFT approaches are related to semi-empirical methods, which are addressed in the next chapter.

The good side of ab-initio methods is that they **eventually** converge to the exact solution, once all of the approximations are made sufficiently small in magnitude. However, this convergence is not monotonic. Sometimes, more approximate calculation gives better result for a given property, than a more elaborate calculation.

The bad side of ab-initio methods is that they are expensive. These methods often take enormous amounts of computer CPU time, memory and disk space. The HF method scales as N^4 , where N is the number of basis functions, so a calculation twice as big takes 16 times as long to complete. Correlated calculations often scale much worse than this. In practice, extremely accurate solutions are only obtainable when the molecule contains a few electrons.

In general, ab-initio calculations give very good qualitative results and can give increasingly accurate quantitative results as the molecules in question become smaller.

You can read more details about theoretical background of main ab-initio methods in the Appendix 2 to this manual.

Note on units: The energies calculated are usually in units called Hartrees (1 H = 27.2114 eV).

- **Semiempirical calculations**

Semiempirical calculations are set up with the same general structure as a HF calculation. Within this framework, certain pieces of information, such as two electron integrals, are approximated or completely omitted. In order to correct the errors introduced by omitting these parts of the calculation, the method is parameterized, by curve fitting in a few parameters or numbers, in order to give the best possible agreement with experimental data.

The good side of semiempirical calculations is that they are much faster than the ab-initio calculations.

The bad side of semiempirical calculations is that the results can be erratic. If the molecule under study is similar to molecules in the data base used to parameterize the method, then the results may be very good. If this molecule is significantly different from anything in the parameterization set, the answers may be poor.

Semiempirical calculations have been very successful in computational organic chemistry, where there are only a few elements used extensively and the molecules are of moderate size. However, semiempirical methods have been devised specifically for the description of inorganic chemistry as well.

- **Molecular Mechanics approach**

If a molecule is too big to effectively use a semiempirical treatment, it is still possible to model its behavior by avoiding quantum mechanics. This is done by constructing a simple expression for “molecular force field”, i.e. the potential energy as

function of all atomic positions, and using it study molecular properties without the need to compute a wave function or total electron density. The energy expression consists of simple classical equations, such as the harmonic oscillator equation in order to describe the energy associated with bond stretching, bending, rotation and intermolecular forces, such as Van der Waals interactions and hydrogen bonding. All of the constants in these equations must be obtained from experimental data or an ab-initio calculation.

In a molecular mechanics method, the database of compounds used to parameterize the method (a set of parameters and functions is called a force field) is crucial to its success. Where as a semiempirical method may be parameterized against a set of organic molecules, a molecular mechanics method may be parameterized against a specific class of molecules, such as proteins. Such a force field would only be expected to have any relevance to describing other proteins.

The good side of molecular mechanics is that it allows the modeling of enormous molecules, such as proteins and segments of DNA, making it the primary tool of computational biochemists.

The bad side is that there are many chemical properties that are not even defined within the method, such as electronic excited states. In order to work with extremely large and complicated systems, often molecular mechanics software packages have the most powerful and easiest to use graphical interfaces. Because of this, mechanics is sometimes used because it is easy, but not necessarily a good way to describe a system.

- **Molecular Dynamics method**

Molecular dynamics consists of examining the time dependent behavior of a molecule, such as vibrational motion or Brownian motion. This is most often done within a classical mechanical description similar to a molecular mechanics calculation.

The application of molecular dynamics to solvent/solute systems allows the computation of properties such as diffusion coefficients or radial distribution functions for use in statistical mechanical treatments. Usually the scheme of a solvent/solute calculation is that a number of molecules (perhaps 1000) are given some initial position

and velocity. New positions are calculated a small time later based on this movement and this process is iterated for thousands of steps in order to bring the system to equilibrium and give a good statistical description of the radial distribution function.

In order to analyze the vibrations of a single molecule, many dynamics steps are done, then the data is Fourier transformed into the frequency domain. A given peak can be chosen and transformed back to the time domain, in order to see what the motion at that frequency looks like.

- **Statistical Mechanics and Thermodynamics**

Statistical mechanics is the mathematical means to extrapolate thermodynamic properties of bulk materials from a molecular description of the material. Statistical mechanics computations are often tacked onto the end of ab initio calculations for gas phase properties. For condensed phase properties, often molecular dynamics calculations are necessary in order to do a computational experiment.

Thermodynamics is one of the best developed physical theories and it give us a good theoretical starting point for analysis of molecular systems. Very often any thermodynamic treatment is left for trivial pen and paper work since many aspects of chemistry are so accurately described with very simple mathematical expressions.

- **Structure-Property Relationships**

Structure-property relationships are qualitative or quantitative empirically defined relationships between molecular structure and observed properties. In some cases this may seem to duplicate statistical mechanical results, however structure-property relationships need not be based on any rigorous theoretical principles.

The simplest case of structure-property relationships are qualitative thumb rules. For example, an experienced polymer chemist may be able to predict whether a polymer will be soft or brittle based on the geometry and bonding of the monomers.

When structure-property relationships are mentioned in current literature, it usually implies a quantitative (which does not mean rigorous) mathematical relationship. These relationships are most often derived by using curve fitting software to find the linear combination of molecular properties, which best reproduces the desired property. The molecular properties are usually obtained from molecular modeling computations. Other molecular descriptors such as molecular weight or topological descriptions are also used.

When the property being described is a physical property, such as the boiling point, this is referred to as a Quantitative Structure-Property Relationship (QSPR). When the property being described is a type of biological activity (such as drug activity), this is referred to as a Quantitative Structure-Activity Relationship (QSAR).

- **Symbolic Calculations**

Symbolic calculations are performed when the system is just too large for an atom-by-atom description to be viable at any level of approximation. An example might be the description of a membrane by describing the individual lipids as some representative polygon with some expression for the energy of interaction. This sort of treatment is used for computational biochemistry and even microbiology.

- **Artificial Intelligence**

Techniques invented by computer scientists interested in artificial intelligence have been applied mostly to drug design in recent years. These methods also go by the names De Novo or rational drug design. The general scenario is that some functional site has been identified and it is desired to come up with a structure for a molecule that will interact with that site in order to hinder its functionality. Rather than have a chemist try hundreds or thousands of possibilities with a molecular mechanics program, the molecular mechanics is built into an artificial intelligence program, which tries enormous

numbers of "reasonable" possibilities in an automated fashion. The techniques for describing the "intelligent" part of this operation are so diverse and their number so large, that it is impossible to make any generalization about how this is implemented in a generic program.

3. How to do a computational research project (lab)

When using computational chemistry to answer a chemical question, the obvious problem is that you need to know how to use the software. Then, you need to have some information and/or intuition, concerning the quality of the answer, and you have to be able to make rational decisions about the possibility to sacrifice accuracy for efficiency. Here is a check list to follow.

- **What do you want to know? How accurately? Why?**

If you can't answer these questions, then you don't even have a research project yet.

- **How accurate do you predict the answer will be?**

In analytical chemistry, you do a number of identical measurements then work out the error from a standard deviation. With computational experiments, doing the same thing should always give exactly the same result. The way that you estimate your error is to compare a number of similar computations to the experimental answers. There are articles and compilations of these studies. If none exist, you will have to guess which method should be reasonable, based on its assumptions then do a study yourself, before you can apply it to your unknown and have any idea how good the calculation is.

- **How long do you expect it to take?**

Often computational chemistry calculations (especially ab-initio ones) are so time consuming that it would take a decade to do a single calculation, even if you had a very power machine with enough memory and disk space. However, a number of

methods exist because each is best for different situations. The trick is to determine which one is best for your project. Again, concerning the time consumption, the answer is to look into the literature and see how long each calculation takes, when particular soft- and hardware is used. If the only thing you know is how a calculation scales, do the simplest possible calculation then use the scaling equation to estimate how long it will take to do the sort of calculation that you have predicted will give the desired accuracy.

- **What approximations are being made? Which are significant?**

This is how you avoid “successfully” performing a calculation that is complete garbage. An example would be trying to find out about vibrational motions that are very anharmonic, when the calculation uses a harmonic oscillator approximation.

Once you have finally answered all of these questions, you are ready to actually do a calculation. Now you must determine what software suitable for your hardware is available, what it costs and how to use it. Note that two programs of the same type (i.e. ab-initio) may calculate different properties, so you have to make sure the program does exactly what you want.

When you are learning how to use a program, you may try to do dozens of calculations that will fail because you constructed the input incorrectly. Do not use your project molecule to do this. Make all your mistakes with something really easy, like a water molecule. That way you don't waste enormous amounts of time.

4. TAU computational chemistry laboratory course

The aim of the laboratory-course that is being given at TAU starting in the 2nd semester 2004 is to give the 3rd and 4th year students fundamental knowledge and basic skills in solving computational chemistry problems using commercial codes on windows

and unix workstations. This lab is not aimed for experts in theoretical chemistry. On the contrary, it is aimed at experimental chemists in all disciplines who are interested in applying computational chemistry methods in their research and in their future professional careers. Therefore no pre-requisites are imposed beyond first year chemistry and second year physical chemistry (including introduction to quantum chemistry).

In 2008 the laboratory will be given still on an experimental basis. From the point of views of the students this will imply that details of the projects required may be adjusted during the semester in order to adjust to students suggestions and needs. Furthermore, instructors will be asked to provide close guidance of the required computational experiments and to respond to students questions and needs throughout the week. No hard core theoretical effort will be required, but the students will be asked to gain a qualitative knowledge of the methods used and to understand the way in which input and output are read in and analyzed. Upon successful completion of the lab requirements a student should be able to apply the codes studied to routine organic compounds and processes.

- **Hardware and software for calculations and visualization available in TAU Computational Chemistry Laboratory (TAUCCL)**

1) Some common computer **software** used in TAUCCL for computational chemistry practicum includes:

- ***Gaussian W03*** (ab-initio, semiempirical, molecular mechanics calculations)
- ***Hyperchem 7.5*** (ab-initio, semiempirical, molecular mechanics & dynamics calculations)

Data visualization is the process of displaying information in any sort of pictorial or graphical representation. A number of computer programs are now available at TAUCCL

to apply a colorization scheme to data or work with three dimensional representations (*Gaussview, MolDen, gOpenMol*).

More detail information about possibilities, limitations and correct exploitation of particular computational chemistry program you can find in the appropriate “Beginner Guide” or in the corresponding Program Manual.

2) **Hardware** in TAUCCL includes 12 PC stations with dual boot Windows XP and LINUX OS.

5. Summary

1) To summarize, computational chemistry is:

- a branch of chemistry that generates data which complements experimental data on the structures, properties and reactions of substances. The calculations are based on quantum and classical mechanics, molecular dynamics, statistical theory and thermodynamics, semiempirical structure-properties relationships, theories of symbolic calculations and artificial intelligence, and include:
 1. Global potential energy surfaces calculations (including equilibrium states, transition structures and Van-der-Waals complexes)
 2. Calculation of wave function and electronic charge distribution and many other non energetic properties (like multiple moments, NMR parameters, etc)
 3. Molecular geometry in ground and excited states
 4. Vibrational and rotational constants
 5. Reaction paths and rate constants for chemical reactions
 6. Details of the dynamics of molecular collisions
 7. thermodynamical properties
- Particularly useful for:
 1. Determination of properties that are inaccessible experimentally
 2. Interpretation of experimental data

2) Before starting a computational chemistry project (lab) one has to make a scientifically grounded choice of the appropriate computational method and the corresponding software which suits the available hardware limits (memory, disc space and CPU-time). Then one has to learn how to make input file, how to run the program and how to interpret output on some simple examples. After all that steps were done the actual calculation could be started.

6. Further information and references

For an introductory level overview of computational chemistry see the main reference and source for this manual :

David Young "Computational Chemistry: A Practical Guide for Applying Techniques to Real World Problems", John Wiley & Sons , 2001

A more detailed description of common computational chemistry techniques is contained in

A. R. Leach "Molecular Modelling Principles and Applications" Addison Wesley Longman (1996)

G. H. Grant, W. G. Richards "Computational Chemistry" Oxford (1995)

F. Jensen "Introduction to Computational Chemistry" John Wiley & Sons (1999)

There are many books on the principles of quantum mechanics and every physical chemistry text has an introductory treatment. The book below is excellent for both intermediate and advanced users.

C. Cohen-Tannoudji, B. Diu, F. Laloe "Quantum Mechanics Volumes I & II" Wiley-Interscience (1977)

For an introduction to quantum chemistry see

D. A. McQuarrie "Quantum Chemistry" University Science Books (1983)

A graduate level text on quantum chemistry is

I. N. Levine "Quantum Chemistry" Prentice Hall (1991)

An advanced undergraduate or graduate text on quantum chemistry is

P. W. Atkins, R. S. Friedman "Molecular Quantum Mechanics" Oxford (1997)

For quantum Monte Carlo methods see

B. L. Hammond, W. A. Lester, Jr., P. J. Reynolds "Monte Carlo Methods in Ab-initio Quantum Chemistry" World Scientific (1994)

A good review article on density functional theory is

T. Ziegler Chem. Rev. 91, 651-667 (1991)

For density functional theory see

R. G. Parr, W. Yang "Density-Functional Theory of Atoms and Molecules" Oxford (1989)

For a graduate level description of statistical mechanics see

D. A. McQuarrie "Statistical Mechanics" Harper Collins (1976)

For thermodynamics study we recommend

I. N. Levine "Physical Chemistry" McGraw Hill (1995)

Another nice introduction to computational chemistry is

S. Profeta, Jr. "Kirk-Othmer Encyclopedia of Chemical Technology Supplement" 315, John Wiley & Sons (1998).

There is a comprehensive listing of all available molecular modeling software and structural databanks, free or not, in appendix 2 of

"Reviews in Computational Chemistry Volume 6" Ed. K. B. Lipkowitz and D. B. Boyd, VCH (1995)

There is a write up on computer aided drug design at

gopher://ccl.osc.edu/00/documents/drug.design.guide

Mathematical challenges from theoretical/computational chemistry

<http://www.nap.edu/readingroom/books/mctcc/index.html>

An online text on molecular modeling using molecular mechanics

<http://www.netsci.org/Science/Compchem/feature01.html>

A Computational Chemistry Primer

<http://www.sdsc.edu/GatherScatter/GSwinter96/taylor1.html>

An online text on computational chemistry

http://www.cryst.bbk.ac.uk/~ubcg8ab/course/os_molf.html