

# Quantum Chemistry with Jaguar for Beginners

Created November 2015 with Release 15-4

---

In this guide:

1. *What do you want to do?*
  2. *What is realistically possible?*
  3. *Understanding the basics of the process*
  4. *Launching and monitoring Jaguar calculations*
  5. *Picking the appropriate functional and basis set*
  6. *Viewing and analysing the results*
  7. *Typical Jaguar workflows*
  8. *Troubleshooting and what you need to be aware of*
- 

## 1. What do you want to do?

This guide is written for researchers who have minimal or no quantum chemistry background but are wondering if they can use quantum chemistry calculations in their research. The text below attempts to avoid technical and mathematical details and concentrates on the practically important questions such as what kind of research is possible with quantum chemistry, how much time it might take to do a project, and what quality of results you should expect to obtain. In case you need more details on any of the terms or workflows mentioned in the text you can always consult the manuals available in Maestro via Help->Documentation Index or access context-sensitive help by clicking the "?" button at the bottom of any Maestro panel.

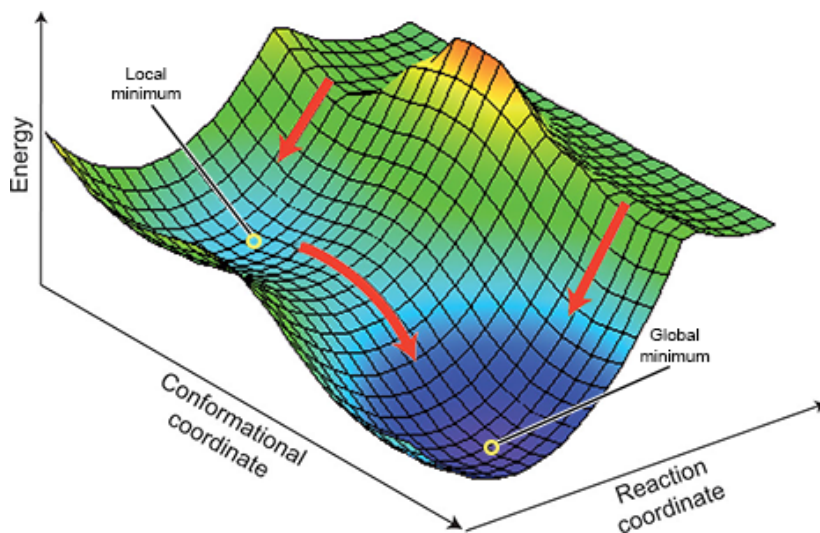
If you are reading this guide you probably already have a scientific problem in mind that you think quantum chemistry, and in particular Jaguar, can help you solve. If your problem is about predicting shapes (the position of atoms in three spatial dimensions) or other *physical* properties of molecules it should be straightforward to address. If your problem deals with *chemical* processes or properties (chemical reactions or mechanisms, binding of small molecules to proteins, reactivity with respect to a certain nucleophilic agent etc.) it necessarily involves multiple molecular entities and may take more work.

Let us talk about predicting shapes of molecules first, since it is the most basic concept and problem. When we say "shape" we mean the shape of the molecule in its relaxed, energetically most favorable configuration and when no extraneous forces (such as the electric field or the presence of a neighboring molecule) are influencing the positions of its atoms. This is the so-called *optimal* shape, or "geometry". Quantum chemists often say "geometry" when they mean "shape", so that it's common to hear phrases like "What is the geometry of this molecule?". It is important to know the optimal geometry when studying molecules because this is the geometry in which the molecule is likely to spend most time and which has the greatest impact on its physical and chemical properties. A non-optimal geometry may have little relation to the experimentally observed geometry and may have properties that are of no interest to the practical scientist.

You can search for the optimal geometry in a simulated gas or solution phase. Which phase to use will be dictated by the needs of the physical experiment you are trying to model. Regardless, in order to arrive at the optimal geometry you need to have a good initial guess, that is, a geometry that is intended to be quite similar to the optimal geometry. This may seem like a strange requirement: how are you supposed to know what is

quite similar to the unknown that you are trying to find? Here, we must use our chemical intuition and guess what the molecule should look like in its optimal geometry, and then if our guess is good enough, Jaguar will find the optimal geometry in the vicinity of the guess structure. If we don't give the program a good initial guess it might arrive at a conformation or even isomer that we are not interested in. It might not arrive anywhere at all! In practice, coming up with the initial guess might be trivial or hard, depending on the circumstances. If we want to find the optimal geometry of benzene it's obvious to every chemist that the guess structure should be flat, with equal C-C and C-H bond distances that have reasonable lengths (that is, approximately 1.4 and 1.1 angstrom, respectively). Any chemical software can build a benzene molecule that will serve as a great initial guess for a quantum chemical geometry optimization. Not so for a large, flexible molecule, such as a typical modern candidate for a drug, or an organometallic complex that has never been seen in an experiment. In the former, while a molecular builder can do a reasonable job at assigning bond lengths, angles, and torsions for the structure, we may not know which *conformation* of the structure is close to the optimal minimum. In the latter, the bond types between the metal and other elements may be (yet) unknown, and therefore we may not know which bond lengths or angles are close enough to those in the optimal structure. Various approaches exist to deal with such complicated cases, and we will talk about them later.

It is important to realize that when we draw a molecule in Maestro (or some other molecule builder) we simply create an initial guess. At this point no quantum mechanical calculations have been conducted yet. The structure needs to be passed to Jaguar for subsequent quantum chemical geometry optimization, a process that is going to produce a much more accurate geometry. Such a geometry can be used for studying the structure's physical and chemical properties.



**Figure 1.** A potential energy surface with local and global minima.

Assuming you have found the optimal geometry for your molecule (corresponding to the global minimum in **Figure 1**), it is now appropriate to try to predict its physical properties. When you are doing a calculation at a single geometry without modifying it in the process you are doing a *single point* calculation. This is in contrast to calculations involving a changing geometry (a geometry optimization) which would be a multiple point calculation. In this terminology a "point" obviously refers to a single geometry. There are numerous properties you can predict in single point calculations - the dipole moment, the infra-red (IR) spectrum, the electrostatic potential, etc. Assuming the properties you are predicting are directly observable in the experiment or *correlated* with experimental data, your task, as far as Jaguar is concerned, may be done.

Recall that so far we have been talking about the physical properties. When it comes to the chemical properties, things can become more complicated because more calculations are involved. Now you are dealing with multiple molecules which are either reacting with each other or regrouping their atoms. Therefore, you might need to run multiple Jaguar calculations on the participating molecules before you can produce computational results relevant to the reaction. Such multiple individual calculations that are performed with a single goal in view are called a *workflow*. For example, computing the enthalpy of a reaction can be a workflow. Predicting the pKa of a certain molecule deals with a protonation (or deprotonation) reaction and is a workflow that involves several single point and geometry optimization calculations. Of course, there can be workflows for predicting both physical and chemical properties. Workflows can be supplied by Schrödinger and accessible from Maestro, or you can design and implement your own workflows through Python scripts or KNIME.

To summarize, there are essentially three general types of Jaguar calculations:

- (i) Geometry optimizations
- (ii) Property calculations at a fixed (often preliminarily optimized) geometry
- (iii) Workflows involving numerous chemical structures and combinations of steps (i) and (ii) above

## 2. What is realistically possible?

If we ignore relativistic effects, which might be important for heavy elements, then the Schrödinger equation should define *all* physical or chemical properties of molecules. Thus, we can view the Schrödinger equation as governing a large part of physics and the whole of chemistry. It is a deceptively simple equation, but its accurate solution for large many-atom systems such as proteins and crystals will likely remain impossible in the foreseeable future. However, there are highly efficient approximations to the Schrödinger equation that permit calculations to a useful accuracy on realistic molecules and solid state clusters involving up to several hundred atoms. This is a good news as the vast majority of drugs, catalysts, polymer and crystal units, species encountered in organic synthesis, active sites of enzymes, and other medium-sized entities from the realm of chemical research fall into this size category, so the quantum chemical methods available today can be very much applicable. This chapter discusses how accurately and how fast one can expect to compute properties of these manageable systems with Jaguar and other quantum chemistry programs.

The time that it may take to finish a quantum chemistry calculation depends dramatically on the size of the molecule that the calculation is dealing with. The time scales non-linearly. This means that a calculation involving a 100 atom molecule will take more than twice as long as that involving a 50 atom molecule in an equivalent calculation. Let us therefore assume for concreteness that our molecule contains about 50 atoms and has a molecular weight of about 400 a.u. Our molecule may be a typical drug candidate or a target in organic synthesis. After we understand the possibilities and limitations associated with molecules of this size we will discuss larger molecules.

A typical good quality geometry optimization of the 50-atom molecule with atomic weight around 400 a.u. might take anywhere from ten minutes to two hours. Next, to compute the chemical properties of this molecule at the optimal geometry might take anywhere from a few minutes to a couple of hours, depending on which property you are trying to calculate. So the computational expense for a geometry optimization roughly equals that of a single point calculation. This might appear strange because a geometry optimization involves multiple single point steps (as we go from geometry to geometry in the process), and should therefore be many times more expensive than a single point calculation. This is correct reasoning, but the geometry optimization is usually performed at a lower accuracy than the final single point calculation, and not all the computationally expensive properties are generated at each successive geometry. Therefore, we have a significant saving of time per geometry in comparison with the final single point calculation. So, if we start from a new, unoptimized 50-atom

molecule it will normally take anywhere from half an hour to 4 hours to predict its optimal geometry and physical properties.

Earlier we said that the time required to conduct a calculation depends on the size of the molecule. But even when dealing with the same molecule, it is possible to select different settings and parameters which will greatly affect the speed of the calculation. While a basic quality, approximate calculation might take minutes a much more accurate calculation utilizing expensive methods might take hours or days. Therefore, when estimating the time required for the calculation it is important to take into account not only the size of the molecule but also the methods and other parameters used. Another factor that affects the speed of calculations is parallelization and the amount of available computational resources. If you have multiple processors on your desktop computer or computer cluster you might be conducting multiple calculations on different molecules simultaneously. Thus, you might screen dozens and even hundreds of molecules for a desired property in a matter of hours, if you have access to a modern computer cluster. You could also speed up one calculation dramatically if you use multiple processors, all working on the same single task.

In any case, a couple of hours of computational time per molecule per processor or two sounds reasonable. After all, research projects take weeks and months, and conducting a few or even a great number of quantum chemical calculations for one project is not a problem. A modern computational chemistry paper can be based on hundreds and thousands of individual quantum chemical calculations. If we have just a few molecules to act on we can use multiple processors per molecule and that will speed up calculations even further. It is important to stress though that using multiple-processor calculation per molecule is efficient only when we have more processors than molecules. When we are screening thousands of molecules it's more efficient to use serial (single-processor) calculations.

How about larger molecules which contain 100-300 atoms? Such molecules are quite common in chemical research. Just think of a short nanotube or a protein active site with one or two adjacent shells of amino-acids included in the molecular model. By a rule of thumb, every doubling in molecular system size leads to an eight-fold (cubic) increase in the computational time spent for a typical task. By this logic, we estimate that a 100-atom molecule will require 4 to 32 hours, and, correspondingly, a 300-atom molecule will require 4.5 days to a little more than a month. This is still doable in a project, but your freedom to compute multiple structures or adjust parameters many times will be restricted.

Often chemists work with even larger structures, such as proteins and solid state systems. Applying Jaguar to a system of over 300 atoms is not advisable (although Jaguar will still likely work, having a limit of 1000 atoms), mostly because the calculations will become prohibitively slow. What should we do then? One can apply periodic boundary conditions, which is especially applicable for solids. At present, Jaguar does not support periodic boundary conditions. However, the simplest and often the best approach is to isolate the part of your system in which the chemical process of interest is happening, and truncate the rest, methyl- or hydrogen-capping the "dangling" bonds. In this approach we make an assumption that the distant atoms in the system (those we truncated) affect the modeled process only marginally and can be neglected.

Calculations in the solution phase or very accurate gas phase calculations will require more time. Allowing a factor of 2-4 when planning for the associated computational expense is necessary. Thus, it's possible to consume 1 to 16 hours for optimizing the geometry and computing properties of a 50-atom molecule in solution, or, alternatively, conducting the calculations at a higher than normal accuracy. Large, flexible molecules containing numerous rotatable bonds are likely to possess a very complicated potential energy surface, having multiple local minima. Optimizing geometries of such molecules might require an additional computational expense and success of locating the absolute minimum might depend critically on selecting a good initial conformation. **Figures 2a-c** show some timings for various job tasks at some common levels of theory.

Atoms	Basis size	Energy	Geopt per step	NMR	Vibrational freqs	TDDFT per root	ESP chg + dipole
15	145	<1 m	<1 m	<1 m	16 m	<1 m	<1 m
49	485	4 m	2 m	4 m	2 h	2 m	<1 m
77	775	21 m	5 m	14 m	9 h	9 m	<1 m
104	1060	26 m	8 m	29 m	25 h	7 m	<1 m

**Figure 2a.** Timings for various tasks run with B3LYP/6-31G\*\*.

Atoms	Basis size	Energy	Geopt per step	NMR	Vibrational freqs	TDDFT per root	ESP chg + dipole
15	233	1 m	<1 m	1 m	2 h	<1 m	<1 m
49	777	14 m	4 m	13 m	12 h	6 m	<1 m
77	1239	1 h	11 m	43 m	52 h	50 m	1 m
104	1692	2 h	20 m	2 h	144 h	1 h	1 m

**Figure 2b.** Timings for various tasks run with B3LYP/cc-pVTZ(-f).

Atoms	Basis size	Energy	Geopt per step	ESP chg + dipole
15	233	1 m	2 m	1 m
49	777	1 h	3 h	2 h
77	1239	5 h	days	20 h

**Figure 2c.** Timings for various tasks run with LMP2/cc-pVTZ(-f).

### 3. Understanding the basics of the process

Now that we know what we want to do and what we *can* do, it might be tempting to go ahead and learn *how* we can do it in Jaguar. For *how* we actually reserve the next chapter but in this chapter we are going to discuss some of the physical and computational processes that run behind the scenes. Modern quantum chemistry programs and their graphical interfaces make it seem as if the calculations are easy to set up and conduct. However, there are numerous complicated technologies and algorithms involved in the calculation, and in some difficult scenarios they may not work smoothly. In any case, it's always worth knowing the basics of what's involved before launching a calculation or analyzing its outcome, especially if the result is unexpected or an outright failure.

Although this chapter goes over the physical and technical aspects it actually skips most of the details, terminology and complicated abbreviations which are often present in the quantum chemical literature. In what follows we try to give a higher-level picture which should be sufficient for becoming comfortable with Jaguar calculations. The user interested in the details of the methods and technologies used should be able to easily find them in countless other sources.

As we mentioned above, quantum chemistry seeks approximate solutions to the Schrödinger equation. Actually, the exact solutions are unreachable for any system containing more than one electron. There is a very large number of methods that will yield an approximate solution. These methods differ by their computational expense, accuracy they provide, properties they can compute or phenomena they can model. Some quantum chemical programs make available to the user dozens of method families and hundreds of individual methods. It is not surprising that navigating such a zoo of methods can be a major hurdle for someone who is not a quantum chemistry expert.

Jaguar keeps the number of method families to the minimum, and the only such family that we will discuss and that a beginner should use is density functional theory (DFT). This class of methods can address the vast majority of problems encountered in practice and is easy to understand and use.

Let us start discussing a typical single point DFT calculation of a single molecule in the gas phase.

First, the atomic coordinates of your molecular system are read, and the settings (such as the desired flavor of DFT, the accuracy level etc) are applied. Importantly, quantum chemical calculations should not depend on the Lewis bond structure (connectivities between atoms). Bonds that we draw between atoms (such as single, double, and triple bonds) are not physical observables and only serve as a visual guide for the human who is trying to make sense of the atom arrangement. Most Jaguar calculations accept just the xyz atomic coordinates, without any information about bonds between these atoms. So it wouldn't matter for the final result of the calculation, for example, which resonance structure you passed to Jaguar, as long as the relative physical locations of the atoms remain the same. There are a small number of calculation types though that require connectivities between atoms before the calculation can be launched. Such are solvation calculations, pKa predictions, and all calculations invoking a few special DFT methods (B3LYP-MM and B3LYP-LOC). This is because these methods use not only quantum-chemical data to formulate their final result but also empirical parameters based on connectivities, and are thus reminiscent of molecular mechanics (MM) atom typing methods. To avoid surprises or the need to learn which Jaguar methods require connectivity between atoms and which not, it's safer to always draw molecules in Maestro and launch calculations from Maestro, as opposed to specifying the molecule in Jaguar input files "manually", using xyz coordinates only.

The following step is a solution of the so-called self-consistent field (SCF) equations which are part of any DFT calculation, and are necessary to find an approximate solution to the Schrödinger equation. This is such a fundamental process to all quantum chemistry and in particular to Jaguar that it's important to dwell on these equations for a few sentences. The SCF equations essentially try to "equilibrate" the nuclei and the electrons in the molecule using an iterative process. In the course of solving the SCF equations the nuclei are kept fixed but the electron density attempts to find the best distribution in the electrostatic field created by the nuclei. The SCF process converges when the equilibrium is achieved. The details of the SCF equations convergence are printed in the Jaguar output file (with the extension \*.out), as in the example below where the equations converged successfully in 6 iterations:

	i u d i g				
	t p i c r				
	e d i u i			RMS	maximum
	r t s t d		energy	density	DIIS
		total energy	change	change	error
etot	1 N N 5 M	-76.16533343457		8.8E-03	2.5E-01
etot	2 Y Y 6 M	-76.37046259127	2.1E-01	9.8E-03	1.0E-01
etot	3 N Y 2 U	-76.41194161727	4.1E-02	3.2E-03	3.8E-02
etot	4 Y Y 6 M	-76.41873322496	6.8E-03	8.6E-04	1.2E-02
etot	5 N Y 2 U	-76.41947763376	7.4E-04	4.6E-05	7.3E-04
etot	6 N N 2 U	-76.41947963211	2.0E-06	0.0E+00	0.0E+00

It is important to realize that the SCF equations are complicated nonlinear equations that are not guaranteed to have a solution. They are solved iteratively, starting from an automatic initial guess and improving on the quality of the solution with each iteration. The initial guess is essentially a density of inferior quality, and it can be constructed automatically. The accuracy of this initial density is too low for obtaining practically useful energies and properties from it but it's good enough for launching the process of solving the SCF equations. In the vast majority of calculations a physical solution to SCF equations will be found, which is a prerequisite for obtaining a physically reasonable result from the Jaguar calculation. If for some reason the SCF equations can't be solved (the common description of this situation is "they do not converge") no results such as geometries, energies, atomic charges,

or IR spectra can be produced. In the following chapters we will discuss what to do if the SCF equations don't converge.

Next, properties such as energies, dipole moments, atomic charges, etc., are computed, using the data provided by a solution of the SCF equations. Some properties (such as polarizabilities) require solving complicated nonlinear equations in their turn, but usually this is not a problem provided the preceding SCF equations were solved successfully.

To summarize, a single point Jaguar calculation in the gas phase goes through the following main steps:

- (i) Read and process the input
- (ii) Solve SCF equations
- (iii) Compute the energy and other requested properties

Solvation phase single point calculations and geometry optimization calculations are more complicated, because they repeat these three single point steps multiple times. For example, geometry optimization calculations adjust the atomic coordinates between the individual single point calculations. Should any individual single point calculation fail the whole geometry optimization or solvation calculation may fail as a result.

Finally, geometry optimizations in solution are some of the most complicated quantum chemical calculations one can imagine. Not only must the SCF equations be solved several times as part of a solution to the Poisson-Boltzmann equation (which is the equation that describes the distribution of the electronic density of the molecule in the environment of the solvent). The Poisson-Boltzmann equation must itself be solved for every new adjusted geometry, until the optimal geometry is located in the process.

We are mentioning these details partly to show how complicated a quantum chemistry calculation might be behind the scenes, and partly to indicate the potential points of failure, which is good to know for successful troubleshooting if anything in the process goes awry.

## 4. Launching and monitoring Jaguar calculations

In order to launch a Jaguar calculation we need to specify *what* we are going to compute and *how* we are going to compute it. The *what* part of the specification consists of the molecule itself (with the associated total molecular charge and spin), the type of the calculation (is it a single point or a geometry optimization? is the simulation to be conducted in the gas phase or in water?) and the physical or chemical properties that we need. The *how* is about choosing the appropriate DFT method and possibly the accuracy level and the initial (guess) conditions.

The molecular system can be drawn in Maestro or imported from a file in a variety of file formats. The properties to be computed are specified in the Properties tab in the Jaguar panels (such as the Single Point Energy panel). This is usually simple and straightforward. The more difficult part of the preparation to launch a Jaguar calculation is in the *how* question.

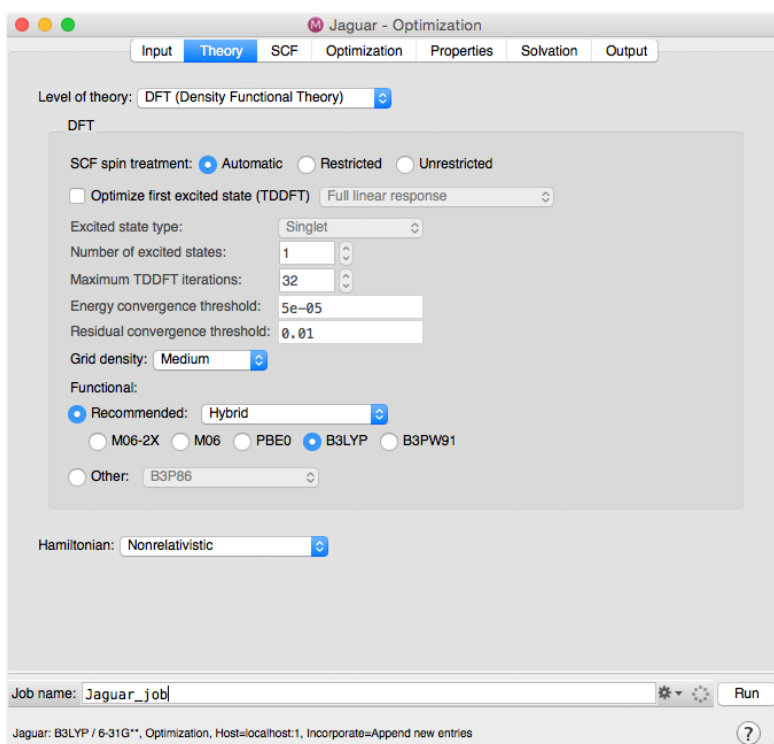
There are two fundamental settings that are necessary to specify for each Jaguar DFT calculation. They are called the functional and the basis set. Several dozen functionals and several dozen basis sets are available in Jaguar. The number of possible combinations of these two fundamental settings may be staggering, especially in view of not knowing which of them might produce the most accurate result for your particular problem. An accurate functional/basis set combination for a given molecular system or a physical property might be a poor choice for a different system or a property. Unfortunately, the quantum chemical community has not developed a reliable algorithm or rules for selecting the best such combination. Experts usually follow the guidelines they themselves developed after years of experience, but beginners feel the frustration acutely. I am not a big wine drinker, for



example, and I experience a similar frustration when I have to pick a wine from a very long wine list at a refined French restaurant. Usually I pick a random wine and I don't like the result. Maybe that's why I don't drink wine in general. And maybe that's why many beginners avoid doing DFT calculations after a few not very successful attempts.

Of course, Jaguar has default choices for a DFT functional and a basis set. You needn't select anything from the list, and you can simply trust the Jaguar defaults from the start [Figure 3]. The Jaguar choice will produce good results in many situations but might produce less optimal or even bad results in other situations. It's worth learning a few general rules about picking the functional and the basis set, and the whole next chapter is dedicated to this subject. While there are no perfect or complete rules the advice we give in the next chapter will serve as reliable guidelines for any but the most sophisticated calculations.

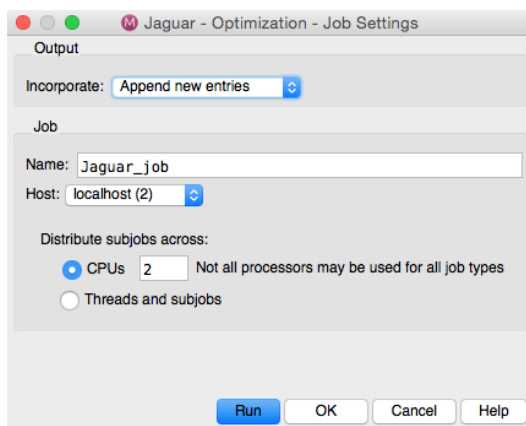
Regarding the accuracy and initial guess settings, those are usually much simpler to handle. In the first approximation the default accuracy and initial guess settings are good and will work well in the vast majority of cases. They can be adjusted if need arises, and when and how to do so is discussed in the Troubleshooting section.



**Figure 3.** Default settings for the Jaguar Optimization Theory panel.

Once the settings are chosen (the molecule, its charge and spin, the calculation type, the properties to be computed, the functional and the basis set, and the desired accuracy and the initial guess) the Jaguar calculation can be launched. You can launch multiple Jaguar calculations (for example, on different molecules) at a time. You also need to indicate how many processors you want to use for your calculation. In Jaguar, you specify the total number of processors for the whole task you are launching [Figure 4]. If your task consists of performing geometry optimizations of, say, 20 different molecules, Jaguar will automatically distribute the number of processors you specified among the 20 individual geometry optimizations.





**Figure 4.** The Jaguar Optimization Job Settings panel.

As we discussed in the previous chapter, some quantum chemical calculations might take hours to complete. We recommend that you begin getting used to the process of launching, monitoring, and analyzing Jaguar results by applying it to small molecules, possibly those with 10-15 atoms, just to make the calculations fast.

While the submitted job is running you can monitor its progress in Maestro (including live results printed in \*.blog, \*.log, or \*.out files, depending on the type of the calculation) from the Monitor panel [Figure 6]. When the job is finished its results are incorporated into the Project Table [Figure 5]. For more information on viewing and analyzing results produced by Jaguar see the chapter “Viewing and analysing the results”.

Row	In	Title	Molecular charge	Spin multiplicity
1	<input checked="" type="checkbox"/>	1 - Jaguar_job	[9]	
2	<input type="checkbox"/>	16088	1	1
3	<input type="checkbox"/>	1167	1	1
4	<input type="checkbox"/>	15722	2	1
5	<input type="checkbox"/>	151943	0	1
6	<input type="checkbox"/>	165006	0	1
7	<input type="checkbox"/>	195476	0	1
8	<input type="checkbox"/>	281488	0	1
9	<input type="checkbox"/>	334669	0	1
10	<input type="checkbox"/>	15650	1	1

Entries: 9 total, 9 selected, 1 included Groups: 1 total, 1 selected Properties: 27 total, 7 shown

**Figure 5.** Jaguar job results incorporated into the Maestro Project Table.

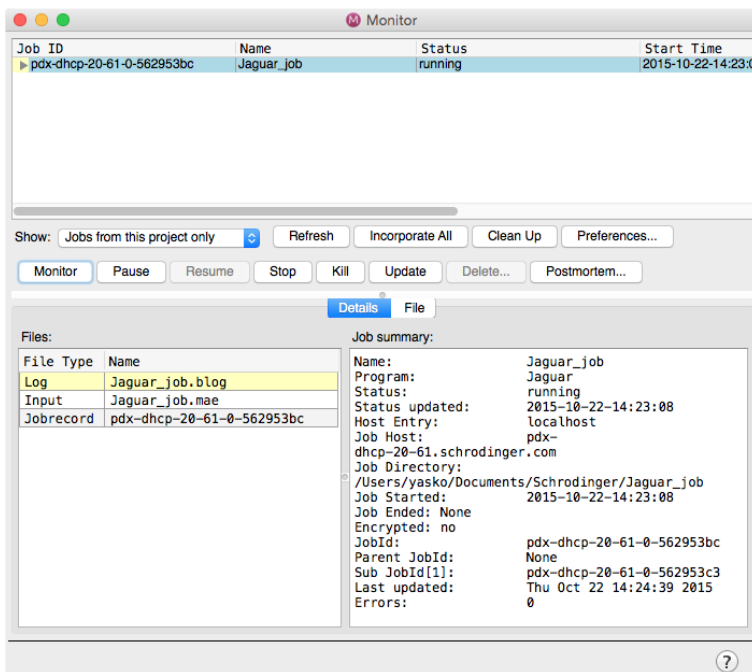


Figure 6. The Maestro Job Monitor panel.

## 5. Picking the appropriate functional and basis set

As we explained in the previous chapter, choosing the functional and the basis set is an unavoidable task that you must perform at the beginning of each Jaguar DFT calculation. To be sure, you needn't physically click on the functional and basis set setting every time you launch a new calculation because the Jaguar panel either remembers your previous choice or offers reasonable default settings. However, you need to be aware that a setting is made either by you or by Jaguar, and that it will affect the quality of your final result.

In the previous chapter we also mentioned that there are no reliable rules for making the best choice of the functional and basis set. But the guidelines given below will help you avoid making definitely bad choices and might steer you in the direction of securing good, if not the best picks.

There are two general approaches to making the functional/basis set selection. The first one is to find a good combination for your problem (the combination that produces good or reasonable results) and apply it to the rest of the structures and calculations. There might be researchers who use the same combination for years, irrespective of the type of the molecule, the computational task, and the property they are computing. This approach has the advantage of simplicity. One can essentially forget about the need to make the choice. Another advantage is consistency. Whatever the quality of your results is it's at least likely to be uniform. If you are interested in trends, as opposed to absolute numbers, the same functional and basis set, whatever it is, is likely to represent the trend and lead to the same conclusion. This approach is good if you are using Jaguar to do essentially the same task (geometry optimizations, but no NMR spectra prediction) for essentially the same kinds of systems (substituents can vary but you don't substitute a carbon atom by, say, cobalt). The downside of this approach is of course that your results might be sub-optimal or even bad if the particular combination you are using is not quite applicable to the chemistry or physics you are trying to model. This might happen when you switched to a new type of system or property. If the functional or the basis set cannot possibly describe the physical effect that should play an important role in your trend then the predicted trend (not only the individual numbers) might

simply be wrong. For example, if you were successfully optimizing geometries of isolated molecules with the B3LYP functional but then switched to optimizing geometries of molecular complexes stabilized by dispersion interactions then you will discover that B3LYP makes these complexes fall apart. This is because B3LYP is not suited for describing dispersion interactions and therefore you need to pick a different functional to treat such systems.

The second approach is that you learn about the basics of the functional and basis set composition and performance and stay reasonably updated about their development (because new functionals and basis sets appear every year). You use a handful of guidelines to decide which functional and basis set might be particularly suitable for your type of system or property, and apply these guidelines to choose an appropriate combination. The downside here is that the selection process is more complicated and you might accidentally arrive at a bad basis set/functional combination if you are not careful. If you work on a large number of different types of molecules, if you apply different functionals and basis sets to them, on a system-to-system basis, then your results (or errors thereof) might be non-uniform. You also must be careful not to use different combinations of functional and basis sets for absolute energy calculations, if those energies are meant to be compared. Only those properties that are in principle observable (relative energies, polarizabilities, IR spectra) can be compared if they are computed with different functionals and basis sets. The advantages are that provided you have learned how to pick good functional/basis set combination and avoid common mistakes your results are likely to be more accurate. If you have an arsenal of several functional/basis set combinations that you apply to your problems you will be more flexible in your research if some combinations turn out to be inapplicable for some reason.

The first approach is more passive: you trust the Jaguar default settings or the choice of your colleagues or other researchers. The second approach is more active. It assumes that you are willing to learn about the strengths and weaknesses of different methods and settings to gain advantage for obtaining the best quality of results. I personally use the second approach and will show here that it just takes a little time to acquire the basics to make an intelligent choice. The choice of the functional is not completely independent of the basis set but in the first approximation it is. As the basis set selection is simpler and more predictable, let's start with it.

Importantly, absolute energies that are compared to decide which energy is lower or higher must be computed with the same functional/basis set combination. If the basis set and functional are not consistent then the comparison (subtraction) of energies will produce meaningless results. On the other hand, geometries and other properties computed with inconsistent combinations *can* be compared.

In DFT, the basis set defines the flexibility of the electron density which is sought for when solving the SCF equations. As we don't know what the final electron density will or should be, the more flexibility our *trial* or intermediate density has, the better the chances of finding a good final electron density. In principle, the basis set should be so flexible as to give an almost infinite flexibility to the electron density. In practice, the larger the basis set and the higher the flexibility, the higher the computational cost of the calculation. Earlier we said that the computational cost depends on the number of atoms in the molecule. The cost is actually more closely correlated with the number of electrons. And the number of electrons is roughly proportional to the number of the basis functions (or the size of the basis set) used in your computation. Because we want to avoid a steep increase in the computational cost when taking larger and larger basis sets we need to reach a compromise solution: pick a basis set that is not too small (so that it gives enough flexibility to the density) and not too large (so that it makes the calculation feasible). Larger and more diverse (in terms of composition) basis sets typically perform best.

Note that different basis sets can be assigned to different elements and even different atoms of the same element in the molecule. Thus, basis set is an atomic setting. If you are using one basis set for your molecule this means that the same basis set is placed on all the atoms in the molecule. Jaguar allows you to put different basis sets on

different atoms. These “mixed” basis sets can be handy to save computational cost, but they require more work for specification.

The number of basis set types available in the literature, and even in Jaguar, is large, possibly a dozen families. Only two such families are used regularly by the computational chemistry community: the 6-31G family (the so-called Pople basis sets) and the cc-pVNZ family (the so-called Dunning basis sets), and therefore these are the families we will discuss. The meaning of these awkward-looking family names (what is “6” and “31” and what is “cc” and “pV” etc) will not concern us here and is not a necessity to understand for an applications chemist.

There are two essential things anyone should know about basis set names - what the “stars” and “pluses” mean (the “\*” and “+” symbols, respectively) in a basis set name like 6-31+G\*\*. The stars and pluses highlight the composition of basis sets and indicate their applicability to certain kinds of calculations.

The stars indicate that the basis set contains the so-called polarization functions. Roughly speaking, these functions help the corresponding electron density deviate from spherical symmetry around each atom (making it polarizable). Using a basis set with polarization functions is important for describing phenomena such as polarization and polarizability but is also needed to correctly describe molecules containing heavier elements (starting from 3p-elements, such as phosphorus and sulfur). In the 6-31G family the presence of polarization functions is indicated explicitly by one or two stars (as in 6-31G\*). In the cc-pVNZ family polarization functions are already included, so you will never see a basis set name like cc-pVNZ\*. In general, we greatly recommend making sure that your basis set contains some polarization functions. If you are trying to save computational cost by avoiding polarization functions, make sure you use them at least on phosphorus and heavier atoms.

The pluses indicate the presence of the so-called diffuse functions. These functions, again roughly speaking, provide the electron density with additional volume, so that the electron density can be found farther from the atomic nucleus. As such, they help describe anionic species (that tend to have their electron density diffused, or spread out in space), or phenomena dependent on distant, non-covalent interactions (such as  $\pi$ - $\pi$  stacking). Note that the cc-pVNZ family does not include these diffuse functions by default (as in the case with the stars), and they can be added optionally. For example, you can have cc-pVDZ+ or cc-pVTZ++. Two pluses in the names of these basis sets are sometimes substituted by the prefix aug- (“augmented”), so that another name found in the literature for cc-pVTZ++ is aug-cc-pVTZ.

You might see one or two stars and one or two pluses in the basis set name, for example 6-31++G\*\*, or 6-31++G\*. One star or plus means that it is applicable to the non-hydrogen atoms in the molecule. Two stars or pluses mean that they are applied to all the atoms (hydrogens and non-hydrogens). Thus, 6-31++G\* means that the polarization functions are placed on non-hydrogens only, but the diffuse functions reside on all the atoms.

Now, after having reviewed the basics of the basis sets names and applicability, let us turn to the functionals. There appear to be more functional families than basis set families. Functionals are also not as easily categorizable in terms of applicability and accuracy. A comprehensive discussion of even a small group of functionals in terms of their ability to predict physical and chemical properties is far outside the scope of this guide. Therefore, we will just mention a few functionals we like and comment briefly on them. This small list of functionals and their short descriptions should provide you, the practical computational chemist, with enough information to choose from in the vast majority of situations.

B3LYP is the most popular functional of all time. It shows good performance for many molecules and properties. Its strong points are universality and predictability, as the world has accumulated thousands of publications in which it was used. That is why B3LYP is chosen as the default functional in Jaguar. Its weak points are non-covalent interactions, such as dispersion. For example, B3LYP does not bind two benzene molecules.

B3LYP-D3 is a version of B3LYP augmented with the D3 energy correction. Being a simple correction based on atomic coordinates, D3 has a negligible contribution to the total computational cost of the calculation. B3LYP-D3 retains all the positive qualities of B3LYP and corrects its inability to describe noncovalent interactions. B3LYP-D3 does bind two benzene molecules. It is important to note that the D3 correction does not change the electron density produced by the functional and affects only the energy (and the derivatives of the energy with respect to the atom coordinates). Even though B3LYP-D3 is not as well researched as B3LYP we recommend using it instead of B3LYP in all practical situations.

M06-2X is a functional which excels at energy-related predictions. For example, it is good for ranking conformers or tautomers in terms of their stability. M06-2X is about 40% more computationally expensive than B3LYP for computing energies, and might be even more expensive for geometry optimizations and vibrational frequency predictions. M06-2X is good at predicting a number of properties (such as describing noncovalent interactions and polarizability) but you should avoid using it for computing NMR spectra, for which it produces very large errors.

B3LYP-MM is B3LYP augmented with the MM energy correction. In some ways, the MM correction is similar to the D3 correction and helps describe noncovalent interactions. The MM correction has the advantage of containing special correction terms for hydrogen-bond and  $\pi$ -cation interactions. Its disadvantages are that it is not applicable to less common elements (for example, phosphorus and iodine) and that it is parameterized for only two basis sets: LACVP\* and cc-pVDZ++.

B3LYP-LOC is B3LYP augmented with the LOC energy correction. This correction is intended to describe covalent interactions better. It is applicable to several types of common elements and bond types only and helps correctly describe properties such as heat of formation. The LOC correction can be used only in conjunction with the following four basis sets: 6-31G\*, cc-pVDZ, cc-pVTZ++, and 6-311+G-3df-3pd.

## 6. Viewing and analysing the results

Provided you launch your Jaguar calculations from Maestro, once the calculation finishes the results can be viewed and analyzed right in Maestro, without the need to export them to other software. The finished calculations are automatically incorporated into the Project Table, which is a central resource for accessing scalar data computed by Jaguar (gas phase and solvation energies, dipole moment components, pKa's etc). The scalar data can be sorted, plotted, edited, etc. If you have requested a computation of 3D surface properties such as electrostatic potential energy or molecular orbital surfaces, the entry in the project Project Table will contain a link to a panel that manages 3D surfaces (look for the little blue "S" button in the Title column). If you have requested vibrational frequencies to be computed, those can be seen and animated through the similar "V" button in the Project Table. The spectral data computed by Jaguar can be visualized via Applications -> Jaguar -> Plot Spectra or alternatively via Tasks -> Quantum Mechanics -> Plot Spectra. And atomic properties such as Mulliken charges are best viewed through Workspace -> Atom Labels.

Sometimes you might want to process the results of your calculations from the command line. This can happen when you use Jaguar results in your custom scripts and workflows. You might parse Jaguar output files directly or use for that purpose Python modules developed by Schrödinger. One command-line resource that can come in handy is the "Jaguar results" script. The script has the ability to parse most types of the Jaguar output and compile the results into a neat plain-text table.

## 7. Typical Jaguar workflows

Jaguar calculations that combine multiple single point calculations or geometry optimizations, and possibly include results from other software packages, can be categorized as “workflows”. An example of a workflow is a pKa prediction which models a thermodynamic cycle that consists of multiple Jaguar calculations. The molecular system might need to be pre-processed by a separate molecular mechanics program, MacroModel, which can be accessed from Maestro (Applications -> MacroModel). MacroModel is used to sample the conformational space before the actual Jaguar calculations are launched. Below we will describe several typical Jaguar workflows and indicate in which research situations they can be useful.

*Ranking conformations by energies.* Often a large, flexible molecule exists as an ensemble of multiple conformers which might have slightly different physical and chemical properties. When predicting properties of such a molecule it is then important to act on the molecule’s most stable conformations and then “average” the properties of the individual conformations accordingly to get the final result. This workflow can start with generating the conformations with MacroModel [Figure 7]. Once the MacroModel calculation finishes the conformations become available in the Project Table. To continue with the Jaguar part of the workflow you need to select these conformers in the Project Table and open the Jaguar Geometry Optimization panel or Single Point Energy panel (either can be used, with a slightly different purpose)[Figure 8]. Then set the appropriate Jaguar options (such as the functional and the basis set) and launch the calculations at once, preferably requesting several CPUs. The results can be viewed and sorted by energies. You might also choose to cluster the conformations after either the MacroModel or Jaguar calculation.

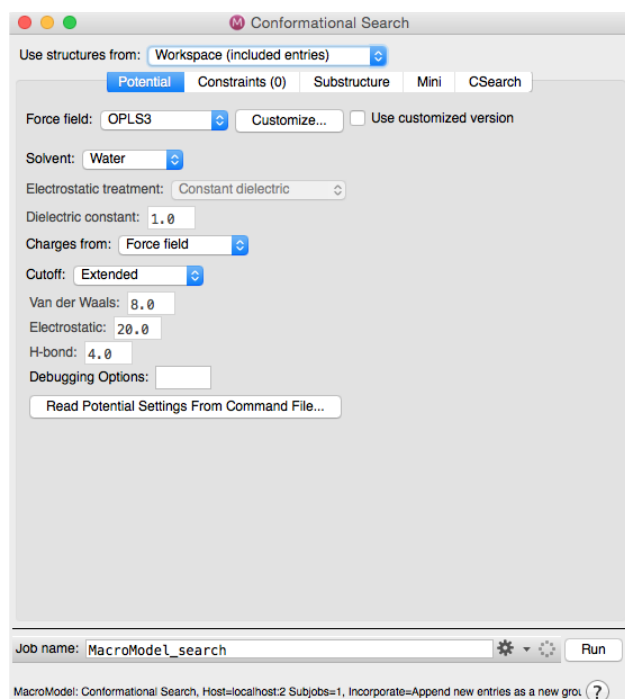
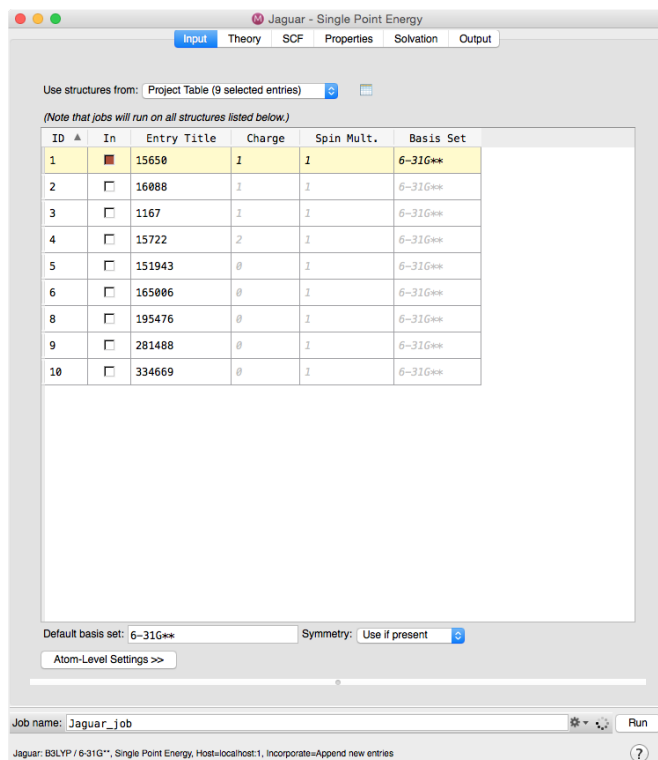


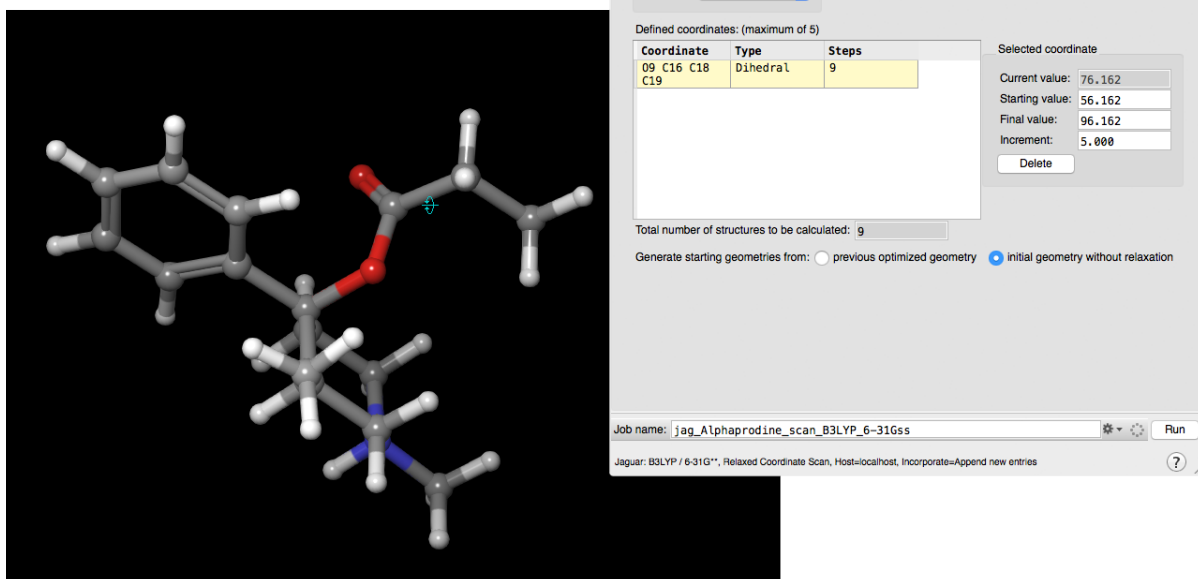
Figure 7. The MacroModel Conformational Search panel.



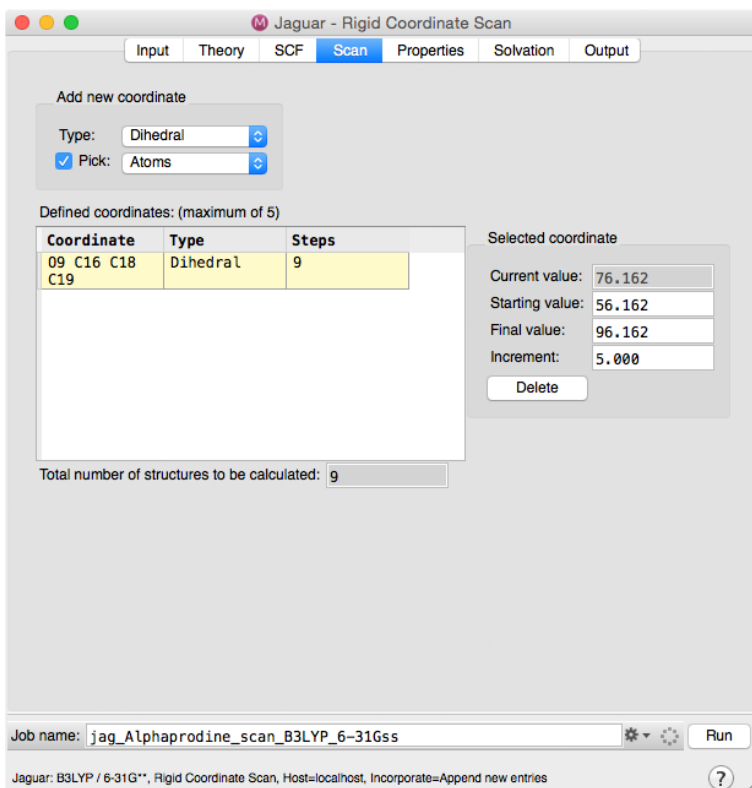
**Figure 8.** The Jaguar Single-Point Energy panel.

*Coordinate scans.* Sometimes it is advantageous to study the change of energy in response to a systematic change in molecule's geometry (often referred to as a scan). For example, you might want to see how the energy changes as you stretch a bond or rotate a group of atoms around a certain bond. For this purpose a rigid or relaxed scan workflow can be used (see the Jaguar Relaxed Scan [Figure 9] or Jaguar Rigid Scan [Figure 10] panels). A rigid scan changes the coordinate of interest while keeping the rest of the molecule rigid. Predictably, a relaxed scan changes the coordinate of interest while adjusting the positions of all the atoms in response to every change of scan the coordinate. Rigid scans are computationally cheaper because they do not involve atom relaxations (which is equivalent to constrained geometry optimizations). They can help you concentrate on the effect of the change of only one coordinate, without complicating the picture by movement of other atoms. However, rigid scans are more likely to produce highly strained and unphysical structures and even lead to atom collisions. Relaxed coordinate scans can be combined with additional constraints. Some scans are difficult to define correctly, if they lead to unphysical structures in the course of the scan. For example, scanning a torsion in a ring might be unphysical beyond certain angles.



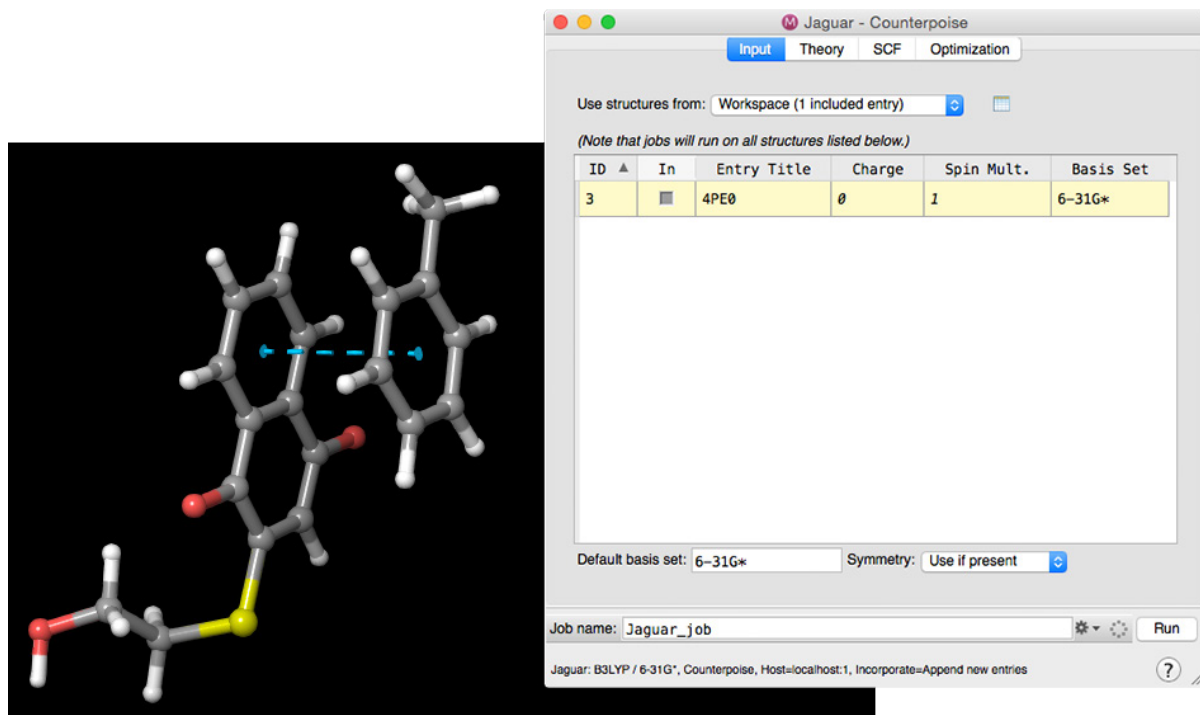


**Figure 9.** The Jaguar Relaxed Coordinate Scan panel with corresponding workspace.



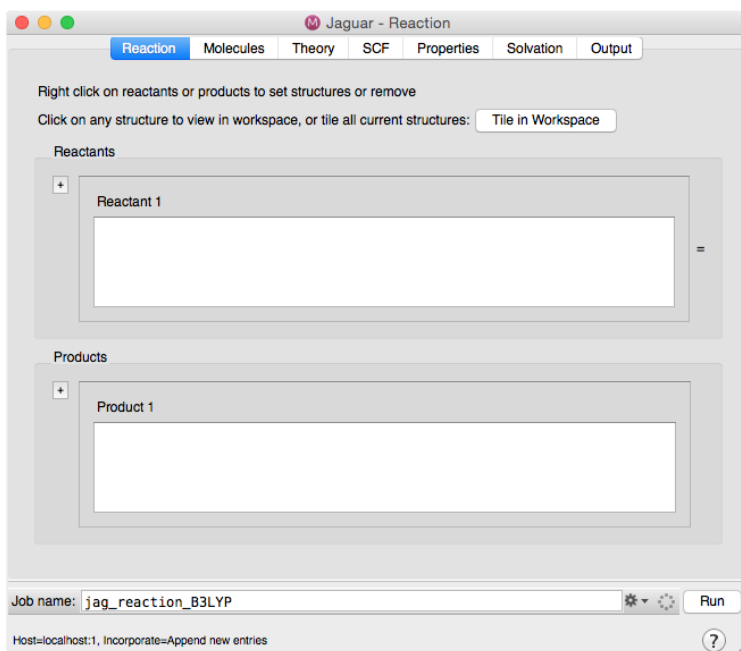
**Figure 10.** The Jaguar Rigid Coordinate Scan panel.

**Counterpoise correction.** The counterpoise (CP) correction is a popular way of accounting for the so-called basis set superposition error (BSSE). This error comes up in calculations of interaction energies of systems bound by non-covalent interactions. We would like to avoid discussing the origin of BSSE and the algorithm behind the CP correction (numerous such discussions are available in the literature) but we will mention that Jaguar has an automatic workflow that applies the CP correction when calculating the interaction energy of a complex. You can launch the workflow from Applications -> Jaguar -> Counterpoise [Figure 11]. Note that the workflow works only for complexes made out of two (but not three or more) non-covalently bound components.



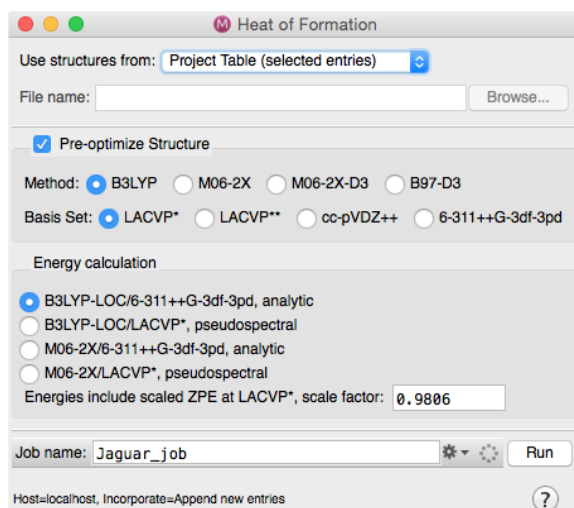
**Figure 11.** The Jaguar Counterpoise panel with corresponding workspace.

**Reaction energetics.** Often you want to compute the  $\Delta E$ ,  $\Delta H$ , or  $\Delta G$  of a reaction. Obviously, to obtain these values you need to do multiple Jaguar calculations on all the individual reactants and products. Jaguar has an automatic workflow that achieves this task for you. It is available from Applications -> Jaguar -> Reaction... [Figure 12]. When you open the Reaction workflow, right-click on the Reactants and the Product spaces to import the species taking part of the reaction, either from files or the Project Table. The reaction that you set up this way must be stoichiometric. You can use as many reactants and products as you want. Computing reaction energetics might be useful for evaluating reactivity of different species with respect to a certain target if it is known that the process is thermodynamically-controlled.



**Figure 12.** The Jaguar Reaction panel.

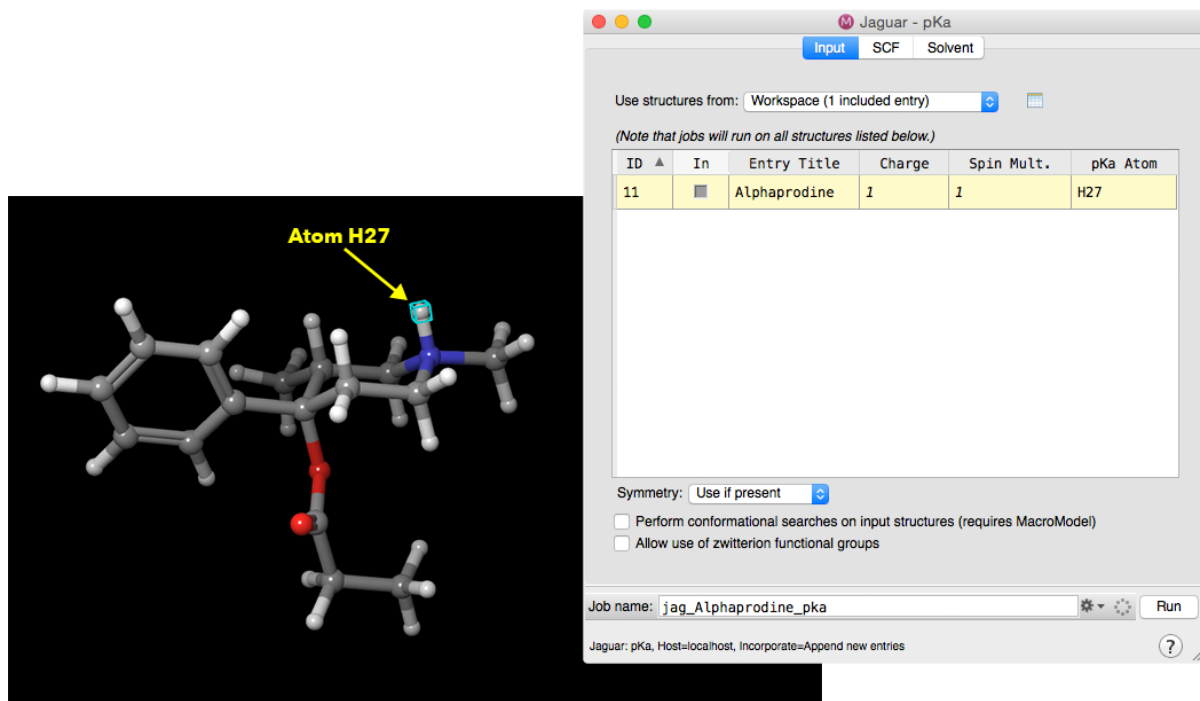
*Heat of formation.* Heat of formation is a fundamental thermodynamic characteristic of a molecule. It can be computed in a standard Jaguar workflow that is invoked from the panel Applications -> Jaguar -> Heat of Formation [Figure 13]. The heats of formation are automatically computed at 0 and 298 K. The Heat of Formation workflow helps in estimating the inherent stability of new molecular models and might even help design explosives.



**Figure 13.** The Jaguar Heat of Formation panel.

*pKa.* This workflow begins with putting the molecule in the Workspace and specifying which atom is supposed to be mostly responsible for the molecule's pKa (the specified atom is the so-called pKa atom)[Figure 14]. The

molecule must be in the correct (dominant) protonation form. An important assumption in the workflow is that only one protonation form dominates the pKa. Should multiple forms contribute in a comparable degree, individual pKa predictions need to be performed on these molecules, and the results of these individual calculations need to be combined for a final pKa. If the form you are starting from is a protonated form that is going to have the hydrogen removed to give a deprotonated form, the pKa atom specified must be the hydrogen. If the form is a deprotonated form that is going to be protonated, the specified pKa atom must be non-hydrogen. It is greatly recommended that you perform a conformational search as part of the workflow (click on the corresponding checkbox). Avoid using zwitterions in this workflow, but if you must, then enable the zwitterion correction (again, check the appropriate checkbox). The zwitterion correction works best for common zwitterions in which the positive and the negative charges are separated by a few angstrom.



**Figure 14.** The Jaguar pKa panel with corresponding workspace.

## 8. Troubleshooting and what you need to be aware of

Quantum chemical calculations are often made easy on the surface, but at their bottom they are extremely challenging. They require a lot of computational power and are highly complicated, both on the physical and engineering levels. Should any part of the calculation or workflow go wrong, the final result might become untrustworthy, clearly wrong, or even unavailable.

We have discussed the basics of quantum chemical calculations and workflows. Now it is time to talk about the possible problems (which can be either the user or the program error) and ways to recognize and fix them.

At the beginning of every calculation, make sure the molecule you are acting on is physically reasonable and is what you actually intend to work with. Many clearly wrong results are generated with molecules that are missing atoms, have an incorrect charge or multiplicity, contain obvious steric clashes, unreasonably long or short bonds,

etc. Often to explain a seemingly nonsensical result produced by Jaguar it is enough to carefully inspect the initial structure and find an obvious error or inconsistency in it. When specifying the basis set, make sure it is available for all the atoms in your molecule (consult the Jaguar User's Manual for a table that indicates which basis sets are consistent with which elements).

As we indicated earlier, one critical step in every DFT calculation is the SCF convergence process. Checking the quality of the SCF convergence is the next step when investigating an unsatisfactory result, after ascertaining that the initial structure was correct. Ideally, there should be 5-15 SCF iterations that lead to a converging energy in a smooth or even monotonic way. In some rare cases it might be normal to converge the SCF in over 20 iterations. However, if the process takes many more iterations than that it's a source of a trouble. Either the process is not going to converge (possibly, running out of the maximum number of iterations), or it may land on a physically unreasonable or undesirable solution (such as an excited state). Schrödinger's Knowledge Base website contains an article about troubleshooting bad SCF convergences from which you can learn the multiple things you can do to try resolve the problem. Essentially, the following steps can help (one at a time, or all together): start from a different guess, increase the accuracy level all the way to Fully analytic, set `nofail=1` in the Jaguar input file, remove the diffuse functions from your basis set, if possible.

Next we should watch for possible problems in geometry optimizations. Ideally, it might take 5-15 optimization iterations, each of which involve a SCF convergence, for the geometry to converge. The details of the geometry optimization convergence can be seen in the Jaguar output file [Figures 14 and 15].

```
geometry optimization step 6
energy:                -819.72894707238 hartrees

energy change:         -3.1979E-04 . ( 5.0000E-05 )
gradient maximum:      2.9168E-03 . ( 4.5000E-04 )
gradient rms:          8.0125E-04 . ( 3.0000E-04 )
displacement maximum:  1.5192E-01 . ( 1.8000E-03 )
displacement rms:      2.2395E-02 . ( 1.2000E-03 )

predicted energy change: -2.7950E-04
step size:             0.21713
trust radius:          0.30000

molecular structure not yet converged...
```

**Figure 14.** A geometry optimization step from the \*.out file of a Jaguar job.

```

*****
**          Geometry optimization complete          **
*****

Checking the geometry optimization convergence pattern ...

best                                worst
==0=====1=====2=====3=====4==
      ^

Convergence category 1: non-monotonic convergence to an optimal structure.
Geometry optimization was OK.

```

Figure 15. The ending of a \*.out file from a Jaguar job.

The closer the initial structure is to the minimum energy geometry, the fewer number of iterations will be needed. The smaller and the more rigid the structure, the easier it will be to find the minimum. It's not unusual for the geometry optimizations of difficult structures to take dozens of optimization iterations. But should the final result be suspect for some reason, it's always worth checking that the final structure found in the course of the minimization corresponded to an energy minimum or at least was close to it. If the minimum is not found this might create problems for such properties as vibrational frequency calculations (shown by the existence of one or more non-physical, negative frequencies). For a better chance of locating the minimum, select the "Switch to analytic integrals near convergence" option in the Optimization tab of the Jaguar panel, or set `nops_opt_switch=10.0` in the Jaguar input file. This option will switch to a fully analytic calculation (most accurate) at the end of the optimization (where the convergence criteria are a factor of 10.0 away from convergence).

If an individual Jaguar calculation or a workflow produces an unreasonable or suspect result, it is always worth checking the structure and the relevant Jaguar or workflow output files for problems in this order:

- the molecular structure itself (are there any clashes or missing atoms?)
- Jaguar settings and keywords (was an incompatible or non-recommended keyword combination requested?)
- SCF convergence (was the convergence achieved in a reasonable amount of iterations without large energy jumps?)
- geometry optimization convergence (did the geometry optimization finish with a minimal energy structure?)
- properties calculations (did the property calculation generate any errors or warnings?)