bio1NFORM
The integrated informatics news source

UW Team Creates High-Performance Workflow To Explore Molecular Dynamics of Proteins

July 7, 2008

Valerie Daggett
Professor of Bioengineering
University of Washington

Valerie Daggett, professor of bioengineering at the University of Washington, has coined the term "dynameomics" to describe an ongoing effort in her group to simulate the molecular dynamics of all known protein folds in their native and unfolding states.

The initiators of this multi-year, computationally intensive venture — part technology development, part computational biology, part proteomics, part biophysics, part software engineering, and part high performance computing — recently published a series of papers in Protein Engineering Design & Selection to describe some recent findings and computational methods (available here, here, and here).

The goal of the project is to perform computational "excursions" that depart from the static, three-dimensional protein structures generated by NMR and X-ray crystallography via molecular dynamics simulations that reveal atomic-level detail of protein movement and dynamics, folding and unfolding.

These simulations track the three-dimensional coordinates of all the atoms of a protein — usually over 100 residues with more than 1,600 atoms — and its solvent for periods of nanoseconds to milliseconds.

Daggett believes that these methods can empower drug design and help researchers understand diseases related to protein misfolding.

In order to support the project, Daggett's team has developed a high-throughput informatics workflow that includes a number of homegrown and adapted tools to store, manage, and analyze its results.

The number-crunching for the project is performed on the lab's cluster, a 5,130-node, quad-core Intel Woodcrest system running Windows Server 2003 R2; and at Seaborg, an IBM SP RS/6000 Power3 supercomputer based at the National Energy Research Scientific Computing Center at Lawrence Berkeley National Laboratory.

Daggett's group is collaborating with Microsoft, who helped set up the Windows cluster and developed a hybrid database blending multidimensional and relational architectures.

BioInform recently spoke to Daggett about the dynameomics project. An edited version of the interview follows.

Your simulations can be viewed as movies showing proteins as they move and jiggle about. Have any skeptics come by your lab asserting that isn't how proteins move?

Sure, but there are more knocking on my door who say, 'Will you do mine, too?' I have already gotten a dozen e-mails today of that sort.

You have coined the term dynameomics to describe your project. How do you describe yourself professionally?

Things have become so interdisciplinary. Physicists call us biologists or biochemists and biochemists call us physicists, so I'd say biophysics. We

do modeling, but most of what we do is computational chemistry, or computational biophysics. We are not modeling *per se*; we are using real physics and then computing these trajectories.

How does your work relate to the Protein Data Bank?

What we have is complementary to the PDB. That's a static repository, which, in some cases, will help you say something about function. It's certainly incredibly useful.

But if you want to go into more of the details, [and] really look at protein behavior as related to function and disease, you have got to let [the protein] move. I sort of think of us as the fourth dimension to the PDB; you have got the three-dimensional coordinates and then we put the time component to it.

I have the utmost respect for crystallographers. We use their structures as starting points for what we do. But you really need to get that dynamic component to say something. Let's see an active site of an enzyme, [and] have that substrate diffuse in. There is usually some flap or pocket — it rearranges or adjusts [itself] to the substrate or it changes shape to facilitate the reaction. Things like that all those require motion.

As an analogy, picture a horse, a racehorse with a jockey. … What can you say about the horse's function from the static picture? You might see that all hoofs are off the ground, but how does it move? You can take time-lapse pictures of this horse and create a movie, [and] in effect I think it's the same thing with a protein.

You see this horse as it gets going ... then you can appreciate that the average picture of a horse just standing is very different from all the excursions around this average that the horse undergoes to make it down the racetrack. Crystal structures of proteins are analogous to the average standing horse. But there are all these excursions [from the average] that molecular dynamics gives you access to.

How many simulations have you performed to date?

We are at 500 proteins and a little over 4,000 simulations. It's a data-management nightmare. We have similar problems to scientists in astrophysics; we have over 64 terabytes [of data] now.

What set you onto your data-intense journey?

It started … at least five years go. We tried taking our simulations and asking more general questions and we realized the statistics were just lousy. That is when we said we have to do this in a more systematic way and walk ourselves through all known protein folds, so that we have coverage of sequence space, protein fold structure space. That's when dynameomics was born, but it was really out of frustration, wanting to learn more general things about our earlier simulations.

What made it possible was an Innovative and Novel Computational Impact on Theory and Experiment award [from the US Department of Energy] in 2005, because these are massive, computationally intensive simulations, and they were just were not possible on our little clusters at the time.

When we got the award, there were three given in the country, [so] the Department of Energy gave us a lot of computer time. That kicked it off — they liked the project, [and] they continued to give us computer time over the years. This year they gave us 10 million CPU hours, which is a huge allocation. Some have told me it is the largest civilian allocation in the country. I don't know if that is true, but it's certainly keeping us busy. So we have been at it for a while, but we have been in data-collection mode.

[To prepare the datasets for the DOE's supercomputer Seaborg] we set up the simulations, meaning we check the proteins, we solvate them, we get them all ready for the production part of the molecular dynamics calculations. At that point we ship them down to [the National Energy Research Scientific Computing Center in Berkeley,] California, [where Seaborg is located], then we run them there.

When we started with Seaborg, you couldn't get in the queue to run your job unless the job used 1,000 CPUs at once. This is a massive thing for little tiny proteins. It meant all kinds of coding to package our job so the queuing system thought it was like the simulation of the universe. We had to make our proteins look bigger.

How did you select the proteins you wanted to simulate?

Our official targets are 1,130 [proteins], 30 of which are on our website. We went to the PDB and looked at how they are classified into fold families based on their overall structure, then ranked them according to fold within a fold family.

The top ones will have hundreds of cousins in this ranking, and then you start to move down this list and then maybe by 500 or so you only have a couple of examples in these families. So it is highly skewed.

The reason the top 30 folds are important is that they represent 50 percent of all the structures we have ever seen before.

So after you selected your target list of 1,130 targets you prepared them for in silico production and simulation, which involves putting them into in silico solvent. Can you explain that?

In real life, proteins — or only very few of them — work as crystals. They are most often in a solution, or solvated. So that's one potential problem with crystal structures and packing effects on the structure and dynamics.

So [for our work] we remove the protein from that kind of artificial environment, which is really crucial to get the structure but may not represent the proper environment for function. Then we solvate it and make it look more like what it is going to see in the body or a test tube or something.

There are different ways of doing simulations. [For example,] lower resolution versions, say, when you don't want to expend all the computational time to really have all the atoms in there, so you might put in sort of a fake solvent, without putting real water molecules in there.

Explicit solvent means we really have got the water in there. And it's represented the same way as the proteins — you have hydrogens and oxygen in there [with all the atoms from the proteins].

Before you could do simulations you shopped for the right software, and it seems your shopping cart ended up rather empty, so you developed your own: *in lucem* Molecular Mechanics, or *il*mm. What is different about your software?

I was involved as a graduate student with a package that is used by a lot of people in industry and academia called AMBER. As a post-doc, I was in a lab where they used software called ENCAD [written by Stanford University's Michael Levitt, who is chair of the computational structural biology department] and which isn't distributed. I liked ENCAD; it did a better job of capturing the detailed protein behavior and dynamics.

I would have students who would graduate and all of a sudden they had to go use other code that was substandard for what they wanted to do. … So I thought, 'This isn't right; we have to do this on our own, so that we have something [we] can release, and which people can take with them when they leave.'

It's taking years [to develop *in lucem* Molecular Mechanics]. I am not sure it's over; we keep finding new things to add. It took two to three years of serious programming and testing before letting it go.

It's not because I go out to find problems to solve from a methods development point of view. I would be quite happy if someone had solved the problem and we could get on with the science. … But we just couldn't find things that worked for us.

Most of the software out there in this computational modeling area for high-resolution simulations is quite old and was first developed in the late 60s and really got going in the 70s. Much of the code is just carried over and people have added to it over time. This is old clumsy code, hardware has changed, software engineering approaches have changed. We really wanted to get away from that; we wanted modern code.

A student in my lab, David Beck, used his computer science background to tackle this [and write the program], so our code is all parallelized, in different ways for different architectures. It's very efficient. It's parallelized at the kernel level so that we can parallelize analysis, all kinds of routines. It's modular, so it's easy to add different things, either from the point of view of analysis or hooking up with a database or for doing different types of simulations.

One of the important things for us was to have something modular, clean, not carrying over the corners people had cut in the past in the hopes of making things run faster. … That doesn't give you continuous clean trajectories.

It has been important for me to [do the science and programming]. It had to be happening as we were doing it. We've written a program and intend to distribute it. We have made it easy for people to add their own analyses. They won't have to write their own engine for doing the simulation; they can tack on their own and hook it up as a module.

How long does each simulation run?

They run at least 21 nanoseconds and a lot of them run at least to 100 nanoseconds. [In] aggregate we are up to about 65 to 70 microseconds, to give you a feeling for the scale, beginning almost to 0.1 milliseconds. So it is a pretty large time range. At the other end, the small end, we are

integrating these equations to look at the molecular dynamics on the femtosecond timescale, so it is a huge time range.

We do 32 automated standard analyses [for example, looking at various gross structural changes, assessing secondary changes, calculating the number of contacts between protein atoms and the solvent accessible surface area] but then we do a lot more. It's actually pretty easy to run a simulation. The hard work comes in analyzing what you've got. We're constantly writing new programs to do analysis. Because you will look at a movie and say, 'There is something interesting going on here. How can I analyze that?' So you are back to the drawing board writing code.

When you collect the trajectories, you have in effect all these snapshots, [and] then we start calculating like mad against [an] experiment. Before starting this project, we had 20 years of running simulations and then in gory detail comparing them against experiment. That is sort of the foundation we are coming from.

Where are you housing your data?

We had to build a database for the coordinates. At first it was rather static; we thought of it more as a warehouse. Then as we began loading these things, we thought we should be able to do a lot of analysis within the database and make it much more flexible — try to aim for interoperability with different programs and operating systems. That is where Mathematica comes in. Mathematica doesn't talk to our ***il***mm, it talks to the database, and ***il***mm talks to the database.

[In terms of the database], when we started this, we were using MySQL and found that didn't scale at all as we were going to move to 1,130 folds. It got too big and would just screech to a halt, we couldn't do queries, and it wasn't fast enough. Oracle is way too expensive. I never wanted to get into the database business, but it kind of was thrust upon us.

So this is when we moved to Online Analytical Processing, OLAP, a multidimensional database [mainly used in the financial sector]. With the database work, we started working more closely with Microsoft. There are different OLAP vendors ... We worked with the Microsoft version.

OLAP had never been used in science before and we met with a lot of resistance. They'd say, 'You can't use OLAP for that,' and we'd say, 'Why not?' We have multidimensional data, plus it's static in the sense that you add your coordinates and you then query these coordinates, or it can be metadata but you are not changing them either. Of course in science, we don't want to change our data. … So that kind of database works really well, but they didn't appreciate that at first.

So you transferred everything to OLAP?

We still have two [systems]. OLAP is fine for certain things but it doesn't have a lot of the tools that you need to maintain a database or back it up, or load the trajectories. So SQL is a front end to OLAP.

What we do depends on the kind of query you have. In some cases it is much faster under SQL; other times it is much faster under OLAP. So it is really nice to have the two.

For example, where OLAP is really much faster than SQL is when you want to do cross-simulation queries. Let's say you want to go find a six-residue helix in 4,000 simulations, it's really fast for things like that.

What kind of knowledge mining can you do with your data?

I am happy right now with the system and how it is working. So now we are still generating data and also getting in there and mining the data, getting interesting things out of this database. This flurry [of papers] is really because we are … trying to answer scientific questions.

Along the way we thought we discovered a new form of secondary structure and on the eve of publishing found it had been proposed by others — [Nobel Laureate Linus Pauling and Robert Corey, both of the California Institute of Technology] — and that we had been scooped by over 50 years. … They had predicted it and we ended up seeing it in our simulation. This rare structure, we think, is linked to amyloidosis in Alzheimer's and mad cow disease. It is one of those transient excursions that we were able to see through molecular dynamics and one that we think is linked to pathology.

It is an example … showing that excursions are really important, that we have to really look at how proteins are really moving. We took these models from the simulations and have been doing drug design. We have things that weakly inhibit propagation of prion diseases and submitted a patent a few months ago.

I think there is going to be a lot [of information in the simulations about] SNPs, things you cannot see in the average crystal structure. The excursions, the variants … the mutations aren't enough to kill you; people live with these. There are subtle effects and they really are not seen in the static state.

It’s exciting after you have worked for years collecting the data, you would like to think you can do something with them.