



## Futures in Biotech, 27: Folding@Home at 1.3 Petaflops

### Leo Laporte

Bandwidth for Futures in Biotech is provided by Cachefly at [cachefly.com](http://cachefly.com).

### Marc Pelletier

This is Futures in Biotech, episode 27 for Wednesday December 12, 2007. Folding@Home at 1.3 Petaflops.

Futures in Biotech is brought to you by Invitrogen, the Zero Blunt TOPO PCR cloning kit. It combines a 5-minute TOPO cloning with a unique zero background technology. Visit [invitrogen.com/topo](http://invitrogen.com/topo) for details.

[Music]

### Marc Pelletier

We're really lucky today because we have one of biotech's true visionaries. He's at the helm of the world's fastest computer, and what's amazing about this is that computer is powered in large part by Sony PS3s, the PlayStation. And those PlayStations together are running over – now let me get this straight, 1,000 trillion floating point operations per second. And as you can see his expertise is outside of mine, it's actually much broader than mine, so I've asked Steve Gibson, our tech guru at TWiT network and host of the great podcast Security Now to help co-host Futures in Biotech.

So our guest is Dr. Vijay Pande, he's an Associate Professor of Chemistry and Structural Biology at Stanford and founder of Folding@Home. So on to the interview.

We had a guest on Episode 15, Larry Smarr who's the founder of NCSA, and before he was doing that he was an astrophysicist studying how supernovas and black holes collide. You're looking at the other end the universe, are those challenges the same and what kind of problems are you trying to address with Folding@Home?

### Dr. Vijay Pande

I would say the problems are actually fairly different, not just in scale but in nature of what we want to do. I think in terms of Folding@Home there's two broad areas that we're interested in, although there are a lot of things we have interest in. So the canonical thing we're studying is protein folding and misfolding. I think your audience is probably familiar with this but protein folding is the process in which a biological molecule or protein assembles itself into a particular shape. The shape is actually really important for its function. And misfolding comes into play as this act of self-assembly doesn't always work well, and in many diseases such as Alzheimer's or ALS or Parkinson's or Huntington's, these proteins don't just not fold correctly but they misfold into structures that are toxic, and that's what leads to the pathology that we see now in these diseases. And so our desire is to both understand this process when it works and when it doesn't work from a biophysical point of view. But also we are very interested in developing therapeutics, whether that might be small molecules or other possibilities to really try to make an impact on Alzheimer's disease in particular.

### Marc Pelletier

Why would you go with the modeling approach rather than biology? Is there, do you think, a time or a technical possibility to actually study the dynamics of how proteins fold? Can it be done without using modeling, can you actually look at the proteins fold?

**Dr. Vijay Pande**

[4:35] So the experiments here are really, really challenging. And my experimental collaborators and colleagues have done some really amazing things. But you can't do what you really want to do. You really would love to be able to look at an individual protein in atomic detail and see what it was doing. Usually in structural biology you have a choice between something like a crystal structure, which gives you great spatial detail, you can see all the atoms but gives you no temporal information at all. It's just sort of locked in one particular structure. Or you can do something like single molecule experiments where you can get some really interesting temporal information but the spatial resolution and spatial detail is actually relatively poor. And it's hard to sometimes interpret what it really means.

So simulations let us have it both ways. What's particularly challenging for something like protein misfolding is that protein misfolding is a very heterogeneous process, in that when you have a soup of proteins that are misfolding it's not that they each reach each stage concomitantly, like they're walking in an army in march step. They're doing things very stochastically and randomly. And so at any given moment you'll have a whole mess of things, a very heterogeneous mess, and that's really the challenge there is to – because it's so heterogeneous you can't use standard structural biology experiments on it.

**Marc Pelletier**

Let's break this down a little bit. So that we can imagine what's going on with a protein that's supposed to be folding, what exactly is that? It's a chain of amino acids, that starts off being made as beads on a string. Then what happens and what are the timescales for this?

**Dr. Vijay Pande**

That's a good question. There's a couple of different types of timescale. There's the timescale for synthesis, which – adding each amino acid takes some fraction of a second, so it's relatively slow to synthesize, but in our body proteins are spontaneously folding and unfolding all the time. A typical protein might fold somewhere in a fraction of a second to a second, maybe even slightly longer. But there are some real speed-racers there that can fold extremely fast, sometimes as much as a million times a second, or 1 $\mu$ s. So we're interested, on the biophysical side we study some of these speed demons because they're interesting just to know why they fold so fast. Something like aggregation in Alzheimer's disease can be extremely slow, it could be like hours, days.

**Marc Pelletier**

Can you approach technically – are there machines that will allow you to do the modeling of the molecular dynamics of folding to take you up to the second range or is that possible?

**Dr. Vijay Pande**

Yeah, that's one of the things that we've been interested in for a while. There's a couple of choices that you could make, and usually what one has to do at this point is you come into a fundamental barrier that if you want to take a detailed model and simulate a long timescale it just doesn't work. Because detailed models maybe you could simulate nanoseconds, essentially billionths of a second. And if you want to simulate a second and you can only do a nanosecond a day it will take a billion days to get to what you want, which is obviously a long time to wait. Even for some simple things, like you want to do a millisecond, that would take a million days and that's still obviously a really long time.

So the community either – some people have focused on events that are really fast and study things that are maybe interesting in the nanosecond timescale. The other approach is to say I'm going to simplify the model to the point where the model is simpler but maybe we can get the

longer timescales. And I was actually dissatisfied with both approaches because when I look at the simple models I don't see that they're really going to be predictive for the types of problems we're interested in. The inherent timescales are so long that studying things on a short timescale wasn't going to be all that interesting.

And so what we did – and we had some original ideas for this in 1999, we had some ideas for how we can, instead of waiting a million days on one machines, could we wait let's say ten days on 100,000 machines. Or 100 days on 10,000 machines. And that sounds very intuitive that that would be the case but it doesn't always work that way. An analogy often used is it takes nine months for a women to give birth to a baby, but nine women can't give birth to a single baby in a month. There are just some jobs that can't be broken up. So that was the daunting situation we had to deal with, which was how do you break up something which on the surface doesn't really look like something that you could break up into nine parts or 9,000 parts or 100,000 parts. But in '99, 2000 we had an idea for how to do that.

[9:20] And then we actually came to an interesting stage academic research wise, which is do we just publish the idea and then let someone else do it and just move on and do something else. Or do we actually publish the idea and then try to actually use the method to do something. If we actually wanted to use it then we have to have the computer power to back all this up, not just show that this could happen if we had that sort of computer power. That's what led to us starting the Folding@Home distributed computing project. That was really the only way for us to get the computing resources we need to basically release our software on the Internet and have people throughout the world run the software to allow us to gain a computer resource that's vastly greater than what we could get from just about any – certainly greater than what we would get at a supercomputer centre, for the types of calculations we're interested in.

**Marc Pelletier**

So how did you decide on a platform? This is the part where I'm really glad Steve's here because you had to build an infrastructure to solve these mathematical problems. Steve?

**Steve Gibson**

Well I'm very impressed with what I've seen. I've had the client running on my machine all afternoon and it's – whatever it's doing it looks wonderful. I have no idea what's going on but I'm more than happy to throw my CPU cycles into the mix. How many systems do you have typically available within this distributed network?

**Dr. Vijay Pande**

There's a couple of different ways to answer that question. One is basically in terms of how many individual machines have ever participated, and that's over 2 million right now. Then you can ask the question well how many are really participating at this very moment, and that's roughly about 250,000.

**Steve Gibson**

And you would look at it in terms of what, like frames that have been processed by these machines?

**Dr. Vijay Pande**

Yeah, we look at to see whether the machine has sent a result back to us within some time-out period. If they take too long then we assume that they're not actually sending something back.

**Steve Gibson**

So that's where you get the quarter million machine mark at this point.

**Dr. Vijay Pande**

Exactly. We also, you may have seen we are running on not just PCs but we recently released a client for the PlayStation 3 and the PlayStation 3 is really interesting because it gives us

processing power that's roughly, somewhere between 20, 30 maybe even 40 times greater than what we could get on a typical PC. And that sounds counterintuitive to a lot of people because you think of a game machine being a toy or something like that. But the Cell processor in the PS3 is really powerful for the types of calculations we need to do. You can't go out and buy a PC twenty times faster than a typical PC, than like a 3.5GHz Pentium or something like that. Those machines just don't exist.

**Steve Gibson**

And I also saw that you have support for other GPUs, even on Windows based platforms.

**Dr. Vijay Pande**

That's right. The GPU stuff has been very experimental. We haven't been making a big push on that. I'm hoping that very soon we'll be able to release our next-generation GPU client. The GPUs actually have the hope I think to be much more powerful than even the Cell. What's interesting about them is we all know about Moore's Law and really Moore's Law is about increasing the number of transistors on a chip. For the longest time more transistors meant faster chips. Until recently when more transistors really isn't helping that much other than maybe giving you more cores within a chip. What's interesting about the Cell processor and especially GPUs is that they use their transistors in a very different way than a Pentium would. Instead of having lots of cache, they have lots of floating point units. So if you can actually set up a calculation that maps well to the GPU, and there's a lot of constraints and a lot of unusual things – it's still very Wild West in many ways running on a GPU, but if you can set that up you can get really dramatic speed increases. The peak flops are almost ridiculous, I think it's like 500 gigaflops or now soon to be a teraflop peak flops. I think very few applications can get peak flops but we can routinely get something like 50 gigaflops, and maybe soon as much as 100 gigaflops.

**Steve Gibson**

[13:33] Wow! That's just beautiful. And you get a lot of machines running that and you're talking serious horsepower.

**Dr. Vijay Pande**

The thing about it is there's this huge initiative to have a supercomputer in the United States at a petaflop, but Folding@Home is already over a petaflop. It's just about right now somewhere between 1.2 and 1.3 petaflops versus other traditional supercomputers are at 0.1 petaflops or 0.2 petaflops. The one tricky thing is that flops, people mean sometimes different things by flops. And I'm talking about not peak flops but actually sustained on our calculation. And our calculation I think we ran it on other supercomputers, I think it would run generally with worse performance than some of the benchmarks that they use to judge these machines. Not necessarily that our code isn't optimised, it's extremely optimised, I think a lot of these machines are themselves optimised to run the benchmarks really well.

**Steve Gibson**

Well I've been impressed with that I've seen. I think you've done a nice job. I notice also that the client maintains a bunch of outbound connections, but from a security standpoint you're doing nothing to lower your own network defences. You're basically, the client makes a bunch of outbound connections and that's the limit. It's very much like a web browser that goes out and connects to a web server.

**Dr. Vijay Pande**

That's exactly right. And actually if you sniff the protocol you'd see it is HTTP that we're talking to. So the clients are really just web browsers and our server essentially is a web server. The nice thing about that is that lets us go through firewalls and all the crap that it has to deal with. It also means that security-wise our stuff is as secure as a web server would be, or a web browser would be. And actually it's more secure because we don't have Java script or anything like that, you can pick up a virus or anything like that. In many ways the nuts and bolts of this is relatively old because people have been doing client server for a while. I think what's challenging about this

is the scale, I think it hasn't been until recently that people have been doing client server on such a large scale. But then especially I think the intellectual contribution is figuring out a way to break up problems into 250,000 bits. And that's the part that is actually fairly difficult and something we've put a lot of effort into.

**Steve Gibson**

Yeah, the whole idea of creating a problem then making it parallel. So in fact you can have a birth by nine women in one month.

**Dr. Vijay Pande**

Exactly. In this case it's like using 90,000 women to get it done in one ninety-thousandth of the time – a fraction of a second, like boom, there's your baby.

**Marc Pelletier**

It seems like the technical hurdle here was not just allowing, by analogy, nine women giving birth to one child, but it seems like you've made a quantum leap in the thought process to get this done such that you're able now to get 100 men to do one child. You've flipped the story on its head. Rather than develop the NCSA to solve how black holes collide you really thought out of the box and you convinced how many people – 250,000 machines at a time, so it's –

**Dr. Vijay Pande**

Probably millions of machines have participated at sometime, so it's maybe a million people something like that.

**Marc Pelletier**

That's absolutely viral.

**Dr. Vijay Pande**

Yeah, it's very viral. Although I should stress that actually there's a lot of precedents before us. SETI@home came out basically a year before us and I think that attracted a lot of attention. And even before them there was like distributed.net and prime number searching and other things like that. So I think there are many ideas we built on top of – you know, shoulders of giants to borrow the phrase. But I think one of the things that was unique for us is all of that stuff was set up for problems that were really trivially parallel, where it was obvious how you would break it up. I think a lot of problems – people assumed that distributed computing would only be useful for those types of things and I think it's only having, when we had to be forced to think of ways to break up a problem given this resource, that creativity came upon us.

One way to think about it, and this is a question I love to ask colleagues when I go visit places and give talks is that I ask them, what would you do with 100,000 processors. And it's a question that people don't think about because there's this chicken and egg thing. You don't think about doing those types of calculations if you don't have the resource. But people – supercomputer centres and other places don't build the resource if people don't think they have the calculations. So it's intriguing to get people to continue to think out of the box and think about what they could do that would ideally really be paradigm shifting, something that would be not just an incremental change but something that would make a huge jump in a given field.

**Marc Pelletier**

[18:30] This is just a sidetrack question: you've got these processors running, mostly personal computers, people's home computers, maybe some people at work. Are there any ways that you could make it a – put a dollar sign to it, not in so much as make money but to get more computer time, where if a company donates a certain amount of cycles, like Google for example, if it were to donate cycles it could get a tax break.

**Dr. Vijay Pande**

So this is something that I've looked into and I have friends that are CPAs and so on and apparently it's actually very difficult in the current tax code, the way this works. For instance if let's say you donated your car to the Red Cross, you couldn't write off the cost of the car. Not donated the car for them to keep but let's say just let them borrow it. If you let the Red Cross borrow your car for an hour, even like an hour a day, you could maybe write off some aspects of depreciation on that car, but you wouldn't be able to write off the complete cost of the car. Maybe you could write off the gas and stuff like that. And so I think it would have to be done very carefully that, and certainly I'm not a CPA or a tax lawyer or anything like that, so I could imagine there could be some way to do this but I think structuring might be tricky and beyond what we think about.

What has happened though is there have been various companies that have just, either due to their benevolence or maybe their desire for PR or marketing, have really helped us a lot in many ways. Sony has been a huge help by putting this on the PS3 and actually committing a whole development team to work with us to make sure that the science was done right. In the past Dell and Apple have helped with various hardware things. And actually Google helped us at the very early stages as well. And actually Intel also helped us at early stages. So there have been companies that helped, although in maybe slightly different ways.

#### **Marc Pelletier**

Could it be added into a – for example, an on-off switch into a Google toolbar inside of Firefox or something where if I'm only going to be browsing the Internet I might as well be donating some cycles or is going with the screensaver approach the best way to go?

#### **Dr. Vijay Pande**

Actually, early on it was in the Google toolbar. The Google toolbar that you would install into Internet Explorer. This was something where, you probably recall that Sergey Brin was a Stanford grad student, he still hangs out here once in a while and we were chatting about what might be an interesting collaboration. And out of that came a collaboration where – which was called the Google Compute Project where Folding@Home was actually built in to the Google toolbar. This came at a really important time in Folding@Home's history, this was like within the first year and Folding@Home only had about 10,000 computers actively processing. The Google Compute Project pushed us from 10,000 to 30,000. It was a huge jump for us.

More recently, now that we're up around 200 – once we even got to 100, 150, then that 30,000 which remained fairly constant wasn't as big of a fraction of computers. And so we mutually decided that might be a good time to stop the Google Compute Project. But it really came at a really important time to help push us forward and give us some good momentum.

#### **Steve Gibson**

What do you guys need in terms of – like, most? Is it more machines? If you had your one wish would it be the technology to leverage GPUs to a greater degree and just raise this sometimes 250,000 to 1 million?

#### **Dr. Vijay Pande**

I think right now, the most important thing for us is to get individual machines that are as fast as possible. We can do a lot with many slower machines, but there are limits to what one can do. Especially limits in terms of some things just need a fair amount of wall clock time. We have run on regular CPUs calculations that literally take two or three years. And if it takes just three years to do the calculation, then you have to analyse the data and write the paper, so this whole process is a really long process, especially since someone's grad school life is probably only about four or five years. So they get their one shot at their big data set, we publish it and that's that.

If you could, let's say, 20 times speed up, which is what we get out of the PS3 pretty routinely, maybe nowadays maybe as much as 30 times – a 30 times speed up takes three years and turns it into a month and a half. And that's something we're – we get to try out a lot more things, we can

try something, let it run for a month or two months, learn from and then improve upon it. That whole process, which maybe originally was taking years for us to really advance the methodology, now we can make really dramatic steps.

**Marc Pelletier**

[23:22] So are you doing one project at a time? One folding experiment at a time?

**Dr. Vijay Pande**

There are several people in my research group. There are about 20 people in my research group, and also we have some collaborators that we interact with. So maybe there's like 20, 25 people that are operating projects within Folding@Home and each one of those people might have somewhere between ten or 100 projects at any given time. Sometimes a single project may take up all of Folding@Home but that happens every once in a while. Or sometimes they're taking some various fraction. At any given time there's probably about somewhere between ten or 100 very active projects going on.

**Marc Pelletier**

Could you tell us a little bit about the nuts and bolts on your side of things? When you're – for example, if I wanted to study how, I had a long protein that I knew was a problematic disease as we found it in disease, say transthyretin which forms an amyloid in the heart. I have the amino acid chain. I know the primary sequence. How would I approach it if I were to do a collaboration with your lab?

**Dr. Vijay Pande**

Let's say for something like aggregation you're interested in studying the process of aggregation and knowing in particular what some of the intermediate aggregate structures are like. We know that the ultimate structures are fibrils for many of these diseases, but it's becoming more and more clear, although still very debatable, but there's more and more evidence that fibrils may not be either toxic at all or may not be the key toxic element, and that the small molecular weight oligomers, with just like four chains or 12 chains or 24 chains would be the true toxic species. This is I think especially starting to be much more the common wisdom in Alzheimer's disease for example.

In that case we would do a simulation where we would put four or 12 chains and we would study the dynamics. One trick is that – and you can ask how do you get to these long timescales by using many simulations. It's like trying to study a marathon, how can you break up the marathon into these bits. Because you can't simulate mile 3 until you've done mile 2. In the end one of the things that we've been really trying to push over the last few years, and I should stress we're not the only group thinking this way, there's now several groups throughout the world that are trying to help pioneer these directions, which is to really try to take simulation from something which is inherently anecdotal, where you might run one or two really long simulations and create this killer movie but have extremely anecdotal data to talk about. To try to go from something where you're running a very few long trajectories to something where you use many short trajectories to build a kinetic model.

And we basically use Bayesian data mining techniques to essentially train the model with the simulations as data. Because this method is Bayesian we can actually have not just expectation values or answers that we care about, we also get uncertainties or errors. The uncertainties are really important in science, because if I tell you a number is 5 and the experimental number is 6, was that agreement good or bad, it depends on the uncertainty. Or just even knowing how to trust anything, the uncertainty is really everything. By using these Bayesian techniques we can, one, calculate uncertainties and just have some truly statistically significant results. But secondly, if our uncertainties are poor the Bayesian method tells us where we need to run more simulation. And then we can send the network to attack those parts of the model and gather more data for that.

And it becomes actually an extremely efficient way to run simulations. In some cases a hundred or a thousand times more efficient than running just a few long simulations. And, unlike a few long simulations, which might not be possible without waiting millions of days, this is something we can do now.

**Steve Gibson**

Wow!

[Advertisement]

**Marc Pelletier**

[30:09] I'm really trying to get a picture of the bioinformatics facility that you've set up and how your lab works to break down these problems as well. Can you do this off your laptop, when you set up an experiment, and then it sends it out to those 250,000 computers, including PS3s, I'm trying to imagine – it's like trying to imagine, everybody wonders what their email, you know, where's email? It's just there.

**Dr. Vijay Pande**

I can tell you a little bit about the infrastructure. There's actually a pretty big serverside cluster handling all this stuff now. I think we have about like 400 terabytes of storage these days that handle all this stuff. Basically there's a huge number of servers that are talking to their clients directly, and they're the ones that hand out the work, and take back the work. And one of the big problems for us compared to other distributed computing projects is that we take back a lot of data, we need to really understand what's going on so we end up bringing back easily a megabyte per computer per day. That's hundreds of gigs per day and multiply that – even just 300GB per day times 300 days is a lot of data. And sometimes we might take back more than that, we might take back 10MB per client per day and something like that.

**Steve Gibson**

Does that data get reduced a lot then?

**Dr. Vijay Pande**

It does, although a lot of the people in my group are always nervous about throwing things away. But there's this interesting thing that we're working on, one of the things that we had in mind is to do distributed storage where we can use the clients as storage. You can imagine clients as essentially one big RAID device where you send data back and it's striped such that if you need to bring a file back, let's say even like a big file, a 100MB file, you don't have to wait for someone to upload 100MB over a DSL. You could take it from 100 computers each giving us 1MB or 1,000 computers each giving us 100KB, and that could come back really fast. The nice thing about striping too is we can get reliability. But you could say well then still though you're keeping track of all these copies to make sure you don't lose something.

The nice thing about computer data is we can play this game of this continuum between calculation and storage. You could store everything, or you could store a little bit less and just do some calculations from that. You could store checkpoints. And maybe you'd have to do some extra computation to get it back from a checkpoint but there's essentially this tradeoff you can do between having lots of storage and no extra computation or minimal storage with a little bit of extra computation. We're trying to explore those tradeoffs just because I think I'm trying to plan for a future where there may be 1 million active computers in Folding@Home or 500,000 GPU/PlayStation machines that are cranking away. And we want to make sure that our CPU power doesn't outstrip what we can actually do with it and handle. So we have ideas like that to handle it.

But anyways, I think in terms of asking the original question of what it looks like, when your client – when you run Folding@Home your client goes to a machine which we call the assignment server, which is a global load-balancing machine. And it will take a look at whether you're a PS3

or a PC or a Mac and some various requests that you've made. Maybe you've told us it's okay to run really big calculations, or maybe you want to run only modest calculations. We take all those requests into effect, we look at what the active jobs are and then we assign you to a particular job. That sends you to a given data server that sends you the calculation, you take maybe a day or two or three to do the calculation and then you send it back to that server, and then the process starts over again.

**Marc Pelletier**

Do you have any thoughts, Steve?

**Steve Gibson**

[34:01] No, I'm just very impressed with the way this is set up. I was surprised thought that you said you started a year after SETI@home, since SETI@home has been around for – I mean like forever.

**Dr. Vijay Pande**

Yeah, we started in – we released Folding@Home in October of 2000. I think SETI@home came out in '99 or something. Maybe '98, so maybe a year and a half.

**Steve Gibson**

Well I'm really glad to have found out about it and I'll certainly do my part in promoting it and assuring people that it's safe to use. Because it's very clear to me that it is safe to use, which is certainly a concern for a security conscious population. It seems to me that this is really advancing substantial research.

**Dr. Vijay Pande**

I hope so. I think it's also just a different model of doing research which I think is interesting. One of the key things that we're trying to do with the way we do things is really push computational biology from the point of where it's anecdotal and cute for making movies, but you can't really compute anything. There's colleagues of mine who will say that they're not really interested in computing numbers, that couldn't be compared with an experiment. That's not what simulation should do.

But then I have other colleagues, especially the ones who actually do the experiments who say that well if you can't give me a number how do I know if I can trust you. It's kind of like, when you read a horoscope, they always seem like they're right. If you give something qualitative it's very easy for there to be fudging and whether what you really meant. But if you actually give a precise quantitative prediction then you can really know whether your method's working or not. It's amazing how much effort it takes though to go from anecdotal stuff to getting numbers that you can really rely on. And especially when you're doing things like drug design it's something where every little thing makes such a big difference that to go from something that makes pretty movies to something that actually could be used to design drugs and is very accurate, takes a lot of effort.

That's how we're spending this CPU power, is to really try to make this a legitimate tool to do things that couldn't be done in these areas. I think the payoff will be is when we actually have these predictions are things that you could never really get from an experiment. And we have a series of papers now that have been submitted and that I'm excited about, but it's probably too early to talk about, where I think every year we're perfecting our ability to do this. For methods like drug design where you can do calculations that I think most people probably thought would not be possible in terms of the accuracy that we can achieve.

And really, the dream is to really think about biophysics and drug design the way we think about building a bridge. So I live in San Francisco here, or a suburb of San Francisco, and they're redoing the Bay Bridge here and the thing costs like \$2 billion or something like that, which is a lot of money. If you think about a new drug, a new drug costs about \$2 billion as well. For this \$2

billion bridge you can imagine the analogy of what it would be like if people designed bridges the way they design drugs. They would have some idea for a bridge but they really wouldn't know whether the bridge would work or not. So you send rats across the bridge or something like to see if they survive. And then you send people across the bridge that really have to get to San Francisco really badly. And it's very empirical because it's not that people who are designing drugs don't know what they're doing it's just that it's a really, really hard problem. Designing a bridge is actually relatively easy because if you understand the physics behind it.

The dream for decades has been to use a physics-based method, because the laws of physics are the same for bridges and atoms. But to use a physics-based method to describe drug design, in many ways this has been oversold for decades because I think there have been many companies that have been founded and failed trying to demonstrate this.

I think one of the major problems was that we were just orders of magnitude too far behind in terms of computational power. And the types of things that we can get from Folding@Home really can make dramatic pushes ahead in terms of the types of resources that you would normally have. A typical researcher might have maybe a 1,000-processor cluster, or something like that. Maybe even only a 100-processor cluster. And our resources are somewhere between a hundred and a thousand times greater than what a typical researcher would have. What that allows us to do is it allows us to do things that people might only be able to do routinely maybe ten years from now. Hopefully we can try to figure out what will be the smart things to do and the right way to do it, and to demonstrate this works such that ten years from now this is the way these calculations will be done.

**Steve Gibson**

That's just very cool.

**Marc Pelletier**

[38:27] Do you foresee within the next, well in ten years or in 20 years from now, that we'll be able to model a biological problem? We interface – you can interface with the human organism with drugs, much in the same way you load software into hardware. And while genetic diseases is hard and our ability to modify the human organism genetically is – it's pretty tricky, but we can interface with drugs. We've been doing it for aeons. But there's still so many diseases that are uncured. We're going to be -- probably the best way to approach those is biochemically, with drugs. Do you foresee in a short period of time, within the next maybe even ten years, the ability to do the entire screening process in silicon?

**Dr. Vijay Pande**

I think it's going to be quite a while before one does the entire thing in silicon. Even within my lab, we're known for our computation but we do a fair number of experiments these days. I think for the next ten years the solution will be a strongly concerted effort between tightly coupling experiments and computation. What I think computation can do is that with experiment typically you're looking for a needle in a haystack, and the haystack's huge. With computation I think we can really, really greatly narrow the haystack to the point where we can screen with accurate experimental screens and just get to the interesting compounds faster. In the end I think it will be probably a long time before we do the dream where we just hit the button and we go get coffee and maybe a couple of days later the drug comes out. But, in fact I don't think we need that to be the case. I think what we need is we just need help.

**Marc Pelletier**

Sure we do, sure we do. If you've got a disease and it's going to kill you it'd be nice to be able to say, well here's the protein that's wrong in my body and you're going to push a button and do that.

**Dr. Vijay Pande**

Well I think one thing that is very exciting to think about, and I think this will happen more and more, is the idea of once we can deal with ADME-Tox [absorption, distribution, metabolism,

excretion toxicity] issues, whether something is going to kill you. Something might be a great – might kill the disease, take care of the disease but it might actually kill you too, which would not be very good. There have been more and more efforts to try to go after ADME-Tox issues computationally, and I think I'm actually very excited about some of the things that are emerging right now. Folding@Home isn't involved with that but other aspects of our work is. I think the ability to do personalized ADME-Tox would be especially interesting. Actually it's interesting when you look at even like drugs for women and kids: kids are basically treated as little men, proportional by weight for the most part. And women are treated as slightly smaller men. There's all of these issues that are just so difficult to be able to go after. And everyone is different and you could imagine if you could build up an intelligent way to do ADME-Tox that might be the natural first thing to do in terms of personalized medicine, just to know that what you're giving is not going to be dangerous.

The next step would be is to try to really design something quickly to be able to go after maybe individualized situations, but the other dream and something that people in my lab are interested in is to go after things like viral infection. Especially to be able to come up with antivirals quickly enough to avoid influenza pandemics or Ebola or HIV pandemics. That's where speed will be really important because the viruses are mutating fast enough, if you take two years to come up with a virus it might be too late. So I think there's a lot of interesting things to do but I think for the near future everything that we're going to do, and I think anything that's going to happen in pharma will have to deal with a tight coupling between the two. And I think pharma can be a little conservative in their approaches for things, and I think that makes sense in many ways. It might be from the biotechs that are usually much more aggressive about incorporating new technology where you'll see some of the most exciting things.

**Marc Pelletier**

[42:37] Here's a potential exit question, though Steve you're welcome to. Here's one question: how many atoms are you looking at in an experiment?

**Dr. Vijay Pande**

You mean in a simulation?

**Marc Pelletier**

Yeah, in one simulation. Just to get an idea of scale.

**Dr. Vijay Pande**

It really varies. We run some simulations that are easily millions of atoms, and then we run some simulations that are thousands of atoms, depending on the situation. One thing that deals with the big difference there is how we handle water. Do we handle water in a mathematical continuum way or do we actually have to an explicit H<sub>2</sub>O for every single water molecule that's there. And if we have water present explicitly the atom count can grow really fast.

**Marc Pelletier**

A typical biological machine, a protein complex, would be how many atoms?

**Dr. Vijay Pande**

It'd probably be hundreds of thousands to millions.

**Marc Pelletier**

You're setting up basically the framework to do atomic level molecular dynamic experiments. Do you see it at a point where we can model the human body – this might be kind of crazy but – in the same way that Google Earth is doing an entire dynamic – with time, with their continued new data – image of a functioning human body? Can you take it from a molecular machine to an organism?

**Dr. Vijay Pande**

I think that is a dream but that probably is fairly far away. I would be happy to just be able to go after the individual complexes. We're putting a lot of effort in the ribosome, we have colleagues that are studying RNA polymerase, these are huge, interesting, complicated molecular machines. I think even just being able to tackle those is still very much a challenge but I think we are getting to a point where I think we can make some really interesting observations. I think in time what we'll see is that you may never see a model of the cell where the whole cell is described all in detail. What's becoming very exciting and popular, although it's probably still more at the idea stage than being fully realised is the concept of multi-scale molecules, where you might have an atomistic model of the ribosome that feeds into some systems biology model of translation.

And so the idea is that maybe you don't need to simulate every single detail atomically. When we simulate cars crashing or airplanes moving through the sky you don't have an atomistic model for the airplane, because you don't need it. And moreover you don't want it...

**Marc Pelletier**

But it would be fun.

**Dr. Vijay Pande**

Yeah, it would be fun but it's – I'd rather do a million cells with multi-scale than maybe doing it all atom. I think what you get out of it though is you have the effective properties of an all-atom model throughout everything. But I think this is obviously very far down the future and probably might be things more for my students to be handling in the future more than I'll see in my career. But I would just be very happy with the ability to just use computation to make predictions in such a quantitative way that biology would get to the point where they couldn't imagine a world without it. That it's – it's interesting to talk about putting people out of business but I think that's really far off. What's more likely is get to the point where they can't live without it and where they really need the insight that you get from computation to be able to shed light on their experiments. Whether we're talking about drug design, or biophysical or structural biology. And I think that we're right on the cusp of.

[46:07] I don't know if you ever heard of this thing called the Hype Curve. It's this plot where hype is on the vertical axis and time is on the horizontal axis. And it usually looks like there's this big peak as you go – peak in hype as you through time, then hype falls down tragically and gradually starts making its way up. People map all these things to the Hype Curve, like you could talk about railroads. People thought railroads were going to revolutionize the world. Think about how long it takes to go from New York to San Francisco by wagon car or something like that. Or just by ship where you have to go round South America. Railroads really did revolutionize things the way the Internet revolutionized things, except it was over-hyped in the beginning but then gradually became very useful.

Computational methods are on this sort of late part of the Hype Curve. In the '80s they were hyped to the point where they were given as this panacea but I think we were still many, many orders of magnitude too far away in terms of computing power. Now I think with things like Folding@Home we can get to the point where we can simulate things of interest in a statistical way that can make direct predictions and comparison to experiment. It's the point where I think it's finally ready to have a seat at the table with these other methods. That's in a nutshell what I'm hoping to be able to do with Folding@Home, to really make this method something where people could not live without it. That would be – I'd be very happy with that.

**Marc Pelletier**

Well let me just say that on this WEEK in TECH Network everybody's always talking about the latest technology, and hype plays an important part of what makes this fun. And you have bragging rights, you have the most CPUs. Is it a Guinness record of the most CPUs working on a project?

**Dr. Vijay Pande**

Or I think it's probably most flops, I think. So most floating point operations. If you think about it, a petaflop is a huge number, because a gigaflop is a billion flops, and a teraflop is a thousand billion – a trillion. And so a petaflop is a million billion flops,  $10^{15}$  flops. It's a huge amount of floating point power. One of the things that I really looked forward to is we've just recently gotten this, we used to be more around like 100 to 200 and people in my group are really excited to be able to start using it. One thing that everyone asks though is, like two days after we got the petaflop they were like, so what are your results?

It will probably take us a year to two years to get all this stuff through, just to get a paper through peer review and get it published will probably take nine to twelve months. It's something we're – we're extremely about what we can do, but it will probably take about two years before I'll have the results from all this. Which is a little bit anticlimactic, it'd be nice if the next day I'd have the result right there, but that's the way science works.

**Marc Pelletier**

When you get the result you'll be mad at yourself that you didn't use the latest CPUs.

**Dr. Vijay Pande**

Yeah, it's kind of like people who want to, let's say travel to the nearest star. So they just get on the fastest star craft they have, but if you waited ten years it would be even faster. There's that feeling of something like that, but we have been moving extremely aggressively to try to run on the best things we can find. We are, in my lab we use all these different types of techniques and we do experiments that collaborate with lots of other types of people, we are really wedded to solving the problems we're going after. We'll do whatever it takes. That leads to some challenges because programming things like GPUs and things like that are somewhat strange and difficult. But I think as these new technologies come along, I think if they're useful for us we'll be very excited to take advantage of them.

**Marc Pelletier**

Steve do you have any closing remarks or questions?

**Steve Gibson**

I'm just dumbfounded.

**Marc Pelletier**

One thing that I think we're going to try to do at the TWiT Network is possibly get you more CPUs. Get the word out for you.

**Dr. Vijay Pande**

That sounds great.

**Marc Pelletier**

We've got a very fortunate endorsement here of Steve Gibson.

**Dr. Vijay Pande**

Yeah, I'm grateful. That's great.

**Marc Pelletier**

With his endorsement I think that – I feel completely safe actually to throw it onto my machine. One problem with my machine though is I do have a laptop and it gets pretty hot. What are your suggestions to...?

**Dr. Vijay Pande**

[50:24] Yeah, there's different things there. And actually I have a laptop too so I have the same issue. On Windows – one thing there's a laptop setting that it will only run when you're not on battery, for one thing. It's not going to chew up your battery if you're on battery. The second thing

is that it can be set up to use only a fraction of your computing time so you can dial it down in processing power. And that makes a big difference. But in the end I think the model that works best is something like a PC that's just sitting there, or even like a PS3, and I think laptops work pretty well but I think there are – a four-core desktop is probably like our dream in terms of a PC>

**Marc Pelletier**

Well, could we – I'm going to do a little editing here, could you please just say, you have to buy a PS3.

**Dr. Vijay Pande**

I probably prefer not to! One of the things...

**Marc Pelletier**

No, I need this for my wife.

**Dr. Vijay Pande**

Okay. One of the things that is a very tricky thing that we have to deal with is that we have to – I want to make sure that everyone – that my scientific colleagues are completely on board and excited about what we're doing. At the same time we have to be able to translate what we're doing to the general public and explain why it takes two years to get the end results out. In some ways it's a challenge but in some ways I think it's an opportunity because all of these complications allow me to talk to everyday people through a forum or whatever and really explain how science is done. People get to see how science is done. I think when people buy their PS3 or buy their PC and watch this happening it's like different than when you're donating like \$50 to the Red Cross or something like that. You see what your donation is doing and you can decide to vote with your feet whether you think this is interesting or not. It's an interesting new way to do science, it's hard to tell what's good or bad but I think it's been extremely useful for us. There's been no way we could do what we do without it and so I'm very grateful to all the people that have bought PCs and PS3s and helped us out.

**Marc Pelletier**

Well you're definitely a real pioneer and the groundwork that you're laying down I'm hoping to be able to use on my own personal machine just to do problem solving as an individual in ten or fifteen years from now when I have the equivalent of 250,000 CPUs running on my laptop.

**Dr. Vijay Pande**

You know, I expect the future will be that there will be these large networks where it's not that common that maybe one person needs that all the time, but there may be some moment like for an hour or a day you need to run a huge calculation. And I think if we all share our processing power this way, there's like 100 million PCs on the Internet, and you compare that to like Blue Gene, the fastest standard supercomputer that has about 200,000 processors. We're talking about three orders of magnitude more computing power that's just sitting there. So it's going to be really intriguing to see how that's tapped and how the politics of it works out, but there's a lot of computing power sitting on the Earth and I think as long as we can figure out intelligent things to do with it there's really great possibilities.

**Marc Pelletier**

Well that's the key, it's got to be an absolutely valid subset of experiments that aren't just not as thought out as they could be. You gave a talk a couple of months ago at Yale and I was very, very impressed at how much effort you're putting into the fundamental questions like what's happening to the water molecule within the experiment. So I'm really glad to see that you're doing this. I appreciate you coming on and Steve as well, thank you very much for coming onto the discussion.

**Steve Gibson**

My pleasure, guys. This has been fascinating.

**Dr. Vijay Pande**

Yeah, great. Thank you.

**Marc Pelletier**

I would really like to thank Dr. Vijay Pande for being a guest on the show. He's the Director of Folding@Home and Associate Professor of Chemistry and Structural Biology at Stanford University. Also many thanks to Steve Gibson, host of the podcast – or netcast, Security Now on TWiT.tv. You can find out more information about Folding@Home in our shownotes at TWiT.tv/FIB.

Last but not least, thanks to Phil Pelletier and Will Hall for the opening and closing themes. For Futures in Biotech, I'm Marc Pelletier.