ZOLTAN SZALLASI: --like modeling and what modeling can achieve, or why people are doing it at all, or is it doable at all. Should one be interested in this, or this is completely harebrained idea? Reverse engineering, where you really want to reconstruct how the system looks like-- what you are going to use for modeling.

And one more issue is that I really need to talk about-- but of course this is like-- nothing really exciting happened in biological modeling yet. People tried and applied the very same tools, methods, that have been around in modeling for decades. They're facing the very same issues-- very same problems. You have three-- couple ODEs. Well, what's going to happen? Very quickly you're going to run into trouble.

So what's the way out? This is really, really the hope that the robustness-- as we know, I mean, life is robust-- can teach us some lessons that we can exploit if we want to do large scale meaningful modeling. So actually that's like the third point. Now for you guys, what's your background?

AUDIENCE: Biology [INAUDIBLE].

ZOLTAN SZALLASI: OK, and yours?

AUDIENCE: Bio and chem engineering. Biology and--

ZOLTAN SZALLASI: And-- and?

AUDIENCE: And chemical engineering.

ZOLTAN SZALLASI: Chemical-- OK. So good. So I mean for a pure biologist to carry the point home that robustness is something important and to look out for, that this might give the solution, is kind of new. For you guys, probably it's pretty obvious that if you do very large scale modeling, especially for chemical engineering, then these type of things are very important. I mean, you know how it is, right? If you think about all the chaotic systems like-- yeah?

AUDIENCE: [INAUDIBLE]? I mean, I have some background, but it's [INAUDIBLE].

ZOLTAN SZALLASI: Oh, I will, I will. It's just a introductory slide. Is this important or not? For you guys, you will understand why this is important. For biologists, why would I care? So that's kind of the distinction.

OK, so what's the goals of science? Of course the main goal of science-- and one can open a whole discussion-- this is predictive power. And the other goal is understanding [INAUDIBLE] and playing in a playground for social [INAUDIBLE] and so forth. But what we want to do is predictions.

So how did biology evolve or how did it work? Obviously you had a black box. Something happened, right? You set certain conditions, perturbations. You get something that you can describe that's the living organism.

And there are some few readouts. For example, you get a drug and the patient responded or not. Or like if there is no oxygen around, then for most organisms this means death. You do not really know-- and we didn't really know for a long time-- what's inside the box.

We just play with this. We had certain inputs and we had certain outputs. Of course later-- biology during the past several hundred years or 100 years-- we started to learn that there is a very complicated intricate network inside the cell, which is like the unit of life.

And if you want to do modeling of course, you can do black box things and that's what brute force or exhaustive reverse engineering is about. That you do all sorts of inputs and outputs, and you can kind of guess what's inside.

[CROWD CHATTER]

But it's much more efficient if you-- I'm sorry. It really bothers me.

[DOOR CLOSES]

It's much more efficient if you have a pretty good idea of what's inside the box. Because then the modeling description is going to be much more efficient. So what's inside? Whoops. Sorry. Come in.

Now you all have some biological background. So there are genes, RNA, proteins-- proteins have certain activation states. These things are translocated. I mean, there is temporal and spatial information and that's going to set up your entire network.

So what systems biology-- or at least some field of systems biology-- is aiming to do is creating a predictive mathematical logical representation of the living organism. And that leads to the whole issue that what is-- again, for you guys it's obvious. For biologists, not that obvious whether modeling is good or bad. Biologists do not really like the whole idea of in silico modeling. And that's kind of obvious-- I mean, that's understandable.

But for you guys acquainted with the backgrounds, you know very well that all we are doing in science in most cases, is actually modeling. We are creating some sort of a mathematical representation of a system. Think about anything. I mean of course, if you describe the movement of the Earth and the sun, that's two mass points. That's kind of representation, and that's going to give you pretty good predictions. But all we are doing in science is essentially some sort of modeling.

So again, let me just mention this very quickly. This is for biologists, because biologists are really abhorred by the whole idea of in silico modeling, although they started to warm up towards the whole concept recently. But biologists have been doing modeling for the very beginning of biology.

If you had an experimental drug, you're not going to give it to the kid-- or little Johnny. We'd have to see whether it works or not. But you're going to use an experimental model. Now there's a very strong underlying assumption here, which is, this model-- this animal-- is going to react to the drug the very same way as a child will-- which might be correct or incorrect.

But some of you may recall or you might have heard of the thalidomide issue, when they just simply chose the wrong animal model. There is an animal model that would have produced the birth malformations. They didn't choose that.

But the bottom line is that biologists have been using models, and they have no problem when they are using an animal when they are trying to model, for example, the human response. So modeling can be an in vitro and in vivo model. And of course, the alternative is in silico model. OK, so this is like the intro for the biologist that you don't need to hear.

So the first issue is that what is in the black box? And this is essentially, how do we represent biological knowledge in a way that could be used for modeling? And that's sort of reverse engineering of the intracellular regulatory network.

So just to give you some estimate of how complex is this thing that's inside the black box, let me just give you some estimates that I was playing with. For a while, I've been giving a tutorial on systems biology for many years now. So it just tries to give people a feeling of how complex the system is.

So the cautious estimate is that the number of interacting parameters is going to be on the order of a couple of hundred thousands, but less than a million. The way this comes out, you have 10,000 to 20,000 active genes per cell. Different post-translational modifications, different localizations, that will have independent regulatory input for something else. It's going to be, let's say that the three per yeast and three to six that's probably less than 10 per protein per gene on average in humans.

So of course the whole thing may be way off either direction because if you have splice variants for example, lots of antisense transcribed genes, this number might be way higher. On the other hand, if modularity really works well in biology, you don't really need to put every parameter in your model. You can work with appropriate modules, and that's actually one of the key hopes. But that's one of the main directions biologists try to move to these days, because that's still very large.

The point here is the following. If you take these numbers and translate into actual numbers of parameters and networks, for bacteria-- E. coli-- that that number, the complexity of that network, is not much higher than the regulatory control mechanism for a jumbo jet. So actually the idea or the whole concept is coming from John Doyle. I'm just kind of stealing this sentence from him.

John Doyle is at Caltech. He's a very famous mathematician. Actually, he's the one who worked out the theory behind stealth.

Stealth is not supposed to fly. The way that looks like, it's not supposed to fly. The reason that flies is that the control mechanism is so well-designed, that actually it's going to remain in the air. But if you describe like the aerodynamic capabilities, that's not supposed to fly.

So the point is that the control theory-- the control mechanism-- in a very complex engine these days is not much much, much higher, but definitely not many orders of magnitude higher than, for example, for E. coli. So if you want to try very, very, very, very hard and work out all the details for E. coli, then one might try to do some sort of a meaningful dynamic modeling on at least the bacteria. And actually, there are many groups or many institutions-- some companies-- who are trying to do that.

Now that's not an obvious thing. There will be still lots and lots of questions open. Like I'm going to talk about parameter optimization, which is-- of course, you cannot solve it. It's just a very, very, very computationally intractable problem. But the point is that if the way you design an airplane-- which has to be robust-- is very similar to the way an organism will work, then there are certain tricks that you can borrow from control theory that you can apply to bacteria.

So representation of biology-- the simplest level, this is just-- stole the epidermal growth factor receptor pathway from the CAG database. So the simplest level is, you have some sort of directed graph. Now this is obvious for you guys. For biologists, I point out that, this is like the simplest level of biological representation.

So of course you have the nodes that are proteins or genes or post-translational modifications or whatnot. Then there is the edges that are the regulatory interactions. Now you cannot do much with this. This is not a dynamic model yet. You need a more detailed description of actually what's happening between some parameters, and you need to make that dynamic by introducing time.

And in this case, if you have a continuous differential equation, then you need some sort of a description of how things are going to change here. And of course you need certain type of parameters-- genetic parameters. You need all these ingredients. Now this is obvious for you guys. Yeah.

So when you do reverse engineering-- and biologists have been doing reverse engineering for many, many decades essentially. When you try to identify a protein, that's essentially reverse engineering the individual nodes. So that's what biochemistry or genome and other large scale node identifying projects are doing.

Biochemistry, protein interaction screening, two-hybrid co-IP, is trying to determine all the regulatory interactions-- all the edges in your graph. But you can actually start, and there are some efforts that try to do brute force reverse engineering. Why would you do that?

I mean, these approaches are extremely time-consuming. Although there are smart ways-- two-hybrid-- you can speed up things. You can do things in a high throughput manner. But working out every single detail for a genome of 20,000, 30,000 things is going to take a while.

So if you know parts of it and you're quite sure about some subnetworks here, you can start to apply at the same time brute force methods which is why it's simple. You have consecutive time steps, and you try to determine a set of regulatory rules that can produce a gene expression pattern or parameter setting at the latest time point based on the previous time point. And if you do have many, many sufficiently diverse consecutive time points, then you can have a fairly good idea how things are going to regulate each other.

Now the most widely used assumption is that you have some sort of an additive model. In this case, the state of a gene at a later time point is going to be determined somehow by the weighted sum of the state of another set of genes-- some sort of bias factor. And you have some sort of skewing function there. It doesn't really matter what actually you're doing. There are different types of methods.

The number crunching can be done different ways. The bottom line is that-- these are just individual methods have been used-- that you need to determine all the bias factors and weight values. And if you think about it, you can just very simply think about it that it's like, you have many, many, many, many linear equations. And you need as many equations as bias factors and weight factors you have there, in order to do reverse engineering.

Which was actually-- these are the exhaustive solutions, and that was actually done on the bound here. And what [INAUDIBLE], who's a very smart mathematician, actually showed [INAUDIBLE] are unknown parameters in a set of ordinary differential equations, then you will need it on the order of 2r, on the order of r, sufficiently diverse time points-- to do efficient reverse engineering. This kind of [INAUDIBLE], this is the number one would expect. But actually he proved it, so that's good.

The point on this slide is that we should think about reverse engineering based on the assumption how your network works-- or network representation if you think about like Boolean networks and different levels of connectivity. You can make estimates of how many different time points, how many different measurements, you need to do a reverse engineering. Yeah?

AUDIENCE: [INAUDIBLE] equates to the [INAUDIBLE]?

ZOLTAN SZALLASI: Sorry?

AUDIENCE: You said it equates to [INAUDIBLE]?

ZOLTAN SZALLASI: In this case, no, no, no. This is the only one that has continuous differential equations. These are like Boolean representations. This is like an old slide. The point here is just showing that, of course, if you have a fully connected network, you cannot do it. If you have Boolean connectivity K, that means that your average number of input per node is K, then the number is much smaller.

AUDIENCE: Oh, no, I think you mentioned [INAUDIBLE].

ZOLTAN SZALLASI: Oh, no. The linearity was simply-- this is just an additive model. We do not-- no lab has produced any sufficiently good quality data and large enough data set to worry about what the function is. What people are trying to determine at this scale, is actually who might be regulating whom. The equation is simply out of question at the moment. This just gives you a feel of how to-- or the difficulties involved in what you need.

If you do not have enough independent measurements, then one other way of going-- and actually biologists have been using it for a long time-- is actually using perturbations. So if you do individual perturbations in the system in a kind of directed perturbation that these are the part of the network I'm interested in, then you can do this type of measurement. And the principle is very simple.

Imagine that you have three genes-- A regulating B, B is regulating C. And then you create a perturbation matrix. You knock out individual genes. And this is going to show you that based on this topology, how the individual downstream regulated genes are going to behave.

So from this perturbation matrix, you can create an accessibility matrix, which means that who might be regulating whom. And of course a given accessibility matrix can be associated with different regulatory networks. Both of these networks can be described by this accessibility matrix. And then usually what people think is like, let's take the most parsimonious, and they tend to believe it.

Now perturbation measurements are not working very well. So I mean, you can work on subnetworks, and it has been shown to work if you have a subnetwork that has been sufficiently well described. But nothing has been really produced by these methods. Nothing really new has been discovered by itself. But this is kind of the gist of it. This is kind of the underlying principle, how are people trying to approach this.

So if you do reverse engineering with these type of methods, given good quality data and enough computational number-crunching power, most probably we are going to have some sort of a regulatory topology of a network. It's almost given. And there are lots of preps who are producing knockouts for yeast. And they started to produce some meaningful data sets.

But if you do dynamic modeling, you would at least hope that you have some of kinetic parameters in that model that describe or approximate reality fairly well. So another aspect of reverse engineering is when you have some idea about the kinetic parameters, but you're trying to refine these measurements. And that's what [INAUDIBLE] lab did. That's what they did actually.

They essentially tagged by green fluorescent protein, every single protein in E. coli. So green for us, protein is like a tag and it's on every single protein. And they just do time series measurements on every single protein. It can be done in a high throughput fashion because in every single well, they know which protein is tagged. And if you do this, then you can actually-- from the measurements-- you can refine the actual kinetic parameters pretty well.

The reason you need this is quite obvious. When biologists or biochemists determine genetic parameters, that's usually done in a free solution. You grind up the bacteria, do something, and that can be quite different from the actual genetic parameter present in the cell.

So for one thing-- think about it-- in a free solution, you approximate things with very large number of molecules. Whereas in cells, sometimes you have 50 to 100 proteins of the same kind. So you might have a very different-- and it's not a free solution, it's a much more dense solution. So the efficient kinetic parameter can be very different. So reverse engineering can give you much better estimate so we can refine the estimates on the actual category.

AUDIENCE: [INAUDIBLE] localized?

ZOLTAN SZALLASI: As well, yes. I mean the point is that if you do-- you want to start with the best parameter set. And in this case, it's-- yes, localization. Of course it's important. I think about it at membrane proteins. I mean things that will happen in the liquid phase is going to be very different from the free water solution. So [INAUDIBLE].

So let's assume that one went through all the difficulties and have a fairly good description of the topology and the regulatory interactions, and then you have good quality kinetic parameters. So should one try-- yeah.

AUDIENCE: Just have a question.

ZOLTAN SZALLASI: Yeah.

AUDIENCE: Two things that stand out. [INAUDIBLE] the schematics slide you had before. That one, yeah.

ZOLTAN SZALLASI: This one, yeah.

AUDIENCE: There are two things about this that stand out. Assuming [INAUDIBLE] I'm sure that some of the GFP tagged proteins actually were lethal [INAUDIBLE] right?

ZOLTAN SZALLASI: Some, yeah.

AUDIENCE: Because they included function or [INAUDIBLE] critical [INAUDIBLE]. And other proteins, obviously because many of them [INAUDIBLE] large protein, will probably reduce-- or have reduced activity or whatever. And both of those actually go towards-- so if we want to understand parameters or the kinetic parameters for gene regulation or whatever, I'm going to seriously bias the information that you get out of the [INAUDIBLE] much the same way that having the solution [INAUDIBLE] parameters [INAUDIBLE]. How do they address that?

ZOLTAN SZALLASI: They didn't.

AUDIENCE: They didn't address that? Oh.

ZOLTAN SZALLASI: At this stage, they were happy to see that they can produce something in a high throughput fashion. Your points are absolutely valid and correct and important. What they do is, they do another expression at an individual protein level. Where they say that if I rewire-- and they took a sub-- this was not done in the actual publication for the entire network. They just took that part of the network that describes how the flagella are assembled.

And if you use these refined more these kinetic parameters-- that actually were significantly different from some of it that was measured before-- then you have a much more accurate description of the dynamics-- how the flagella are assembled. So they are kind of proving at the level of the overall functional module.

AUDIENCE: Do you see a possibility in that case-- assuming that two proteins, particularly the network [INAUDIBLE], are important for genetic function. Assuming that those are tweaked by the GFP in different ways so that maybe the implication of looking at the GFP tag on one protein [INAUDIBLE].

ZOLTAN SZALLASI: So the point is-- OK, I see.

AUDIENCE: Can you [INAUDIBLE] de-convolute the amount of error?

ZOLTAN SZALLASI: Yeah, yeah. So what they were actually doing was, the GFP is not really trying to measure the protein. It's measuring the activity of the translation. So when you have a GFP tagged protein for, let's say protein A, then what they are really interested in is how fast that protein is activated by the non-tagged transcription factors.

And what they were interested in, the parameters they wanted to refine, is the activation of that gene. So the GFP protein is simply a--

AUDIENCE: So it says nothing about--

ZOLTAN SZALLASI: --a marker.

AUDIENCE: It says nothing about the actual--

ZOLTAN SZALLASI: If you have a very quick, very immediate feedback loop, and it interferes with that, yeah, that's a problem. But if you do not have that, you have much larger feedback for many, many, many different proteins. All they wanted to see, how that actual protein is activated. And the activators of that transcription unit is unaffected. There is no GFP tag on those proteins.

AUDIENCE: Well I think my question-- and I'll just modify it [INAUDIBLE] back and forth. Is there a way to-- so let's say there are two competing regulatory proteins for transcription.

ZOLTAN SZALLASI: Yes.

AUDIENCE: [INAUDIBLE]

ZOLTAN SZALLASI: Of a certain protein that is GFP labeled, right?

AUDIENCE: And you get differing regulation of-- I'm sorry. I screwed that up. Two proteins that are GFP tagged that you know are [INAUDIBLE] and you get different regulation of those genes--

ZOLTAN SZALLASI: Yes.

AUDIENCE: --from this. Do you assume that there's a way to actually de-convolute what the actual right-- like, so there's an error associated with the tagging [INAUDIBLE].

ZOLTAN SZALLASI: Yes.

AUDIENCE: And you're going to get two different expression profiles for both of those proteins in each experiment even though only one is tagged.

ZOLTAN SZALLASI: Well, the only way you would do it is actually you take two different tags. Probably that's the best way to go. From one set of measurements, you can attempt those things. But--

AUDIENCE: Would that give-- is it possible then to get information out about the degree of perturbation of the actual system from using that double tagged system? Do you understand--

ZOLTAN SZALLASI: Yes.

AUDIENCE: [INAUDIBLE]

ZOLTAN SZALLASI: I'm not quite sure whether it's worth the effort. What they are doing is-- as I said--

AUDIENCE: Because right now they don't have an estimate on how accurate their--

ZOLTAN SZALLASI: Yes.

AUDIENCE: --parameters are, right?

ZOLTAN SZALLASI: Absolutely. So what they do is they try to estimate whether their effort is worth or not based on the outcome of the entire modeling effort, and with issues like whether the description is a robust network or not. Every single measurement will give you a somewhat biased parameter. So if you measure a single parameter in different settings, they will always be different. What is the error there?

The point is that with this way, you can refine parameters to some extent. And that was a significant improvement. Sometimes it is like an order of two decimal difference from what they thought it was before. Based on that, they have a more accurate description of the system.

Now within certain error range, very different parameters would give you the very same result. This is what you expect from robustness. We are getting the error. You just missed the first couple of slides when I said-- if you keep playing with parameters, you can very easily overfit your entire system.

So I'm not quite sure whether that's-- yes, for individual process, you can do it. And obviously, if you do very different taggings and very different types of measurements for the same proteins, then in some way you can de-convolute-- or at least you can catch very obvious contradictions. And you say, well for this parameter, I have no idea what's going on. Or for this parameter, I need to measure it individually.

Because if you develop an antibody or some other measurement technique, then probably you can measure it much more accurately. So probably that would be the best way. For most proteins, probably you won't have that many problems.

AUDIENCE: I see. So within the network, the parameters are defined in a way that it doesn't matter to the individual parameter [INAUDIBLE]--

ZOLTAN SZALLASI: Exactly.

AUDIENCE: --how they combine [INAUDIBLE].

ZOLTAN SZALLASI: Yes.

AUDIENCE: And making sure that that [INAUDIBLE].

ZOLTAN SZALLASI: Yes, yes, yes. And that's kind of the whole gist of large scale modeling. How much effort do you need? How much information do you put in to get some sort of a reasonably accurate prediction? OK? So let's move on to that.

So what do you need to do for modeling? The obvious steps, you have to ask-- this is like step zero. You are going to ask interesting questions. You need to collect good quality data. You need to create an appropriate mathematical logical representation of the system.

That's what we have talked so far about. We need to run simulation and test how well the simulation is fitting the data. Test for robustness or for other network properties. You update the model based on how well your model description is describing the data and then use it. So if you--

AUDIENCE: Quick question. Can you define robustness?

ZOLTAN SZALLASI: We are getting there.

AUDIENCE: OK.

ZOLTAN SZALLASI: If-- well, it's relative intersensitivity to individual parameter values for the entire network. That's kind of the definition. So you have a network. Now you still have to ask another question. How does this network really behave?

So essentially your question is, what is this equation here? And in most cases, what people think about is-- and this is the only models that have been really employed in some sort of a realistic way to describe biological systems-- is just using ordinary differential equations. Of course, there are other ways of doing. You can use differential equations for discrete steps.

The point here is these are deterministic systems. So the question is, is biology a deterministic system? Or is it true that a deterministic description of biological network is going to give you something meaningful?

Deterministic system-- you all know what it is. There is only one outcome from a previous parameter setting. But biology, as we know, is most probably not that highly deterministic. Actually, we know for a fact that individual biology calculations are always stochastic. It's another question how the biological network behaves.

But for individual reactions, you know that you have very few molecules-- less than 400 transcription factors in a nucleus. Fairly far from a quick solution. Reaction can be often slow. So there's all the hallmarks of stochastic reactions that chemical engineers are all too familiar with.

So in that case, you have to do different type of modeling. You want to use stochastic differential equations where you have kind of a stochastic error in differentiable equations-- which is kind of cheating. The real thing is what a long time ago, Gillespie introduced-- like 30 years ago. They actually do a stochastic modeling of individual chemical reactions.

This is the thing that is actually closest to reality. Of course, the problem is that computation is so difficult. It's so intensive. And there is-- unless there's a good reason that you need to do this type of modeling, then this is not really an economical approach.

Now there are other tricks people are trying to do. Like John Doyle is trying at Caltech with [INAUDIBLE] who came back from retirement, to do something called a tall leap. So this is like a really detailed description of how things are hitting each other and how chemical reactions develop.

Now can you guess where the reaction is going to be in, let's say, 10 steps? You can tell what's going to happen in one step if you do a stochastic modeling. Can you guess what happens in 10 steps? So they're trying to cheat and they are trying to see whether-- if you think about like, ODE is going to describe very simply what's going to happen.

The stochastic equation is going to jump around this in some way, and you will have alternative outcomes of the same reaction. With the tall leap, what they are trying to see-- they just jump from here to here. And there has to be certain-- what is the probability that the system is going to be in this state, in let's say 10 time units, or something like that? So this is, again, a wide open question.

The big question is, is this really an important thing, that we think about biology and model it in a stochastic fashion? Now let me just show you an example from real experiments that shows that, yes, it's important. There was a very nice experiment in which they created a plasmid of two proteins. They were tagged with two different GFPs.

And this was a very carefully controlled experiment, when two proteins were translated. On the same plasmid in every single bacteria, and everything was driven gene expression by the very, very same promoters. And everything was in principle, identical in every single cell.

So what you would expect in the relative expression of these proteins should be the same in every single cell. So we should show the same kind of mixture of red and green. So that was the expectation.

And instead of this, what you see is there are very red, very green, and yellow cells, showing that individual cells will choose very different pathway-- or path. So there will be cells that will have much more of one of the proteins than the other ones, although everything was identical-- as much as we can control experimentally.

We know stochastics is important. We know that probably differentiation is driven by specificity. You have a parent cell, and after division you have two cells that are going to go two complete different paths. One remains a stem cell, the other guy goes the opposite direction.

And that might be just simply driven how certain proteins are distributed randomly between the two daughter cells during the cell division. It is known actually that for many differentiation steps, probably the decision whether a cell goes one way or other is going to be a stochastic process. Now whether this is important for modeling the cell cycle, cell growth, and other features, we do not know. Yes?

AUDIENCE: Might be a silly question. But how do you define stochastic? Is it just randomness or--

ZOLTAN SZALLASI: Oh, it's a very good question. So-- well in a way-- OK. How would you define stochasticity? Again, this is the representation of scientific knowledge. So we describe temperature with a single number, although you know very well that--

AUDIENCE: [INAUDIBLE]

ZOLTAN SZALLASI: Yeah, you have individual kinetic energies of all molecules. So if you have many molecules, it doesn't really matter. You give temperature. If you have 10 molecules, the whole issue is very different.

So stochasticity is that you go down one more level of detail description of the individual reactions, that will happen with a certain probability. So you can call it randomness, but the point is that most of these things as we understand, will occur with a certain probability. So probabilistic descriptions of individual molecular reactions are much more accurate and much, much closer to reality than continuous-- continuous differential equations are nice, and they were invented by Newton-Leibniz to do useful things. But I don't think that even they thought that this was an accurate-- well, probably they thought-- but for a long time, we know that this is not a very accurate description of-- certainly not since quantum chemistry and physics is around.

So once you have a description and you decide which way to go-- so this slide just showed you that we have evidence that stochasticity is present in biology. We have lots of evidence. I just showed like one experiment.

Whether it's worth doing for modeling and not going with all these, that's an open question. Probably it's going to determine which-- not, what's your question. That's why it's very important that you formulate your question appropriately.

Now, once you decided-- so you have the network, topology, kinetic parameters, equations, everything. Then you actually start to run simulations and see what happens. So let me show this example that-- actually, this can describe certain biological phenomena. And you can actually formulate useful hypotheses based on this.

This was something called a nucleocytoplasmic transport. This is an active reaction itself. Proteins get in the nucleus and they can't come out. And that's actually driven by this regulatory-- or this dynamic network-- involving a group of proteins.

Essentially this protein is Rand protein, which is a G protein reaction. So it uses GTP as an energy. It's an active reaction. This is what is actually moving things in and out from the cytoplasm.

When people started to think about modeling, they have a strong assumption-- everybody does-- that if you have a network that's not very stochastic, that's sufficiently isolated from the rest of the network, and we understand it well enough, then probably that network can be described pretty well with a mathematical model. And you can probably make pretty good predictions based on that. So these people actually who did this experiment, they were experts in the nucleocytoplasmic transport system. That's why they knew that probably there were not that many blank spots in this network.

So what they did-- they took all the kinetic parameters. They already measured it several times. This is not such a large network. They created that and they started to do measurements. They actually measured pretty well-- pretty accurately-- how quickly proteins enter the nucleus.

This is the nucleus. It's empty and then it's getting into the nucleus. So they can measure it quite accurately. So the first question you have, does the model produce time series of charts that feed the data? And they did all these experiments.

And these are the measurements and these are the predictions. And yes, it seemed that the model produced fairly good descriptions-- fairly accurate descriptions. So that is good news.

Now the next question you ask is, OK. I describe a certain experimental setting, but is the model robust? And how sensitive is the parameters? That is, if you have the right ratio of parameters and measurements-- I'm sure you understand this-- you can always overfit the system. You can always find a set of parameters that will give you the right result.

Now is this anything useful? Probably not. Because that's what you'd like to predict. What's going to happen if you take a very different parameter setting? Or you put in a new drug or combinatorial treatment? So what these people do-- and in this case you need to do-- is check for sensitivity.

So robustness is, you check the outcome of the behavior of the system-- whatever you measure. For example, in this case, how quick was the dynamic that things get into the nucleus. And you start to play around with every single parameter and ask the question that, in what parameter range do you still have the same result-- let's say with a 10% error?

So these are all the parameter settings-- these are individual perturbations. These are perturbations on individual parameters at the same time that still give you the same result with a 10% error. And so that's the next step. And if you are-- see that, well, it looks fairly good. Then you start to trust your description is fairly robust. That's cool.

Now the next thing you need to do is actually, can the model produce something useful-- for example, a testable hypothesis? And that's what these guys did. So they said, well, the model would predict that if you knock down this RCC-1 protein that helps RAM to perform its function, then it has to behave in a certain way.

And that's what they did. They used the temperature-sensitive mutant to knock that protein out. And this is the measured dynamic or kinetics of the proteins getting into the nucleus after knockdown. And this is the simulated.

So in this case, actually this describes the system fairly well after you introduce a very profound perturbation-- you knock down a protein. So these are the key steps you want to do in the case of the following. And if you have all these things, then you can assume-- or you have actually-- you can have fairly high confidence that your model is describing at least the subnetwork fairly well.

AUDIENCE: [INAUDIBLE] the prediction model [INAUDIBLE]?

ZOLTAN SZALLASI: Well this is the part of-- I'm losing you. Thank you. This is the simulation, which you have on the bottom.

AUDIENCE: OK, I guess-- I guess--

ZOLTAN SZALLASI: So you have two simulation states, right? That RCC-1 protein is present or absent. So that's what you have there. And you actually measure these things in the presence or absence of RCC-1.

AUDIENCE: So are you able to predict like-- without actually speaking to examiners-- what is also going to be before you actually do the experiment or [INAUDIBLE]? You kept saying, like-- can you actually predict that curve without re-tweaking the parameters?

ZOLTAN SZALLASI: This is not re-tweaking the parameters.

AUDIENCE: [INAUDIBLE]

ZOLTAN SZALLASI: You build the model, you check for robustness. And then you do not touch the model anymore. Right? Then you start to do in silico experiments, saying, with RCC-1, how does the system behave? And then actually you measure whether the prediction was correct.

And after you are done with five of these experiments, you say, I have a fairly good model. So I can start to play with this thing and ask questions. Introduce random changes-- what's the model that's going to speed up?

And then the in silico model is going to give you certain answers and then you can test those. But the point is-- and that's going to be like the last slide in this lecture-- that you're trying to speed up the discovery process here. Of course, every single time you need to do the experiment.

But you can go through a much larger number of possible experiments in silico than in vitro or in vivo. So that's kind of-- I've introduced this concept of measuring the robustness. The robustness is the relative insensitivity of a network to parameters. And this will answer the question that this is a relatively small network.

This was still a couple of panels of parameters. But what people are trying to do-- and there are companies that try to do this. There is some company that my friends started five years ago. It's called G Network Sciences. They are physicists-- solid state physicists.

And they say, well, we can model very large things. And these guys actually build networks on the order of 1,000 or 2,000 interacting things-- parameters, proteins. They use ordinary differential equations and they will do forward modeling-- forward simulations.

But if you think about this whole robustness issue or network fitting, parameter fitting, well, it's going to be a problem. These are the time series data predicted by the model. And you can actually do measurements on these things, right? So you can start to see how well the simulations fit the experimental data.

Now if you think about it, you have very many parameters here. You have thousands of parameters here. And they can change in all sorts of combinations. So when you try to fit your model to the experimental data, if you loosen here, you have to tighten there.

So this is a very, very complicated-- computer scientists will understand right away. This is an intractable condition. Well, actually, if you think about, these parameters have only two values that it's going to be, and you have n parameters. Then what you have to go through-- exhaustive [INAUDIBLE].

So it's not doable. So they use also the trace genetic algorithms or whatnot. But nobody has any idea right now about the landscape of this parameter space. Is this like-- really have certain optimal states? Or is it kind of local optima and we're just playing around with these things?

And you'd like to know these things before you trust your model. So you can play and you can do relatively exhaustive searches for small networks. But then you start to enter the realm of realistic size networks of 1,000 parameters. You have no idea what you are-- where you are.

So that's what answers the whole issue of robustness. If we understand what robustness means in this network, maybe you do not really need to worry about the actual parameter values that much. Because within certain ranges, the system has to behave the very same way. Yeah, that we all know.

And there is some evidence for this. So there was a very nice paper from Gary O'Dell when they took all the knowledge we have about the [INAUDIBLE] priority network of [INAUDIBLE]. So this is all the knowledge we know. These are all the proteins that play a role in it and all the regulatory interactions.

And they describe these things and they put all this into MATLAB and they start to play with this. And what they found-- they found two striking things. One is that if you have this network, parameter values essentially do not matter. In the range of two to three orders of magnitude, you could change any parameter setting. You still have the same outcome.

Also, this network behaved in a robust way only if you introduced two regulatory interactions-- these two guys. And when they looked harder, they actually found evidence that those are there. So the important message of this work was that maybe it's mainly the topology that's going to determine how your network behaves.

People tried-- like Andrew Murray here at Harvard-- tried to de-convolute the network. One of the questions is that, if you have a network that's robust, is this working the very same way as we understand in control theory? So you have models, redundancy, feedback loops. Or there is some of an other interesting topology at play that evolution developed or invented.

So they try to do this, but they still-- the sense of this study was that the parameter values can change a lot and you still have the very same outcome. So if this is true for many biological networks, large networks-- and this is actually the way it's always designed, although it will take a lot of work and a lot of independent confirmation, then you do not really need to worry that much about individual parameter fitting once you understand the typology and you have some sort of approximate field about the individual parameters. So that's good news. And also that they could make predictive-- they could make predictions.

So robustness one of the key properties of a link organism. And-- oh, this is the kind of semi-official definition. And OK, so how does this work? As we understand, robustness is derived in control theory. Jumbo jet, three things-- we have feedback loops, we have redundancy, and you have modular design.

If you apply these things in the right combination, the right way, involving lots and lots of complicated math, then you have probably a very robust system. We do not know whether we have some other design at work in biology. In principle, that's possible. There are lots of issues raised like-- it was raised that if you have a parallel distribution of regulatory interactions, if you think about this graph description-- like small neural nets. You might have heard a lot about these things.

Now, these things do not translate in any way into dynamic pressure. So nobody knows whether this makes any sense or not. But I'm just throwing something at you that, well maybe, yes. Maybe if you have some of parallel distribution or regulatory interactions, then maybe that's going to be the key-- or some sort of a design.

Nobody really understands why the World Wide Web works. It's not supposed to work. It works but nobody understands why. There are some ideas, but nobody can describe really the dynamics. And it's not supposed to work the way it is described.

Whether that could be de-convoluted to describe the work of World Wide Web, why it works-- into these categories, into these terms, or there is some of a self-organizing evolutionary principle at work there, nobody knows. Lots of people like [INAUDIBLE] and some other people who are betting on these think that there is some kind of self-organizing evolution principle at work here.

Nobody understands in real terms, what that means. But as you understand, there is kind of a philosophical or like a religious issue here, that there is control theory as an engineer puts together a robust system, and there is evolution. Is the two things essentially the same or there is difference?

So how can you use robustness? You can exploit this for biology. There very nice people from [INAUDIBLE] who did-- actually they took Drosophila again. It's kind of interesting.

Bone morphogenic proteins that will tell your cell which way to develop or not to develop are distributed spatially in a well-determined way. So for example, this is a protein. This is the whole Drosophila egg.

Some proteins are expressed only in the midline. So these proteins tell you what's the top, what's front, back, ventral, dorsal, midline, lateral, in an animal. So if you think about it-- if you look at this thing, this is odd. You'd expect things spreading out.

So how can this thing happen? This is like a real distribution. This is the central lateral distribution. So how can this thing happen? So these are involving very, very few proteins-- three or four-- and they describe this in kind of a one dimensional way. So they kind of unfold in the animal.

So this is the distribution as a central and lateral. And what they did-- well, this is what we know about this system. We do not know the parameters-- roughly no. But let's generate a very large number of random networks.

They generate 60,000 random networks. You can do it, they're relatively small network. A couple of differential equations-- partial differentiable equations. And then ask a question-- what are the common characteristics of all the robust networks relative to those that are not robust?

And they came up with a handful of observations. [INAUDIBLE], which is a protein, does not diffuse. [INAUDIBLE], which is another protein, is not chopped. This thing has to break down.

And only the observed distribution is robust. There was no distribution that was like flat and robust. And we are talking about a very large number-- tens of thousands of randomly generated networks that retained the overall wiring-- the topology-- but the parameter values were randomly changed.

So they came up with these two hypotheses. And it showed that it's only the-- robust networks must have these features. And actually they could experimentally verify this, which is very, very nice. So the idea again is that you can play around with these networks, ask what's the common feature of robust networks, and that will help you-- that will teach you something about the actual biology present.

So this is kind of modeling based on all the differential equations real networks. Now you all know that until we understand-- you understand that until we really understand how the parameter fitting is going to work, and how the system works, we cannot really do very large scale modeling. But there were other people who took alternative approaches. And they said, well, maybe we can understand something about the overall topology of networks if we take very rough models-- like Boolean networks.

Genes are on and off and they regulate each other. And [INAUDIBLE] and [INAUDIBLE] Glass, they started in the '60s and they played around with this. And they wrote interesting books and they became very popular. And they were not really part of real experimental science, but nevertheless they produced some very interesting suggestions that kind of keep coming back.

So let me mention a couple of things here briefly. Because as I said, these things-- these Boolean networks-- never had anything to do with real description of biology. They're very, very far from that. And that was the '60s, '70s-- they made huge errors in their analysis. They very much undersampled the gene expression space and so forth.

But they introduced a couple of concepts which were not exactly new, but they showed that it's an important thing. That if you have a Boolean network, things turn on or off each other, then it's a deterministic network. Then depending on what's the average number of input per gene or per parameter, the system can display chaotic or organized behavior. Chaotic behavior means that the system just keeps wandering about in the whole gene expression space. Organized behavior means that you have limit cycles or attractors-- limit cycles in continuous differential equations, and in Boolean networks, you have attractors.

Which means that it doesn't really matter, what's the initial parameter setting, in terms of individual genes on or off. The system tends to fall towards attractors. This is not new. Because people knew that this thing exists. But the concept that these attractors can behave or can display very robust properties is still very important.

Now is this important or not? And then at that time, [INAUDIBLE]-- he's a very interesting guy. I mean just-- you know, he didn't do any experiments. In the '70s he came up with a set of papers. And it's just intellectually very entertaining, what he did.

He actually suggested that differentiation states are essentially different attractors. And during malignant transformation, development of cancer cells go from a normal attractor towards a malignant-- a new attractor. And then they will stay there whatever you do. Or you have to change a lot of things to drive it back to normal.

Now there might be some evidence this is actually true. You heard about gene expression measurements. And what [INAUDIBLE] group did, they took cancer-- breast cancer samples from patients before treatment, after treatment, and from the metastasis of the same tumor in the same patient. And they measured gene expression pairs.

And then-- well if you look at cancer-- you might have heard about it or you might be aware of this-- [INAUDIBLE] completely messed up genome-- chromosomes recombine in a completely random fashion. And you would expect that well, yeah, the same tumor is going to keep changing a lot. Well based on this, it seems that this is not the case.

A tumor in a given patient is always going to be very, very similar to the tumor in the same patient given a different time point or during metastasis whatever you do. It's kind of the same--

AUDIENCE: In terms of gene expression.

ZOLTAN SZALLASI: In terms of gene expression, I'm sorry-- gene expression pairs. And they are going to be very different from somebody else. So it seems that once a tumor develops, it finds a new stable state, it's going to stay there. It's not going to wander about.

And now there is lots of evidence-- well, different labs work with different cell lines. And certain cell lines that have been established for a long time, and unless you introduce very huge genetic changes, they are pretty similar to each other. They are always much more similar to each other than, say, anything else.

So it seems that the genetic network-- and this kind of makes sense-- finds a new robust stable state. And then it tends to stay there because if you do those individual small perturbations, the system can't fix itself. So this pretty much is pretty similar to what they suggested. You have strong attractors.

Now, is this is useful? Not this representation-- we do not know. But you can ask questions like, how many perturbations do you need to introduce to drive the system from one attractor to another one? And if it shows that for a large network, you cannot do it with less than 100 perturbations, well that would suggest a very different therapeutic strategy than if you can do it with one. Or which things you need to change-- things that are highly connected or intermediate connected? So you can actually-- these representations might teach you something about the robustness of the network. That's why I wanted to mention it.

Also I want to warn you about a couple things like, when we talk about modeling, we always talk about a single cell. What's happening in that cell and which way it's going to develop? And we shouldn't forget that whatever measurements you do, you always work with population average data.

This kind of leads back to this stochasticity issue. And let me just show you this very nice little experiment by James Ferrell, when they measured the Jun Kinase activity of individual xenopus oocyte. And actually he had to do this whole experiment himself, although he's like a well-established full professor, because there was no student in his lab who's willing to do the following experiment.

So what they did is that they measured Jun Kinase activity on the progesterone treatment. This is the dose response curve in a population of xenopus oocytes. And what happened is that-- it shows this very nice dose response curve and the coefficient 1. Now the problem was that this was very far from what he thought was the case.

And so he asked some of the students to actually measure the Jun Kinase activity on individual oocytes. That's a lot of work. Instead of running a single [INAUDIBLE], you have to run 100 different [INAUDIBLE] for individual xenopus oocytes.

So he started to take individual oocytes and experiment. And actually he had a very strong assumption what he was expecting. That's why he invested all the time. But what he found is that for an individual oocyte, they are either always inactivated or activated, with a very sharp dose response curve-- with a [INAUDIBLE] coefficient of close to 100. He had a strong positive feedback loop there.

So the point is that what you see here is like an average. Because each oocyte is activated at a slightly different concentration. But what you have in a single system is a strong positive feedback loop. So when you do and describe a system and when you try to do reverse engineering, if you go with population average data, your reverse engineering is going to give you a [INAUDIBLE] coefficient of 1 and some sort of a description that this is what reality is.

And this is what you should do for your modeling if you want to describe the system correctly. Right? So this is kind of another warning how complicated biology is, that most measurements we are doing right now is grinding down a lot of cells. And you're doing some sort of a population average measurement.

AUDIENCE: [INAUDIBLE]

ZOLTAN SZALLASI: Yep.

AUDIENCE: Could it be hypothesized that [INAUDIBLE]?

ZOLTAN SZALLASI: Oh, yeah. Actually he de-convoluted and then published another paper. He showed exactly why you have that. And there is a feedback parameter that you can tweak and you can actually show that you can completely change the hysteresis curve.

AUDIENCE: Again, that's very similar to what [INAUDIBLE] stable [INAUDIBLE].

ZOLTAN SZALLASI: Yes.

AUDIENCE: And like high induction [INAUDIBLE].

ZOLTAN SZALLASI: Absolutely.

AUDIENCE: And low induction [INAUDIBLE].

ZOLTAN SZALLASI: Absolutely. The difference there is that this is a real network. And what these guys are doing, they put it in. Yeah, and that's what Jim Collins is doing at BU. These guys-- James Ferrell did it for an actually existing network and [INAUDIBLE] tweaking that individual feedback parameter. We have seen that.

So that was kind of modularity, and this is the positive feedback loop here. It's kind of de-convoluted here. Is it important to understand or not? Well, this is something very simple, but this might be very useful for drug development.

Because the concept is simple. You do not want to design a drug that's going to interfere with something inside a feedback loop. That's why it's a feedback loop. So if you keep an input there, that's kind of going down the drain.

So if you understand this network and you understand you have feedback loops, you want-- that would give you very strong indications where to design drugs-- where to interfere with the system, because it makes sense. In the other case, when you are inside a feedback loop, it doesn't make any sense to put in too much effort.

OK, the last thing I want to mention is constraint-based models. So these things are very, very, very complicated dynamic networks, and there are lots of problems involved. And is it a kind of a simple way or smart way to make a shortcut, and learning something about biology with much less effort? And those are the constraint-based models.

This essentially-- there's nothing-- they are like [INAUDIBLE] analysis and these things have been around for ages for chemical engineers. And they are useful for certain things. They are essentially the beefed up version of Kirchhoff's Law, stating that if you have a metabolic network, for example, then whatever goes into a node, it has to come out. Otherwise the system is going to explode or that metabolite is going to disappear.

These are not dynamic things. All you are describing is that at every single metabolite node, the net flux has to be zero. And then you have a set of linear equations. And of course, usually you have much more parameters than linear equations. At least this can start a question.

And that's what [INAUDIBLE] group has been doing for a while and they did very interesting things. For example, they showed that if this is like the flux cone-- which is, I'm sure that-- those of you in chemical engineering are familiar with-- this gives you the-- underneath the flux cone, you have all the possible or allowed solutions. I mean you can-- since you have more parameters than equations, there are many different combinations of the parameters that give you the solution.

So the animal can live always underneath this-- anywhere underneath this flux [INAUDIBLE]. The interesting thing they hypothesized is that evolution optimizes. In other words, the animal is going to live on the edge. Produce maximum energy with a certain amount of oxygen input.

And actually they measured it. They took the metabolic network of E. coli-- that's fairly well described. You have essentially the entire E. coli network around and you can download it and you can play with this. And that's pretty well known.

And that's simple, just linear equations. You solve it and then you have an idea what are the allowed parameter combinations. These are very simple combinations. So this is like just stoichiometric numbers, how many things go in and come out.

And then you simply measure that if you measure the oxygen input or the energy source for growth, how fast is the mass produced? How fast will the animal grows? They show that it is always living on the edge-- so it's optimizing.

And then you are switching from one energy source to another one. It's going to find the other edge. And if you do the experiment 10 different times, the animal is going to take very different routes to get there. If you keep taking time series points during the development, it's finding a new edge. But it's always arriving to the new edge and it's going to find that new stable optimal growth condition.

So this is actually very interesting because this is an entirely non-dynamic description. It's a very simplified description. But it teaches you something very important about the biology of E. coli, and it can be very useful. For example, for chemical engineers who are working in fermentation, this is very important. You can really optimize the production of certain type of proteins if you understand the metabolic network underneath.

So one more thing, it's just the end. Again, this is not entirely new. This is a very old paper-- it's not actually very old-- 1994. Then people describe like the coagulation pathway that has been known for a long time. And you can actually describe pretty well.

So of course, modeling has been around in biology as well. It's just, we didn't have that much measurement. And there was certainly no public or private funding to do all the experiments. And so, this is what we are having.

So traditionally, biology try to predict things and then do experiments. And people did everything in their head and based on their intuition. And what we are trying to do here is speed up or make it more efficient-- this process you have to experiment. You have the prediction, you do modeling in between.

That's going-- in silico, you have lots of alternative solutions. You throw out all the obviously wrong solutions, and you just experiment to test the ones that are supported by your model. And that can speed up designing of new therapeutic approaches. For example, combinatorial treatment, which is very difficult if you think about it-- a combinatorial explosion of all potential treatments. Any questions?

AUDIENCE: Do you see any similarities with-- I just like [INAUDIBLE] that came out of the paper [INAUDIBLE] in terms of the physics of kinetic gas theory and just the dynamic [INAUDIBLE]. Some of the models that people are building. Or is that [INAUDIBLE]?

ZOLTAN SZALLASI: I haven't seen any experimental verification one way or another. Most of these approaches actually were done by physicists. Like people describe large Boolean networks based on stained glasses. And then they said, well, it's going to behave something like this.

If you can formulate experimental testable hypothesis based on that, and then you do the experiments and you see this is the way it works or doesn't work, it's great. The problem is that there are some theoretical approaches that physicists work out that just cannot be translated into anything biologically testable. That's my only problem. And those are not that-- of course, those are not that obvious how to do.

AUDIENCE: [INAUDIBLE]

ZOLTAN SZALLASI: Possible to use what?

AUDIENCE: Gene expression data [INAUDIBLE].

ZOLTAN SZALLASI: What gene expression data?

AUDIENCE: [INAUDIBLE]

ZOLTAN SZALLASI: Yeah, that's a good question obviously. And that's-- I mean if-- if you were here when I talked about microarray measurements or noise, those are pretty noisy measurements. So you can do it. If you want to do parameter fitting or microarray, I wouldn't do it.

Microarray is compressing and if you can guess pretty well what's going up or down right now, that's already a nice achievement. One can play with this and-- well, microarray is improving and we are certainly taking our share. Showing, for example, that based on some chips, like half of the [INAUDIBLE] probes are wrong. They are just not what they are supposed to be.

And if you throw those out, you have very good [INAUDIBLE] measurements. So one can improve this type of measurements. But right now if you just take any microarray measurement from literature or yourself, and you want to do a parameter fitting on that, that's essentially noise propagation.

If your question is, if I do a modeling-- I have a fairly good description of the model-- very large network. And my microarray measurement can give you, with a 95% confidence, who goes up or down, and my prediction is just going to be from the model-- for many proteins, who goes up or down. And you match it in a probabilistic fashion to the outcome of the microarray.

In that case, yes. And that is being done and that's a useful approach because you have a very large network. And you're going to say that I want to predict how most or many of the parameters are going to change. And if I can then tell you who goes up or down, even without giving you very good quality numbers, that's going to be something very useful.

So that's something actually-- I must have had an extra slide somewhere, which is-- yeah, actually I do. That was it-- huh. That's it. Kind of predicted what I wanted to.

So there is a low throughput data where you have very good quality models. And there are other ways of going that high throughput data that are not that high quality and you can do probabilistic modeling. You can create-- you have one network, and you create lots of alternative networks with individual parameter changes. And then you just keep pruning that tree-- which model is giving you the best fit to that low quality microarray measurement. So in that case, yes, you can use it. For parameter fitting, I wouldn't do it.