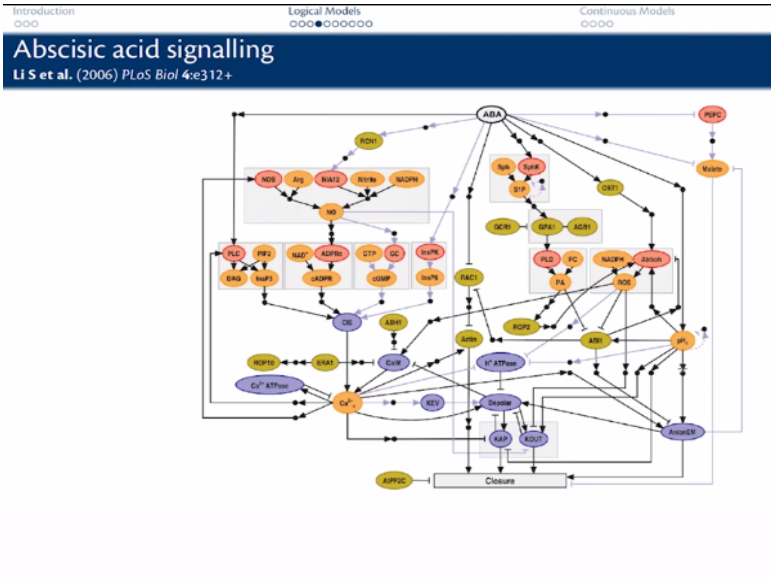
Computational Systems Biology
Karthik Raman
Department of Biotechnology
Indian Institute of Technology – Madras

Lecture – 87
Modelling Gene Regulatory Networks

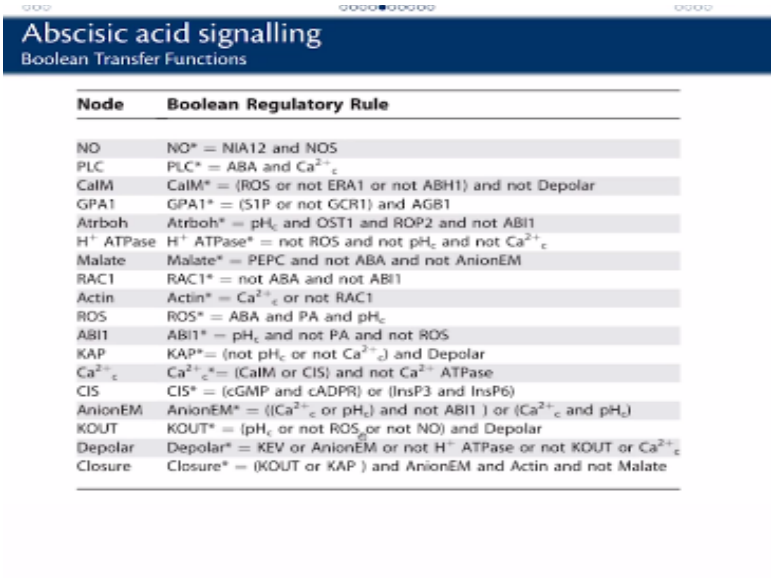
Continue with modelling gene regulatory networks and we look at some simple examples like Abscisic acid signalling which is the classic example of Boolean network modelling to understand a signalling network.

(Refer Slide Time: 00:23)



So, let us look at more realistic examples, so how do we model such systems, right, so this looks at Abscisic acid signalling, it is an important plant hormone which regulates stomatal closures, so when does the stomata close and it is dependent on a complex regulatory network that involves several different proteins and their states and so on, so you have some inhibition, some activations, some complexation so on and so forth, right.

(Refer Slide Time: 00:58)



Abscisic acid signalling
Boolean Transfer Functions

Node	Boolean Regulatory Rule
NO	NO* = NIA12 and NOS
PLC	PLC* = ABA and Ca^{2+}_c
CaIM	CaIM* = (ROS or not ERA1 or not ABH1) and not Depolar
GPA1	GPA1* = (S1P or not GCR1) and AGB1
Atrboh	Atrboh* = pH_c and OST1 and ROP2 and not ABI1
H^+ ATPase	H^+ ATPase* = not ROS and not pH_c and not Ca^{2+}_c
Malate	Malate* = PEPC and not ABA and not AnionEM
RAC1	RAC1* = not ABA and not ABI1
Actin	Actin* = Ca^{2+}_c or not RAC1
ROS	ROS* = ABA and PA and pH_c
ABI1	ABI1* = pH_c and not PA and not ROS
KAP	KAP*= (not pH_c or not Ca^{2+}_c) and Depolar
Ca^{2+}_c	Ca^{2+}_c*= (CaIM or CIS) and not Ca^{2+} ATPase
CIS	CIS* = (cGMP and cADPR) or (InsP3 and InsP6)
AnionEM	AnionEM* = (Ca^{2+}_c or pH_c) and not ABI1) or (Ca^{2+}_c and pH_c)
KOUT	KOUT* = (pH_c or not ROS or not NO) and Depolar
Depolar	Depolar* = KEV or AnionEM or not H^+ ATPase or not KOUT or Ca^{2+}_c
Closure	Closure* = (KOUT or KAP) and AnionEM and Actin and not Malate

So, you have a pretty complex system, so how do you model these, you start writing out equations like this, so you now say that NO depends on NIA 12 and NOS, so if you can spot it in this figure, you have several nodes, NO is here, it depends on NIA 12 and NOS and you will see that there is a complex relationship for each of these things, so calcium depends on calmodulin or CIS and not calcium 2 + ATPase and so on and so forth.

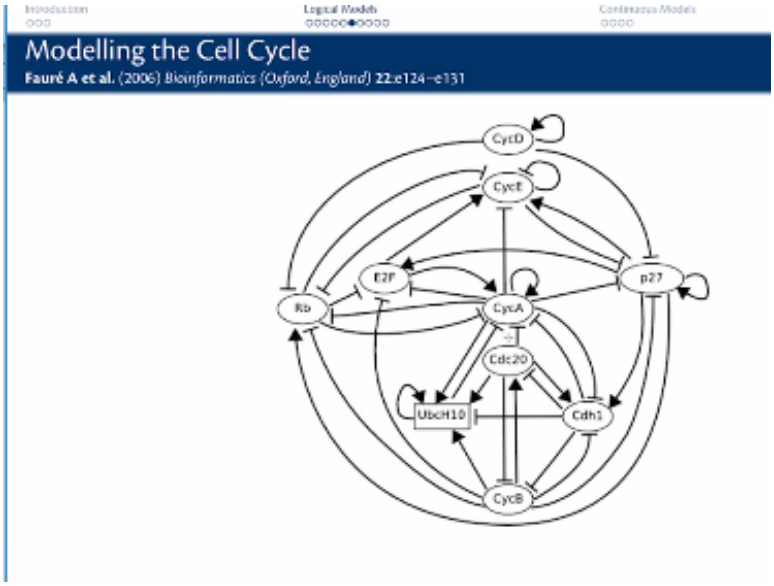
And you know ROS should not be there for something else to happen, so you basically integrate all possible relationships into the model and you will have to understand how each of these, you know the representation works and if there is the one difficult thing would be at nodes like this where you have to find out whether it is an AND or a NOR, right, when you have multiple inhibitions incident on a particular protein which of them supersedes, right.

So, is it an AND or is it an OR, right those things will start becoming important, so you will have to essentially go back to the literature to find these. For example, here it says PH inhibits so many of these things, right, K out is pH pr not ROS or not NO, K out is pH ROS not and NO that comes where are you know, this, okay, so it is easy to kind of frame these rules, you just pick your node of interest, look at all the incident arrows and encode them up, right and depolar.

So, how may arrows are incident here, 1, 2, 3, 4, right and you have 1, 2, 3, 4, it is straightforward but difficult part is making this figure, right, this involves digesting a whole lot

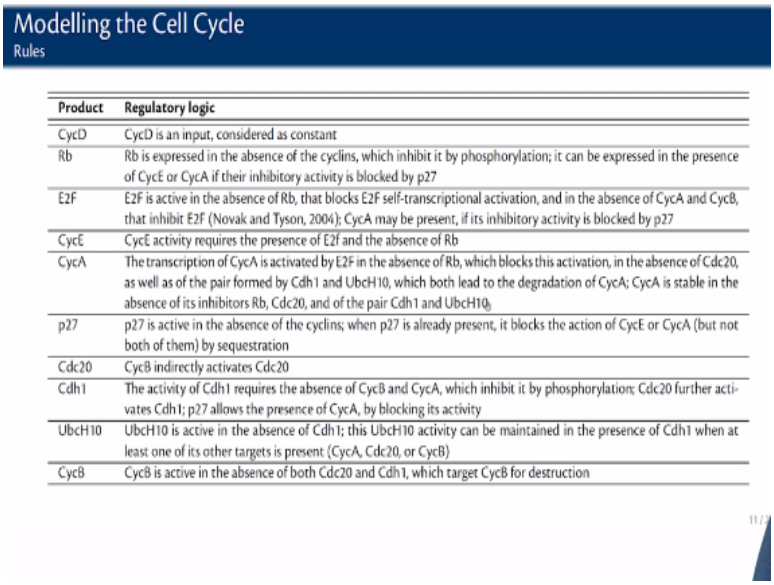
of literature figuring out all sorts of you know inter dependencies and so on and making this figure, once you make this figure, it is easy to build this up.

(Refer Slide Time: 03:02)



Let us look at another example, this is the classic example of modelling the cell cycle, so you must be familiar with the cyclins and so on, so the several cyclin proteins involved and so on.

(Refer Slide Time: 03:13)

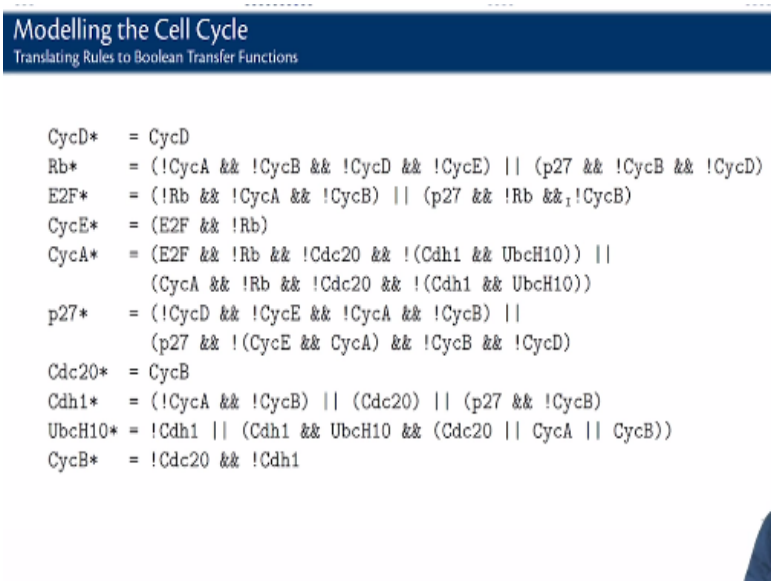


And this is the regulatory logic, it says CycD is an input and is considered constant, which is what would be the rule; there is no update, right, so CycD star as CycD right that would be so, Rb is expressed in the absence of cyclins which inhibited by phosphorylation, it can be expressed

in the presence of cyclin E or cyclin A if their inhibitory activity is blocked by p27, okay or cyclin E activity requires the presence of E2f and the absence of Rb.

So, it is easy to; once you write on these statements, so these are statements that you can make from a figure like this or rather this is what you would get from literature right, if you consolidate so, each statement written here will have the strength; backing strength of 3 or 4 papers which try to describe these kinds of interactions and so on. So, you take these and put them together, you can then build rules such as these, right.

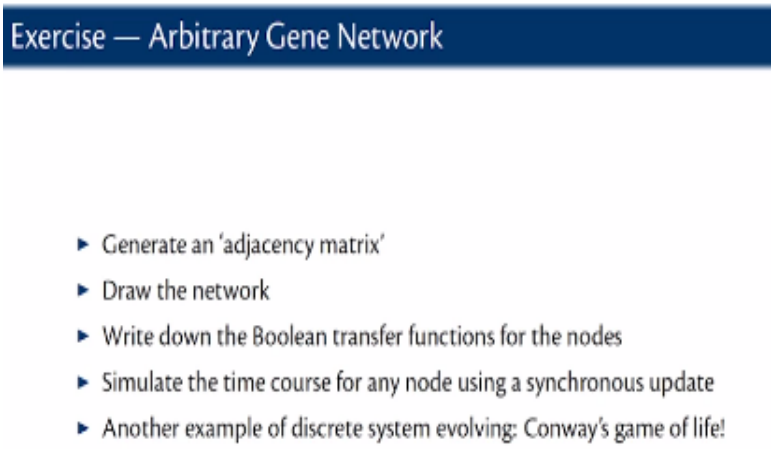
(Refer Slide Time: 04:12)



So, cyclin D star is cyclin, so E2F start is not rubisco and not cyclin A and not cyclin B or p27 and not rubisco and; I do not know if it is rubisco, it is Rbs and cyclin and not cyclin B, so this is you know or let us look at cyclin E that was a simpler rule, so it requires the presence of E2F and the absence of Rb, right. So, Cdc 20 is indirectly activated by cyclin B and cyclin B should not have both of Cdc 20 and Cdh 1.

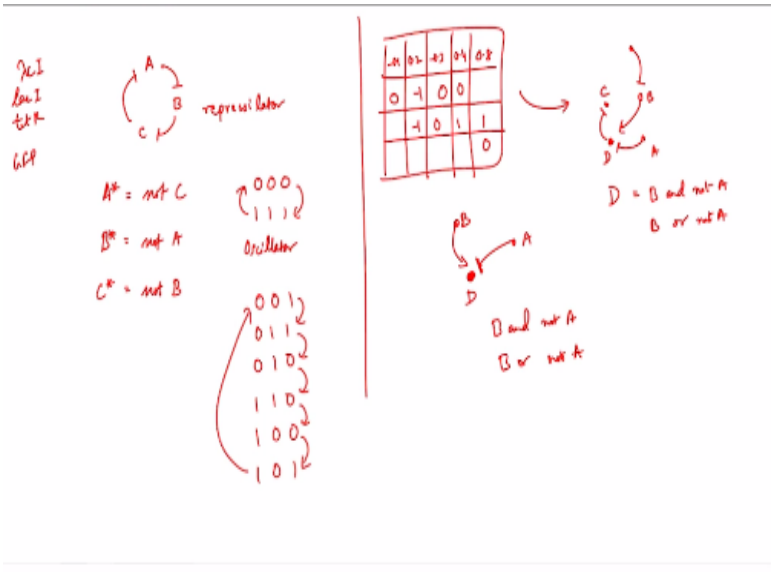
So, cyclin B is active in the absence of both Cdh 20 and; CDC 20 and Cdh1, so you set up rules like these. So, how many states do we have; 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 so, what are the total number of possible states of the model? 2 to the 10, right so you have 2 to the 10 possible states or 1024 and you know where each of these are going to be on and off and there are various nice ways to simulate these models.

(Refer Slide Time: 05:35)



So, let us take some other you know arbitrary gene network and so one thing you can do is; you can just take a adjacency matrix; generate an adjacency matrix, then draw the corresponding network, write down the Boolean transfer functions for the nodes and simulate the time course for any given node using a synchronous update, we will just see how we do this for a very simple system.

(Refer Slide Time: 06:01)



We already did it for one system but let us consider, it is very popular system, we can generate a random adjacency matrix that is what I was trying to say, right so let us say you just start filling the random numbers here, so -0.1, 0.2, -0.3, 0.4, .0.8 something, right, you could have zeros also,

something of this sort and then you create the corresponding adjacency, corresponding gene network and so on, right.

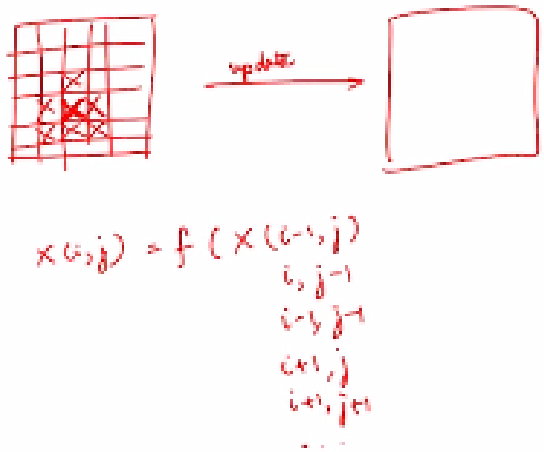
And you can simulate it, so the only thing you will have to bear in mind is; when you have let us say this is C, B and A, now is D going to be B and not A or B or not A, so this is a problem when you start reconstructing from a matrix of this sort but if you are going to look at literature, this will be quite obvious, it will tell you what is the dependencies, so let us anyway leave this out for the moment, let us come back to this circuit.

Have you seen this circuit before, it is classic oscillators that was first synthetic; one of the most popular synthetic oscillators built, there is some sort of cyclic inhibition, there were 3 proteins used, I think there were lambda Ci, lac I, and tet R with a GFP reporter. So, we have a system like this, now what are the rules that you would write; A star is not see C, B star is not A, C star is not B.

Now, let us look at 000, this will just become 111, this will again become; so you have a nice oscillator and the other states will have again you know you will see they cycle through several other states, so let us start with 001, so this will become 011, 010, 110, 100, 101, 001, so here again you have 2 cycle attractors. No, if we just add an adjacency matrix kind of representation, you would not know, right.

So, I have a node like this D, which has 2 incident nodes and repression from A and then activation from B, which of these supersedes; so is it B and not A or B or not A, how important are these 2 interactions relatively? You cannot tell that from a graph, so you cannot tell from adjacency matrix either, right, so the best way to go about this is; you look at the rules carefully and you construct the Boolean transfer functions appropriately and then simulate them.

(Refer Slide Time: 10:55)



There are nice tools that are there for simulating these; I will show them in the next lab. So, another example system of a; example of a discrete system is Conway's game of life, I do not know if you are; some of you are familiar with this as this, you can think of a very large grid and you start with some sort of pattern like this saying there are bacterial cells present here and so at each time, you do an update.

So, every given square has 9 adjacent squares, if at least 5 of them have bacterial cells, then this will become on else it will go off, so this is also like a classic cellular automaton, right so then in the next step, here you will see that you know this has so many neighbours active, so this will again remain but many of this will go off, so it might be; you can say at least, 3 need to be on, or 5 need to be on and so on.

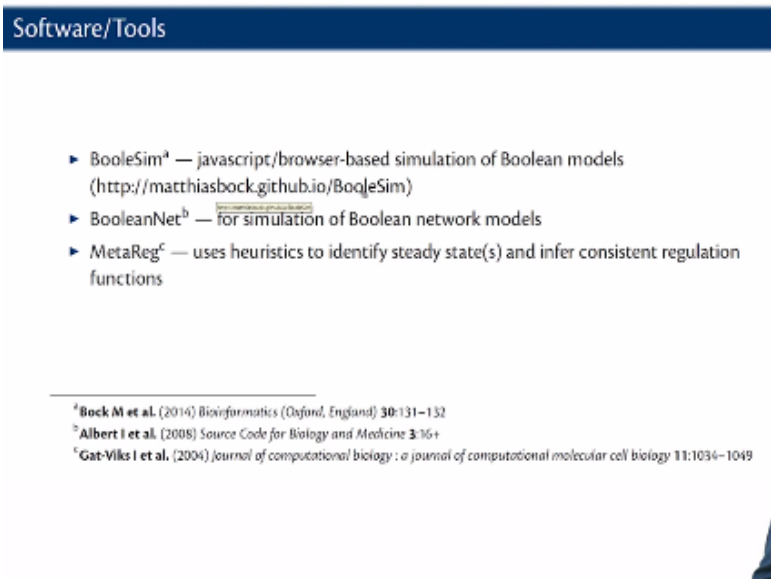
Depending on that you will get different patterns and so on, so you can just update it but basically, this is a discrete update, right, so your X of t +1 depends on X of or let us say, X of I, j depends on X of i – 1j, ij – 1, I – 1, j – 1, I + 1, j; I + 1, j + 1 so on, right, so all these are going to affect each state but this is just a nice example for how you can do discrete modelling, right most of the modelling that we have seen so far are you know, in some sense a little more continuous.

Of course, networks are discrete, the initial networks that we saw in the first module of the course, right. So, the good thing is; you can use this kind of Boolean models for various complex

scenarios, right you can use them, so let us maybe go back a couple of slides, so let us just consider this rule. Here the level of p 27 could very well come out of some other signalling network or something like that which you may obtained by solving ODE's.

And then, maybe do a simple thresholding to say whether it is on or off, so you can use these kinds of Boolean models as a sort of framework to which multiple other kinds of models can connect, so this level of cyclin B could come from a FBA flux if you want, right, so you can do things like that so, you can use these kinds of discrete models to connect several models.

(Refer Slide Time: 13:45)

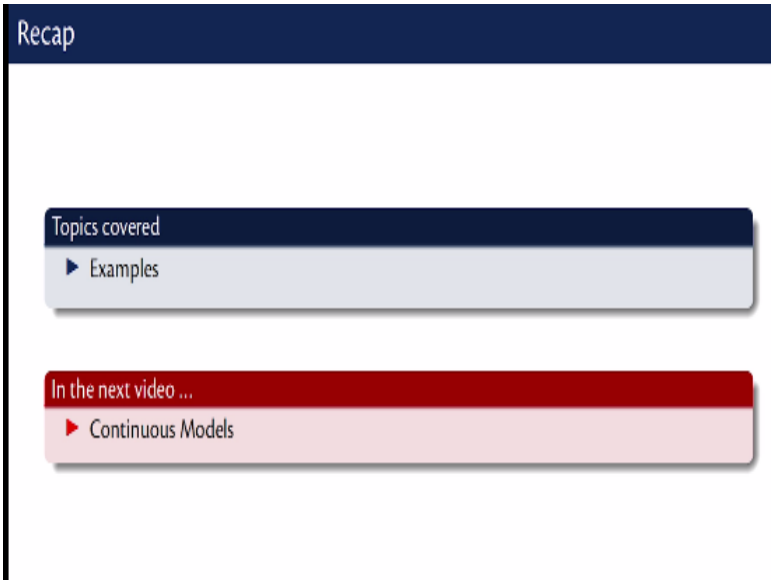


So, there are very nice software tools, the best of them being the more recent BooleSim, which we will work on in the lab in the next class, so this is just a JavaScript based simulation which is this so easy to use, you literally copy paste a bunch of rules like A star = not B, B star = not C and click simulate, you will get a nice time course. This BooleanNet, which is based on Python again, very useful, very powerful.

It can automatically do things like knocking out a single node, as always we wanted to do some perturbations and it is quite easy to do perturbations using tools like Boolean Net, right, so what does perturbing here mean, or what is deleting a node mean, so make it permanently false, right, so make a node permanently false, right. If the node is permanently false, then it is going to have an effect.

So, you just make the update rule for that node as A star = false, it will just be false because it is that protein or whatever is no longer present. There is a MetaReg which uses heuristic to identify steady states and like consistent regulation functions and so on, so there are many interesting tools that are available.

(Refer Slide Time: 15:16)



So, in this video, we looked at some examples and I hope with these examples, you have a good understanding of how one builds and simulates models of Boolean networks to understand say signalling or gene regulation. In the next video, we will look at continuous models wherein we look at model similar to the dynamic models we saw much earlier on in the course for gene regulation and so on.