[SQUEAKING]

[RUSTLING]

[CLICKING]

ROBERT M. TOWNSEND: So today, what I want to do is to tell you about this class and its contents. So it's 14.129. And the correct title is Blockchain and the Design of Financial Systems. It's incorrectly listed in the registrar's office as Advanced Contract Theory, which is true, but we're going to focus on this topic.

And the way I'm going to organize today is to jump right in and tell you the contents of each of the lectures. It's a bit tricky in the sense that I don't want to actually give each one of those lectures. That's going to happen as the class evolves, but I want to give you a good sense of the topics we're going to be covering and a bit of the way we're going to be covering it, and in particular, featuring this blend of computer science and economics.

And after I scroll through a summary of the lectures, then I will turn to the syllabus, tell you a review, again, more about the organization of the class. And I'll also mention a parallel class and its contents, which are relevant, but not a requirement. And that way, you'll see how everything fits together, ideally.

So this is a synthesis of computer science and economics. The buzzwords, so to speak, of computer science will be cryptocurrency, blockchain, tokenization, platforms, and computational algorithms. And the buzzwords, so to speak, of economics are going to be contract theory, mechanism design, general equilibrium theory, and monetary theory.

So we're going to try to get these items on the same page. We want to understand topically the assumptions and shortcomings of three objects-- distributed ledgers, smart contracts, and encryption-- and the potential impact that these technologies have on legacy systems and explore the possibilities for new financial design and the implications for regulation.

So each lecture, as you shall see, or at least a cluster of lectures, features one or several of these computer science technologies and doing that in the context of historical or contemporary applications and in that context using the various economic tools.

So examples that you'll see more momentarily-- community currencies, coordination and financial crashes, tokenized assets and trade fails, multilateral payment and trade credit set-offs, liquidity injections, and financial contagion, risk-sharing auctions, and online markets.

So the goal here is to unpack these different computer science technologies and their assumptions, sometimes rearranging them rather than accepting conventional bundles and labels. There is a lot of difference of opinion about these technologies, a lot of hype and exaggeration on the pro side of what can be done with them, but also a lot of criticism on the negative side of Bitcoin and cryptocurrencies and so on.

So we want to set this polarization aside and get in the middle and be objective. So for example, we can combine the authentication and commitment schemes of encryption with an automated execution engine on a distributed ledger. That's the standard package. But we can also do this with computer code without a consensus algorithm or even use trusted third parties and escrow accounts.

So we're quite open to and should review alternative ways of implementation. Another way to put this-- different technologies allow different implementations of the central planner, which is econ jargon. You'll see more. That's a key construct of economics.

We can gradually abstract away the centralized role of the central planner into concretely deployed mechanisms and into commercial applications. So we're shooting for this balanced perspective, avoiding the hype and ideological positions, focusing instead on ways to implement use cases, allowing policy designers to leverage these tools for the different problems that they're facing.

OK, so lecture 1-- 2-- 1 is today-- is about the blockchain. On each of these slides, as I go through lectures 2, 3, et cetera, you'll see a longer title, which is more descriptive of the components. So this lecture 2 is about a unified view of distributed ledgers and financial accounts, thinking of them both as databases, distinct views of money, and also providing a larger community perspective.

So blockchains and financial accounts are each a database of transactions. And so we're going to think about it that way. However, each is associated with a notion of money. As I've already anticipated, for the blockchain, we think of Bitcoin and cryptocurrency. And many people stop thinking about it after that.

For financial accounts, money is a particular item on the balance sheet of the holder. For example, Fiat money or demand deposits in a commercial bank. So it's also a liability of an issuer, but the truth is that each of distributed ledgers and financial accounts are broader and do not require us to force either one into the construct of money.

So these evocative pictures are more details coming next time on Thursday about financial accounts. And this is actually the blockchain picture of what's going on with the databases. So you basically have data that are hashed pairwise, hashed again, and hashed the third time. So this is basically an archive system that allows you to uncover the individual transactions on, say, Bitcoin.

So block, chain, together-- that's where the terminology comes from as a database. OK? Both blockchains and financial accounts can incorporate multiple objects. And from a monetary perspective, that's fine. You have goods or assets or certificates that, in a monetary theory definition, appear frequently in exchange.

So we have-- you can barely see it-- a matrix of transactions of grain, clothing, labor, money, IOUs, and so on. And we count up how often they are used as we are, say, looking at an individual household or aggregating up to a village or the US economy. So a money is something with a high velocity that appears frequently in exchange or the amount traded per unit outstanding is large.

So it's fine to have multiple monies. In this picture that you can barely see, you would notice that grain is actually being used as well as money. Workers get paid in grain, and they use used the grain to purchase other items. And you have IOUs that are denominated in money and IOUs that are denominated in grain. And all of this is coexisting. And you have objects that rarely appear except in the bilateral exchange.

The point here is that blockchains are broader and the financial accounts are broader. Blockchains in particular are a way to write and execute contracts. So you take the concept of a balance of an item and the transfer change in the balance, as in a transaction. That generalizes to the balance as the state and the transfer corresponds to state changes. So with that simple change in vocabulary, we're writing contracts on the blockchain.

An interesting difference, tension really, appears with an individual versus community perspective. From the financial accounts perspective, where money is an asset, the tension is store the asset as a store of value on the one hand, or use it in transactions. And the more people are, shall we say, hoarding it, the less they're going to be using it for transactions. And we could have a liquidity shortage in the market, even though from each individual's point of view, they're being rational.

Now, very much surprising from a financial accounting perspective-- if you go back to the invention of double entry bookkeeping in Italy with Pacioli, he embraced a community perspective, not individual accounting perspective. And for him, money-- yes, you could hold it, but it was really a liability, not an asset. It's a liability in the sense that there was an obligation to spend it to accomplish some community objective.

I promised you balance. So nothing perfect about the blockchain. There's a theorem in computer science called the CAP theorem, which is an impossibility theorem that you cannot have all three of consistency, availability, and partition tolerance at the same time. And so you have to make choices. And that same trilemma can carry over and allow us to interpret village economies and contemporary US economies and so on in the sense of having to make choices.

Well, again, just to repeat, I don't expect us all to completely understand these slides. I'm trying to give you an introduction to what we're going to be doing in more detail in this case on Thursday. So I'll resist the temptation to lecture in too much detail.

So this part is having to do with the opportunities of the course, which is these so-called readings, which are featured. Here we have Agustin Carstens, "The Future Monetary System-- Vision to Reality" from the BIS. Relatively recently, this strange thing called "The Baby-Sitting Economy," which Paul Krugman reviewed, has to do with an object, which are babysitting coupons or certificates that were used in exchange and analogies to money and what can go wrong.

Goldstein and co-authors explore this tension in real-time gross settlement between "Payments, Reserves, and Financial Fragility." And De Meijer is an article about Kenya, where cryptocurrency blockchains are being used to serve the purpose of targeting poor households in rural or urban communities.

So in the lectures, I touch on these articles, but the main reason they're here is for your interest. These or other papers may be something you would like to explore. And there's, as I'll say later, no midterm, no exam. It's all about research. So you have the option to pursue what you're interested in most. And we provide some pathways.

So at the end of each of these lecture two-page summaries, I'll just list some of the papers. OK, lecture 3 goes back to these tools-- in this case, distributed ledgers-- and what you can do with them. The longer title is Fragmented Markets, Policy Objectives, Regulatory Solution, and Distributed Ledgers as a Technology Solution.

We're going to put some friction in the markets and then see what problem emerges and how to remedy the problem. Sometimes, there are regulatory solutions, as in this US national marketing system, the way the US does the regulation. And we will also feature the use of distributed ledgers as a solution to an information problem.

Now, time and time again, when you're reading on the web, you'll come across a use case or an objective, something perhaps the private sector is doing or central banks are doing. And they just launch right into the application. Why? What are they trying to accomplish, exactly?

Heaven forbid just use the new technologies for the sake of doing it without thinking about social welfare. So we will get into this. Economists have a criterion typically used, called Pareto optimality, which is a weak standard that says whatever is going on currently, you should not be able to make some participants better off without harming other participants because then there's a gain for everyone. So it could not possibly be efficient the way it is.

So these slides have to do with prices should be equated, even if we have fragmented markets, which is what the US is doing. They look at the execution of stock market trades and have regulations that try to make sure people aren't being cheated and buying things more expensively than they would have if they had bought on a different exchange.

The blockchain point of view is let's fragment the markets artificially, take that as a given, and then try to achieve what would have been a competitive outcome or an efficient outcome with agents constrained to be trading with each other pairwise in these fragmented markets.

And what happens here is that even if you impose the correct prices, it is, in general, impossible for agents to solve the problem in a decentralized way, looking at their own histories or the histories of people that they've met. It's just like programming code and deciding what information the code is allowed to use or would be required to use in order to get to the objective.

And bilateral credits and debits are, in general, not sufficient. You need to invoke this community perspective, like the common but distributed ledger. These pictures at the bottom are like a balance sheet at a point in time and the transactions that change the balance. And they evolve over time as an individual is trading.

And having individuals see only their own accounts like this or even those of their trading partner will not allow you, in general, to get to the Walrasian outcome. Now, you may think there is a way to do better with money, just like a decentralized system where you carry around your own cash or e-credit and spend it as you wish, each person doing that.

The problem is it takes an enormous amount of liquidity for agents to achieve the objective of the efficient Walrasian allocation. So the decentralized way in which we're used to thinking about money is actually inadequate. And in any event, in real-time gross settlement systems, there's a misnomer because if it really were instantaneous real-time, you'd have to have a lot of liquidity for banks to be able to honor presentation of commitments due.

So almost simultaneously, with the coming of real-time gross settlement instead of end-of-day settlement, there came this quest to find liquidity-saving mechanisms. And Ostroy and Starr anticipates this. Another thing you could do is have a warehouse facility so nobody trades until they meet the big trader.

And what goes wrong? Well, a couple of things. That trader would have to have enormous inventories of all the goods or assets in order to honor the requests for outflows. And likewise, having a very big trader like that, he or she will be tempted to exploit their market power, which I claim we also see in practice. And I'll show you more when we get to this lecture.

And finally, you could think, well, why money? Why not credit? Let's just have deficit accounts, overdraft accounts, and let people spend out of those accounts. Well, that's a little bit like the end-of-day settlement. And the problem is people show up having bought more in value than they sold.

And they can't honor the commitment to balance their account at the end. So that's like a default, and we see that a lot. So this theory article is amazingly capturing a lot of the current institutions we see, but remember. We're talking about a blockchain-- distributed ledgers as a way to solve those problems.

OK, so again, a hint of things to explore. Chester Spatt has written, with his co-author, this "Regulating Market Microstructure." You can read all about what goes on in the US in terms of regulation and other related articles, or this liquidity savings mechanisms that I've already talked about, as described in Martin and McAndrews.

This Alcazar has to do with a very concentrated provision of core banking services-- in other words, imperfect competition. And the "Tri-Party" paper is about these enormous potential liabilities that the Federal Reserve took on in the operation of the repo markets, often having within middle-of-day liabilities, larger than the entire US money stock. Of course, they figured it out eventually and tried to fix it. So again, as your interests emerge-- and you can browse and then decide no, I don't want to go down that path. That's fine.

OK. So featured distributed ledgers-- now, I want to feature smart contracts, and also how you can use them, in this case, as a solution to a coordination problem. So here we're going to have the commodity space being locations and dates and randomly realized dates of the world, but we're going to stick with this Pareto criterion for efficiency.

And with fragmented markets, we will try to implement the solution with privately issued securities or monies. They will be high velocity and circulate in exchange, but it's going to lead to a coordination problem and to crises. And the solution will come with this multi-agent smart contract.

So this picture is meant to evoke that agents are getting partitioned. They're either in one location or another. We settle for two. They meet pairwise like that in trade, but then they get buffered around and sorted with different agents in yet other locations. And they can issue these securities that circulate.

One can issue a promise to pay at date 4, issue it to 2, 2 carries it to 4, 4 trades to 3, and 3 presents it for redemption. So this private security, this IOU, will circulate in this model with high velocity. That's fine. It's a privately issued money.

The problem is there are other privately issued monies issued in the other location. And if you don't have this coordination problem solved, they may incorrectly anticipate what's going on in other locations where assets are being issued, but they cannot see that.

And if they guess incorrectly, you'll get a crash. Now, we see this in theory very clearly because we've simulated the model. And I'll show you that in this lecture 4. We've also seen it historically. So Walter Bagehot was describing the money markets in London, where these bills of exchange were circulating with high velocity.

But then there are these periodic crashes. And I was over the break in London at the central bank, went looking. It occurred to me-- I was too early for my appointment-- to look for Lombard Street. Lombard Street is still there. And that's where the first clearinghouse was ever established.

So it was real, and it was the center of the world's money markets. And what was going on then several hundred years ago is still-- we still face those problems today. There are related things. There's a concern in low and middle income countries with this digitization that is emerging and the fact that it's not coordinated.

So parties will be having to basically cover balances without having the assets, like a big liability. That's a concern in some of these countries. And likewise, even with Decentralized Finance, or DeFi, which are these markets for cryptocurrencies, there is a concern that we don't have enough liquidity or that there is a liquidity problem. So the solution to these problems turns out to be available now with the distributed ledger and smart contracts.

So these articles are using the Pareto criterion to explore whether or not you need to intervene. In this case in the US, the paper by Gorton is about those bills of exchange in England and how-- he calls it private money production without banks. Bagehot, I've just shown you on the screen.

Sargent and Wallace is the paper about real bills versus the quantity theory, having to do with causes of inflation. The quantity theory is that inflation is determined by the quantity of money, and we should be careful that all monies be limited in supply for that reason.

The real bills doctrine says no, that's wrong. Financial intermediation is fine as long as there is real backing for the liabilities, the so-called real bills doctrine. And this is the basis of a debate that is still going on today that is relevant for central banks like the US Federal Reserve. And I've mentioned the liquidity fragmentation. OK.

Lecture 5 goes further in exploring tokenized and programmable assets. So now, the ledger has become dynamic. We can talk about atomic trade and settlement and do that on platforms and compare these tokenized immediate settlement assets on the blockchain with the current legacy systems where there are trade fails.

So the buzzwords are tokenization and programmable assets that are made possible by multilateral smart contracts. Now, when you think about the accounting point of view of legacy systems, one thing that hits you is the balance sheets, assets, and liabilities should not be just simple static representations of a trader's assets and liabilities.

It should be a dynamic, explicit representation of future assets and liabilities that they contract for under these multilateral smart contracts. And likewise, you can agree for trades into the future and commit to them now. So you have this odd juxtaposition of immediate settlement in the sense of not only trading, but committing to the outcome of the trade, whereas in legacy systems, it's end-of-day or t plus 3. And you may or may not get the asset you thought you purchased.

And the guy with the money may not show up or vice versa. The asset is not there when you need it. So this little example here at the bottom is about three agents, simple prototype, where it's asset lending. And they want to hold different levels of the asset at different points in time as an example of multilateral asset trades.

I mentioned the legacy system and trade fails. And it's shocking at first, when you first read about it, the magnitude of the fails. So for example, here are treasuries in it says, billions of dollars. But 10,000 billion-- we're into the trillions. So we're measuring here the value in dollars of trades that were agreed to, but not executed. So it's, quote, "arguably a big problem."

Now, the blockchain isn't a cure-all, either, because various problems would emerge, as will be clear from that example. If it were the case the asset can only be programmed by someone who currently holds it, then entering into a trade could reveal information that they would rather keep private, which, in turn, may mean that they're not going to get a good price.

And there are potential solutions having to do with centralizing these markets or being able to use the conditional release function on the tokenized asset, which we can talk about more later, or with encryption. But in general, these systems that emerge may be only partially interoperable.

And even the tokenization raises issues of trust if, in fact, it's a pre-existing asset that has to be put in escrow in order to allow us to tokenize and write code on it. You're relying on the issuer to hold it to the escrow account to be valid and for them not to cheat and use the asset. It's got to be there when you need it.

So there are various explorations going on among domestic central banks and international agencies for a cross-border exchange problem. Such a blockchain has been proposed for foreign exchange transaction. And the BIS has now switched gears a bit and is proposing unified ledgers even for domestic systems.

But we have to be careful that, in fact, we don't end up violating the coherence guarantee, which is a big feature of what you can do with these multilateral smart contracts. So again, these readings-- read all about settlement fails, understand better this risk of tokenization, be mindful of what it means to have programmable money with this coherence guarantee.

And these two lines at the end have to do with what we're seeing out there currently from the Swiss National Bank and other consortia of large banks. So again, each of these, the last in particular, are options for you to explore in your readings. How does it work, exactly, critique it, understand it better-- would all be potential projects.

OK, so moving on, we come to algorithmic flows on networks as solution to multilateral settlement problems. Or the longer title is How to Improve Financial Infrastructure-- Taking Advantage of What We Know about Network Cycles and Chains Using Algorithms to Maximize Timely Payment of Multilateral Trade Credit Offsets, or alternatively, back to the US repo market-- what to do about credit guarantees, time permitting.

So we put the obligations of, say, trader I to pay trader J on a network diagram. So for example, A owes 2 units to B, B owes 2 units to C, et cetera. C owes 1 unit to A. So those edges, when they connect the two nodes, represent a liability for one party and an asset for the other party if it's paid.

But agents can be indirectly connected in various complicated ways. In fact, when you look at an actual picture of networks-- this is Italian data-- you can actually see the individual nodes and the edges because they're just tens of thousands of transactions.

So you can begin to conjure up this interconnectedness, which is very complicated. In fact, individual agents are very unlikely to know much at all about the overall structure of the network. They'll know their own assets and liabilities, maybe a bit about their partners, but not much more than that, typically.

The second problem is even if we had a multilateral smart contract node that did have the complete knowledge of the network, like a community ledger, it would still have to determine the ordering of the payments. Does A pay B, B pay C, then C pays A? But B has to be paid first in order to pay C, et cetera. It becomes a very large combinatorial problem to figure out the correct judicious ordering of the payments.

And finally, even if there were a liquidity pool or some kind of overdraft facility that, in principle, could maximize the amount cleared, you'd still have to decide where to inject the liquidity, how much to inject, and how to recover that liquidity at the collection point in such a way that the system still remains balanced.

So these problems are very typical. Here we're, in this lecture, going to feature the problem of trade credit overdues and how to have this multilateral set-off, either clearing enclosed cycles or injecting liquidity through chains. You can better see that here, where this, what's left of it, is going to turn out to be a closed cycle, where one is paid, paid, paid, and it clears, but the red is the liquidity injection from the source to the target, which also clears a lot more of the trade.

Oh. So for this lecture, I have invited Tomaž Fleischman, who is a collaborator in this work, who is a computer scientist and also helps run a company with a blockchain. So he will tell you much more in that lecture, lecture 6.

And there's, indeed, Fleischman with his co-authors, "Liquidity-Saving Obligation." There's a lot to read. I just picked one of his papers. You may, depending on how far you are in your studies, recognize this Cormen, et al. book on *Introduction to Algorithms.* It's co-authored with MIT professors. And again, time permitting, we could get into the repo market. I put this here because if we don't do it in class, then that's an option for one of you in terms of your own research project.

So the previous slide, lecture 6, was something static, but you can make these networks dynamic and stochastic. So this is a picture of the federal funds market. And you can see, depending on the time of day, the evolution of the network where these traders are, again, are connected with edges.

So in this case, we go back to the basic general equilibrium idea of keeping track of dates and states of the world, but in this case, not just for incomes, but also for who's trading with whom. So that's a larger state. And it becomes a risk-sharing problem to try to stabilize consumption out of income.

And then think about injecting liquidity in this context. So if extra liquidity were available, say, from some new monetary policy-- because that is not the standard way that monetary policy is implemented, but it could be implemented on these blockchains-- who should get the liquidity?

And the answer is going to turn out to be the most valued node or recipient of the liquidity, assuming that they have to get the money in advance, would be a trader that participates in market clusters, even when the number of total participants is low, when there's aggregate correlated shocks to those in the cluster where average incomes and yields are low for those in the cluster, and where agents are very risk-averse.

So we have a criterion to determine the most valued players. You would see this in large financial markets if we thought of a bond which pays off when someone is in the market. An individual I-specific bond-- the value of that bond would be the value of this key player.

Or in village settings-- and we've done this-- key players turn out to be people who are, as the theory says, participating with other agents, even when there aren't very many transactions. And those agents turn out to get higher consumption on average, as if they were receiving a premium, all of which is informal. And they don't use that language.

And then a juxtaposition. There is another literature that seems to start out the same way-- financial markets with nodes and interconnectedness-- but it's a literature about contagion. There the idea is that traders get hit with an adverse shock, like a disease, and it's contagious, and it spreads to other traders.

And that argues for limiting markets. It's exactly the opposite of the risk-sharing point of view, where you want to increase the number of traders in order to allow risk mitigation. And this picture is just meant to tell you that these two notions of financially central players are different from one another. And I will go into more detail on that later.

Here's two papers, this *Contagion!* book-- exclamation mark-- *Systemic Risk in*-- that's the framework policymakers are using today. This is the way they think about financial markets. And so they're trying to limit the damage that would happen if something went wrong rather than the other perspective, which is to try to make markets stick. So you can explore the book with its many chapters. And Martin Sumner has a very nice review piece about financial contagion.

OK. So then we get into mechanism design, and in particular, incentives to follow protocols and notions of trust. So the longer title is Private Information, Incentives to Report and Take Actions, the Byzantine Generals Problem, Differences between Economics and Computer Science Approaches to Trust.

So when we do mechanism design, we're clear about the incentives to report the truth and take appropriate actions, as with an insurance company prepared to cover the loss of a client, but the loss is private to the agent, who will be inclined to lie about it in order to get the indemnity, although there are ways to mitigate that problem.

Implementation looks like we might need that central planner that I mentioned earlier, implementing the smart contract code, but we really don't need the central planner at all. We just need the code. The code can be validated by agents in advance with a public commitment to carry out the plan.

The payouts that might be necessary can be put in escrow and programmed, as with the dynamic assets, to be available when needed. Messages about claims can be recorded and put into this database of the contract. And implementation does not require implementation on the blockchain.

This layer 1, layer 2 terminology has to do with having the code with the smart contract taking messages from the agents about the state of their ledgers and assigning new balance sheet positions based on what they claim. And I mentioned at the beginning this pragmatic point of view, that you don't have to have every piece of the blockchain in order to make good use of the technology. This is an example of that.

In contrast, the computer scientists tend to rely on parallel computing notions of failure. So the idea is a given computer can fail or give an incorrect answer, but if you're able to put a bound on the probability, then if you have a sufficient number of computers replicating what should be the same outcome, you can determine the truth of the matter.

And they use the same framework to think about rogue traders. That is to say a certain number of evil people could not follow a protocol, but you design the protocol that's tolerant to that, as in Byzantine fault-tolerant validation algorithms.

The Bitcoin with its proof of work is, again, a way to-- by effectively randomly choosing the proposed truth of the validation, the odds that you would choose a rogue trader to do the value can be small if there's a large number of people trying to do the proof of work algorithm.

And this bottom part here has to do with the Byzantine generals problem, having to do with two generals who have to communicate with one another. One gets the signal. The enemy is prepared. You don't want to attack if the enemy is it's prepared. But even if the enemy is unprepared, you have to communicate to the other general. And these messages may fail to arrive with a certain probability.

There is a protocol that works quite well if agents follow it, which is for the second general to attack under certain conditions. I won't go into the details, but that protocol is vulnerable to the economic concept of incentives and Bayesian Nash equilibrium.

Traders can convince themselves to never attack, actually, even when the enemy is unprepared because they're second-guessing and third-guessing what the other guy is doing. So we'll talk about that more in lecture 8. Here are two articles on computing and mechanism design and distributed ledgers and the governance of money that both have to do with computing and validation algorithms. So again, these are options for you to consider when you're thinking about reporting in or doing your research proposal.

So we finally get, in lecture 9, to the third leg, so to speak, of these new technologies. We covered distributed ledgers and smart contracts. And now, we get to encryption. This little picture over there is meant to remind you that keeping secrets has been with us for a very long time.

In Mesopotamia, they used to create tokens as an invoice of what's in the shipment. And they put those tokens in a clay envelope and baked it hard and then wrote little pictures of the internal tokens that accompanied the shipment. When you think about it, you begin to realize that it would be very, very difficult for someone to steal the shipment without the knowledge of the recipient when it finally gets there.

So it's a bit like private and public keys, which are the centerpiece now of modern encryption. It allows people to send a secret message. And the recipient can decipher it. For outgoing messages, it allows a receiver to authenticate the origin of the message, know who sent it so the sender cannot repudiate it, and verify the message has not been modified.

So these are very powerful tools related to hashes and cryptographic puzzles, also to fully homomorphic encryption, which, with the math of it, is like setting up parallel spaces that are one-to-one, the actual text versus the encoded system.

So you can do addition and multiplication on these encrypted systems and end up with the correct answer, even without knowing the actual inputs. And multiparty computation is another leg of this including zero-knowledge proofs, which we will go into.

And these are papers, one by this Harvard professor on zero-knowledge mechanisms, which is a mechanism design problem. And again, back to-- see, in practice, chain link, layer zero, and this new Dfinity internet computer in your readings to see. So these are actual implementations that industry and central banks are adopting.

Then in lecture 10, we will go through designs using encryption for auctions, for example. We can encrypt the bids and nevertheless determine the winner. We don't need some third-party auctioneer who is in a position to cheat. Back to that insurance example with agents experiencing shocks to their balance sheets, which they would like to have insured but don't want to reveal to third parties the state of their balances, we will show how to handle that.

And the information here gets scrambled up so no one has the whole picture. And yet somehow, amazingly, the code can do what you want, including randomize at the crucial key moments, even though the code, quote unquote, "is not a person," and does not know the underlying true state. So you get these miracles happening using fully homomorphic encryption and multiparty computation.

And you can use this also for centralized matching of agents who submit supply and demand schedules, but they're encrypted. And nevertheless, you can basically intersect supply and demand schedules in the encrypted space and figure out the equilibrium outcome. And you can put encryption, as in these papers, including this work that MIT is doing with Visa on encryption in the blockchain.

And finally, we get lecture 11, which has to do with this juxtaposition-- on the one hand, Dubey, "Price-Quantity Strategic Market Games," which is a classic economics paper on an implementation that gets you to the Walrasian outcome.

And on the other hand, we have SPEEDEX as a decentralized exchange that bears not a small amount of similarity to Dubey. And we need to put those things together somehow and deal with the fact that we have many, many agents, each with their strategies. And it is typically hard to find the Nash equilibrium that we're looking for.

It's computationally complex, certainly at the level of individual agents. So this ongoing work-- the idea is to allow the algorithm to compute the [? ideal ?] candidate for the equilibrium and then signal to the agents what their strategy ought to be.

And they know that the mechanism operates that way, so they don't have to have endless conjectures and what-if statements about what the other agents are doing. So that work is based in part on Daskalakis. And Costas is, as you guys know, a computer science professor here.

So those are the 11 lectures. I hope you did not find it too bewildering. It is hard to go through it without getting into the details. And yet we always move on to yet another lecture. I was taught somewhere that if you're teaching well, you tell students what you're going to do and then do it, and then tell them what you did. So this was in the spirit of here's what we're going to do.