[SQUEAKING]

[RUSTLING]

[CLICKING]

TOMAŽ FLEISCHMAN: So thank you, everyone, for coming. So my name is Tomaž Fleischman. I'm coming from a company called Informal Systems, and I'm working with your professor here on a subject called MTCS, or Multilateral Trade Credit Set-off.

So today, we will explore what this is in a series of small examples, and we will dive into details to understand what implications this has to the real world.

So the schedule for today. We will look at one of the problems that is prevalent on the market. This is late payment. We will try to understand what this is. Then we will look into how can we observe this through observing networks, obligation networks. Then we will formalize the MTCS as an algorithm, look how it works, and then we will look at the empirical setting. So how does it play out in the real-life situations?

And we will look at some things that are, let's say, challenges in this area. So the network topology. And what can we do further by injecting liquidity using these techniques? So this is the program for today. And let's start with definition of the late payment.

So late payment, you might know something about it from the personal settings like friends owing you money. So in business setup, that means me offering someone my goods or services on credit, expecting payment later in, let's say, 30 days. And when this does not happen, we have a late payment.

And this late payment is not just, OK, it's delayed. So there are many, many issues concerning with this. So it raises cost because we have to finance this, that it is not paid. It depletes the cash reserves, and it depletes the cash reserves of those who are least capable of doing so. Small, vulnerable companies.

So there are administrative costs, so we have to manage the late payments. It's a drain on labor productivity because it's unnecessary work. It creates substantial distractions from the work. So this is just from the business point of view of an individual firm, it's a problem.

Looking at systemically, so it might look nice, your firm might look nice on paper, but it's not because you have basically losses and not properly managed working capital because of late payment. It places a huge burden on small companies to finance this. It causes unemployment. It causes bankruptcy.

Actually, this is the major-- this is the number one reason for bankruptcies. It's killing otherwise profitable businesses. And all this basically creates huge barriers for small firms to enter. So as such, late payment is not just something that is not pleasant to see, it is an actual systemic problem that needs to be resolved.

So how do we measure this? So how do we observe late payment? So this is an example of a report. This is by Intrum Justitia. It's a firm that does this for European Union annually. This is just an example from their report, United Kingdom, year 2020.

And what we see here, for example, the top chart is comparison of agreed terms and actual terms. And you see that agreed terms are always lower than the actual. So 21 to 31, 45 to 64. So this extra time needed to actually pay is actually late payment.

Then here on the middle, you see, from the size of the companies, how it looks like. So the larger companies more often demand from the users-- from their suppliers to extend the payment terms. And here on the bottom, it's this was just in the COVID, so you see that the sentiment went very sour during the COVID. So the expectations that there will be more late payment went up significantly during the COVID.

So this is how you can see that late payment is there, and we can see, it's a prevalent problem. It's not just UK. This is the latest data from 2024. And you see, across a number of European countries, you have this around 60-day payment, which is definitely above the agreed terms.

And you can also see that this goes across the sectors. So it's not just manufacturing or services, it's basically everywhere. OK.

So from a point of a firm, how does it look like. So let's look at this chart. This chart was prepared by the English Association of Chartered Accountants. It's basically a toolkit for members of this association to estimate and then plan, how do I handle the late payment? And here, you see in this Venn diagram, that this-- try to sort the reasons.

So you can have a very simple tactical reason, like, I just don't want to pay. It serves me well to pay later. There could be real trouble, like default-- I don't have means to pay. And there can be other stuff, like you are going beyond the regulatory practice, or maybe you have some administrative issues and you are just outside of the agreed payment terms.

The interesting one is here in the middle-- so if you mix everything, you get this, buyer default in bad faith, which basically means you did it on purpose. And unfortunately, this is happening, too. So the normal thing that happens is like this. So it starts with, OK, I don't want to pay. And then it goes into, oh, I demand that you approve that I pay later. And then from that, it goes into not just I demand that I pay later, I want a discount. So this is a typical extortion scenario that happens all the time.

And from the other side, I'm a little bit tight on money, I just don't want to pay right now. Then you get in all kinds of issues. And finally, you end up with the same time. So you are basically going into default and you have to-- you have to take action as a firm against non-performing customer.

So what you can learn from this-- so this is a tool for a firm. What you can learn from this, that late payment is a very complex issue. So for a firm, it takes a lot of effort to manage this. So it's-- it is not a trivial matter.

So another way to measure this is to just take into the account all the late payments, but this is very difficult to do. So here, we have an example of a country-- it's Slovenia, it's my home country-- where we actually did it. So here, you have by years, '91 to '94. Percent of GDP. And the blue line is reported late payment.

So there was a mechanism set up. Firms would simply report, I'm late. Why would they report? Because this agency doing this also organized clearing of late payments. And the red line shows you, in a percentage of GDP, how much of the late payment was actually resolved through the mechanism.

The other thing, you can observe this yellow bars, show you the annual change of the GDP. So obviously, we were in bad shape in '91. And then getting better as the economy gets better, the late payment as a problem is diminished, so the reported late payment obligations go down.

But this scheme-- so this is a very specific moment in time for Slovenia. Slovenia just became independent in '91-- we had war here, so this is a war economy result. And '92 is the first year of being a state and have everything function.

The interesting thing, the mechanism was set up by Parliament and monitored by Parliament, so all the data was public. Then in '95, things went under government, and then the data became unavailable. Then in year 2002, the mechanism went into a government agency and the data became public again. And we see, we were still doing fine with some nice GDP growth year by year.

And the reported late payments going down until you-- I think you know this area. So we have this financial crisis, 2008 and later. And you see the interesting jump, yeah. The late payments went up significantly, and so did the results. The similar thing you can observe here for the COVID.

So by observing this graph, you might say, OK, but this mechanism is not really working. Why is it going down all the time? So there is some background information you need, and this is that, in this crisis, private firms establish their own clearing mechanisms, and currently, private firms clear more than the public agency, but private firms don't share the data. But I know the biggest one is bigger than the public service right now.

So we have this clearing of late payment set up in Slovenia for years. And as such, basically the motivation for me to go outside was to explore, why is not everyone doing this? So this is the point

OK, so late payments. Now the question, how do we do-- how do we do it? So how does the management of late payment actually work? So what is this agency doing-- or what can a private firm do to help the wider economy? And we will look first how to form a graph.

So the example here is very simple. My favorite actors, Alice, Bob, and Charlie, are the main characters of this game. So Alice owes 2 to Bob, Bob owes 2 to Charlie, and Charlie owes 1 to Alice. So these are the outstanding trade credit obligations.

And in this simple example, it's trivial. So you can see a cycle. Alice, Bob, Charlie, this is a cycle. The amounts are not the same. So we have the smallest amount here, 1. So a cycle of 1 between Alice, Bob, and Charlie can be set off in a multilateral setup. So you can simply clear this without any use of outside money. And what remains, then, would be Alice owing 1 to Bob and Bob, 1 to Charlie. So this is how it works. It's very simple.

The problem is, even in this example, they cannot do it alone. So Alice, how can she know that Bob owes 2 to charlie? She cannot, so she has to ask. So you need some kind-- so you cannot do this alone because you don't have a full view of a network.

You see yourself, you see your obligation, you see what other owes you, but without some kind of coordination, you cannot achieve a thing. And also detecting a cycle here, as it looks trivial. Now imagine you have thousands of them, so how do you do this?

So the trick here is that you don't look at cycles at all. So we have to reformulate the problem. So we are not looking for cycles. We are looking for balances. This is a very simple approach.

So Alice, she has a balanced position minus 1. Why? Because she owes 2 and she is out 1, so the difference is she owes 1 more. So she is minus 1. Bob is neutral, so 2 in, 2 out, 0. And Charlie has balance position of 1, 2 in, 1 out. So this is our initial setup.

So what we want to do, then, with this is we say, let's balance this network fully. So we don't want to have Alice and Charlie with minus 1 and 1, we want to balance this. And we do this by introducing imaginary liquidity because we are talking about late payment. So late payment happens because there are no funds.

So we are doing here, introducing imaginary source of funding, and we have a sink of funds. And we use this source and sink to balance the network. So Alice was minus 1, so she needs 1 from the source. Charlie was plus 1, so he has 1 too much, so this excess flow goes to the sink. So this is how we balance the network.

Now with the network balanced, at least the original network, Alice, Bob, and Charlie, balanced, now we can do some nice algorithms. And this one is, let's find a saturating flow from source to sink. So we are just pushing liquidity from source to sink in the most efficient way possible, and this is shown here by the dotted red arrows. So we have a flow of 1, going from source to Alice, then 1 from Alice to Bob, 1 from Bob to Charlie, and 1 from Charlie to the sink. So we have a saturating flow.

By construction, the outflows of source are always equal to inflows of the sink. So if you saturate, it just means you have to push everything through. So now we have this saturating s-t flow. And the next step we do is we simply remove the saturating flow. And while removing it, it shows the cycle. So this is how the cycle is identified. And this technique works in-- doesn't matter how large the network.

So at this point, are there any questions? So anyone confused, or was I clear enough? So everyone is nodding, so for the camera because they don't hear you. OK. So everyone is clear? Good.

So this was just a really simple example. So let's try to formalize it in a bit of mathematics. So what do we have to do? So we have our network, and we have to introduce the sink and-- source and sink. And we do this in order for everything to be balanced.

So we introduced the network, G0 with nodes with edges. And we say that-- we define the balance as a difference between the outflows and inflows for each node. And then we define the balanced network G, which is basically the original network plus source and sink. And we add the edges.

So just for every node that has a negative balance in the original network, we add an edge from the sink, and for every node that has a positive, we add an edge from that node to the sink. And the capacity of these edges are equal to the balance-- or minus balance in the case of the source. So this is this part.

And the resulting structure is a balanced graph. So we can define this balanced graph as a graph with capacities on each edges. Capacities are always positive, so we don't allow negative capacities. Each edge also gets assigned a cost.

So I will make here a comment. In today's lecture, we use only the basic case. Each edge has cost 1. There are further possibilities, further optimizations you can do if you play with cost, but for the basic case, we have cost on all edges set to 1.

So we have a source and the sink, and all the nodes in the graph except the source and sink are balanced. So this is our balanced graph, this is the base. Now we have to apply the minimum cost, maximum flow algorithm.

So those who study computer science, if you went to some graph flow algorithm classes, should know this. I assumed that at least computer scientists know this because otherwise, it's another lecture about this. And we apply this to identify the saturating s-t flow.

So just a basic definition. The minimum cost, maximum flow is basically a function that assigns a value to each edge. So the value is somewhere between 0 and the capacity of the edge. There must be a flow conservation principle respected. So that means whatever goes into the node goes out.

And we want to maximize the total flow. This is to achieve the saturation of s-t flow. And we minimize the total cost where the cost is just flows times the cost of the flow through the edge. So this is how the minimum cost, maximum flow is defined.

And the next step is then, as we said, just a subtraction. So when we subtract, we can, again-- what we need to observe here is-- so this is interesting thing to prove, let's say. If we subtract saturating flow, the expected result is a cycle, and no flows from source and to sink.

So to prove this we say, OK, the balance in the new graph, in the resulting graph is what? Is the new flows, the new capacities of the edges. But the new capacities are all capacities minus the flow. And if you rearrange this a little bit, you see, this is the original balance minus the inflows and outflows of the saturating flow.

But if you look at the balances, you will see that, this is all 0, so we have this on the next slide. So because we have a flow conservation principle, the inflows and outflows of the saturating flow for each node are 0. And the balance, because we made the network balance, is also 0, so we can easily prove that we have this situation here where every node in the resulting graph is balanced.

And now we only need to prove that this is empty. And we follow the same logic, and we can easily see that by construction, both source and sink result in 0.

So this is a little bit of mathematics behind it. Any questions at this point? No? Too simple. I should lift the level here for the next time I come. OK. So we looked at the formal definition, so what remains now is to look at the actual algorithm. So how do you-- how do you do-- how do you do this?

So we have here a pseudocode. We just follow the same stuff, so what needs to be done. We have to balance, find the saturating flow, and then subtract it. So to make it easier to read, so this is how we do the balancing. So we just define new nodes, source and sink.

Then we simply go, for every node in the original graph, you establish the net position and add edges. So if net position is less than 0, you add an edge from source to the firm. If the net position is positive, you add an edge from that node to the sink. And this is all there is to it, so there's nothing more complicated.

The next step is finding the saturated s-t flow. And here, we use the minimum cost, max flow as a standard element. So it's this condition here, if source is element.

So the way we constructed the balanced network, if the network is already balanced as an input, there is no source included in the network. So this is just to identify the already balanced network, so then you just pass. So only if there is a source, then you use the minimum cost, maximum flow.

So for any one of you who will be tempted after this lecture to try this by themselves, so the easiest way to do is use Python. NetworkX is my preferred library. There are several minimum cost, maximum flow algorithms available. They all perform well, but not equally well. So the difference is shown when you try it on a really large networks. But for toy models, everything will work. Yes?

AUDIENCE: Yeah, so how do you know-- I mean, the costs that you pick, you said you're going to pick uniform costs everywhere. In a real application, how do we interpret those costs? Are those the costs of the late payments that could arise from a cycle like this? And how do you know what they are?

TOMAŽ FLEISCHMAN: Yeah. So we go for cost of one as a basic. So you can imagine-- so what are we actually doing? Normally if you have a network of obligations, you expect everything will be paid. But this is not a fact of life. So what we do, we assume there is no money.

And then we introduce this imaginary money through source and sink, and we say, let's use-- let's make as little transactions as possible with this imaginary money and throw this away. And what remains is what clears by itself. And having equal cost of one just means that we really made the flow as small as possible.

If you introduce costs for whatever reasons-- for example, you want to play with risks, so you say, OK, not all obligations have the same risk, then the solution will optimize for that situation. But then it depends. So you might do this like an economist, so with risks, but then there are other options. For example, maybe there is some social benefit, like food safety, and I want to put a cost on how much does this contribute to the community food safety, and then you get a different solution.

AUDIENCE: So you can allow for arbitrary costs?

TOMAŽ FLEISCHMAN: Yes.

AUDIENCE: Then you can then have a backstory?

TOMAŽ FLEISCHMAN: Yeah, but then-- this is another lecture, then. So how do we do this? Yeah. Any other question at this point? So my experience is, if you want to truly understand this, it takes like an hour or so to do it in Python. It shouldn't be-- it's easy.

So the last thing to do, then, is simply to subtract, and here, we just go-- we say, OK, the result is an empty network, and then we go through the original network and subtract the amounts that were returned by the minimum cost, maximum flow function. And this is how we get-- how we get to the result. Any more questions on the algorithm side?

OK, so this was simple. It was just Alice, Bob, and Charlie, but we went through everything, so now the question is, what happens if you put this into a real-life situation? And I will show you here the setting on an empirical network that we have from Infocert.

So Infocert is an Italian company. They do tax compliance. So for you-- in Italy, they have VAT. And when you issue an invoice, you have to report to the tax authorities that you issued. And the process is so complicated that firms decide to outsource the processing, and Infocert is one of such invoice processing firms, and this is why we were able to get high-quality data about the invoices among the companies in Italy.

So this is how the Italian network looks like. So this is just part of the data we have. We have 45,000 dots on this, so every dot is a firm. The edges between firms are transparent because otherwise, it would be a black blob, but you see here in the center of the network some blackness from the edges. So there are roughly 2 million edges in this figure here.

So on the right side, it's just aggregated. So according to the firm size, you see that, by far, largest group of companies are micro-companies. Then you have small, medium, and large. And the largest transactions are among the large firms. So these blue dots here, basically, act as major hubs for the network.

But the traffic goes among all of them. So there is no hierarchy here, really. So this is our network, and we ran the MTCS algorithm on this. So how do you do this? You introduce source and target nodes. We have a source target. We do the edges, and here is just the illustrations of what you saw earlier with the small example, Alice, Bob, and Charlie, it's now-- this is how it looks like on a really, really large network.

The same algorithm is applied, and these are the results. So how heavy is the remaining network? So we see that the weight of the remaining network on average is around 11% of the initial depth. So these are the results of monthly clearing over two years. And, OK, this is a 95% accuracy rate. And we have some outstanding values, but not much.

So we see that things are pretty stable. And it's interesting that they are stable even in this two-year period. 2019 and 2020, we had the COVID crisis. And it turns out that COVID crisis, yes, it decreased the amount of business, but the structure of the business remained the same. So the structure is quite stable.

We also see that micro-firms, those little dots on the edges on the network, do not gain much. So the major beneficiaries of this are a bit larger firms, small, medium, and especially large firms. The interesting thing is, if you look by the net position, whether you are debt or net debt or net credit or we see that the net creditors benefit more. So actually, the mechanism works from the business point of view. It works like a accounts receivable collecting service. So it's very good. It helps collecting your receivables.

Now the question here is, how do we understand these numbers? Is this a lot or is this a little? So it depends. So when you say 10% to business, it's always significant. So it's important. So it's not like we don't care. But still, it looks a little bit small. So why just 10%? Why just that much?

And here at this point, I would just like to point out some topological features of the network that actually prevent the mechanism to have a different result. And this feature is the scale-free topology. So I don't know how much you know about this. I don't want to put a lot of thought into this, just very roughly. It is all about power law distributions.

So here, we have-- this is what economists like, these whisker plots. We have amounts on the invoices that were issued. And we see, by the firm size, the amounts can be as low as 1 euro or in millions of euro, but millions are outliers. So the average amount on the invoice is relatively small. So 183 euros or 332 for large firms. So the average amounts are very small.

But what is interesting is the distribution. So this is what network scientists like. So we have here, on the x-axis, we have the amounts in a logarithmic scale. And on the y-axis, we have a probability of such amount appearing in the sample. And you can see that the large amounts are very, very unlikely, but the distribution, the probability distribution follows the power law. And you have this typical long-tail distribution.

And the problem with this is the following. So we have a cycle, Alice, Bob, and Charlie, but the most likely amount on this cycle, the most likely amount is somewhere here, something small. And this basically, then, acts as the kind of a brake for larger amounts in the clearing.

Interesting thing, this is-- it's also about how the firm is connected. So here again, we have a number of business partners for different firm sizes, micro, small, medium, and large. And you see that-- yeah, OK. Here, it's more diverse than with the amounts, but when we look at the network science picture, when we look at the degree distribution, we, again, see a very, very specific power law distribution of the probabilities how many partners a firm has.

And this is, again, working as a brake because it is highly likely that a firm in a cycle will have only a few connections, and this is basically constraining the amount of flow that can go through this node. So this is why we somehow cannot go above this 10%, 11%, 12% in the network. So it's a purely topological stuff. OK, any questions on this?

AUDIENCE: So the fact that this is a power law is coming from-- do we know why?

TOMAŽ FLEISCHMAN: Yes. It's coming from nature. It's life. So the mechanism that drives this is preferential attachment rule that says when something grows, a network grow. When something new comes to the network, it is more likely to connect to already existing strong player than the one on the fringes.

AUDIENCE: So any growth process that has this feature--

TOMAŽ FLEISCHMAN: Yeah. I sometimes play with this. So I have a nice slide, I didn't include it here. I have a network of village community economy in Kenya and protein interaction in yeast. And then I ask which one is which, and nobody knows, because it's the same mechanism. So it goes really from nature.

But we need to do something about it. So we cannot just say, OK, this is how it is, let's do nothing. So it doesn't work like this, and this is how we arrive to liquidity injection.

So yeah, OK, we need to fight the forces, we need to fight the topology, and we do this by introducing engineered obligations because liquidity is something we made up. And let's start with Alice, Bob, and Charlie, my favorite players.

So now they have a new player, an engineered node. It's not a firm, it's a concept. It can be a bank. It can be something else. Mutual credit, crypto, whatever you want. The point is, it's engineered.

So we have this Alice, Bob, and Charlie cycle. And Alice, she is in a possession of something that is accepted. So Charlie is happy to say, I rather have two of those than nothing. So this is what the dashed line means. And the full line means I have it.

And the whole use of the algorithm of the MTCS on such an enhanced network where we introduce-- where we engineer the liquidity into the network is the same as before. So we include source and target, and we are looking for the saturating s-t flow.

When we remove the saturating flow, this is the solution. And what does this solution mean? So it means that Alice was able to pay-- or resolve the debt of 2 with Bob, Bob resolved a debt of 2 with Charlie. Alice used her 1 of whatever the money was. Charlie stored 1. Charlie doesn't owe anything to Alice anymore because it's 1. And everything that remains of this network would be, if you look at here, 1-- Bob still owes 1 to Charlie.

And because this is it, Charlie was unable to store 2 units on this new liquidity source, but just 1 because there was not enough liquidity engineered in the system to clear everything. Any question on this concept? No? OK.

So how does it work if there is more than Alice, Bob, and charlie? So the same as before. We've seen this picture. So we just say source and target can be joined, and they represent the new engineered source of liquidity. And we did experiments on this by not providing enough liquidity.

But because the sample is large-- and I didn't want to complicate stuff, we said that everyone has the same restriction in the liquidity. So I either have nothing or I have 1% or 2% or I have 100% of whatever I owe as my engineered liquidity.

And we set this parameter how much? we set it as a gamma, so in an interval from 0 to 1. So this is how the experiment is set up to show the effects of liquidity injection. And these are the results.

So, here on this graph, we have-- on x-axis, we have gamma. So how much of the indebtedness was created in the-- engineered in the liquidity source? And here on the y-axis, we have how much debt in the network was actually cleared.

And we see, at 0 gamma, we already start somewhere. It's not at 0 because we have these cycles with no liquidity clearing. But then as the gamma increases, it first starts quite steep, and then it goes like 45 degrees, and at the end, the efficiency of adding more liquidity is not so obvious, but when you provide everything, you also clear everything.

But the results in the middle are quite interesting. So for example, if you engineer 20% of required liquidity, but you clear close to 50% of all indebtedness. So there are huge network multipliers effect. And this line, it looks like a line, it's not really a line. These are 12 months combined in a 95% confidence interval, and this just shows that the structure of this real-life network that we observe is really stable even from the point of view of liquidity injection.

Here on the right side, you have the same line, just broken down to the sizes of the firms where you see that small firms are really dependent on-- or micro firms are really dependent on the liquidity that is engineered into the system, while larger firms benefit from this kind of stuff even more.

Let's go back to where we started. So how does this reflect into the late payment? So we started with the late payment as a problem. So the red line here, late payment. So depending on how much liquidity we engineered into the system, at the beginning, if the assumption is everything that was reported is late, you decrease the late payment a little bit, but as much as you increase the involvement of the liquidity, the late payment as a problem drops significantly.

And here are the contributions. So the blue line is how much liquidity was cleared in several steps through the network. So it went from Alice to Bob to Charlie to Dave and so forth. And the green line is the use of liquidity where only one hop was done. It just went from Alice to Bob and that's it. It's just the breakdown of the network effects.

Now, if we try to measure the value added of the mechanism, you see that it is clearly a mechanism when there is some kind of crisis. So when liquidity is short, the use of this mechanism produces the biggest effect. And we have seen this in the case of Slovenia, in the war economy, and then the COVID crisis. But then you can apply this differently.

There are regions that are liquidity constraints. Everything is fine in the economy, but you have regions. And so it makes sense to apply these kind of mechanisms in places where liquidity-- or general economic situation is bad and you want to do something to kickstart the stuff. Any questions on this point?

So why is it important to observe this kind of stuff? So the whole economic life is about transactions. So we transact. So what we are observing here is to collaborate on how do we resolve the economic situation by collaboration.

And we can go far if we want. So here, I'm showing a concept, but it's made on the real data. So this is made on a data from the Italian community from Sardex from Sardinia. They call the system-- they have a system of a mutual credit called Sardex, and I made this on their data.

So what we have here is the real obligations between the firms. So the firms are all these dots. And we engineered three different liquidity source, not one. Why just one? They can be different. And in a real-life example, this could be bank deposits, it could be mutual credit, it could be cryptocurrency, it could be many other things. Vouchers, for example. Whatever are people willing to accept to discharge the indebtedness works.

And the interesting thing, is when you combine these liquidity sources, you are increasing the amount of debt resolved, but even more interesting result is because of the-- because algorithm follows the flow conservation principle and follows the principle of balance, there is no flow from one liquidity source to another.

So basically, you can have a really different types of liquidity, but they will not influence each other. So there is no exchange from Bitcoin to dollar to mutual credit, something like that. Yeah?

AUDIENCE: Doesn't that make it less efficient? It feels like if you have different types of liquidity that you cannot-- that aren't exchangeable with each other, it seems like it will be more costly to resolve--

TOMAŽ FLEISCHMAN: No. It resolves more cycles because what happens is, you have a flow that goes from here to here to there and back, but there is no exchange. There is no trade. The interaction happens based on the obligations within the network.

Exchange without the trade, this is a tough one, but we are actually working on this. So it's a new look on this stuff.

AUDIENCE: Like one Bitcoin goes out of there, and then how does it go into-- how does it turn into $1?

TOMAŽ FLEISCHMAN: It's a flow conservation principle. Bitcoin is a Bitcoin, it goes nowhere. It's always here. Because all the outflows equal inflows. Bitcoin is a Bitcoin, it stays here.

AUDIENCE: Right.

TOMAŽ FLEISCHMAN: Bank deposit is a bank deposit. But this is a bank. So let's say these are all banks. These are all banks. Dollar on your bank is not the same as dollar on my bank. Dollar on your bank account, it stays within the bank, it goes nowhere.

AUDIENCE: Sure.

TOMAŽ FLEISCHMAN: Yeah.

AUDIENCE: But it will be-- will it be more efficient if banks can clear among the banks?

TOMAŽ FLEISCHMAN: Yes.

AUDIENCE: So it will be more efficient here if there are additional exchange over different kind of asset?

TOMAŽ FLEISCHMAN: So when you introduce-- so this is kind of a bias. So all economy is based on the idea, markets are everything. So what we see here is the exact opposite. Collaboration is everything. So for all practical reasons, none of this system is good by itself. So the message is, why not take the best from both worlds?

There is no need to compete for everything, and there is no way to arrange some kind of dream system where collaboration resolves everything. But certain amount of collaboration helps. And this kind of system, this is a very neutral system. It doesn't actually break-- it doesn't interfere with the competition. It doesn't interfere with the market. But it introduces wider possibilities for coordination. So this is why it's important to study this kind of stuff, and to see where we could implement it.

So the class is about the blockchain, so I'm coming from Informal Systems. We are a blockchain company. We are starting a new company. Now it will be called Cycles Protocol, and what we will deliver in next few months is something based on this idea.

Let's use blockchain as a single source of truth to create the networks. So the users, firms, individuals, they will encrypt their-- we call this intents. So I want to pay someone, so we have an agreement that there is a debt between two of them. So these are the intents. So this goes encrypted on chain.

Then these intents, encrypted, go to something we call a solver where they get decrypted, solved with the algorithm we looked just now, they are encrypted again and written back on chain. So we put together with the encrypted message, we include the attestation and the zero-knowledge proof that the MTCS process was done as it should be.

Verifiers can check this and write the results on chain, and this then enables the users to read the status from the chain. So the whole design is fully privacy-preserving. And this is exactly to the point to enable collaboration within the competitive environment.

So you don't want to expose the network to the market because knowledge of the network is power, but we want to have a network to be more efficient in resolving the indebtedness. So this is the next steps or where does this lead to.

So anyway, I hope you will find this interesting, and I'm just inviting you to follow us because we will deliver this stuff very soon in various forms. So thank you very much.

PROFESSOR: Do we have time for discussion?

TOMAŽ FLEISCHMAN: Sure!

AUDIENCE: I mean, I guess how new is this whole thing for the startup? Like, how new is like this technology?

TOMAŽ FLEISCHMAN: This technology? So in terms of privacy-preserving, it's brand new. Because what is used right now for Zcash, for example, is not useful for what we do because we need this-- so the encoding has to be done in a way where multiple parties have access to data, which is not what is normal for the shielded pools design. So this will be very new.

When it comes to the idea, let's clear. It's somewhere in the European renaissance. So the first documented clearings-- collaborative clearings within the groups are from ancient merchant fairs where it worked like this. So you would go to a fair with your stuff, whatever-- rugs, for example. And you would trade there. And then you would go back. But travel to the fair and back, you go through forests, so having cash, gold coins would be a very bad idea. So they didn't do this. So how did it work?

So you go to a fair, you trade, you write notes to other merchants. And at the end of the trading, there was a banking fair, and banking fair would last for two days. And it was actually a choreography.

So merchants were invited. And then they said, let's commence. And I owe you-- I would hold your left shoulder with my right arm, and we go like this together until you find someone. And when we are holding a cycle-- oh. Then we go, sit at the table, put our notes down, reduce, write new ones. And this choreography would last like for two days. And you go home with just new credit notes. No money.

And then this was-- in the banking world, the most known example is our English banking clearing clubs. And just-- there is a funny anecdote. A letter from one of the banker explaining how he is going to these kind of clubs. He said, oh, it's good. I only take enough money for food and women, everything else gets cleared. So this is--

So it's an old-- it's ancient, ancient method that we stopped using because it's difficult to implement. So you cannot have a choreography here in Boston, for example. So how would this look like? But with privacy-preserving blockchain technology, we can revive this stuff, and it's very useful.

PROFESSOR: So maybe-- can go back to the liquidity thing? Because--

TOMAŽ FLEISCHMAN: Yeah.

PROFESSOR: We've done more work than you had time to present. So let me just say that the one way the engineering works is to solicit from the individual participants liquidity that they're willing to put in escrow.

TOMAŽ FLEISCHMAN: Mm-hmm.

PROFESSOR: Once it's in escrow, it can be used to clear the system. So Alice was the one who was in a net debtor position, so she owes the money, and she could say, I'm willing to contribute a quarter or maybe even half of the money in advance to clear my own debt.

And so all the net debtor nodes are contributing a certain fraction of the net debt. That's what the gamma was in that slide. And initially-- and that's what you were presenting, it's obligatory-- I mean, we're running an experiment from gamma equals 0 to gamma equal to 1.

When gamma is equal to 1, there's no network effect because they've all contributed enough liquidity to pay their own debts. And when there's 0, then nothing is happening. So something in between is what Tomaž is showing where we get, especially initially, you get a very big multiplier.

TOMAŽ FLEISCHMAN: Yeah, but in real life, it's always somewhere in the middle. It's never nothing, it's never everything. And just--

PROFESSOR: Another thing to say, I think, is that it's not as though Alice got something for nothing because Alice is left owing 1 unit, and Charlie over there is the guy with the surplus, so he is owed 1 unit. And they could take some of the money in escrow as payment for-- as the end of underplaying that outcome. But the other thing that we've been exploring is to solicit the liquidity.

TOMAŽ FLEISCHMAN: Yeah.

AUDIENCE: Which, again, you didn't have time. Maybe you could say--

TOMAŽ FLEISCHMAN: Yeah, it's an interesting concept. So if you remember the English Association of Chartered Accountants. So many firms choose not to pay, although they have. So what is the logic? I have money on my bank account. I'm a powerful market player, so I choose not to pay. The effect is, my balance sheets are a bit larger and maybe my credit score in the bank looks a little bit better, so I choose not to pay.

So what can we do to encourage these firms to pay early? And one of the ideas is, OK, let's offer them a premium. If you commit liquidity, you receive 1% premium. And 1% in one month in terms of interest rate is huge. So there is no way a bank deposit is worth as much as 1% premium.

On the other side, you have firms that are desperate for cash. They want to collect. Collection is expensive. What can they do? They can go to factor, for example, but how much does factor cost? Factor will take 10% easily. 5% if you are super lucky and you have good relationships and everything, but it's a lot.

So maybe I'm happy receiving a 1% discount from the network. And this way, we create a market for liquidity. And it is possible to do this on different discount and premium levels. So I'm willing to engage with my liquidity at 1% or maybe at 2%, and I'm willing to take liquidity for 1%, 2%, 4%. So you can make a whole range.

But when you do a range, there is always something remaining, and this reminder can be then given back to participants as a cashback, let's say. So the options to engineer here are countless, let's say.

AUDIENCE: So practically, I imagine there will be a network effect which will be very strong in this circle, right?

TOMAŽ FLEISCHMAN: Yes.

AUDIENCE: So how-- I mean, think about how you want those small firms to adopt this technology, how that will happen now they are using this traditional system. So how do you get them to adopt this new blockchain--

TOMAŽ FLEISCHMAN: Yeah. The incentive to use. So the basic incentive is, it helps my collection process. So we have seen from the chartered accountants example, managing accounts receivable is a costly business. So the message from anyone who provides this kind of service is, we will help you reduce the cost of your working capital management, let's say.

But then in addition to this, you can add other incentives in terms of liquidity provision. But the major friction here is knowledge. People need to know that this exists. And in terms of administrative process, it is all about accounting. Basically, what we do is all about accounting. And the use of this system-- so the data is coming from accounting and the results should go to the accounting.

So I think the major friction for the adoption would be integration with ERP systems or accounting systems in general. And we as a firm, we are doing on this. Fortunately, stuff in accounting is quite standardized, so it should go well. Yeah.

And so it's interesting. So me coming from Slovenia where we use this for decades, nobody thinks about this, so it's not-- so it's everywhere. So every accounting package has a button. Report. That's it.

AUDIENCE: So one question I have in the context of instant payments, and you're worried about real-time settlement, is the algorithm that you described, there's no notion of timestamp. And you can think about It, as for each discrete period, you're running something like what you described, but in practice, the timestamps could slightly misalign, that's a motivation for liquidity lines from central banks.

But then how does this algorithm run when there is not really a single diagram, but different timestamps associated with each flow on each edge?

TOMAŽ FLEISCHMAN: You are right. You have to construct a network, and network happens in a certain time period. So the basic idea is that everything within a observed time period is equal. But we mentioned weights. So our cost of the flow.

So you can adopt different policies. You can put different weights on flows depending on, let's say, various things, like timestamps. It's possible. But then when it comes to how useful is this, do we do it monthly, like we did it with Italian data? Do we do it daily?

So it all depends on the network and the network topology. So similar, not the same, but very similar techniques are used, for example, in CHAPS. CHAPS is a real-time gross settlement system. And they do this every 15 seconds.

So it really-- so this is an engineering question. So not every financial system is the same. The dynamics, the number of participants, the number of transactions among them, size of transactions is different, and it's an engineering question.

PROFESSOR: So there's another aspect-- I don't want to overwhelm people, but just to point out the research possibilities here. This system that Tomaž has been describing respects the individual bilateral relationships that the nodes have.

There are alternatives. For example, like netting, you could have a rule that just says, for a given node, I'm happy if a certain fraction of my debts are offset with receivables that I have. And you could implement that, but you don't necessarily get the identical result.

TOMAŽ FLEISCHMAN: Yeah. The question, netting versus multilateral set of-- yeah. So let's do Alice, Bob, and Charlie. Hopefully this works. Yes, it does. So we have our Alice, Bob, and Charlie. The same network we used, 2, 2, 1. So now, how does the netting solution look like?

The netting solution in this example looks like this. Or let's say what remains to be paid after netting. So this is the netting solution. And let's do the setoff solution. So what remains after setoff is Alice owes 1 to Bob, owes 1 to Charlie.

So you can say netting is more efficient because there is less remaining. It's just Alice of 1 to Charlie. But there is a problem. The problem is, Charlie owes Alice originally, but now, after netting Alice, this is something new. It's a novation.

And now imagine larger networks. It can be that Alice has nothing to do with Charlie in the larger network, but now, all of a sudden, whoa, where does this come from? So basically, what netting assumes is that everyone knows everyone, and that everyone is equal in terms of risk, then it can work. So network netting typically works inside a very tight closed groups.

The setoff, we have just a reduction of the original depths. So there is no redistribution of-- or the structure of risk remains the same, it's just reduced. So a setoff is a pure risk reduction technique, while netting introduces innovation, new risks. And so this would be the difference.

And it's a quite important-- so especially in larger groups, like trade credit, where players don't necessarily know each other or trust each other equally, you cannot rely on netting. You have to use techniques where risk is not redistributed.

AUDIENCE: In trade credit networks, part of the problem is, indeed, the timing mismatch. And when some firms want liquidity from-- when others provide it. And that's why it was kind of curious. So you said monthly. Like, on what scale do you simulate?

TOMAŽ FLEISCHMAN: Yeah. The interesting-- so we are observing business-to-business networks. So the monthly time frame comes from the European business-to-business context. Because in Europe, we have monthly salaries, and a lot of firms issue invoices monthly for the business within that month.

This might be different in the United States slightly because you have this concept of weekly salaries and so on. So it really depends on this. So what you can see from the sample also is that in Italy, you still can observe a bit of a annual invoicing.

So the December, as such, is quite an outlier. And it's not just because December is like a Christmas time and people shop more and things like that, it's also because, at least in Italy, it goes into extreme in these cases. So issuing one invoice for the full year is nothing special there.

I hope you will find this interesting, and will lead you to explore this more in whatever you are studying otherwise.