# Distributed Ledgers

## Design and Regulation of Financial Infrastructure and Payment Systems



The title-level DOI for this work is:

doi:10.7551/mitpress/13382.001.0001

# 12

# Summary and Conclusion

Distributed ledgers have the potential to transform economic organization and financial structure, yet the subject is embroiled in controversy, hype, and lack of consistent terminology, as chapter 1 makes clear. Rather than get hung up on proper definitions about what a distributed ledger is and isn't, and how broadly or narrowly the term should be interpreted, we focus instead on the economics of what distributed ledgers can do by breaking down the concepts and analyzing the key individual components. We also compare and contrast the economic framework with the frameworks of the computer science and data management disciplines, clarifying the terminology and the technology where possible. Finally, we attempt to combine economics and computer science and forge new ground.

One of the ingredients missing from much of the discussion of distributed ledgers is the context and clarity in the application to actual economies. Chapter 2 thus begins with conceptual background on what economists mean by an economy, natural welfare functions to judge efficiency, and ideal systems of measurement. Many of these ingredients come together in the Townsend Thai project, which is featured frequently in examples in subsequent chapters, but this book is about much more than that.

Chapters 3 through 6 describe distributed ledgers in terms of the key, though familiar, component parts: ledgers as accounts, e-messages and e-transfers of value, cryptography, and contracts, including multiparty mechanisms. We put each in context and provide examples, emphasizing what each of these parts of digital ledger technology brings to the table, one at a time.

For the first component, ledgers, in chapter 3 we link the ledgers of cryptocurrencies to a statement of currency flow as a standard financial account and to currency as a balance-sheet item. Cash flow accounts in Thai villages are presented as an example; in this context currency flows are reasonably well measured and have been used in analysis, but of course, the idea is much more general. Financial accounts are a universal concept. The ledgers of DLT are then put into this context, taking an additional step which, in essence, is simply creating a common integrated database of cash flows across households and the associated balance-sheet entries. With this, one uncovers discrepancies, not unlike how the use of DLT can help in the reconciliation of trades. A related analogy: Standard financial accounts were an invention associated with double-entry bookkeeping, for more accurate measurement, a huge innovation at the time. Likewise, DLT enhances measurement across diverse parties.

Relatedly, the DLT ledgers could be used with transaction data to create standard financial accounts—namely, the income statement in addition to a complete balance sheet and complete statement of cash flow. These accounts are useful in analysis, and for policy, as illustrated by two timely examples: the impact of tariffs and regional isolation and drawing a distinction among various sources of liquidity, not simply cash. Likewise, forging a connection between DLT and financial accounts makes clear the even greater potential for DLT, though not yet realized—what we might term *consensus categorization*.

Chapter 3 concludes with a revealing summary of issues known from the science of database management. There are key impossibility theorems for decentralized systems: One cannot have all of consistency, accuracy, and partitioning. Yet systems that are periodically synchronized, and thus consistent, require costly communication and do not scale up, as at some point in the process of validation every node is connected to every other. This conundrum is sometimes not stated baldly or faced squarely. Trusted third parties do solve this problem, hence their appeal, but this centralization not only requires trust but raises the issue of data integrity and data security cyber-risk issues. One conclusion of this chapter is that hybrid systems with partial meshes, in which not all nodes connect to each other, may be best for many applications. An illustrative example is Lightning Network, which shifts small transactions to a cryptographically secure "off-chain" environment so that only large netting transactions need to be directly settled on the blockchain. Thus, although hybrid systems between the end points of strictly hierarchical database systems do emerge in practice (ignoring incentives and fully connected network meshes, which exacerbate privacy concerns), they are not necessarily a deliberate choice among the universe of possibilities. The economics literature provides clear examples of how costly connections among agents can lead to constrained-optimal partitioning (numbers in a cluster and how many clusters). A second conclusion of this chapter is that more work needs to be done on optimal design.

For the second component of DLT, e-transfers and e-messages, we compare and contrast in chapter 4 two countries, Thailand and Sweden, from high to low ratios of currency to GDP, respectively. For Thailand, there are large welfare losses to the current reliance on paper currency. For Sweden one worry, ironically, is the other way, at least in the near term—groups vulnerable to the disappearance of currency. We then

feature an innovation, e-money, for a country in between, Kenya, where M-Pesa has generated large welfare gains. This is done with a trusted third party, Safaricom, and without cryptocurrency, though the ledgers of Safaricom resemble tokens in some ways. The key is allowing easy ins and outs from Kenyan shillings to cell credits, facilitating transfers from urban migrants back to villages, for better risk-sharing and poverty reduction.

Some caveats bring us back to the featured issues of this book. For one, in the Kenyan context with Safaricom, trust is less obvious than it might seem when taking into account the larger financial system, with banks indirectly holding accounts, the Kenyan shillings that were turned in for e-tokens. For another, a relatively undeveloped part of value exchange is the infrastructure for liquidity, a common shortcoming in Kenya with its dual currency system and in New York financial markets in securities and central bank reserves. Ironically, most exchanges for DLT cryptocurrency that provide liquidity in and out of fiat monies and other cryptocurrencies rely on traditional technology with trusted third parties and not on DLT. That is, the tokens that feature absence of third-party trust rely on such trust for market exchange. The exchanges that are DLT-based are currently slow and illiquid. This highlights an area where further work is needed. However, we will not fall into the trap of insisting that all parts of a system be decentralized, as some do. That is, money needs to be decentralized, hence cryptocurrency, or exchanges need to be decentralized, hence digital assets. To the contrary, hybrids may well be desirable but, again, what is striking is the surprising lack of work in this area of overall optimal design of institutions and markets for a given economy.

E-transfers naturally raise issues related to cryptography and verification of messages if there is not universal trust in a single third party. As covered in chapter 5, ideas for overcoming

lack of trust in parts of a financial system are ancient, dating back to Mesopotamia, where sealed clay envelopes containing tokens were used as a manifest of shipments from supplier to destination to guard against tampering in between. Similarly, in medieval times, tally sticks split into pieces were used, with the stock circulating as money that uniquely matched the foil held by the borrower, checked on redemption. More recently, but before Bitcoin, we have secure multiparty communication, IBM crypto express cards, public vs. private keys, and zero-knowledge proofs and protocols. The chapter on cryptography contains an important discussion of the various validation systems used, from Bitcoin's proof of work, to proof of stake, to federated decentralized systems of trust, which require a listing of trusted nodes, but a list that is different across different nodes. To some, the term *distributed ledger* refers uniquely to only some but not all of these decentralized validation systems, though that point of view seems excessively narrow. More generally, but still narrow in the view of this book, the consensus part of DLT is often taken as a defining characteristic. But one can easily lose sight of the underlying economic purpose of a system, and why messages need to be validated in the first place, if at all, other than to claim the title of being decentralized.

This brings us to mechanism design in chapter 6 and the fourth economic component of distributed ledgers. For contracts, we highlight from standard contract literature the needed distinctions to get at the meaning of the word *trust* as in "trusted third parties," as featured in the introduction. These distinctions include full commitment versus limited commitment; contract theory allows the latter, but this does not preclude some trades. Likewise, from contract theory and mechanism design we can speak of incentive compatibility for actions and truth-telling for messages for initial and interim unobserved states. Lack of trust in this specific sense is crucial

in the design but is not an insuperable barrier. Related, it is key to prevent parties from pulling out or not performing. Default has remedies in collateral and/or reputation made explicit under long-term contracting. Reneging and a restriction to time consistency can be remedied with commitment. Problems of collusion and nonuniqueness in implementation have remedies in better design. We need to get specific across the various trust aspects, and contract theory helps us do that, especially in hybrid systems.

For the contract part of DLT innovation, we feature the key technical capabilities of smart contracts, highlighting the commitment in entering into the agreement and carrying it out, the immutability of its terms, and the conditionality aspects that help resolve the various trust problems, as agreed-to options are executed automatically as a function of the state. Smart contracts help to mitigate underlying frictions in the environment. Key concepts here also include states of the system and transitions, commands, validity consensus, unique consensus, notaries and nonunique consensus, multiple trusting or nontrusted notaries, public and private nodes, oracles, and broadcast communication versus selectively private communication.

A key point is that the various aspects of trust and incentives that come from mechanism design can be implemented on smart contracts and hence have implications for the messages, transfers, and recommendations that would be happening in real time. If messages and so on satisfy underlying incentive and truth-telling constraints, then the only reason to validate messages is because of faulty but not malicious nodes. This seems like a key distinction, but it is rarely made. Furthermore, depending on the contract, messages need not be sent of a domain of states, as in costly state verification, and past histories can be summarized by key statistics as in the use of promised utility and utility threats to collapse the dimensional of the problem.

Economists have had these mechanisms in mind, and so in some sense have taken smart contracts for granted without realizing it. However, DLT and smart-contract technology are more limited than it might appear at first blush, raising conceptual issues. The "planner" problem of mechanism design is a metaphor for economists, not taken literally *per se*, but in the context here is it clear that one must specify how contracts and multiparty agreements are entered into and validated and how they are executed over time. The doorkeepers and notaries involved with smart-contract execution, if single entities, are hierarchical features, though competing, nontrusting notaries conjure up the image of more decentralized systems. Decentralizing on Ethereum can be quite costly, suggesting that the writing of code, validation, and numerical computation be done offline. On the positive side, some of the underlying basics of mechanism design survive some substantial limitations, though the definition and nature of an optimal arrangement would change. A version of the revelation principle works with noisy messages, and with no communication at all, so one need not abandon mechanism design when facing the reality of imperfect messaging. Though there are impossibility results regarding consensus in the economics literature, there are also contributions on the effectiveness of multiple repeat messages and how iterations of decentralized validation can be truncated and achieve coordination. It seems some of these are not yet incorporated into consensus protocols.

Chapter 7 continues the discussion of "decentralization," highlighting some paradoxical results in mechanism design. Ledgers can be partitioned by the information that agents in a multiparty smart contract are supposed to have or, better put, not have. Here the contract node plays a crucial preprogrammed role, receiving all incentive-compatible communication but not necessarily broadcasting all incoming messages. Explicit randomization can be done at the level of the contract node

as planner, to ensure a gain from concealment and keeping secrets. Another instance: the delegation to a preprogrammed, third-party custodian to deal with private and public shocks that allow back- and front-loading in contracts. That function looks quite centralized. A final section of chapter 7 looks at portable physical tokens, privately held but voluntarily disclosed, as a decentralized way of recording key aspects of history, also implementable with partitioned ledgers. Multiple colored tokens can also have a role, though, again, in some environments we do not want to display full history. This part of the mechanism design literature is of course about design and does not appear to be incorporated into the computer science discussion of validation.

Chapters 8 and 9 go on to feature an example where innovation would allow large gains. As noted in chapter 8, in Thailand and across other countries in Southeast Asia, traditional formal financial infrastructure is currently extremely limited. From Asian Development Bank studies, we know there are large gaps in services for credit, saving, payments, and insurance. From the Townsend Thai project we know that informal risk-sharing is good for idiosyncratic risk. There are in effect village money markets replete with credit chains that resemble the sophistication of advanced country systems. But there are shortfalls in reallocating risk across villages and regions, and management of cash is inefficient in rural areas. Interventions can make a difference. A government village fund program had positive impacts on consumption, borrowing, investment, profits, and intermediation. Yet this was uneven. Those without kin did not benefit nearly as much, and villages remain largely disconnected from one another. The ironic virtue of DLT work in Thailand is that it is easier to start from scratch in implementing optimal designs, as gains are large and there are no coordination or consortia problems.

Innovations in this Southeast Asia context can come in two related forms. First, individual smart contracts are given as examples of how to take advantage of distributed ledger capability: escrow with nonbanks; savings products for automated deposit and portfolio management; and securitized waterfall payments along the path of supply chains from buyer to seller to employee loans. A second form is contract competition with open access to providers and free entry, as in general equilibrium models with an intermediary broker sector. Here then we are combining the needed micro side (the contracts) with the more macro side (general equilibrium), incentives of participants in open markets, and whether or not this will work, drawing on theory literature.

Featured as an example of innovation is EvryNet, an intelligent financial automation operating system that aims to provide open-source banking services and financial contracts to unbanked and underbanked populations. An interoperable smart-contract platform enables not only traditional banks but also microfinance institutions and others to initiate and execute banking products and financial contracts. Contracts can be provided at competitive prices for computer memory (storage) and computation power. A rating system tracks the performances of providers.

Chapter 9 features two payment systems on distributed ledgers. One, an experiment, is Project Jasper from the Bank of Canada, which is designed for domestic interbank payment settlement and is focused on payments. The role of the notary node is played by the Bank of Canada. Phase 2 of Project Jasper appears to be one of the first instances of a central queue within a DLT platform for payments, for matching and netting. A matching algorithm on a distributed ledger platform employs the language of states and conditionality. Netting promotes funding efficiency and smoother intraday

payments flow. In Project Jasper, parties see only their own activity, a traditional point of view, while the role of the unique notary is played by the Bank of Canada. It is a sophisticated multiparty platform taking advantage of smart contract possibilities.

The second featured payment system is Lightnet, creating for money transfer operators (MTOs) in Southeast Asia a highly liquid, decentralized settlement layer on a permissioned blockchain. Remittances in fiat money in Southeast Asia have transfer fees currently at 7.1%. The high transfer fees are partly the result of legacy technology in the formal sector and limited access to formal currency exchange markets. The Velo network is a settlement layer that avoids direct transfers of fiat money yet enables participants to conduct cross-border transactions efficiently. An optimized liquidity-management layer will efficiently search for offsetting cross-country balances to allow for expedient clearing while minimizing risk. MTOs can be viewed as agents with varying underlying balance sheets hit by their customers' needs for trade and hence credit and insurance. Some MTOs are already engaged in a contractual bilateral relationship, with both of their fiat credit lines supporting their financial obligations bound by a bilateral agreement. The point is that Lightnet can allow the implementation of constrained-optimal contract arrangements implemented on distributed ledgers, as envisioned in the examples on credit and insurance described earlier. That should not appear as abstract and otherworldly, as it has its real-world counterpart here. Notably, economic outcomes under contracts, data acquired, and data disseminated are all endogenous and codetermined.

Chapter 10 addresses some regulatory issues or, better put, some regulatory opportunities. Distributed ledgers mitigate, if not eliminate, bank and market runs through time-stamped and immutable records of the histories of transactions. For that last result on runs, caveats from computer science come

into play—namely, latency on networks—but potential remedies are noted.

The limits of contract competition are noted here, too. Competition among providers can fail to complete the financial system because of unexploited complementarities and lack of coordination in a Nash equilibrium, even in environments in which all players are tiny such that no player has any mass. But there is a message in this for regulators. Traditional categories, separating insurance from credit, for example, may block necessary innovation. Relatedly, a key issue for regulation is the time line of competition and when to impose exclusivity; for example, free competition ex ante could be fine, but with exclusivity and restrictions on trade ex post. Regulators should not come to this subject with the point of view that "anything goes" or that everything is bad.

Other model environments make clear that consensus and public ledgers can be needed for coordination and prudential regulation. A theorem in economics on the impossibility of decentralized monetary exchange is quite relevant. Knowledge of identities of agents, histories of trade and payments, and initial excess demands are needed for implementation of optimal allocations, not simply pairwise knowledge of those contemporaneously matched but also from others with whom the contemporaneous set of matched traders, the payment parties, have not been matched previously. Put simply, the history of trades needs to appear on the common immutable ledger. Implementation of a Walrasian optimum in a decentralized way is both forward- and backward-looking. One has to know where the system should be headed, and the remaining options in the future to get there, hence what trades have been accomplished in the past and what trades are needed now in order to make this feasible.

For another example, circulating private debt is the medium of exchange for contemporaneous transactions in both short-term

noncirculating securities and consumption commodities. Yet there can be many equilibria that achieve the Pareto optimal target; for example, either all the debts that are allowed to circulate could be issued by initial parties in one of the two locations or by the parties in the other, second location. But by assumption, in the example environment of the informationally decentralized model, there is no way for traders in one location to know what is going on in the other. Mismatch is likely, with too much or too little debt issued, with resulting crashes later. These conflicts and the need for coordination are likely to arise with multiple cryptocurrencies.

Chapter 11 addresses head-on issues surrounding cryptocurrency and tokens. The subject inevitably brings controversy, as advocates and detractors of Bitcoin argue about whether or not cryptocurrencies are monies and whether they should or should not be allowed to be alternatives to fiat monies. It is here in particular that monetary theory and mechanism design can be brought to the table, though typically this is not the case. First, as a reminder to the reader, chapter 11 reviews the various objects, including private objects that are considered money even under standard definitions, though the best measures look at velocity and frequency of use in exchange. Typically, there are multiple media of exchange in use in a given economy, so from that standpoint it is a bit surprising that detractors do not see room for more. Nevertheless, the idea persists that there is room for fiat money only and not cryptocurrency. To the contrary, tokens can be a useful tool in the implementation of mechanism design problems, and there one does not want fungibility between the two. From monetary theory we know that models with endogenously valued money can have good equilibria, Pareto improving on autarky, even though money is an intrinsically useful object—a sharp reminder to central bankers who make this charge against cryptocurrency as if to say crypto values have to be zero and

the rest is speculation. Actually, some bubbles can be good, and there is a companion empirical literature. Nevertheless, both equilibria with value money and equilibria with valued tokens can suffer from indeterminacy and many other potential equilibria, not as extreme as autarky but with inflation and the value of fiat money, or cryptocurrency, going to zero or with fluctuations, the latter a potential explanation for what we are observing for cryptocurrency exchange rates. A way to ensure fiat money has value, with a positive lower bound, is to require its use in the payment of taxes or require its use in exchange by fiat. There are many models that follow those lines for modeling fiat. However, the same argument applies to cryptocurrencies that are utility tokens, required as part of business operations with smart contracts, even if at the same time these tokens are traded on exchanges. Finally, fiat and crypto monies can coexist and play useful roles, if the use of one only—say, fiat—would leave economic gaps and room for the other. One type of stable coin achieves value by backing, similar to the concept of narrow banks with 100% reserves. The discussion of other stable coins seems to suffer from limited knowledge of the monetary-theory literature or lessons from fiat exchange rates managed by central banks. An algorithmic digital reserve system with smart contracts on distributed ledgers can offer not only commitment, it could implement optimal activist token policies suggested by the monetary models, armed with transactions data from the distributed ledger.