

8. Signals and incentives in blockchain applications

Cameron Harwick

1. INTRODUCTION

What made the invention of Bitcoin in 2009 so revolutionary was that it managed to combine three features that had been previously thought mutually exclusive: (1) a distributed network, (2) on which consensus about the state of the network can be reliably reached, but (3) *without* trusting any particular node on the network to relay information honestly.¹ For electronic money in particular, an authoritative central party had been commonly understood to be necessary to prevent fraudulent spending. To reliably achieve decentralized consensus *without* such an authority is not merely a feat of software engineering, but a feat of *mechanism design*. It is not enough that the network merely exists; it must be structured in such a way that participants have nothing to gain by relaying dishonest information. On a network where consequential actions are taken on the basis of that information, and with limited ability to punish dishonest participants, this task is far from straightforward.

This chapter asks: under what circumstances is it worthwhile to act on signals given by others? – this is the fundamental problem not only of blockchain payments, not only of payments in general, but also of communication systems more broadly, including language. If you can't trust the veracity of the information communicated, and especially if it pays others to deceive you using that information, it will never pay to join the network, whether a linguistic or an exchange network. This problem has been solved in a variety of ways by different communication systems, but the technical limitations of blockchain networks will shed light on the necessary and sufficient conditions for such trust. Importantly, these limitations give us a sense of exactly how far *information* may be trusted without trusting the *source*.

The situations where this is possible are those where decentralized blockchains have the clearest use case as a “technology for economic coordination” (Davidson et al. 2018) and are indeed where the most rapid progress has been made in the past decade. The situations where this is *not* possible have

seen stalled progress or, at the very least, hybrid models that *do* incorporate information trusted by virtue of its source, whether through centralized permissioned blockchains or through “oracles” allowing outside trusted data to enter blockchain ledgers. Accordingly, a clear sense of the signal verification problem solved by blockchain technology not only explains the pattern of development so far, but can also guide the development of future applications: payments, finance, data storage, and anywhere else decentralized ledgers might in the abstract be a useful tool.

2. THE PROBLEM OF CHEAP TALK: A PARABLE

Before coming to blockchains, it will be useful to illustrate the general problem with a parable. Suppose you, a risk-neutral reveler, are offered two boxes at a carnival booth. One has a payoff of 1, the other is empty. You have the choice to walk away, or you can pay some fraction π of the payoff to choose a box and take the contents.

There is some threshold price π^* below which you will take the gamble, and above which you will not. Provided you know the game is fair,² and have no priors as to which box contains the payoff, $\pi^* = 0.5$. More generally, if you believe with probability $\Pi \geq 0.5$ that you know which box contains the payoff,³ $\pi^* = \Pi$.

Now suppose you are told (by whom does not matter yet) the payoff is in the first box. How this affects your π^* depends on how much you trust the information. By Bayes’ rule, if we call σ your assessment of the likelihood that the information is accurate, $\pi^* = 2\sigma\Pi$ (see the derivation in the appendix). If you start with no priors so that $\Pi = 0.5$, your willingness to pay depends *entirely* on your trust in this information: $\pi^* = \sigma$.

Consider now several possibilities as to the source of the information. If the source is, say, a bystanding heckler, you know he doesn’t know the answer any better than you. So $\sigma = 0.5$ and your estimate – whatever it was – is unchanged.

If the source is the proprietor of the booth, more interesting problems arise, and the credence you give to the information depends on your assessment of the incentives facing the proprietor.

1. The proprietor has an interest in keeping the payoff. You know he cannot reliably lose money and stay in business, so you must assume he is trying to deceive you. But like Vizzini’s dilemma in *The Princess Bride*, answers that negatively correlate with the correct answer still provide information, so the possibility of “reverse psychology” means you should place no weight on such a remark (i.e., σ is also bounded from below by 0.5).

2. The proprietor has an interest in his own reputation. Your knowledge of his dependence on this reputation influences the credence you should give him. How anonymous is the fair he is set up at? How likely are you to return in the future? What are the consequences for him if you leave feeling cheated?
3. Banter and cheap talk may be part of the intrinsic fun of the game, in which case it is not intended to convey information in the first place (although it may do so inadvertently).

Note the difference between (1) and (3) on the one hand, and (2) on the other. In (3), the proprietor loses nothing from conveying false information to you. In (1), the proprietor even *gains* from conveying false information if you should take him at his word. In (2), on the other hand, the proprietor stands to *lose* from a lie. And only in (2) does it make sense to give positive weight to his claim and assign a $\sigma > 0.5$.

In this example, the proprietor's remark to you is a *signal*: it is intended to convey information to induce you to take a costly action, costly because choosing one box means you give up the other. You have a choice of how much credence to place on the signal. You place credence on it to the extent that you can verify that the source of the signal will face or would have faced costs for deceiving you. Importantly, though, if you place *no* credence on the signal, you do not do the reverse of what it indicates; you simply ignore it.

This is the fundamental problem of communication systems in general, and the problem of language evolution is formally identical. Consider now the prospects of a pre-linguistic hominid who may or may not give credence to meaningful voluntary utterances by others in his clan. One who *does* give credence can be exploited, say, by being falsely informed of an approaching predator while feeding on a choice fruit patch. It is notable in this respect that, though their spatial and social cognition is relatively sophisticated, chimpanzees have no *verbal* ability whatsoever: they will not give credence to any *voluntary* vocalizations proceeding from other chimps. All meaningful vocal signals, including alarm calls, are *involuntary*, meaning they cannot reliably be faked, and thus can be trusted (Knight 1998). Thus, the receptivity to signals – the willingness to give credence to them and take costly actions based on them – is logically prior to the intentional production of signals.

Money, too, is such a signal, intended to induce you to part voluntarily with valuable goods or services. It must, therefore, be costly to falsify, if people are to place any credence on the signal – that is, to accept the money. This costliness has been ensured in a variety of ways in the past. The following section compares traditional methods of doing so with the novel cryptographic solution pioneered by Satoshi Nakamoto in 2008.

3. THE SIGNAL VALUE OF MONEY

A decentralized *electronic* payments system entails several pieces that an otherwise equivalent *physical commodity* standard does not. In order to see the applicability of the parable from the previous section, this section breaks down the problem of monetary payments into its most basic parts, with language intended to defamiliarize what exactly it is that monetary systems do.

Money, again, is a signal. A signal conveys information about the state of the world, a state upon which you might like to condition your behavior. Money in particular conveys an expectation that you will be able to obtain valuable things later with it. This expectation, if valid, is what makes it worthwhile for you to part with goods or services voluntarily.

In order for it to be rational to trust a signal, it does not have to be costly to *emit* the signal; it simply has to be costly to *falsify* it, that is, to indicate a state of the world other than the actually existing one. This is why, for much of history, money consisted in a scarce physical object of some sort or another, one that is both (1) difficult to produce and (2) easy to verify (Alchian 1977). So long as these conditions roughly hold, the signal value of money can be trusted without having to trust your trading partner. Ease of verification is important because a more easily counterfeitable money reduces the certainty of the expectation of being able to dispose of it later. Similarly, difficulty in production is important because it is precisely this difficulty that ensures against false signals. You will be willing to take actions in response to a signal *only up to the point* where the value of your action equals the difference in the cost of generating a true signal and a false signal. To do otherwise is to make yourself vulnerable to exploitation; for example, selling goods in exchange for money that you cannot spend, or can only spend at a less-favorable-than-expected rate.

For a scarce commodity such as gold, this difference in cost – and thus its value as a monetary signal in equilibrium – is the marginal cost of producing it. In other words, equilibrium obtains when the net value to be gained by creating a new signal is no more than the net value of acquiring an existing one (say, through selling goods or services). In the case of gold, for example, its value would be the cost of mining one additional unit (White 1999, ch. 2). Thus, advances in mining such as the cyanide process lowered the value of the signal, and reduced the willingness of people in the gold bloc to part with goods and services in exchange for it – i.e. prices rose (and note that we have derived a basic quantity theory of money in flow terms here).

In this light, the difficulties of both redeemable and fiat money, where the value of the signal is *greater* than the net marginal cost of generating it, are immediately apparent. In both cases, this divergence presents a pure arbitrage opportunity to anyone with the ability to generate the signal at a lower cost than

the value attributed to it. In order to maintain the high signal value, therefore, it is necessary to *trust* anyone with that ability. In practice, this entails trusting trading partners to have not counterfeited the money, but more importantly, trusting the issuer not to have exploited the arbitrage opportunity available to it. Suppressing third-party counterfeiting is a straightforward enough task with sufficient state capacity, but for the issuer the problem is more potent. Redeemability was one way of engendering trust, by providing legal penalties to nullify any arbitrage profit if the issuer should overissue. In a fiat system, on the other hand, this trust is provided for by insulating the issuer – usually a central bank – from profit-maximizing incentives and ensuring administrative independence from the central government, which might also be tempted to profit-maximize with the privilege. And indeed, fiat money has only been viable on large scales for a very short period of recent history, after a long development of self-enforcing bureaucracies in the developed world.

Compared with redeemable and physical fiat monies, the difficulties of electronic money are similar in kind, but even more extreme. If trust should be lost in the issuer of a paper money after all, at least the paper has a positive (if small) marginal cost to reproduce, and can devolve back to a low-valued commodity money.⁴ If the signal constituting the money is *electronic*, the marginal cost of reproducing an electronic signal is for all intents and purposes zero. Furthermore, while counterfeiters of fiat currencies can be adequately prosecuted, the transmission of unauthorized electronic signals is generally too difficult to prosecute with any regularity, a fact which caused considerable upheaval in media industries following the widespread adoption of the internet.

Conventionally, therefore, electronic payment systems – for example, PayPal for consumer payments, or the ACH clearing system for interbank payments – have had to rely on the same kind of trust as redeemable and fiat monies to prevent the issuer from succumbing to the arbitrage temptation. The issuer maintains a centralized ledger documenting current balances,⁵ and payment signals must go through the centralized issuer (who is in a position to verify the signals) in order to update the ledger. Money users must, therefore, trust the issuer not to take advantage of its position, or at least the legal system to prosecute any breaches of that trust.

Blockchains take a different approach to the problem: rather than relying on trust (whether in the issuer or the legal system) to counteract the temptation to arbitrage a costless signal, cryptocurrencies make the signal *artificially* costly to forge, exploiting the property of certain cryptographic problems that they are hard to solve but easy to verify a solution. In this sense, while cryptocurrencies are similar to fiat monies in their immateriality, they are more similar to commodity monies like gold in their design, where the costliness of generating the signal is what ensures its reliability.⁶ Bitcoin's pioneering proof-of-work

solution is discussed below, followed by a newer alternative way to accomplish the same task.

4. “WORK” AS A COSTLY SIGNAL

The Bitcoin protocol, documented in Nakamoto (2008) and launched in 2009, is the world’s first blockchain protocol. While the technical aspects have been discussed extensively, it will be worth contextualizing them in light of the signaling problems discussed so far.

Nakamoto (2008) is concerned to solve the “double spending problem” without needing to trust an authoritative ledger keeper. In essence this is the problem that electronic signals are easy to falsify: how is it possible to prevent a spender from transmitting a signal to one party, and then transmitting the same signal to another party, inducing both to part with costly goods or services? With an ordinary commodity money, double spending is of course physically impossible. With a centralized electronic money, where the duplication of electronic signals is costless, this problem is resolved by trusting an authoritative ledger where balances are recorded: a signal indicating payment of a certain amount of money can be cross-checked against the ledger to ensure that the payer in fact has the means to do so, and then updating the ledger with the new balances.

A *decentralized* electronic money *without* an authoritative central ledger keeper poses a more difficult problem: in order to distinguish valid monetary signals from an invalid ones, participants in the monetary network must *agree* on a ledger state. Thus, a ledger by itself does not solve the problem of determining valid payment signals, for now we have the problem of determining valid states of the ledger *as a whole* against which payment signals can be validated. If every participant has an incentive to broadcast a self-serving ledger state – say, with one’s own balances inflated – no one has an incentive to ratify the “true” ledger state, valid and invalid signals cannot be distinguished, and the value of any individual payment signal falls to zero.

The problem, then, will be to structure the protocol so that it is more costly to broadcast a self-serving ledger state than to ratify the true one. The Bitcoin protocol does so by interposing decentralized *validators* (more commonly referred to as “miners”) between payers and payees in a *proof-of-work* protocol. The validators maintain the ledger state and are rewarded for doing so with transaction fees, new coins, or both; collectively they are therefore the recipients of all relevant signals on the network to update the ledger state. There are two types of signals in such a protocol:

1. A transaction, broadcast from a payer to tell local validators to update the ledger. A transaction signal is (aside from voluntary transaction fees) cos-

tless to generate, and can be broadcast by anyone, provided it is consistent with the ledger state already maintained by the validators.

2. A block verification, broadcast from a validator to other validators. A verification takes a series of queued transactions received from payers, and performs a cryptographic operation on it that is costly to arrive at, but whose result can be easily verified as valid. This is the “work” which is proved in a proof-of-work protocol. The first validator to find a valid result broadcasts this result to other validators, who then append the included transactions to *their* ledger states and begin working to verify further transactions that have accumulated in the meantime.

Given that we cannot assume agreement on the ledger state among different validators, conventionally, a new validator will begin appending transactions to the ledger with the *most verified blocks* from its local network of validators. This feature of Nakamoto’s protocol is less to structure incentives, and more to create common knowledge as to what other validators will regard as valid modifications to the ledger – a coordination game, to use the language of game theory. Like a driver who decides to go on red and stop on green, a dishonest validator who decides to implement some other rule⁷ will not get very far, given what other validators are doing, if the goal is to broadcast a self-serving ledger state.

Nevertheless, it may still not be obvious how such a structure prevents dishonest signals from being acted upon. Consider, then, the prospects of an attacker, acting as both payer and validator, trying to broadcast a self-serving ledger. Given that the ledger is composed of a record of transactions (or a chain of transaction blocks, hence the name *blockchain*), and that each block in the chain is ratified by a costly-to-compute but easy-to-verify transformation of the data in the block, it will be extraordinarily difficult to generate an entirely new ledger of the requisite length from scratch.

A double spending attack, on the other hand, is easier: it is not a wholesale *replacement* of the ledger, or even an arbitrary change, but rather the alteration or reversal of the latest block of transactions. In this case, the process would go something like this:

1. The attacker broadcasts a transaction in order to spend a coin.
2. The payee waits until a validator successfully includes the transaction in a mined block before rendering services.⁸ Let us call the validator who does so A.
3. At this point, the longest chain is the one with the transaction included in it. Validator A, as well as those in his local network, start working on the next block of transactions.

4. In order to reverse the transaction, therefore, the attacker would have to generate a block *without* his own transaction, *and then another*, before validator A or any others in his local network succeed in generating one more.⁹ The odds of him doing so rise with the computing power available to him as a proportion of the network,¹⁰ and fall with the density of the network,¹¹ but for most intents and purposes, such an attack is exceedingly unlikely to succeed.

If signal falsifications are too costly to benefit the falsifier, even marginal ones like a double spending attack, signals on such a network can be regarded as presumptively valid. As in the example in the first section, we can feel confident in taking costly actions on the basis of a signal (say, delivering goods after having been paid) *only to the extent* that we are confident it would cost at least as much to falsify the signal than could reasonably be gained from doing so. Unlike a commodity money where it is costly to physically produce the signal, in the case of a proof-of-work blockchain, it must be costly to *update the ledger* (though not necessarily costly to the payer). Otherwise, the principle is exactly the same. If it were *not* costly, nothing would prevent an attacker from generating and broadcasting false transaction blocks, and it would not be possible to trust the reliability of the signal without trusting the person sending it.

5. “STAKE” AS A COSTLY SIGNAL

Bitcoin and other proof-of-work blockchains are commonly criticized for their resource costs. As of 2019, block verification on the Bitcoin network was absorbing 0.28 percent of global electricity production (McCarthy 2019), more than the entire country of Switzerland; and the demand for specialized mining hardware at the height of the crypto bubble caused GPU prices to briefly double in 2017. Hardware prices have tracked crypto prices since then. Like the resource costs of a commodity money (Friedman 1959; though see White 2015), the computational and electrical resources used have been criticized as being essentially waste. If we *can* trust the sender of a signal, the costs become unnecessary.

One response would be to point out that this trust must be generated some way or another. If we *do* trust a bank, or a government, or a language speaker, it must, at some level, be more costly for them to fake a signal than to produce an honest one; else the signal invites dishonesty, and can have no value in long-run equilibrium. Whether these costs arise from overt punishment such as legal sanctions in the case of banks, large downsides to failure such as increased susceptibility to external threats in the case of political communities (Bowles and Gintis 2011, ch. 8), or wholly artifactual sunken investments such as rituals in the case of language and normative communities (Knight 1998,

Iannaccone 1992); whether they are more or less transparent, more or less immediate, in the long run they must be borne one way or another in order for social life to function at all (Harwick 2020; Zahavi 1977, 1993).

True as this may be, a more useful response in the shorter run would be that these costs do not have to be resource costs, such as electricity or computational power, which we may concede to have unpalatable environmental externalities. Importantly, they may be opportunity costs of *any* sort, including the non-use of otherwise valid signals. This is the approach taken by *proof-of-stake* protocols, first proposed in King and Nadal (2012).

The transaction side of a proof-of-stake protocol is similar to proof-of-work protocols. Transactors broadcast signals to validators, who accumulate them into blocks and compete against each other for the right to have their block appended to the consensus state. But where in a proof-of-work system this competition consisted of computing a hard-to-solve but easy-to-verify transformation of the block, a proof-of-stake system obviates the hard-to-solve problem by allowing validators to send two types of signals:

1. A block proposal, which consists of a block of transactions, as well as certain information indicating the validator's "stake" – that is, an index of opportunity costs. There is a considerable variety of possible indices in proof of stake systems, including current balances (Vasin 2014), the cumulative time that the validator's balances have remained unspent (King and Nadal 2012), or a balance explicitly escrowed for stake purposes (Buterin and Griffith 2019).
2. An attestation, which forms a consensus on *which* of the proposed blocks to include. The protocol defines a criterion for selecting the next block out of the set of proposals. For example, it may be the block from the validator with the highest stake index, or it may be probabilistic, with the likelihood of being selected rising with the stake index, provided all reliable validators will identify the same block.¹² Here, the protocol's criterion establishes a coordination game. By rewarding attestation of the block that the most other validators agree on, and by allowing validators to agree that any other validator attesting a different block forfeits his stake, each validator is incentivized to attest the same block, even if it is not his own proposal.

In all cases, the signal can be trusted because it is costly to falsify. A false payment signal requires a false validation signal. The signal indicating a block is costly to generate, not in terms of physical energy, but in terms of unspent purchasing power, a cost worth bearing only for a chance to be selected. And the signal indicting agreement on a selected block is costly to falsify, due to

the possibility of punishment for failing to adhere to the protocol (Buterin and Griffith 2019).

Of course, people do hold onto money balances even without an explicit reward, sometimes even large sums of it. Accordingly, Poelstra (2015) has argued that the costs involved in some proof-of-stake specifications, especially the attestations, do not bite hard enough. Actual implementations thus far have largely been either hybrid proof-of-stake/proof-of-work systems or involve trusted elements somewhere in the network (for example centralized check-points). The area remains open and under active research.

6. INCENTIVES IN NONMONETARY BLOCKCHAIN APPLICATIONS

Monetary payment can be understood as a signaling game. Coordination upon a canonical ledger state is possible provided it is prohibitively costly to forge a self-serving payment signal, and blockchain technology has successfully accomplished this for a decentralized electronic currency. There have been suggestions and explorations for nonmonetary applications as well; to separate “blockchain” the distributed ledger technology from “cryptocurrency” the payment system (Davidson et al. 2018).

Although permissioned blockchains have taken up some of these applications, permissioned blockchains are fundamentally centralized and trusted systems, despite sharing some of the underlying technology. It is no surprise that traditional applications could be built straightforwardly this way, as they pose no new problems aside from the technical implementation. More promising is the Turing-complete Ethereum Virtual Machine, a decentralized blockchain architecture that allows network participants to use validators to run open-ended computer code on the blockchain, although with transaction fees that rise with the computing power necessary. In practice, this allows for a great deal of flexibility and automation in the management of funds. *Smart contracts* are virtual addresses that can receive and disburse funds algorithmically, in principle entirely without human intervention, and the types of smart contracts possible are limited only by what is writable in code and economically feasible to run.

Even with smart contracts, however, the track record for nonmonetary blockchain applications has been mixed at best. Recall that Bitcoin was as much a feat of mechanism design as it was a feat of engineering. Likewise with smart contracts, in applications from finance to identity to DNS, the mechanism design – the structure of the incentives facing the participants – is typically a harder constraint than the engineering problem of writing the code. In particular, for many nonmonetary applications, it is not sufficient to simply make it costly to forge an electronic signal. Especially where *risk* is involved,

where verification is imperfect, trust becomes necessary: not simply trust *in the signal*, as can be ensured by making it costly to forge, but trust *in the source* of the signal.

Avoiding the latter, of course, is the *raison d'être* of decentralized blockchains. Thus, the failure to branch out should hardly be regarded as surprising. A few examples illustrate the difficulty.

7. THE FAILURE OF ON-CHAIN FINANCE¹³

Recall our original problem of deciding to take a costly action in response to a signal. In the case of monetary exchange, what makes it worthwhile for you to take a costly action (say, to perform a service) upon receiving a signal is your expectation that you can use that signal in the future to elicit costly behavior from others. In other words, it will be worthwhile to take a payment if and only if you can be sure that you can spend it later on similar terms. Monetary exchange is defined, in part, by the constitutive nature of the signal: verifying the validity of the signal implies the warrantedness of your expectation.

Imagine, however, that the costly action being elicited is not a contemporaneous service, but a *loan*, the payment of a principal today, in exchange for the principal plus interest later. You receive a signal – say, a loan application – trying to induce you to give such a loan. How do you determine whether it is worthwhile for you to do so?

It will not do simply to check the formal validity of the signal, as suffices for monetary signals. A loan application, as normally understood, provides no guarantee that the borrower will not abscond with the principal, and – for any loan worthwhile to request – the costs of applying are far less than the benefits of defaulting. If this is the case, the validity of the signal is not sufficient to establish the warrantedness of your expectation.

To see what would make the lender's expectation warranted, and therefore his or her loan worthwhile, consider Figure 8.1. The lender must expect (4a) to be realized with enough likelihood to counteract exposure to the other states in stage 4. If this is to be true, the lender must be assured of the borrower's future *ability* to repay (stage 3), as well as the borrower's future *inclination* to repay (stage 4). Both of these are generally less certain prospects than the future acceptability of the monetary unit, not least because of the borrower's incentive to misrepresent both. Our question, then, will be: are there any mechanisms available to the lender to be able to reliably form such expectations without trusting the *source* of any signals, i.e. without identity information?

A solution analogous to the previous section would be to make false signals more costly. There are several ways of doing so algorithmically. First, the lender might require collateral upfront, giving the borrower a “stake” in cooperation in exactly the same manner as in proof-of-stake validation by giving the

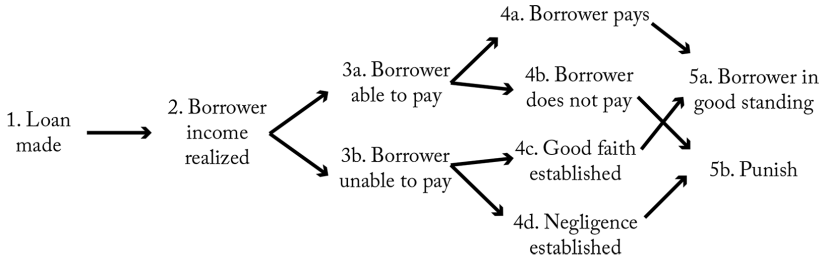


Figure 8.1 An extensive form diagram of the interaction between borrower and lender

lender leverage to punish the borrower *ex post* in the event of default. In this case, the problem collapses back into a problem like the monetary problem: if the collateral (as part of the loan application signal) can be verified to be valid and sufficiently valuable, it will be in the borrower's interest to repay the loan, and therefore in the lender's interest to lend it. This is the approach taken by smart contract platforms such as ETHlend and Ripio, which require *overcollateralization* in order to apply for a loan. If the borrower defaults, regardless of if the borrower is unwilling or unable, the contract automatically remits the collateral to the lender.

Algorithmic enforcement of a collateral contract amounts to the elimination of (4c), where the borrower does *not* repay and yet remains in good standing (in this case the punishment in 5b would be forfeiture of the collateral). An important function of finance, however, is to facilitate investment or consumption smoothing *before the income is available*. If it were necessary to have saved the purchasing power beforehand (albeit in illiquid form), much of the usefulness of finance for pricing and transferring risk would be nullified.

Another way to increase the cost of a false signal would be *reputation*, one of the considerations from the carnival parable. Reputation is incompatible with *anonymity*, but it is not incompatible with *pseudonymity*: identity on the blockchain is optionally persistent across interactions, although not linked to a real-world identity, and can be abandoned at the user's discretion. Reputation makes it possible to establish repeated dealings, which for many interactions hold forth enough future benefits that the prospect of losing them is a sufficient opportunity cost to dissuade bad faith. These dealings do not even have to be with the same party if interaction history is public; think of Yelp reviews left by tourists. Provided lenders as a whole are unwilling to deal with any borrower who has defaulted in the past, the signal (the loan application) would in principle reliably indicate the expectation (the borrower's future willingness to repay).

Unfortunately, Harwick and Caton (2022) show that reputation, in a pseudonymous environment and without the possibility of punishment beyond terminating the relationship, is *not* sufficiently costly relative to the benefits of defaulting to induce the borrower to repay. Even under the best-case scenario where a borrower could repeatedly borrow a sum forever, because the costs and benefits of defaulting both rise linearly in the size of the stream of loans, the size of the loan does not matter for this conclusion. The threat of losing a reputation is costly, but not enough to ensure the incentive-compatibility of repayment, provided the borrower can rejoin the network with zero reputation, as he always can on a permissionless blockchain. This is *especially* true if the lender is only imperfectly able to distinguish (4c) from (4b) or (4d), which gives some amount of cover for (and therefore lowers the cost of) opportunistic default.

If (4c) cannot be ruled out, it is incumbent upon the lender first, to form expectations about the borrower's future behavior *ex ante*, and second, to evaluate the borrower's behavior *ex post* in the case of default to determine good standing or not. Both of these are questions of judgment involving open-ended information, and neither is algorithmically decidable.

The expectations problem, as shown above, requires the lender to form an expectation over both stages (3) and (4) while still prior to stage (1). Forming an expectation over *ability* to pay may involve bank statements, pay stubs for consumer loans, or detailed plans and market analyses for investment loans. Trust can be offloaded from the borrower to a cosigner, or to a bank, but it is not possible to make such an evaluation without trusting *some* party by virtue of its identity. The *ex post* evaluation is similarly open-ended: if the borrower is *not* able to pay, the lender must establish whether the failure is due to bad luck (4c), or moral hazard (4d). In the former case, it is in the lender's interest to continue doing business with the borrower; in the latter case, the lender can terminate the relationship or (if the option is available) prosecute for negligence.

The fact that this information may exist off-chain is not, by itself, an insuperable hurdle: oracles such as Teller allow trusted information to be fed into smart contracts as part of their decision rules (Antonopoulos and Wood 2019). Although oracle-provided information compromises the trustless and pseudonymous qualities of blockchain interactions, the ability to incorporate trusted information through oracles does not impinge on existing trustless applications. Those willing to provide or rely upon trusted information can do so; others are not affected.

More importantly however, the limitation is not simply that insufficient *information* is available to be evaluated, but that *any deterministic decision rule* that outputs a decision based on a given information set, whether an *ex ante* expectation or an *ex post* evaluation, can always be exploited, and will not

suffice to discourage the borrower's moral hazard. Both decisions, therefore, are – at least in the breach – necessarily discretionary; that is to say, irreducibly non-computable (Harwick 2020).

Indirect signals – that is, *nonconstitutive* signals that indicate (but are not equivalent to) a state of the world – are vulnerable to Goodhart's Law, that any metric which becomes a target will eventually lose its usefulness as a metric. In our case we can reframe it as a general law of signaling systems: any indirect signal of a state of the world (for example, bank statements and market analyses) will fail to indicate that state of the world (namely, the profitability of the project) once people (e.g. lenders) begin to condition costly behavior (loans) upon it, and thus once it becomes worthwhile to falsify the signal, unless (1) discretionary judgment can be used to assess the validity of the signals non-deterministically and identify new signals if necessary, or (2) the payoffs can be modified *ex post* by punishment, a process which itself entails discretion as in (1). In the absence of these latter two possibilities, it will never be worthwhile to condition behavior on such an indirect signal. Our σ , returning again to the opening parable, is locked at its minimum of 0.5, and the signal is ignored.

Furthermore, in addition to the problem of forming an expectation over the borrower's future income, there is also the problem of ensuring the borrower has an incentive to repay *regardless* of his future income, the decision between (4a) and (4b). If we are to be able to avoid simply placing trust in the borrower, which makes the problem trivial, it must be in the borrower's interest to repay the loan at the end of the period. Like the problem of the borrower's future income, the problem of the borrower's inclination to repay cannot be solved purely algorithmically, but for different reasons. Because reputation is not sufficient collateral, viable lending requires the ability to impose costs for opportunistic default *outside* the bounds of the loan in question, or in other words, legal penalties. This, again, requires an inalienable identity of the borrower. If the borrower can simply abandon his place on the network and rejoin with a clean slate, there is no way for the lender to reliably avoid states such as (4b) and (4d), and the lender does not lend.

In short, the difference between blockchain payments and blockchain finance is that the former is feasible because the expectation problem can be reduced to verifying *present and past* states using information internal to the blockchain. Finance, on the other hand, requires verifying *future* states in ways that bear no simple relation to present states. Any applications of blockchain technology to uncollateralized finance, therefore, must involve trusted and inalienable identity information one way or another, possibly integrating it along with other trusted information using oracles. The structure of traditional borrowing and lending, especially in the complex forms it has taken on in common law legal systems, is such as it is in order to deal with these very

problems. Blockchain finance, therefore, is not necessarily dead on arrival, but it will not look radically different in its essentials from traditional lending institutions where identity is crucial.

8. NONMONETARY APPLICATIONS

Other uses of decentralized ledgers besides cryptocurrencies have proliferated. If we regard ledger entries not as spendable coins, but as generic ownable *tokens* that can signify anything, potential applications abound. Some, such as DNS resolution, file storage, and some aspects of identity ownership, are like coins in that ledger entries are valuable in their own right. Others, such as supply chain management, land titling, and other aspects of identity ownership, are more like deeds in that ledger entries are valuable only as a token of something else, and thus require – in addition to the technical infrastructure – an organizational interface to ensure congruence between sign and signified.

Interestingly, though many of these applications started life as independent blockchains, many have either become private or permissioned (and thus fall outside the scope of this chapter), or have set up tokens using smart contracts on the Ethereum blockchain.

There are two possible explanations for this kind of consolidation. First, it may simply be a result of network effects. If validators prefer to work on a well-established chain, new entrants face a chicken-and-egg problem and may be vulnerable to bad actors if they cannot rely on large amounts of computing power on the network. Given Ethereum's flexibility and established base, there may simply be no good reason to roll one's own blockchain for bespoke uses.

No doubt there is some validity to this logic. But even supposing a sufficiently large user base, there is a more fundamental problem with nonmonetary blockchain applications: namely, that the incentives to maintain the ledger require compensation in terms of something (1) network-internal, (2) valuable, and (3) fungible. Money obviously satisfies these criteria. It is unclear that other types of ledger entries would do so.

Consider Namecoin, which aims to use a blockchain as a decentralized replacement for domain name lookups. In principle, we could simply regard ledger entries as domain name registrations rather than coins, with payment occurring off-chain. Domain name registrations, however, are valuable, but not particularly fungible. If this is the extent of the data represented on the blockchain, it will be difficult to incentivize validators. For if validators are to be compensated through transaction fees, payment will need to happen on-network with an associated cryptocurrency. And if validators are to be paid in newly created ledger entries, nonfungibility of those entries makes costly network verification a rather less appealing prospect. While not impossible

for a sufficiently large network, the bootstrap problem would be particularly severe in this case (Luther 2019).

Accordingly, Namecoin consists of a decentralized domain name system *and* a payment system, grafted onto one another. Not only does this facilitate payment for domain names, but – more importantly – it allows the Namecoin blockchain to incentivize validators enough to ensure its self-sufficiency.¹⁴

The fungibility and the network problems can be traded off against one another, i.e. a particularly fungible ledger entry type will face lower network hurdles, and a particularly large network could (in principle) get away with a less fungible ledger entry. But the dearth of independent nonmonetary blockchains, even for fairly fungible entry types and large networks, suggests a relatively stringent constraint. File storage, for example – say, the right to a gigabyte of space – is more fungible than domain names, though less so than money. The most prominent players in this space, however, Filecoin and Sia, are both – like Namecoin, and for the same reasons – a payments layer on top of a decentralized file storage system. Storj, on the other hand, is a two-layered hybrid system (Storj Labs 2018): the retail side is not properly a blockchain at all and operates on the basis of some amount of trust; the wholesale side uses Ethereum tokens as payment. In no case is file storage itself offered as a reward to validators.¹⁵

The infeasibility of independent permissionless blockchains without a payment layer is by no means a serious barrier to these applications: like Storj, it is always possible to generate tokens on the Ethereum blockchain to handle the payments and verification side – again, not simply for network effect reasons, but in order to piggyback on a network that can compensate its validators with fungible monetary value. This possibility is sufficiently attractive that even extremely large networks with relatively fungible entries find it worthwhile to issue Ethereum tokens rather than establishing an independent blockchain. Reddit’s recent announcement of a blockchain-based “community points” system, for example, is based on an Ethereum token, despite the fact that the sheer size of Reddit as a social network means that community points could conceivably be fungible enough to incentivize validators on a standalone network. In fact, piggybacking on Ethereum is *less* convenient as Reddit will have to compensate validators with the general ether currency rather than its own internal token. Depending on the fungibility we attribute to Community Points, a valuation of this project could place a plausible lower bound on the network value of the Ethereum blockchain.

9. CONCLUSION: IMPLICATIONS FOR FUTURE DEVELOPMENTS

In evaluating both the historical development and the future prospects of blockchain technology, it is too easy to fall into broad pessimism or broad optimism. Realistically, blockchains are well-suited to some applications, and poorly suited to others, in a way that depends less on the application's informational requirements, and more on its game-theoretic structure.

Blockchain development over the past decade has succeeded largely in applications whose basic form is congruent with the kind of problem solved by blockchain technology. In particular, blockchains are a good solution to signaling problems where *the validity of the signal itself* is the key difficulty. Monetary payment is the paradigmatic example of such a problem.

Applications become less straightforward where the signal does not, by itself, convey the relevant expectation, and particularly where discretion is necessary to form those expectations. Finance is the paradigmatic example of such a problem, a form of exchange that cannot generally be accomplished in a pseudonymous blockchain environment. These, in particular, will require the ability to incorporate trusted information, e.g. with oracles. Provided it is transparent *who* exactly is being trusted in each instance, it is difficult to see a downside to this incorporation. Other applications, however – ones that rely on ledger consensus more generically – have the potential to piggyback on the payments layer, and it is these that we may be more optimistic about, especially as the Ethereum ecosystem matures.

NOTES

1. Which is not to say that one may trust *none* of the nodes; only that it is possible not to trust any *particular* node. Overall reliability will still depend on some threshold proportion of nodes on the network being reliable.
2. While this may seem question-begging, your uncertainty over the fairness of the game is formally identical to your uncertainty over which box the prize is in. For convenience therefore, we allow mistrust to enter only at one point.
3. Π is bounded from below by 0.5 because any probability $\Pi' < 0.5$ of it being in one box implies a probability $1 - \Pi'$ of it being in the other.
4. Such a devolution is documented in Luther (2015).
5. Although its payments are denominated in dollars, a PayPal balance (for example) should be regarded as a fully backed liability of PayPal. In this sense the ledger operator of a payment system is necessarily an issuer, even if the necessity of maintaining the asset side of the ledger cedes its autonomy on the liability side to the ultimate issuer of the unit of account (i.e. to the central bank).
6. Selgin (2015) refers to it as a “synthetic commodity money” for this reason.

7. One could imagine, for example, a rule that the canonical ledger will be the one with the most *individual transactions* rather than the one with the most verified blocks (since the number of transactions per block can vary).
8. A particularly cautious payee might even wait for several subsequent blocks, making the attacker's task that much more difficult.
9. It is a property of this problem that even minor changes to the block data will completely change the result. A's initial solution, therefore, will be of no use to the attacker, who must start over from scratch.
10. Hence the alternate name of a *51% attack*, meaning that this can be pulled off by any entity that controls >50% of the computing power on the network. These have become more of a concern with the rise of centralized mining pools, but other protocols such as Litecoin have addressed this problem with "ASIC-resistant" proof-of-work algorithms that eliminate the efficiency advantage of specialized hardware, and therefore an important source of returns to scale in computing power.
11. Low-density networks are vulnerable to so-called *eclipse* attacks, where attackers can isolate and dominate fragments of the network. See Heilman et al. (2015).
12. Specifically, Ethereum's proposed proof-of-stake protocol selects the block based on a combination of the block's hash value and the validator's staked balance. The former is cryptographically unpredictable based on the input, but deterministic. A validator who wanted to search for a block with a hash value that advantaged him in the search, would essentially be solving the same difficult problem as in a proof-of-work protocol. Ex ante, therefore, the block selection is essentially random with respect to all factors but stake, but all other validators will be able to identify the same "random" block from a given set.
13. This section draws on Harwick and Caton (2022), where a more formal version of the argument can be found.
14. Although it has increased its integration with the Bitcoin blockchain for network effect reasons as well through *merged mining*, which allows validators to verify on both chains simultaneously without diluting the computing power dedicated to either. See Judmayer et al. (2017).
15. In this kind of application, the primary reward would be for offering storage space rather than computing power (see Ateniese et al. 2014), with a secondary reward for validators maintaining the ledger/database coordinating the file system (although these roles are not necessarily separated in practice). The latter could conceivably be compensated with storage as a reward – though in practice it is not, for the aforementioned reasons – but for the former it would not make sense to offer storage as a reward for offering storage.

REFERENCES

- Alchian, A. A. (1977), Why money? *Journal of Money, Credit, and Banking*, 9(1), 133–140.
- Antonopoulos, A. M. and Wood, G. (2019), *Mastering Ethereum: Building Smart Contracts and DApps*. Sebastapol, CA: O'Reilly Media, Inc.
- Ateniese, G., Bonacina, I., Faonio, A., and Galesi, N. (2014), Proofs of space: When space is of the essence. In M. Abdalla and R. De Prisco (eds), *Security and Cryptography for Networks*. Oslo: Springer.
- Bowles, S. and Gintis, H. (2011), *A Cooperative Species*, Princeton, NJ: Princeton University Press.

- Buterin, V. and Griffith, V. (2019), Casper the friendly finality gadget. Available at <https://arxiv.org/abs/1710.09437>.
- Davidson, S., De Filippi, P., and Potts, J. (2018), Blockchains and the economic institutions of capitalism. *Journal of Institutional Economics*, 14(4), 639–658.
- Friedman, M. (1959), *A Program for Monetary Stability*. New York: Fordham University Press.
- Harwick, C. (2020), Inside and outside perspectives on institutions: An economic theory of the noble lie. *Journal of Contextual Economics*, 140.
- Harwick, C. and Caton, J. (2022), What's holding back blockchain finance? On the possibility of decentralized autonomous intermediation. *Quarterly Review of Economics and Finance*, 84, 420–429.
- Heilman, E., Kendler, A., Zohar, A., and Goldberg, S. (2015), Eclipse attacks on bitcoin's peer-to-peer network. In *24th USENIX Security Symposium*, USENIX Security, pp. 129–144.
- Iannaccone, L. R. (1992), Sacrifice and stigma. *Journal of Political Economy*, 100(2), 271–291.
- Judmayer, A., Zamyatin, A., Stifter, N., Voyiatzis, A., and Weippl, E. (2017), Merged mining: Curse or cure? In J. Garcia-Alfaro, G. Navarro-Arribas, H. Hartenstein, and J. Herrera-Joancomarti (eds), *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Oslo: Springer, pp. 316–333.
- King, S., and Nadal, S. (2012), PPCoin: Peer-to-peer crypto-currency with proof-of-stake. Accessed 22 June 2021 at <https://decred.org/research/king2012.pdf>.
- Knight, C. (1998), Ritual/speech coevolution: A solution to the problem of deception. In J. R. Hurford, M. Studdert-Kennedy, and C. Knight (eds), *Approaches to the Evolution of Language*. Cambridge, UK: Cambridge University Press, pp. 68–91.
- Luther, W. (2015), The monetary mechanism of stateless Somalia. *Public Choice*, 165(1/2), 45–58.
- Luther, W. (2019), Getting off the ground: The case of Bitcoin. *Journal of Institutional Economics*, 15(2), 189–205.
- McCarthy, N. (2019), Bitcoin devours more electricity than Switzerland. *Forbes*, accessed 22 June 2021 at <https://www.forbes.com/sites/niallmccarthy/2019/07/08/bitcoin-devours-more-electricity-than-switzerland-infographic/>.
- Nakamoto, S. (2008), Bitcoin: A peer-to-peer electronic cash system. White paper available at <https://bitcoin.org/bitcoin.pdf>.
- Poelstra, A. (2015), On stake and consensus. Accessed 22 June 2021 at <https://nakamotoinstitute.org/static/docs/on-stake-and-consensus.pdf>.
- Selgin, G. A. (2015), Synthetic commodity money. *Journal of Financial Stability*, 17, 92–99.
- Storj Labs (2018), Storj: A decentralized cloud storage network framework. Accessed 22 June 2021 at <https://storj.io/storjv3.pdf>.
- Vasin, P. (2014), BlackCoin's proof-of-stake protocol v2. Available at <https://blackcoin.org/blackcoin-pos-protocol-v2-whitepaper.pdf>.
- White, L. (1999), *The Theory of Monetary Institutions*. Malden, MA: Blackwell.
- White, L. (2015), The merits and feasibility of returning to a commodity standard. *Journal of Financial Stability*, 17, 59–64.
- Zahavi, A. (1977), Reliability in communication systems and the evolution of altruism. In B. Stonehouse and C. Perrins (eds), *Evolutionary Ecology*. London: Macmillan, pp. 253–259.
- Zahavi, A. (1993), The fallacy of conventional signaling. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 340(1292), 227–230.

APPENDIX

Bayes' rule is an equation that governs how a rational person – that is, a person interested in making optimal decisions based on probability assessments – should update his beliefs. It says nothing about how you arrive at your assessment of these probabilities: expectation formation is a fundamentally open-ended and non-computable process, which is – incidentally – exactly the problem that makes on-chain finance infeasible. But provided you *have* a probability assessment, Bayes' rule can tell you how you should update your beliefs given some new information.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (8A.1)$$

That is, the probability you should assign to the occurrence of A , given that you have a piece of knowledge B , is a function of (1) $P(B|A)$, the probability that B would have happened if A were true, (2) $P(A)$, your assessment of the probability of A if you didn't know B (also known as your *prior*), and (3) $P(B)$, the probability that B would be true if you weren't sure of A . The numerator $P(B|A)P(A)$ represents the probability that A and B *both* occur. Therefore, the equation states, the probability of A given B is the probability that A and B both occur as a proportion of all the times B occurs.

In the example from the first section, A stands for “I know which box the payoff is in”, and B stands for “someone just told me which box it's in”. The problem, then, is how much you should revise your confidence given B . Recall that your original confidence in your choice was denoted by Π ; this is $P(A)$. $P(B)$ is simply 0.5: given you know someone is about to tell you which box the payoff is in, you have no way of knowing before the fact which box *they* think the payoff is in until they tell you. And on the left-hand side, the probability that you know which box the payoff is in, given the advice you heard, will be your willingness to pay with a payoff normalized to 1: thus, $P(A|B) = \pi^*$.

The key component here, however, is how much credence you give to the advice. $P(B|A)$ is the probability that they would have pointed you to the box they did *given* that you eventually pick the correct box. Or, in other words, it is the probability that the advice is correct – what we have called σ .

Replacing the more abstract symbols from equation (1) with our own to which we have assigned concrete meanings, we arrive at the formula from the text: $\pi^* = 2\sigma\Pi$.