Zhang *et al. Applied Network Science* (2020) 5:19
https://doi.org/10.1007/s41109-020-0254-9

Applied Network Science

RESEARCH Open Access

On modeling blockchain-enabled economic networks as stochastic dynamical systems

Zixuan Zhang[1*†], Michael Zargham[2†] and Victor M. Preciado[1]

*Correspondence:
zixuanzh@seas.upenn.edu
†Zixuan Zhang and Michael Zargham contributed equally to this work.
[1]Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, USA
Full list of author information is available at the end of the article

Abstract

Blockchain networks have attracted tremendous attention for creating cryptocurrencies and decentralized economies built on peer-to-peer protocols. However, the complex nature of the dynamics and feedback mechanisms within these economic networks has rendered it difficult to reason about the growth and evolution of these networks. Hence, proper mathematical frameworks to model and analyze the behavior of blockchain-enabled networks are essential. To address this need, we establish a formal mathematical framework, based on dynamical systems, to model the core concepts in blockchain-enabled economies. Drawing on concepts from differential games, control engineering, and stochastic dynamical systems, this paper proposes a methodology to model, simulate, and engineer networked token economies. To illustrate our framework, a model of a generalized token economy is developed, where miners provide a commodity service to a platform in exchange for a cryptocurrency and users consume a service from the platform. We illustrate the dynamics of token economies by simulating and testing two different block reward strategies. We then conclude by outlining future research directions that will integrate additional methods from signal processing and control theory into the toolkit for designers of blockchain-enabled economic systems.

Keywords: Economic networks, Differential games, Stochastic processes, Control systems

Introduction

During the 2007-2008 global financial crisis, serious abuses by major financial institutions initiated a series of events resulting in a collapse of the loosely regulated financial network. This crash unveiled major weaknesses of traditional financial systems and instigated a general feeling of distrust on banking institutions. In this context, decentralized economic systems based on cryptocurrencies, such as Bitcoin (Nakamoto 2008), were developed and launched by a group of cryptographic activists who believed in social change through censorship-resistant and privacy-enhancing technologies (Rainer et al. 2015). Prior to Bitcoin, several attempts to establish digital currencies were made, including *b-money* by Dai (1998), *hashcash* by Finney (2002), and *bit gold* by Szabo (2005). Instead of relying on large

financial institutions for their operation, these cryptocurrencies proposed a decentralized economy in which a collection of economic agents coordinate through a peer-to-peer network of computers via a blockchain protocol.

At a high level, blockchain aims to keep track of a ledger of valid transactions between agents of the economy without the need of a central institution for coordination. In order to keep track of a faithful and accurate list of transactions, the ledger is broadcast and replicated across all the machines in a peer-to-peer network. To enforce that the transactions in the ledger are valid (i.e., there is no negative balance or double spending), the network 'as a whole' coordinates to accept or reject new transactions according to a set of rules aiming to detect and block the operation of malicious agents (e.g. Byzantine attacks where malicious nodes can send arbitrary messages to different nodes in the network (Lamport et al. 1982)). Blockchain implements this idea by bunching together a group of new transactions that are added into a chain of blocks only if these transactions are validated by the peer-to-peer network. Consensus protocols are particularly important in this validation step, since they are commonly used to reconcile conflicting versions of the ledger. A particular protocol used to enforce consensus is *Proof of Work (PoW)* (Bentov et al. 2014), currently used in Bitcoin and Ethereum. PoW is just one particular example of many other consensus protocols, such as the *Practical Byzantine Fault Tolerant* algorithm (PBFT) (Castro et al. 1999), *Proof of Stake* (PoS) (Buterin and Griffith 2017), *Delegated Proof of Stake* (DPoS) (Larimer 2014), or *Proof of Useful Work* (Ball et al. 2017) to mention a few. Such a hierarchical representation of blockchain state has also been articulated in Shorish (2018). Canonical results in decentralized coordination using state space representations are presented in Olfati-Saber et al. (2007).

A token economy

Regardless of the underlying network architecture and consensus mechanism, the resulting economic networks remain largely similar, with economic agents driven by incentives under a set of rules. It has become increasingly important to design the right set of microscopic rules and incentives that can achieve the desirable system-level behavior (Voshmgir 2019). The term *cryptoeconomics* has been used to describe the design and study of incentives and mechanisms in a blockchain network (Voshmgir and Zargham 2019).

Although mechanism design and algorithmic game theory have been used to approach this design problem, agents in a real-world economy are not fully rational and are frequently exposed to disturbances (Nisan et al. 2007). In this paper, we draw inspiration from early military pursuer-and-evader differential games, where the models used are agnostic to the exact behavior of the players, to develop a theoretical and computational framework to validate the behavior of a networked economy from permissible actions and trajectories of the system (Isaacs 1999). Compared to other games and distributed control problems (Marden 2012; Marden and Shamma 2015; Ragavendran et al. 2011), the design problem in blockchain protocols has its own set of challenges, as agents have access to global information that can be used to collude or form Sybil attacks which are a class of malicious strategies leveraging synthetic accounts (Göbel et al. 2015; Heilman et al. 2015; Douceur 2002).

Differential games and control

Differential games bridge many concepts in control theory with game theory. Control theory is foundational to design systems that are robust to environmental noise and system-failures (Dockner et al. 2000). To fully harness the power of control theory, we first need to define the system-level objectives for the blockchain network which are often neglected in the business narratives. In addition, traditional control theory often deals with a physical system over which designers have direct control (Zargham et al. 2018). In designing an economic network of human agents and incentives, designers at best have indirect control over the incentive structure with little control over the exact behavior.

In this paper, we propose a stochastic dynamical system modeling approach for decision making and economics inspired by Peters and Adamou's work on ergodicity economics (Peters and Adamou 2018), Sterman's work on business dynamics (Sterman 2002), Foster and Metcalfe's work on complexity in economics (Foster 2005; Foster and Metcalfe 2012), Roth's and Lux's work on computational methods (Roth 2002; Lux 2008). A stochastic dynamical system model enables us to understand complex relationships within a system, as well as observe the business level impact resulting from secondary or tertiary dynamics which are often counterintuitive but obvious after the fact (Forrester 1971). Another major difference of our model from a classical control system model is that each individual follows their own control policies, not necessarily the ones that protocol designers hope for. This is where concepts from differential games come in. Agents follow their own policies as a function of their own beliefs of the system state and their own payout function. The collective behaviors of different agents result in sometimes unexpected system-level outcomes. The role of the designer is to design a set of rules and incentives such that the system-level goal can still be achieved irrespective of the exact behavior of the agents. In our model, we will model agent's decisions using random variables but the exact distribution varies widely by applications. For example, when dealing with arrival, a Poisson distribution is often used (Borowski and Marden 2015; Guzman and Mojica-Nava 2017). Details around how to select and parameterize a distribution are beyond the scope of this paper.

In Fig. 1, we present our model for agents' beliefs and policies. Random processes are fed into the agent policies, which collectively update the unobservable full system state. Agents in the system can then observe the state and form their beliefs, signals, and objectives that impact their policies in the next iteration. Despite the presence of these unobservable states and random processes, allowable agent policies and system mechanisms should still achieve certain observable outcomes.

Overview

This paper proposes a new mathematical framework, based on tools from dynamical systems theory and control engineering, to model and analyze the function of blockchain-enabled systems. In particular, we borrow a modeling framework widely used in control engineering called *state space representation* (Sontag 2013). According to this framework, there is a set of abstract variables, called states, evolving over time (either continuous or discrete) according to a set of rules. In the discrete-time case, the evolution rules are described in terms of a first-order difference equation, in which the values of the states at a given time $t \in \mathbb{N}$ depend exclusively on the values of the states at time $t - 1$. Hence, given the initial values for the states at the origin of time (i.e., $t = 0$), it is possible to

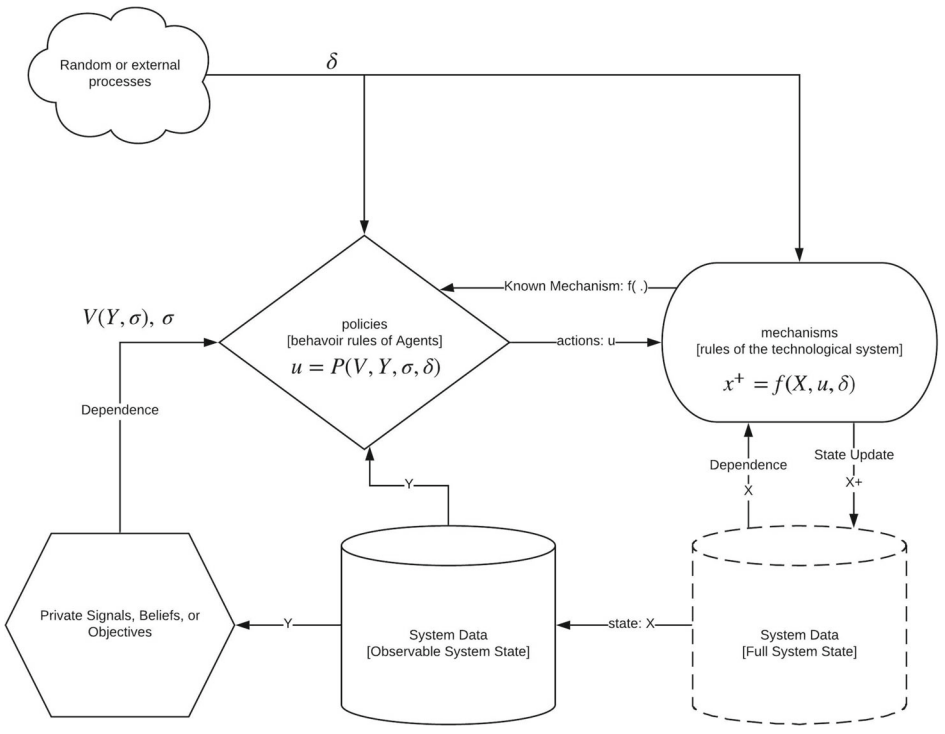

Fig. 1 Expanded representation of the differential game accounting for a variety of factors beyond the states and actions, including private signals σ, private goals $V(\cdot)$, a partially observable state $\mathbf{X}$ with an observable subspace $\mathbf{Y} \subseteq \mathbf{X}$, and some additional environmental stochastic processes δ drawn from potentially unknown distributions

recover the states at any time $t > 0$ by solving this recursion. There is a rich mathematical theory to analyze state-space models, specially in the linear case, in which the states at time t depend on the states at time $t - 1$ according to a linear transformation. In this paper, we propose a linear state-space model of the blockchain network whose set of states represent transaction addresses. Notice that, as new transactions take place in the decentralized economy, the number of states in the model increases over time. This results in some technical difficulties that we overcome by proposing a linear time-expanding (LTE) state-space model which we will use to analyze the temporal behavior of the blockchain. As a reference case, we will model and analyze the evolution of the state in the public Bitcoin network using the aforementioned LTE modeling framework. Using tools from state-space theory, Lyapunov-like functions (Zargham et al. 2018; Khalil and Grizzle 2002; Smith 1984; Park et al. 2019) in particular, we will illustrate how to enforce a global property, namely, the total amount of currency in the system, using local state-transition rules.

To further elucidate the power of the state-space modeling framework, a model of a generalized token economy is proposed where miners provide a commodity service to a platform in exchange for a cryptocurrency and users consume a service from the platform. Certain key metrics of the economy are defined, such as the growth of the network, intrinsic value of the token, and volatility of the service performed denominated in fiat currency. A momentum-based secondary market of the price of the token is also included

in the system. This baseline model allows us to run Monte Carlo simulations and observe outcome of complex dynamics and reason about the evolution of the system.

The paper is structured as follows: after providing appropriate background, we describe the LTE state-space model and analyze the behavior of the Bitcoin network using our framework. We then present a model of a generalized token economy as an extension to the theoretical framework with a baseline simulation based on different open-loop block reward schedules. The paper is concluded by exploring model limitations and outlining future research directions.

Theoretical framework

The following characterization attempts to create a useful abstraction over the properties of a blockchain network. It is not an attempt to describe how a blockchain network works, but to provide tools to engineer economic systems within such a network. All state variables are real-valued to make derivations more intuitive and straightforward.

Characterizing the economic network

The following definitions are used to characterize the unbounded agent state space and to relate that to the bounded state space over which system requirements may be defined. This model does not require that the system be implemented with a centralized state but mathematics characterizes the macro scale evolution of the system in terms of formally defined micro elements. This framework is critical for understanding the relationships between agents and accounts within a decentralized system.

Furthermore, timescale of the state space model is defined so that agent actions can be properly modeled. A discretization of the sequence and aggregation of actions into transaction blocks is necessary to simulate the system at this level. However, the construct provided is sufficiently general that if the discrete time is mapped to atomic events, the differential equations models may be used for event based simulation without deriving new models.

This model is defined over addresses. All addresses are defined by public key private key pairs, where private keys are used for cryptographic proof of the right to act as agent. Without loss of generality more complex schemes such as multi-signature schemes can be substituted for the simple private key proofs assumed here. Further characterization of cryptographic schemes is not required as this section focuses on the system dynamics rather than the means of enforcing those dynamics.

Definition 1 *Let $\mathcal{A}$ be the set of all possible **Addresses** as determined by the range of the cryptographic hash function used in the system implementation. At any time the set of addresses that exist is $\mathbf{A} \subset \mathcal{A}$.*

An address $a \in \mathbf{A}$ is referred to as an agent when the address is assumed to map to an identity and thus can perform an action. An address $a \in \mathbf{A}$ is called an account when it contains code declaring one or more states and associated mechanisms which are exposed to other agents. Agency is the ability of agents to take actions through proof of control over accounts. In other words, an agent controlling account $a \in \mathbf{A}$ has access to a set of actions ($\mathbf{U} \subset \mathcal{U}$ as defined in Definition 4) and mechanisms ($\mathbf{F} \subset \mathcal{F}$ as defined in

Definition 5). The following definitions are concerned with the mutation of system state by agents' actions as defined through Definitions 2 to 9.

Definition 2 *The **Ledger State** is a shared data structure* $\mathbf{L} \in \mathcal{L}$ *which evolves when agents perform transactions by taking actions with respect to mechanisms;* $\mathcal{L}$ *denotes the space of all valid ledger states* $\mathbf{L}$*. The state* $\mathbf{L} = \{\mathbf{X}, \mathbf{T}\}$ *where* $\mathbf{X}$ *is a set of local states over all accounts as defined in Definition 3 and* $\mathbf{T}$ *is a (partially) ordered list of transactions as defined in Definition 6.*

The list of transactions $\mathbf{T}$ need not be strictly ordered because transactions are local operations on the state with respect to accounts. It suffices to know the order of any transactions with dependence on or modification of a shared element of the state $x \in \mathbf{X}$. Further formalization of event orderings is deferred from discussion in this document.

Definition 3 *Consider the **Global State** $\mathbf{X}$ as an aggregate set of **Local States**, X_i, over each address $i \in \mathbf{A}$. To further distinguish between account state and controlling agent for each address, **Local States** can also be defined in terms of the state declared by account $i \in \mathbf{A}$ but controlled by agent $j \in \mathbf{A}$ as X_{ij}.*

It is immediate that each local state X_{ij} must be declared by a unique account $i \in \mathbf{A}$. Therefore, the local states can be interpreted as a partition of $\mathbf{X}$ over accounts. It is not, however, assumed that there is a partition over agents dimension since an agent can control multiple accounts which only maintain the states.

Definition 4 *For any agent $a \in \mathcal{A}$ there is a state dependent **Action Space** representing all legal actions of agent a given a global state $\mathbf{X}$ under some mechanisms f as defined in Definition 5. A particular **action** is denoted as $u \in \mathbf{U}$ where $\mathbf{U}$ is the set of actions that exist at that point in time and $\mathbf{U} \subset \mathcal{U}$, a set of all possible legal actions.*

Definition 5 *Consider the set of **Mechanisms** to be $\mathcal{F}$ such that any $f \in \mathcal{F}$ is an operator*

$$f : \mathcal{X} \times \mathcal{U} \longrightarrow \mathcal{X} \tag{1}$$

where $\mathcal{X}$ is the space of all possible states $\mathbf{X}$ as defined in Definition 3 and $\mathcal{U}$ is a space of all legal actions as defined in Definition 4.

Mechanisms like states, must be declared by an account and in many cases will have been declared alongside specific local state variables X_i which the mechanism operates on. However, no such assumption will be made formally.

Definition 6 *The set of all possible **transactions** is denoted as $\mathcal{T} = \mathcal{A} \times \mathcal{F} \times \mathcal{U}$ where an element $t \in \mathcal{T}$ is defined $t = (a, f, u)$. In order for the transaction to be valid, agent a must have the right to perform the state update operation $\mathbf{X}^+ = f(\mathbf{X}, u)$ given the current state $\mathbf{X}$.*

As previously defined in the Ledger State, a sequence of transactions organized into a (partially) ordered list is denoted as $\mathbf{T}$. The partial ordering may be considered without

loss of generality by noting that such a partial ordering is defined precisely by the independence of the final output to the ordering of the list. This occurs when there is strong separation in the states, accounts, and mechanisms involved in the transactions in $\mathbf{T}$.

Definition 7 *A **policy** $P : \mathcal{X} \longrightarrow \mathcal{U}$ is a state dependent strategy over a particular mechanism $f \in \mathcal{F}$. An agent $a \in \mathcal{A}$ is said to be using policy P over mechanism $f \in \mathcal{F}$ if it monitors the state $\mathcal{X}$ and broadcasts transaction $t = (a, f, u)$ associated with action $u = P(x) \in \mathcal{U}$.*

This definition allows for the case that no transactions are made because the conditions for a transaction under policy P are never met but it excludes the degenerate cases where there is no state for which a transaction will be generated. A practical consideration of this framing is the agents engagement level in monitoring the state. For the purpose of simple notation the sampling rate of the monitoring process is absorbed into the definition of P. During simulation. it is made explicit for tuning purposes.

Definition 8 *Consider ledger **State Transitions**; the Ledger State may be updated for any valid sequence of transactions $\mathbf{T} = [\ldots, t, \ldots]$, where $t = (a, f, u)$ is valid given $\mathbf{X}$ when the operation is applied. Without loss of generality, it is assumed all transactions are valid because invalid transactions are rejected.*

An atomic update is defined

$$\mathbf{X}^+ = f_t(\mathbf{X}, u_t) \text{ for any valid } (a_t, f_t, u_t) \text{ given } \mathbf{X} \tag{2}$$

where f_t is the mechanism used in transaction t and u_t is the action taken for transaction t. For a block defined by sequence of transactions $\mathbf{T}$, the state update is

$$\mathbf{X}^+ = f_N(f_{N-1}(\cdots f_1(f_0(\mathbf{X}, u_0), u_1) \cdots, u_{N-1}), u_N) \tag{3}$$

where the list of transactions $\mathbf{T}$ is indexed by $\{0, 1, \ldots, N\}$.

Definition 9 *The **Ledger Trajectory** is a sequence of ledger states $\mathbf{L}(k) = \{\mathbf{X}(k), \mathbf{T}(k)\}$, indexed by $k = 0, 1, \ldots, K$ such that*

$$\mathbf{X}(k+1) = f_N(f_{N-1}(\cdots f_1(f_0(\mathbf{X}(k), u_0), u_1) \cdots, u_{N-1}), u_N) \tag{4}$$

for transactions $\mathbf{T}(k)$ is indexed by $k = 0, 1, \ldots, N$.

Notation may be simplified by defining $\mathbf{F}_k$ to be the composition of the transactions in $\mathbf{T}(k)$ such that

$$\mathbf{X}(k+1) = \mathbf{F}_k(X(k)) \tag{5}$$

denoting the closed loop state update accounting implicitly for the actions $u = P(X)$.

At any time the most recent Ledger State may be denoted as $\mathbf{L}(K) = \{\mathbf{X}(K), \mathbf{T}(K)$ where the integer K is the block height and $\mathbf{L}(0) = \{\mathbf{X}(0), \mathbf{T}(0)\}$ is the genesis block. The number of transactions is dependent on the block $N = N(t)$. Under this definition the **Blockchain** is characterized precisely by the trajectory of generalized dynamical system in conical form. As defined, the differential equations can be used to characterize the system with atomic transactions as the basic unit of time. Organizing transactions into blocks provides a means of testing block based logic.

Any mechanism that can be implemented as an account under this framework provides an explicit contribution to the actions available to all other accounts within the system. The explicit characterization of an account and its subsequent state changes permits the estimation of changes in any utilities defined over the network state. Using the second order discrete networked system model, it is possible to both formally analyze the reachable state space and simulate the response to incentives with respect to a variety of behavioral assumptions.

Characterizing the peer-to-peer network

Under this formal model there are two distinct concepts matching the term **Network**. The state space model defines the evolution of a network of interacting accounts. From this point of view, the economic network is a robotic network with agents represented by accounts, each of which has its own unique state space and action space defined in part by all of the other agents (accounts) in the network. The agent (account) states of all network participants and their backward discrete difference equations are visible to other agents and any external observers capable of querying the Ledger State.

Consideration of the external viewer brings attention to the other concept of a network which is required to model this system; the communication and computation network responsible for maintaining account states, computing state updates, verifying the validity of blocks of transactions, and to agree on the correct sequence of blocks when multiple valid sequences are available.

Definition 10 *A **Node** is a member of the Peer-to-Peer Network with the ability to broadcast a transaction* $\mathbf{tx}$ *for which it can prove control of the initiating account* a_0 *using the associated private key, and the ability to verify the validity of transactions broadcast by other nodes.*

Definition 11 *The **Peer-to-Peer Network** is the set of nodes* $j \in \mathcal{V}$*, participating in the communication and computation network, each maintaining a copy of the Ledger State* $\mathbf{B}_j(k)$ *and edges in this network represent communication between nodes. The Ledger State here refers to the Ledger State at the underlying peer-to-peer network layer. Nodes in the network reach a consensus over their individual Ledger States* $\mathbf{B}_j(k)$ *to form a Ledger State on the economic layer,* $\mathbf{L} \in \mathcal{L}$*, as defined in Definition 2.*

Note that each node j may have its belief of the Ledger State $\mathbf{B}_j(k)$ such that for any two nodes $\mathbf{B}_j(k) \neq \mathbf{B}_{j'}(k)$ for $j \neq j'$. However, It is guaranteed by the underlying cryptographic protocol that both $\mathbf{B}_j(k), \mathbf{B}_{j'}(k) \in \mathcal{B}$, where $\mathcal{B}$ is the set of all possible Ledger States.

Definition 12 *A **Chain** is a valid sequence of Ledger States,* $\mathbf{C}(K) = \{\mathbf{B}(k) \in \mathcal{B}$ *for* $k = 0, 1, \ldots, K\} \in \mathcal{B}^{K+1}$ *where* $\mathbf{B}(0)$ *is the genesis block, K is the block height, and* $\mathbf{C} \subset \mathcal{C}$*, where* $\mathcal{C}$ *denotes all possible and valid Chain trajectories.*

The cryptographic protocol maintaining the ledger uses sequences of hashing functions to maintain a strict ordering on blocks such that any attempt to manipulate the history of the ledger state is immediately detectable by all nodes in the communication

and computation network. Since the cryptographic protocol only accepts blocks for which all state transitions are defined by legal transactions, the chain is also guaranteed to contain a self consistent historical trajectory of the state space model, both states and derivatives, starting with the initial condition $x(0)$ as defined in the genesis block.

We now return to the issue of nodes maintaining valid but different chains $\mathbf{C}_j(K_j) \neq \mathbf{C}_{j'}(K_{j'})$, which implies that Ledger States $\mathbf{B}_j(K_j), \mathbf{B}_{j'}(K_{j'}) \in \mathcal{B}$ but K_j and $K_{j'}$ may be different block heights. When nodes i and j have different chains $\mathbf{C}_j(K_j) \neq \mathbf{C}_{j'}(K_{j'})$ there must be a protocol by which they agree which one is correct or else the state could fragment into as many trajectories as there are nodes. The set of rules by which these inconsistencies are made consistent is called the consensus protocol. Here the focus is on the critical properties of the consensus protocol rather than algorithms through which those properties are realized.

Definition 13 *The **Consensus Protocol**, $\mathbb{C}$ is the process by which agents resolve inconsistency: $\mathbb{C} : (\mathcal{C}, \mathcal{C}) \to \mathcal{C}$ returning which of the two otherwise valid chains superseding the other.*

The existence of such a function alone does not ensure that the network does not fragment into partitions maintaining conflicting states. To further ensure consensus, the consensus protocol requires the following property.

Conjecture 1 *The Consensus protocol must impose a strict ordering on valid chains $\mathbf{C} \in \mathcal{C}$. It is sufficient that there exists a function $\Psi : \mathcal{C} \to \mathbb{R}$ such that for any $\mathbf{C}, \mathbf{C}' \in \mathcal{C}$*

$$\mathbf{C} \neq \mathbf{C}' \implies \Psi(\mathbf{C}) \neq \Psi(\mathbf{C}'). \tag{6}$$

Two nodes may resolve their inconsistency by each setting their Chain to $\mathbf{C}^* = arg\max_{C \in \{\mathbf{C}, \mathbf{C}'\}} \Psi(C)$.

The formalism is consistent with the Nakomoto consensus paradigm (Nakamoto 2008) where the function Ψ is the amount of work done to reach the current state in the competing chains. While it is possible for two chains to have exactly the same amount of work and still differ, such a discrete event is a measure zero outcome in a continuous probability distribution; thus, for the Bitcoin network using total work, (6) can be expected to hold with probability one.

For the purpose of the economic specification and subsequent design and analysis, it suffices to use any consensus protocol for which Conjecture 1 holds with probability one. Using a proof scheme as the one described above results in a lack of finality, meaning that there is always the possibility that another chain will supersede the one that a node is maintaining. A consensus algorithm with **finality** is one that agreement on $\mathbf{B}(k)$ for block height K would not be reversible by some later observation. The state update would hence be Markovian, meaning that the nodes would not be required to compare full trajectories $\mathbf{C}(K)$ to come to consensus over $\mathbf{B}(K)$. *Proof of Stake* based consensus methods under development aim to achieve this property (Buterin and Griffith 2017).

Bitcoin reference case

The public nature of data in the Bitcoin economic network has made it a great candidate for research on financial flows. Many of these models consider graphs of flows between accounts (Bovet et al.) or evolutionary market share of Bitcoin in the overall cryptocurrency market (ElBahrawy et al. 2017). This analysis will instead focus structurally on how very simple rules about what constitutes a valid transaction result in well-defined global properties.

Linear time-expanding model

The Bitcoin economic network is defined over block heights $k = 0, 1, 2, \ldots$, and there are $n_k = |\mathcal{A}_k|$ accounts at each block height k with the additional caveat that $n_{k+1} \geq n_k$. For consistency of notation with dynamical models on networks, accounts will be referenced with indices $i \in \{1, \ldots, n_k\}$.

Definition 14 *A **Linear Time-Expanding (LTE)** system has a state space model in the form of a discrete time varying linear model with the dimension of the state space $x \in \mathbb{R}^{n_k}$ which is monotonically non-decreasing while the state update matrices vary only in n_k.*

Consider a canonical form discrete time linear time varying model:

$$x(k+1) = A_k x(k) + B_k u(k) \tag{7}$$

where $x(k) \in \mathbb{R}^{n_k}$. Under this framework $A_k \in \mathbb{R}^{n_{k+1} \times n_k}$, but since there are no internal dynamics

$$A_k = \begin{bmatrix} I_{n_k} \\ 0 \end{bmatrix} \tag{8}$$

where I_{n_k} is the identity matrix. The matrix B_k is an all-to-all incidence matrix encoding all possible sends $B_k \in \{0, 1, -1\}^{n_{k+1} \times m_k}$ where $m_k = n_{k+1} \cdot (n_k - 1) = |\mathcal{E}_k|$ and $u(k) \in \mathbb{R}^{m_k}$. The edge set is given by $\mathcal{E}_k = \mathcal{A}_k \times \mathcal{A}_{k+1}$ because flows must originate from accounts that exist at time k. Hence,

$$[B_k]_{ie} = \begin{cases} 1 & \text{if } e = (j, i) \text{ for any } j \\ -1 & \text{if } e = (i, j) \text{ for any } j \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

Result 1 *The system in (7) is an **Linear Time-Expanding** (LTE) system because, for all k, A_k is an augmented identity matrix as defined in (8) and B_k is an all-to-all incidence matrix as defined in (9).*

The incidence matrix construction enforces the requirement of no double spend. Under this construction the local action $u_e(k) \in \mathcal{U}_e(k)$

$$\mathcal{U}_e(k) = \left\{ u_{e=(i,j)} \in \mathbb{R} \Big| \sum_j u_{(i,j)} \leq x_i(k) \forall i \right\}, \tag{10}$$

where (10) enforces the requirement that accounts cannot spend funds that they do not have. Note that the requirement is locally enforceable, imposing only constraints on the balance of account i and transactions to or from account i during block k. Viewed from the

perspective of account i, the local constraints on flow (no double spend and no negative balance) can be written as,

$$x_i(k) + \sum_j u_{(j,i)}(k) - u_{(j,i)}(k) \geq 0. \tag{11}$$

In practice, transactions encoded by the inputs u are processed with a strict ordering that can be enforced with only the sender's state, as defined in Definition 9,

$$u_{(i,j)} \leq x_i. \tag{12}$$

The model in (11) is a relaxation of the enforced requirement in (12); any block comprised of actions $u(k)$ that respect the individual transaction validity requirement (12) will satisfy the conservation law in (11). The relaxed equation is presented to demonstrate that the case of the Bitcoin network flow is in fact stronger than the conical network flow models in the controls literature (Zargham et al. 2013).

Globally invariant properties from local rules

Even though the Bitcoin economic network is a somewhat trivial system to study from a dynamical systems' perspective, it is actually much like a biological evolutionary system with complex global behaviors emerging from simple local rules. The trajectory of the system is defined entirely in terms of its state transitions and initial conditions. The dynamical system model remains structurally invariant even as the number of account grows unbounded. Each account, or local agent, has full control over its actions and the set of legal actions for each agent are defined and verifiable with information local to its agent state. These definitions of local legal actions provide properties that are suitable for a financial ledger of record. In order to introduce funds into the economy, a driving function $M(k) = \mu_k \nu(k)$ is added to (7) where $M(k) \in \mathbb{R}^{n_{k+1}}$ is the product of a scheduled positive scalar reward $\mu_k \in \mathbb{R}_+$ and a stochastic vector $\nu(k) \in \mathbb{R}_+^{n_{k+1}}$ such that $\sum_i \nu_i(k) = 1$ and obtain:

$$x(k+1) = A_k x(k) + B_k u(k) + \mu_k \nu(k). \tag{13}$$

Block reward schedule as a driving function, together with a set of local legal actions, guarantees that a low dimensional global property is enforced throughout the entire trajectory.

Theorem 1 *Given the local 'no double spend' rule as defined by (11) and the driving function defined in (13), the global quantity of Bitcoin over time, denoted as $y(t)$, can be proven to converge to the desired quantity:*

$$y_\infty = \lim_{k\to\infty} y(k) = \sum_{k=1}^{\infty} \mu_k. \tag{14}$$

The details of the proof can be found in the Appendix.

In the Bitcoin network the mining rewards are defined over $i = (1, \ldots, 32)$ halving intervals $r_i = (k_0, k_1, \ldots, k_{209999})$ each including 210000 blocks resulting in $\mu_k = \frac{\lfloor \frac{50\cdot 10^8}{2^i} \rfloor}{10^8}$ where $k \in r_i$. After the 32^{nd} interval the minted block rewards cease and the total quantity of Bitcoin is conserved. By computing the sum over the intervals, the final sum of Bitcoin $y_\infty = 20,999,999.9769$, generally quoted as 21 million Bitcoin. This does not

account for the potential loss of control of accounts with Bitcoin balances which reduces the effective supply.

The most powerful part about this characterization is that the system literally tracks a desired property $y(k) = \sum_k \mu_k$ for the entire trajectory, in fact in any valid trajectory, with no assumptions about actions of individual agents. This indicates that it is proper to think of blockchain-enabled economic systems as engineered economies where it is possible to encode legal state transitions in such a manner as to mathematically ensure the emergence of a low dimensional global property.

A generalized token economy

Cryptocurrencies in their tokenized form have the potential to be more than just currencies. A token can be a claim over some right, access, utility, or return. Depending on the design, tokens can align agent incentives and shape agent behavior through rewards, punishment, and future expectation. Ever since the advent of Bitcoin, there have been numerous tokens telling stories of various token economies (Buterin 2014; Maker 2017). Here we propose a generalized token economy model rooted in our theoretical framework above and can be easily extended to any specific token economies.

Model setup

Traditional networked economies have been characterized by a centralized platform where users and producers can exchange goods and services. The value of such a platform is often considered to be proportional to the square of the number of active users by Metcalfe's Law (Zhang et al. 2015). Similarly, new token economies are introducing new models where miners perform a service that is useful to users on the platform and get paid in a native token. Incentivized peer-to-peer file storage network and video streaming network are just two promising examples (Filecoin 2014; Petkanic and Tang 2017). In contrast to wasteful energy usage in Bitcoin, the work done by producers in these new token economies can actually be useful. Such new token economies also differ from traditional platform economies by coupling platform producers with service providers, commonly referred to as miners. The role of protocol designers is to design rules, policies, and state update mechanisms within the system while not actively participating in the economy. The model proposed aims to help protocol designers better understand their systems and make good design decisions. The following assumptions are important in simplifying the model and providing a baseline:

Assumption 1. The model considers each miner and user to be identical, with unit service capacity (similar to work in mean field games (Lasry and Lions 2007; Sahneh et al. 2013)). Each user demands one unit of service and each miner only provides one.

Assumption 2. The model assumes a perfectly competitive market where miners and users are both price takers and the service provided is a commodity with no product differentiation, given the open nature of a token economy (Gregory 2014; Hayek 1948).

With these assumptions in mind, a state space representation of the system can be defined with a minimal set of internal states that can capture different aspects of the system. We will also define *TOK* as the native cryptocurrency to the network. There are two

subsystems within our token economy. The first is with regard to the flow of *TOK* and the second is related to the service provided on the network. The two subsystems are connected with three important signals, miner's profitability, price of service on the platform, and the price of the token itself which are all treated as part of the Global State, $\mathcal{X}$.

Given our efficient market assumption, we have decided to leave out the modeling of *TOK* holdings entirely. For one, it is a subsystem in and of itself with its own dynamics about price movement relative to the amount of *TOK* in liquid pool and trading activities. In addition, with the efficient market assumption, we can assume that all the trading volume and liquidity has been embedded in the price (Malkiel 1989).

Assumption 1 reduces individual action spaces to population distribution which is a common approach in evolutionary dynamics (Mabrok and Shamma 2016). Instead of considering actions of each agent, our model considers aggregated forces that agent actions collectively exert on the system and treats that as part of the system states.

State variables

Given the above scope and assumptions, system state variables can be defined in Table 1. It is also worth noting that the price of the native token at time t, $K(t)$, is not just dependent on the internal states of the system as the token is openly traded on secondary markets from day one. $K(t)$ will have significant impact on agent incentives within the network as it bridges the token economy with fiat economy. $K(t)$ is also subject to influences from unobservable secondary market dynamics, speculation, and sentiments that may or may not correlate with actual activities on the network itself (Chu et al. 2015).

Each of the state variable defined in Table 1 constitutes the Global State $\mathcal{X}$ defined in Definition 3. State update operations as defined in Definition 6 can be represented discretely as $X(t+1) = f(X(t))$, where agent action u is collapsed into $X(t)$ given Assumption 1 above. As a general framework, our model is concerned with the consequences

Table 1 Definitions of system state variables

State Variable	Definition
$S(t)$	supply of unit service at time t
$D(t)$	demand of unit service at time t
$Q(t)$	quantity of unit service transacted at time t
$P(t)$	price of a unit service at time t
$R(t)$	miner profitability at time t
$K(t)$	price of a native token at time t
$B(t)$	block rewards released at time t
$C(t)$	cost of unit service at time t
$\Delta S(t)$	arrival of new unit service supply at time t
$\Delta D(t)$	arrival of new unit service demand at time t
$X_s(t)$	departure of unit service supply at time t
$X_d(t)$	departure of unit service demand at time t
$V(t)$	native token earned per unit fiat invested in the system at time t
$\frac{1}{V(t)}$	intrinsic value of native token at time t
$U(t)$	speculative value of native token at time t
$W(t)$	amount of native token left in block rewards pool at time t
$I(t)$	abstract index on network's progress to achieve its goal at time t

and dynamics of agent decisions, not the decisions themselves. The following system dynamics are different mechanisms that state trajectories can evolve.

System dynamics

Given some initial states, the following steps are taken at every time step t. A new network state, $\mathbf{X}^+$ or $X(t+1)$, will be returned and become an input to the same iterative steps in the next time step.

- New unit service supply and demand are determined based on private signals from the previous state.
- Service transactions happen based on unit service supply and demand in the current time step.
- Block rewards are minted and miners make profits from the system.
- Some speculation happens to the native token resulting in a new token price.
- Other system level signals and beliefs are updated.

System Dynamics 1 *$S(t)$ and $D(t)$ is first modeled as an arrival-departure stochastic dynamical system, a classic approach in stochastic models (Muntz 1972).*

$R(t-1)$ and $P(t-1)$ are the two driving signals for two main feedback loops in the system. $R(t-1)$ represents how much the network is paying for the service that miners provide. Similarly, $P(t-1)$ represents how much the network is willing to accept in exchange for the service capacity it provides. Hence, we are modeling the arrivals of new service supplied and new service demanded as two Poisson processes, as follows:

$$\Delta S(t) \sim Po(\lambda_s(t)) \tag{15}$$

$$\Delta D(t) \sim Po(\lambda_d(t)), \tag{16}$$

where $Po(\lambda)$ denotes a Poisson distribution of mean λ. State variables $\lambda_s(t)$ and $\lambda_d(t)$ capture aggregated agent actions in bringing new supply and demand onto the network.

On a population level, when providing the service is more profitable, more service supply will arrive. Similarly, when the price of the service gets lower, more demand will arrive as rational agents respond to price signals. One can enrich this model by introducing a reservation price and the probability of an agent consuming a service (Bimpikis et al. 2019). However, the current model is simple and robust without going into specific agent behavior. The mean of $\Delta S(t)$ equals to the mean of $\Delta S(t-1)$ multiplied by a percentage increase in miner profitability:

$$\lambda_s(t) = \lambda_s(t-1) \times \frac{R(t)}{R(t-1)}. \tag{17}$$

Similarly, the mean of $\Delta D(t)$ equals to the mean of $\Delta D(t-1)$ multiplied by a percentage decrease in the price of service as described by:

$$\lambda_d(t) = \lambda_d(t-1) \times \frac{P(t-1)}{P(t)}. \tag{18}$$

Treating departure from the system as constants simplifies the dynamics without losing much of its meaning. In the case where supply is increasing quickly, we can consider service departure as departing and then immediately arriving again. This can be accounted

for by a very positive $\Delta S(t)$. The same can be said about departure in demand. As such, the overall dynamics of service supply and demand can be described by the equations below:

$$S(t) = S(t-1) + \Delta S(t) - X_s(t) \tag{19}$$

$$D(t) = D(t-1) + \Delta D(t) - X_d(t). \tag{20}$$

System Dynamics 2 *$P(t)$ and $Q(t)$ are derived from $S(t)$ and $D(t)$ based on aforementioned simplifying assumptions.*

With the efficient market assumption, price can be set by relative strength between demand and supply (Gregory 2014),

$$P(t) = \frac{D(t)}{S(t)}. \tag{21}$$

When demand is greater than supply, a higher price is expected and vice versa. This price model, despite being very simple, accurately captures this relationship. It can be further expanded to include momentum, user valuation, and other factors that will affect service price. Quantity of unit service traded on the network at time t is the minimum between net new demand and supply. After all, no transactions will take place with unmet demand or supply. This can be made more realistic with a slippage later since not all matching supply and demand can find each other in the market. Nonetheless, this has been taken care of by the efficient market assumption where supply always meets demand. As such we can write the quantity equation as

$$Q(t) = \min(D(t), S(t)). \tag{22}$$

Both division in 21 and minimum in 22 are instances of mechanisms $f \in \mathcal{F}$ that can be replaced with any other mechanisms under reasonable assumptions depending on the use case. Our model serves as an example of how the state space model can be generally applied.

System Dynamics 3 *Block rewards are minted as defined by the protocol and released into the system as $B(t)$.*

In most blockchain protocols today, *TOK* issued in block rewards is usually set by a predetermined open-loop release schedule. However, we can consider a more abstract and generalized version of block rewards issuance by introducing the concept of a Key Performance Index (KPI) and tracking the network's progress towards achieving that with $I(t)$. Most traditional open loop block rewards scheme with a pre-determined release schedule is effectively treating the block time as the KPI, which can be written as,

$$B(t) = f(\Delta I(t)) = f(\Delta t). \tag{23}$$

As a mechanism, $f \in \mathcal{F}$, an open loop block reward scheme is a function of just the possible system state, $\mathcal{X}$ and is irrespective of agent action, u, i.e. $f : \mathcal{X} \longrightarrow \mathcal{X}$. However, if we define the release of block reward as a function of the rate of change in achieving the KPI, we can then create direct incentives based on what the network desires. Subsidy can be given out as a function of the change in $I(t)$ and the amount of *TOK* left in the rewards

pool. This set of subsidies $B(0), B(1), B(2), ..., B(t)$ can be the set of control policies that protocol designers can control to bootstrap the network to some target with some initial capital. This KPI can be as simple as the cumulative service transacted on the network over some period of time and it will bring agent action u back into the mechanism equation f in determining state changes related to block reward, i.e. $f : \mathcal{X} \times \mathcal{U} \longrightarrow \mathcal{X}$.

System Dynamics 4 *Agents speculate on secondary market prices of TOK as reflected in $K(t)$ that influences other agent behaviors.*

The secondary market price is a speculative estimator of future $\frac{1}{V(t)}$. Market decouples from current state of $\frac{1}{V(t)}$, because speculation is estimating future $\frac{1}{V(t)}$ in an effort to create returns. Hence, it makes sense to model $K(t)$ as a convex combination of its intrinsic and speculative value, as defined by $\gamma \in [0, 1]$,

$$K(t) = \gamma \frac{1}{V(t)} + (1 - \gamma)U(t). \tag{24}$$

Note that while the range of γ is technically a closed interval, the interesting cases are when $\gamma \in (0, 1)$. State variable $U(t)$ is the result of aggregating all individual agent's speculative actions and translating them as a form of pressure exerted on the system. This is in line with the general asset pricing framework that the price of an asset can be attributed to its fundamental and speculative value (Dimson and Massoud 1999; Keynes 1964; Bachelier 2011). Speculative value $U(t)$ captures momentum in price movement with a naive projection. The tuning parameter γ is treated as a constant but it can also be a randomized value that can randomize the composition of the mixture model.

System Dynamics 5 *Other system wide beliefs and signals such as $C(t)$, $V(t)$, and $R(t)$ are calculated.*

$C(t)$ is the cost of providing a unit service at some point in time in fiat currencies. This can be the output of another model but given our definition of a unit service, $C(t)$ can be any arbitrary value in our model. After all, the value of $C(t)$ only matters in relationship to the definition of the unit and the value of the unit service itself. We will thus model $C(t)$ as a stochastic process that is noisy but it is neither diverging nor converging, similar to sampling from a normal distribution with momentum,

$$C(t) = \alpha C(t - 1) + (1 - \alpha)N(\mu, \sigma). \tag{25}$$

Another signal $V(t)$ is defined as the revenue earned in *TOK* per unit spend in fiat currency for miners. $P(t) \times Q(t)$ refers to how much a miner can earn from transaction fees and $B(t)$ is the block reward subsidy provided by the protocol for $Q(t)$ transactions at time t, both in *TOK* denomination. Each unit of $Q(t)$ will cost $C(t)$ in fiat denomination and hence $C(t) \times Q(t)$ is the fiat spend to earn $P(t) \times Q(t) + B(t)$ in *TOK*. $V(t)$ hence represents how much tokens miners are earning for the service they provide,

$$V(t) = \frac{P(t) \times Q(t) + B(t)}{C(t) \times Q(t)}. \tag{26}$$

Multiplying $V(t)$ which is in $K/FIAT$ by the price of token at time t, $K(t)$, we get miner profitability in a unitless denomination. The system also cares about the inverse of $V(t)$, which is $\frac{1}{V(t)}$. This represents the intrinsic value of the token as it measures the value of

service provided per unit of token in $FIAT/K$, or the fiat cost involved in the production of one token (David et al. 2014). Miners' profit level in fiat currency can hence be written as

$$R(t) = V(t) \times K(t). \tag{27}$$

Simulation and evaluation

This paper leveraged an open-source computer aided design software, cadCAD (BlockScience 2019), for complex adaptive dynamics. System states and dynamics are encoded in every discrete time step of the simulation with certain key performance metrics defined as part of the system state. As a token-enabled platform economy, the platform needs to grow in adoption and in underlying value of the token. In other words, $Q(t)$, $V(t)$, and $K(t)$ should grow over time. The distribution over the total aggregated growth of $Q(t)$ and volatility in the price of the service provided $P(t) \times K(t)$ are also important for the network to succeed.

The Bitcoin block reward function follows a step decay function with 50% decrease in supply approximately every four years (Nakamoto 2008). This often introduces arbitrary shocks and unpredictable dynamics around the period when the supply is reduced. A similar but slightly different block reward schedule can be introduced where it follows a smooth exponential decay with a predetermined half-life. For a generalized token economy, we will now experiment with two different block reward functions. One that follows the step decay function of Bitcoin and the other follows a smoothed exponential decay.

Since agents know these block reward schedule ahead of time, future expectation of token supply reduction is taken into account in driving speculative behaviors (Alexandre et al. 2018). The two systems are different in the block reward function as defined in 23 that can potentially lead to different behaviors in 24, holding all else equal. More specifically, this experiment involves holding agent policy, $P : \mathcal{X} \longrightarrow \mathcal{U}$, constant, and changing an open loop system mechanism, $f : \mathcal{X} \longrightarrow \mathcal{X}$, to observe their impact on state trajectory. Monte Carlo simulations are then performed 100 times for each configuration over 1040 weeks or 20 years. A summary of the two configurations can be found in Table 2.

As observed in Fig. 2, a pre-announced step function that significantly reduces the supply injection into the system at discrete time steps introduces arbitrary shocks into the system. Agents adapt their policies to speculate around these moments when the expected new supply is sharply reduced. Figure 3, on the other hand, shows a much smoother exponential rise in token prices as time step increases. It is also worth noting that a step decay function resulted in greater speculative activities as intense speculation happens around those shocks whereas a smoothed decay does not provide such pivotal points for speculation. Similar results are observed in $\frac{1}{V(t)}$ on logarithmic scale because sharp supply reduction means that for the same amount of resource input, much fewer tokens are minted. Figures 4 and 5 further confirm that token price trajectories in our simulations are largely similar to cryptocurrency price movement observed in real life.

Table 2 Open loop block reward release schedule comparison

Decay Function	Total Supply	Initial Block Reward (weekly)	Half Life (weeks)
Step	21,000,000	50,400	208 (4 years)
Exponential	21,000,000	69,865	208 (4 years)

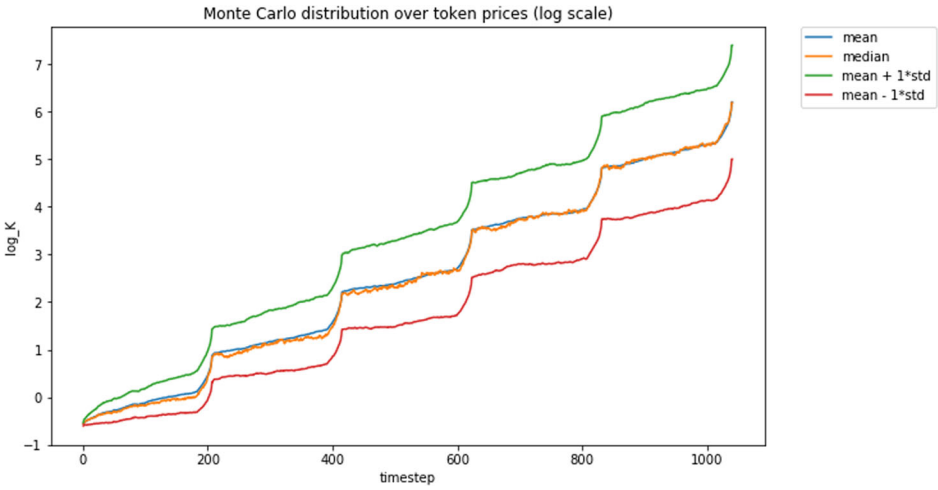

Fig. 2 Token prices over 1,040 weeks; statistics from 100 Monte Carlo Runs under step decay block reward schedule on log scale

Moreover, increased speculative activities as a result of regular and consistent token supply contraction shocks might lead to increase in service transaction on the network. Token price speculation around step changes in supply results in increased miner profitability in fiat denomination which attracts more miners to join the network. This will further lower the cost of unit service on the platform which will in turn attract more users and hence more service demand. More service demand leads to more revenue for miners and further reduces the cost of service provided. The presence of this positive feedback loop within the system has resulted in a rightward shift in the distribution of aggregated growth in Q when the block reward schedule is a step decay, as evident in Figs. 6 and 7.

The model and simulation presented are no doubt imperfect but they provide a mathematical framework and an example simulation derived from tools and theories in

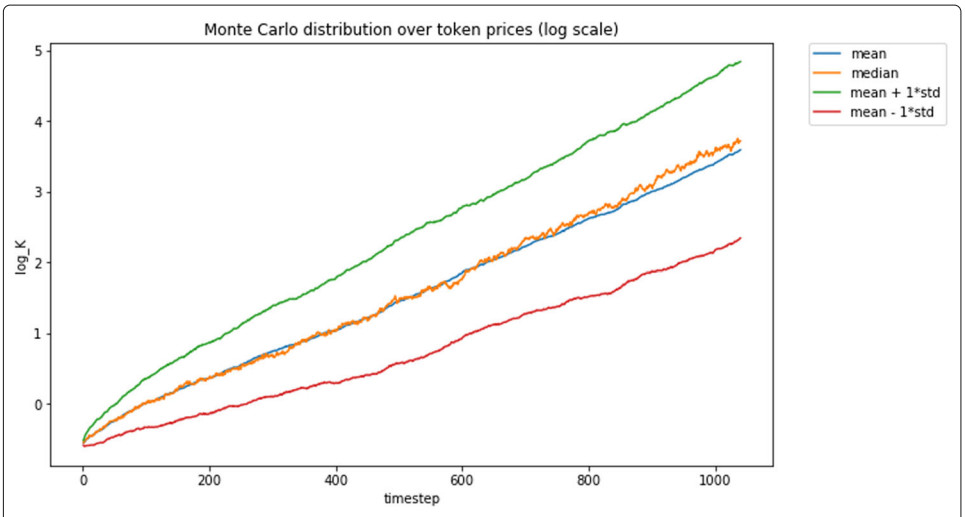

Fig. 3 Token prices over 1,040 weeks; statistics from 100 Monte Carlo Runs under smooth exponential decay block reward schedule on log scale

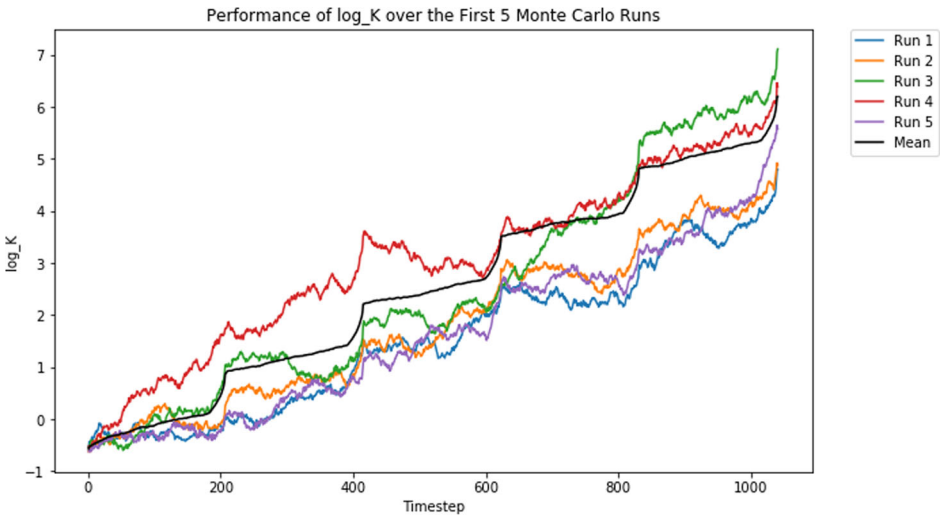

Fig. 4 Token prices over 1,040 weeks; 5 realizations under step decay block reward schedule on log scale

stochastic dynamical systems. These tools and theories are powerful in understanding and designing complex token networks.

Conclusion and future considerations

This paper builds a bridge between dynamical systems theory and blockchain-enabled economic systems by proposing a state space representation of the economic system in terms of linear time-expanding system. This novel representation allows us to use a plethora of powerful tools developed in the context of control theory for the analysis and design of blockchain-enabled systems. A baseline simulation based on open-loop block reward was introduced to illustrate the power of combining our approach with computational methods. Given our general modeling framework, we can further explore

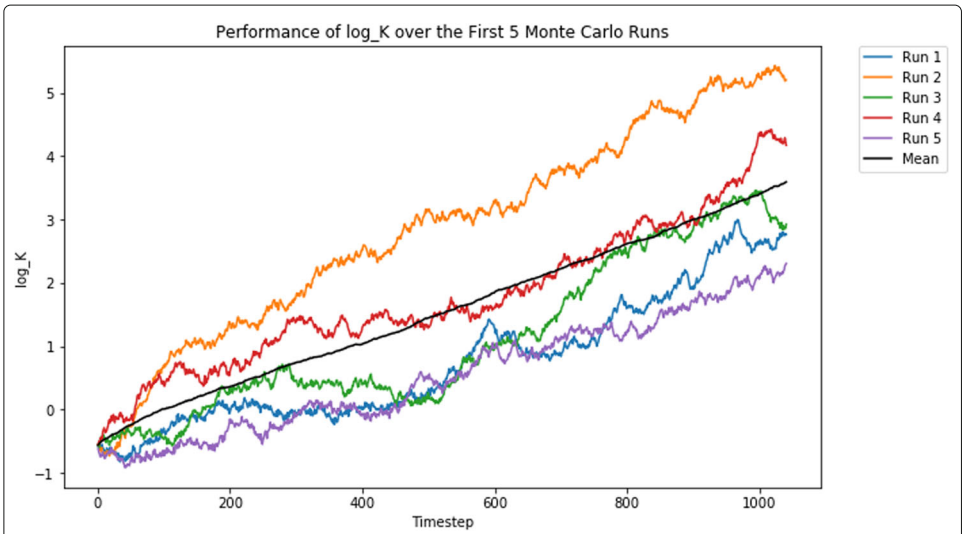

Fig. 5 Token prices over 1,040 weeks; 5 realizations under smooth exponential decay block reward schedule on log scale

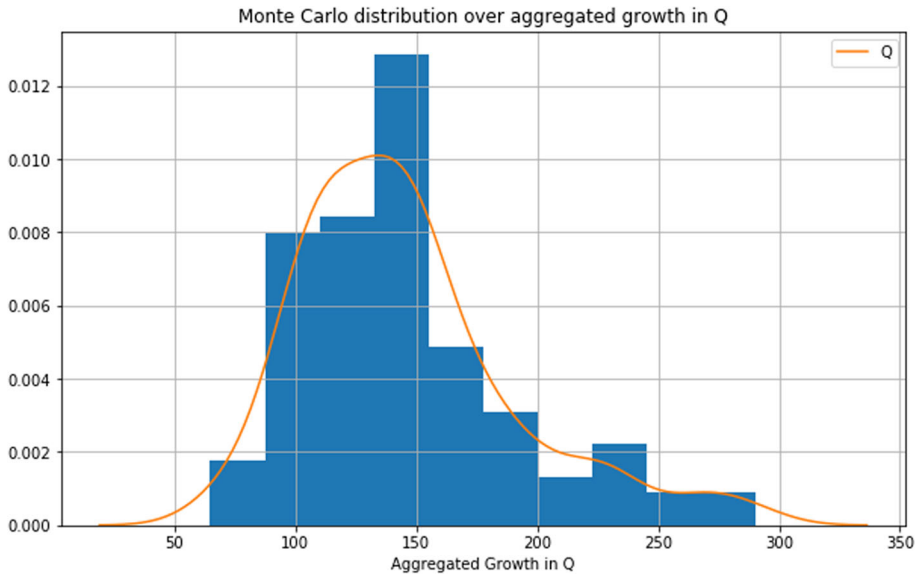

Fig. 6 Distribution of aggregated growth in Q over 100 Monte Carlo Runs under step decay block reward schedule

Lyapunov-like functions, commonly used in control theory (Araujo and Moreira 2014; Lechevin and Rabbath 2004), to provide greater guarantee on network economic robustness and security (Smith 1984; Park et al. 2019). Systems should be input-output stable under an *energy* function, (Klages-Mundt and Minca 2019). Similar methods are used for robotic systems in Olfati-Saber and Murray (2002). Lastly, it will also be interesting to further this research through the lens of an agent (Monnot and Piliouras 2017). Given the states and dynamics of the system, we hope to understand and compute optimal strategies for a population of agents to maximize their gains given some constraints. Classic optimal

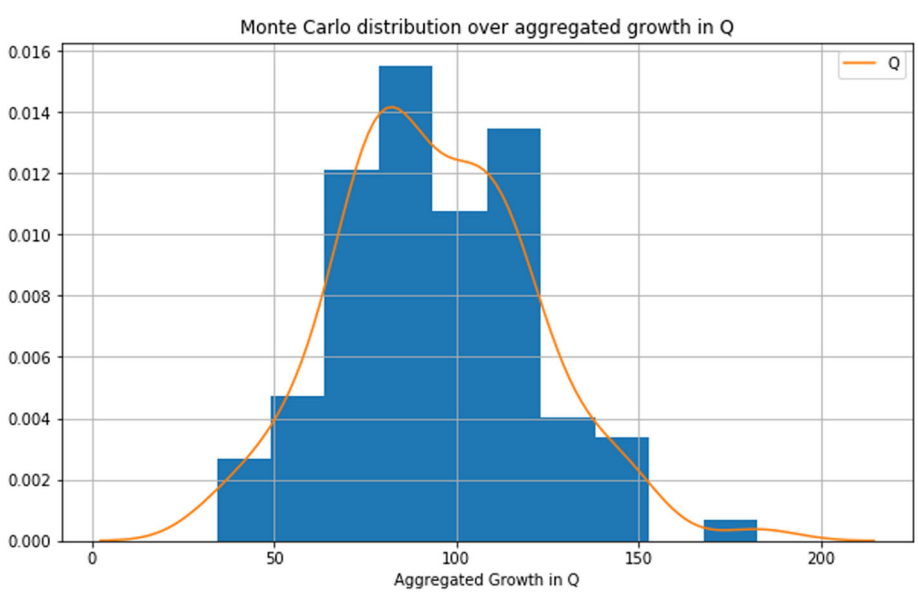

Fig. 7 Distribution of aggregated growth in Q over 100 Monte Carlo Runs under smooth exponential decay block reward schedule

control theory with discounted future utility (or some other notion of a terminating condition) will be applicable (Bertsekas 1995). From an economic research perspective, the approach presented has addressed the Lucas critique in macroeconomics (Lucas 1976) as it started from agent-level incentives, evolved into dynamics between states, actions, and mechanisms, and then derived global emergent properties from these underlying structures and functional relationships. More details on how the presented approach addresses the critique can be found in Zargham et al. (2020).

Appendix

Proofs

Proof of theorem 1

Proof Since the genesis block contained an empty state, these requirements would make for a trivial trajectory. Hence, a driving function $M(k) = \mu_k \nu(k)$ was introduced in (13) and reproduced here:

$$x(k+1) = A_k x(k) + B_k u(k) + \mu_k \nu(k). \tag{28}$$

The function $M(k) \in \mathbb{R}^{n_{k+1}}$ is decomposed into a scheduled positive scalar reward $\mu_k \in \mathbb{R}_+$ and a stochastic vector $\nu(k) \in \mathbb{R}_+^{n_{k+1}}$ such that $\sum_i \nu_i(k) = 1$. The vector $\nu(k)$ denotes the distribution of the mining rewards across all accounts including potential allocation to new accounts or may be distributed by any arbitrary rules across an arbitrary subset of accounts, such as a mining pool. Another key property of the Bitcoin is again recovered from our state space model. Define a scalar subspace of the state

$$y(k) = \mathbf{1}' x(k) = \sum_i x_i(k). \tag{29}$$

We want to prove that the following equation holds true

$$y(K) = \sum_{k=1}^{K} \mu_k. \tag{30}$$

To prove that, we first rearrange (13) to the following,

$$x(k) - A_{k-1} x(k-1) = B_{k-1} u(k-1) + \mu_{k-1} \nu(k-1) \tag{31}$$

and construct the state by summing the history of changes

$$\begin{aligned} x(K) =& A_{K-1} \cdots A_0 x(0) \\ &+ \sum_{k=1}^{K-1} \Pi_{i=k}^{K-1} A_i \left(x(k) - A_{k-1} x(k-1) \right) \end{aligned} \tag{32}$$

becomes

$$\begin{aligned} x(K) =& A_{K-1} \cdots A_0 x(0) \\ &+ \sum_{k=1}^{K-1} \Pi_{i=k}^{K-1} A_i \left(B_{k-1} u(k-1) + \mu_{k-1} \nu(k-1) \right). \end{aligned} \tag{33}$$

When this expression is used to compute

$$\begin{aligned} y(k) =& \mathbf{1}' x(K) \\ =& \mathbf{1}' A_{K-1} \cdots A_0 x(0) \\ &+ \sum_{k=1}^{K-1} \mathbf{1}' \Pi_{i=k}^{K-1} A_i \left(B_{k-1} u(k-1) + \mu_{k-1} v(k-1) \right). \end{aligned} \tag{34}$$

Since $x(0) = 0$, it follows that

$$y(K) = \sum_{k=1} \mu_k \tag{35}$$

when one recalls that A_k is an augmented identity matrix, that $\mathbf{1}' v(k) = 1$ observes that B_k is an incidence matrix: $\mathbf{1}' B u = 0$ for all u.

In the case of Bitcoin the mining rewards are on a convergent schedule ensuring the maximum total supply

$$y_\infty = \lim_{k \to \infty} y(k) = \sum_{k=1}^{\infty} \mu_k \tag{36}$$

converges to the desired quantity. □

Acknowledgements
This work would not be possible without the BlockScience team developing and open-sourcing cadCAD simulation engine.

Authors' contributions
MZ developed the mathematical framework and advised ZZ on the model design. ZZ carried out the simulations, analyzed the data, and wrote the manuscript with input from MZ and VP. VP provided network science domain expertise and helped supervise the project. All authors read and approved the final manuscript.

Funding
BlockScience Academic Research Program.

Availability of data and materials
Simulation engine: https://github.com/BlockScience/cadCAD. Simulation result: https://github.com/zixuanzh/ans.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, USA. [2]BlockScience Inc, Oakland, USA.

Received: 21 September 2019 Accepted: 24 January 2020
Published online: 19 March 2020

References

Alexandre B, Campajola C, Lazo JF, Mottes F, Pozzana I, Restocchi V, Saggese P, Vallarano N, Squartini T, Tessone CJ (2018) Network-based indicators of Bitcoin bubbles. arXiv:1805.04460
Araujo RA, Moreira HN (2014) Lyapunov stability in an evolutionary game theory model of the labour market. Economica. https://doi.org/10.1016/j.econ.2014.03.006
Bachelier L (2011) Louis Bachelier's theory of speculation: the origins of modern finance. Princeton University Press. https://doi.org/10.1515/9781400829309
Back A, et al. (2002) Hashcash-a denial of service counter-measure. http://www.hashcash.org/papers/hashcash.pdf. Accessed 9 Mar 2016
Ball M, Rosen A, Sabin M, Vasudevan PN (2017) Proofs of Useful Work. http://eprint.iacr.org/2017/203.pdf. Accessed 29 June 2017
Bentov I, Gabizon A, Mizrahi A (2014) Cryptocurrencies without proof of work. http://arxiv.org/pdf/1406.5694.pdf. Accessed 9 Mar 2016
Bertsekas DP (1995) Dynamic programming and optimal control. Vol 1, No. 2. Athena scientific, Belmont
Bimpikis K, Candogan O, Saban D (2019) Spatial pricing in ride-sharing networks. Oper Res. https://doi.org/10.1287/opre.2018.1800
BlockScience (2019) cadCAD: a differential games based simulation software package for research, validation, and Computer Aided Design of economic systems. https://doi.org/https://github.com/BlockScience/cadCAD

Borowski H, Marden JR (2015) Fast convergence in semianonymous potential games. IEEE Trans Control Netw Syst 4.2:246–258
Bovet A, Campajola C, Mottes F, Restocchi V, Vallarano N, Squartini T, Tessone CJ The evolving liaisons between the transaction networks of Bitcoin and its price dynamics. arXiv preprint. arXiv:1907.0357
Buterin V (2014) Ethereum: A next-generation smart contract and decentralized application platform. https://github.com/ethereum/wiki/wiki/White-Paper. Accessed 22 Aug 2016
Buterin V, Griffith V (2017) Casper the Friendly Finality Gadget. https://arxiv.org/pdf/1710.09437.pdf. Accessed 6 Nov 2017
Castro M, Liskov B, et al (1999) Practical byzantine fault tolerance. In: OSDI, vol. 99. pp 173–186. http://pmg.csail.mit.edu/papers/osdi99.pdf
Chu J, Saralees N, Chan S (2015) Statistical analysis of the exchange rate of bitcoin. PloS ONE 10(7). https://doi.org/10.1371/journal.pone.0133678
Dai W (1998) bmoney. http://www.weidai.com/bmoney.txt. Accessed 31 Apr 2016
David G, Tessone CJ, Mavrodiev P, Perony N (2014) The digital traces of bubbles: feedback cycles between socio-economic signals in the bitcoin economy. J R Soc Interface 11(99). https://doi.org/10.1098/rsif.2014.0623
Dimson E, Massoud M (1999) Three centuries of asset pricing. J Bank Finance. https://doi.org/10.2139/ssrn.203108
Dockner EJ, Jorgensen S, Long NV, Sorger G (2000) Differential games in economics and management science
Douceur JR (2002) The sybil attack. In: International Workshop on Peer-to-Peer Systems. Springer. pp 251–260. http://www.cs.cornell.edu/people/egs/cs6460-spring10/sybil.pdf. Accessed 1 Nov 2018
ElBahrawy A, Alessandretti L, Kandler A, Pastor-Satorras R, Baronchelli A (2017) Evolutionary dynamics of the cryptocurrency market. R Soc Open Sci 4(11). https://doi.org/10.1098/rsos.170623
Filecoin (2014) A Cryptocurrency Operated File Storage Network. http://filecoin.io/filecoin.pdf. Accessed 14 Oct 2014
Forrester JW (1971) Counterintuitive behavior of social systems. Technol Forecast Soc Chang 3:1–22
Foster J (2005) From simplistic to complex systems in economics. Camb J Econ 29(6):873–892
Foster J, Metcalfe JS (2012) Economic emergence: An evolutional economic perspective. J Econ Behav Organ 82(2):420–432
Göbel J, Keeler P, Krzesinski AE, Taylor PG (2015) Bitcoin Blockchain Dynamics: the Selfish-Mine Strategy in the Presence of Propagation Delay. http://arxiv.org/pdf/1505.05343.pdf. Accessed 1 Mar 2015
Gregory MN (2014) Principles of Economics. Cengage Learn. https://doi.org/10.24926/8668.1601
Guzman MAC, Mojica-Nava E (2017) Mechanism design for demand response programs with financial and non-monetary (social) incentives. arXiv preprint. arXiv:1709.10122
Hayek FA (1948) The meaning of competition. Individualism Econ Order: 92–106
Heilman E, Kendler A, Zohar A, Goldberg S (2015) Eclipse attacks on bitcoin's peer-to-peer network. In: 24th USENIX Security Symposium (USENIX Security 15). pp 129–144. https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-heilman.pdf. Accessed 1 Nov 2018
Isaacs R (1999) Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization. https://doi.org/10.2307/2343511
Keynes JM (1964) The general theory of employment, interest and money. Collected Writings John Maynard Keynes. https://doi.org/10.1007/978-3-319-70344-2_1
Khalil HK, Grizzle J (2002) Nonlinear systems. Prentice hall, Englewood Cliffs, NJ
Klages-Mundt A, Minca A (2019) (In) Stability for the Blockchain: Deleveraging Spirals and Stablecoin Attacks. arXiv preprint. arXiv:1906.02152
Lamport L, Shostak R, Pease M (1982) The byzantine generals problem. ACM Trans Program Lang Syst (TOPLAS) 4(3):382–401
Larimer D (2014) Delegated proof-of-stake (DPOS). Bitshare whitepaper
Lasry J-M, Lions P-L (2007) Mean field games. Jpn J Math 2(1):229–260
Lechevin N, Rabbath CA (2004) Lyapunov-based nonlinear missile guidance. J Guid Control Dyn. https://doi.org/10.2514/1.8629
Lucas RE (1976) Econometric policy evaluation: A critique. Carnegie-Rochester Conf Ser Publ Policy 1:19–46. https://doi.org/10.1016/S0167-2231(76)80003-6
Lux T (2008) Applications of statistical physics in finance and economics. https://doi.org/10.4337/9781781952665.00017
Mabrok MA, Shamma JS (2016) Passivity Analysis of Higher Order Evolutionary Dynamics and Population Games. In: IEEE 55th Conference on Decision and Control (CDC). https://doi.org/10.1109/cdc.2016.7799211
Maker (2017) The Dai Stablecoin System. https://makerdao.com/whitepaper/DaiDec17WP.pdf. Accessed 1 Nov 2018
Malkiel BG (1989) Efficient market hypothesis. Finance:127–134. https://doi.org/10.1007/0-387-26336-5_735
Marden JR (2012) State based potential games. Automatica. https://doi.org/10.1016/j.automatica.2012.08.037
Marden JR, Shamma JS (2015) Game theory and distributed control. In: Handbook of game theory with economic applications. Vol. 4. Elsevier. pp 861–899
Monnot B, Piliouras G (2017) Limits and limitations of no-regret learning in games. Knowl Eng Rev 32. https://doi.org/10.1017/s0269888917000133
Muntz RR (1972) Poisson departure processes and queueing networks. IBM Thomas J. Watson Research Center
Nakamoto S (2008) Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf. Accessed 1 July 2015
Nisan N, Roughgarden T, Tardos E, Vazirani VV (2007) Algorithmic game theory. Cambridge University Press, Cambridge
Olfati-Saber R, Fax JA, Murray RM (2007) Consensus and cooperation in networked multi-agent systems. In: Proceedings of the IEEE. https://doi.org/10.1109/jproc.2006.887293
Olfati-Saber R, Murray RM (2002) Distributed cooperative control of multiple vehicle formations using structural potential functions. IFAC Proc Vol 35(1):495–500. https://doi.org/10.3182/20020721-6-ES-1901.00244. 15th IFAC World Congress
Park S, Martins NC, Shamma JS (2019) Payoff Dynamic Models and Evolutionary Dynamic Models: Feedback and Convergence to Equilibria. arXiv preprint. arXiv:1903.02018
Peters O, Adamou A (2018) Ergodicity Economics. London Mathematical Laboratory
Petkanic D, Tang E (2017) Livepeer Whitepaper: Protocol and Economic Incentives for a Decentralized Live Video Streaming Network

Ragavendran G, Marden JR, Wierman A (2011) An architectural view of game theoretic control. ACM SIGMETRICS Perform Eval Rev. https://doi.org/10.1145/1925019.1925026
Rainer B, Christin N, Edelman B, Moore T (2015) Bitcoin: Economics, technology, and governance. J Econ Perspect 29(2):213–38
Roth AE (2002) The economist as engineer: Game theory, experimentation, and computation as tools for design economics. Econometrica 70(4):1341–1378
Sahneh FD, Scoglio C, Mieghem PV (2013) Generalized epidemic mean-field model for spreading processes over multilayer complex networks. IEEE/ACM Trans Netw. https://doi.org/10.1109/TNET.2013.2239658
Shorish J (2018) Blockchain State Machine Representation. SocArXiv. https://doi.org/10.31235/osf.io/eusxg
Smith MJ (1984) The stability of a dynamic model of traffic assignment—an application of a method of lyapunov. Transp Sci 18(3):245–52
Sontag ED (2013) Mathematical control theory: deterministic finite dimensional systems. Vol 6. Springer Science & Business Media
Sterman J (2002) System Dynamics: systems thinking and modeling for a complex world
Szabo N (2005) Bit gold. http://unenumerated.blogspot.co.at/2005/12/bit-gold.html. Accessed 31 Apr 2016
Voshmgir S (2019) Token Economy
Voshmgir S, Zargham M (2019) Foundations of cryptoeconomic systems. Working Paper Series / Institute for Cryptoeconomics / Interdisciplinary Research 1, Research Institute for Cryptoeconomics, Vienna. https://epub.wu.ac.at/7309/
Zargham M, Ribeiro A, Ozdaglar A, Jadbabaie A (2013) Accelerated dual descent for network flow optimization. IEEE Trans Autom Control 59(4):905–920
Zargham M, Shorish J, Paruch K (2020) From Curved Bonding to Configuration Spaces. In: IEEE International Conference on Blockchain and Cryptocurrency. https://epub.wu.ac.at/7385/
Zargham M, Zhang Z, Preciado V (2018) A State-Space Modeling Framework for Engineering Blockchain-Enabled Economic Systems. arXiv preprint arXiv:1807.00955
Zhang X-Z, Liu J-J, Xu Z-W (2015) Tencent and Facebook data validate Metcalfe's Law. J Comput Sci Technol. https://doi.org/10.1007/s11390-015-1518-1

Publisher's Note