A Control Theoretic Approach to Infrastructure-Centric Blockchain Tokenomics

Oguzhan Akcin, Robert P. Streit, Benjamin Oommen,
Sriram Vishwanath, and Sandeep Chinchali

The University of Texas at Austin {oguzhanakcin, rpstreit,
baoommen, sriram, sandeepc}@utexas.edu

Abstract. There are a multitude of Blockchain-based physical infrastructure systems, ranging from decentralized 5G wireless to electric vehicle charging networks. These systems operate on a crypto-currency enabled token economy, where node suppliers are rewarded with tokens for enabling, validating, managing and/or securing the system. However, today's token economies are largely designed without infrastructure systems in mind, and often operate with a fixed token supply (e.g., Bitcoin). Such fixed supply systems often encourage early adopters to hoard valuable tokens, thereby resulting in reduced incentives for new nodes when joining or maintaining the network. This paper argues that token economies for infrastructure networks should be structured differently – they should *continually incentivize* new suppliers to join the network to provide services and support to the ecosystem. As such, the associated token rewards should gracefully scale with the size of the decentralized system, but should be carefully balanced with consumer demand to manage inflation and be designed to ultimately reach an equilibrium. To achieve such an equilibrium, the decentralized token economy should be *adaptable* and controllable so that it maximizes the total utility of all users, such as achieving stable (overall non-inflationary) token economies.

Our main contribution is to model infrastructure token economies as *dynamical systems* – the circulating token supply, price, and consumer demand change as a function of the payment to nodes and costs to consumers for infrastructure services. Crucially, this dynamical systems view enables us to leverage tools from mathematical control theory to optimize the overall decentralized network's performance. Moreover, our model extends easily to a Stackelberg game between the controller and the nodes, which we use for robust, strategic pricing. In short, we develop predictive, optimization-based controllers that outperform traditional algorithmic stablecoin heuristics by up to $2.4\times$ in simulations based on real demand data from existing decentralized wireless networks.

Keywords: Blockchain Token Economics · Optimal Control Theory · Game Theory

1 Introduction

The space of Blockchain-based physical infrastructure networks is rapidly growing, including decentralized wireless, storage, compute, and electric vehicle charg-
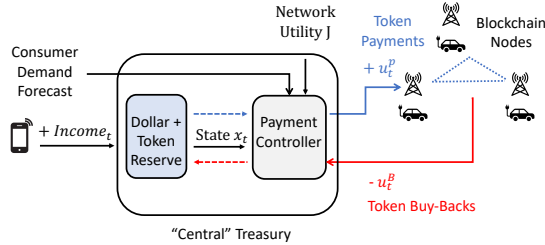
Fig. 1: A Control System for Blockchain Tokenomics: We design a controller (gray) to achieve a burn and mint equilibrium. The controller takes in a forecast of consumer demand and supplier growth s_t, as well as the treasury state x_t. Then, it adaptively controls token payments u_t^P and buy-backs u_t^B to achieve a stable token price.

ing networks. As an example, Helium [18] and Pollen [22] are two prominent decentralized wireless networks (DeWi) that reward the general public to build, maintain, validate, secure and ultimately, send data over 5G hotspots. Similarly, projects such as FileCoin [20], Storj [21] and ComputeCoin [27] offer decentralized file storage and computing services. These networks reward suppliers using a corresponding (cryptocurrency) token to build, maintain, secure, and offer services over this decentralized infrastructure network. Likewise, consumers can often exchange US dollars (USD) for tokens, which enables them to utilize infrastructure services and/or participate in the associated crypto-economy.

Despite the popularity of decentralized infrastructure networks, we lack systematic tools to design their token economies to incentivize supply growth and consumer demand. Today's token economies largely target finance, such as Bitcoin, and can operate with a (typically) fixed supply of tokens. However, these fixed supply monetary systems are starkly different from physical infrastructure networks. For example, in a fixed supply system such as Bitcoin, early adopters can hoard tokens since they are scarce. Moreover, late adopters might not be adequately incentivized to join or maintain the network as token rewards could prove to be smaller than those of early participants.

Our central thesis is that a token economy must be designed to continually incentivize new suppliers to join the ecosystem and provide services, such as 5G connectivity for Helium or electric vehicle charging stations. As such, the number of tokens should gracefully scale with the size of the infrastructure network, which we do not know a-priori. However, continually rewarding suppliers with newly created tokens can result in inflation if such payments are not carefully balanced with the consumer demand for infrastructure services. To solve such problems, a number of projects have recently considered adopting/adopted a "burn-and-mint" token economics (tokenomics) model, where a central reserve "mints" tokens to reward suppliers, while tokens are "burnt" (deleted from the circulating supply) when consumers want to use network services. By adaptively burning tokens, we can reduce the token supply to reach an overall supply-demand equilibrium.

Moreover, such a burn-and-mint equilibrium (BME) [1] must be "programmable" so that Blockchain-based infrastructure networks can maximize the total utility of all users. For example, this network utility function (performance criterion)

can include maintaining a stable, steadily growing token price with low volatility. Likewise, this network cost function can incentivize new suppliers/consumers to expand geographical coverage. Moreover, the BME-based token economy could be designed to satisfy strict performance guarantees and constraints, such as limiting the number of tokens minted and/or burned per day. Taking this even further, participants in the economy are likely rational and so it is important to consider their agency – and any impacts – in taking actions to maximize the value of their holdings. In short, solutions deployed in infrastructure-centric Blockchain networks must address these aspects in their design when managing token supply.

Our key insight is that token economies can be modeled as dynamical systems, which allows us to leverage powerful ideas from **mathematical control theory** to maximize a Blockchain network's utility function under chosen constraints. Control theory is a natural tool since the token economy is a dynamical system – the circulating token supply, token price, and consumer demand change as a function of our burn and mint decisions. Likewise, we have control authority – we are able to adapt the burn or mint mechanisms to regulate the token economy. Moreover, we can design a control cost function that captures key metrics for desired performance and evolution of the Blockchain dynamical system. Crucially, we can model the dynamics of the system, since we engineer the Blockchain protocol and token economy dynamics. As such, regulating the Blockchain token economy is a model-based control problem, which can be solved using powerful ideas from nonlinear optimization and optimal control theory.

Overall, the contributions of this paper are three-fold. To the best of our knowledge, we are the first to apply optimal control theory to Blockchain tokenomics and introduce a general-purpose dynamical systems model that flexibly captures both fixed-supply as well as burn-and-mint systems. We design a control system for a token economy using nonlinear model predictive control (MPC) methods that are used in high-performance, safety-critical applications like autonomous driving [34,7], robotic manipulation, and rocket guidance [4]. We demonstrate that these methods perform better than common heuristic controllers, such as proportional integral derivative (PID) controllers used by some algorithmic stablecoins. Specifically, we improve on PID by $2.4\times$ on simulated timeseries demand patterns and by $2.7\times$ on real demand patterns from the Helium DeWi Blockchain. Finally, we introduce a novel game-theoretic formulation for how owners of tokens and a central reserve strategically interact to maximize network welfare.

Related Work: Generally, prior research on Blockchains as dynamical sytems [35,36,10] focus on miner profitability and on the influence of Block rewards on supply and demand dynamics. Our work differs from existing literature in that we focus on understanding incentives and equilibrium in infrastructure-centric Blockchain systems. Specifically, in our case, the supply is fully specified by the actions of the controller, while the demand is specified via forecasts. Thus, our controller specification is decoupled from the possibly complex trajectory of the

demand, and the strength of our controller's predictions relates with the strength of the forecasts used in the system.

In order to better understand the robustness of our methodology, we also consider the impact of rational behavior on the part of the consumers in our system. To achieve this, game theoretic analyses in Blockchain systems have been used over many years, starting with the original Bitcoin whitepaper [26]. Since the discovery of the selfish mining attack [16], game theoretic methods have been used to investigate rational deviations [12], mining pools [15], and more recently transaction fee auctions in Ethereum like Blockchains [31,13]. Our work differs from existing literature as we focus on the effects of rational behavior on buy-back and pay strategies used to stabilize token prices, and not necessarily on modeling the effects on an underlying Blockchain protocol.

Finally, as our aim is to stabilize a token price in a Blockchain network, our work bears a degree of similarity to algorithmic stable-coins. However, our interests are in intelligently controlling the circulating supply of a token to balance payments to service providers needed to scale a network with a pre-specified control trajectory on the token price. Thus, our work is more related to service networks employing burn and mint systems such as Helium [18] (which inspired our model) and Factom [32] than more general purpose stable-coins like Reflexer [2] or Terra [19]. Furthermore, most existing literature is *reactive* through the use of heuristic methods such as PID, whereas our work is *predictive* through optimal adaptive control methods. As our focus is on infrastructure networks, our work is applicable to DeWi [24] scenarios like Helium [18], as well as file sharing [20] and decentralized video streaming [30].

2   A Primer on Optimal Control

We now provide a basic primer on optimal control theory, which enables us to naturally model the token economy as a controlled dynamical system. Using this, we describe the state of a dynamical system, the control inputs, dynamics, and the high-level performance criterion (cost function).

The state vector is denoted by $x_t \in \mathbb{R}^n$, the control vector by $u_t \in \mathbb{R}^m$, and the dynamics are given by $x_{t+1} = f(x_t, u_t, s_t)$, where s_t is an exogenous timeseries input, such as a demand forecast. Since we have forecasts of node and consumer growth for H steps in the future, we naturally have a finite horizon control problem of H steps. Our goal is to optimize the performance metric, which is to minimize the aggregate control cost J. Typically, the control cost J is a sum of a terminal cost and stage costs penalizing state deviations from a reference trajectory (tracking error) and control effort, of the form $J = c_H(x_H) + \sum_{t=0}^{H-1} c_t(x_t, u_t)$, where $c_t(x_t, u_t)$ is a possibly time-variant cost.

Crucially, we have a good nominal model of the dynamics, since the token economy is under our design. Of course, there are uncertainties which arise due to the stochastic demand forecast s_t. Since we have known nominal dynamics, we use standard *model-based* control techniques, which solve an optimization problem to find the optimal set of controls to minimize the cost function subject

to dynamics constraints [11,9]. Moreover, we often constrain the state x_t and control u_t to lie in sets $\mathcal{X}$ and $\mathcal{U}$ respectively to capture, for example, strict actuation limits. Thus, the general optimal control problem can be stated as:

$$
\begin{aligned}
\underset{u_{0:H-1}}{\text{minimize}} \quad & \underset{s_{0:H-1}}{\mathbb{E}} \left[J(x_0, u_{0:H-1}; s_{0:H-1}) \right], \\
\text{subject to} \quad & x_{t+1} = f(x_t, u_t, s_t) && \forall t \in \{0, \ldots, H-1\} \\
& x_t \in \mathcal{X} && \forall t \in \{0, \ldots, H\} \\
& u_t \in \mathcal{U} && \forall t \in \{0, \ldots, H-1\},
\end{aligned}
\tag{1}
$$

where x_0 is the initial state and we plan for H steps given a stochastic demand forecast $s_{0:H-1}$. We now model the token economy in Sections 3 and 4.

3   The Token Economy as a Dynamical System

To model the token economy for infrastructure networks as a controlled dynamical system, we define our problem's state x_t, controls u_t, exogenous forecast s_t, dynamics $f(x_t, u_t, s_t)$, and cost function J. For the dynamics, we show the relationships between the token's circulating supply, price, and reserve used by the controller. As shown in Fig. 1, these quantities will form our state variables which are controlled via buying back tokens and paying tokens to the user base.

We capture the following interaction: *Nodes* provide services (5G base stations, EV charging, etc.) for *consumers*. The consumers pay dollars to the controller to use these services, and the controller converts these dollars into tokens at market price to pay to the nodes as rewards for their services. This reward payment is the *minting* action. Furthermore, the dollars received from consumers is *income* that is placed in a dollar reserve. Funds from this reserve are in turn used to buy back tokens from the nodes at a price posted by the controller. The bought back tokens are removed from the circulating supply and placed into their own reserve, which is effectively the *burn* action. We model the token economy as a closed system, meaning that the power to change the circulating supply is endowed only in the buy back and pay mechanism and no exogenous actor. Effectively, this means that no node will intentionally burn their own tokens and lose assets. Finally, we examine the token economy as a discrete time system.

Before proceeding, one more comment: Recall, as in Fig. 1, the Blockchain network has a reserve that mints new tokens and buys back tokens from the market. Our controller resides in this reserve, and can be considered as logically "centralized". Of course, the control logic is implemented by *distributed* nodes running smart contracts. However, all the smart contracts come to a consensus on the key state variables, which are aggregate (not node-specific) quantities such as the circulating supply, token price, total consumer demand etc. As such, the nodes can decide how much to pay/buy-back in a decentralized way.

Circulating Supply   The circulating supply of tokens S_t increases when u_t^{P} tokens are paid as rewards to nodes and decreases when tokens are bought back:

$$
S_{t+1} = S_t + \underbrace{u_t^{\mathrm{P}}}_{\text{Tokens Paid}} - \underbrace{\frac{u_t^{\mathrm{B}}}{p_t^{\mathrm{Tok}} + \Delta p_t}}_{\text{Tokens Bought Back}} .
\tag{2}
$$

Here, u_t^{B} is the number of dollars the controller pays to active token owners to purchase their tokens. This means the number of tokens bought back in time t is u_t^{B} divided by the purchasing price $(p_t^{\mathrm{Tok}} + \Delta p_t)$. We define p_t^{Tok} as the current market price of the token, while Δp_t is the extra amount the controller pays the users over the market price so as to incentivize them to part with their tokens. If the token owners are not rational, Δp_t simply equals zero. However, a rational owner may not part with their tokens if they believe the price will increase in successive timesteps, and so the controller may need to pay an added incentive price Δp_t to overcome the agents' beliefs. We formulate a mathematical game to calculate Δp_t for rational agents in Section 5.

Reserve The controller will hold two reserves, one being comprised of tokens and the other dollars. At time t, the values of these reserves are given by R_t^{Tok} and R_t^{USD} respectively. The dollar reserve increases with the income received from users purchasing tokens, and decreases with buy backs from the market:

$$R_{t+1}^{\mathrm{USD}} = R_t^{\mathrm{USD}} - u_t^{\mathrm{B}} + \mathrm{Inc}_t, \tag{3}$$

where u_t^{B} is the dollars used to buy back tokens at time t, while the new variable Inc_t is the income received from users purchasing tokens. Inc_t is a linear function of the number of consumers in the system, since the consumers pay a fixed dollar amount for a unit of infrastructure service. For example, in DeWi networks, a consumer pays a fixed number of USD for each gigabyte of data [18]. Analagous to the dollar reserve, the token reserve R_t^{Tok} increases with buy backs and decreases with payments made to the service providers, i.e.,

$$R_{t+1}^{\mathrm{Token}} = R_t^{\mathrm{Tok}} - u_t^{\mathrm{P}} + \frac{u_t^{\mathrm{B}}}{p_t^{\mathrm{Tok}} + \Delta p_t}. \tag{4}$$

Token Price The token price at time t is given by p_t^{Tok}, and we assume it is market clearing. Thus, for supply S_t and demand D_t at time t, we have:

$$p_t^{\mathrm{Tok}} = \frac{D_t}{S_t}.$$

The demand for tokens is an (unspecified) function of the number of consumers in the system n_t and the price of the token p_t^{Tok}. We assume that for a fixed price, demand is proportional to the number of consumers, and therefore for a fixed token price, $\mathrm{Inc}_t \propto D_t$. In practice, demand can be forecasted using historical data, as shown for public Helium DeWi data in Section 6.

Remark 1. Vanilla buy back and pay strategies, like those in many operational BME protocols, reach an equilibrium. This is a strategy entirely clearing its income in each step, i.e.,

$$\mathrm{Inc}_t = u_t^{\mathrm{B}}, \qquad\qquad u_t^{\mathrm{P}} = \frac{u_t^{\mathrm{B}}}{p_t^{\mathrm{Tok}} + \Delta p_t}.$$

Then, by verifying Eqs. 2–4, one sees that the circulating supply remains invariant, $S_t = S_{t-1} = \ldots = S_0$, as do the reserves R_t^{USD} and R_t^{Tok}.

Why Control? While the simple strategies in Remark 1 are stable, they do not provide much-needed flexibility for tokenomics. For example, we might need to increase the token dollar reserve to pay for upgrades, which requires saving income instead of directly paying it all to nodes. Further, we might want the token price to steadily grow at a certain rate, so that nodes can sell tokens to recover their capital investment costs (e.g., buying a 5G hotspot). In essence, we need control to flexibly steer the system away from the equilibrium in Remark 1 to achieve a high-level performance criterion, as formalized next.

State, Control and Cost Variables: The state x_t captures the dynamic quantities that are necessary to control the system. Likewise, the control vector u_t consists of how much we adaptively pay, buy-back, and our incentive price:

$$x_t = \begin{bmatrix} S_t, R_t^{\text{USD}}, R_t^{\text{Tok}}, p_t^{\text{Tok}} \end{bmatrix}^\top, \qquad u_t = \begin{bmatrix} u_t^{\text{B}}, u_t^{\text{P}}, \Delta p_t \end{bmatrix}^\top.$$

Additionally, as our method is *predictive*, we use forecasts predicting the future income and consumer demand. In practice, these can come from data-driven modeling using historical transaction data. The forecasts at time t are:

$$s_t = \begin{bmatrix} \widehat{D_t}, \widehat{\text{Inc}_t} \end{bmatrix}^\top.$$

Now, we can write the nonlinear dynamics of our system using Eqs. 2–4:

$$x_{t+1} = f(x_t, u_t, s_t) = \begin{bmatrix} x_t(0) + u_t(1) - \frac{u_t(0)}{x_t(3)+u_t(2)} \\ x_t(1) + s_t(1) - u_t(0) \\ x_t(2) + \frac{u_t(0)}{x_t(3)+u_t(2)} - u_t(1) \\ \frac{s_{t+1}(0)}{x_{t+1}(0)} \end{bmatrix}. \tag{5}$$

Cost Function The cost function J encodes the token economy's high-level performance criterion. For example, we often want the price to follow a smooth, steadily increasing reference trajectory denoted by $\bar{p}_t^{\text{Tok, ref.}}$. Thus, we penalize the L_2 norm difference between the real price p_t^{Tok} and the desired reference price $\bar{p}_t^{\text{Tok, ref.}}$. Likewise, we might want to penalize the enacted controls u_t from a nominal, reference set of controls $\bar{u}_t$. In practice, these reference controls can come from the income-clearing strategy in Remark 1 or be computed to achieve the reference price trajectory in the absence of forecasting errors. Thus, our control cost function is:

$$J(x_0, u_{0:H-1}) = \sum_{t=0}^{H} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}^\top \begin{bmatrix} (x_t(3) - \bar{x}_t(3))^2 \\ (u_t(0) - \bar{u}_t(0))^2 \\ (u_t(1) - \bar{u}_t(1))^2 \end{bmatrix}. \tag{6}$$

Blockchain designers can flexibly set parameters $\beta_1, \beta_2, \beta_3$ to trade off how closely we track the reference price and our control effort. Our formulation is extremely general, since we can easily follow a reference token supply that scales with the number of nodes, or even a fixed reference token supply. Indeed, our formulation encompasses fixed supply systems, since we can implement an upper bound on the number of tokens (state constraints). Likewise, we can use any differentiable, non-convex cost function amenable to gradient-based optimization.

State/Control Constraints We now define the feasible state set $\mathcal{X}$ and control set $\mathcal{U}$. First, each element in the state x_t should be non-negative and ideally above a safety margin, since they represent the circulating supply, token dollar reserves etc. For example, we might want the dollar reserve above a positive safety margin, specifically $R_t^{\text{USD}} \geq R_t^{\text{USD,min}}$. Likewise, we want to pay and buy back a positive number of tokens, given by $u_t^{\text{P}}, u_t^{\text{B}} > 0$. Finally, we have no constraints on Δp_t, since we can offer to buy back tokens below market price.

Formal Control Problem Equations 5 - 6 define the state space, controls, dynamics, cost function, and constraints for control of the token economy. Thus, we define the formal control problem exactly as in Eq. 1. Our problem has nonlinear dynamics (since we divide by the price variable p_t^{Tok}) and a quadratic cost function. Thus, it is a non-convex optimization problem, which can be solved using the nonlinear optimal control methods introduced next.

4   Control Design Methodology

We now illustrate methodologies that solve our formal control problem Eq. 1 by outlining techniques from nonlinear optimal control theory.

Iterative Linear Quadratic Regulator (iLQR): A canonical model in optimal control is the linear quadratic regulator (LQR), where we have linear dynamics and a quadratic cost function that weights state tracking error and control effort, such as energy expenditure [5]. The linear dynamics are given by $x_{t+1} = A_t x_t + B_t u_t$ and the cost function is given by $J(x_0, u_{0:H-1}) = x_H^T Q_H x_H + \sum_{t=0}^{H-1} x_t^T Q_t x_t + u_t^T R_t u_t$, where all Q_t, R_t are positive (semi)-definite matrices to ensure non-negative costs. The optimal solution to LQR is an affine state feedback controller of the form $u_t^* = K_t x_t + k_t$, where K_t and k_t are obtained via dynamic programming and the discrete Algebraic Ricatti Equation.

In practice, we desire that our system follows a non-zero reference trajectory given by $(\bar{x}_t, \bar{u}_t)_{t=0}^H$, such as regulating the system to track a time-varying token supply. Moreover, the dynamics $f_t(x_t, u_t)$ could be non-linear and the cost function $c_t(x_t, u_t)$ could be non-convex nor quadratic. The key insight behind *iterative* LQR (iLQR) [33] is that we *linearize* a system's nonlinear dynamics around the reference trajectory $(\bar{x}_t, \bar{u}_t)$ if the dynamics are differentiable. Thus, we compute the Jacobian matrix of the nonlinear dynamics anchored at the current reference $\bar{x}_t, \bar{u}_t$ to obtain the dynamics matrix $A_t = \frac{\partial f_t}{\partial x}(\bar{x}_t, \bar{u}_t)$ and control matrix $B_t = \frac{\partial f_t}{\partial u}(\bar{x}_t, \bar{u}_t)$. Likewise, for a twice continuously differentiable cost function, we use the Taylor Series to form its local quadratic approximation.

Given a nominal reference trajectory, linearized dynamics, and a quadratic approximation of the cost, we can simply invoke LQR to improve our nominal reference trajectory. The process repeats until the control cost converges, which is analagous to Newton's method. We can incorporate strict state or control constraints by adding them as penalties to iLQR's cost function using *Augmented Lagrangian* iLQR (AL-iLQR) methods. Crucially, AL-iLQR is

our solution method of choice for tokenomics, since we have smooth nonlinear dynamics, a quadratic cost function, a well defined reference trajectory for the token price/circulating supply, and strict constraints for non-negative treasuries.

Sequential Convex Programming (SCP): SCP extends the core ideas behind iLQR to control problems with strict state or control constraints [8,23]. First, we linearize the dynamics around a reference trajectory, just as in iLQR. Then, we form a convex approximation of the cost function, often using a local quadratic approximation. Thus, we recover a constrained convex optimization problem, which we solve to obtain a new nominal trajectory. We then re-linearize around the updated nominal trajectory until the control cost saturates.
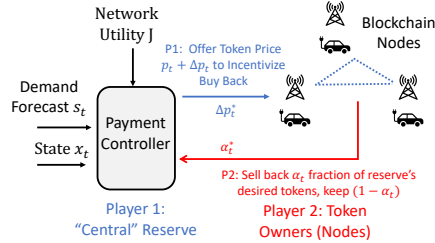
Model Predictive Control (MPC): In practice, we receive a stochastic forecast s_t of income from consumer demand. Then, given the current state x_0, and forecast $\hat{s}_{0:H-1}$, we can propagate the known dynamics $f(x_t, u_t, s_t)$ and solve the control problem in Eq. 1 to obtain an optimal sequence of controls $u^*_{0:H-1}$. In MPC, we simply implement the first control u^*_0, observe the next state x_1, and re-plan an optimal trajectory to mitigate forecast uncertainty [9]. Crucially, all our experiments run MPC where the underlying control strategy in Eq. 1 can use AL-iLQR or SCP without loss of generality.

Proportional Integral Derivative (PID) Control: Our proposed solution uses AL-iLQR and SCP since we can model the token dynamics and explicitly desire to optimize a cost function. We compare these methods to a benchmark proportional integral derivative (PID) controller. While PID controllers achieve stability [5], they do not explicitly optimize a cost function like iLQR/SCP, and often require extensive tuning and can overshoot a reference trajectory. PID is a fitting benchmark due to its simplicity and the fact that recent algorithmic stablecoins, such as the RAI index [2], use PID.

A PID controller operates on an error signal, which penalizes the deviation of the current state and desired/reference state. In our setting, the error signal $e_t = \bar{p}_t^{\text{Tok, ref.}} - p_t^{\text{Tok}}$ is the difference between the reference token price $\bar{p}_t^{\text{Tok, ref.}}$ and actual, measured token price p_t^{Tok}. Then, a PID controller generates a control decision u_t that is proportional to the instantaneous error, integrates past errors, and forecasts how the error will change as: $u_t = k_p e_t + k_i \sum_{j=0}^{t-1} e_j + k_d e_{t+1}$. Here, k_p, k_i, k_d are gain parameters that are often manually tuned for stability.

5 Strategic Pricing: A Game-Theoretic Analysis

In our control-theoretic formulation, we assume token owners will gladly sell their tokens to the reserve when it offers to buy back tokens with an incentive price of Δp_t. However, as shown in Fig. 2, strategic token owners might only sell a fraction of their tokens for immediate revenue and retain the rest for their future expected value. As such, the reserve must offer a sufficiently high incentive Δp_t to goad token owners to sell their valuable tokens so that the circulating token supply is regulated to avoid inflation. Our key insight is that strategic pricing can be formulated as a two-player Stackelberg game (see [28]).

Fig. 2: Strategic Pricing as a Game: The reserve (player 1) offers to buy-back tokens at the market price plus a bonus incentive of Δp_t to reduce the circulating supply. However, all token-owning nodes (collectively, player 2) might want to only sell a fraction α_t of tokens since they could be valuable in the future. We solve for an optimal game strategy for each player (Δp_t^* and α^*) using bi-level optimization.

Market Dynamics The market dynamics arise from the selfish behavior of rational consumers. At each timestep, we have a two-step sequential game of complete information between the reserve's controller (player 1) and all token-owning nodes (player 2). The controller optimizes program (1) and the consumers seek to maximize the value of their token holdings over the time horizon. By complete information, we mean that facts about the opponent respectively are common knowledge. For example, token owners are aware of the controller's strategy, which is encoded in smart contracts distributed across the blockchain. Likewise, each player can also perfectly observe the token price and supply.

The controller moves first by posting a price ($p_t^{\text{Tok}} + \Delta p_t$). Then, the consumers respond by selling an α_t-fraction of their collective holdings, where α_t is chosen to maximize the value of the consumers' holdings (utility U_t) at timestep t. Specifically, the utility U_t is a sum of the price they can get *today* and the price they can get *tomorrow*, which is a function of the game strategies:

$$U_t(\alpha_t, \Delta p_t) = \underbrace{\alpha_t \cdot S_t(p_t^{\text{Tok}} + \Delta p_t)}_{\substack{\text{Current earnings from selling} \\ \alpha_t\text{-fraction of supply.}}} + \underbrace{(1 - \alpha_t) \cdot \gamma S_t \mathbb{E}(p_{t+1}^{\text{Tok}})}_{\substack{\text{Future expected earnings from} \\ \text{holding } \alpha_t\text{-fraction of supply.}}} .$$

In the above, γ is a risk factor attenuating the expected future earnings from not selling. Further, the randomness in the expectation is due to forecasting noise as our controller is not randomized. Thus, for any controller-chosen incentive Δp_t, the nodes' optimal strategy is to choose α_t such that:

$$\max_{\alpha_t \in [0,1]} U_t(\alpha_t, \Delta p_t) \qquad \text{NODE UTILITY OPTIMIZATION.} \tag{7}$$

Moreover, due to program (7), the controller can compute the consumer strategy for any incentive price Δp_t. This means that we can transition the control problem (1) into the setting with incentives by equating the amount of tokens the controller buys back to the amount of tokens the nodes agree to sell:

$$\frac{u_t^{\text{B}}}{p_t^{\text{Tok}} + \Delta p_t} = \alpha_t \cdot S_t, \tag{8}$$

where α_t is computed as a function of incentive price through program (7). This means that in practice the controller can account for selfish behavior through roll out simulations to identify good incentive prices. However, we now formalize how the controller can play optimally by identifying the incentive price Δp_t^* inducing a subgame perfect equilibrium ($\alpha_t^*, \Delta p_t^*$).

A Stackelberg Game for Strategic Pricing Since the controller first posts a price Δp_t and the nodes respond with the fraction α_t of the holdings they wish to sell, we naturally have a leader-follower (Stackelberg) game. As mentioned above, we use (8) to constrain the tokens bought back by α_t. Then, recalling that the node's strategy is given by (7), the controller's optimization problem is:

<div align="center">

GAME FORMULATION

$$\min_{u_{0:H-1} s_{0:H-1}} \mathbb{E}\big[J(x_0, u_{0:H-1}; s_{0:H-1})\big],$$

$$\text{s.t,}\ \ x_{t+1} = f(x_t, u_t, s_t)$$

$$x_t \in \mathcal{X}, u_t \in \mathcal{U}$$

$$u_t^{\mathrm{B}} = \alpha_t S_t(p_t^{\mathrm{Tok}} + \Delta p_t)$$

$$\boxed{\alpha_t = \arg\max_{\alpha_t \in [0,1]} U_t(\alpha_t, \Delta p_t)}$$

</div>

<div align="center">

BI-LEVEL OPTIMIZATION

$$\min_{\substack{u_{0:H-1}, \lambda_{1:2} \\ \alpha_{0:H-1}} s_{0:H-1}} \mathbb{E}\big[J(x_0, u_{0:H-1}; s_{0:H-1})\big],$$

$$\text{s.t,}\ \ x_{t+1} = f(x_t, u_t, s_t)$$

$$x_t \in \mathcal{X}, u_t \in \mathcal{U}$$

$$u_t^{\mathrm{B}} = \alpha_t S_t(p_t^{\mathrm{Tok}} + \Delta p_t)$$

$$\boxed{\color{blue}{\nabla_\alpha \mathcal{L}(\alpha_t, \Delta p_t, \lambda) = 0}}$$

$$\boxed{\color{blue}{\lambda_i \leq 0, \alpha_t \in [0,1]}}$$

$$\boxed{\color{blue}{\lambda_1 \alpha_t = 0, \lambda_2(1 - \alpha_t) = 0}}$$

</div>

Solution via Bi-Level Optimization Notice that the Stackelberg game can be cast as a *bi-level* optimization problem [14,6], which is a nested optimization problem where an *outer* optimization problem involves a decision variable that is, in turn, the solution to a second *inner* problem. In our setting, the outer problem is the controller's non-convex control problem (1), which outputs Δp_t^* and requires α_t^* as input. Here, α_t^* is the solution to the inner utility maximization problem of the nodes. Crucially, we observe that the inner utility maximization problem is convex for any given value of Δp_t, and that Slater's constraint qualification condition holds. Thus, we know that the Karush-Kuhn-Tucker (KKT) conditions are sufficient for optimality for the inner problem. Then, we simply reduce the bi-level (nested) optimization problem as a single optimization problem for the controller where the optimal choice of α^* is encoded by adding its KKT conditions as extra constraints, with the Lagrangian denoted by $\mathcal{L}$. For clarity, we show the KKT constraints in blue above.

Remark 2. Since we have a Stackelberg Game, the horizon is finite and a subgame perfect equilibrium can be found via backward induction. First, the best response function of the nodes is calculated. Then, the controller picks an action maximizing its utility, anticipating the follower's best response. For more details, see [28]. Then it is clear that our method finds a subgame perfect equilibrium since the KKT conditions of the inner problem give a certificate of optimal play on the nodes part. Encoding them as extra constraints on the part of the controller simply gives an explicit route for backward induction in this game.

6   Experiments

The goal of our evaluation is to show that (i) using control theory enables us to achieve a more stable, increasing token price and (ii) we can reduce control cost

using an optimization-based controller instead of a reactive PID controller. We now describe our benchmark algorithms and evaluation metrics.

Evaluation Metrics and Benchmark Algorithms: We compare reserve controllers on the following metrics:

– **Stable Token Price**: We track whether the token price is increasing, its volatility, and the mean squared error (MSE) from a reference price trajectory.
– **Control Cost**: This is a weighted sum of the tracking error (MSE between the token price and reference price) as well as control effort.

We report these metrics for various realistic scenarios where (i) the supply of nodes out-paces the consumer demand, (ii) the supply and demand roughly match, and (iii) the supply lags the consumer demand. Our experiments compare the following schemes:

– **Model Predictive Control (MPC)**: We implement the predictive, optimal control scheme proposed in Section 4.
– **Proportional Integral Derivative (PID)**: PID controllers are a fitting benchmark since they are widely used in industrial systems such as cruise control, robotic manipulators, and in some algorithmic stablecoins.
– **No Control**: We consider the worst case where the economy has no adaptive control and simply uses the income clearing strategy from Remark 1.

Experiment Design First, we evaluate our control system on synthetic node supply and consumer demand/income growth data. Then, we repeat our simulations with real-world demand timeseries from the Helium DeWi network. In all experiments, we have a noisy forecast $\hat{s}_t$ of the demand and income growth for a horizon of H future steps. Fig. 3 plots a distribution of control cost over many diverse timeseries patterns and initial conditions x_0 (initial token supplies etc.) to emulate each controller in various scenarios. Second, we visualize the resulting token supply S_t and token price p_t^{Tok} compared to the desired reference trajectory in Fig. 4. All our experiments are coded in Python, run within a few minutes on a standard laptop, and use the Gurobi Optimization package [17].

Experiments with Helium and Synthetic Data Our synthetic experiments aim to generate realistic supply, demand, and income growth patterns with additive, zero-mean Gaussian noise. For example, we use a sigmoidal growth pattern where demand and supply rapidly rise during the middle of network adoption, but slowly taper off. We also simulate a logarithmic growth function where the demand steadily grows over time. Each simulation trial has a different reference price trajectory to allow a stable price growth, such as a logarithmic reference. In each trial, the initial token price p_t^{Tok} is randomly chosen away from the reference, and show the controller can dynamically regulate the system.

We then repeat the same experimental procedure, but with real timeseries of node growth and consumer demand from the Helium DeWi network, which
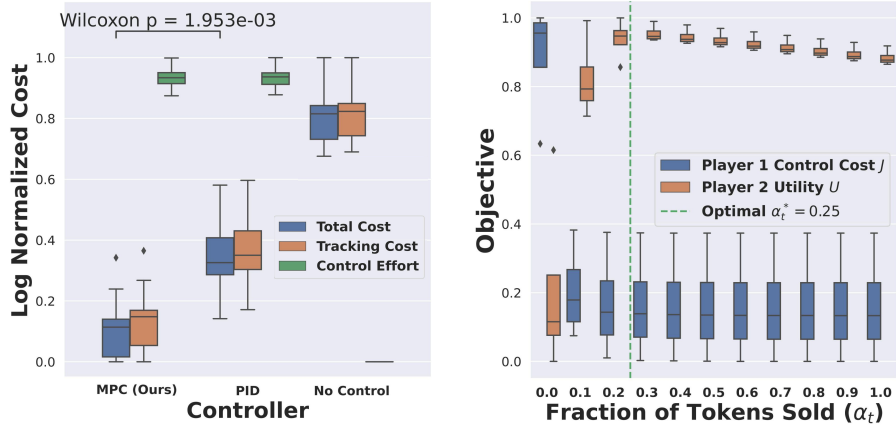
Fig. 3: Control Cost with Synthetic Data: (Left) Our MPC scheme achieves a lower total cost than the PID and the No Control benchmarks by exerting control effort to actively track the price trajectory. (Right) Our bi-level optimization approach finds a game strategy to sell only α_t^* tokens (green line) to maximize node utility and minimize control cost compared to simply holding ($\alpha_t = 0$) or selling all tokens ($\alpha_t = 1$).

uses a BME protocol. First, since the Helium growth patterns are smooth, we use a classical Auto-Regressive Integrated Moving Average (ARIMA) forecasting model to predict network growth $H = 20$ days in advance. Then, we implement our controller on a distribution of reference price trajectories. Fig. 4 shows that MPC significantly reduces the control cost compared to PID benchmarks.

Does MPC reduce control cost compared to heuristic controllers? We now evaluate our ultimate performance metric, the control cost, across a wide variety of growth patterns and initial conditions. Specifically, we used 3 growth patterns with Gaussian noise (sigmoidal, logarithmic, exponential) and many scenarios where demand outstrips supply and vice versa. Fig. 3 shows the overall control cost, tracking error, and control effort for all 3 benchmarks. Clearly, MPC achieves a lower control cost than the PID heuristic (Wilcoxon p-value of .001953 is statistically significant at the 0.05 level). As shown in Fig. 3, the key reason for this difference is that PID is largely *reactive* – it proportionally responds to the current error and integrates the cumulative error, but does not forecast the future system state accurately to optimize performance. In stark contrast, MPC explicitly solves an optimization problem to minimize the control cost.

Does our Stackelberg game solution maximize each player's objective function? Fig. 3 (right) illustrates the token owning nodes' utility U_t (player 2) and controller cost function J (player 1). Clearly, our bi-level optimization approach is able to find an optimal strategy α_t^* of tokens to sell back, since the node utility is maximized and controller cost is minimized. The solution from bi-level optimization clearly outperforms naive strategies of simply holding all tokens with $\alpha_t = 0$ and selling back all tokens with $\alpha_t = 1$. As expected, the control cost
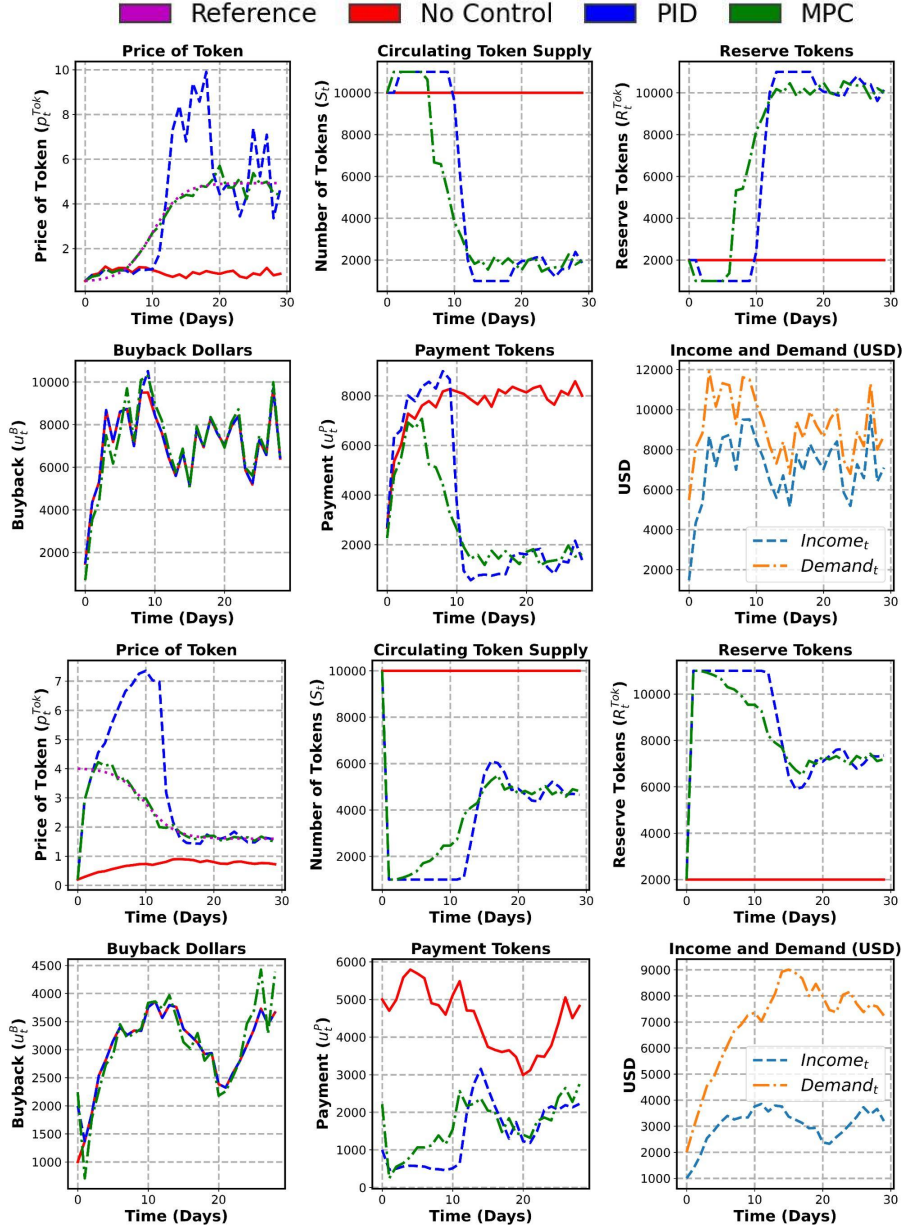
Fig. 4: Results with Helium (top) and Synthetic (bottom) Data: Our key result is on the top left for the token price. Our MPC scheme (green) is able to track the reference (purple) extremely well, while reactive PID (blue) captures the general trend but is highly oscillatory. Crucially, the price plummets without control since we pay too many tokens, which is mitigated by our adaptive MPC payment scheme. The bottom two rows show another price reference to illustrate the generality of our approach.

J (blue boxplots) decreases as α_t increases, since the controller is able to buy back more of the tokens that it desires. Likewise, the node utility (orange) is low when they hold all their tokens ($\alpha_t = 0$) since the price drops because the controller can not remove tokens from circulation through buy backs.

Does MPC outperform benchmarks to yield a stable token price growth? Fig. 4 shows two example trajectories of our system, where the top two rows are with real Helium data and the bottom two are with synthetic data. Our key result is on the top left for the token price. For the Helium data, our MPC scheme (green) is able to track the reference (purple) extremely well, while heuristic PID captures the general trend but is highly oscillatory since it is *reactive*. Crucially, the price plummets without control since we pay too many tokens, which causes inflation. However, our MPC scheme adaptively curtails token payments to reduce the circulating supply and avoid inflation (middle). Importantly, the vanilla income-clearing strategy from Remark 1 (red) immediately pays out the exact same number of tokens it buys back from the market. Thus, the net change in the circulating token supply (and hence reserve) is zero, as indicated by the horizontal red lines for the token plots.

The last two rows confirm the generality of our approach – we can just as easily follow a smoothly decreasing price trajectory. For example, we might want to price the token more affordably over time. Crucially, the initial token price (row 3, top left) is very low and far from the reference for all schemes, but our MPC method (green) quickly rises to track the reference unlike PID (blue), which overshoots. Importantly, the price without control is very low but slightly increasing since the income/demand gradually increase over time. Finally, MPC slowly increases the adaptive payments after timestep 11 for the price to decrease.

Limitations Our trace-driven simulations are limited by offline, historical Helium DeWi data. However, the growth of nodes and consumers might significantly deviate from historical patterns if we actually implemented our proposed controller in the network. In future work, we plan to answer such "what-if" questions using recent advances in counterfactual analysis [25,29,3].

7   Discussion and Future Work

Our central thesis is that Blockchain tokenomics should be programmable and dynamically adapt to node growth and consumer demand. Our key contribution is to model a token economy as a controlled dynamical system, which allows us to leverage rigorous systems theory to design token economies that meet high-level performance metrics (network cost functions). We believe our work is timely as several Blockchain projects are working with burn and mint strategies and our framework enables us to (a) explicitly prove these systems reach a stable equilibrium and (b) flexibly steer this equilibrium to incentivize stable network growth. We are working with several Blockchain projects to instantiate these ideas in practice, which we hope to report on in future work.

References

1. Burn and mint equilibrium. https://messari.io/report/burn-and-mint-equilibrium (2019), [Online; accessed 19-October-2022]
2. Summoning the money god: Applying control theory to cryptocurrency monetary policy. https://medium.com/reflexer-labs/summoning-the-money-god-2a3f3564a5f2 (2021), [Online; accessed 10-June-2021]
3. Agarwal, A., Shah, D., Shen, D.: Synthetic interventions. arXiv preprint arXiv:2006.07691 (2020)
4. Agrawal, A., Barratt, S., Boyd, S., Stellato, B.: Learning convex optimization control policies. In: Learning for Dynamics and Control. pp. 361–373. PMLR (2020)
5. Åström, K.J., Murray, R.M.: Feedback systems: an introduction for scientists and engineers. Princeton university press (2021)
6. Bard, J.F.: Practical bilevel optimization: algorithms and applications, vol. 30. Springer Science & Business Media (2013)
7. Bhardwaj, M., Sundaralingam, B., Mousavian, A., Ratliff, N.D., Fox, D., Ramos, F., Boots, B.: Storm: An integrated framework for fast joint-space model-predictive control for reactive manipulation. In: Conference on Robot Learning. pp. 750–759. PMLR (2022)
8. Bonalli, R., Cauligi, A., Bylard, A., Pavone, M.: Gusto: Guaranteed sequential trajectory optimization via sequential convex programming. In: 2019 International conference on robotics and automation (ICRA). pp. 6741–6747. IEEE (2019)
9. Borrelli, F., Bemporad, A., Morari, M.: Predictive control for linear and hybrid systems. Cambridge University Press (2017)
10. Caginal, C.: A dynamical systems approach to cryptocurrency stability. arXiv preprint arXiv:1805.03143 (2018)
11. Camacho, E.F., Alba, C.B.: Model predictive control. Springer Science & Business Media (2013)
12. Carlsten, M., Kalodner, H., Weinberg, S.M., Narayanan, A.: On the instability of bitcoin without the block reward. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. pp. 154–167 (2016)
13. Chung, H., Shi, E.: Foundations of transaction fee mechanism design. arXiv preprint arXiv:2111.03151 (2021)
14. Colson, B., Marcotte, P., Savard, G.: An overview of bilevel optimization. Annals of operations research **153**(1), 235–256 (2007)
15. Eyal, I.: The miner's dilemma. In: 2015 IEEE Symposium on Security and Privacy. pp. 89–103. IEEE (2015)
16. Eyal, I., Sirer, E.G.: Majority is not enough: Bitcoin mining is vulnerable. Communications of the ACM **61**(7), 95–102 (2018)
17. Gurobi Optimization, L.: Gurobi optimizer reference manual (2018)
18. Haleem, A., Allen, A., Thompson, A., Nijdam, M., Garg, R.: Helium - a decentralized wireless network. https://whitepaper.io/document/649/helium-whitepaper (2018)
19. Kereiakes, E., Do Kwon, M.D.M., Platias, N.: Terra money: Stability and adoption. White Paper, Apr (2019)
20. Labs, P.: Filecoin: A decentralized storage network. https://whitepaper.io/document/599/filecoin-whitepaper (2020)
21. Labs, S.: Storj: A decentralized cloud storage network framework. https://www.storj.io/storjv3.pdf (2018)

22. Levandowski, A.: Intro to pollen mobile. https://docs.pollenmobile.io/pollen-mobile-docs/white-paper/intro-to-pollen-mobile (2021)
23. Malyuta, D., Reynolds, T.P., Szmuk, M., Lew, T., Bonalli, R., Pavone, M., Acikmese, B.: Convex optimization for trajectory generation. arXiv preprint arXiv:2106.09125 (2021)
24. Messié, V., Fromentoux, G., Marjou, X., Omnes, N.L.: Baladin for blockchain-based 5g networks. In: 2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN). pp. 201–205. IEEE (2019)
25. Morgan, S.L., Winship, C.: Counterfactuals and causal inference. Cambridge University Press (2015)
26. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system. Decentralized Business Review p. 21260 (2008)
27. Network, C.: Computecoin network: The infrastructure of web 3.0 and the metaverse. https://whitepaper.io/document/649/helium-whitepaper (2020)
28. Osborne, M.J., et al.: An introduction to game theory, vol. 3. Oxford university press New York (2004)
29. Pearl, J.: Causal inference. Causality: objectives and assessment pp. 39–58 (2010)
30. Petkanics, D., et al.: Protocol and economic incentives for a decentralized live video streaming network (2018)
31. Roughgarden, T.: Transaction fee mechanism design. ACM SIGecom Exchanges **19**(1), 52–55 (2021)
32. Snow, P., Deery, B., Lu, J., Johnston, D., Kirby, P.: Factom: Business processes secured by immutable audit trails on the blockchain. Whitepaper, Factom, November **111** (2014)
33. Van Den Berg, J.: Iterated lqr smoothing for locally-optimal feedback control of systems with non-linear dynamics and non-quadratic cost. In: 2014 American control conference. pp. 1912–1918. IEEE (2014)
34. Williams, G., Drews, P., Goldfain, B., Rehg, J.M., Theodorou, E.A.: Information-theoretic model predictive control: Theory and applications to autonomous driving. IEEE Transactions on Robotics **34**(6), 1603–1622 (2018)
35. Zargham, M., Zhang, Z., Preciado, V.: A state-space modeling framework for engineering blockchain-enabled economic systems. arXiv preprint arXiv:1807.00955 (2018)
36. Zhang, Z., Zargham, M., Preciado, V.M.: On modeling blockchain-enabled economic networks as stochastic dynamical systems. Applied Network Science **5**(1), 1–24 (2020)