

CroCoDAI: A Stablecoin for Cross-Chain Commerce

DANIËL REIJSBERGEN, Nanyang Technological University, Singapore

BRETISLAV HAJEK, National University of Singapore, Singapore

TIEN TUAN ANH DINH, Deakin University, Australia

JUSSI KEPPO, National University of Singapore, Singapore

HENRY F. KORTH, Lehigh University, USA

ANWITAMAN DATTA, Nanyang Technological University, Singapore

Decentralized Finance (DeFi), in which digital assets are exchanged without trusted intermediaries, has grown rapidly in value in recent years. *Stablecoins*, which are pegged to a non-volatile asset such as the US dollar, are a prominent feature of DeFi as they mitigate the risk associated with price fluctuations. However, existing stablecoin systems are tied to individual blockchain platforms, and trusted parties or complex protocols are needed to exchange stablecoin tokens between blockchains. Our goal is to design a practical stablecoin system for cross-chain commerce, and to do so we must overcome two main challenges. The first is to support a large and growing number of blockchains efficiently. The second is for the stablecoin to be resilient to blockchain platform failures and to price fluctuations that affect its collateral. We present CroCoDAI to address these challenges. We demonstrate CroCoDAI's efficiency by comparing the performance of a prototype implementation to related baselines, and its resilience through an empirical analysis of historical token price data.

ACM Reference Format:

Daniël Reijsbergen, Bretislav Hajek, Tien Tuan Anh Dinh, Jussi Keppo, Henry F. Korth, and Anwitaman Datta. 2024. CroCoDAI: A Stablecoin for Cross-Chain Commerce. 1, 1 (September 2024), 31 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

Decentralized Finance (DeFi) promises to revolutionize finance by removing the need for trusted middlemen to process transactions. In DeFi, transactions commonly involve a change of ownership of digital *tokens* that exist on a distributed ledger such as a *blockchain*¹ – e.g., one user sending bitcoins to another [66]. Many blockchain platforms (e.g., Ethereum [10]) also support *smart contracts*, which enable automated versions of financial concepts such as auctions, loans, swaps, decentralized exchanges, and ownership shares. The total value of the tokens held in DeFi smart contracts peaked at over \$150 billion (USD) in late 2021 [23, 76], with at least \$100 billion held in decentralized exchanges alone [78], however it had fallen to around \$50 billion as of December 2023 [23].

¹In the remainder of this paper, we will use the terms ‘blockchain’ or ‘chain’ to refer to any type of distributed ledger, including ledgers that are technically not chains (e.g., Avalanche’s directed acyclic graphs [71]).

Authors’ addresses: Daniël Reijsbergen, Nanyang Technological University, Singapore; Bretislav Hajek, National University of Singapore, Singapore; Tien Tuan Anh Dinh, Deakin University, Australia; Jussi Keppo, National University of Singapore, Singapore; Henry F. Korth, Lehigh University, Pennsylvania, USA; Anwitaman Datta, Nanyang Technological University, Singapore.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2024 Association for Computing Machinery.

XXXX-XXXX/2024/9-ART \$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

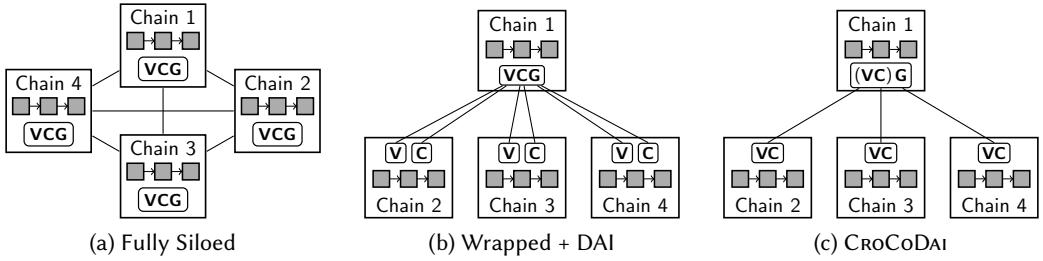


Fig. 1. Overview of different cross-chain stablecoin systems. **V**, **C**, and **G** respectively indicate the presence of vault, stablecoin, and governance contracts.

The DeFi ecosystem is currently fragmented over a multitude of different blockchain platforms which each offer their own choices in terms of efficiency, security, functionality, and structure of DeFi contracts. For example, Cardano [41, 46], Solana [80], Avalanche [72], and Algorand [34] each had a valuation of more than a billion USD in early 2023. These blockchains are not designed to *interoperate*: i.e., there are no native mechanisms for transferring tokens between chains. However, the emergence of blockchain *bridges* [15, 27] and dedicated interoperability platforms such as Cosmos [9] and Polkadot [50] demonstrates an increasing demand for cross-chain interoperability. A major challenge in cross-chain trades is *atomicity*: the transactions in a trade must be committed on all blockchains, or not at all. Existing protocols for cross-chain atomicity, including *relays* and *hashed time locks* [3, 39, 73], essentially lock the assets involved in a trade and only release (or return) them once the transactions have been committed (or aborted) on all blockchains. While ensuring atomicity, this approach increases the financial risks for the users in the following ways: (1) the tokens held in escrow may lose value due to price fluctuations, (2) one party may abort the transaction, leading to an opportunity cost for the other parties, and (3) the DeFi platforms that facilitate the trade may be faulty or malicious. These risks are interrelated: if there is a significant change in the value of the tokens in escrow (1), then this may cause a party to abandon a trade (2), creating an opportunity cost to the other users. Price volatility can be mitigated in principle if the tokens held in escrow are *stablecoins* [12, 47, 65] whose value is pegged to a less volatile real-world asset (e.g., the US dollar). However, a flawed stablecoin design can itself put users' tokens at risk, as witnessed by the collapse of the Terra/Luna stablecoin in May 2022 [7], in which stablecoins worth \$60 billion lost nearly all of their value.

Our goal is to design a practical stablecoin as a central point of exchange for cross-chain commerce. We focus on stablecoin designs that are backed with crypto collateral – e.g., Dai [58] on the Ethereum blockchain – to minimize the dependence on trusted parties outside the DeFi ecosystem (e.g., banks and asset management companies). To realize our goal, we must address two main challenges. The first challenge is to support a large and growing number of blockchains efficiently, including new and relatively low-use platforms. These low-use blockchains typically have lower transaction fees, but also less liquidity and existing DeFi support. The second challenge is to ensure that the stablecoin is resilient to *black swan events* (e.g., 51% attacks or critical smart contract bugs) that affect individual blockchain platforms, and to *price fluctuations* that affect multiple blockchains simultaneously. If the total value of the collateral drops below the intended value of the issued stablecoins – which we will refer to as *system failure* – then this may trigger the DeFi equivalent of a bank run, forcing stablecoin holders to sell their tokens at a loss. Our key insight is that a wide range of collateral can address both challenges. In particular, it allows users from smaller chains to create robust stablecoins without interacting with a more expensive

blockchain or relying on a trusted third party. Furthermore, it reduces the exposure of the stablecoin system as a whole to sudden crashes and price fluctuations.

In this work, we present CROCoDAI, a stablecoin for cross-chain commerce that addresses the above challenges. Compared to its closest baseline approaches, CROCoDAI has greater efficiency, i.e., lower transaction costs, and better or similar resilience to black swan events and price fluctuations.

A first baseline is for each blockchain to support its own crypto-backed stablecoin and required components: *vaults* to store collateral (**V**), *coin* contracts to process stablecoin transfers (**C**), and *governance* contracts for system-level decisions (**G**) – this is depicted in Figure 1a. In this baseline, stablecoins can be transferred between chains through bridges that are unique to each chain pair. Such bridges either lock tokens as collateral on one chain and create *wrapped* [51] tokens on the other, or *burn* tokens [42] on one chain and mint them on the other.² However, the lack of liquidity on smaller blockchains makes it difficult to find collateral locally, and each individual stablecoin is highly vulnerable to black swan events and price fluctuations on its platform.

A second baseline is to combine an existing crypto-backed stablecoin on a large (i.e., high-use) blockchain, e.g., Ethereum’s Dai, with two bridges for each smaller chain: one to deposit wrapped versions of the smaller chains’ native tokens on the large chain for collateral, and another to create wrapped versions of the stablecoin on the smaller chains. In this setting, each smaller chain has a vault contract (**V**) for collateral and an independent coin contract (**C**) for the wrapped stablecoins – see Figure 1b. However, this approach requires unnecessary transactions on the large blockchain to create stablecoins or move them between chains – such transactions are typically more expensive than on the smaller chains. By contrast, CROCoDAI uses integrated vault and coin contracts (**V** and **C**) on each smaller chain, and leverages the governance layer on a dedicated relay chain for cross-chain transactions and system-level decisions, as depicted in Figure 1c.

In this paper, we make the following contributions:

- (1) We present CROCoDAI, a stablecoin for cross-chain commerce (Section 3), and explore two main design variants: one uses a quorum of trusted nodes for the relay chain, and the other uses an existing blockchain (Section 3.3). While CROCoDAI re-uses parts of the existing Dai stablecoin system, interactions with vaults are strictly local to the smaller chains, and it supports cross-chain price oracles to improve price discovery on smaller blockchains.
- (2) We present a theoretical analysis of CROCoDAI to show that its resilience to black swan events (Section 4) and price fluctuations (Section 5) is greater than the first baseline (fully siloed), and similar to the second baseline (Dai + wrapped tokens), due to its greater ability to diversify its collateral. In particular, our analysis shows that CROCoDAI has $\approx 30\%$ lower risk of system failure than the first baseline, based on historical token price data from a period that includes the collapse of the FTX exchange.
- (3) We present a prototype implementation of CROCoDAI and perform microbenchmark and end-to-end experiments (Section 6) to compare CROCoDAI to the second baseline (Dai + wrapped tokens) in terms of transaction costs and processing times. We find that the gas cost of stablecoin creation is only half as much in CROCoDAI compared to the second baseline – furthermore, CROCoDAI can complete the entire process in a single block, whereas the baseline requires block finalization (which takes 32+ blocks on Ethereum) on two chains for security. The total gas cost of cross-chain transfers is $\approx 40\%$ higher in CROCoDAI than in the baseline, but the baseline consumes $\approx 120\%$ more gas on the (expensive) relay chain. For

²The Circle-created USDC bridge between Ethereum and Avalanche [5] is an example of the latter.

example, if gas on the relay chain is $8\times$ more expensive than on the smaller chains,³ then CROCoDAI is roughly 40% cheaper than the second baseline.

In summary, CROCoDAI is cheaper than a baseline with wrapped tokens, is less vulnerable to price fluctuations and black-swan events than a siloed stablecoin, and is not reliant on any trusted third parties unlike existing stablecoins such as Tether and Paypal USD. By facilitating stablecoin use on smaller chains and stablecoin transfers between chains, CROCoDAI is a practical medium of exchange in cross-chain commerce [21]. To aid reproducibility, we have made the data and the source code for our experiments publicly available via <https://github.com/ntublockchain/crocodai>.

The remainder of the paper is structured as follows. Section 2 provides background information on DeFi and related work. Section 3 presents the design of CROCoDAI. We analyze CROCoDAI's resilience against black swan events in Section 4 and price fluctuations in Section 5. Section 6 details the prototype implementation and corresponding experimental results. Section 7 discusses future work and concludes the paper.

2 BACKGROUND

2.1 General Background

Tokens and DeFi. In DeFi, the principal function of blockchain transactions is to transfer *tokens*, which represent ownership of assets. There are two common types of tokens: *fungible* tokens – also commonly called *cryptocurrencies* – and *non-fungible tokens* (NFTs) [20]. Fungible tokens of the same type are identical, whereas each NFT represents a unique asset. Smart contracts can be used to define cryptocurrencies that are different from the blockchain's native cryptocurrency. In Ethereum, this is particularly facilitated by the *ERC20 token standard* [74] which defines a common set of smart contract functions required by any new cryptocurrency. Smart contracts can also support NFTs, e.g., via OpenSea or Mintable. A prominent use case for user-defined tokens is to create a version of one blockchain's token on another blockchain – e.g., Wrapped Bitcoin (WBTC) on the Ethereum blockchain. In general, *wrapped* tokens [51] are tokens maintained by *custodians* – e.g., for WBTC this includes the Kyber Network and BitGo – who are trusted to create wrapped tokens if and only if an equivalent amount of tokens under their control have been locked on the supporting blockchain. XCLAIM [82] is an academic proposal that is similar to WBTC but which requires custodians to deposit additional collateral, thus adding a financial penalty as a disincentive for abuse (e.g., stealing the locked tokens).

Stablecoins. A stablecoin is a cryptocurrency whose value is pegged to an underlying fiat currency, e.g., the US dollar. There are multiple types of stablecoins [47, 65], of which the most common ones are:

- (1) *Fiat-backed stablecoins*, which are backed by fiat currency held largely in cash or cash equivalents. The essential guarantee that underpins the price stability of a fiat-backed stablecoin is that in case of system shutdown, each stablecoin can be used to redeem a corresponding amount of the pegged fiat currency.
- (2) *Crypto-collateral-backed stablecoins* (or *crypto-backed stablecoins* for brevity), which are backed by other cryptocurrencies that are stored in one or more smart contracts. In this case, the guarantee is that in case of system shutdown, each stablecoin can be redeemed for a basket of tokens whose value equals one unit of the pegged currency.
- (3) *Algorithmic stablecoins*, which do not use reserves or collateral and which attempt to maintain a peg by adjusting the supply of a paired token [65].

³The $8\times$ difference is a conservative estimate based on a sample of gas cost differences between Ethereum and layer-2 chains such as Arbitrum and Optimism [26, 52, 67] in early 2024. Recently, far greater differences ($200\times$ and above) have been observed, e.g., 0.06 GWei/gas for Optimism vs. 10 GWei/gas for Ethereum on 18 July 2024.

A multitude of fiat-backed stablecoins exist, the most prominent (as of September 2023) being Tether (USDT), USD Coin (USDC), Binance USD (BUSD), and TrueUSD (TUSD). The most prominent crypto-backed stablecoin is DAI on the Ethereum blockchain, which is maintained by the MakerDAO foundation and which we discuss in more detail in Section 2.2. Other examples of crypto-backed stablecoins include Liquity USD (LUSD) and Reflexer’s RAI, which are backed entirely by ETH. Other recent stablecoin proposals include FRAX [43], which is an algorithmic stablecoin with a crypto-backed component, and Celo [18], which is backed by a mixture of fiat and crypto assets.

During our observation period (see Section 5), most fiat- and crypto-backed stablecoins did an overall sound job in maintaining their peg with the US dollar: the observed price ranges of DAI, USDT, USDC, BUSD, and TUSD were [0.9921, 1.0062], [0.9834, 1.0005], [0.9991, 1.0073], [0.9988, 1.0126], and [0.9937, 1.0056], respectively. The analysis in [57] suggests that fiat-backed stablecoins have a trade-off between price stability and risk of system failure in the sense that better arbitrage improves the former at the cost of increasing the latter. Recently, established financial services have launched their own fiat-backed stablecoins, e.g., PayPal’s PYUSD, which is available on the Ethereum and Solana blockchains [68]. Finally, there have been proposals by central banks to issue their own digital tokens, which would have the same functionality as stablecoins – e.g., the digital euro, which recently completed its 2-year investigation phase and is now moving to the preparation phase [28].

Fiat-backed stablecoins introduce a trust assumption, namely on the agents who handle and store the fiat currency. In Tether, this trust assumption has been widely questioned – accusations include Tether using fiat collateral to buy high-risk assets [29] and creating unbacked stablecoins to prop up the price of bitcoins [35]. Furthermore, algorithmic coins have nothing to underpin them in case of a *death spiral* [7, 65], as witnessed by the loss of over 60 billion USD when the Terra/Luna stablecoin system failed. The main threat to crypto-backed stablecoins is that the system may fail if the value of the collateral decreases too rapidly – however, we show in this paper that this risk can be made small with sufficient overcollateralization. In the following, we focus particularly on Dai, which is the most prominent crypto-backed stablecoin.

2.2 The Dai Stablecoin System (DSS)

In this section, we describe the existing single-chain DSS. Although there are documents and code repositories that describe the DSS in its various stages of evolution, e.g., [4, 16, 58–60, 65], we focus on the GitHub code [60] and the description of the multi-collateral DSS [59].

Dai Tokens. In the DSS, Dai tokens are minted by users who deposit cryptocurrency tokens as collateral. In the first iteration of the Dai stablecoin (the single-collateral Dai or “Sai”), only ETH tokens could be deposited for this purpose. The procedure for a user, Alice, who wants to mint new Sai tokens is as follows. First, she must send a transaction to a Dai vault to create a *Collateralized Debt Position* (CDP) [58]. Next, she must send ETH tokens as collateral to the contract to fund the CDP. Alice can use a funded CDP to create Dai stablecoins and withdraw them to her wallet, incurring *debt*. At any later time, she can send Dai tokens back to the CDP to repay her debt. The above can be generalized to a multi-collateral (but single-chain) DSS by associating each CDP with a token type: in the DSS, this could theoretically include any ERC20 token, including wrapped tokens such as WBTC. Debt increases exponentially over time, and the rate at which it increases is determined by the *stability fee*.

Liquidations. At all times, Dai CDPs must be *overcollateralized*, i.e., the ratio of the total value of the Alice’s collateral to her debt must exceed some threshold called the *liquidation ratio* [4, 58]. By default, the liquidation ratio in Dai is set to 1.5 – i.e., Alice can withdraw Dai stablecoins worth at most $\frac{2}{3}$ of the total value of the collateral in their CDP [65]. Similarly, Alice can return Dai and withdraw collateral from her CDP as long as the above condition holds. If a CDP has no associated

debt, then it can be *closed* and all remaining collateral can be withdrawn. If, at any time, a CDP's ratio of its collateral value to its debt is insufficient, then Alice's CDP is *underwater* and can be liquidated. CDPs can go underwater either because the value of the collateral has dropped due to price fluctuations, or because the stability fee has caused the debt to become too high. To avoid liquidation, Alice must either deposit additional collateral, or raise Dai to clear an at-risk position – if this fails, a *collateral auction* can be started by other users. During a collateral auction, other users can bid to buy (portions of) Alice's collateral using Dai tokens. Once the CDP's debt has been cleared, Alice receives the remaining collateral (if any) minus a penalty.

Price Feeds. To determine the value of deposited collateral, MakerDAO uses information from price feeds supplied by *oracles* – e.g., Chainlink or Uniswap contracts [55]. The number of price feeds used by the DSS has increased from 14 between 2017 and November 2019 [37] to 20 in late 2023 [64]. Smart contracts that maintain CDPs – i.e., the vaults – periodically update token prices by setting each price equal to the median of a set of recent prices for the associated token.

Governance. All major decisions about the functioning of the protocol are taken by users who hold MKR, which is MakerDAO's governance token. Such decisions include 1) verifying and adding smart contracts that implement any of the functionalities above, 2) changing the Dai savings rate, and 3) occasionally changing other system-level parameters. The following are the main system-level parameters (see page 15 of [59]): (1) the *debt ceiling* for each token type, i.e., the maximum amount of total debt that can be created using a given collateral type, (2) the stability fee, (3) the liquidation ratio, (4) the liquidation penalty, (5) the duration of collateral auctions, (6) the duration of auction bids, and (7) the minimum auction step size (to prevent bids with a minor increment).

Out-of-Scope Concepts. The following DSS concepts are beyond the scope of this paper: (1) the Dai Savings Rate mechanism, which allows users to deposit Dai to collect excess stablecoins created from the stability fee, (2) debt auctions, which occur when a collateral auction fails to raise enough Dai to cover the CDP's debt, (3) surplus auctions, which occur when the amount of Dai left in vaults due to auctions and stability fees exceeds some threshold, and (4) emergency shutdowns, which can be triggered after a catastrophic event (e.g., system failure) and during which the total remaining collateral is distributed to Dai holders.

2.3 Related Work

The DSS can be seen as an extension of *red-black coins* [70], which feature two token types of which one has reduced price volatility (e.g., Dai tokens vs. CDP debt). In [70], single-asset price simulations are used to estimate the price of the volatile tokens – the main difference with the analysis in Section 5 is that we estimate failure risks, that we use multiple correlated assets, and that we use a heavy-tailed distribution to better represent token price fluctuations. Price simulations of DSS collateral can also be found in [4], and [48] presents a stochastic model for a single-asset stablecoin. In [36], the relationship between stablecoin and Bitcoin price fluctuations is investigated. In [57], fiat-backed stablecoins are found to have a trade-off between price stability and risk of system failure.

For secure cross-chain transfers, *cross-chain deals* [39] provide a modeling framework to describe sequences of cross-chain token transfers. In cross-chain deals, the cost of a participant abandoning the protocol before completion (*sore-loser attacks* [79]) can be made explicit – this form of misbehavior can be disincentivized through a *premium* [25] paid by the parties that abandon the protocol before termination. Cosmos [9] and Polkadot [50] are blockchain platforms that were specifically developed with blockchain interoperability in mind, but which do not trivially interoperate with existing blockchains outside their ecosystems. However, a version of CROCoDAI could be implemented as a native stablecoin on these platforms with collateral on multiple subchains. Alternatively, collateral on multiple chains can be integrated into CROCoDAI through trusted relays

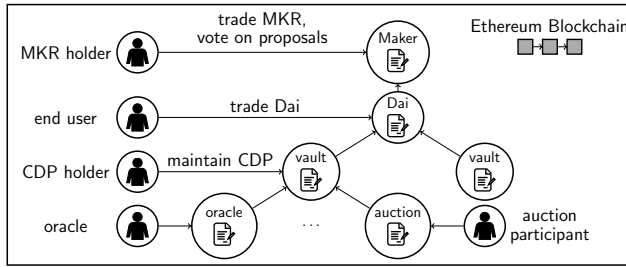


Fig. 2. Overview of the existing single-chain DSS.

or zero-knowledge proofs as discussed in Section 3. Finally, XCLAIM [82] claims to provide secure cross-chain transfers, but has limitations when transferring to/from smaller blockchain platforms that lack sufficient liquidity because the custodian must submit additional collateral.

3 CROCODAI: A CROSS-CHAIN STABLECOIN

In this section, we present CROCODAI, which extends the functionality of the single-chain DSS (depicted in Figure 2) to a multi-chain setting (depicted in Figure 3). Our main design goal is a stablecoin that is backed by collateral on multiple blockchains, and whose tokens can be moved securely and efficiently between chains. In the following, we will refer to blockchains that store collateral and stablecoin tokens as *coin chains* (as per [39]). The communication between the coin chains is managed by a *relay chain*, which may be an existing (coin) chain. The main difference between the single-chain DSS and CROCODAI is the inclusion of *relay nodes* – depicted with double circles in Figure 3. In CROCODAI, all vaults are local to the coin chains, and cross-chain price oracles are supported to complement the oracles on smaller coin chains. This allows CROCODAI to have lower costs for local operations such as creation and transfer of stablecoins on smaller blockchain platforms (we will discuss this in more detail in Section 6.)

In the following, we first discuss in more detail which DSS operations remain local and which require interaction with the relay chain. We then discuss the main components of the coin and relay chains. Next, we discuss two design variants for the relay chain: the first relies on a quorum of trusted nodes who perform most operations off-chain, and the second uses an existing blockchain. We conclude by listing the security requirements of a cross-chain stablecoin, which will be the subject of the Sections 4 and 5.

3.1 Overview

Local Operations. CROCODAI supports the following local operations. **CDP Management:** Each coin chain has its own vaults that allow users to create CDPs using tokens on that blockchain. As such, stablecoins can be created locally on minor chains, i.e., without the need for transactions on a more expensive chain, as long as the debt limit on the minor chain is not violated. Since CDPs are local, liquidation auctions and other mechanisms such as savings rates are also processed locally. **Local Stablecoin Transfers:** Each coin chain keeps track of the balance of stablecoins held by its accounts, so the stablecoins can be exchanged locally between accounts. **Local Price Information:** Finally, local oracles may provide the vaults with price information.

Cross-Chain Operations. The following operations require relay chain involvement. **Cross-chain Transactions:** In a cross-chain transaction, stablecoins are burned on one coin chain and an equivalent number is minted on another coin chain, as we discuss in more detail below. **Ward Management:** Governance layer approval is needed to add/remove addresses that are authorized

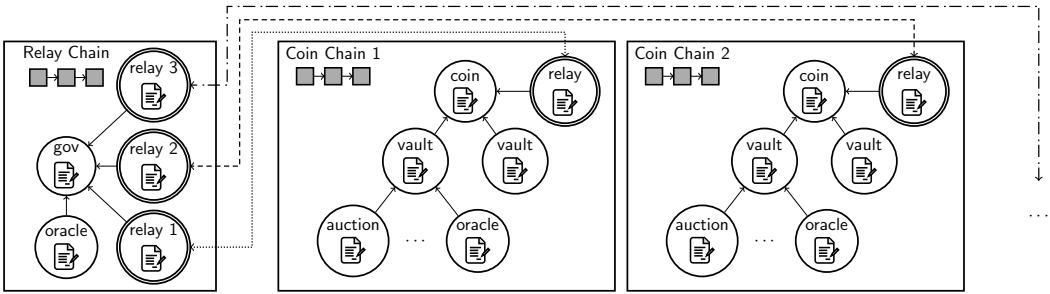


Fig. 3. Overviews of CROCoDAI, a multi-chain stablecoin system based on the DSS. The relays (double circle) are not shared with the single-chain DSS.

to call security-critical functions such as mint or burn on another contract – we call such addresses *wards* as per DSS terminology. Periodically, the governance layer may grant or revoke ward status to/from smart contracts for a variety of reasons – e.g., (1) to add support for an emerging blockchain platform, (2) because a critical bug has been found in an existing smart contract, (3) because a chain has undergone a hard fork that interferes with the functioning of the contract, or (4) to implement a more efficient version of a component. **System Parameter Updates:** Governance layer approval is also needed to update the system parameters discussed in Section 2.2, e.g., a chain’s debt ceiling or a token’s overcollateralization rate on a chain. **Global Price Information:** The final function is to provide an additional source of token price information to vaults through relay chain oracles – this particularly benefits smaller blockchains that lack a mature DeFi ecosystem.

The exact mechanism for cross-chain transfers is as follows: a user who holds stablecoins on a coin chain first submits a cross-chain transfer request to a relay node and transfers the tokens for escrow. The request is added to a hashmap where it awaits commitment. The nodes on the relay chain monitor the coin chains, and once they detect a cross-chain transfer request, they sign a commit message. When the coin chain receives the commit message, it commits the transaction and burn the tokens held in escrow. A commit message is also sent to the relay node on the target blockchain, which mints an equivalent number of stablecoins. If the transaction is aborted for any reason, then the stablecoins in escrow are returned to the sender.

3.2 Core Components

Coin Chain Components. Each coin chain has the following components. **Coin:** implements a contract that is similar to Dai in the DSS, and processes transfers of stablecoins between users on the coin chain. There is only one coin contract per chain. **Vault:** implements contracts that are similar to Vat, GemJoin and DaiJoin in the DSS, which allow users to create CDPs and withdraw/deposit collateral and stablecoins. There can be multiple vaults per chain. **Auction:** implements auction contracts that liquidate the collateral of underwater CDPs, with any appropriate auction logic – e.g., Cat or Dog in the DSS. There can be multiple auction contracts per chain. **Bottom-Level Oracle:** implements a price feed using a contract similar to Spot in the DSS, combined with a Medianizer. **Bottom-Level Relay:** validates messages from the relay chain. It processes (1) the addition of wards to CROCoDAI contracts, (2) incoming cross-chain transfers, (3) commitments of outgoing cross-chain transfers, and (4) updates of system parameters.

Relay Chain Components. The relay chain has the following components. **Governance:** tracks the ownership of governance tokens, and allows users to transfer them. As in the DSS, users who hold governance tokens can vote on system-level actions such as adding wards and updating

system-level parameters. Our security assumption regarding governance token holders is that a majority is honest and online at any point during the voting window for system-level decisions. **Top-Level Oracle:** The relay chain may implement system-level price feeds to complement the price feeds on the coin chains. **Top-Level Relay:** To process cross-chain transfers, the relay chain must be able to process information from the coin chains, particularly the transfer requests – we discuss this in more detail below.

3.3 Relay Chain Designs

In this section, we discuss two possible design variants of the relay chain. The first relies on a quorum of trusted nodes with known public keys – each trusted node runs an off-chain (full or light) blockchain client for each of the underlying chains to check the validity of cross-chain transfers, and the nodes maintain a distributed ledger for governance decisions. The second uses smart contracts on an existing chain to implement the consensus logic of the underlying chains (potentially using zero-knowledge cryptography) to verify cross-chain operations on-chain.

Off-Chain Blockchain Clients. In this approach, the stablecoin system is maintained by n relay nodes with known public keys. The relay nodes use their (light) clients to detect and validate cross-chain transactions, and generate signatures using their private/public key pair to commit cross-chain transfers and process system-level actions approved by governance token holders. In this setting, the only functionality of the bottom-chain relay contracts is to verify digital signatures. We discuss three different subvariants of this approach in Section 6. As is common in Byzantine fault tolerant systems [13, 53], we assume that at most f nodes are faulty or malicious such that $n \geq 3f + 1$. We note that it is vital that the relay nodes are controlled by independent entities: e.g., in the attack against the Ronin bridge, 5 out of 9 validators were successfully attacked [45], of which 4 belonged to the same entity [56].

On-Chain Consensus Logic. In this approach, the consensus logic of the coin chains is implemented in the top-level relay contracts, and the consensus logic of the relay chain is implemented in the bottom-level relay contracts. Essentially, the functionality of a light client – i.e., validating block headers and proving the state of a contract using the header – is implemented inside the smart contracts. This is the approach taken in, e.g., XCLAIM [82]. For example, in Bitcoin’s Nakamoto Consensus [66] this means that the smart contract computes header hashes and compares their integer value to the proof-of-work difficulty,⁴ whereas in Proof-of-Stake Ethereum the smart contract verifies that the total stake of the nodes who have approved a block is sufficient. In this setting, the main role of the relay nodes is to generate and submit validity proofs. Techniques to increase the efficiency of light clients have been proposed in the literature for both proof-of-work (e.g., Bitcoin [22]) and proof-of-stake blockchains (e.g., Algorand [63] and Mina [6]) – in some cases using zero-knowledge proofs that may incorporate the blockchain’s consensus logic [77]. Zero-knowledge proofs that are verifiable on multiple chains, e.g., ZKM’s entangled rollups [85], can also be used to bridge the chains that implement collateral to a layer-2 network that implements the vaults. Finally, the data on the different coin chains can be integrated using TrustBoost [75].

Although the second design variant avoids a trust assumption on a quorum of relay nodes, current iterations of such designs have a large computational overhead, typically requiring several kilobytes of data sent to relay contracts and more than 10 seconds to generate signatures [63, 77].⁵ As such, we focus on the first variant in Section 6 and leave the second as future work.

⁴This may be computationally efficient if blockchains share built-in hash functions, otherwise this can be expensive.

⁵Improving the performance of such zero-knowledge schemes, e.g., using folding schemes [49], is an active research area.

3.4 System Requirements

The central requirement that we focus on in this paper is the following:

- (*) **Sufficient Overcollateralization.** At any time $t \geq 0$, the ratio of the value of the collateral tokens to the total number of stablecoins in the system exceeds a *safety threshold* $\theta > 1$.

The motivation behind this requirement is as follows: if the value of the collateral becomes too low, then the system is no longer able to reimburse all stablecoin holders with an equivalently valued basket of tokens in case of system shutdown. As such, we can expect users to start selling their stablecoin tokens at a loss to redeem as much value as possible, causing the stablecoin to lose its peg. We assume that this sell-off may occur before the critical point is reached at which the value of issued stablecoins exceeds that of the collateral, because nervous investors may preempt the loss of the peg. As such, we choose a value of θ that is greater than 1: a value of strictly 1 corresponds to a stablecoin system that fails as soon as the aforementioned critical point is reached, whereas $\theta > 1$ also covers scenarios in which system failure occurs earlier, in anticipation of the depegging.

Due to random price fluctuations, we cannot guarantee with absolute certainty that full backing will hold in all circumstances. However, we use our analysis to show that for reasonable choices of θ and the liquidation ratio, the probability that full backing is violated after a medium-length period (i.e., one day) is very small for historical data. In Section 4, we also investigate under which conditions full backing continues to hold in the event of a black swan event such as a sudden token price collapse or a 51% attack on a blockchain.

We also require the following properties under the threat and token price model.

- (1) **Safety.** Users do not lose tokens during stablecoin transfers and CDP operations.
- (2) **Liveness.** Stablecoin transfers and CDP operations eventually happen.
- (3) **Safety of Collateral** (informal). The probability that a minority of corrupted oracles cause a CDP that is not underwater to be inadvertently liquidated is small.

More formally, for the last property we consider a minority of corrupted price oracles whose goal is to convince the protocol to accept a price p^* instead of a true base price p – safety of collateral then holds if their probability of success vanishes exponentially as $|p - p^*|$ increases (see also Theorem 4.3 in Section 4).

As we will see in Section 4, the former two properties largely follow from results in the existing scientific literature. The latter is a result that we prove for the multi-chain setting in Section 4, but which is also applicable in the single-chain setting.

3.5 Illustrative Application

We consider an example of cross-chain commerce: a setting in which an asset – e.g., a basket of cryptocurrencies or an NFT – is auctioned on one chain, and for which there are interested bidders who own tokens on multiple other chains. For the duration of the auction, price stability is important because changes in the value of bids could lead to unnecessary resubmissions or withdrawals of bids, causing a diminished user experience and/or opportunity costs [79]. The auction example allows us both to illustrate the end-to-end processes of creating and transferring stablecoins in CROCoDAI, as well as to compare CROCoDAI to existing approaches in terms of performance and user experience. For this example, we will refer to the chain on which the asset is sold as the ‘auction chain’ and to the chains on which bidders hold tokens as ‘coin chains’.

To achieve an auction with price stability, we consider two main approaches: 1. bidders use a centralized exchange (CEX) to swap their tokens into stablecoins on the auction chain, 2. bidders using CROCoDAI’s built-in mechanism for cross-chain transfers. The first represents the most straightforward way to achieve a cross-chain auction in the current ecosystem by leveraging

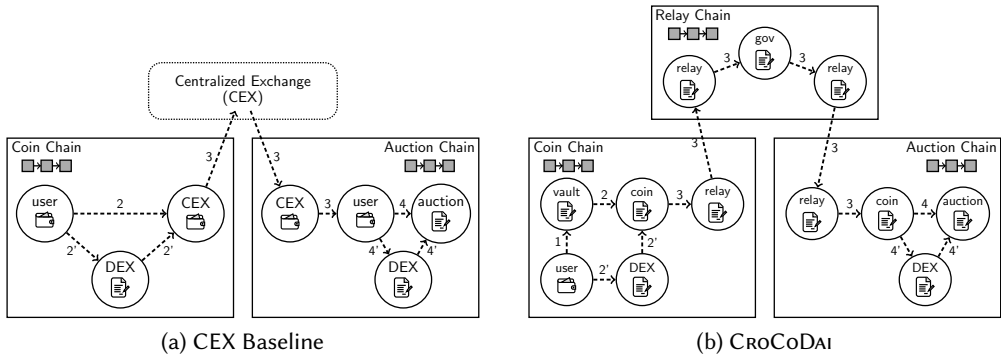


Fig. 4. Steps of a cross-chain exchange in the CEX baseline and CROCoDAI. The edge labeling corresponds to the step numbers in Section 3.5.

established exchanges such as Coinbase or Binance, whereas the second is our proposal. We also discuss other approaches at the end of this section.

Comparison. In the first approach, we assume the existence of a centralized exchange that can be used to buy and sell stablecoins on the coin chain and auction chain. In particular, the bidder would then complete the following steps:

- (1) Create an account with the CEX if she doesn't have one already.
- (2) Transfer her tokens to the CEX to add it to her wallet on the CEX.
- (3) Sell the tokens in her account and buy stablecoins, or swap the tokens for stablecoins directly if supported by the CEX, and withdraw the stablecoins to her account on the auction chain.
- (4) Place a bid by transferring the stablecoins to the auction contract.

In CROCoDAI, the bidder would complete the following steps:

- (1) Deposit her tokens into a CROCoDAI vault.
- (2) Create new CROCoDAI stablecoins and withdraw them to her account.
- (3) Issue request to the relay chain for a cross-chain transfer of her CROCoDAI tokens on the coin chain to an account owned by her on the auction chain.
- (4) Place a bid by transferring the stablecoins to the auction contract.

The fundamental difference between the two approaches is the reliance on a trusted third party in the first approach. By contrast, the second approach relies on a decentralized relay chain. Furthermore, the CEX baseline is more likely to suffer from a lack of liquidity: whereas steps 1 and 2 in the first baseline always rely on the CEX as a counterparty to swap tokens, in CROCoDAI the bidder can always create new stablecoins and transfer them to the auction chain via the relay chain. In terms of fees and transaction delays, there are no reasons to expect major differences between the two approaches, since the cost of sending and processing cross-chain messages is typically small compared to the cost of monitoring the supported coin chains through a blockchain node, and the latter is a requirement for all cross-chain systems.

Additional Intermediaries. In both approaches discussed above, either the CEX or CROCoDAI's relay chain acts as an intermediary to facilitate the cross-chain token transfer. Additional steps that involve intermediaries may be needed if 1) the CEX does not support sales of the bidder's tokens, 2) CROCoDAI does not accept the bidder's tokens as collateral (e.g., because they belong to an unsupported type, or because the token type's debt limit would be violated), 3) the auction contract does not accept any token sold by the CEX, or 4) the auction contract does not accept

CroCoDAI stablecoins. In those cases, a DEX may exist that allows for on-chain swaps between unsupported and supported tokens. The additional steps involving DEX swaps are depicted in Figure 4 through step 2', which replaces step 2, and step 4', which replaces step 4.

Other Approaches. A cross-chain auction can also be implemented using the framework of cross-chain deals [39]. However, this requires that the auctioneer creates smart contracts on each of the supported coin chains, instead of just the auction chain. Finally, the bidder could create a wrapped version of her tokens on the coin chain, but this would typically require another trusted intermediary and is therefore similar to the CEX baseline.

3.6 Deployment

Real-world use of CroCoDAI would require the deployment of the relay chain, the smart contracts (e.g., coin and vault contracts) on the supported smaller chains, and the development of tools that allow users to interact with the system. The costs of developing and deploying the relay chain are similar to those of existing bridges such as Celer⁶ and Relay⁷. The deployment of smart contracts has roughly the same cost as in a fully siloed system, and are small over long run compared to fees (e.g., at a gas price of 20GWei and an ETH price of \$3000, deploying a 1m gas contract would cost \$60). The throughput and scalability of the relay chain are highly dependent on the design choices and implementation, e.g., Celer's cBridge claims a latency of at most 20 minutes during peak demand periods [14], whereas the zkBridge prototype has a latency of under 2 minutes [77] but at a high computational cost, while the Hephaestus protocol [2] claims to be able to process 600 cross-chain transactions in under 6 seconds.

In practice, we expect the user interface design to be very similar to existing designs for the DSS: web apps such as summer.fi⁸ allow users to interact with vaults and to process local and cross-chain transfers. The incentives for developers are the same as for Dai, i.e., a combination of fees, publicity (an app can draw attention to other services), and in some cases volunteer work. The active DAI ecosystem, including summer.fi, daistats.com [19], DeFiExplore,⁹ demonstrates that such incentives are sufficient in practice.

Finally, the system parameters in CroCoDAI can be set to similar values as in the DSS. The overcollateralization rates $-\gamma$ will likely be in the order of 30 – 50% for the supported token types, as they are in existing Dai vaults [19]. Similarly, the debt limits for smaller blockchains are expected to be similar to those for ERC20 tokens such as Aave and Gnosis – as in the DSS, these are proposed and voted on by governance token holders.

4 SECURITY ANALYSIS

In this section, we analyze whether our system satisfies the following technical properties from Section 3.4: *safety* and *liveness*, *full backing* under the threat model, and *safety of collateral*. We first discuss our threat model and then present our results. In particular, we prove that if tokens whose total debt limit¹⁰ is ζ' suffer a price crash (Section 4.3) or platform-level attack (Section 4.4), then full backing continues to hold if $\theta < \gamma(1 - \zeta')$, where γ denotes the liquidation ratio (see Section 2.2). In practice, this allows stablecoin holders to judge whether they find the risk of system failure low enough given their estimations of θ and ζ' . We conclude by comparing the security of

⁶<https://cbridge.celer.network/bridge/ethereum-polygon/eth/>

⁷<https://relay.link/bridge/ethereum>

⁸<https://summer.fi/>

⁹<https://maker.defiexplore.com>

¹⁰We note that price crashes in different tokens and platforms may occur simultaneously, e.g., the collapse of FTX also had consequences for other platforms such as Solana.

CROCoDAI to the two baselines from the introduction. The technical proofs of the main theorems can be found in Appendix A.

The following notation is used throughout this section and Section 5. Token prices are represented in terms of their exchange rate with a non-volatile base asset, e.g., the US dollar (the *numéraire*). Let M denote the set of all tokens supported by CROCoDAI. We denote the base price of token $m \in M$ at time $t > 0$ by p_{mt} , and the initial base price of token m by p_{m0} . (Here, $t = 0$ represents an arbitrary start point of time.) Given a batch of $k \geq 0$ tokens of type m , the *value* of this batch at time t is given by kp_{mt} . We assume that the price of token m at time t returned by an honest price oracle $o \in [O]$ equals $p_{mot} \triangleq p_{mt} + \Delta_{mot}$, where Δ_{mot} is a random variable with a normal/Gaussian distribution [69] with mean $\mu_{mo} \in \mathbb{R}$ and variance $\sigma_{mo}^2 > 0$. The assumption of normality is in accordance with Section 4.1, and is motivated by the common assumption that measurement errors have a normal distribution, e.g., in linear regression [24].

4.1 Security Model

In the following, we first discuss the threat assumptions that are specific to CROCoDAI, and then discuss the threats and countermeasures shared with generic stablecoin and cross-chain systems.

In CROCoDAI's security model, a majority of the governance token holders are honest and online, and the relay chain satisfies safety and liveness. Each underlying blockchain is either compromised or not. If it is not compromised, then we assume that it guarantees safety and liveness for all of its transactions. However, both safety and liveness can be violated in a compromised blockchain – in particular, transactions in which collateral is sent to/from a CDP can be rolled back. We assume that the underlying communication networks satisfy partial synchrony, i.e., bounded delivery times, to ensure that nodes do not wait indefinitely for information to be transmitted between chains. It has shown that liveness cannot be guaranteed in the presence of even a single faulty node in a fully asynchronous model [30]. We do note that CROCoDAI itself provides synchrony to the overall cross-chain process by requiring that a transfer is completed on the sending chain before processing it on the receiving chain, as depicted in Figure 4b. For the price oracles, a majority is online and honest at all times, in which case they report a measurement error that is random and normally distributed. Furthermore, the output of each oracle is independent of the output of other oracles. Dishonest oracles can report arbitrary prices. We emphasize that these assumptions are common in the scientific literature, e.g., Byzantine fault tolerant protocols also require honest and online (super)majorities and partial synchrony [53].

As a cross-chain stablecoin system, CROCoDAI also inherits threats that affect generic blockchains, cross-chain infrastructures, and stablecoins. Recent surveys that investigate the security of such systems are available for individual blockchains [40, 84], cross-chain systems [3, 39, 81], and stablecoins [62]. Threats that affect individual blockchains include the following categories:

- *Network-layer threats*, which may prevent users from connecting with honest blockchain nodes, potentially violating safety (if the remaining nodes are malicious) or liveness. Examples include eclipse attacks on blockchain peers, DNS/routing attacks, and Denial-of-Service (DoS) attacks. Countermeasures include generic network security proposals such as DNSSEC and BGPsec, or blockchain-specific defences such as SABRE [1].

Consensus protocols that announce a single block proposer beforehand – e.g., many Proof-of-Stake protocols [8] – may be vulnerable to targeted DoS attacks against the block proposer, violating liveness. CROCoDAI itself does not introduce such a vulnerability because cross-chain transfers can be initiated by any of the relay nodes.

- *Consensus-layer threats*, which may cause established transactions majority to be reverted or the network to accept incorrectly formed blocks. Examples include majority (51%) attacks or

selfish mining. Countermeasures include protocol-level defences, e.g., finality/checkpointing protocols, and robust incentive mechanism design.

Finality protocols are also helpful to CROCoDAI implementations, because CROCoDAI needs a logic for each supported chain to determine when a cross-chain transfer has been processed (i.e., highly unlikely to be reverted). Finality protocols provide such a logic by default.

- *Contract-layer threats*, which may allow attackers or malicious owners to subvert the intended functionality of a smart contract. Examples include smart contract bugs or deliberate backdoors. Countermeasures include static code analysis or formal verification, potentially run by third parties in the case of malicious owners or insider threats. CROCoDAI's similarity to the DSS suggests that we can leverage a larger market for smart contract validation services.
- *Application-layer threats*, e.g., bugs or backdoors in applications that support the stablecoin system, such as tools that allow users to interact with CDPs. As with contract-layer threats, countermeasures include static code analysis and formal verification.

The main threats to stablecoin systems as enumerated by researchers at NIST [62] are categorized as security, stability, and trust “issues”. The following security “issues” are mentioned: (1) Unauthorized or Arbitrary Minting of Stablecoins, (2) Collateral Theft, (3) Malicious Smart Contract Update and Hijack, (4) Data Oracles, (5) Exploiting the Underlying Blockchain, and (6) Writing Secure Software and Vulnerabilities. Of these threats, (1), (2), and (5) are covered by the analysis in Section 4.4, as all scenarios effectively result in threats to the system's overcollateralization. Finally, (4) is covered by the analysis in Section 4.5, whereas (3) and (6) are generic blockchain threats as discussed previously, which can be addressed using the countermeasures mentioned in [40, 84].

Security threats to generic cross-chain systems are an active area of research [3, 81], although some threats such as cross-chain replay attacks and data availability violations have been studied in the scientific literature. We discuss the fundamental properties of safety and liveness in CROCoDAI in more detail below.

4.2 Technical Safety and Liveness

For local operations, the consensus protocol of the coin chain on which the local transfer occurs ensures the safety and liveness of operations on this chain – see, e.g., [32] – under the assumption of partial synchrony. This includes local stablecoin transfers, CDP operations, auctions, and interest withdrawals. As such, tokens cannot be stolen during local transfers or CDP operations or locked forever in a CDP unless the coin chain is compromised. For cross-chain operations, safety is violated if tokens in a bottom-level relay are burnt without a corresponding amount being minted on the receiving chain, and liveness is violated if tokens are held in escrow forever. However, the relay chain satisfies safety and liveness in our threat model, i.e., the transfer is eventually committed and coins are burned and minted on the outgoing and incoming chains, respectively. An alternative way to reach this conclusion is to view cross-chain transfers in CROCoDAI as simple cross-chain deals [39] with the sender as the only party who votes, and the relay chain as a certified blockchain – depending on the design (as discussed in Section 3.3), either $2f + 1$ nodes or the smart contracts on the relay chain sign the commit vote.

To ensure safety and liveness, we emphasize that it is essential that the relay network is not compromised. A compromised relay network can burn coins without creating a corresponding amount on the receiving chain, violating user safety. Likewise, if the relay network is unable to process requests because more than f relay nodes are offline or malicious, then the system does not satisfy liveness as cross-chain transfer requests may never happen. We also note that if the relay network is compromised, then full backing can be violated by creating an incoming transfer request without a matching outgoing transfer request, thus creating tokens without collateral. Although

the security of the *relay* chain is therefore essential to CROCoDAI, we investigate the impact of compromised *coin* chains on full backing below.

4.3 Sudden Crash of Tokens

Occasionally, the price of a cryptocurrency token may suddenly collapse, e.g., because of the discovery of a serious bug in one of its smart contracts, or because of off-chain events such as a bankruptcy or a legal inquiry into the organization responsible for its maintenance. As an example: the bankruptcy of Celsius caused the value of the CEL token to collapse, whereas the inquiry into FTX did the same for the FTT token and Serum (SRM). In this case, the collateral auctions are likely to fail to generate enough revenue and the system's over-collateralization is under threat. In the following, we consider a worst-case scenario: the price of some tokens falling to zero within a single time slot, i.e., between a given time slot t' and time slot t'' – for simplicity, we assume $t' = 0$ and $t'' = 1$. We show that CROCoDAI remains fully backed at time t'' if it is sufficiently over-collateralized and if the total debt created using tokens on the affected chains is limited.

Formally, let \mathcal{J}_t denote the set of all CDPs at time t , c_{jt} the total number of collateral tokens held in CDP $j \in \mathcal{J}_t$ at time t , and D_{jt} the outstanding debt resulting from Dai withdrawals/deposits from/to CDP $j \in \mathcal{J}_t$ at time t . We denote the token type associated with CDP j by $y_j \in M$. Let $C_t^* = \sum_{j \in \mathcal{J}_t} p_{y_j t} c_{jt}$ denote the total value of the system's collateral at time t and $D_t^* = \sum_{j \in \mathcal{J}_t} D_{jt}$ the total debt. Let $M' \subset M$ denote the set of affected tokens and $\zeta' = \sum_{m \in M'} \zeta_m$ the total debt ceiling of the affected tokens. Let

$$C_t^- = \sum_{\substack{j \in \mathcal{J}_t \\ y_j \in M'}} p_{y_j t} c_{jt}, \quad C_t^+ = \sum_{\substack{j \in \mathcal{J}_t \\ y_j \notin M'}} p_{y_j t} c_{jt}$$

i.e., C_t^- represents the total value of the collateral tokens affected by the collapse, and C_t^+ the total value of the unaffected tokens. Similarly, let

$$D_t^- = \sum_{\substack{j \in \mathcal{J}_t \\ y_j \in M'}} D_{jt}, \quad D_t^+ = \sum_{\substack{j \in \mathcal{J}_t \\ y_j \notin M'}} D_{jt}$$

so that D_t^- represents the debt associated with affected CDPs and D_t^+ the debt of unaffected CDPs at time t . We consider an attack that occurs at time 1, so that $C_1^- = 0$, $C_1^+ = C_0^+$, $D_1^- = D_0^-$, and $D_1^+ = D_0^+$. We can then prove the following theorem (see Appendix A):

THEOREM 4.1. *Let $C_1^- = 0$, $C_1^+ = C_0^+$, $D_1^- = D_0^-$, and $D_1^+ = D_0^+$. If $p_{y_j 0} c_{j0} / y_{j0} > \gamma$ for all $j \in \mathcal{J}_0$, then $C_1^* / D_1^* > \gamma(1 - \zeta')$, i.e., full backing holds at time $t = 1$ if $\theta < \gamma(1 - \zeta')$.*

4.4 Individual Compromised Coin Chain

If a majority of consensus power on a coin chain is compromised then the consensus nodes could perform a 51% attack in the following way: send some amount of stablecoins to another chain, wait for the transaction to be committed by the relay network, then fork the compromised coin chain and undo the outgoing transfer request and the burning of the blockchain tokens. Similarly, they could undo the funding of a CDP. As a consequence, stablecoins are created on another chain without backing collateral. In the worst case, this attack would be performed simultaneously on all of the token types supported by the compromised chain. Again, let $M' \subset M$ denote the set of affected tokens. We again consider a worst-case attack that has been completed at $t = 1$. Let h' be the total amount of stablecoins created in the attack, so that $D_1^- = D_0^- + h'$. We can then prove the following theorem (see Appendix A):

THEOREM 4.2. *Let $C_1^- = C_0^-$, $C_1^+ = C_0^+$, $D_1^- = D_0^- + h'$, and $D_1^+ = D_0^+$. If $p_{y_j,0}c_{j0}/y_{j0} > \gamma$ for all $j \in \mathcal{J}_0$, then $C_1^*/D_1^* > \gamma(1 - \zeta')$, i.e., full backing holds at $t = 1$ if $\theta < \gamma(1 - \zeta')$.*

4.5 Compromised Price Feeds

Finally, we consider an attack in which a dishonest minority of price oracles tries to corrupt the price maintained by a vault. This could cause CDPs to be liquidated even if they are not underwater, thus violating safety of collateral. We note that the DSS already has implemented safety mechanisms for corrupted price oracles, e.g., its Oracle Security Module (OSM). The OSM¹¹ increases the available time to detect corrupted price feeds by adding an artificial delay before price information is accepted by the system. Such safety mechanisms are complementary to our analysis in the sense that they make individual price oracles harder to compromise, whereas our analysis shows that attacks against the system as a whole have negligible probability of success if only a minority of oracles is compromised. Furthermore, CROCoDAI's inclusion of global price oracles on minor chains provides another complementary safety mechanism. After all, if price oracles from established global sources are included, then it is harder for attackers to compromise a majority among the full set of oracles than if this set only includes local sources from the minor chain's less developed DeFi ecosystem.

Let p be the base price for token $m \in M$ at time $t \geq 0$ (we omit m and t from the notation in this section as they are not relevant to the analysis). Let O denote the set of all oracles and p_o the price returned by oracle $o \in O$, so that $p_o = p + \Delta_o$ where Δ_o is a normally distributed random variable as per Section 4.1. In the following, we assume some subset $O' \subset O$ is controlled by malicious parties who aim to convince a CROCoDAI vault to accept a price p' that is different from the base price p . Let p^* be the actual median price among all price feeds, including those that have been distorted. We can then prove the following theorem (see Appendix A):

THEOREM 4.3. *If $|O'| < M/2$, then the probability that the difference between the base price p and the median price p^* exceeds a constant $c > 0$ vanishes exponentially as c increases.*

4.6 Comparison to Baselines

In a fully siloed baseline, each coin chain has its own DSS-like stablecoin. If the chain is not compromised, then local and cross-chain transfers are secure (i.e., tokens cannot be stolen or locked permanently), whereas they are not if the chain is compromised. In this sense, the baseline is similar to CROCoDAI. The fully siloed system is unaffected by price crashes or platform failures on other blockchains – however, it is less resilient to price crashes and platform failures on its own chain than a system with more diversified collateral: e.g., in CROCoDAI, a stablecoin holder on a minor chain that experiences a price collapse would still be able to transfer stablecoins to another chain and exchange them for other tokens. Finally, because CROCoDAI supports additional price oracles, it is more difficult for an adversary to compromise a majority of oracles than in the baseline.

The baseline that consists of the DSS with a wider set of wrapped tokens has the same security guarantees as CROCoDAI, as it is similarly able to diversify its collateral. However, we will see in Section 6 that this baseline has lower efficiency.

5 FINANCIAL ANALYSIS

The main system requirement from Section 3.4, full backing, can be violated not just due to critical attacks and platform failures, but also due to natural price fluctuations in token prices. This vulnerability to price fluctuations is compounded by the strong correlation between the prices of different cryptocurrencies (see Figure 6 in the appendix). In this section, we use a financial risk

¹¹<https://docs.makerdao.com/smart-contract-modules/oracle-module/oracle-security-module-osm-detailed-documentation>

model to estimate the probability that the system violates full backing over a 1-day period. As is common in financial analysis, we will refer to a basket of collateral tokens as a *portfolio*, and to the relative increase in value of a token type over one time slot as a *return*. For our risk analysis, we compare a collateral portfolio that approximates the collateral composition of the existing DSS (without other stablecoins) to a portfolio made up of a wider range of blockchain tokens. In the following, we first present our model for token prices, and then present our risk analysis for the DSS-based portfolios.

5.1 Token Price Model

As mentioned in Section 4, we denote the base price of token $m \in M$ at time $t > 0$ by p_{mt} . To model the evolution of token prices over time, we use *Geometric Brownian Motion* (GBM). This is a common choice in the financial literature on asset prices [69] and related work on stablecoins, e.g., by Salehi et al. [70], particularly due to the GBM's role in the Black-Scholes formula for option prices. A stochastic process S_t that follows GBM is defined using a stochastic differential equation:

$$dS_t = \mu S_t dt + \sigma S_t dW_t, \quad (1)$$

where μ determines the drift of S_t (i.e., its expected change per unit of time), σ represents the volatility of S_t (i.e., the variance of the change), and W_t is the Wiener process that represents standard Brownian motion [69]. In this model, each token price process p_{mt} , $t \geq 0$, can be represented using a process defined as in Equation (1). Essentially, this means that the logarithm of the relative difference of token prices, $\log(p_{mt+t'}/p_{mt})$, is normally distributed for all $t' > 0$ with mean and variance dependent on t' . Unlike [70], we consider a setting with multiple assets; i.e., we have a stochastic process S_{mt} for each token m with its own mean μ_m and variance σ_m^2 . In this setting, we must also consider the *covariance* between the token prices [69], as cryptocurrency prices are often highly correlated – we will discuss this in more detail below.

The standard GBM as described above is based on the normal distribution, i.e., $\log(S_{t+t'}/S_t)$ has a normal distribution. However, it is known from the academic literature that the probability distribution of stock price fluctuations is usually more heavy-tailed and that Student's t -distribution is often a better fit [44, 54]. This is also noticeable in the data for price changes in our dataset. As such, we will focus on the t -distribution in our analysis. Although the t -distribution does not allow for an analytical solution for the asset price change after each individual time slot, we can use Monte Carlo simulation to analyze its behavior, as we discuss in more detail below.

5.2 Risk Analysis: Approach

In this section, we investigate the probability that CROCoDAI remains fully backed even in the presence of standard price fluctuations. In particular, we will estimate this probability for several choices of supported collateral tokens: 1) a single token type, 2) tokens that exist on the Ethereum blockchain, and 3) tokens that exist on a variety of independent blockchains. The second probability corresponds to the safety of the single-chain DSS, whereas the third option corresponds to the safety of CROCoDAI. The purpose of this section is to demonstrate that the third option has considerably lower risk than the second, using multi-chain collateral baskets that are similar to those observed for the DSS in practice (see Table 5 in Appendix B). We discuss our approach in Section 5.2 and present our empirical results in Section 5.3.

Problem Statement. Let $\mathcal{J}_{mt} = \{j \in \mathcal{J}_m : y_j = m\}$ be the set of CDPs at time t whose token type is m , and $\alpha_{mt} = \sum_{j \in \mathcal{J}_{mt}} c_{jt}$ be the total number of collateral tokens of type m at time t . In the following, we assume for simplicity that $\alpha_{mt} \equiv \alpha_m$, i.e., that the composition of the portfolio does not change during the observation period. This is a reasonable assumption as interactions with CDPs do not occur frequently. In our model, failure of the stablecoin system occurs if, for any t we

have

$$\sum_{m \in M} \alpha_m p_{m,t} \leq \theta D_t^*, \quad (2)$$

where θ is the required over-collateralization and D_t^* as defined in Section 4. We introduce a new variable γ' that denotes the initial over-collateralization calculated as $\gamma' = \frac{\sum_{m \in M} \alpha_m p_{m,t}}{D_0}$. Based on that, we can reformulate the failure condition as: $\frac{\sum_{m \in M} \alpha_m p_{m,t}}{\sum_{m \in M} \alpha_m p_{m,0}} \leq \frac{\gamma'}{\theta}$. As one would expect, failure does not depend on the absolute size of the portfolio, but on its relative composition. Our first goal is then to estimate the probability of failure for a choice of $\alpha_1, \dots, \alpha_{|M|}, \gamma'$, and θ .

In our simulation, we represent the portfolio as relative proportions of different assets. We define a vector \vec{v} that denotes the value of individual assets in the portfolio as $\mathbf{v} = (v_1, v_2, \dots, v_{|M|})^T$, where v_i is value of asset i . Similarly, we define a vector of random variables for log returns of individual assets $\mathbf{X} = (X_1, X_2, \dots, X_{|M|})^T$, where X_m is the random variable for log returns of asset $m \in M$. We then simulate the price of each asset as an individual GBM with Student's t -distribution for the return after each time slot, as discussed in Section 5.1. To accurately simulate price fluctuations, we must consider the correlation between the prices – to do so, we use Cholesky decomposition to compute a matrix \mathbf{L} that satisfies $\text{Cov}(\mathbf{X}) = \mathbf{L}\mathbf{L}^T$. A matrix of random samples \mathbf{R} can then be generated as $\mathbf{R} = \mathbf{L}\mathbf{T}$, where the matrix \mathbf{T} contains a t -distributed random sample for each asset type and simulated time slot.

5.3 Risk Analysis: Evaluation

In this section, we analyze the risk of system failure in both single-chain and multi-chain settings using the approach described above. We first describe the data for our model, then the portfolio compositions, and finally the results of our analysis.

Description of Data. We have collected data using the CoinMarketCap API over six different periods that span a total of 1264 hours of data, separated into 5-minute slots. The duration of these periods can be found in Table 6 in Appendix C. The final period includes the collapse of the FTX exchange on November 11, 2022. Our main motivation was to choose a diverse range of market conditions, including a period of turmoil, to ensure that the estimated system failure probabilities represent a worst-case scenario (i.e., in more typical market conditions, CroCoDAI would be safer). In each time slot, we collected price information of 100 tokens, namely those that were the 100 largest in terms of market capitalization during the first time slot. A complete list of the included token symbols, and the observed correlation between them, can be found in Figure 6 in Appendix C.

DSS-Based Portfolios. For our analysis, we have considered single-asset portfolios consisting of Bitcoin and several of the highest-value smart contract platforms, namely: Ethereum, Cardano, Solana, Polkadot, Tron, Avalanche, Algorand, and EOS – their cryptocurrency symbols are: BTC, ETH, ADA, SOL, DOT, TRX, AVAX, ALGO, and EOS. The results are displayed in Table 1 (single-asset). We use the following multi-asset portfolios:

- (1) D-All, which approximates the collateral portfolio of the DSS as of early 2023 (see Table 5 in Appendix B). (We replace GUSD, which was not in our dataset, with another fiat-backed stablecoin (USDT), and Maker's real-world assets with gold-backed PAXG.)
- (2) D-ERC, which approximates the DSS portfolio after removing all fiat-backed assets.
- (3) D-Mix1 and D-Mix2, which use the same tokens as D-ERC but which increase the proportion of the minor ERC-20 tokens.
- (4) C-Mix1 and C-Mix2, which replace the ERC-20 tokens in D-Mix1 and D-Max2 with tokens from other popular blockchain platforms: ADA, DOT, TRX, and SOL.

We use these portfolios to show how the system performs with a mix of different types of collateral: other (fiat-backed) stablecoins, major cryptocurrencies (e.g., ETH/WBTC), and minor ERC-20

tokens. Tokens of the first type typically have much lower volatility, but introduce a dependence on trusted third parties. We will discuss this trade-off in more detail later in this section. The exact compositions of the portfolios can be found in Table 7 in Appendix D.

In our experiments, the expected over-collateralization to avert system failure is $\theta = 1.1$. For each table entry, we calculate the probability of failure within 1 day, i.e., 288 5-minute time slots, using 100 000 simulation runs. In each run, we simulate the evolution of token prices by randomly drawing, in each consecutive time slot, the price change for each token type in the portfolio. We then compute the risk of failure as the fraction of runs that resulted in failure among the total number of runs. The results are displayed in Table 1. Since the 95% confidence intervals widths are very small for the results in this table, we have omitted them for brevity.

From Table 1, we observe a considerable difference between the failure probabilities of the single-token portfolios. The most high-risk portfolio was Solana (SOL), which was severely affected by the collapse of FTX, since it was one of FTX’s most prominent backers [17]. Even for an initial collateralization $\gamma' = 1.5$, a pure SOL portfolio would have a failure probability of more than 4% after a day, given the data from our observation period. In contrast, BTC and TRX were the most stable cryptocurrencies over this period.

Furthermore, we observe from Table 1 that the DSS’s approximate portfolio has a very low probability of failure due to 80% of its collateral being stablecoins and other pegged assets. In fact, for $\gamma' = 1.2$, its observed failure probability (0.001 for D-All) was roughly 40-50 times lower than for the pure crypto-backed portfolios (0.051, 0.040, and 0.050 respectively for D-Erc, D-Mix1, and D-Mix2). For the pure crypto-backed portfolios, we find that D-ERC has slightly lower failure probabilities than a pure ETH portfolio, which can be explained by the higher share of low-volatility

Table 1. Chance of failure for single-asset portfolios and initial over-collateralization γ' .

γ'	single-asset									multi-asset					
	BTC	ETH	ADA	SOL	DOT	TRX	AVAX	ALGO	EOS	D-All	D-ERC	D-Mix1	D-Mix2	C-Mix1	C-Mix2
1.2	0.021	0.047	0.078	0.228	0.063	0.018	0.126	0.166	0.097	0.001	0.051	0.040	0.050	0.030	0.029
1.3	0.003	0.008	0.014	0.093	0.012	0.003	0.029	0.051	0.016	0.000	0.007	0.005	0.006	0.003	0.003
1.4	0.001	0.003	0.005	0.059	0.004	0.001	0.009	0.019	0.004	0.000	0.002	0.002	0.002	0.001	0.001
1.5	0.001	0.001	0.002	0.041	0.002	0.001	0.004	0.007	0.002	0.000	0.001	0.001	0.001	0.001	0.001

Table 2. Chance of failure for single-asset portfolios and initial over-collateralization γ' . In the table, ‘0.0 ± 0.0’ indicates that system failure events were observed during the 1,000,000 simulation runs, but that the estimated probability has been rounded down to 0, whereas ‘—’ indicates that no failure events were observed.

γ'	single-asset					C-Mix1, stablecoin contribution δ				
	USDT	USDC	BUSD	TUSD	DAI	$\delta = 0.5$	$\delta = 0.25$	$\delta = 0.1$	$\delta = 0.05$	$\delta = 0$
vol.	4.85e-5	9.70e-5	4.06e-4	3.48e-4	2.84e-4	7.86e-4	1.17e-3	1.39e-3	1.47e-3	1.55e-3
1.2	0.00001 ±2.2e-6	0.00009 ±9.2e-6	0.00177 ±4.2e-5	0.00103 ±3.2e-5	0.00078 ±2.8e-5	0.00314 ±5.6e-5	0.01198 ±1.1e-4	0.02153 ±1.5e-4	0.02548 ±1.6e-4	0.02976 ±1.7e-4
1.3	0.0 ±0.0	0.00002 ±4.7e-6	0.00041 ±2.0e-5	0.00027 ±1.6e-5	0.00023 ±1.5e-5	0.00047 ±2.2e-5	0.00137 ±3.7e-5	0.00243 ±4.9e-5	0.00304 ±5.5e-5	0.00345 ±5.9e-5
1.4	—	0.00001 ±3.0e-6	0.00021 ±1.4e-5	0.00012 ±1.6e-5	0.00009 ±9.3e-6	0.00018 ±1.3e-5	0.00052 ±2.3e-5	0.00098 ±3.1e-5	0.00105 ±3.2e-5	0.00119 ±3.5e-5
1.5	0.0 ±0.0	0.00001 ±2.8e-6	0.00012 ±1.1e-5	0.00001 ±8.3e-6	0.00001 ±7.7e-6	0.00009 ±9.6e-6	0.00029 ±1.7e-5	0.00046 ±2.1e-5	0.00033 ±1.8e-5	0.00065 ±2.5e-5

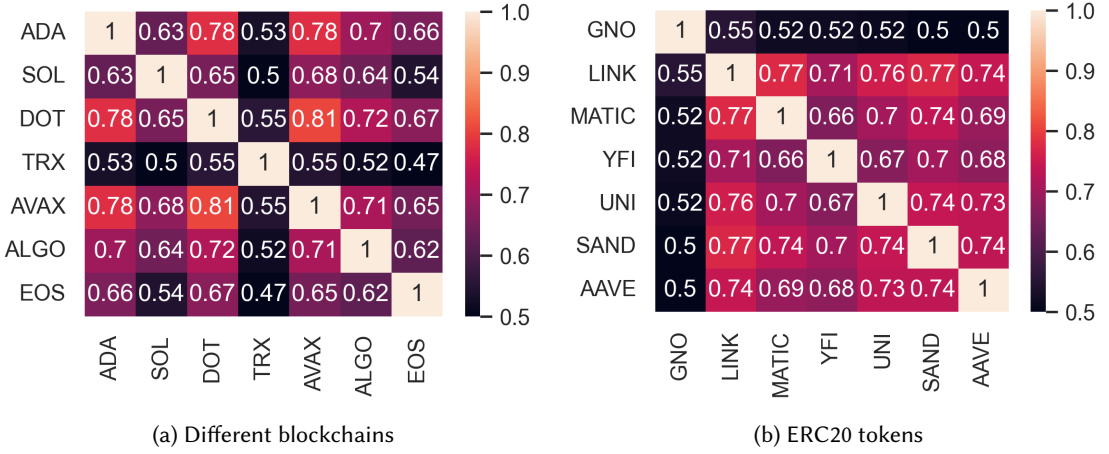


Fig. 5. Correlation between the different token types.

WBTC in the portfolio. Similarly, C-Mix1 and C-Mix2 portfolios have lower risk than the D-ERC due to the inclusion of more WBTC. We do not observe much benefit from the inclusion of more different asset types due to the high correlation between the included tokens. However, when comparing the C-Mix1 and C-Mix2 portfolios we do note that C-Mix2 has a lower failure probability despite the inclusion of less low-volatility WBTC. The reason is the inclusion of TRX, which is not only similarly low-risk as WBTC, but which also has a relatively low correlation with the other asset types – this can be seen in Figure 5a. We also note that the relative differences become starker when γ' increases – e.g., for $\gamma' = 1.2$, the risk of D-Mix1 is roughly a third higher than C-Mix2 (4.0% vs. 2.9%), whereas for $\gamma' = 1.3$ the risk is two thirds higher (0.5% vs. 0.3%).

Comparison to Existing Stablecoins. From Table 1, we observe that D-All has a considerably lower failure risk than the other portfolios. D-All represents the collateral portfolio of the DSS and is the only multi-asset portfolio from Table 1 that includes other stable assets (around 75% as per Table 5). All stablecoins except Dai in our dataset are fiat-backed stablecoins, which exhibit much smaller price volatility than regular tokens, but rely on trusted third parties which expose the system to external black swan events. We conduct experiments to further understand the price volatility. In particular, we estimate the failure risks for single-stablecoin portfolios and for a range of DSS-based portfolios with varying degrees of included stablecoins. For the latter, we consider modifications of the C-Mix1 portfolio, which represents a portfolio similar to Dai's collateral portfolio but which includes a minor fraction (20%) of tokens from other blockchains. We modify this portfolio by assigning a fraction δ of fiat-backed stablecoins to it and multiplying the contribution of the other token types by $1 - \delta$, for varying values of δ . The low volatility of other stablecoins means that the failure probabilities of single-stablecoin portfolios are very low. Therefore, we conduct 1,000,000 runs for each table entry and display the 95% confidence interval half-width below each estimate in Table 2. We also display the variance, which is strongly related to the failure risks, of each portfolio in the table (under vol.).

From Table 2, we observe that the failure risks of all single-stablecoin portfolios are much lower than those of the single-asset portfolios in Table 1, although there are still considerable differences between individual stablecoins: e.g., BUSD's volatility is 10 times higher than USDT's, and, for γ' , BUSD's estimated 1-day probability of failure is 177 times higher than that of USDT. For the

modified C-Mix1 portfolios, we have used USDC as a representative stablecoin because of its high trade volume (there have been over 20bn USDC in circulation since July 2021) and because its volatility is neither exceptionally high nor low among the included stablecoins. It can be seen that, as expected, the inclusion of other stablecoins leads to low failure risks, although we observe a small decrease for higher values of γ' : in particular, the relative difference between $\delta = 0$ and $\delta = 0.25$ is roughly $2.5\times$ (0.02976 vs. 0.01198), but this decreases to $2.24\times$ for $\gamma' = 1.5$. For $\delta = 0$ and $\delta = 0.5$, the relative difference decreases from $9.48\times$ for $\gamma' = 1.1$ to $7.22\times$ for $\gamma' = 1.5$. Overall, we observe that the inclusion of stablecoin fraction of 25% can reduce the system's failure risk by more than 60%.

We note that a stablecoin system that is overcollateralized through a *mixture* of fiat-backed stablecoins is more resilient to black-swan events than the individual stablecoins. The reason is that the collapse of any *single* trusted third party would have a limited impact. However, black swan events may affect multiple fiat-backed stablecoins simultaneously, e.g., due to new legislation affecting stablecoin collateral. Thus, it would still be advisable to ensure that the *total* contribution of fiat-backed stablecoins is limited. We also note, that, in theory, a principal crypto-backed stablecoin could include other crypto-backed stablecoins as collateral. In this setting, the portfolio of the supporting stablecoins should be included in the portfolio of the principal stablecoin. Finally, protocol designers with substantial fiat assets (e.g., Dai) also have the option to steer collateral portfolios towards those with minimal risk [38], which would reduce the risks associated with the crypto collateral even further.

6 IMPLEMENTATION

In this section, we present a prototype implementation of CroCoDAI and compare its performance in terms of both cost and processing time to a baseline that combines the existing DSS with wrapped tokens. We present three different subvariants of the relay chain design with a quorum of nodes (i.e., the offchain variant of Section 3.3). Each subvariant has a different tradeoff in terms of scalability and generality, and consists of two smart contracts written in Ethereum's Solidity language: Relay and CCTransfer, with only Relay different between the three subvariants. The Relay and CCTransfer contracts interact with an ERC-20 compliant contract for token transfers, Coin, which re-uses code from the Dai contract in the DSS [61]. We discuss the smart contracts in more detail in Section 6.1. The other components depicted in Figure 3 (i.e., the vault, auction, and oracle contracts) are also re-used from the DSS. We perform two sets of experiments: microbenchmarks for the relay subvariants with varying quorum sizes, and end-to-end experiments for CroCoDAI's main operations: creating new tokens, and cross-chain transfers.

6.1 Smart Contracts

CCTransfer. This contract stores cross-chain transfer requests as a hashmap of Request instances, which are implemented using a Solidity struct. Each CCTransfer contract is linked to a specific Coin and Relay contract pair, and must be made a Coin contract ward to allow it to mint or burn stablecoins upon the commitment of cross-chain transfers.

Relay. In the first design (named n/n), a hashmap of relay node addresses in Solidity's address format is stored in the contract. For each cross-chain transfer, at least $2f + 1$ relay nodes must call the Relay contract's approve function using their stored address. This design exploits the EVM's built-in signature verification, and generalizes trivially to other blockchain platforms. However, the downside is that each relay node must have an account and spend gas on each supported blockchain for every cross-chain transfer. In the second design ($n/1$), a hashmap of relay node public keys is stored in the contract, and a batch of $2f + 1$ 32-byte signatures is passed in the form of a byte array as a function parameter for each transaction. These signatures are then validated one by

one by repeatedly calling the built-in `ecrecover` function. Although this is not as generic as the previous approach (e.g., each chain with its different cryptographic primitives would require its own off-chain signature generation script) it is more efficient and does not require each individual relay node to maintain an account. In the third design (1/1), we use a t -out-of- n threshold signature scheme (where in our case $t = 2f + 1$) to produce a single 32-byte signature for the relay node quorum, e.g., using the procedure of [33]. The resulting signature is a standard ECDSA signature that can be verified using `ecrecover`. For our implementation we have used the multi-party ECDSA library by ZenGo X [83].

6.2 Relay Microbenchmarks

Table 3 compares the different designs in terms of the gas cost and the processing time at the relay nodes, T_{proc} . Both are essential to system performance: if the on-chain cost is too high then each transfer will incur large fees, whereas if the computation time is too high then cross-chain transactions incur a large delay. For the experiments, we used a Lenovo IdeaPad with an Intel(R) Core(TM) i5-1035G1 CPU (1.00GHz) and 8GB RAM – however, the gas costs are machine-independent. The scripts that performed off-chain computations use Python’s Web3 package for cryptographic operations, and the underlying blockchain was simulated using Truffle’s Ganache tool [31]. To execute our experiments, we ran the Python scripts in different terminals that each interacted with an active Ganache instance on a single machine. From Table 3, we observe that the first subvariant is the worst in terms of performance, which is the cost of its generality and ease of implementation. The second subvariant is more efficient in terms of gas costs for large quora, and only has little processing time at the relay nodes. The third subvariant is the most efficient in terms of gas costs, but at the cost of considerable processing time at the relay nodes, especially for large quora. For the first subvariant (i.e., n/n), we were unable to distinguish T_{proc} from the transaction processing time in Ganache as both happen in a single step in Python’s Web3 package – however, these are assumed to be similar to signing in the $n/1$ design, and are indicated by ‘–’ in the table.

6.3 End-to-End Experiments

As discussed in Section 4.6, a baseline that uses the existing DSS with a greater variety of wrapped tokens has similar resilience to black swan events and price fluctuations as CROCODAI. However, we provide experimental results in this section that demonstrate that the DSS baseline has worse end-to-end performance than CROCODAI. We focus on the two main stablecoin actions: (1) creating new stablecoins on a coin chain and (2) transferring stablecoins from one coin chain (Chain A) to another (Chain B). To ensure that gas costs are comparable, we implement Chain A, Chain B, and the relay chain as private Ethereum chains using `geth`. For collateral, we implement a custom ERC20 token contract on the underlying chain (GemCoin) – for the DSS baseline, we also require a bridge for the custom token on Chain A (GemBridge), and another (DaiBridge) for Dai on the relay chain. For the collateral vault, we use instances of `Vat`, `DaiJoin`, and `GemJoin` where the latter is linked to the GemCoin contract – in CROCODAI the vault contracts exist on Chain A, whereas in

Table 3. Performance comparison of the three implementation subvariants.

design	gas cost			T_{proc} (s)		
	$n = 4$	$n = 10$	$n = 16$	$n = 4$	$n = 10$	$n = 16$
(1) n/n	171791	511473	977350	–	–	–
(2) $n/1$	214373	430690	647118	0.025	0.071	0.091
(3) 1/1	53112	53117	53110	1.101	7.780	34.513

Table 4. Computation costs (units of gas) of the transactions in CroCoDAI and the DSS baseline, grouped by block. **Bold**: relay chain computations.

		CroCoDAI		DSS baseline	
		creation	transfer	creation	transfer
Block 1	(Chain A)	466537	229325	244036	27762
Block 2	(Relay)		133280	770724	294016
Block 3	(Chain B)		185503		71498
Block 3'	(Chain A)			71498	
Total	(coin chains)	466537	414828	315534	99260
Total	(relay chain)	0	133280	770724	294016

the DSS baseline they exist on the relay chain. Finally, the baseline requires a `WrGemCoin` contract on the relay chain to implement the wrapped collateral token as an ERC20 token, and a `WrDai` contract for wrapped Dai tokens on each coin chain.

Creating Stablecoins. For CroCoDAI, the procedure for creating new stablecoins is as follows: the user first transfers the collateral to the vault and creates a CDP using `GemCoin` and `GemJoin`. Next, she uses the CDP to create stablecoins and transfers them to her account using `Vat` and `DaiJoin`.

For the DSS baseline, the procedure is as follows. The user first sends `GemCoin` tokens to the bridge using `GemCoin` and `BaseBridge`. Next, a relay detects the transfer and creates an equivalent amount of wrapped `GemCoin` tokens for the user using `WrGemCoin`. The user then transfers the collateral to the vault, creates a CDP, creates stablecoins, and transfers them to her account using `WrGemCoin`, `GemJoin`, `Vat` and `DaiJoin`. Next, the user sends the Dai tokens to the bridge using `Dai` and `DaiBridge`. Finally, a relay detects the transfer and mints an equivalent amount of wrapped Dai tokens for the user using `WrDai`.

Cross-Chain Transfers. For CroCoDAI, the procedure for transferring stablecoins from Chain A to B is as follows. The user first transfers the tokens to `CCTransfer` and creates a transfer request. Next, a relay detects the request, sends a signed approval to the relay chain and `Relay` on Chain A, which calls `Coin` to burn the stablecoins. Finally, the relay sends the signed message to `Relay` on Chain B, which calls `Coin` to mint the stablecoins.

For the DSS baseline, the procedure is as follows: the user first burns her wrapped Dai tokens using `WrDai`. Upon detection by a relay, the burned Dai is refunded to the user on the relay chain. The user then transfers the Dai to the `DaiBridge` for Chain B. Finally, a relay detects the transfer and mints an equivalent amount of wrapped Dai tokens for the user using `WrDai` on Chain B.

Results. Table 4 displays the cost of the processes discussed above, with transactions that can occur within the same block grouped together. For the DSS baseline, we assume that a single trusted custodian acts as a relay – to keep the comparison with CroCoDAI as fair as possible, we hence use the first prototype subvariant (n/n) but with a single signer. For our implementation, we assume that the recipient data for outgoing cross-chain transfers (i.e., the ID of the receiving chain and the recipient’s account data) is not written to the blockchain; even in the fully on-chain design (see Section 3.3), this data can be exchanged off-chain and incorporated in the commit message or zero-knowledge proof.

We observe that the DSS baseline has considerably higher gas costs (namely 1 086 258 vs. 466 537 gas, or $\approx 130\%$ more) for creating stablecoins, due to the expensive two-way bridge – furthermore, it requires three blocks on two chains, whereas CroCoDAI can perform all operations within a single

Chain A block. This is a major difference because, to ensure security against fork attacks, blocks must be finalized before their information is processed on another chain – e.g., using Ethereum’s Casper FFG [11], which currently takes between one and two 32-block epochs in Ethereum. As such, local stablecoin creation is considerably cheaper and faster in CROCoDAI than in the baseline system, which allows CROCoDAI to exploit the availability of local collateral without depending on it. For cross-chain transfers, the total gas costs are slightly lower for the DSS baseline than for CROCoDAI (393 276 vs. 548 108), but the DSS baseline consumes nearly twice as much gas on the relay chain (**bold**), which is in the DSS is Ethereum. This means that as long as gas is roughly 1.96× cheaper on the coin chains than on Ethereum, then CROCoDAI has cheaper cross-chain transactions – this is a reasonable assumption in practice, as lower transaction fees are a key feature of smaller chains. For example, transactions on Arbitrum and Optimism were typically at least 8× cheaper than on Ethereum in September 2023 [52].

7 CONCLUSIONS AND FUTURE WORK

We have presented CROCoDAI, a stablecoin for cross-chain commerce. CROCoDAI facilitates cross-chain transfers by design, and is more resilient than a siloed baseline to black swan events and price fluctuations. We have presented a prototype implementation that relies on a quorum of relay nodes for cross-chain transfers and governance decisions, and which has better performance than a DSS-based baseline in terms of (expensive) transactions on the relay chain. For future work, an interesting direction is to implement a version of CROCoDAI that performs all relay operations on-chain, as per the second variant of Section 3.3 – e.g., using the approach of zkBridge, or succinct proofs of state as in Algorand or Mina. Another interesting direction is to further investigate alternative collateral liquidation mechanisms – for example, Salehi et al. [70] propose to mitigate liquidations from the DSS or remove them completely. A final interesting research direction is system recovery after a 51% attack on a coin chain.

8 ACKNOWLEDGEMENTS

This work was supported by Ministry of Education (MOE) Singapore’s Tier 2 Grant Award No. MOE-T2EP20120-0003.

REFERENCES

- [1] Maria Apostolaki, Gian Marti, Jan Müller, and Laurent Vanbever. SABRE: Protecting Bitcoin against routing attacks. In *NDSS*. The Internet Society, 2019.
- [2] Rafael Belchior, Peter Somogyvari, Jonas Pfannschmidt, André Vasconcelos, and Miguel Correia. Hephaestus: Modeling, analysis, and performance evaluation of cross-chain transactions. *IEEE Transactions on Reliability*, 2023.
- [3] Rafael Belchior, André Vasconcelos, Sérgio Guerreiro, and Miguel Correia. A survey on blockchain interoperability: Past, present, and future trends. *ACM Computing Surveys (CSUR)*, 2021.
- [4] Shreyas Bhat, Ayten Kahya, Rohit Kumar, and Bhaskar Krishnamachari. Simulating the MakerDAO stablecoin. In *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pages 1–2. IEEE, 2021.
- [5] Tom Blackstone. Circle launches cross-chain usdc transfer protocol for ethereum, avalanche. <https://cointelegraph.com/news/circle-launches-cross-chain-usdc-transfer-protocol-for-ethereum-avalanche>, 2023.
- [6] Joseph Bonneau, Izaak Meckler, Vanishree Rao, and Evan Shapiro. Mina: Decentralized cryptocurrency at scale. <https://docs.minaprotocol.com/assets/technicalWhitepaper.pdf>, 2020.
- [7] Antonio Briola, David Vidal-Tomás, Yuanrong Wang, and Tomaso Aste. Anatomy of a stablecoin’s failure: The Terra-Luna case. *Finance Research Letters*, 51:103358, 2023.
- [8] Jonah Brown-Cohen, Arvind Narayanan, Alexandros Psomas, and S Matthew Weinberg. Formal barriers to longest-chain proof-of-stake protocols. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 459–473, 2019.
- [9] Jeff Burdges, Alfonso Cevallos, Peter Czaban, Rob Habermeier, Syed Hosseini, Fabio Lama, Handan Kilinc Alper, Ximin Luo, Fatemeh Shirazi, Alistair Stewart, and Gavin Wood. Overview of Polkadot and its design considerations. arXiv 2005.13456, 2020.

- [10] Vitalik Buterin. A next-generation smart contract and decentralized application platform, 2014.
- [11] Vitalik Buterin, Daniël Reijnsbergen, Stefanos Leonardos, and Georgios Piliouras. Incentives in ethereum’s hybrid Casper protocol. *International Journal of Network Management*, 30(5):e2098, 2020.
- [12] Yizhou Cao, Min Dai, Steven Kou, Lewei Li, and Chen Yang. Designing stable coins. *Available at SSRN 3856569*, 2021.
- [13] Miguel Castro, Barbara Liskov, et al. Practical Byzantine fault tolerance. In *OsDI*, volume 99, pages 173–186, 1999.
- [14] cBridge. <https://cbridge.celer.network/bridge/ethereum-flow/>, 2024 (last accessed).
- [15] Celer. <https://cbridge.celer.network/1/56/USDC>, 2023 (last updated).
- [16] Jason Chen, Kathy Fogel, and Kose John. Understanding the Maker protocol. *arXiv preprint arXiv:2210.16899*, 2022.
- [17] Morgan Chittum. Sam Bankman-Fried championed Solana. Now, execs are trying to break the connection as FTX’s implosion weighs on the blockchain’s reputation. <https://markets.businessinsider.com/news/currencies/sam-bankman-fried-ftx-solana-blockchain-crypto-sol-damage-reputation-2023-1>.
- [18] cLabs Team. The Celso protocol: A multi-asset cryptographic protocol for decentralized social payments, 2019.
- [19] Dai Stats. <https://daistats.com/>.
- [20] Dipanjan Das, Priyanka Bose, Nicola Ruaro, Christopher Kruegel, and Giovanni Vigna. Understanding security issues in the NFT ecosystem. In *ACM CCS*, pages 667–681, 2022.
- [21] Anwitaman Datta, Daniël Reijnsbergen, Jingchi Zhang, and Suman Majumder. BlockChain I/O: Enabling Cross-Chain Commerce. *IEEE Access*, 2024.
- [22] Stelios Daveas, Kostis Karantias, Aggelos Kiayias, and Dionysis Zindros. A gas-efficient superlight Bitcoin client in Solidity. In *ACM AFT*, 2020.
- [23] DefiLlama. <https://defillama.com>, 2023 (last accessed).
- [24] Annette J. Dobson and Adrian G. Barnett. *An introduction to generalized linear models*. CRC press, 2018.
- [25] Daniel Engel, Maurice Herlihy, and Yingjie Xue. Failure is (literally) an option: Atomic commitment vs optionality in decentralized finance. In *SSS 2021*, 2021.
- [26] Ethereum Gas Tracker. <https://etherscan.io/gastracker>, 2024 (last accessed).
- [27] Ethereum.org. Blockchain bridges. <https://ethereum.org/en/bridges/>, 2023 (last updated).
- [28] European Central Bank. Eurosystem proceeds to next phase of digital euro project. <https://www.ecb.europa.eu/press/pr/date/2023/html/ecb.pr231018~111a014ae7.en.html>, 2024.
- [29] Zeke Faux. Anyone seen Tether’s billions? Bloomberg, 2021. <https://fortune.com/2021/10/15/tether-crypto-stablecoin-fined-reserves>.
- [30] Michael J Fischer, Nancy A Lynch, and Michael S Paterson. Impossibility of distributed consensus with one faulty process. *Journal of the ACM (JACM)*, 32(2):374–382, 1985.
- [31] Ganache. <https://trufflesuite.com>, 2023 (last accessed).
- [32] Juan Garay, Aggelos Kiayias, and Nikos Leonardos. The Bitcoin backbone protocol: Analysis and applications. In *EuroCrypt*, pages 281–310. Springer, 2015.
- [33] Rosario Gennaro and Steven Goldfeder. Fast multiparty threshold ECDSA with fast trustless setup. In *ACM CCS*, pages 1179–1194, 2018.
- [34] Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. Algorand: Scaling byzantine agreements for cryptocurrencies. In *ACM SOSP*, 2017.
- [35] John M Griffin and Amin Shams. Is Bitcoin really untethered? *The Journal of Finance*, 75(4):1913–1964, 2020.
- [36] Klaus Grobys, Juha Junntila, James W Kolari, and Niranjana Sapkota. On the stability of stablecoins. *Journal of Empirical Finance*, 64:207–223, 2021.
- [37] Wanyun Catherine Gu, Anika Raghuvanshi, and Dan Boneh. Empirical measurements on pricing oracles and decentralized governance for stablecoins. *Available at SSRN 3611231*, 2020.
- [38] Bretislav Hajek, Daniël Reijnsbergen, Anwitaman Datta, and Jussi Keppo. Collateral portfolio optimization in crypto-backed stablecoins. *International Conference on Mathematical Research for Blockchain Economy (MARBLE)*, 2024.
- [39] Maurice Herlihy, Barbara Liskov, and Liuba Shrira. Cross-chain deals and adversarial commerce. *The VLDB Journal*, 2021.
- [40] Ivan Homoliak, Sarad Venugopalan, Daniël Reijnsbergen, Qingze Hum, Richard Schumi, and Pawel Szalachowski. The security reference architecture for blockchains: Toward a standardized model for studying vulnerabilities, threats, and defenses. *IEEE Communications Surveys & Tutorials*, 23(1):341–390, 2020.
- [41] Charles Hoskinson. Why we are building Cardano. <https://whitepaper.io/document/581/cardano-whitepaper>, 2017.
- [42] Kostis Karantias, Aggelos Kiayias, and Dionysis Zindros. Proof-of-burn. In *Financial Cryptography and Data Security: 24th International Conference, FC 2020, Kota Kinabalu, Malaysia, February 10–14, 2020 Revised Selected Papers 24*, pages 523–540. Springer, 2020.
- [43] Sam Kazemian, Jason Huan, Jonathan Shomroni, and Kedar Iyer. Frax: A fractional-algorithmic stablecoin protocol. In *2022 IEEE International Conference on Blockchain (Blockchain)*, pages 406–411. IEEE, 2022.
- [44] Bryan Kelly and Hao Jiang. Tail risk and asset prices. *The Review of Financial Studies*, 27(10):2841–2871, 2014.

- [45] Olga Kharif. Hackers steal about \$600 million in one of the biggest crypto heists. Bloomberg, 2022. <https://www.bloomberg.com/news/articles/2022-03-29/hackers-steal-590-million-from-ronin-in-latest-bridge-attack>.
- [46] Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynkov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *CRYPTO*. Springer, 2017.
- [47] Ariah Klages-Mundt, Dominik Harz, Lewis Gudgeon, Jun-You Liu, and Andreea Minca. Stablecoins 2.0: Economic foundations and risk-based models. In *ACM AFT*, 2020.
- [48] Ariah Klages-Mundt and Andreea Minca. While stability lasts: A stochastic model of noncustodial stablecoins. *Mathematical Finance*, 32(4):943–981, 2022.
- [49] Abhiram Kothapalli, Srinath Setty, and Ioanna Tzialla. Nova: Recursive zero-knowledge arguments from folding schemes. In *Annual International Cryptology Conference*, pages 359–388. Springer, 2022.
- [50] Jae Kwon and Ethan Buchman. Cosmos whitepaper: A network of distributed ledgers. https://wikibiting.fx994.com/attach/2020/12/16623142020/WBE16623142020_55300.pdf, 2020.
- [51] Kyber Network, BitGo Inc, and Republic Protocol. Wrapped tokens: A multi-institutional framework for tokenizing any asset. <https://wbtc.network/assets/wrapped-tokens-whitepaper.pdf>, 2019.
- [52] L2 Fees. <https://l2fees.info/>, 2024 (last accessed).
- [53] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, 1982.
- [54] Rui Li and Saralees Nadarajah. A review of Student’s t distribution and its generalizations. *Empirical Economics*, 58:1461–1490, 2020.
- [55] Bowen Liu, Pawel Szalachowski, and Jianying Zhou. A first look into DeFi oracles. In *2021 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*, pages 39–48. IEEE, 2021.
- [56] Jordan Lyanchev. The biggest ever crypto hack: What happened in the Ronin bridge attack. CryptoPotato, 2022. <https://cryptopotato.com/the-biggest-ever-crypto-hack-what-happened-in-the-ronin-bridge-attack>.
- [57] Yiming Ma, Yao Zeng, and Anthony Lee Zhang. Stablecoin runs and the centralization of arbitrage. *Available at SSRN 4398546*, 2023.
- [58] Maker Team. The Dai stablecoin system: Whitepaper. <https://makerdao.com/whitepaper/DaiDec17WP.pdf>, 2017.
- [59] Maker Team. The Maker protocol: MakerDAO’s multi-collateral Dai (MCD) system. [https://makerdao.com/whitepaper/White%20Paper%20-The%20Maker%20Protocol_%20MakerDAO%E2%80%99s%20Multi-Collateral%20Dai%20\(MCD\)%20System-FINAL-%202021720.pdf](https://makerdao.com/whitepaper/White%20Paper%20-The%20Maker%20Protocol_%20MakerDAO%E2%80%99s%20Multi-Collateral%20Dai%20(MCD)%20System-FINAL-%202021720.pdf), 2017.
- [60] Maker Team. <https://github.com/makerdao/dss/tree/fa4f6630afb0624d04a003e920b0d71a00331d98>, 2022 (last updated).
- [61] Maker Team. <https://github.com/makerdao/dss/blob/master/src/dai.sol>, 2022 (last updated).
- [62] Peter Mell and Dylan Yaga. *Understanding stablecoin technology and related security considerations*. US Department of Commerce, National Institute of Standards and Technology, 2023.
- [63] Silvio Micali, Leonid Reyzin, Georgios Vlachos, Riad S Wahby, and Nikolai Zeldovich. Compact certificates of collective knowledge. In *IEEE S&P*, pages 626–641. IEEE, 2021.
- [64] MIP10c5: List of Active Oracles. <https://mips.makerdao.com/mips/details/MIP10c5SP#sentence-summary>, 2023 (last accessed).
- [65] Amani Moin, Kevin Sekniqi, and Emin Gün Sirer. SoK: A classification framework for stablecoin designs. In *Financial Crypto*, pages 174–197. Springer, 2020.
- [66] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.
- [67] Optimism Gas Tracker. <https://www.useweb3.xyz/gas/optimism>, 2024 (last accessed).
- [68] PayPal. PayPal stablecoin. <https://www.paypal.com/us/digital-wallet/manage-money/crypto/pyusd>.
- [69] Sheldon M. Ross. *An elementary introduction to mathematical finance*. Cambridge University Press, 2011.
- [70] Mehdi Salehi, Jeremy Clark, and Mohammad Mannan. Red-black coins: Dai without liquidations. In *DeFi Workshop, Financial Crypto ’21*. Springer, 2021.
- [71] Team Rocket. Snowflake to Avalanche: A novel metastable consensus protocol family for cryptocurrencies, 2018.
- [72] Team Rocket, Maofan Yin, Kevin Sekniqi, Robbert van Renesse, and Emin Gün Sirer. Scalable and probabilistic leaderless BFT consensus through metastability. *arXiv preprint arXiv:1906.08936*, 2019.
- [73] Sri AravindaKrishnan Thyagarajan, Giulio Malavolta, and Pedro Moreno-Sanchez. Universal atomic swaps: Secure exchange of coins across all blockchains. In *2022 IEEE S&P*, pages 1299–1316. IEEE, 2022.
- [74] Fabian Vogelsteller and Vitalik Buterin. ERC-20 token standard, 2015. <https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md>.
- [75] Xuechao Wang, Peiyao Sheng, Sreeram Kannan, Kartik Nayak, and Pramod Viswanath. Trustboost: Boosting trust among interoperable blockchains. *arXiv preprint arXiv:2210.11571*, 2022.
- [76] Sam M. Werner, Daniel Perez, Lewis Gudgeon, Ariah Klages-Mundt, Dominik Harz, and William J Knottenbelt. SoK: Decentralized finance (DeFi). *arXiv preprint arXiv:2101.08778*, 2021.

- [77] Tiancheng Xie, Jiaheng Zhang, Zerui Cheng, Fan Zhang, Yupeng Zhang, Yongzheng Jia, Dan Boneh, and Dawn Song. zkBridge: Trustless cross-chain bridges made practical. *ACM CCS*, 2022.
- [78] Jiahua Xu, Krzysztof Paruch, Simon Cousaert, and Yebo Feng. SoK: Decentralized exchanges (DEX) with automated market maker (AMM) protocols. *ACM Computing Surveys*, 55(11):1–50, 2023.
- [79] Yingjie Xue and Maurice Herlihy. Hedging against sore loser attacks in cross-chain transactions. In *ACM PODC*, pages 155–164, 2021.
- [80] Anatoly Yakovenko. Solana: A new architecture for a high performance blockchain, 2018.
- [81] Alexei Zamyatin, Mustafa Al-Bassam, Dionysis Zindros, Eleftherios Kokoris-Kogias, Pedro Moreno-Sanchez, Aggelos Kiayias, and William J Knottenbelt. SoK: Communication across distributed ledgers. In *Financial Crypto*, 2021.
- [82] Alexei Zamyatin, Dominik Harz, Joshua Lind, Panayiotis Panayiotou, Arthur Gervais, and William Knottenbelt. XCLAIM: Trustless, interoperable, cryptocurrency-backed assets. In *IEEE S&P*, 2019.
- [83] ZenGo X. <https://github.com/ZenGo-X/multi-party-ecdsa>, 2023 (last updated).
- [84] Rui Zhang, Rui Xue, and Ling Liu. Security and privacy on blockchain. *ACM Computing Surveys (CSUR)*, 52(3):1–34, 2019.
- [85] ZKM Research. Entangled rollups: Multi-chain interoperability without bridges, 2024.

A PROOFS OF SECURITY THEOREMS

Before we prove the main theorems, we need the following technical lemmas. We need the first to prove that if all CDPs are overcollateralized, then so is system as a whole.

LEMMA A.1. *If $a_i/b_i > c$ for $i = 1, \dots, n$, then $\frac{\sum_{i=1}^n a_i}{\sum_{i=1}^n b_i} > c$.*

PROOF. Note that $\frac{\sum_{i=1}^n a_i}{\sum_{i=1}^n b_i} = \sum_{i=1}^n \frac{a_i}{b_i} \frac{b_i}{\sum_{j=1}^n b_j}$. Since $\sum_{i=1}^n \frac{b_i}{\sum_{j=1}^n b_j} = 1$, $\frac{\sum_{i=1}^n a_i}{\sum_{i=1}^n b_i}$ is a convex combination of $a_i/b_i, \dots, a_n/b_n$. As such, it cannot be smaller than $\min_{i=1, \dots, n} a_i/b_i$, which is greater than c by the statement of the lemma. \square

The second lemma is a commonly-known result, but we include the proof here for completeness.

LEMMA A.2. *If X has a normal distribution with mean 0 and variance 1, then*

$$R(x) = \mathbb{P}(X > x) \leq \frac{e^{-x^2/2}}{x\sqrt{2\pi}}.$$

PROOF. Since for $y \geq x \geq 0$ we have that $1 \leq \frac{y}{x}$, we have that

$$\begin{aligned} \mathbb{P}(X > x) &= \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-y^2/2} dy \\ &\leq \frac{1}{\sqrt{2\pi}} \int_x^\infty \frac{y}{x} e^{-y^2/2} dy = \frac{e^{-x^2/2}}{x\sqrt{2\pi}}. \end{aligned}$$

\square

With these lemmas, we are ready to prove the three theorems from Section 4.

PROOF OF THEOREM 4.1. If $p_{y,t} c_{j,t} / y_{j,t} > \gamma$ holds for all $j \in \mathcal{J}$, then by Lemma A.1 it must hold that

$$\frac{C_0^+}{D_0^+} > \gamma. \quad (3)$$

Furthermore, we know from the definition of the debt ceiling that $D_0^- < \zeta'(D_0^- + D_0^+)$, where ζ' (as defined in Section 4) is the total debt ceiling of the affected tokens. The above can be rewritten to

$$D_0^- < \zeta' / (1 - \zeta') D_0^+. \quad (4)$$

We therefore find that

$$\frac{C_1^*}{D_1^*} = \frac{C_0^+}{D_0^- + D_0^+} > \frac{C_0^+}{\left(1 + \frac{\zeta'}{1-\zeta'}\right) D_0^+} > \gamma(1 - \zeta')$$

where the first inequality holds because of (4) and the second because of (3). \square

PROOF OF THEOREM 4.2. Because of the debt ceilings on the tokens, the total debt incurred for each token type m must still satisfy $\sum_{\substack{j \in \mathcal{J}_t \\ y_j = m}} D_{jt} < \zeta_m \sum_{j \in \mathcal{J}_t} D_{jt}$. – as such, even after the attack, we must have $D_1^- = D_0^- + h' < \zeta'(D_0^+ + D_0^- + h')$ or

$$D_0^- + h' < \zeta' / (1 - \zeta') D_0^+ \quad (5)$$

where D_0^- , D_0^+ , and ζ^- are as defined in the previous section. We have that

$$\frac{C_1^*}{D_1^*} = \frac{C_0^- + C_0^+}{D_0^- + D_0^+ + h} > \frac{C_0^- + C_0^+}{\left(1 + \frac{\zeta'}{1-\zeta'}\right) D_0^+} > \gamma(1 - \zeta')$$

where the first inequality holds because of (5) and the second because $C_0^- > 0$ and because of (3). As such, we again find that the debt ceilings ensure that CROCoDAI remains fully backed if it is sufficiently overcollateralized. \square

PROOF OF THEOREM 4.3. In the following, we first assume that O is odd and that some subset $O' \subset O$ is controlled by malicious parties who aim to convince a vault in CROCoDAI that the price is at most than some $p^* > p$. In this case, the median equals the $\frac{1}{2}(M+1) + |O'|$ th order statistic from a sample of $M - |O|$ normally distributed random variables with mean p and standard deviation σ_o^2 . This order statistic is greater than the maximum of the $|M| - |O'|$ random variables, which we denote by Y . Let $\sigma_* = \min_{o \in O} \sigma_o$. Y 's cumulative distribution function equals

$$\begin{aligned} \mathbb{P}(Y \geq p^*) &= \mathbb{P}\left(\bigwedge_{o \in O \setminus M'} \Delta_o > p^* - p\right) = \prod_{o \notin O'} R\left(\frac{p^* - p}{\sigma_o}\right) \\ &\leq R^{|M| - |O'|} \left(\frac{p^* - p}{\sigma_*}\right) \leq \frac{e^{-(|M| - |O'|)(p^* - p)^2 / 2}}{(p^* - p)\sigma\sqrt{2\pi}}, \end{aligned}$$

where the first inequality holds because R is a monotonically decreasing function, and the second inequality follows from Lemma A.2. As such, the probability $\mathbb{P}(Y \geq p^*)$ vanishes exponentially as $p^* - p$ increases. The proof for $p^* < p$ is very similar and uses the symmetry of the normal distribution. \square

B DSS PARAMETRIZATION

The parameters of the various different vault types in the DSS can be found [here](#). A sample of parameter values as of 25 April 2023 (Ethereum block 17121982) can be found in Table 5. For each of the supported token type j , we indicate the token name, total value of the collateral of CDPs of this type $\sum_{j \in \mathcal{J}_t} p_{jt} c_{jt}$, total debt of the collateral of CDPs of this type $\sum_{j \in \mathcal{J}_t} D_{jt}$, collateralization ratio, and debt ceiling ζ_j . Although the exact values vary throughout the day, this table is meant to illustrate the typical composition of the collateral type in the DSS.

The following token types were supported in the DSS, but were later deprecated: AAVE, BAL, BAT, COMP, KNC, LRC, MANA, RENBTC, TUSD, UNI, USDT, and ZRX. The following liquidity pool tokens were deprecated: AAVE/ETH, LINK/ETH, UNI/ETH, wBTC/Dai, and wBTC/ETH.

Table 5. Information about the various collateral types used by the DSS (as of Ethereum block 17121982).

	Name	Token/Asset Platform	debt	coll. value	overcoll.	debt ceil.	stab. fee	liq. rate	contrib.
Standard ERC-20 Tokens	ETH-A	Ether	233580929	421276897	180%	1500000000	1.50%	145%	4.69%
	ETH-B	—	45480335	67797760	149%	250000000	3.00%	130%	0.91%
	ETH-C	—	306467435	407905059	133%	2000000000	0.75%	170%	6.16%
	WSTETH-A	Wrapped Ether	105440104	123821333	117%	500000000	1.50%	160%	2.12%
	WSTETH-B	—	198020222	212522461	107%	500000000	0.75%	185%	3.98%
	RETH-A	—	3503590	6503590	186%	20000000	0.50%	170%	0.07%
	WBTC-A	Wrapped Bitcoin	35684532	55869143	157%	500000000	1.75%	145%	0.72%
	WBTC-B	—	22535846	32535847	144%	250000000	3.25%	130%	0.45%
	WBTC-C	—	40106301	63565858	158%	500000000	1.00%	175%	0.81%
	LINK-A	ChainLink	711324	3708567	521%	5000000	2.50%	165%	0.01%
	YFI-A	Yearn	148553	1941526	1307%	4000000	1.50%	165%	0.00%
	MATIC-A	Polygon	194080	5193556	2676%	15000000	3.00%	175%	0.00%
GNO-A	Gnosis	3020632	5000000	166%	5000000	2.50%	350%	0.06%	
Total (ERC-20 tokens):			994893882	1407641598					20.00%
LP Tokens	UNIV2USDCETH-A	USDC/ETH	2139902	0	0%		1.50%	120%	0.04%
	UNIV2DAIUSDC-A	Dai/USDC	23983390	0	0%		0.02%	102%	0.48%
	GUNIV3DAIUSDC1-A	Dai/USDC	59841202	0	0%		0.02%	102%	1.20%
	GUNIV3DAIUSDC2-A	Dai/USDC	104141233	0	0%		0.06%	102%	2.09%
	CRVV1ETHSTETH-A	ETH/wETH	56643346	66542992	117%	100000000	1.50%	155%	1.14%
Total (Liquidity Pool ERC-20 tokens):			246749074	66542992					4.96%
Real-World Assets	RWA001-A	6s Capital	14348036	15000000	105%		3.00%		0.29%
	RWA002-A	Centrifuge	6429638	20000000	311%		3.50%		0.13%
	RWA003-A	—	1824487	2000000	110%		6.00%		0.04%
	RWA004-A	—	1468187	7000000	477%		7.00%		0.03%
	RWA005-A	—	6076346	15000000	247%		4.50%		0.12%
	RWA006-A	AEA	0	0			2.00%		0.00%
	RWA007-A	Monetalis	499950318	125000000	250%		0.00%		10.05%
	RWA008-A	SG Forge	0	30000000			0.05%		0.00%
	RWA009-A	H. V. Bank	100000000	100000000	100%		0.00%		2.01%
	RWA010-A	Centrifuge	0	20000000			4.00%		0.00%
	RWA011-A	—	0	30000000			4.00%		0.00%
	RWA012-A	—	13619456	30000000	220%		4.00%		0.27%
	RWA013-A	—	69525268	70000000	101%		4.00%		1.40%
Total (Real-World Assets):			713241736	1589000000					14.34%
PSMs	PSM-USDC-A	USD Coin	2118503205	2608700392	123%	10000000000			42.58%
	PSM-USDP-A	Pax Dollar	499997224	500000000	100%	500000000			10.05%
	PSM-GUSD-A	Gemini Dollar	387538598	437538598	113%	500000000			7.79%
Total (Peg Stability Modules):			3006039027	3546238990					60.42%

C PRICE DATA TABLE

Table 6 contains a specification of the time periods during which we collected price data for the empirical analysis of Section 5. Figure 6 visualizes the correlations between the 100 tokens in our dataset.

Table 6. Data periods (times are in Greenwich Mean Time).

	start	end	number of slots
Period 1	26-08-2022 02:16	30-08-2022 00:26	1131
Period 2	07-09-2022 03:15	15-09-2022 08:45	2371
Period 3	30-09-2022 07:37	02-10-2022 05:27	551
Period 4	07-10-2022 04:31	15-10-2023 22:46	2010
Period 5	17-10-2022 02:51	29-10-2022 16:26	3620
Period 6	04-11-2022 03:50	23-11-2022 05:15	5490

Table 7. DSS-based portfolio compositions.

portfolio name	token distribution
D-All	47% USDC, 19% ETH, 14% PAXG 10% USDP, 8% USDT, 2% WBTC
D-ERC	90.2% ETH, 9.4% WBTC, 0.2% GNO, 0.1% LINK, 0.05% MATIC, 0.05% YFI
D-Mix1	50% ETH, 30% WBTC, 5% GNO, 5% LINK, 5% MATIC, 5% YFI
D-Mix2	20% ETH, 20% WBTC, 15% GNO, 15% LINK, 15% MATIC, 15% YFI
C-Mix1	50% ETH, 30% WBTC, 5% ADA, 5% DOT, 5% TRX, 5% AVAX
C-Mix2	20% ETH, 20% WBTC, 15% ADA, 15% DOT, 15% TRX, 15% AVAX

Table 8. Optimal portfolio compositions.

name	optimal token distribution
D-Opt	WBTC 74.015%, GNO 25.985%
D-Opt ($\lambda = 0.2$)	WBTC 20%, ETH 20% GNO 20%, LINK 20%, YFI 20%
C-Opt	WBTC 62.204%, TRX 37.796%
C-Opt ($\lambda = 0.2$)	WBTC 20%, ETH 16.656%, ADA 14.493% DOT 20%, TRX 20%, EOS 8.851%
A-Opt	BSV 8.728%, BTG 0.796%, BTT 36.821%, DCR 4.357%, GNO 5.221%, HT 0.0762% KCS 2.726%, KLAY 5.705%, LEO 17.568% OKB 3.223%, TRX 13.017%, TWT 0.483%, XLM 0.593%
A-Opt ($\lambda = 0.1$)	BSV 10%, BTG 1.354%, BTT 10%, CAKE 0.644%, DCR 7.278%, GNO 10%, HT 1.742%, KCS 5.819%, KLAY 9.434%, LEO 10%, NEXO 0.311%, OKB 5.604%, TRX 10%, TWT 8.61%, WBTC 10%, XLM 6.951%

Table 9. Optimal simulation portfolio if other stablecoins are included.

BTT 0.34%	BUSD 0.1504%	PAXG 8.72%
USDC 35.90%	USDN 0.12%	USDT 39.89%