Frankencoin*

Luzius Meisser**
Meisser Economics

Basile Maire***
Desma Eight, LLC

January 19, 2022

*Preliminary

**luzius.meisser@gmail.com

***basile.maire@desma8.io.

Frankencoin

Abstract

blabla

JEL Classification Codes: D40, G23

1 Introduction

We start from the use-case of a Swiss Frank Lombard loan, collateralized with tokens such as Bitcoin, or tokenized shares. The borrower deposits collateral into the system and thereby mints a token, the “Frankencoin” ZCHF, that is pegged to the Swiss Frank. If the value of the borrower’s collateral falls below a certain threshold, the loan can be liquidated and the borrower incurs a haircut. If the value of the collateral falls below the loan value before the liquidation ends, the loss eats into capital. Implementing this system in a fully decentralized way on the Blockchain leads us to our main contribution.

First, we introduce an algorithmic stablecoin that allows for *different collateral types and liquidation rules*. This diversification of collateral and liquidation rules reduces the impact of non-systemic market events to the stablecoin peg. Second, we propose a specific mint plugin setup for which

- We use an *auction mechanism* that avoids using external Oracle price feeds
- To calibrate the system, we apply traditional risk management techniques that have not found their way into the world of stablecoins

The system entails a decentralized governance that governs risk parameters and can deny the community’s addition of new peg-methodologies if deemed unfit.

The Frankencoin can be minted by anyone using one of the available mint plugins. We define mint plugins as smart contracts that have the ability to mint Frankencoin in accordance with their rules. Mint plugins have to be approved by the governance msimilarechanism and there might be different types of mint plugins for different types of collateral and different liquidation mechanisms. Anyone can propose to add a new mint plugin and if it is approved by the governance mechanism, it can be used according to its rules. These rules can include interest rates that are owed by the minters, liquidation mechanisms, collateral requirements, and required ZCHF to be held by “stakers” as a reserve per ZCHF token minted. While every mint plugin defines their required reserve capital, the reserves are shared among all mint plugins.

Stakers are participants that lock their Frankencoin into the system. Stakers earn interest rates paid for by borrowers and in turn risk to lose Frankencoin in case the liquidation process is not able to recover the loan amount with the available collateral. Staked Frankencoin are subject to a lockup period. This ensures that in case of a "bank run" on the system, stakers are the last who can liquidate their Frankencoin and will therefore also suffer from the greatest losses while the other participants are likely to be able to reclaim the full value.

This paper focuses on two aspects of Frankencoin. First, we discuss how the Frankencoin embeds with the wider economy. Many existing stablecoin systems fall short of this aspect and can arguably only be sustained with significant growth of the system or demand of their own token, see e.g., [Clements, 2021]. We propose a specific setup of mint plugins for which we use standard risk methodology techniques to quantify adverse events and calibrate capital reserves and fees. This is in stark opposition to existing stablecoins that set parameters such as the collateralization-ratio without risk methodological quantification methods. *TODO: reference??*

2 Economy

The immediate participants of Frankencoin consist of *minters*, *users*, and *stakers*.

Minters deposit collateral into a mint plugin and thereby enter a Frankencoin Lombard loan. Minters benefit from the system to the extent that it allows them to use liquid or illiquid assets that they want to hold long-term, to generate short-term liquidity. This can be motivated e.g., by tax considerations in some juristicions, or by justifications that hold for traditional Lombard loans. Similar to Lombard loans, the position is overcollateralized.

Stakers stake Frankencoin in a dedicated contract, thereby locking these funds up for a certain period of time. As a consequence, stakers are the last to be able to liquidate their holdings in case of a collapse of the value of Frankencoin. Stakers have to be compensated for their risk. This compensation is provided by the minters. Fees and interest rates paid by the minters are distributed to the stakers. By staking Frankencoin, stakers release governance tokens that give them voting rights.

Users hold and transfer Frankencoin as a means of payment or store of value. No fees are charged to the users, but they also are not provided with any financial gain from holding Frankencoin by the system. The system should be designed such that the tail risk of a complete default is negligible for the users as that risk is outsourced to the stakers.

This setup separates risk-takers (stakers) from the users. Governance token holders have "skin in the game" and are thus incentivized to maintain a healthy system. For instance, because staker capital is shared for all mint plugins, stakers have a vested interest to retain a healthy ecosystem of mint plugins.

Creditors of Frankencoin loans should be paid a risk-free rate corresponding to the Swiss Frank, plus a risk-premium. Otherwise, there is no rationale why stakers should sustainably lock their funds in Frankencoin (other than for non-pecuniary reasons). From the perspective of the minters, this means that costs to minters should be in line with the market capital costs for their loan.

We now address the conditions under which the *peg to the Swiss Frank* should hold. We approach the valuation of the Frankencoin from the perspective of a *perpetual bond*. A perpetual bond, or consol, is a bond with coupon payments but no redemption date, see, e.g., [Jorion et al., 2010]. The staked Frankencoin is subject to default risk, because the system burns ZCHF when a minter's position is undercollateralized. We price this credit-risky perpetual along the lines of [Jarrow and Turnbull, 2000], by discounting the interest payments on a credit-risky term structure. Let's assume that interest payments happen at discrete time-steps $0, ..., \infty$ and we have corresponding risky rates of the term-structure so that the date-0 value of a promised Swiss Franc at time t of a credit-risky Franc promise is equal to $\exp(-r_t t)$. Let the constant coupon rate per Frankencoin be c. Now, the value of the perpetual can be written as

$$\begin{aligned} v(0) &= \sum_{t=0}^{\infty} c e^{-r_t t} && (1) \\ &= \sum_{t=0}^{\infty} c e^{-yt} && (2) \\ &= \frac{c}{1-e^{-y}}, && (3) \end{aligned}$$

where the second line replaces the time-specific discount rates by a yield, and the last line is an application of geometric series. For the value to be at par, $v(0) = 1$, we have to choose the coupon rate accordingly: $c = 1 - e^{-y}$. Hence, if the interest earned from staking ZCHF are in line with discounting, the present value of one ZCHF is equal to one Swiss Franc.

The credit risky term-structure corresponds to the Swiss Franc risk-free term-structure plus a spread that compensates the investor for the risks. Hence whenever the risk-free term-structure, or the Swiss Franc risk changes, c hast to be adapted for the value $v(0)$ to be equal to one. This is difficult to automate, and we therefore allow the stakers to collectively set the interest rate (i.e., the risk-free rate plus spread). That is, if the exchange rate of the ZCHF is too low, the stakers use the governance mechanism to increase the interest rate and vice versa.

We design the system so that it is in the interest of the stakers to set the parameters of the system such that the peg is maintained. There should be no abuse of power, for example to set interest rates too high, therefore pushing the value of their ZCHF way beyond one CHF and essentially stealing the collateral as it would become too expensive for the minters to buy ZCHF to get their collateral back. To prevent such an attack, we ensure that stakers can only slowly adjust the interest rate, so that it is possible for the participants to trade-in their ZCHF before rates are too punitive (e.g., minters redeem their collateral by repaying their loan).

The next section presents a specific setup of mint plugins and proposes a calibration method to determine the appropriate spread that should bring the value of the Frankencoin close to a valuation of one Swiss Franc.

3 Specific Mint Plugin Setup

The Frankencoin system is open to accept any type of mint plugins. In this paper we describe a setup with two specific mint plugins that we consider to be of particular relevance when bootstrapping and growing the Frankencoin.

Each mint plugin $i \in 1, ..., K$ charges a minting fee $\Theta_i^{(F)} \geq 0$, and can define a coupon rate paid to stakers $\Theta_i^{(I)} \geq 0$. Depending on the plugin type, there can be additional plugin-specific parameters.

3.1 Direct Peg Plugin

The simplest possible mint plugin is one that is based on a stablecoin with the same reference currency. Specifically, "off-chain" custodial stablecoins. For the Frankencoin, this could for example be the CryptoFranc (XCHF) issued by Bitcoin Suisse or the Digital Swiss Franc (DCHF) issued by Sygnum. This mint plugin allows anyone to deposit the specified stablecoin and to get Frankencoins in return. Also, the minting contract would allow anyone to convert Frankencoins back into the specific stablecoin for as long as there are any left.

Direct peg plugins have the advantage of strongly anchoring the value of the Frankencoin to one Swiss Franc by delegating the collateralization mechanism (e.g., directly backed by Swiss Francs). The disadvantage for direct peg plugins is the dependency on the issuers. Overall, direct peg are a great method to bootstrap the Frankencoin and diversify the Frankencoin system.

3.1.1 Fee Calibration

In case of issuer default, stakers have to burn ZCHF equal to the amount of loss given issuer default times the exposure to that stablecoin. To compensate stakers for this risk, we charge a minting fee. Again, we have the advantage that calibration is outsourced to the market. If the value of one stablecoin trades at $1-\delta$ to the Swiss Franc, governance sets a minting fee equal to $\Theta_F^{(i)} = \delta$.

3.1.2 Reserves

Each mint plugin defines the required reserves of ZCHF to be held against the issued volume of ZCHF. XCHF and DCHF come with their own guarantee that extend beyond the issuer default. Therefore, for the direct peg plugin and XCHF and DCHF as collateral, we require no reserves.

3.2 Liquid Collateral Plugin

The second type of mint plugin is designed for liquid collateral, such as Bitcoin. This type of plugin implements the use case of a Lombard loan that we motivated the paper with.

- The participants are minters, challengers, auction participants, and ZCHF stakers.
- The minter deposits collateral and thereby mints ZCHF. The ZCHF are overcollateralized at the time of minting, that is, the value of collateral deposited exceeds the value of the minted ZCHF.
- Challengers can initiate an auction process for a given position at any time. To do so, they deposit collateral of the same type. After the challenge initiation, auction participant bid for the collateral by despositing ZCHF.
    1. If, according to the auction, the value of the collateral falls below a specified threshold, the position is liquidated and the minter loses their collateral. *E.g., the collateral deposited is 1500 LUSD, 1000 ZCHF were minted. Now, the position is challenged and the best bid for 1500 LUSD closes at 1095 ZCHF. Let's assume that the threshold is 10%. Now, because 1095 ZCHF < 1000 (1+10%) ZCHF, the position is closed out.* The challenger earns a fee, and the bidder gets the the collateral he was bidding for. The ZCHF posted by the bidder are distributed as follows. The challenger receives a reward. An amount equal to the outstanding loan is burned:
        - If any ZCHF above the loan amount is left, the stakers get this amount
        - If the posted ZCHF are not sufficient to burn an amount equal to the outstanding loan, the stakers lose this amount
    2. If, according to the auction, the collateral value is above the threshold, the position remains in the minter's ownership. The bidder gets the challenger's collateral and the challenger get's the bidder's amount of ZCHF.

Loan issuance can be implemented by using the last auction price and the requirement of a collateral x-times above the last price, e.g., the minter sets the loan amount to their own value and it is granted if the amount of the collateral is at least equal to the last price times $x = 1.8$.

Figure 1 illustrates this auction mechanism graphically.

Our liquid collateral mint plugin has the following parameters. Variable h denotes the threshold that defines whether the position can be liquidated or not. That is, if, according to the auction,

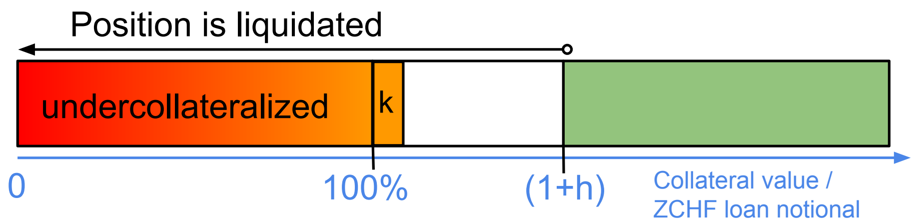

Figure 1: Auction. Any participant can challenge a loan position consisting of the ZCHF loan and the deposited collateral. The challenger deposits collateral of the size of the loan. Auction participants bid for the collateral (here BTC) by depositing ZCHF. If the best bid price in ZCHF is above (1+h) times the notional ZCHF lent, the challenger receives the best bidder's ZCHF and the bidder the challenger's collateral (green zone). If the best bid price is below (1+h), the borrower loses their collateral (position is liquidated). In this case the challenger gets back their collateral and earns a fee of k times the loan amount. The bidder gets the collateral deposited by the borrower. The ZCHF posted by the bidder are used to pay a reward k to the challenger, $(1+h-k)$ times the loan amount go to the staking pool, and an amount equal to the loan amount is burned. If the ZCHF are not sufficient to burn an amount equal to the loan amount after paying k, stakers have to burn this amount of ZCHF. Hence the unshaded area in the center is where stakers earn liquidation rewards, the area to the left is costly for stakers.

the value of the collateral is below $Z(1+h)$, where Z is the amount of ZCHF minted for a given position, the position is liquidated. Variable k denotes the challenger reward (e.g., 2% of the ZCHF position). Finally, τ is the duration of the auction (e.g., 24 hours). To summarize:

$$h : h > 0, \text{ liquidation threshold, liquidate if highest bid is below Z(1+h)} \tag{4}$$

$$\tau : \text{duration of liquidation process} \tag{5}$$

$$k : 0 < k < h, \text{ challenger reward} \tag{6}$$

Since our choice of collateral is liquid and can be traded at many different venues, we expect arbitrageurs to bid in the liquidation process and arbitrage between the liquid collateral mint plugin and other exchanges. This is best done via algorithmic trading, and we therefore propose to use a short liquidation horizon (hours, rather than days).

We define an "efficient liquidation process" as the situation in which the collateral is challenged at a price $(1+h)$, and at the end of the liquidation period τ, the collateral is sold at a competitive price. In practice, likely the challenge will not be issued at a price $(1+h)$ but rather at a lower price $(1+h')$ where $h' < h$. We proceed by setting the period τ to 24 hours and calibrate the remaining parameters: h, h', $\Theta_F^{(i)}$.

3.2.1 Fee Calibration

Stakers are at risk to lose funds between the liquidation start and liquidation end. Specifically, if during the liquidation horizon the value of the collateral drops from $(1+h')$ to below $(1+k)$, see Figure 1. The key challenge now is to value this risk so that the stakers are adequatly compensated.

We have a single period, of duration τ, a random variable, $\tilde{r}_\tau$, that corresponds to the log-return of the collateral over this period (using CHF as the numeraire). Now, we can express the loss to the stakers as the following random variable:

$$\tilde{L} = -\left[(1+h')e^{\tilde{r}_\tau} - (1+k)\right] \mathbf{1}_{\tilde{r}_\tau < \log \frac{1+h}{1+h'}}, \tag{7}$$

per unit of ZCHF minted (e.g., if the position consists of Z ZCHF, the loss to the stakers is $Z\tilde{L}$). To see this, first note that the starting value of the position is $(1+h')$ per unit of $ZCHF$. The value of the position at the end of the auction is $(1+h')e^{\tilde{r}_\tau}$. Stakers need to burn ZCHF for the amount that the end-of-auction value falls short of the minted amount plus the challenger reward, hence we subtract $(1+k)$. Second, the stakers earn in the liquidation process only if the return $\tilde{r}_\tau$ leads to a value below $(1+h)$, otherwise there is no liquidation, see Figure 1, so we need the indicator function $\mathbf{1}_{\cdot}$ that equals one if

$$\text{condition for liquidation:} \qquad (1+h')e^{\tilde{r}_\tau} < (1+h) \tag{8}$$

holds, zero otherwise. Note that if the liquidation starts at $h' = h$, the indicator function simplifies to $\mathbf{1}_{\tilde{r}_\tau<0}$. Finally, the negative sign is convention to have a positive number for a loss and a negative number for gains.

To tackle the valuation of $\tilde{L}$, we resort to the arbitrage free pricing principle, which states that the value of a contingent claim is given by its discounted expected value under the risk-neutral probability measure, see for example [Björk, 2009]. We assume that the risk-free rate used for discounting is equal to zero, which we consider adequate especially since the period τ is very short. Let $f_\tau(x)$ be the density function for the return distribution over the period τ. Now, we can value $\tilde{L}$ conditional on the starting level of liquidation h' as follows

$$\mathbb{E}_\tau\left[\tilde{L}|h'\right] = -\int_{-\infty}^{\log\frac{1+h}{1+h'}} \left[(1+h')e^x - (1+k)\right] f_\tau(x)dx \tag{9}$$

where the subscript τ emphasizes that the distribution depends on the time-horizon of the auction. If we knew the distribution of starting levels h', we could integrate out h' to arrive at $\mathbb{E}_\tau[\tilde{L}]$.

We now discuss h'. The challenger does not receive the challenger reward in case the auction finalizes at a price above $(1+h)$, but they instead swap their collateral posted against the best ZCHF bid. Even if the challenger is likely to get a competitive price, they are arguably more interested in collecting the 2% liquidation reward than performing a trade. From this perspective, the challenger would be better off not starting the challenge at $(1+h)$ but a price quite below that to increase the probability that the auction ends below $(1+h)$. However, if the challenger target too low entry points, other challengers could enter above, or prices could correct to above $(1+h)$ so that no more challenge is possible. Hence, challengers trade off the probability of collecting the reward given they issue a challenge versus the probability of not being able to enter a challenge anymore. To investigate this trade off, we assume that challengers issue a challenge when the collateral value reaches a level for which there is a probability α that they receive the challenger reward. Formally, using Equation (8), we determine h' so that the following equation holds:

$$\mathbb{P}\left[\tilde{r}_\tau < \log\frac{1+h}{1+h'}\right] = \alpha, \tag{10}$$

for a given α. We then assume that the liquidation level is given and equal to the value h' that solves (10).

Equation (9) uses the risk-neutral probability measure, often referred to as $\mathbb{Q}$, rather than the objective measure $\mathbb{P}$. In practice, parameters for the measure $\mathbb{P}$ are extracted directly from market data (e.g., sample volatilities and expected returns), whereas parameters for the measure $\mathbb{Q}$ have to be extracted from option data under the same model assumptions (e.g., option implied volatilities). With risk averse investors, the $\mathbb{Q}$-measure puts more weight on adverse market events, see for example [Breeden and Litzenberger, 1978], leading to a higher risk-neutral price of $\tilde{L}$, compared to the value obtained when integrating Equation (9) under the objective measure. We therefore proceed by calibrating a probability distribution to observed market data and use this as a lower bound for the price of $\tilde{L}$, or, equivalently the minting fee should be at least equal to the price of $\tilde{L}$:

$$\Theta_F^{(i)} \geq \mathbb{E}_\tau \left[\tilde{L} \right]. \tag{11}$$

We are now ready to calibrate the parameters. To do so, we set the level α to estimate h', and then estimate the fee $\Theta_F^{(i)}$ for the given level of α. In Appendix A we describe the BTCCHF data that we use to calibrate the fees. The normal distribution does not fit the empirical distribution of 24-hour log-returns well, as we demonstrate in the appendix. Therefore, instead of fitting another parametric distribution and integrating Equation (9), we use the bootstrap method introduced by [Efron, 1992] for estimation.

For the bootstrap, we define the following variables. Let N be the number of return observations, B the number of bootstrap replications, and let $\hat{\mathbf{r}}_b = \{\hat{r}_\tau^{(b,1)}, ..., \hat{r}_\tau^{(b,N)}\}$ be the b^{th} bootstrap replication. We use the variance of the quantiles for each bootstrap replication to estimate confidence intervals.

We can estimate h' for a given α using Equation (10), by extracting the sample quantile from each bootstrap replication:

$$\hat{h}' = \frac{1}{B} \sum_{b=0}^{B-1} Q((1+h)\exp(-\hat{\mathbf{r}}_b) - 1, 1-\alpha), \tag{12}$$

where $Q(\mathbf{x}, a)$ is the empirical quantile function for level a applied to vector $\mathbf{x}$, and we define $\exp(\cdot)$ to be an element-wise application of the natural exponentiation. Finally, we use use Equation (7) to arrive at our bootstrap point estimate for the value of L:

$$\hat{\mathbb{E}}_\tau \left[\tilde{L} | h' \right] = -\frac{1}{BN} \sum_{j=0}^{B-1} \sum_{n=1}^{N} \mathbf{1}_{\{\hat{r}_\tau^{(jN+n)} < \log \frac{1+h}{1+h'}\}} \left[(1+h')e^{\hat{r}_\tau^{(j,n)}} - (1+k) \right]. \tag{13}$$

Again, we use the variance of the elements from each bootstrap replication to estimate confidence intervals.

Table (1) shows the results of the calibration. We use $B = 5,000$ bootstrap replications and calculate confidence intervals at the 1%-level by applying the central limit theorem, that is, recycling the variances described above and normal-quantiles.[1] In Table (1), we see that if the challenger want a 95% chance of earning the challenger reward, they start the challenge at $(1+3\%)$. At this low level, the stakers still make a profit of 0.63% on average. From the first row of the table, we see that if the challenger starts their challenge at the liquidation level of $(1+0.1)$, there is about a 50% chance that the position will be liquidated at the end of the 24h-period. The earnings for the stakers decrease when moving from $h' = 0.10$ to a lower α. This is because in many cases, the stakers have no profit when starting from 0.10, because there is no liquidation, and hence, the expected profit is slightly higher when starting from a lower h'. From the last row of the table we see that if challenges are issued only after the collateral fell below the loan value by 1%, the stakers expect a loss of 2%.

A 95% probability of receiving a challenger reward seems a very good position to be in for the following reason. The challenger at this point has the option of waiting for the prices to change more. The collateral could either depreciate more (which would increase the 95% chance of getting

[1]Having a vector of bootstrap estimates $\mathbf{x}$, we report the point estimate as the mean of $\mathbf{x}$, and the error at the a-level as $z_{1-a}\sqrt{V[\mathbf{x}]/B}$ with z_{1-a} the standard normal quantile.

the reward), or appreciate, in which case the probability is decreased. So very likely at this point, the challenger would issue the challenge. At $\alpha = 0.95$, the stakers still earn on average. Therefore we conclude that for the Liquid Collateral Plugin collateralized with Bitcoin and the chosen risk parameters, no minting fee has to be charged, unless the risk-premium exceeds about 0.60%.

Table 1: Liquid Collateral Minting Fees. The column labelled α shows the probability that the challenge results in a liquidation and the challenger is paid a reward. The second column shows the challenge level h' for which the challenge ends in a liquidation with probability α for a liquidation level $h = 10\%$ and a challenge period of 24 hours. For each given h' we calculate the expected loss for the stakers (last column; negative numbers are gains). The numbers followed after $\pm$ correspond to the error of our estimates at the 1%-level. We see that if challenges are issued at a collateral value of $(1 + 0.03)$, quite below the liquidation level of 1.1, there is a 95% probability that the collateral is liquidated, while stakers will still on average earn 0.69%.

α	h'	$\hat{E}_\tau\left[\tilde{L}\vert h'\right]$
0.50	0.10 $\pm$ 0.00	-2.59% $\pm$ 0.00%
0.75	0.08 $\pm$0.00	-3.30% $\pm$ 0.00%
0.80	0.07 $\pm$0.00	-3.21% $\pm$ 0.00%
0.90	0.05 $\pm$0.00	-2.37% $\pm$ 0.00%
0.95	0.03 $\pm$0.00	-0.63% $\pm$ 0.00%
0.99	-0.01 $\pm$0.00	2.99% $\pm$ 0.00%

3.2.2 Reserves

Each mint plugin defines the required reserves of ZCHF to be held against the issued volume of ZCHF. The assessment we perform now differs from the one we made to determine the minimal fee in two main aspects, (1) we want to estimate a loss for the whole Liquid Collateral plugin not only for an individual position, and (2) we are not looking for an insurance valuation but for the capital required. The latter point implies that we do not need any assumption on risk-neutral measures but we can directly work with the observed data.

We would like to base out capital reserves on a risk-measure such as Expected Shortfall, see for example [McNeil et al., 2015]. We may want to set our risk-appetite via quantile-level of 1% so the system could survive an average loss observed in the worst 1% cases. The plugin-wide loss depends heavily on the current loan-to-value ratios in the system, which we do not dare to model without having gathered historical data.

In the worst case, the loan-to-value ratios are all at 1:(1+h), just before liquidation can occur, and a single shock could wipe out the system. We construct empirical samples of loss data by applying the 24h log-returns to the loss function given by Equation (7). Figure 2 gives an overview of the loss data for different challenger levels h' via boxplots. We see that the median level of losses are benign, even for very low challenger levels of $h' = -1\%$ (meaning the liquidation is only started when the collateral has a value of 99% of the loan notional, although the position would liquidate if the collateral is at 110% of the loan notional). However, we also see that there are high losses for all levels of h' depicted, driven by two extreme returns in the data, where the loss exceeds 20% of the loan notional. These losses are at a loan level and apply if the loan challenge starts at a loan-to-value ratio of $1 : (1 + h')$. On a plugin-wide level, we would only lose the same percentage of the outstanding loan notional, if all loans started at $1 : (1 + h')$ simultaneously. Realistically, loan-to-value ratios are more spread out and hence, the loan-level losses observed in Figure 2 are an upper bound on the plugin-wide loss.

To gain further insight into the loan-level loss distribution that serves as an upper bound for the plugin-wide loss, we estimate employ a parametric approach. expected shortfall.

To do so, we estimate the expected shortfall by fitting a Generalized Pareto Distribution.[2] Appendix B describes details of the estimation and presents sensitivities. We choose a quantile of 5% and assume a conservative challenge level $h' = 0.05$ (which corresponds to a 90% probability that the challenge results in a liquidation, see Table 1). The result,

Quantile	**ES**
5%	25%

suggests that 25% of the outstanding loan amount should be held as capital reserves. As we demonstrate in Appendix B, there is a high sensitivity of this estimate to the parameters. However, the 25% seem to be a reasonable upper bound, when comparing to the empirical extremes presented in Figure 2, and considering that this is on loan level.

Once the upper bound for the capital reserve is estimated, the next question is how far can we deviate from the upper bound. We suggest to first set the capital requirement according to the upper bound. Once data on loan-to-value ratios is available, the protocol governance can reassess.

[2] One necessary condition for the loan-level ES being an upper bound is the subbadditivity of expected shortfall, see [Artzner et al., 1999]

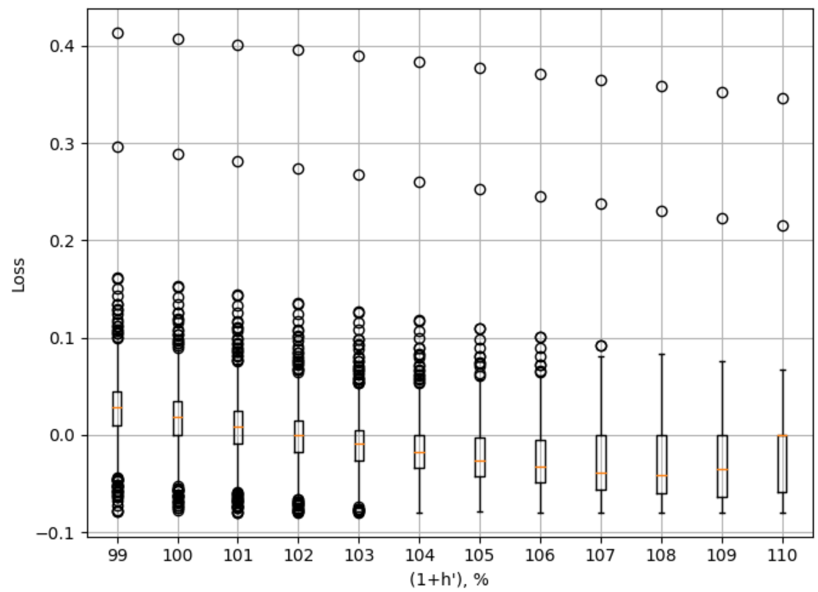

Figure 2: Empirical Loss Boxplot. This figure shows standard boxplots for different challenger levels h', given that a liquidation is triggered when the challenge results in a collateral value of below 110% of the loan. The vertical bar depicts the median, the boxes reach from the first quartile to the third quartile, the whiskers extend to the max/min or at most 1.5 times the range of the box, circles are plotted outside the span of the whiskers. For values h' close to $h = 0.10$, the loss usually ends in a gain (negative number). Losses stay on average benign but we also observe a few very high losses that exceed 20% of the loan notional.

4 System-wide Minimum Capital Requirements

In this section, we define overarching rules to make the Frankencoin system resilient. We regard the system as a bank and apply risk mitigation methods used for banks.

Figure 3 presents a balance sheet view of the Frankencoin system and to illustrate the risks, we deviate from usual accounting rules. Collateral is depicted on the asset side.[3] Like on a central bank balance sheet, the ZCHF in circulation are shown as a liability. The "book value of equity" is the difference between assets and liabilities. If the collateral of the liquid collateral plugin falls in value so that the book value of equity is zero or negative, the ZCHF in circulation are no longer backed and reserves have to be burned. This leads to a reduction of ZCHF in circulation and puts the balance sheet back to a healthier state.

We motivate the following risk mitigating measures from the Basel III banking regulation and the Dodd-Frank Act, see, e.g., [Basel Committee on Banking Supervision, 2010] and [Acharya et al., 2010]:

1. *Risk-based capital requirements*: each mint-plugin defines its own risk-based reserve requirement. E.g., for the liquid collateral plugin, we choose to estimate a reserve requirement based

[3] Per accounting rules, securities for collateralized credits are not reported as assets if they cannot be sold by the entity without default of the borrower.

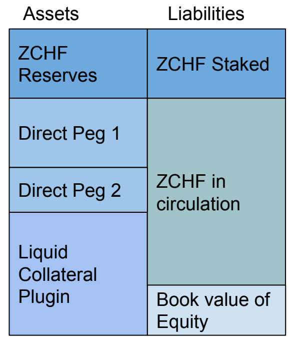

Figure 3: Balance Sheet. This diagram schematizes the balance sheet of the Frankencoin system for a specific mint plugin setup.

on expected shortfall. The fact that the system uses shared capital, but plugins define individual reserve requirements adds a diversification benefit.

2. *Leverage limits* have two main benefits. First, they are largely model-free and are therefore a safeguard against model risk (e.g., model risk due to the risk-based calibration of capital). Second, a leverage limit provides us a global (i.e., Frankencoin-system-wide) capital limit.

3. *Concentration limits*. If the collateral of Frankencoin is singularly exposed to the price of Bitcoin, the Frankencoin is doomed to collapse when the Bitcoin price falls sufficiently. We therefore aim to limit the concentration of collateral.

If the capital is below the minimal capital derived from the above measures, no more Frankencoins can be issued.

We have detailed the *risk-based capital* requirements in the previous sections. The risk-based capital requirements involve relatively complex calculations (for current blockchain capabilities), however, they are done off-chain and we are only required to store the resulting parameters in the blockchain.

A simple way to measure *concentration* is the one-firm concentration ratio, see [Curry and George, 1983], which equals the percentage of market share held by the largest firm. We apply this to our context. Let K be the number of mint plugins. Each mint plugin tracks the amount of ZCHF issued and not burned, Z_j.[4] The relative amount issued by plugin j is given by $p_j = Z_j / \sum_i^K Z_i$. We aim to prevent that the largest p_j is beyond a threshold. However, when applying this restriction on a system-wide level, we run into the problem that the system could consist of multiple different plugins with the same or very similar collateral. We therefore require that each mint plugin sets its own threshold $\theta_i^{(C)}$. If a single mint plugin reaches its threshold, $\theta_i^{(C)}$, the plugin cannot issue any more ZCHF until the concentration is reduced.

Finally, we detail the *leverage limits*. In the spirit of the creators of the leverage limits, we aim to have the least possible assumptions to define leverage limits in our system. We therefore do not want to rely on any collateral valuations or distributional assumptions. Direct Peg Plugins should not count towards the leverage ratio, because their value is stable and diversified through

[4]Direct Peg Plugins do not have the notion of a position and therefore set Z_j equal to the collateral. If collateral is liquidated, the ZCHF burnt are subtracted from Z_j.

the concentration limits, and because they provide a convenient way to issue new ZCHF that are subsequently staked. However, we want to limit the amount of ZCHF issued through Liquid Collateral Plugins relative to the ZCHF held as a reserve in the staking pool. We therefore define the leverage ratio as

$$LR = \frac{1}{S} \sum_{i \in \mathcal{C}} Z_i, \tag{14}$$

where Z_i is the amount of tokens issued by mint plugin i, $\mathcal{C}$ is the set of plugin indices that are to be included in the leverage calculation (i.e., not the Direct Peg plugins), and S is the total amount of staked ZCHF. If the leverage ratio is larger than a governance set threshold $\theta^{(L)}$, plugins other than the Direct Peg plugins can no longer mint ZCHF.

Table 2: Summary of Data used in Smart Contracts. This table lists the parameters and data that needs to be stored to implement the desired risk mitigation measures.

Element To Store	Type	Description
Z_i	Data	Each mint plugin keeps track of ZCHF issued by the plugin and not burned
Z, $Z^{\mathcal{C}}$	Data	The governance contract keeps track of the sum of Z_i and the sum over the Z_i issued by leverage constraint relevant contracts
S	Data	Amount of ZCHF staked
$\mathcal{C} = \{c_0, ..., c_{N-1}\} \in \{0,1\}^N$	Parameter	Each mint plugin defines whether they are part of the leverage ratio calculation or not
h, τ, k	Parameter	Parameters for Liquid Collateral Plugin
$\Theta^{(L)}$	Parameter	Maximal leverage ratio
$\Theta_i^{(C)}$	Parameter	Maximal relative amount of ZCHF issued by mint plugin i to limit concentration risk
$\Theta_i^{(F)}$	Parameter	Each mint plugin has a minting fee
$\Theta_i^{(R)}$	Parameter	Each mint plugin sets its own risk-based reserve requirement, relative to the issued ZCHF
$\Theta_i^{(I)}$	Parameter	Each mint plugin can define an interest rate paid to stakers (can be zero, e.g., in case of Direct Peg Plugins)

Table 2 summarizes the parameters and data that needs to be stored to implement these risk measures. Despite the comprehensive measures, there are only a few parameters that need to be stored.

5 Conclusion

TODO

References

[Acharya et al., 2010] Acharya, V. V., Cooley, T. F., Richardson, M. P., Walter, I., of Business, N. Y. U. S. S., and Scholes, M. (2010). *Regulating Wall Street: The Dodd-Frank Act and the new architecture of global finance*, volume 608. Wiley Hoboken, NJ.

[Artzner et al., 1999] Artzner, P., Delbaen, F., Eber, J.-M., and Heath, D. (1999). Coherent measures of risk. *Mathematical finance*, 9(3):203–228.

[Basel Committee on Banking Supervision, 2010] Basel Committee on Banking Supervision (2010). Basel iii: A global regulatory framework for more resilient banks and banking systems.

[Björk, 2009] Björk, T. (2009). *Arbitrage theory in continuous time.* Oxford university press.

[Breeden and Litzenberger, 1978] Breeden, D. T. and Litzenberger, R. H. (1978). Prices of state-contingent claims implicit in option prices. *Journal of business*, pages 621–651.

[Clements, 2021] Clements, R. (2021). Built to fail: The inherent fragility of algorithmic stablecoins. *Wake Forest L. Rev. Online*, 11:131.

[Curry and George, 1983] Curry, B. and George, K. D. (1983). Industrial concentration: a survey. *The Journal of Industrial Economics*, pages 203–255.

[Davison and Smith, 1990] Davison, A. C. and Smith, R. L. (1990). Models for exceedances over high thresholds. *Journal of the Royal Statistical Society: Series B (Methodological)*, 52(3):393–425.

[Efron, 1992] Efron, B. (1992). Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*, pages 569–593. Springer.

[Engelmann and Rauhmeier, 2006] Engelmann, B. and Rauhmeier, R. (2006). *The Basel II risk parameters: estimation, validation, and stress testing.* Springer Science & Business Media.

[Ghosh and Resnick, 2010] Ghosh, S. and Resnick, S. (2010). A discussion on mean excess plots. *Stochastic Processes and their Applications*, 120(8):1492–1517.

[Jarrow and Turnbull, 2000] Jarrow, R. A. and Turnbull, S. M. (2000). *Derivative securities.* South-Western Pub.

[Jorion et al., 2010] Jorion, P. et al. (2010). *Financial Risk Manager Handbook: FRM Part I/Part II*, volume 625. John Wiley & Sons.

[McNeil et al., 2015] McNeil, A. J., Frey, R., and Embrechts, P. (2015). *Quantitative risk management: concepts, techniques and tools-revised edition.* Princeton university press.

A Data

Figure 4: BTCCHF Level Data. This figure plots the level data of the BTCCHF time-series. We have 756 observations of daily candle data without gaps from 2019-12-07 to 2021-12-31.

We gather 1-hour candle data from Kraken, consisting of a timestamp, open, low, high, close, number of trades, and volume.[5] From each open and close price we calculate 24h log-returns. Our return data has the following summary statistics. Table 4 test the time-series of 24h log-returns for

Table 3: Summary Statistics for 24h log-return data.

num. observations	756
min, max	[-49.09%, 25.13%]
mean	0.26%
variance	0.0019
skewness	-2.07
kurtosis	27.31

stationarity via Augmented Dickey-Fuller test and rejects the null-hypothesis of non-stationarity.

Table 4: Stationarity Test. The ADF test suggests that the log-return data is stationary.

ADF Statistic:	-12.6
p-value:	0.000000
Critical Value for 1%:	-3.4

Figure 5 present a quantile-quantile plot against the normal distribution and a histogram. Figure 5 also analyses serial correlation of the 24h log-returns. The Autocorrelation Function shows a significant negative correlation at lag one, which however turns out to be driven from extreme returns, as we can see on plot (d).

[5]See `https://support.kraken.com/hc/en-us/articles/360047124832-Downloadable-historical-OHLCVT-Open-High-Low-Close-Volume-Trades-data`

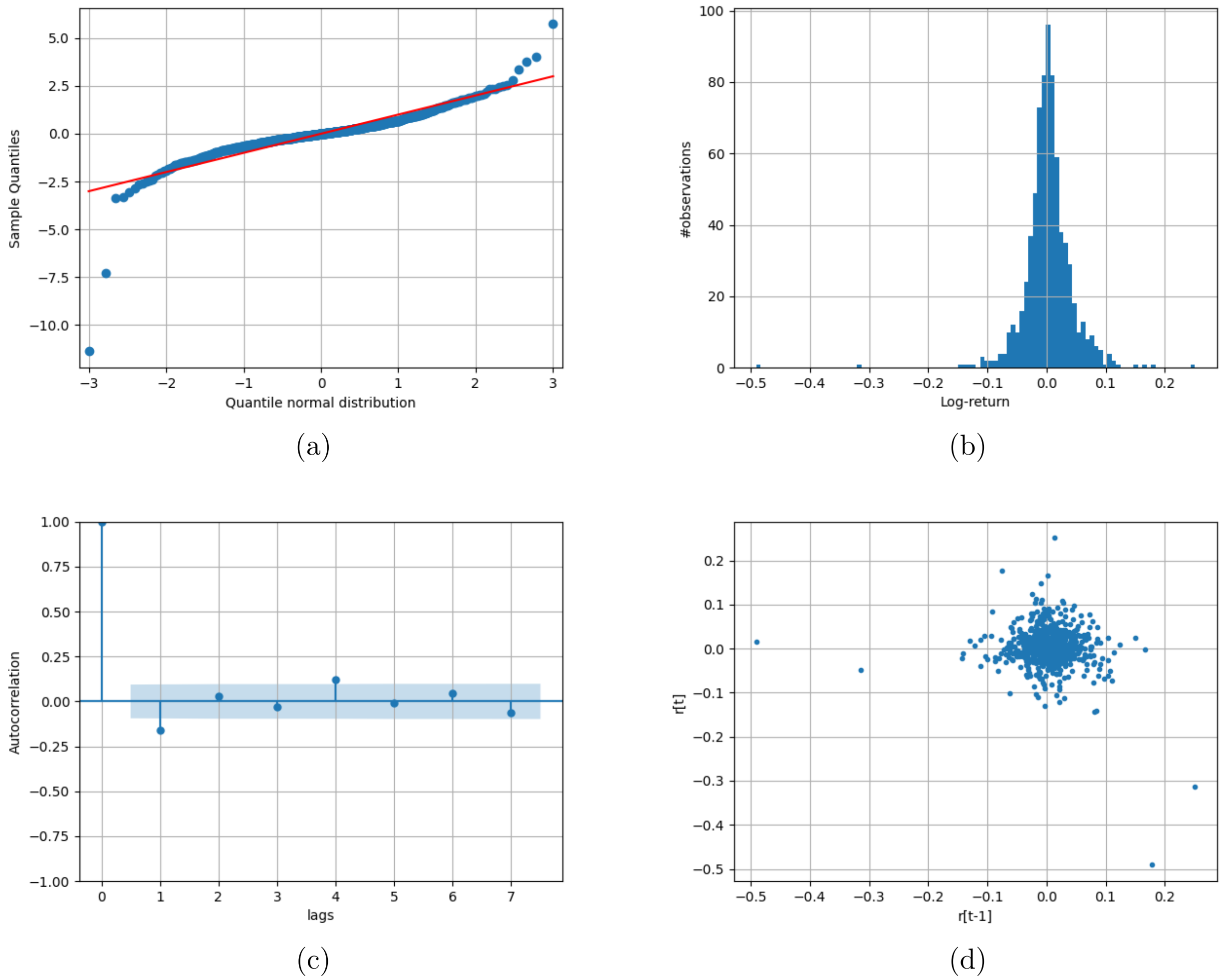

Figure 5: Return Data from Kraken. Plot (a) shows a quantile-quantile plot of the 24h log-return data against a normal distribution, (b) plots a histogram of the 24h log-returns. From (a) we see that we have more extreme returns than what a normal distribution would suggest. In the center of the distribution, the returns move less than the normal distribution would imply. Plot (c) shows the autocorrelation function and a confidence band at the 0.95-level. Chart (d) plots the returns against the previous day return. The figure indicates that the significant autocorrelation at level 1 must be driven by rare extreme returns.

B Expected Shortfall

We construct the loss data by applying the 24-h log-returns to the loss function, Equation (7). We then calibrate a Generalized Pareto Distribution. Table 5 summarizes the loss data for $h' = 0.05$.

Table 5: Summary Statistics for loss distribution data ($h' = 0.05$).

num. observations	756
min, max	[-7.94%, 37.73%]
mean	-2.25%
variance	0.001153
skewness	3.33
kurtosis	30.72

The calibration procedure requires us to choose a cutoff point u, beyond which the Generalized Pareto Distribution is fitted. We choose the cut-off level using the Mean Excess plot, see Figure 6. If the Mean Excess plot is close to linear for high values of the threshold then there is no evidence against use of a GPD model, see [Davison and Smith, 1990], or [Ghosh and Resnick, 2010]. We choose $u = 0.03$. After choosing the cutoff point u, we proceed by estimating the distribution parameters via Maximum Likelihood, see [McNeil et al., 2015]. Table 6 presents sensitivities to the Expected Shortfall estimate to the choice of cutoff points and quantiles.

Table 6: Sensitivities. This table selects cutoff points and quantiles to show the sensitivities of the Expected Shortfall (ES) estimates.

Challenge Level	**Cutoff Point**	**Quantile**	**ES**
0.05	0.03	1%	35.1%
0.05	0.02	5%	28.8%
0.05	0.03	5%	25.1%
0.05	0.04	5%	37.8%
0.05	0.03	10%	20.9%
0.05	0.04	10%	28.0%

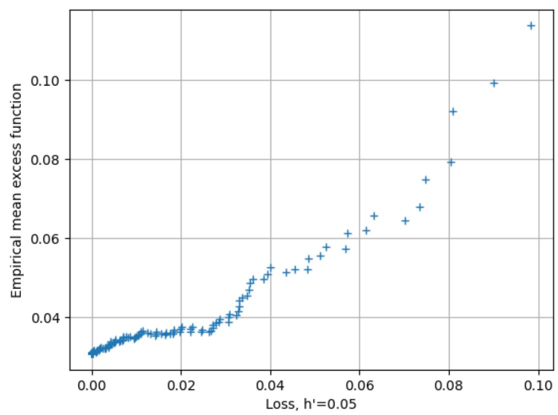

Figure 6: Historical Mean Excess Plot. This plot shows the mean observed over a threshold loss u, plotted against u. For example, on average we observe a loss of about 0.04 above u=0.03 if the loss exceeds u=0.03. Losses are relative to the loan notional and calculated for a liquidation start level of $h' = 0.05$. As often done for the mean excess plot, the last part of the data is omitted (0.5%). This chart indicates that we might set the threshold for the Generalized Pareto Distribution to $u = 0.03$.