The Malt Protocol

0xScotch
scotch@malt.money

FelipeDlB
felipe@malt.money

June 2023

Abstract

This paper introduces the Malt Protocol, a novel system designed to optimize the use of multiple collateral sources to maintain the price of its native token close to a desired target. The initial implementation aims to create a stablecoin pegged to $1, a decision made to minimize complexity during the live alpha. Unlike traditional stablecoin designs that use static collateral, Malt Protocol employs a dynamic approach, capturing arbitrage and seigniorage profit and generating real native yield paid in non-native tokens like DAI or ETH. This paper presents a new mechanism where the collateral ratio's rate of change outpaces supply's rate of change, denoted by $\frac{dC}{dt} > \frac{dS}{dt}$. Allowing supply volatility to boost collateral and provide real yield. The protocol aims to minimize capital used for price stability, thereby enhancing the protocol's total collateral ratio.

Contents

1 Introduction

Most stablecoin designs today utilize collateral in a static manner, either through passive mint/redeem flows or collateral liquidations. In both scenarios, the protocol consistently values the stablecoin at its intrinsic collateral value. For instance, if the market price of a mint/redeem stablecoin drops to $0.98, an arbitrage opportunity arises between the market price and the protocol's valuation of the token. This arbitrage allows a savvy trader to buy at the market rate of $0.98 and redeem against the protocol for $1.

This mechanism serves two key functions:

1. It applies direct buying pressure to the market, driving the price back towards its peg.

2. It reduces the stablecoin supply when demand decreases.

However, this system has a drawback: while it stabilizes the token's price, the protocol itself does not profit. Instead, all profits go to the arbitrageur. This raises the question: Can the protocol capture some of this profit while still fulfilling the two functions above? If feasible, this would create a native yield source for the stablecoin, eliminating reliance on other yield incentives like token emissions or bribes.

Consider a scenario where the protocol has 100% collateral and the market price falls to $0.98. In the standard mint/redeem flow, a trader buys for $0.98, and the protocol purchases it from the trader for $1, subsequently burning the native token from the supply. Here, the protocol spends $1 to burn 1 token. However, if we remove the trader from the equation and allow the protocol to buy the token directly from the market for $0.98, the protocol spends less capital to achieve the same supply reduction. This capital saving is retained as collateral, thereby increasing the protocol's total collateral ratio.

2 The Collateral Change inequality

This concept can be generalized by stating that the collateral ratio's rate of change must exceed the supply's rate of change. This principle applies to both supply growth and shrinkage scenarios and is mathematically represented by the following Collateral Change inequality:

$$\frac{dC}{dt} > \frac{dS}{dt} \tag{1}$$

where

C is the Total Collateral.

S is the Total Supply.

t is time.

For mint/redeem stablecoins, $\frac{dC}{dt} \equiv \frac{dS}{dt}$ by the mechanism's very definition. However, this paper proposes that it's possible to introduce mechanics that satisfy inequality 1. This introduces the potential for supply (S) volatility to increase collateral (C).

In practical terms, this means:

- When new tokens are minted, the collateral increase should exceed the token supply increase.
- When tokens are burned, the collateral used should be less than the tokens' intrinsic value.

3 The Malt Protocol

Modern Automated Market Makers (AMMs) function as self-contained supply-demand ecosystems. The current price of an asset within an AMM is determined solely by the asset distribution at that specific moment. The Malt Protocol leverages this principle at its core. For each AMM pool requiring active stabilization, a unique "Stabilizer Pod" is assigned. A Stabilizer Pod is essentially a set of contracts tasked with aligning the asset distribution within its designated AMM pool with a desired target price. The stabilization process generates seigniorage and arbitrage profit, which is then split between enhancing the protocol's collateral and generating yield for the pool's Liquidity Providers (LPs).

Each Stabilizer Pod has it's own collateral and can act entirely independently of any other Stabilizer Pod. Each Malt/X AMM pool will be collateralized by X and will pay yield to LPs in X. The amount of collateral contained by and yield paid by the Stabilizer Pod depends entirely on the market activity in the pool it is stabilizing.

The sum of the collateral across all Stabilizer Pods makes up the total global collateral backing Malt. The fact that higher liquidity pools are required to have higher collateral to maintain the protocol's rules provides a market driven mechanism to determine the distribution of assets backing the protocol. This allows each market participant to vote on what assets should collateralize Malt

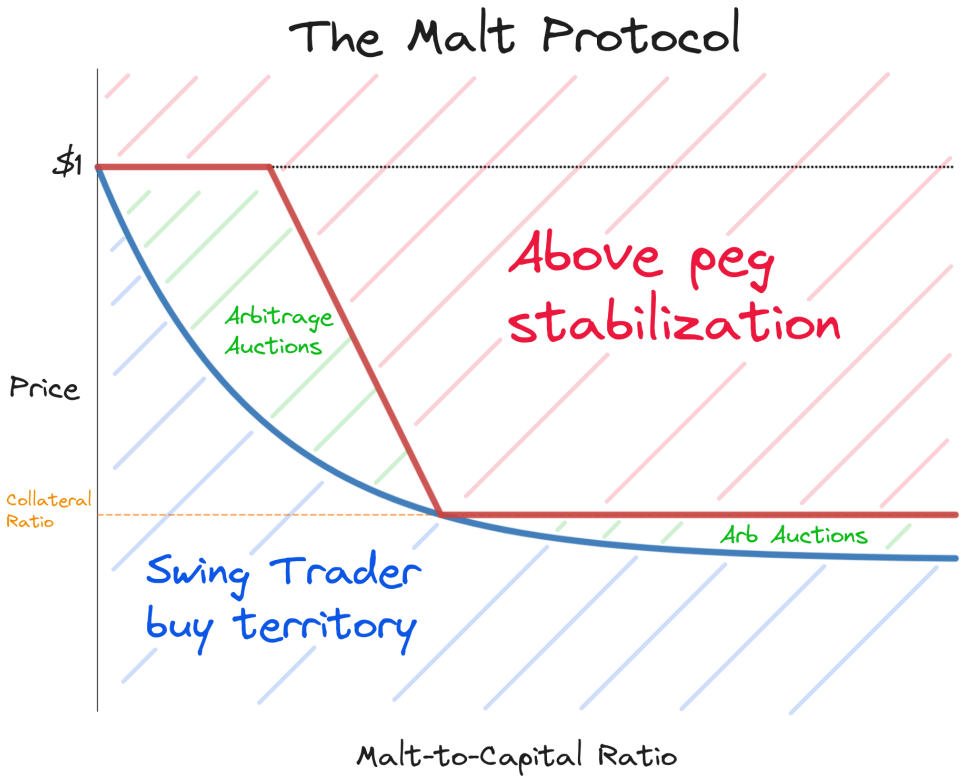

Figure 1: A graphical representation of Malt Protocol for different values of Swing Trader Malt-to-capital ratios

by simply providing liquidity to those asset's Malt pools. This is a much more direct approach to asset allocation than most other stablecoin designs around today.

The Malt Protocol implements four distinct price stabilization strategies—two for when the market price exceeds the desired level (the "peg"), and two for when it falls below. This desired level is denoted as T. Each Stabilizer Pod is responsible for executing these strategies within the specific AMM pool it manages, with the ultimate aim of adjusting the asset distribution in the pool to achieve the desired market price.

When the price is above T, the actions are:

1. The "Swing Trader" sells a portion of its Malt holdings.
2. New Malt is minted and sold into the AMM.

When the price is below T, the actions are:

1. The "Swing Trader" uses collateral to buy back Malt.
2. A public auction is conducted to raise funds for Malt repurchase, subsidized by liquidity extension (explained further in the following sections).

Each of these strategies has adjustable parameters that influence the relationship between the collateral's rate of change and the supply's rate of change, thereby ensuring the Collateral Change inequality 1 is maintained.

3.1 Above target price

Whenever the peg actions necessitate the sale of Malt, the key parameter to adjust is the market price threshold above the peg that triggers this action. Suppose the stabilization action is triggered when the price reaches $T + 2\Delta$. We'll assume the trade will execute at a clearing price of $T + \Delta$. Therefore, for each Malt sold, collateral worth $T + \Delta$ is received. Considering that T collateral is required per Malt for full backing, each new Malt adds a surplus of Δ collateral to what it introduces into circulation.

$$\frac{dC}{dt} - \frac{dS}{dt} = \Delta$$

For a Swing Trader selling Malt at $T + 2\Delta$, the situation is even more advantageous. This is because the Swing Trader has previously used some of its collateral to buy the Malt it is now selling. The entire round trip for the swing trader has no effect on the total supply, as we simply bought Malt and are now selling it back. Assume the average cost basis of the Malt being sold is $T - \phi$. Therefore, the entire swing trader round trip yields a profit of $\Delta + \phi$ per Malt sold, as this is the difference between the cost price and selling price. This means that the collateral increased by $\Delta + \phi$ without affecting the supply.

$$\frac{dC}{dt} - \frac{dS}{dt} = \Delta + \phi$$

Instead of using the static peg as the target all the time, the protocol uses another curve, T_a, the "Actual Target". This is the actual price that the stabilizer pod uses to determine which actions to take and what price to return the market to when stabilizing. Ideally, $T = T_a$. However, there are cases, which will be explained later, where this is not true—specifically when the collateral ratio $C_r < 1$ and the protocol has already made some attempts at recovering the peg. The specific definition of T_a will be provided in 3.2.3

3.2 Below target price

When the price of assets within the pool falls below the target price T_a, the preferred stabilization method is for the Swing Trader to buy Malt using its collateral. However, this purchase isn't triggered arbitrarily. It's guided by an equation that takes into account two crucial factors: the current global collateral ratio C_r, and the ratio of Malt to the total capital within the Swing Trader m_r.

Moreover, the Swing Trader's buying process is refined by two internal protocol variables. The first, P_b, sets the lowest possible price that triggers the Swing Trader to make a purchase (i.e., the curve has an asymptote at this value). The value of P_b sets a soft floor for the price of Malt. While the market price can fall below this price, the Swing Trader will trigger at any price at or below P_b. The second variable, z, designates a specific value of m_r at which the buying price equals Malt's intrinsic value, C_r. In other words, z is the exact value of m_r where $\frac{dC}{dt} \equiv \frac{dS}{dt}$, i.e., Malt behaves like a traditional mint/redeem stablecoin.

In practice, P_b is defined in relation to the current collateral ratio C_r. For example, it might be set at a specific percentage below the intrinsic value, such as 10%.

3.2.1 Swing Trader entry pricing

The price at which the Swing Trader will enter, denoted as P, is defined as:

$$P = (1 - P_b)e^{\lambda m_r} + P_b \tag{2}$$

Where λ is:

$$\lambda = \frac{\ln(C_r - P_b) - \ln(1 - P_b)}{z} \tag{3}$$

Where:

P = Swing Trader Entry Price (% of the peg price)

C_r = Implied Collateral Ratio (% of supply)

m_r = Swing Trader Malt-to-capital ratio (% of total capital)

P_b = Swing Trader Bottom Price (% of peg price)

λ = Decay Rate

z = Intrinsic Value Crossover (Value of ST Malt-to-capital ratio)

Given that P is a percentage of the target price, the actual target price is:

$$\text{Target Price} = PT$$

However, for the rest of the paper, we will assume T to be \$1, which allows P to directly represent a price in dollars. This assumption also applies to other variables in the equations that represent percentage values. By assuming $T = 1$, all percentage values convert into dollar values.

3.2.2 Explaining the equations

Equation 2 defines the Swing Trader entry price P as a weighted sum of P_b and the exponentially decayed term $e^{-\lambda m_r}$. In practice, this means the curve decays exponentially from 1 towards the value of P_b—thus, adjusting the free variable P_b allows us to alter the lowest entry price for the Swing Trader.

The value of m_r controls the decay of the exponential, as seen in figure 2 where m_r is used for the x-axis. The shape of the exponential decay is determined by the value of λ from 3. The expression that defines λ has been chosen to scale the curve such that it intersects the point (z, C_r).

This was calculated by setting up the weighted sum using the value of z for m_r and setting the whole expression equal to C_r, then rearranging to find the expression for λ:

$$(1 - P_b)e^{\lambda z} + P_b = C_r$$

This leads to:

$$e^{\lambda z} = \frac{C_r - P_b}{1 - P_b}$$

Taking the natural log and rearranging gives:

$$\lambda = \frac{\ln(C_r - P_b) - \ln(1 - P_b)}{z}$$

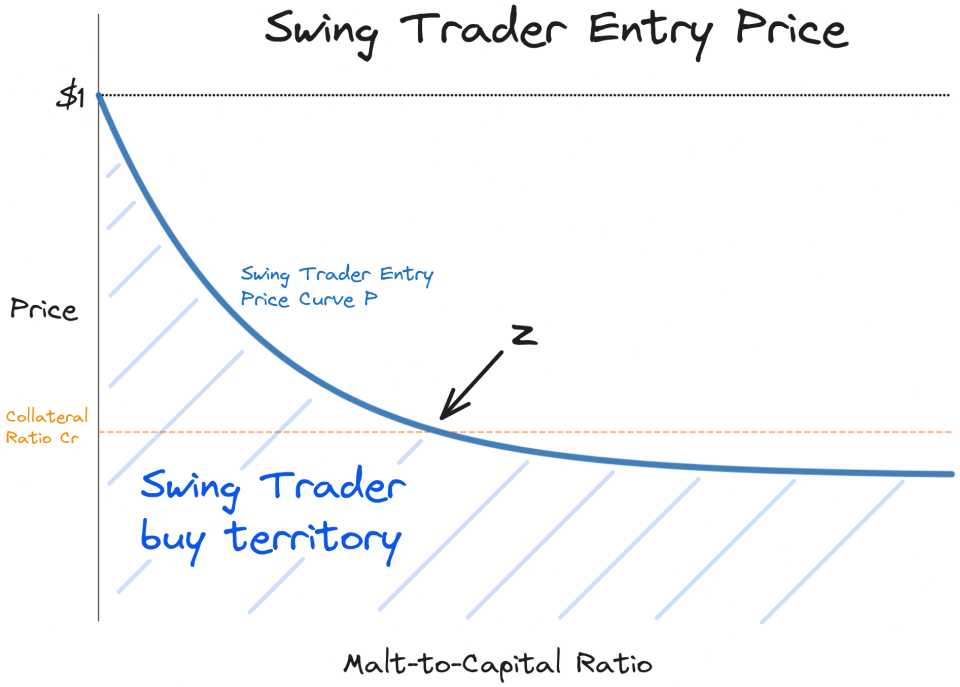

Figure 2: A plot of the Swing Trader entry price as a function of m_r and C_r.

Ultimately, z acts as a scaling factor on λ. Increasing the value of z results in a slower decay of the curve. The expression for λ ensures that the curve always passes through (z, C_r), so increasing z forcibly moves the intersection point along the x-axis, flattening the curve. In real-world terms, this value of z corresponds to a particular value of m_r where $P = C_r$. For a fixed value of z, varying all other parameters will keep that value of $m_r = z$ fixed in place at the value of C_r.

3.2.3 Bringing back T_a

The curve alone is insufficient to ensure that the inequality 1 holds. This is because equation 2 only controls the entry price. However, it's the execution price of the trade, not the entry price, that determines the relationship between capital spent and supply burned. The Swing Trader's default behavior would be to fully stabilize the price back to the peg target price T. This often results in the trade's execution price being above the intrinsic value, despite the entry price being below it. To control the execution price, we can reintroduce the previously mentioned curve T_a. This curve determines the target price that the above and below stabilization strategies will aim for. T is the absolute desired peg, while T_a is a more conservative target that also considers the current health of the protocol.

$$\rho = \frac{C_r - 1}{z - d}$$

$$v = 1 + \rho m_r - \rho d \tag{4}$$

$$T_a = \begin{cases} 1 & \text{if } m_r < d \\ v & \text{if } d \leq m_r < z \\ \min(1, C_r) & \text{otherwise} \end{cases} \tag{5}$$

$$\text{where } 0 \geq d < z$$

where

T_a = Actual Target

d = Price Clamp where $0 \leq d < z$

ρ = Gradient between d and z

v = Equation of segment between d and z

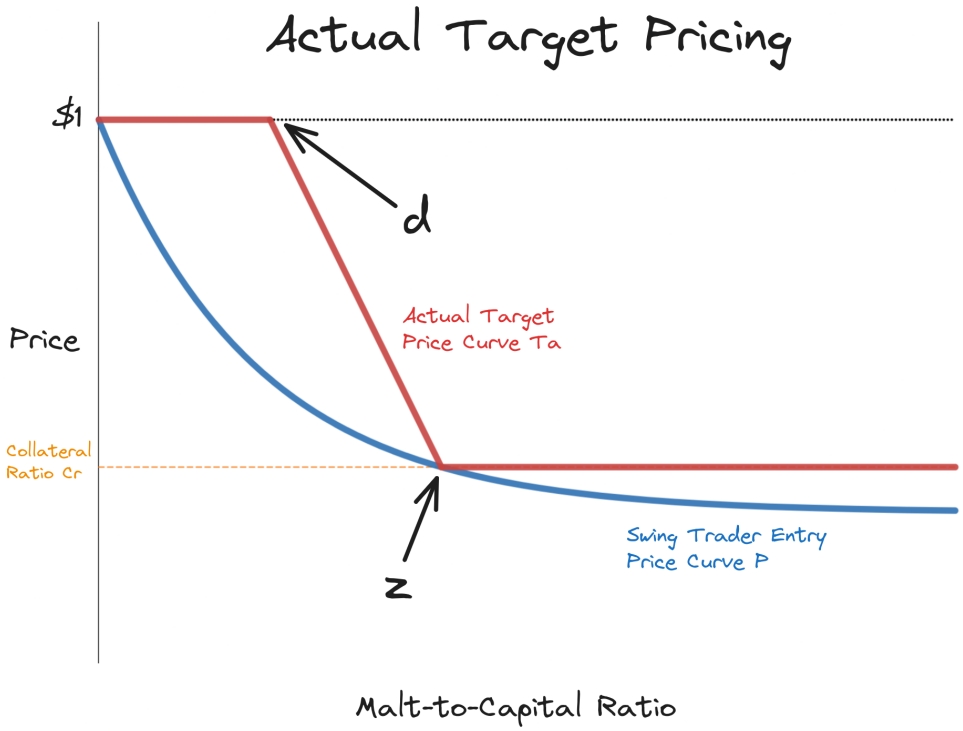

Figure 3: A graph depicting the actual target price T_a

Figure 3 shows that the equation for T_a is clamped to 1 when $m_r < d$ (i.e., $T = T_a$ when $m_r < d$) and clamped to C_r when $m_r \geq z$. Between d and z is a simple linear plot between 1 and C_r. The interplay between the z and d parameters becomes a useful dial to tune how the protocol chooses to balance between quickly returning to the peg vs. being as conservative as possible and

trying to maximize the delta in the Collateral Change inequality 1. The larger the values of z and d, the more the protocol is prioritizing returning the price to the peg, even at the expense of temporarily breaking the Collateral Change inequality. Both z and d can be thought of as the free variables that the protocol can set to determine its general aggressiveness. The value of z provides a coarse control on how the protocol behaves, while d offers a fine-tune control.

It's worth noting that $T_a \equiv T$ when $C_r \geq 1$ as all conditions collapse to 1. This is when the Malt Protocol resembles the characteristics of traditional stablecoin designs. The existence of T_a for cases where $C_r < 1$ is a logical continuation of this already established stablecoin mechanism that provides more control for a wider variety of situations.

3.2.4 Characteristics

Delving further into the characteristics of z and d, we find that z is an inflection point where the protocol switches from aggressively recovering the peg to conservatively growing the collateral. By definition, z is the point at which $\frac{dC}{dt} \equiv \frac{dS}{dt}$. It is the point where Malt behaves like a mint/redeem stablecoin. When $m_r < z$, Malt acts more aggressively, willingly paying more than Malt's intrinsic value to quickly recover the peg. However, as m_r increases—indicating that it has already bought back an increasing amount of Malt—the protocol transitions towards a more favorable construction of the inequality 1. z is the free variable that controls where this transition happens. A smaller z is more conservative, while a larger z is more aggressive.

The other value to consider is d. This allows the protocol to fine-tune its behavior for values of m_r between 0 and z, tuning the behavior in the "aggressive" part of the T_a curve 5. The value of d can range from $0 \leq d < z$. What d actually does is clamp all values $m_r < d$ to 1. This makes the protocol more aggressive for values of $m_r < d$. Therefore, for higher values of d, the protocol will behave increasingly more aggressively. The overall aggression is still set by z. d just acts as a fine-tuning mechanism within that range.

3.2.5 Balancing aggression with being conservative

As discussed above, any trade that occurs when $m_r < z$ willingly breaks the Collateral Change inequality, as it willingly pays higher than the token's intrinsic value. But what does the protocol gain in return for these concessions? Firstly, it recovers the peg quickly. Perhaps more importantly, the protocol is making a bet that the peg will be recovered and it will get an opportunity to sell the tokens back above the peg. If that bet pays off, then the protocol will have made a substantial profit on the swing trade with no impact to the supply. Therefore, it grows the collateral and provides yield for LPs.

Of course, the protocol can't keep making those concessions and will eventually have to adopt a more cautious approach. That's where z and d come into play. They control how the target price curve T_a decays towards the point where the protocol guarantees the Collateral Change inequality 1 holds, i.e., $\frac{dC}{dt} \geq \frac{dS}{dt}$.

An additional point to remember is that d also influences how abruptly the value of T_a transitions from 1 to C_r. The closer the value of d is to z, the steeper

the linear section of the curve (the curve is only unclamped between d and z).

When the market price is below the Swing Trader entry price P from equation 2, collateral will be used to buy Malt. The mathematics of how this process works can be tuned to satisfy the Collateral Change inequality 1 and thus ensure a system that tends towards increasing collateral. The behavior when the price is below T but above P will be discussed in the following section on the Arbitrage Auctions.

3.3 Arbitrage Auctions

There may be instances where the market price of Malt is below the target price T_a but above the Swing Trader Entry price P. In such cases, a mechanism to push the price back towards the desired peg (and reduce supply in the process) is still desirable. However, the Swing Trader won't be able to have any effect at this point as the rules of entry are not met. This is when the public Dutch Auction comes into play.

- Traders can purchase "Arbitrage Tokens", each of which has a claim on \$1 and will be automatically pro-rata redeemed as the protocol revenue accrues.
- Each auction starts at the Time-Weighted Average Price (TWAP) of the associated AMM pool and decreases linearly towards a predefined lower price over the course of 10 minutes (more on the pricing mechanism in 3.3.3).
- Each auction aims to raise a certain amount of capital.
- The auction ends when that amount is raised or the 10 minutes elapses.
- All participants in the auction receive the same clearing price P_{clear}.

There's also an early exit mechanism to improve the risk profile of the Arbitrage Tokens. The Arbitrage Auction is a way of raising the capital required to return the market back to the target price T_a. Using a Dutch auction allows for faster market-driven pricing of the risk associated with tokens.

3.3.1 A Layer 2 for price discovery

When the token deviates from the peg, speculators may wait for the market price to fall to a level where they deem the risk-to-reward ratio on betting on the price returning to a price closer to the desired peg is enticing. However, the speed of price discovery on that risk is very slow, and the speculators will all have different goals and strategies that don't align with the protocol's desired outcomes. To address this, a separate channel - the Dutch auction - is used to price the risk on that trade. This process moves much faster and provides a market-driven process to price the premium above the market price of Malt for taking on the risk of speculating on price returning to the peg.

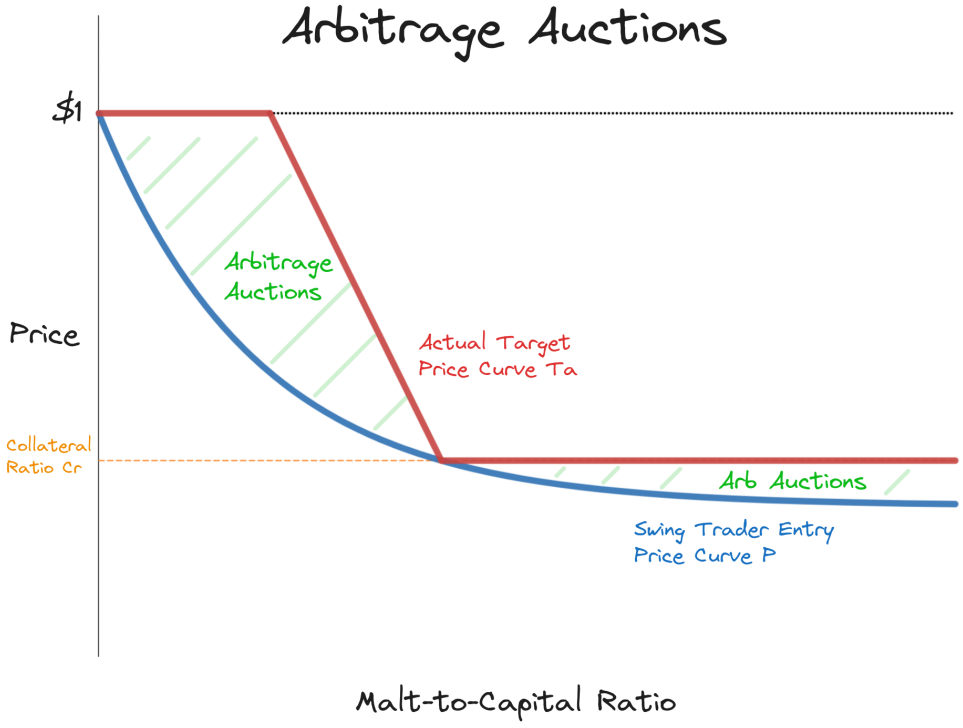

Figure 4: A graph showing the areas where Arbitrage Auctions are active

An additional benefit of conducting this outside of the AMM pool in a separate Dutch auction is that the profit-taking can also be controlled via whatever mechanics the protocol uses on the redemption of the Arbitrage Tokens.

The Dutch auction can be thought of as a "layer 2" for price discovery on the risk of speculating on the peg being recovered. Instead of relying on the AMM for price discovery, the protocol provides a faster-moving mechanism in the Dutch auction. Once the auction is fully subscribed, every user receives the same final clearing price P_{clear}. Users pay P_{clear} per Arbitrage Token, which is worth \$1 in the future when the token becomes redeemable. The profit (in dollars) the user makes per token can be expressed as

$$\nu = 1 - P_{\text{clear}}$$

Now that the risk has been determined in the "layer 2 AMM" via the Dutch auction, we can now settle the trade on the AMM by using the capital raised during the auction to buy and burn Malt from the AMM. This trade on the AMM will settle at P_{settle}, which is the actual price received from the AMM. In general, $P_{\text{settle}} > P_{\text{clear}}$, meaning the price that traders paid per Arbitrage Token is lower than the price the protocol paid for the Malt. The difference here represents the risk premium on the trade.

This mechanism provides a faster and more direct process of price discovery. However, on its own, this mechanism poses a problem related to the net change in supply throughout the lifecycle of the Arbitrage Token. Redeeming 1 Arbitrage Token will require a maximum supply increase of 1 Malt (redemption

automatically happens when Malt is trading above \$1). Unfortunately, without some additional mechanics in play, the creation of an Arbitrage Token will burn less than 1 Malt. This is due to the fact that one Arbitrage Token is purchased for P_{clear} dollars and then that P_{clear} dollars is used to buy (and then burn) Malt on the open market for P_{settle}. However, it has already been established that $P_{\text{settle}} > P_{\text{clear}}$. Because P_{settle} is larger, less Malt is burned than Arbitrage Tokens created.

3.3.2 The Malt Burned inequality

This brings us to the crux of the issue. The root cause of the depegging was an excess of supply over demand. However, the full lifecycle of the peg recovery mechanism necessitates an increase in supply. The protocol can recover the peg, but it ends up with a larger supply than it started with, exacerbating the supply/demand imbalance that caused the depegging in the first place. This creates a runaway positive feedback loop that could eventually lead to an unrecoverable depeg event, a trap that many stablecoin designs have already fallen into.

So, how do we solve this? We need to introduce another inequality that the auction must satisfy:

$$B \geq A_{\text{arb}} \tag{6}$$

where

B = Amount Malt Burned

A_{arb} = Amount Arbitrage Tokens Issued

The only way to ensure the total lifecycle of an Arbitrage Token doesn't increase the supply is to introduce another mechanic that burns more Malt each time an Arbitrage Token is created. How can the protocol just burn more supply? The key lies in headroom left on the Collateral Change inequality 1. As stated so far, the dutch auction mechanism shrinks the supply by B while leaving the collateral unchanged. The protocol can still use some collateral while still satisfying the Collateral Change inequality $\frac{dC}{dt} > \frac{dS}{dt}$. The additional capital used here to satisfy this new Malt burned inequality will be referred to as the "extension". The maximum extension possible is C_r, which will be explained below.

3.3.3 Auction pricing

Each auction initiates at the current Time-Weighted Average Price (TWAP) of the Automated Market Maker (AMM) pool that triggered the auction. The final auction price must still satisfy both inequalities (1 and 6), which places certain constraints on the pricing. These constraints can be used to compute the minimum allowable auction price.

The Malt Burned inequality 6 stipulates that the number of Malt burned must be at least equal to the number of Arbitrage Tokens created. Assuming the worst-case scenario where it costs \$1 to burn 1 Malt, at least \$1 must be spent burning Malt for every Arbitrage Token created. The auction's clearing price determines the trader's contribution, with the remainder coming from the protocol.

$$P_{\text{clear}} + c_p = 1 \tag{7}$$

where

c_p = Protocol contribution

In situations where the protocol needs to use collateral to adjust the stablecoin's supply, the Collateral Change inequality 1 comes into play. The protocol can spend up to $C_r D$ dollars to reduce the supply to D, meaning it can spend up to C_r dollars to burn one Malt.

This sets C_r as the upper limit for the protocol's contribution per Arbitrage Token c_p. Therefore, only auction clearing prices of $P_{\text{clear}} > (1 - C_r)$ will satisfy the Malt Burned inequality 6. Consequently, the lowest price an auction can permit is:

$$P_{\text{low}} = 1 - C_r \tag{8}$$

For any auction that is fully subscribed before its conclusion, the final clearing price $P_{\text{clear}} > P_{\text{low}}$. This implies that the protocol contribution $c_p < C_r$. Hence, the protocol can now ensure at least one Malt is burned per Arbitrage Token and no more than C_r dollars of collateral is expended to burn a single Malt. As both inequalities hold, the protocol can increase its collateral and maintain the stablecoin supply's economic stability.

In reality, one Malt does not cost the protocol \$1 during an auction. Instead, it will settle at some $P_{\text{settle}} < 1$.

$$P_{\text{clear}} + c_p = P_{\text{settle}}$$

This implies that the true minimum price for the auction is

$$P_{\text{low}} = P_{\text{settle}} - C_r$$

This provides some leeway on the minimum price in edge cases where the auction start price is below $1 - C_r$. In such edge cases, the protocol can assume P_{settle} to be a short TWAP and calculate P_{low} accordingly. In all other cases, the protocol will take the conservative approach and assume $P_{\text{settle}} = 1$.

To summarize:

1. Each auction commences at the TWAP of the AMM pool.
2. The auction duration is 10 minutes.
3. The price decreases linearly from the start price to P_{low}.

3.3.4 Burn supply or increase collateral

Whenever the auction concludes at a price higher than its minimum:

$$P_{\text{clear}} > P_{\text{low}}$$

The protocol can opt to either (or a combination of):

1. Burn the minimum value (which is less than C_r) to satisfy the Malt Burned inequality 6. This implies $c_p = 1 - p_{\text{clear}}$, ensuring $c_p < C_r$.

2. Burn more Malt than the minimum such that $p_{\text{clear}} + c_p > 1$. This leads to more than one Malt being burned per Arbitrage Token created.

The choice between these options hinges on the Collateral Change inequality 1. This is because the protocol must decide to expend collateral to decrease the supply. In this scenario, no additional external funds are available to assist with the Collateral Change inequality. Consequently, the maximum the protocol can spend to burn a single Malt is C_r dollars.

The excess capital will be used to burn additional Malt if:

$$p_{\text{settle}} <= C_r$$

Otherwise, the excess capital will be retained as collateral.

3.3.5 Liquidity Extension

In practice, to maintain a conservative approach, the full value of C_r is not utilized. Instead, a portion of the protocol collateral is stored in a contract named *LiquidityExtension*, which maintains its own collateral ratio C_{ext}, strictly ensuring $C_{\text{ext}} < C_r$. All the above calculations for the Auction pricing make use of C_{ext} in place of C_r. This approach results in much less than C_r in extension capital being used in the auction, leading to a more favorable Collateral Change inequality and, consequently, a better improvement in C_r through the auction process.

The sole purpose of the Liquidity Extension is to support the Auction in a manner that upholds the Malt Burned inequality 6. The capital in the Liquidity Extension is segregated from the rest of the collateral in the Swing Trader. Each Stabilizer Pod is allocated its own Liquidity Extension, and that Liquidity Extension contract maintains its own C_{ext}. This arrangement allows for individual fine-tuning of each pool to optimize its performance.

3.3.6 Profit and risk

As discussed thus far, the trade on an Arbitrage Auction appears straightforward. A trader purchases tokens for a $price < \$1$, which will later be redeemed for \$1. For instance, buy 1 Arbitrage Token for \$0.60, wait for it to redeem for \$1, and walk away with the profit. This round trip offers an efficient extraction of the value in the swing back to peg, plus an associated premium for the risk taken, with the protocol absorbing the slippage.

However, the situation is more complex. There are additional risks involved. The token may not recover peg quickly. The protocol may not generate revenue fast enough to cover the tokens in a short time frame. Given these risks, the Arbitrage Auction resembles an infinite timeframe binary option on future protocol revenue, which may not appeal to all traders in terms of risk profile.

Another factor to consider is the soft floor to the price of Malt, which occurs when $m_r > z$. At this point, the protocol will never purchase Malt for more than its intrinsic value, and when it does make a purchase, it will return the market price back up to its intrinsic value. This is a process the protocol could perform indefinitely—it could buy back the entire supply. While the market price can fall below its intrinsic value, it won't stay there indefinitely as the protocol can continually restore the price.

The soft floor provides a significant level of downside protection to those that purchased Arbitrage Tokens. Due to the early exit mechanic that will be discussed in the next section the soft floor provides a guideline for the maximum loss on the Arbitrage Tokens. The maximum loss per token is going to be around:

$$\text{Maximum Loss} = \text{AMM Price}^* - \text{Soft Floor}$$

* AMM price at the time of the Arbitrage Token purchase

As will be explained further in the next section, the early exit mechanism allows the user an "escape hatch" on the Arbitrage Tokens that allows them to settle the underlying Malt trade. This is why it is the AMM price not the Arbitrage Token price that determines the max downside. If a user purchases Arbitrage Tokens when the AMM price is \$0.90 and $C_r = 0.8$ then the max downside on the Arbitrage Tokens is around \$0.10. The figure is approximate because \$0.80 isn't a hard floor, so it is possible for the trade to temporarily be losing more. But with patience the market will return to the floor again.

Furthermore, the Swing Trader is still patiently waiting for the market price to continue to fall. Should the price fall beneath the Swing Trader entry, the market price will be returned to T_a. Given that auctions trigger at prices $< T_a$, it is likely that the position is profitable after the Swing Trader intervenes.

With a soft floor providing a well-defined downside, the Swing Trader acting to remove additional risk, and the early exit strategy to be discussed later, the protocol aims to clearly define the risk and reward of the Arbitrage Tokens. This allows traders and other protocol builders to develop strategies around it.

3.3.7 Early Exit

Internally, the protocol purchases Malt on the AMM when a trader buys Arbitrage Tokens. The early exit mechanism on the Arbitrage Tokens allows a trader to compel the protocol to settle that trade at the current AMM price. The protocol will execute the trade on the AMM, and if this underlying trade made any profit, the trader is given a certain percentage of it. If the trade is losing, the trader is given all the tokens received from the exit swap.

The percentage of the profit given for the early exit is based on the age of the Arbitrage Tokens. The older the tokens are, the larger the share of the profit they will receive. There are two internal parameters to modify this behavior:

1. The maximum profit payout percentage M. This sets the upper limit for the percentage of the profit that will be paid out for an early exit.
2. The "cool-off period" l, which is the time during which the actual percentage of the profit paid out linearly increases from 0% up to the maximum payout percentage.

Define the profit from the trade p as:

$$p = R - A$$

$$x = \min\left(\frac{t}{l}, 1\right)$$

$$\text{Total Returned For Early Exit} = \begin{cases} R & \text{if } p < 0 \\ A + pMx & \text{otherwise} \end{cases} \tag{9}$$

where

P = Swing Trader Entry Price (% of the peg price)

p = Profit from trade

R = Total value returned from exit swap

A = Purchase value of the Arb Tokens

x = Progression through cooloff period, capped at 100%

t = Time since end of auction

l = Cooloff period

M = Max profit payout percentage

pMx = Profit Payout

For instance:

1. A trader purchases 1 Arbitrage Token for 60 cents.
2. That 60 cents is then used to purchase Malt from the AMM at a price of 80 cents per Malt. Receiving 0.75 Malt
    - The difference between the 60 cents paid by the trader and the 80 cents paid by the protocol represents the premium being offered by the protocol for the risk of the trade. If the trader had bought directly on the AMM they would have received the same 80 cents price.
3. The trader has a maximum upside of 66.66% return. This is because they could receive $1 for the 60 cents they put in.
4. Some time later the market price is at 90 cents and the trader decides to early exit.
5. The 0.75 Malt purchased on their behalf is now sold at the 90 cents market price. The protocol receives 67.5 cents in return.
6. That represents a 7.5 cents profit above the 60 cents originally put in.
7. The trader is paid their 60 cents back plus (for example) 50% of the 7.5 cents for a total return of 63.75 cents.
8. The trader made a 6.25% return.

Early exits do not influence Malt's supply, as the tokens are merely bought and sold. Any residual profit, not allocated to the early exiting trader, is held as collateral by the protocol. If the trade is not profitable, the trader bears the loss.

In practice, the Liquidity Extension burned additional Malt when creating the Arbitrage Tokens so the full lifecycle that lead to the early exit caused a net reduction to supply. This process may or may not have increased the Collateral Ratio too depending on if the underlying trade was profitable.

3.3.8 Risk ramifications

The ability to exit the Arbitrage Tokens early makes their risk profile resemble that of executing the swing trade on the AMM independently. This is because the protocol literally performs that trade internally, and the early exit simply provides a way for the trader to settle it prematurely. In return for the premium that the tokens pay upon redemption, the trader must accept receiving only a percentage of the profit on the trade for an early exit.

Consequently, the Arbitrage Tokens no longer resemble long-dated binary options. Instead, they appear more akin to an appealing debt product with a structured payoff.

4 Protocol profit distribution

Any profit generated through the protocol's actions is channeled through a profit distribution pipeline. The protocol has parameters to control the distribution of the profit and its proportions. The distribution buckets include:

1. Swing Trader
2. Liquidity Extension
3. Paying outstanding Arbitrage Tokens
4. Liquidity Provider (LP) Yield
5. DAO Treasury (when established)
6. Core Team Funding

The Swing Trader and Liquidity Extension are both forms of internal collateral, and any capital in these buckets is considered part of C_r. The Liquidity Extension has its own desired ratio of capital to the size of the AMM pool it's attached to C_{ext}. Once this ratio is reached, no more capital is added to the Liquidity Extension, and any capital earmarked for it will be redistributed elsewhere.

If there are any outstanding Arbitrage Tokens, some of the profit can be used to pay those off. There is a percentage cap on the amount of profit used to pay Arbitrage Tokens. Each auction has all of its tokens redeemed automatically pro-rata each time some capital is allocated to it. Only once an auction is fully paid off will the protocol start paying off the next auction. This ensures the oldest tokens are paid off first.

The LP yield bucket is a complex topic that perhaps warrants its own paper. For the purposes of this paper, it can be thought of as a black box. However, this black box has the ability to use some of its capital to backstop collateral, a concept the protocol calls "Implied Collateral". The exact mechanics of how this works are governed by a yield throttling control system.

The most conservative configuration (from the protocol's perspective) for these parameters is to direct all profit towards the Swing Trader and Liquidity Extension (until it's full). However, LP Yield is required to incentivize users to provide liquidity. Similarly, the treasuries need capital to continue work on the protocol and maintain the ecosystem. This is why the profit distribution contract allows time-locked configuration of the allocation to each of these destinations.

Depending on the protocol's real-world performance, these parameters can be updated to ensure a configuration that maximizes collateral growth while maintaining a good yield for LPs and sufficient runway for the ecosystem.

5 Bringing it all together

With the core mechanics of the entire Malt Protocol now laid out, it's time to synthesize how all the parts work in harmony to create a stablecoin that tends towards increasing its own collateralization.

The two core curves to keep in mind are 5 and 2 which define T_a and P respectively. The value of T_a controls the market price that the protocol's actions will aim to return to. When Malt is fully collateralized ($C_r \geq 1$) or when the Swing Trader isn't holding a significant amount of Malt, T_a will simply be the desired peg price $T_a = T$.

Under these conditions Malt functions with similar characteristics to a regular mint/redeem stablecoin. However, the Malt Protocol unlocks new behaviours that are a superset of traditional stablecoins by removing static value assumptions for the collateral.The system has a mode of operation where it will transition T_a towards the intrinsic value of the collateral C_r. This is dictated by the movement of the Malt-to-capital ratio in the Swing Trader m_r.

P is the curve that defines the entry price for the Swing Trader. Again, in the case where $C_r \geq 1$, the curve $P = T$. The interplay between P, T_a, C_r, z, and d allows the protocol to behave dynamically according to market conditions while having free parameters to tune its behavior.

When $m_r > z$, the value of $T_a = C_r$. This guarantees any purchase by the Swing Trader will satisfy the Collateral Change inequality $1 — \frac{dC}{dt} > \frac{dS}{dt}$. The above target mechanism will also play to T_a and as discussed previously, this will also satisfy the Collateral Change inequality. Meaning every action when $m_r > z$ satisfies the inequality (including auctions). Every action that satisfies the inequality improves the collateral ratio C_r slightly. With every action improving collateral slightly, the floor for the next action will be slightly higher. Over time, this will tend the collateral towards 100%.

An interesting consequence of this system is that it is theoretically possible to recover the desired peg T if the only market action that occurs is selling Malt. This is because every time Malt gets sold below intrinsic value, the delta below intrinsic value is an opportunity for the protocol to arbitrage it back to its intrinsic value. The profit from this arbitrage goes towards increasing the internal collateral.

In practice, it won't be just selling happening, but because every action the protocol takes in this situation satisfies the Collateral Change inequality 1, the tendency towards fully collateralizing itself is still in play. Any time the market price deviates from the intrinsic value of the underlying collateral, the protocol has an opportunity to arbitrage the difference and improve its collateral.

When $m_r < z$, the protocol is willingly not capturing some of this deviation away from C_r in exchange for trying to maintain the desired peg T. But as the value of m_r increases, the protocol uses that as a heuristic to indicate that the health of the protocol is declining and will transition towards a more conservative and sustainable strategy that will tend towards full collateral over time.

6 Future considerations

So far, this paper has assumed a single Automated Market Maker (AMM) pool being stabilized. In reality, there would be multiple pools, each with their own Stabilizer Pod. Each pod operates independently, focusing entirely on stabilizing the single AMM pool it oversees. However, in practice, there will also be arbitrage between these pools.

The protocol aims to evolve such that some of this intra-pool arbitrage can also be captured, ideally in a risk-free manner. This would further reduce the capital requirement to maintain price stability and increase the potential for profit.

This intra-pool arbitrage can be viewed as a communication mechanism between the stabilized pools within the protocol. Any global shifts in supply/demand will propagate through all of the pools via arbitrage. Then, each pool can stabilize its respective AMM. If some pools are less healthy than others and can't fully stabilize themselves, the intra-pool arbitrage will allow other healthier pools to compensate.

7 Conclusion

The Malt Protocol seeks to innovate in the field of stablecoin design by introducing a dynamic collateral system that profits from maintaining the stablecoin's peg to its target price. Unlike traditional stablecoin protocols that allow arbitrageurs to profit from price discrepancies, the Malt Protocol proposes a mechanism where the protocol itself can capture some of this profit. This profit can then be directed towards productive ends, such as increasing protocol collateral and providing yield for Liquidity Providers (LPs).

This is achieved by ensuring that the rate of change of the collateral ratio is larger than the rate of change of the supply. In other words, when new tokens are minted, the collateral should increase more than the token supply, and when tokens are burned, the collateral used should be less than the intrinsic value of the tokens burned. This approach not only helps to maintain the stablecoin's price stability but also increases the total collateral ratio of the protocol, providing a native source of yield.

This novel approach challenges the traditional mechanisms of mint/redeem stablecoins, and it opens up the possibility for the volatility of the token supply to increase the collateral, thereby benefiting the protocol itself rather than external arbitrageurs. Malt aims to leverage a proportion of total arbitrage available on its own native token to both increase collateral and provide a native source of yield for LPs.