AAU
ÅmpleForthGold
Synthetic Gold forged using Ampleforth technology.

AAU Rebase Functionality

Design Notes.

1. Simplified design

The calculation and management of the rebase functionality is (mostly) contained in the `Orchestrator.sol` contract. The `Orchestrator.sol` contract is available on our github, or by searching for it on etherscan.io.

We have simplified design (from the original Ampleforth project) to lower gas fees. In some places we have removed things that were "nice to have" because of the cost of gas. Specifically we have lowered the number of events and hard coded things that we know are going to be constant (such as not looking up a uniswap pair, we just pass the pair pointer into the contract). This was done to save gas and lower the execution cost of the contract.

1.1 Upgradeable contracts

All contracts used by the AmpleForthGold project can currently be upgraded (replaced) by newer contracts. We will continue to improve the design and fix any issues as time goes by. Ultimately we would like to make the contracts immutable, however at this stage of the project there are too many possibilities of issues for the team to do that.

2. Oracles

The original Ampleforth (AMPL) project uses chainlink (LINK) oracles. We have moved away from that design and are using uniswap pairs to derive the price of gold. Specifically we use PAXG-WETH uniswap pair to derive the price of gold. This has been done because:

1. It is a much simpler solution then using chainlink.
2. It is a much cheaper solution then using chainlink.
3. Paxus (who own PAXG) are a reasonably sized company that operate in the US and are regulated.
4. The only way you can substantially move a uniswap pair is with real money, in many ways it is safer then using an off-chain oracle such as those supplied by chainlink (LINK).

The main issue we currently face with using the PAXG-WETH uniswap pair as a price oracle is that it floats around a bit due to lower liquidity then larger gold markets, however it usually stays within 5% of the real world gold price – arbitrageurs keep it within that range.

Relying on price Oracles (either on chain or off chain) will never be perfect. Some oracles may go bad, others may become available. While at present we will use liquidity pools on uniswap to provide the oracles for pricing, at some point in the future those oracles may go bad and need to be replaced. We think that oracle failure in the short term is unlikely, but not impossible. In the long term it may be likely to see oracle failure. Due to this the AmpleForthGold team shall have an 'off switch' in the code to disable and override rebase operations and replace oracles as needed. At some point it may be needed, but we hope it is not needed.

3. Rebase delta

The rebase delta is the change to the total amount of tokens available (the total supply) when a rebase occurs. It can be positive or negative. We have a number of rules on the magnitude of a rebase. Specifically:

1. If the AAU price is in the +-5% range of the PAXG gold price then do not rebase at all;
2. else if the AAU price is within +-10% range of the PAXG gold price then only rebase by 1% of the price;
3. otherwise the change shall be half the delta required to exactly match the two prices. i.e. if the price difference is -28% then the change will be -14%.

The `Orchestrator.sol` contract has a public function to provide the current calculated rebase delta. That function provides the delta used when a rebase occurs in the contract. It is also the source of our ‘delta’ calculation on our website and it can also be read using blockchain explorers such as etherscan.io.

4. Rebase timing

4.1 Randomness on the blockchain

With respect to random rebase timing, there is currently no recognised way of generating a random number/event in a Ethereum contract that cannot be seen/used by the miner. This is because a miner can see the block they are generating and knows what is in it. Thus a miner could use information on a rebase timing for their advantage. We are yet to solve that problem. As far as we are aware, none of the other tokens that have random rebase events have been able to solve that problem either.

We do not want to give any advantage to a (large) miner over a little trader, thus the traders ability to generate and see a rebase (ahead of time) should be about the same as a that of a large miners.

If (in the future) the ability to provide true randomness changes then we would like to re-write our contracts to provide true random rebase events where no one gets an advantage.

4.2 Rebase events are (at least) a day apart

In normal operations[1], no rebase event shall occur in the 24 hours after a rebase. This is to allow the market to respond to the rebase, and to allow users to transact in that period without a rebase complicating their transactions.

4.3 Users can generate rebase events.

A day after the last rebase, anyone can ***try*** and generate a rebase event as along they pay for the gas, and are willing to accept that the odds of them generating a rebase is about 20%. Anyone can do this by calling the `rebase()` function on the current AAU `Orchestrator.sol` contract.

Two days after the last rebase, anyone can generate a rebase event as along they pay for the gas. Anyone can do this by calling the `rebase()` function on the current AAU `Orchestrator.sol` contract.

At some random time that is at least a day after the last rebase, the AmpleForthGold team shall trigger a rebase event.

4.3 Traders & Miners gaming the system

There are no easy answers to stop traders & miners taking advantage of the rebase timing. All we can do is make it hard for them, and provide ways of spotting it if it occurs:

MINERS: To game it the miner would need to adjust her coinbase to correctly solve an XOR with the preceding block hash. That is do-able, but the miner would need to go out of there way to do it. It also may be obvious from looking at the coinbase used by the miner that they were doing it.

TRADERS: To game it they could just call the `rebase()` function on the current AAU `Orchestrator.sol` contract many times until it triggers. They have a 20% chance of triggering each time they call it. They could get lucky, or they could burn a lot of gas. Whatever they do it will probably be obvious from the many calls to this function.

--==00==--

1 The AmpleForthGold team have the power to perform a rebase at any time, or revert a rebase, or disable rebase operations. They may do just that if operations are abnormal.