DISCLAIMER

: This post is in no way an attack or criticism about the strategies currently being used by the participating solvers; its intent is to highlight a potential problem in the current rules and propose ways to fix it.

This is based on discussions with @marco , @fleupold and @voyta.eth .

What is pennying?

During the last few weeks, it has been observed that, in order to win the competition, multiple solvers are increasing their chance of winning an auction by “buying” additional surplus for the users using a technique we call pennying

. We describe pennying with an example.

Example of pennying

Suppose there is a single user order in a batch, that sells 1 ETH for at least 1500 USDC, and that a solver finds a Uniswap v3 pool to execute the trade, such that the pool returns 1510 USDC. This means that the computed user surplus is 10 USDC. If there is no slippage when settling the transaction on-chain, then the pool indeed returns 1510 USDC, and nothing would be pending in the weekly re-adjustment between the protocol and solvers. Now, the solver can instead report a price of 1515 USDC for 1 ETH, effectively incurring “artificial” negative slippage on the solver side, that is taken into account in the weekly accounting. What this means is that the solver is willing to pay 5 USDC from its pocket, in order to increase the objective value of its solution and make it more competitive. In the case where the solver ends up winning the auction, it pockets the 50+35=85 COW reward (currently valued at ~$10), while paying a penalty of 5 USDC = $5. Thus, the solver makes a profit of $5.

This technique of slightly inflating the prices by a small amount (usually ranging from $1 to $5) so as to increase the chance of winning an auction is what we call “pennying”.

Sample data from mainnet solver competition

To demonstrate that this is not a theoretical problem but a very real one, we give examples of different solvers, coming from a dune query that identifies slippage, using data from August 2022. Figure 1 shows the distribution of slippage for solvers that we believe are not pennying (mass concentrated around zero), while Figure 2 shows the distribution of slippage for solvers that are likely to be pennying (mass concentrated around some small negative value).

[

image (8)

1133×201 29 KB

](https://europe1.discourse-cdn.com/business20/uploads/cow/original/1X/11afff35653d5e4b124e5ccd51906b8d04eb5eec.png)

Figure 1

[

image (9)

847×201 19.2 KB

](https://europe1.discourse-cdn.com/business20/uploads/cow/original/1X/ecfb3a3e8cc693ecedeb2a5fcc8deb29a60dcb34.png)

Figure 2

Is pennying bad?

Solutions that incorporate pennying are valid given the current competition rules. The protocol does allow for deviations of reported prices compared to the actual executed AMM prices since volatility is in many cases present; this is why accounting is needed in the first place. However, we believe that allowing for some slack does not mean that solvers should be able to systematically report better solutions than they are actually able to find,and play games with the sole intent of getting a fixed COW reward for a particular batch. By using pennying, solvers are effectively getting an unfair advantage over other solvers that may have a better (or at least as good) solution but are not pennying. Again, we stress here that this is not just some theoretical observations, but we have enough evidence that demonstrate that pennying has played an instrumental role in solvers winning many batches (this is also due to many batches only containing a single order that is quite straightforward to match against some AMM); the competition itself also shows that if pennying was not effective, it would not have been used so extensively.

While all solvers could employ pennying, under aggressive competition, it is clear that pennying will lead to solvers exhausting most of their COW rewards in order to win auctions. And, in an extreme case where a rich solver enters the competition, one could imagine that particular solver essentially giving away free money, incurring losses that it can sustain for a relatively long time, to the point that other solvers cannot do anything but give up on the competition.

Dedicating time and resources into developing sophisticated pennying strategies does not lead to any meaningful progress/growth for CoW Protocol, so we believe that such a systematic manipulation of solutions should not be allowed.

How to deal with pennying

It is not trivial to detect pennying and distinguish it from normal on-chain volatility. Because of this, we believe that pennying should be explicitly disallowed, and that we could agree on some soft monitoring rules that could detect it. The course of action after that would be up to the DAO to decide, but in any case we would like a mild process where the first step would be to notify a solver and monitor its behavior, in order to see whether things improve, and if not, attempt to identify the problem and whether it is intended or not. At its most extreme form, systematic pennying could lead to slashing.

Some additional steps can also be taken. As of now, solvers are required to report expected input/output amounts in all the interactions they include in their response json. The numbers reported could be interpreted as a commitment on the solvers' side that indeed these are the amounts they expect to see on-chain. Thus, checking global token conservation, also known as token conservation per token (see here for more details), could help; as a clarification, we would only require that the reported numbers are such that token conservation per token is satisfied. This does not immediately prevent a solver from lying about these numbers, but would help serve as written commitment from the solver that indeed these are the numbers that are expected to be seen if the solution is settled on-chain. Systematic upwards deviation from these numbers could be considered as evidence of pennying (see Figure 2 above). Moreover, a requirement on top of that could also be that solvers must provide a block (around the time the batch was created) in which their solution simulates with no slippage at all; this would serve as strong proof that indeed their calculations are accurate given the information they have about the state of the blockchain.

Conclusion

The purpose of this post is to raise awareness about the quite controversial phenomenon of pennying and have meaningful discussion about it and about ways of dealing with it. Depending on how the discussion goes, this could potentially lead to a new CIP addressing pennying (e.g., prohibiting it or proposing ways to reduce its impact).