IEEE ICBC'24 1570974595

Private, Anonymous, Collateralizable Commitments vs. MEV

1st Anonymous
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

***Abstract*—We introduce private, anonymous, collateralizable commitments (PACCs), a generalized commitment scheme based on smart contract wallets and non-interactive zero knowledge proofs. PACCs allow any smart contract wallet holder to collateralize a claim, request, or commitment in general, in a private and anonymous manner. PACCs can prove arbitrarily much or little about the wallet generating the commitment, and/or the transaction which is being committed. We demonstrate that PACCs can be applied to effectively eliminate MEV in DeFi where it currently occurs, shifting MEV instead to censorship.**

***Index Terms*—Zero Knowledge, MEV, Commitments, Decentralized Finance**

I. Introduction

Miner-/maximal-extractable value (MEV) losses on Ethereum are upwards of $\$500M$ [14], [30], with actual losses across all blockchains likely in the billions. These losses are being incurred by protocol users. If genuine protocol users are losing money, this diverts funds away from protocols to extractors, entities solely interested in investing in extraction techniques, while standing as a clear deterrent to prospective users. This is a direct consequence of users announcing an intent to do something (adding an unencrypted transaction to the mempool) before that action is executed on-chain. In the two primary sources of MEV, decentralized exchange (DEX) and auctions (liquidation auctions, NFT auctions, etc.), this intent typically reveals some combination of order price, order size, sender identity, trading history, and account balances. All of these pieces of information allow extractors to extract value from players in a blockchain.

Although some protocols have emerged in literature to hide some of this information [1]–[3], [10], [16], [20], [24], no satisfactory decentralized solution has been found (discussed in Section II). One emergent technology which effectively hides transaction information is that of zero-knowledge (ZK) mixers as used in [9], [18], [24]. These allow players to join an anonymity set and send/commit to send transactions by (only) revealing membership in a set. Unfortunately, existing solutions require players to join anonymity sets in advance of proving membership, and for many other players to join the set to adequately hide identity. If protocols have disjoint anonymity sets, which all of the listed protocols do, this requires a player to restrict tokens to one of these protocols in advance of using the tokens. Furthermore, on-boarding and off-boarding into these anonymity sets require costly on-chain operations (merkle-tree additions, proof-verification), serving as bottlenecks to prospective protocols based on these anonymity sets.

A. Our Contribution

In this paper, we address the limitations of single-use anonymity sets, and their potentially costly waiting times for collateral to be usable. We propose an improved variant of single-use collateralized commitments, as in [22], [24], with a dynamic variation where collateral can be locked instantaneously for its required purpose, with commitments possible for almost any transaction, while effectively keeping that transaction private and anonymous. These crucial improvements have immediate consequences for MEV, minimizing MEV opportunities as they currently occur in DEXs, auctions, and beyond.

We describe *Private, Anonymous, Collateralizable Commitments*
(PACCs), a commitment protocol based on smart contract wallets and non-interactive ZK proofs. PACCs can be used to convince a prospective block builder or relayer that the wallet generating the PACC has enough funds to pay required fees, that the wallet is committed to performing certain actions, and importantly, that the wallet loses some amount of collateral if this commitment is broken. PACCs delegate typically expensive ZK operations off-chain, only requiring an additional one or two mapping checks when compared to transactions being sent from basic externally owned accounts, as in Ethereum [8]. Mappings are gas-efficient storage structures implemented on smart-contract enabled blockchains, and are used in our construction to enforce commitments.

We outline some basic properties of PACCs (Section IV-E), and demonstrate how PACCs can be applied to effectively eliminate MEV as it is known in DEXs, liquidations and auctions. PACCs shift MEV to censorship, which is still a concern. However, we believe censorship can be made arbitrarily expensive [15], [31] by forcing protocols to accept commitments for long enough (we introduce a parameter Δ for this *long enough* in the model assumption of Section IV), and by making all transactions practically indistinguishable, something which PACCs achieve.

B. Organisation of the Paper

Section II analyzes previous work related to the information hiding in the context of MEV-protection. Section III introduces the ZK primitives needed to formally reason about Diamond. Section IV defines the PACC protocol, with some general properties of PACCs outlined in Section IV-E. Section V demonstrates the potential of PACCs in protecting against MEV, specifically with respect to decentralized finance. We conclude, with some further discussion on PACCs, in Section VI.

II. Related Work

Related to the concept of hiding transaction information before it is committed to the blockchain are Eagle [2], P2DEX [3], and Penumbra [28]. Each of these protocols use trusted committees to keep order information hidden. If the committee colludes, all trade information is revealed. As sophisticated committees can choose when and how to use this information, users can be convinced that MEV is not occurring. Although PACCs may be replicated using these committee-based protocols, PACCs are intended to be implemented using decentralized underlying protocols free from committee-based dependencies. User-run commit-reveal [22], [24], or revealing using delay encryption [7], [32] offer potential solutions in this regard, both of which are compatible with PACCs.

With respect to Eagle, P2DEX, and LibSubmarine [22], any tokens committed to protocols run by the committee are known. These inputs are indicators of imbalances in upcoming protocols, and can be used by all players in the system to extract value from these protocols, and as such, the users. Combating this requires many transactions in every time slot to sufficiently hide the imbalance signals produced by individual users. This brings risk for early senders in every time slot, which stands as a barrier for adoption. With PACCs, commitments do not require tokens to be sent before revealing, while still ensuring these tokens must be used in the protocol, as attested to by the commitment.

FairTraDEX [24] proposes hiding trade information using non-interactive zero-knowledge proofs and anonymity sets such as those used in [4], [9]. A player wanting to participate in a FairTraDEX auction is required to join an anonymity set specific to that auction some time before an auction begins, waiting until the user is sufficiently hidden within the anonymity set before submitting an order. Joining an anonymity set and restricting one's tokens to a single use far in advance of using the tokens all have costs for users, making the practicality of the MEV protection guarantees in [24] limited.

Flax [11] allows users to anonymize sends, resembling work on burn addresses [22]. This is a definite improvement on the basic externally-owned account model where users typically send all transactions from one address. However, with regard to DEXs and auctions, price, size and direction are all revealed in Flax. As such, it provides minimal improvements with respect to MEV protection.

III. Preliminaries

This section outlines the *non-interactive zero-knowledge* (NIZK) tools for set membership as used in this paper, such as those stemming from papers like [4], [5], [17], [26]. To participate in the protocol, retail users privately generate two bit strings, the *serial number S* and *randomness r*, with $S,\ r \in \{0,1\}^{\Theta(\kappa)}$. To describe the protocol we define a commitment scheme h, a set-membership proof scheme *SetMembership*, an NIZK proof of knowledge scheme *NIZKPoK* and a NIZK signature of knowledge scheme (*NIZKSoK*). We do not specify which instantiation of these schemes to use, as the exact choice will depend on several factors, such as efficiency, resource limitations and/or the strength of the assumptions used. In the following $||$ denotes the concatenation of two strings.

- $h(m)$: A deterministic, collision-resistant function taking as input a string $m \in \{0,1\}^*$, and outputting a string $com \in \{0,1\}^{\Theta(\kappa)}$.
- *SetMembership*(com, Com): Compresses a set of commitments *Com* and generates a membership proof π that *com* is in *Com* if $com \in Com$.
- *NIZKPoK*(r, S, Com): For a set of commitments *Com*, returns a string *S* and NIZK proof of knowledge that the person running *NIZKPoK*() knows an *r* producing a proof when running *SetMembership*$(h(S||r), Com)$. This revelation identifies to a verifier when a proof has previously been provided for a particular, albeit unknown, commitment as the prover reproduces *S*.

IV. PACCs

In this section we describe PACCs and how they can be constructed using existing blockchain functionalities. We then outline some basic properties of PACCs.

A. Model

To reason about the properties of PACCs when applied to MEV, we introduce the following assumptions.

1) A transaction submitted by a player for addition to the blockchain while observing blockchain height H, is finalized in a block of height at most $H + \Delta$, for some known $\Delta > 0$.
2) The public NIZK parameters are set-up in a trusted manner.[1]
3) External market prices exist for all tokens, and follow Martingale processes.
4) There exists a population of arbitrageurs able to frictionlessly trade at the external market price, who continuously monitor and interact with the blockchain.
5) All players in the system are represented by a wallet located on a single blockchain-based distributed ledger, and a corresponding PKI.

[1] An example of such a set-up is a Perpetual Powers of Tau ceremony, as used in Zcash https://zkproof.org/2021/06/30/setup-ceremonies/

B. Smart Contract Wallets

This section introduces smart contract wallets as they are used in this paper. For ease of notation, we shorten smart contract wallet to just wallet. It suffices to consider wallets as extensions of externally-owned accounts on which we can apply additional constraints. To reason about our framework, we assume some finite number of token denominations n, with the total quantity of tokens in the system denoted $T \in \mathbb{Z}_+^n$. For simplicity, we assume all tokens in the system are contained in the set of wallets. For W the set of all wallets, and some wallet $w \in W$, w can be considered as a set of tokens. The function $bal(w) := [v_1, ..., v_n]$ indicates that there are exactly v_i token i's in wallet w. As such, $\sum_{w \in W} bal(w) = T$. With this in hand, we introduce two distinct commitment mappings:

- For every wallet $w \in W$, a *secret commitment mapping* $Com^w_{scrt} : \{0,1\}^{\Theta(\kappa)} \to v \subseteq w$. $Com^w_{scrt}(x)$ is such that if $x \neq y$, $Com^w_{scrt}(x) \cap Com^w_{scrt}(y) = \emptyset$. This $Com^w_{scrt}(x)$ is a mapping of secrets to mutually exclusive subsets of tokens in the wallet w.
- A *global transaction commitment mapping* $C_T : \{0,1\}^{\Theta(\kappa)} \to \{0,1\}^{\Theta(\kappa)}$. This is used to track transaction commitments made by users. Transaction commitments are mapped from a unique piece of information which is also linked to a secret commitment mapping, and as such, a set of tokens. This is used to ensure that if a transaction commitment is made, the only way to use the set of tokens linked to the secret commitment is to reveal the unique committed transaction.

Players in our system use wallets which maintain a mapping of secret commitments to tokens within those wallets. At initialization all tokens in the wallet are mapped from the 0 secret commitment. To submit a transaction tx using some subset of tokens v in the wallet w, users must submit (S, r) satisfying 2 requirements:

1) Users must provide a signature, as in a basic externally-owned account, such that $v \in Com^w_{scrt}(h(S, r))$.
2) If $C_T(S)$ is non-zero, it must be that $C_T(S) = h(tx)$. This is used to ensure that the user adheres to any commitment that they have made.

If either of these requirements are not met, the transaction is invalid.

At the end of a transaction, the user must generate new mappings for all unmapped tokens in the wallets, with the default being the 0 secret commitment. If $Com^w_{scrt}(com = h(S||r)) = v$ for some set of tokens v, v is fixed (tokens cannot be added or removed from v) until S is revealed. This is crucial in preventing a player from committing to a transaction which sends tokens without the player owning those tokens. If a player commits to bidding in an auction without any tokens in their wallet, but can add them before the tokens are needed, the player is able to effectively only reveal bids when bids are favourable.

C. PACCs Framework

In the PACCs framework, there are two types of entity: players controlling wallets and *relayers*. For any valid transaction commitment $h(tx)$, there exists a *relayer fee*, a set of tokens described by fee, and a *collateral*, a set of tokens described by $collateral$. The relayer fee and collateral are such that for any wallet w, a relayer adds $h(tx)$ to the blockchain in at most Δ blocks if:

1) w commits to send fee to the relayer when tx is executed.
2) w commits to burning $collateral$ if tx is not executed.

Importantly, in this description of a relayer, there is no requirement for the relayer to know tx, only $h(tx)$, fee and $collateral$. Depending on how the transactions are revealed, this introduces risk for a relayer, either due to some possibility of not revealing or opportunity costs related to accepting a fee now, but receiving it later. For professional relayers handling thousands of transactions, fee can be adjusted to reflect these risks. Considering the cost for not revealing is strictly greater for provers due to $collateral$, which itself can be made arbitrarily large, relayers will be incentivized to participate. This is either through fee collection, or potential deflation due to forfeited collateral.

Consider a player P_i wishing to add $h(tx)$ to the blockchain without anyone knowing that P_i generated tx, or exactly what tx is. We emphasize exactly as we believe it is still important that P_i convinces a relayer that tx is executable, and that tx pays the relayer upon execution. To do this we introduce the concept of a *break point* with respect to a transaction. A break point is such that for a transaction included for execution on a blockchain with a break point, the transaction executes every valid command up until the break point, regardless of what follows the break point. Thus, inserting a break-point can be considered as splitting up a transaction into two consecutive sub-transactions. With this, we define base PACC transactions.

Definition IV.1. A transaction tx is a *base PACC transaction* if $h(tx) = h(tx_1 \,||tx_2)$ for tx_1 equivalent to: *pay* fee*, break-point*. The set of all base PACC transactions is denoted by T_{base}. Given the indistinguishability of base PACC transactions given only $h(tx)$ and a knowledge of the prefix tx_1, we consider fee to be independent of the transaction contents $\forall$ $tx \in T_{base}$.

Recall that given a set of commitments *Com* as specified in Section III, any valid NIZKPoK from a player corresponding to $com = h(S||r)$ must reveal S. As such, consider a set of players $P_1, ..., P_k$ owning wallets $w_1, ..., w_k$ who create a single non-zero secret commitment mapping mapping for all of the tokens in their respective wallets. Let these mappings be of the form $com_i = h(S_i||r_i)$ for privately generated values S_i and r_i for each P_i. We place the restriction on S_i that there exists a $rootKey_i$, with $(rootKey_i, S_i)$ a valid private key, public key pair in the pre-defined PKI for C_T. Typically, this means S_i is derived from $rootKey_i$.

With this, we have enough to ensure relayers add transaction commitments on-chain within Δ blocks. Specifically, relayers will add transaction commitments to the global transaction commitment mapping, C_T. To demonstrate this, we outline a protocol which can be run between relayer and a wallet

w wishing to insert a transaction commitment $h(tx)$ to the blockchain.

D. PACCs Protocol

Consider the set of all wallets $W = \{w_1, ..., w_n\}$. Although PACCs allow for multiple secret commitments per wallet, WLOG, let the set of secret commitments $Com = \{com_1, ..., com_n\}$ be such that $Com^{w_i}_{scrt}(com_i) = bal(w_i)$, $\forall\ i \in [1, ..., n]$. That is, there is one secret commitment per wallet, with the secret commitment mapping for each wallet mapping a single secret commitment to all of the tokens in that wallet. Let $W_b \subseteq W$ be the set of wallets with $bal(w_b) \geq fee + collateral$, $\forall\ w_b \in W_b$, and Com_b be the secret commitments corresponding to these wallets. Specifically, $Com^{w}_{scrt}(com_b) \geq fee + collateral$, $\forall com_b \in Com_b$. Therefore for a wallet w_i with secret commitment com_i, $com_i \in Com_b$ if and only if $w_i \in W_b$. This implies such a wallet $w \in W_b$ can produce a valid NIZKPoK(r, S, Com_b).

Further consider a wallet $w_i \in W_b$ wishing to insert a commitment to a transaction $tx_i \in T_{base}$ into the global transaction commitment mapping. As $tx_i \in T_{base}$, this implies $tx_i = tx_1||tx_2$ with tx_1 equivalent to: pay fee, break point. This means w_i can produce a valid NIZKPoK($tx_1, tx_2, \{h(tx_i)\}$). As this NIZKPoK reveals tx_1 and proves membership in $\{h(tx_i)\}$, it must be that tx_1 is a prefix of tx_i, without revealing anything else about tx_i.

Therefore, let w_i send (NIZKPoK(r_i, S_i, Com_b), NIZKPoK(tx_1, tx_2, $\{h(tx_i)\}$), and a signature of this message using $rootKey_i$. This signature can be verified using S_i, which is revealed by NIZKPoK(r_i, S_i, Com_b). This further ensures the player generating $h(tx_i)$ must also have generated S_i, under standard PKI assumptions.

From the first NIZKPoK, the relayer knows that $bal(w_i) \geq fee + collateral$, as such a proof is only possible if $w_i \in W_b$. Accompanied with the second NIZKPoK, the relayer then knows:

1) w commits to send fee to the relayer when tx_i is executed.
2) w commits to burning $collateral$ if tx_i is not executed. This is because the relayer knows that at least $collateral$ exists in the wallet. Furthermore, as S_i is mapped to $h(tx_i)$ in the global transaction commitment mapping C_T, by definition of C_T, only revealing tx_i allows w_i to use the tokens mapped from $com_i = h(S_i||r_i)$ in $Com^{w_i}_{scrt}()$. As such, if tx_i is never revealed, at least $collateral$ is burned in w_i.

E. Properties of PACCs

Towards applying PACCs to MEV, we detail some of properties that PACCs possess. The first result is that, within the competition assumptions of Section IV-A, the expected collateral required to submit a PACC diminishes to 0.

Lemma IV.2. For a base PACC transaction tx from a wallet w with $bal(w) > fee$, the Nash Equilibrium for $collateral = 0$.

Proof. For any wallet w posting a commitment to such a transaction tx with $bal(w) > fee$, the payoff for revealing tx is at least $collateral$, compared to 0 for not revealing. Therefore, any collateral greater than 0 is sufficient to incentivize revelation. Given a cost to w for locking collateral, the optimal collateral for w is 0. Therefore, w posts the minimum possible collateral, which approaches 0. □

Next, we show that PACCs can be used to run sealed-bid auctions.

Lemma IV.3. Sealed-bid auctions can be run among all wallets W such that $bal(w) > fee,\ \forall\ w \in W$.

Proof. Consider such a wallet $w \in W$, and an on-chain auction A. Let A accept bid commitments for Δ blocks, followed by Δ blocks in which bids committed to A can be revealed. Consider a base PACC transaction tx, with $tx_2 \equiv$ *bid X in auction* A for some set of tokens X. We know for any $w \in W$ committing to tx and proving membership in W, a relayer adds $h(tx)$ to the blockchain within Δ blocks. Therefore any $w \in W$ submitting this bid commitment and proofs to a relayer is included in A. Put differently, after the initial Δ blocks, any player who wanted to commit a bid for A has been included in the blockchain.

After the initial Δ blocks, all players reveal their bids. As bids can be revealed for up to Δ blocks, this is sufficiently long for all players revealing bids to be added to the blockchain. As the committed transactions are base PACC transactions, all players committing to a bid are incentivized to reveal. Importantly, all bids were committed before any information regarding other bidders was revealed, with revealed bids matching committed bids due to the requirement of the global transaction commitment mapping for revealed transactions from w to match any transaction committed by w.

Thus, at the end of Δ blocks, all bids are revealed on chain. Any player in the blockchain can then settle the auction, deducing the auction winner and settlement price from the revealed bids. This is sufficient to run a sealed-bid auction. □

This has immediate consequences for MEV in liquidations. Such MEV is highlighted empirically in [29]. The source of this MEV is the ability for players to trigger liquidation of collateral and buy the collateral at a discount. To address this, we can replace the *Auction Liquidations* as labelled in [29] with sealed-bid auctions based on PACCs. Using a result from auction theory on sealed-bid auctions among competing players with the same view of external market prices [21], we know that seller revenue of such an auction is at least the value of the tokens being sold. As PACCs allow player with tokens to participate in any auction, PACCs also allow for such a sealed-bid auction among competing player. Given the external market prices of these tokens follow Martingales with which arbitrageurs can interact with frictionlessly (Section IV-A), the expected revenue of a liquidation auction when the auction starts is at least that of the external market value of the collateral when the auction starts, as required.

Finally, a subtle yet important property of PACCs is that a commitment to a valid send transaction requires the tokens to be present in the sending wallet when the commitment is made. Specifically, for a wallet w with $Com^{w}_{scrt}(h(S,r)) = v$ for some set of tokens v and any transaction tx from w sending tokens to another smart contract or wallet, if $C_T(S) = h(tx)$ before tx is revealed, for tx to be valid, it must be that there were enough tokens in v when the secret commitment map was created. This is by definition of $Com^{w}_{scrt}()$, with only tokens mapped from S being usable in tx.

V. PACCs and MEV

With the core PACCs protocol described in Section IV, we are able to shift almost all known MEV to censorship. Given competing block producers and transaction fees, censorship can be made arbitrarily expensive and unlikely, as the cost to bribe block producers to censor is at most $O(\Delta)$ [15]. This section applies PACCs to the main sources of MEV beyond basic auctions, and demonstrates some of the potential of PACCs in tackling MEV. We focus on demonstrating how PACCs enable users and/or protocols to buy or sell tokens at relevant external market prices, excluding transaction fees. This is analogous to MEV prevention in most documented sources of MEV [12], [29], [30].

A. Decentralized Exchange

Decentralized exchange is the primary source of MEV in current blockchains ($> 99\%$ of the MEV identified as extracted by [14], as seen in the chart labelled "Extracted MEV Split by Type"). From the same principles as Lemma IV.3, PACCs can implement a significantly more powerful auction with respect to decentralized exchange, a frequent batch auction (FBA) [6]. FBAs are auctions designed to connect buyers and sellers of a particular token/set of tokens, and are proven to settle at external market prices under competition among MMs. A decentralized variant of an FBA is implemented in the FairTraDEX protocol [24]. FairTraDEX demonstrated that implementing FBAs was possible in a decentralized setting without the trusted auctioneer required in [6].

Corollary V.1. Frequent batch auctions can be run among all wallets W such that $bal(w) > fee,\ \forall\ w \in W$.

Proof. From Lemma IV.3, we know we can implement any auction involving sealed-bids in any set of tokens. For any pair of token sets (X, Y), consider an auction A where players submit bids as base PACC transactions, of the form $tx_1||tx_2$, with tx_2 equivalent to *buy* $v_x > 0$ *of* X *for* $v_y > 0$ *of* Y or *buy* $v_y > 0$ *of* Y *for* $v_x > 0$ *of* X. If the auction is run over 2Δ blocks, this is enough for players to commit sealed bids to the blockchain, and have them revealed. From [24], we can implement an on-chain clearing-price verifier to ensure orders of this format get executed at a unique clearing price which maximizes traded volume, and as such, replicates a frequent batch auction. □

FairTraDEX requires bids to be collateralized, indistinguishable, and committed to the blockchain before being revealed and settled at a unique clearing price. PACCs allow for all of these requirements, while providing significant improvements on the FairTraDEX protocol. Players using PACCs can participate in FBAs in any token/token-set pair at any time (in addition to optionally performing normal transactions without additional on- or off-boarding), compared to players in FairTraDEX being restricted to FBAs in one trading pair. With PACCs, players can immediately join any ongoing FBA. With the anonymity set being the set of wallets, there is never a need to diffuse or on-board other players for an auction. Furthermore, PACCs remove the need to perform Merkle-Tree inserts and verifications on-chain, making PACCs significantly cheaper and more scalable.

1) PACC RFQ protocol: Despite the promises of FBAs, adoption is limited. With PACCs, there is potential for other DEX alternatives with similar price guarantees, which can be seen as more aligned with the desired experience of retail users. In this regard, we consider first a request-for-quote (RFQ) style DEX protocol based on PACCs, which resembles a fee-escalator as proposed in one of the Ethereum Improvement Proposals [13].

For a swap between two sets of tokens X and Y with external market price ϵ, specifically $\epsilon X = Y$, consider a PACC from a wallet w proving membership in a set of wallets with *at least* X *tokens or* Y *tokens*. Consider $f_H : \mathbb{N}_{\geq H} \to \mathbb{R}_{\geq 0}$, an increasing function with $f_H(h)$ undefined for $h < H$. The function $f_H(h)$ defines a commission to be paid by the user to the relayer including their transaction in the chain at block height h.

The accompanying transaction commitment is proved to be a member of the set $\{com_1, com_2\}$, with:

- $com_1 = h(tx_1 = tx_a||tx_b)$ and $tx_a \equiv$ *sell* X *for* Y*, pay* $f_H(h)$ *to relayer, break point*,
- $com_2 = h(tx_2 = tx_c||tx_d)$ and $tx_c \equiv$ *sell* Y *for* X*, pay* $f_H(h)$ *to relayer, break point*.

A relayer adding the commitment to the blockchain must collateralize the order with at least X and Y to be valid, with the relayer trading with w when the committed transaction is revealed. After Δ blocks from when w's transaction is committed to the blockchain, the relayer can reclaim their own collateral. After this point, w's tokens are burnt if w reveals the committed transaction, or locked indefinitely if not.

Lemma V.2. Given no transaction fees, the PACC RFQ protocol has Nash Equilibrium involving MMs committing user orders to the blockchain with $E(f_H) = 0$.

Proof. Firstly, given the order is generated when the external market price is ϵ, for any block generated after this point (after block H), the expected external market price is ϵ due to the Martingale assumption. As all MMs observe ϵ, the expected revenue for a MM responding to a PACC RFQ is at least 0 and strictly increasing in block height, making this a Dutch Auction in the commission specified by $f_H()$ between MMs. By the revenue equivalence principle [21], the expected revenue for the seller is the same as if this were an sealed-bid auction to receive $f_H()$. For any positive value of f_H, MMs

will bid for this opportunity. As the user chooses $f_H()$, the commission to pay, but does not receive the revenue of the auction, for any positive value for $f_H(h)$, $h \geq H$, the user is strictly incentivized to reduce $f_H(h)$. As such, the Nash Equilibrium of $E(f_H)$ is $E(f_H) = 0$. □

Corollary V.3. Given no transaction fees, the PACC RFQ protocol has Nash Equilibrium in which users trade at the external market price.

This protocol has many of the expected price benefits of an FBA based on PACCs, with users expecting to trade at the external market price excluding fees in both cases. PACC RFQs come with the added benefit for users that users are only required to reveal when the order has been executed. Depending on the MM preferences or requirements, PACC RFQs can also be used to enforce Know-Your-Customer and/or anti-money laundering checks by MMs before responding, with MMs able to require arbitrary membership proof rules. For example, if European MMs choose to only respond to wallets who have received a verification token from European regulators, PACCs allow users to preserve anonymity within this set, while maintaining the privacy required to trade at the external market price in expectancy.

As mentioned, this resembles a fee escalator [13], with the strict benefit of not revealing trade price, direction, or identity to searchers before the trade has been committed to. All of these pieces of information together can be enough to move the external market price before the order gets interacted with. If searchers see a protocol creator selling protocol tokens, this could have significant negative sentiment effects on the price of the token. Therefore, the searchers might pre-emptively move the price when such an order enters the mempool. In contrast, if the same player committed to either buying or selling tokens using a PACC, with a membership proof in a group equally likely to buy or sell, the user expects to trade around the external market price.

2) PACCs and AMMs: After the highlighting of the phenomenon of loss-versus-rebalancing [27], we expect block producers to arbitrage an AMM price to the external market price. A recent AMM proposal using ZK commitments [23] provides an interesting use case for PACCs. Consider then an AMM, such as that introduced in [23] which accepts PACCs. Given the delay between commitment and revelation, it seems necessary that the AMM is required to lock-up reserves to trade with user orders. For this to be viable from an AMMs standpoint (opportunity costs are added), the AMM should require some fee to be deposited by a relayer allocating funds for a user order (which can be incorporated by the relayer into the fee/collateral required from a PACC). If all orders outside of the block producer arbitrage are executed at the same price, a user should expect their order, at worst, to be executed at the external market price at time of commitment, minus the fees and impact for interacting with the AMM. As the arbitrage from the producer is unaffected, all PACC orders paying inclusion fees should be added to the chain.

Current AMM users only expect their orders to trade after a searcher has moved the price of the AMM, meaning interacting with the AMM at the external market price is the best-case scenario for users currently. Given the extent of user-level MEV [14], [30], this best-case scenario has not been considered attainable. As such, PACCs have the potential to drastically reduce MEV in AMMs. Following recent work on LVR-proofing AMMs [19], [25], PACCs may be pivotal in making AMMs LVR- and MEV-proof.

VI. Conclusion

We outline PACCs, a protocol allowing anyone with sufficient capital to anonymously and privately commit collateralized transactions for any protocol to a blockchain. This is compared to earlier solutions based on anonymity sets [2], [3], [24] that force players trying to achieve the similar levels of anonymization and privacy for collateralized commitments to join anonymity sets in advance of the opportunity, typically before it exists. This necessity to lock up capital at some, potentially significant, opportunity cost limits the applicability of such solutions.

The trade-off with PACCs is the dependency on relayers to post collateral on behalf of unknown players, with only game-theoretic guarantees of repayment. We see this as an acceptable trade-off, introducing some unpredictability with respect to when and if rewards are paid out. Importantly, collateral and fees from wallets can be enforced to reflect this unpredictability, and given competition among relayers for these fees, the equilibrium for these fees should approach the gas costs paid by relayers for including the transaction as indicated in Lemma IV.2.

Although we propose PACCs for use in commit-reveal protocols, mainly due to their provable decentralization, and ability to be implemented immediately on any smart-contract enabled blockchain, there are several alternatives that are worth mentioning. In our description of PACCs, the relayer includes transaction commitments to the blockchain. These commitments can be replaced by a threshold encryption of the transaction [1], [28], or using a delay encryption scheme [7]. In these encryption schemes, it may be possible to reduce collateral requirements, although practical and decentralized variations of these schemes have yet to be proposed. Improvements in these areas will greatly improve the usability and capabilities of PACCs. Importantly, all schemes, including the current description of PACCs, have a distinct committal of information to the chain, followed by a revelation of information. As such, commit-reveal accurately describes the process taking place, regardless of the specific revelation scheme being used.

References

[1] Avi Asayag, Gad Cohen, Ido Grayevsky, Maya Leshkowitz, Ori Rottenstreich, Ronen Tamari, and David Yakira. Helix: A Fair Blockchain Consensus Protocol Resistant to Ordering Manipulation. IEEE Transactions on Network and Service Management, 18(2):1584–1597, 2021.

[2] Carsten Baum, James Hsin-yu Chiang, Bernardo David, and Tore Kasper Frederiksen. Eagle: Efficient privacy preserving smart contracts. https://eprint.iacr.org/2022/1435, 2022. Accessed: 23/01/2023.

[3] Carsten Baum, Bernardo David, and Tore Frederiksen. P2DEX: Privacy-Preserving Decentralized Cryptocurrency Exchange. In Kazue Sako and Nils Ole Tippenhauer, editors, Applied Cryptography and Network Security, pages 163–194. Springer International Publishing, 2021.

[4] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized Anonymous Payments from Bitcoin. In 2014 IEEE Symposium on Security and Privacy, pages 459–474, New York, NY, USA, 2014. IEEE Computer Society.

[5] Daniel Benarroch, Matteo Campanelli, Dario Fiore, Kobi Gurkan, and Dimitris vKolonelos. Zero-Knowledge Proofs for Set Membership: Efficient, Succinct, Modular. In Nikita Borisov and Claudia Diaz, editors, Financial Cryptography and Data Security, pages 393–414, Berlin, Heidelberg, 2021. Springer Berlin Heidelberg.

[6] Eric Budish, Peter Cramton, and John Shim. The High-Frequency Trading Arms Race: Frequent Batch Auctions as a Market Design Response *. The Quarterly Journal of Economics, 130(4):1547–1621, 07 2015.

[7] Jeffrey Burdges and Luca De Feo. Delay encryption. In Anne Canteaut and François-Xavier Standaert, editors, Advances in Cryptology – EUROCRYPT 2021, pages 302–326, Cham, 2021. Springer International Publishing.

[8] Vitalik Buterin. Ethereum: A next-generation smart contract and decentralized application platform. https://github.com/ethereum/wiki/wiki/White-Paper, 2014. Accessed: 01/12/2020.

[9] Tornado Cash. https://tornado.cash/. Accessed: 25/07/2022.

[10] Theodoros Constantinides and John Cartlidge. Block Auction: A General Blockchain Protocol for Privacy-Preserving and Verifiable Periodic Double Auctions. In 2021 IEEE International Conference on Blockchain (Blockchain), pages 513–520, United States, 2021. IEEE Computer Society.

[11] Wei Dai. Flexible anonymous transactions (flax): Towards privacy-preserving and composable decentralized finance. https://eprint.iacr.org/2021/1249, 2021. Accessed: 23/01/2023.

[12] Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels. Flash Boys 2.0: Frontrunning, Transaction Reordering, and Consensus Instability in Decentralized Exchanges. https://arxiv.org/abs/1904.05234, 2019. Accessed: 19/01/2022.

[13] EIP-2593 . https://eips.ethereum.org/EIPS/eip-2593. Accessed: 23/01/2023.

[14] Flashbots. https://explore.flashbots.net. Accessed: 11/10/2022.

[15] Elijah Fox, Mallesh Pai, and Max Resnick. Censorship Resistance in On-Chain Auctions. https://arxiv.org/abs/2301.13321, 2023. Accessed: 12/07/2023.

[16] Hisham S. Galal and Amr M. Youssef. Publicly Verifiable and Secrecy Preserving Periodic Auctions. In Matthew Bernhard, Andrea Bracciali, Lewis Gudgeon, Thomas Haines, Ariah Klages-Mundt, Shin'ichiro Matsuo, Daniel Perez, Massimiliano Sala, and Sam Werner, editors, Financial Cryptography and Data Security. FC 2021 International Workshops, pages 348–363, Berlin, Heidelberg, 2021. Springer Berlin Heidelberg.

[17] Jens Groth. On the Size of Pairing-Based Non-interactive Arguments. In Marc Fischlin and Jean-Sébastien Coron, editors, Advances in Cryptology – EUROCRYPT 2016, pages 305–326, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[18] Kobi Gurkan, Koh Wei Jie, and Barry Whitehat. Community Proposal: Semaphore: Zero-Knowledge Signaling on Ethereum, 2020. Accessed: 25/01/2022.

[19] Josojo. MEV capturing AMMs. https://ethresear.ch/t/mev-capturing-amm-mcamm/13336, 2022. Accessed: 18/10/2022.

[20] Mahimna Kelkar, Fan Zhang, Steven Goldfeder, and Ari Juels. Order-Fairness for Byzantine Consensus. In Daniele Micciancio and Thomas Ristenpart, editors, Advances in Cryptology - CRYPTO 2020, pages 451–480, Cham, 2020. Springer International Publishing.

[21] Vijay Krishna. Auction theory. Academic press, 2009.

[22] LibSubmarine. https://libsubmarine.org/. Accessed: 23/01/2023.

[23] Conor McMenamin and Vanesa Daza. An AMM minimizing user-level extractable value and loss-versus-rebalancing. https://arxiv.org/abs/2301.13599, 2023. Accessed: 12/07/2023.

[24] Conor McMenamin, Vanesa Daza, Matthias Fitzi, and Padraic O'Donoghue. FairTraDEX: A Decentralised Exchange Preventing Value Extraction. In Proceedings of the 2022 ACM CCS Workshop on Decentralized Finance and Security, DeFi'22, page 39–46, New York, NY, USA, 2022. Association for Computing Machinery.

[25] Conor McMenamin, Vanesa Daza, and Bruno Mazorra. Diamonds are Forever, Loss-Versus-Rebalancing is Not. https://arxiv.org/abs/2210.10601, 2022. Accessed: 04/01/2023.

[26] Ian Miers, Christina Garman, Matthew Green, and Aviel D. Rubin. Zerocoin: Anonymous Distributed E-Cash from Bitcoin. In 2013 IEEE Symposium on Security and Privacy, pages 397–411, United States, 2013. IEEE Computer Society.

[27] Jason Milionis, Ciamac C. Moallemi, Tim Roughgarden, and Anthony Lee Zhang. Quantifying Loss in Automated Market Makers. In Fan Zhang and Patrick McCorry, editors, Proceedings of the 2022 ACM CCS Workshop on Decentralized Finance and Security. ACM, 2022.

[28] Penumbra. https://penumbra.zone/. Accessed: 23/01/2023.

[29] Kaihua Qin, Liyi Zhou, Pablo Gamito, Philipp Jovanovic, and Arthur Gervais. An empirical study of defi liquidations: Incentives, risks, and instabilities. In Proceedings of the 21st ACM Internet Measurement Conference, IMC '21, page 336–350, New York, NY, USA, 2021. Association for Computing Machinery.

[30] Kaihua Qin, Liyi Zhou, and Arthur Gervais. Quantifying Blockchain Extractable Value: How dark is the forest? In 2022 IEEE Symposium on Security and Privacy (SP), pages 198–214, 2022.

[31] Alvaro Revuelta. Statistical analysis on Ethereum k-consecutive block proposal probabilities and case study. https://alrevuelta.github.io/posts/ethereum-mev-multiblock. Accessed: 23/01/2023.

[32] James Hsin yu Chiang, Bernardo David, Ittay Eyal, and Tiantian Gong. Fairpos: Input fairness in proof-of-stake with adaptive security. https://eprint.iacr.org/2022/1442, 2022. https://eprint.iacr.org/2022/1442.

APPENDIX