Amoveo: Peer-to-Peer Synthetic Derivatives

Luke Besser
Cosimo Capital
luke@cosimo.fund

Phil Bonello
Ikigai Asset Management
phil@ikigai.fund

Jack Miller
Cosimo Capital
jack@cosimo.fund

The most interesting use cases in crypto are those that are uniquely enabled by trust-minimized, decentralized technology. Global, permissionless prediction markets are one of those use cases.

Amoveo is a proof of work blockchain that uses a variant of SHA256. It facilitates trust-minimized prediction markets. It does this through the use of oracles which learn facts about the real world and enforce the outcomes through contracts governed by network users. Prediction markets have the potential to be wildly powerful, and Amoveo is the soundest implementation to date. With its genesis block on March 2, 2018, it was also the first working implementation of this type of system.

Permissionless synthetic assets, at the heart of Amoveo's design, allow for investors to gain exposure to underlying products of almost any kind. There is no need for a trusted price feed like some other solutions have proposed. This opens a wide range of possibilities for an open financial system.

The thought space for prediction markets, Amoveo's direction of design and the developer focused community inspire excitement. We also believe many of the approaches taken by Amoveo could inspire and streamline development of other projects in the ecosystem.

Goal of Overview

- This paper serves as a high-level review of Amoveo, a project that we believe presents a novel and game theoretically sound approach to solving portions of the oracle problem, widely considered to be one of the most difficult in the industry.
- Amoveo is not widely known. Because of its relative obscurity, there is limited developer mindshare. We hope to change that. This is a call to arms to build on one of the most interesting projects in the space.
- Liquidity is king. Without sufficient liquidity, markets are inefficient and ineffective. We hope to educate the community on Amoveo in an effort to bring liquidity to prediction markets that are live today and catalyze the creation of new markets.
- The derivatives market is one of largest addressable blockchain opportunities. "Unstoppable derivatives" can serve as the foundation for countless incentive designs. We are very excited about the possibilities. Through iterative discourse, we hope to drive the development of Amoveo and other promising solutions.

Background

Prediction markets aim to capture the "wisdom of the crowd" by providing a financial incentive for correctly predicting an outcome. Historically, similar prediction markets have shown to be incredibly accurate. The most popular example is that of Wall Street betting odds. "The Wall Street odds represent the consensus of a large body of extremely impartial opinion that talks with money and approaches Coolidge and Davis as dispassionately as it pronounces judgment on Anaconda and Bethlehem Steel." - New York Times 1924.

Prediction markets have historically been useful in predicting election results and outcomes of sporting events, but the implications are much larger. In a recent research paper by Circle Research the team provided a useful example: *Theoretically, a middle class citizen in China can now get exposure to Apple stock by buying shares in the prediction market "What will the share price of Apple be at the end of 2018?"* There has been a lot of talk around open finance recently. *That* is open finance. Prediction markets could even enable participants to actively create new

markets to hedge against the possibility of real-world events - think insurance against losing your home to an earthquake. Through monetary rewards, prediction markets can incentivize users to share accurate real-world data, ultimately communicating probabilities about future events to decentralized, immutable blockchains.

Past prediction market experimentation has revealed shortcomings that have historically skewed the accuracy of predictions.

- Illiquidity creates exorbitant opportunity costs resulting in inefficient market pricing.
- High transaction costs further distort the equilibrium bid price. It is important to have cheap transactions to allow for efficient price discovery.
- Regulation, capital controls, geographical barriers and counterparty risk have historically prevented widespread adoption of prediction markets. Censorship resistant networks have the ability to mitigate these risks.
- Relying on third parties to host prediction markets requires a high degree of trust. Because the third-party entity is handling the transactions, you have to trust that the party will fulfill the original agreements of contract and not steal your money.

Public blockchains have the potential to solve the issues of past prediction markets. A well implemented solution, governed by a decentralized user-base, would transcend geographical borders and thus bring increased liquidity and low barriers to entry. Because of the censorship resistant nature of many public blockchains, prediction markets can more freely operate increasing accuracy and power of predictions. Furthermore, because these networks are open source and remove the need for a middleman, transaction fees trend towards the cost of execution. Counterparty risk is removed which further adds to the efficacy of the prediction market. Amoveo is especially efficient as there is no additional capital requirement on behalf of reporters in the way Augur's oracles function.

Introduction

Amoveo was founded by Zack Hess. He has been architecting and developing the solution for about four years. Prior to launching Amoveo, Zack designed and implemented MVPs for Augur and Aeternity. He has been one of the most prominent voices in addressing the game theory and security around oracle solutions. Much of his commentary can be found on the Bitcoin Hivemind forum where he and Paul Sztorc share extensive thought on the design of prediction markets. In preparing this overview, it became evident that Zack has been a consistent leader in the prediction market space.

Amoveo adopts concepts such as futarchy from Robin Hanson, the idea of trustless prediction markets from Paul Sztorc, Nakamoto consensus from Bitcoin, and atomic swaps from Tier Nolan. Uniquely, Amoveo uses betting in combination with Nakamoto consensus to build secure on-chain oracles. Additionally, it implemented a new form of off-chain non-custodial exchange through payment channels, a concept that allows for derivatives trading with synthetic assets. A recent paper by Dan Robinson titled *The Rainbow Network: An Off-Chain Decentralized Synthetics Exchange* summarizes many of the concepts that Amoveo has implemented since its launch in March 2018.

Amoveo supports both binary and scalar approaches. An oracle in the Amoveo network is a binary question that is hosted on-chain. Scalar oracles are made by interpreting 10 binary oracles as the 10 bits of a number. Paired with Nakamoto consensus, the outcome of an oracle is determined by the aggregate amount of VEO on each side. There are three potential outcomes: True, False, and Bad Question. An oracle resolves after a predetermined resolution date and an abatement of betting occur in conjunction. Intuitively, betting activity subsides as there is diminishing opportunity for arbitrage between the oracle and the real-world outcome. If an oracle receives inadequate turnout, it resolves to "Bad Question."

Think of an oracle as the underlying asset in the ecosystem - the ultimate truth, the foundation. From there, Amoveo gives users the ability to create off-chain markets that settle with the outcome of the underlying oracle. Complex contracts can be hosted within channels allowing for the development of derivatives. While the Amoveo blockchain is not Turing-complete, channels are. Markets can then be hosted through a collection of bidirectional channels that interact with the same server. Additionally, these channels are compatible with the lightning network allowing users to route a bitcoin lightning payment through Amoveo markets. Any number of markets can be hosted on top of a respective oracle. These markets are hosted by third-party servers that have no control of the off-chain contracts.

Additionally, the aggregate size of the market economy can far exceed the size of the oracle betting without concerns of corruption.

On Amoveo, oracles run independently of the markets that sit atop. Unlike voting based solutions, oracles cannot be corrupted through bribery because there is no notion of voters. Instead, there are only bettors. A bad actor can attempt to corrupt an oracle only by betting money on an incorrect outcome. In this scenario, an economic actor would fill the arbitrage opportunity by betting on the correct outcome. This is in contrast to prominent implementations such as Augur which separate the role of bettor and voter. Augur's system, and others which introduce voter-based oracles, are susceptible to parasitic markets whereby a voter reward is less than a potential bribe to the voter, creating a positive expected value for corrupting the system.

To simplify, below are the steps for interacting with an oracle:

1. An entity creates an oracle and pays a small fee which bootstraps liquidity.
2. After an event occurs, bettors place a bounty on a side of the oracle to report the outcome.
3. Entities create markets pointing to the outcome of an oracle. These markets can be hosted P2P or by third party providers that can host non-custodial channels.
4. Users bet on off-chain markets that are hosted in bidirectional channels. Bets are denominated in VEO but currency risk can be minimized through the creation and use of stable synthetic instruments.
5. Market betting stops at a predetermined closing time.
6. The oracle is resolved after an event has occurred and betting has stopped. Bettors are paid.
7. The market contract is resolved based on the outcome of the underlying oracle.
8. Bettors are paid accordingly.

Architecture and Design Approach

Oracles

Amoveo takes a unique approach to solving the oracle problem. An oracle is effectively a binary question, the results of which are stored as native objects in the network state. These questions are a way of aligning real-world truths with immutable, blockchain truths. A question might be: "Will the price of Bitcoin be greater than $10,000 on January 1st, 2019?" The answer to the question is determined by an on-chain betting mechanism. Users bet on a side of an outcome and they are rewarded for betting on the correct side. After enough time has passed without any new bets, the oracle resolves to the side with most VEO behind it. The initial liquidity is provided by the party who proposed the oracle question.

If someone tried to cheat by placing a large bet on the incorrect side there would be a large economic incentive to stake VEO behind the correct side and, if needed, enlist other profit-motivated token holders. This incentive for users to place bets against false information is the key to the security of Amoveo's oracle system. The last resort would be to fork the chain in the event of a successful attack on the oracle that caused it to report an incorrect value. The miners are incentivized to follow the honest chain because following the dishonest chain is uneconomical.

Amoveo's approach allows for a simple design with a small attack surface. It removes the need for an additional token by using the native currency of its blockchain. It allows smart contracts to use real-world data while keeping the total size of the network state very small. This lowers the complexity of on-chain scaling and also ensures a low cost of running a full node.

The oracle problem is a massive hurdle for smart contract systems. Many real-world problems simply require awareness of certain data points. Applications using smart contracts have the additional requirement that this data be provided in a way that doesn't create a single point of failure. Rune Christiansen of MakerDAO, one of the premiere Ethereum dapps, has stated that the oracle layer is the weak-link in Maker's system. Amoveo elegantly solves a problem that plagues other systems, reducing complexity, cost, and chain size. The fact that oracles are native on-chain objects also means that oracle data can be read by off-chain smart contracts, which are a unique innovation in Amoveo. It's possible that the oracle problem is so critical to decentralized financial applications that an elegant solution with on-chain objects might give Amoveo a distinct advantage in cost and transaction throughput.

Governance
Governance in Amoveo runs on a process known as futarchy, whereby changes are determined through betting. A proposal is submitted, and an oracle is created. The market will be something like "proposal X will increase the value of VEO by y" and users will bet on whether they believe this is true. **One interesting property of this system is that instead of "one token, one vote," we have "one token risked, one vote."** This means you could lose your money by voting recklessly or trying to manipulate the market in bad faith. It's possible this results in a more accurate representation of the stakeholders' beliefs about a given proposal compared to direct token voting. The governance model is defined by the value of certain protocol parameters. These parameters can be broken down into four different categories:

- Fees for certain types of transactions
- Oracle mechanics such as the initial liquidity requirement, duration, and maximum question size
- Blockchain mechanics such as block time, size, and reward, and how much of the block reward is distributed to the developers (currently 0.2 VEO per block)
- Gas limit and maximum size of the smart contracts

These values are stored on-chain and can be adjusted using an oracle question which is resolved with bets like any other oracle question. The efficacy of this process depends on VEO holders being incentivized to bet on values that make VEO most valuable. Changes can be made without requiring a hard fork update nor software upgrades to propagate across users of the network to enact the change. Instead, the value is changed on-chain and will be utilized by the smart contracts immediately.

Channels
A channel is an agreement between two parties to lock up VEO for a certain amount of time with the promise to sync up with the blockchain when the time expires or when one or both parties decide to close the channel. Channels require one transaction to open and another to close. A small fee is required to open a channel, but any number of transactions may be executed while the channel is open. Once a channel is closed, any change in the VEO balances are written to the blockchain. This is useful because users can perform fewer transactions on-chain, which require paying a fee and waiting for your transaction to be confirmed. Instead, users make transactions that are executed instantly, have no transaction fee, and which are guaranteed to be authentic by digital signatures from each party. **Channels allow for the same performance and experience users expect from centrally hosted markets but with the assurances of a secure, censorship-resistant network.** Using channels, Amoveo can achieve massive throughput without the high fees and slow wait times experienced by other blockchains.

Smart Contracts Inside Channels
Besides simply sending VEO between users, channels are used to execute Amoveo's smart contracts. These smart contracts are Turing Complete, meaning there is no restriction on the programs that can be run inside them. Complex logic and financial derivatives can be defined in these smart contracts without the need to store all information inside the blockchain state. This massively reduces the scaling requirements for throughput on-chain and means *Amoveo could scale to thousands of transactions per second, or more, and execute high-volume, trustless financial agreements at a low cost.*

The trade-off is that users are required to be online to participate in channels. This means that in practice, market operators will likely be those with the resources and expertise to be service providers. This has a centralizing trend; however, security guarantees are still enforced by Merkle proofs and a challenge mechanism penalizes channel operators that try to cheat. It's trivial to prove there was cheating using digital signatures and those caught cheating have their funds slashed.

Another protection for users making bets is that all off-chain markets in Amoveo match orders in batches instead of first come, first serve. The market contract will pick the optimal price and combine any matching orders all at once at a single price instead of combining orders that match as they come. **This prevents front-running and makes it more difficult for trading bots or high frequency traders to take advantage of users with an advantage in hardware, internet connection, or privileged access to markets and order books hosted by central entities.**

Channels are created when both parties publish a transaction that they have signed using their private key. Any updates to the state of the data inside of the smart contract within the channel are signed with digital signatures from

both parties which verifies their integrity. Under normal circumstances, both parties will agree, and a special transaction will be signed by both parties and submitted to close the channel. Upon closing the channel, the appropriate amount of VEO will be distributed to each party as defined by the rules of the smart contract and any updates to the state.

In the case of a disagreement or the disappearance of one party, the channel can be closed unilaterally by a single party with a transaction that only requires one signature. This transaction is submitted with the most recent data inside the smart contract. Any updates to the data in the smart contract will increase a nonce, which is just an ever-increasing number that tracks the order of transactions. The nonce is how the blockchain is able to adjudicate disputes about smart contract state without an intermediary. For example, if someone tried to cheat by submitting a transaction to unilaterally close a channel with an old contract state that didn't contain some recent losing bets they made, the other party in the channel is given a certain amount of time to submit evidence of cheating. If the other party submits a different contract state, then the blockchain will simply compare the nonces between the two states and side with the state with a higher nonce. Since we know the states are valid by digital signatures from both parties, the nonce is the only remaining data point required to determine which state is the most recent and thus correct. Any party caught trying to cheat will forfeit funds to the submitter of the valid proof.

This system introduces the requirement for the user of a channel to maintain the latest state within the smart contract along with the channel operator or else they would not be able to submit proof against the cheater. While this is an additional hurdle, wallet software and other third parties can safely store backups of the current state as the signed state does not contain any sensitive data. It does require that a user maintain vigilance and internet connectivity to detect against a dishonest channel operator closing early. However, evidence of cheating can be submitted by parties who are not members of the channel. Third parties will be incentivized to monitor the blockchain and verify that channels are being closed honestly to receive the reward that comes from slashing the funds of any cheaters.

Derivatives, not Subcurrency

Users can create financial derivatives using any logic they wish in the off-chain smart contracts. These derivatives exist entirely off-chain which keeps the network state small. This also minimizes the memory required on-chain, and thus reduces the costs to maintain the network state. When you open a channel in Amoveo, the network only needs to store the VEO balance of the parties in the channel yet can interact and send any arbitrary asset defined by a smart contract. **Amoveo could host thousands of different synthetic assets without any increase in the size of the network state.** In Ethereum or Aeternity, a channel needs to store the balance of each token needed in the channel, meaning the memory requirements for the channel grow with the number of tokens.

It's possible to make stablecoin derivatives contracts. One example of a stable asset would be an asset pegged to the US dollar. A stablecoin contract requires a buyer and a seller. If you want $100 of a USD stablecoin then you need someone to take the other side and buy $100 of "long-VEO" which will be worth the market price of VEO at the time of the market's closing. If the price of VEO goes up, then the holder of long-VEO will have made a little money by buying VEO at a discount. The stablecoin holder will receive less VEO, but they will retain the same amount of value in USD terms. If the price of VEO goes down, upon closing the contract the stablecoin holder receives some additional VEO to maintain the USD value and the long-VEO holder loses a little VEO.

If the market is illiquid then you might have difficulty finding someone to take the other side, or you might have to take the bet at an unfavorable price. However, assuming there is a large enough profit motive and enough interest in the markets around the price of stable assets, users will be able to hedge risk appropriately. It is currently not possible to use a stablecoin to bet in another market directly. A user is free to buy another derivative to hedge their risk, but this would require two bets: one bet in their desired market and one in the stablecoin market. Another way is that the creator of an oracle can also set up the market such that the market's payout is denominated in some currency, like USD, for example. Markets can be effectively priced in any desired currency at the creator's discretion assuming the use of that currency has sufficient demand.

Chalang Smart Contract Virtual Machine

Chalang is a functional programming language created by Zack Hess. It is particularly designed for off-chain smart contracts. The virtual machine does not contain the "goto" opcode that other smart contract language VMs have, which reduces complexity and makes it easier to analyze. When a channel is opened, the two parties digitally sign the smart contract signaling that they agree. The VM also contains the concept of gas to prevent abuse, which

causes the contract to stop executing when all the gas has been used up. There is one type of gas that is consumed for each line of code executed and another type consumed by the amount of memory required.

Lightning Network
Although channels are opened between two parties and run off-chain, smart contracts can be associated with one another across different channels allowing multiple participants within a single smart contract application. This is done using a similar procedure to an atomic swap, or a simultaneous moving of funds across different blockchains. An atomic swap involves making one transaction on each blockchain with rules that state that the transaction will remain locked until a predefined secret is revealed. Once the secret is revealed, both transactions become unlocked and are added to the blockchain at the same time. A key property is that the two transactions are either both locked or both added to the blockchain, but never one without the other. This same technique can be applied to smart contracts within different channels, where code is executed instead of simply funds being moved. This mechanic is what allows for the Lightning Network in Bitcoin and enables participants to participate in dapps or make payments across different channels in Amoveo.

Competitive Advantage

The status quo for decentralized prediction markets is currently Augur, but Amoveo presents compelling advantages.

Amoveo presents stronger game theoretic guarantees through the combination of market-based (opposed to voter based) resolution combined with Nakamoto consensus. There is no opportunity for parasitic actors to corrupt the integrity of a market.

Amoveo is more capital efficient in that there is no need for voters. Voter based systems require voters to lock in capital to resolve an oracle in order to receive a reward for their work. These fees required to incentivize voters increase the aggregate cost of participating in a market and make it more inefficient.

The ability for Amoveo to host off-chain non-custodial derivatives markets allows for fast and cheap execution while providing strong security and infrastructure assurances. Far less customization is required to build a scalable application like Veil for hosting markets. While the interface is not as smooth as Veil yet, Amoveobook uses the channels to trustlessly host derivatives markets rather than hosting private orderbooks.

Current State

Amoveo has been operational since March 2, 2018. The futarchy-modeled governance system has worked as anticipated. Markets are hosted off-chain and are currently available to the non-technical user through interfaces like amoveobook.com. In short, the blockchain and the accompanying off-chain solutions are operational.

The project is young. The current applications are promising but few. In general, the user experience in cryptocurrency leaves a lot to be desired. For markets to gain traction, we would like to see continued UX work to decrease friction. This is a common issue with blockchain products, but Amoveo has solved some of the most pressing issues, namely, scalability. Additionally, there are a number of teams that are focused on developing wallets and user interfaces for hosting markets on Amoveo.

Amoveo did not hold an ICO and has not prioritized marketing. In our view, these are positive indicators, but additional bootstrapping is required. **For Amoveo to flourish, it needs an increased amount of developer mindshare and market participants. In the same way projects like Veil Markets have launched on Augur, entrepreneurs will continue to build out ancillary products for Amoveo.** These projects will help drive the efficacy of Amoveo oracles and markets.

Zack's development work represents 98% of GitHub activity. For a "decentralized" project, we would like to see a wider development distribution. Prediction markets are one of the most interesting blockchain use cases. Open derivatives could serve as the foundation for thousands of incentive mechanisms. Currently, there is limited developer mindshare devoted to this sector. There is significant room to explore and we hope to see more in the near future.

Conclusion

Amoveo has designed and implemented a very unique set of features with a special focus on game theory. This focus allows for interesting experimentation in prediction markets, fundraising (dominant assurance contracts), and governance (futarchy). The project handles complex contracts such as the creation of synthetic assets through state channels allowing for the blockchain state to remain lean. The design approach and the development-focused community are positive signals for continued innovation, and there are a number of approaches that the broader community may find valuable. **Because markets are held off-chain but point towards on-chain oracles, channel operators who will be the predominant UI owners, can concentrate on the UI/UX.** The architecture offers strong assurances to the developers and to the users. We plan to support the project with operational, go-to-market, and development resources.

Zack originally developed the MVP for both Augur and Aeternity before contentious departures. In both past projects, he took different approaches which he ultimately decided were flawed. The evolution of his thinking, and his ultimate departure from the prevailing approach of voter-based markets are chronicled in Hivemind forums. Zack is well respected for his economic discourse and game theory. He has published some of the most progressive thought around the subject of decentralized prediction markets.

Amoveo has taken the approach that many are now realizing is the correct one - host complex, computationally heavy operations off-chain. This decreases the cost of creating and betting on markets and minimizes blockchain bloat. The game theory that governs Amoveo's oracles is the soundest we've seen. This is incredibly important for the efficacy of large-scale prediction markets. Again, it offers strong assurances network to stakeholders.

We hope this overview of Amoveo piques curiosity in the crypto community and drives meaningful development work. Amoveo has its quirks but is one of the most interesting projects in the space.

Resources

Whitepaper
- https://github.com/zack-bitcoin/amoveo/blob/master/docs/white_paper.md

Source code and original documentation
- https://github.com/zack-bitcoin/amoveo

Newsletter & Blog
- https://amoveo.substack.com/
- https://medium.com/amoveo

Additional Reading
- https://medium.com/@tallakt/amoveos-first-futarchy-market-233d01b9fe53
- https://github.com/zack-bitcoin/amoveo/blob/master/docs/design/limit_order_in_channel.md
- https://github.com/zack-bitcoin/amoveo/blob/master/docs/design/oracle.md

Exchanges
- https://qtrade.io/
- https://gozo.pro/
- https://amoveo.exchange/

Statistics
- https://coinpaprika.com/coin/veo-amoveo/

Mining
- https://amoveo.io/mining/

Tools
- https://veoscan.io/
- https://amoveobook.com/
- https://amoveo.io/en/

Commands

- https://veodocs.github.io/#docs/api/commands.md

The information stated above represents Ikigai Asset Management's (together with its subsidiaries and affiliates and their principals, "Ikigai") current views and opinions, and may change at any time without notice. This article may contain "forward-looking statements," including Ikigai's expectations regarding the performance of the companies and assets discussed herein. Although Ikigai believes that the assumptions and expectations reflected in these forward-looking statements are reasonable, there can be no assurance that these expectations will prove to be correct and actual results may vary. Ikigai, its affiliates or principals, own an interest in the project discussed in this article.

Cosimo Capital has an investment interest in Amoveo. This document does not in any way constitute an offer or solicitation of an offer to buy or sell any investment or token. This document may contain "forward-looking statements" which are not statements of historical fact. All forward-looking statements are inherently uncertain as they depend on various expectations, assumptions, and ideas concerning future events and they are subject to known and unknown risks and uncertainties which could cause actual events or results to differ materially from the projected events.