Augur: a Decentralized Oracle and Prediction Market Platform

Jack Peterson, Joseph Krug, Micah Zoltu, Austin K. Williams, and Stephanie Alexander
Forecast Foundation
(Dated: March 5, 2018)

Augur is a trustless, decentralized oracle and platform for prediction markets. The outcomes of Augur's prediction markets are chosen by users that hold Augur's native Reputation token, who stake their tokens on the actual observed outcome and, in return, receive settlement fees from the markets. Augur's incentive structure is designed to ensure that honest, accurate reporting of outcomes is always the most profitable option for Reputation token holders. Token holders can post progressively-larger Reputation bonds to dispute proposed market outcomes. If the size of these bonds reaches a certain threshold, Reputation splits into multiple versions, one for each possible outcome of the disputed market; token holders must then exchange their Reputation tokens for one of these versions. Versions of Reputation which do not correspond to the real-world outcome will become worthless, as no one will participate in prediction markets unless they are confident that the markets will resolve correctly. Therefore, token holders will select the only version of Reputation which they know will continue to have value: the version that corresponds to reality.

Augur is a trustless, decentralized oracle and prediction market platform. In a prediction market, individuals can speculate on the outcomes of future events; those who forecast the outcome correctly win money, and those who forecast incorrectly lose money [1–3]. The price of a prediction market can serve as a precise and well-calibrated indicator of how likely an event is to occur [4–7].

Using Augur, people will have the ability to trade in prediction markets at very low cost. The only significant expenses participants assume is compensation to market creators and to users that report on the outcomes of markets once the event has taken place. The result is a prediction market where trust requirements, friction, and fees will be as low as competitive market forces can drive them.

Historically, prediction markets have been centralized. The simplest way to aggregate trades in a prediction market is for a trustworthy entity to maintain a ledger; similarly, the simplest way to determine the outcome of an event and distribute payouts to traders is for an impartial, trusted judge to determine the outcomes of the markets. However, centralized prediction markets have many risks and limitations: they do not allow global participation, they limit what types of markets can be created or traded, and they require traders to trust the market operator to not steal funds and to resolve markets correctly.

Augur aims to resolve markets in a fully decentralized way. Decentralized, trustless networks, such as Bitcoin[8] and Ethereum[9], eliminate the risk that self-interest will turn into corruption or theft. The only role of the Augur developers is to publish smart contracts to the Ethereum network. The Augur contracts are totally automated: the developers do not have the ability to spend funds that are held in escrow on-contract, do not control how markets resolve, do not approve or reject trades or other transactions on the network, cannot undo trades, cannot modify or cancel orders, etc. The Augur *oracle* allows information to be migrated from the real world to a blockchain without relying on a trusted intermediary. Augur will be the world's first decentralized oracle.

I. HOW AUGUR WORKS

Augur markets follow a four-stage progression: *creation*, *trading*, *reporting*, and *settlement*. Anyone can create a market based on any real-world event. Trading begins immediately after market creation, and all users are free to trade on any market. After the event on which the market is based has occurred, the outcome of the event is determined by Augur's oracle. Once the outcome is determined, traders can close out their positions and collect their payouts.

Augur has a native token, Reputation (REP). REP is needed by market creators and by reporters when they report on the outcome of markets created on the Augur platform. Reporters report on a market by *staking* their REP on one of the market's possible outcomes. By doing this, the reporter declares that the outcome on which the stake was placed matches the real-world outcome of the market's underlying event. The consensus of a market's reporters is considered the "truth" for the purpose of determining the market's outcome. If a reporter's report of a market's outcome does not match the consensus reached by the other reporters, Augur redistributes the REP staked on the non-consensus outcome by this reporter to the reporters that reported with the consensus.

By owning REP, and participating in the accurate reporting on the outcomes of events, token holders are entitled to a portion of the fees on the platform. Each staked REP token entitles its holder to an equal portion of Augur's market fees. The more REP a reporter owns, and reports correctly with, the more fees they will earn for their work in keeping the platform secure.

Although REP plays a central role in Augur's operations, it is not used to trade in Augur's markets. Traders will never need to own or use REP, as they are not required to participate in the reporting process.

Figure 1. Simplified outline of the lifetime of a prediction market.

A. Market Creation

Augur allows anyone to create a market about any upcoming event. The *market creator* sets the *event end time* and chooses a *designated reporter* to report the outcome of the event. The designated reporter does not unilaterally decide the outcome of the market; the community always has an opportunity to dispute and correct the designated reporter's report.

Next, the market creator chooses a *resolution source* that reporters should use to determine the outcome. The resolution source may simply be "common knowledge", or it may be a specific source, such as "The United States Department of Energy", `bbc.com`, or the address of a particular API endpoint.[1] They also set a *creator fee*, which is the fee paid to the market creator by traders who settle with the market contract (see Section I D for details on fees). Finally, the market creator posts two bonds: the *validity bond*, and the *designated report no-show bond* (also referred to as the *no-show bond* for brevity).

The validity bond is paid in ETH and is returned to the market creator if the market resolves to any outcome other than *invalid*.[2] The validity bond incentivizes market creators to create markets based on well-defined events with objective, unambiguous outcomes. The size of the validity bond is set dynamically, based on the proportion of invalid outcomes in recent markets.[3]

The no-show bond consists of two parts: the *no-show gas bond* (paid in ETH) and the *no-show REP bond* (paid in REP). These bonds are returned to the market creator if the market's designated reporter actually reports during the first three days after the market's *event end time*. If the designated reporter does not submit their report during the allotted 3-day window, then the market creator forfeits the no-show bond and it is given to the *first public reporter* who reports on the market (see Section I C 6). This incentivizes the market creator to choose a reliable designated reporter, which should help markets resolve quickly.

The no-show gas bond is intended to cover the first public reporter's gas costs. This prevents the scenario where the first public reporter's gas costs are too high for reporting to be profitable. The no-show gas bond is set at twice the average gas cost for reporting during the previous fee window.

In the event that the designated reporter fails to report, the no-show REP bond is given to the first public reporter in the form of stake on their reported outcome, so that the first public reporter receives the no-show REP bond if and only if they report correctly. As with the validity bond, the no-show REP bond is adjusted dynamically based on the proportion of designated reporters who failed to report on time during the previous fee window.[4]

The market creator creates the market and posts all required bonds via a single Ethereum transaction. Once the transaction is confirmed, the market is live and trading begins.

B. Trading

Market participants forecast the outcomes of events by trading *shares* of those market outcomes. A *complete set of shares* is a collection of shares that consists of one share of each possible valid outcome of the event [10]. Complete sets are created by Augur's on-contract matching engine as needed to complete trades.

For example, consider a market that has two possible outcomes, `A` and `B`. Alice is willing to pay 0.7 ETH for a share of `A` and Bob is willing to pay 0.3 ETH for a share of `B`.[5] First, Augur matches these orders and collects a total of 1 ETH from Alice and Bob.[6] Then Augur creates a complete set of shares, giving Alice the share of `A` and Bob the share of `B`. This is how shares of outcomes come into existence. Once the shares are created, they can be traded freely.

---

[1] For example, if a market on "The high temperature (in degrees Fahrenheit) on April 10, 2018 at the San Francisco International Airport, as reported by Weather Underground" specifies a resolution source of `https://www.wunderground.com/history/airport/KSFO/2018/4/10/DailyHistory.html`, reporters would simply go to that URL and enter the high temperature displayed there as their report.

[2] An *invalid market* is a market determined to be invalid by reporters because none of the outcomes listed by the market creator is correct, or because the market wording is ambiguous or subjective; see Section III F for discussion.

[3] See Appendix E 1 for details.

[4] See Appendix E 2 for details.

[5] Initially, trades in Augur's markets will use Ethereum's native coin, Ether (ETH). Subsequent releases of Augur will include support for markets denominated in arbitrary tokens issued on the Ethereum network, including shares of other markets as well as tokens pegged to fiat currencies ("stablecoins"), if/when these become available.

[6] The 1 ETH figure is used here for ease of discussion. The actual cost of a complete set of shares is much smaller than this; see `docs.augur.net/#number-of-ticks` for details.

The Augur trading contracts maintain an order book for every market created on the platform. Anybody can create a new order or fill an existing order at any time. Orders are filled by an automated matching engine that exists within Augur's smart contracts. Requests to buy or sell shares are fulfilled immediately if there is a matching order already on the order book. It may be filled by buying shares from or selling shares to other participants, which, may involve issuing new complete sets or closing out existing complete sets. Augur's matching engine always sequesters the minimum amount of shares and/or cash needed to cover the value at risk. If there is no matching order, or the request can be only partially filled, the remainder is placed on the order book as a new order.

Orders are never executed at a worse price than the limit price set by the trader, but may be executed at a better price. Unfilled and partially-filled orders can be removed from the order book by the order's creator at any time. Fees are paid by traders only when complete sets of shares are sold; settlement fees are discussed in more detail in Section I D.

While most trading of shares is expected to happen before market settlement, shares can be traded any time after market creation. All Augur assets – including shares in market outcomes, fee window tokens, shares in dispute bonds, and even ownership of the markets themselves – are transferable at all times.

C. Reporting

Once a market's underlying event occurs, the outcome must be determined in order for the market to finalize and begin settlement. Outcomes are determined by Augur's oracle, which consists of profit-motivated reporters, who simply report the actual, real-world outcome of the event. Anyone who owns REP may participate in the reporting and disputing of outcomes. Reporters whose reports are consistent with consensus are financially rewarded, while those whose reports are not consistent with consensus are financially penalized (see Section I D 3).

1. *Fee Windows*

Augur's reporting system runs on a cycle of consecutive 7-day long *fee windows*. All fees collected by Augur during a given fee window are added to the *reporting fee pool* for that fee window. At the end of the fee window, the reporting fee pool is paid out to REP holders who participated in the reporting process. Reporters receive rewards in proportion to the amount of REP they staked during that fee window. Participation includes: staking during an initial report, disputing a tentative outcome, or purchasing *participation tokens*.

2. *Participation Tokens*

During any fee window, REP holders may purchase any number of participation tokens for one attorep[7] each. At the end of the fee window, they may redeem their participation tokens for one attorep each, in addition to a proportional share of the fee window's *reporting fee pool*. If there were no actions (*e.g.*, submitting a report or disputing a report submitted by another user) needed of a reporter, the reporter may purchase participation tokens to indicate that they showed up for the fee window. Just like staked REP, participation tokens may be redeemed by their owners for a *pro rata* portion of fees in this fee window.

As discussed in Section II, it is important that REP holders are ready to participate in market resolution in the event of a fork. The participation token provides an incentive for REP holders to monitor the platform at least once per week, and, thus, be ready to participate if the need arises. Even REP holders who do not want to participate in the reporting process are incentivized to check-in with Augur once per 7-day fee window in order to buy participation tokens and collect fees. This regular, active checking-in will ensure that they are familiar with how to use Augur, will be aware of forks when they occur, and thus should be more ready to participate in forks when they happen.

3. *Market State Progression*

Augur markets can be in seven different states after creation. The potential states, or "phases", of an Augur market are as follows:

- Pre-reporting
- Designated Reporting
- Open Reporting
- Waiting for the Next Fee Window to Begin
- Dispute Round
- Fork
- Finalized

The relationship between these states can be seen in Fig. 2.

4. *Pre-reporting*

The *pre-reporting* or *trading* phase (Fig. 1) is the time period that begins after trading has begun in the market,

[7] One *attorep* is 10^{-18} REP.

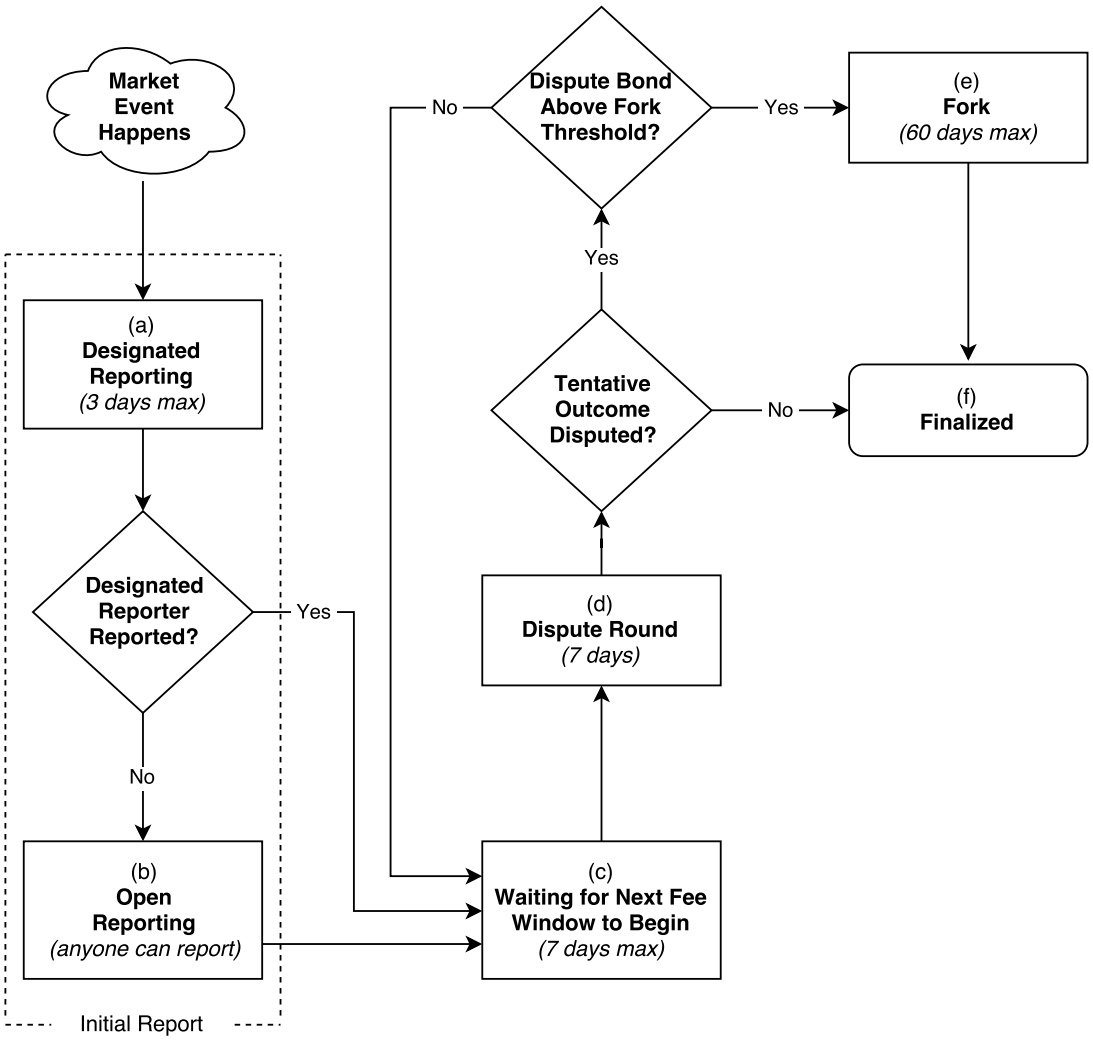

Figure 2. Reporting flowchart.

but before the market's event has come to pass. Generally, this is the most active trading period for any given Augur market. Once the event end date has passed, the market enters the *designated reporting* phase (Fig. 2a).

5. *Designated Reporting*

When creating a market, market creators are required to choose a designated reporter and post a no-show bond. During the designated reporting phase (Fig. 2a) the market's designated reporter has up to three days to report on the outcome of the event. If the designated reporter fails to report within the allotted three days, the market creator forfeits the no-show bond, and the market automatically enters the *open reporting* phase (Fig. 2b).

If the designated reporter submits a report on time, then the no-show bond is returned to the market creator. The designated reporter is required to post the designated reporter stake[8] on its reported outcome, which it will forfeit if the market finalizes to any outcome other than the one they reported.[9] As soon as the designated reporter submits its report, the market enters the *waiting for next fee window to begin* phase (Fig. 2c), and the reported outcome becomes the market's *tentative outcome*.

6. *Open Reporting*

If the designated reporter fails to report within the allotted three days, the market creator forfeits the no-show bond, and the market immediately enters the *open*

[8] See appendix E 3 for details on the size of the designated reporter stake.

[9] Forfeited stake is added to the reporting fee pool of the market's assigned fee window, and is used to reward honest reporters and disputers; see Section I D 3 for details.

reporting phase (Fig. 2b). As soon as the market enters the open reporting phase, anyone can report the outcome of the market. When the designated reporter fails to report, the first reporter who reports on the outcome of a market is called the market's *first public reporter*.

The market's first public reporter receives the forfeited no-show bond in the form of stake on their chosen outcome, so they may claim the no-show REP bond only if their reported outcome agrees with the market's final outcome. They also receive the no-show gas bond after the market has finalized only if their reported outcome agrees with the market's final outcome.

The first public reporter does *not* need to stake any of their own REP when reporting the outcome of the market. In this way, any market whose designated reporter fails to report is expected to have its outcome reported by *someone* very soon after entering the open reporting phase.

Once an *initial report* has been received by the initial reporter (whether it was the designated reporter or first public reporter), the reported outcome becomes the market's tentative outcome, and the market enters the waiting for next fee window to begin phase (Fig. 2c).

7. Waiting for Next Fee Window to Begin

Once the market receives its initial report, it enters the waiting for next fee window to begin phase (Fig. 2c). During this phase, reporting for the market is on hold until end of the current fee window. Once the next fee window begins, the market enters the *dispute round* phase.

8. Dispute Round

The dispute round (Fig. 2d) is a 7-day period during which any REP holder has the opportunity to dispute the market's *tentative outcome*.[10] (At the beginning of a dispute round, a market's tentative outcome is the outcome that will become the market's final outcome if it is not successfully disputed by REP holders.) A dispute consists of *staking* REP (referred to as *dispute stake* in this context) on an outcome *other than* the market's current tentative outcome. A dispute is *successful* if the total amount of dispute stake on some outcome meets the *dispute bond size* required for the current round. The dispute bond size is computed as follows.

Let A_n denote the total stake over all of this market's outcomes at the beginning of dispute round n. Let ω be any market outcome *other than* the market's tentative outcome at the beginning of this dispute round. Let $S(\omega, n)$ denote the total amount of stake on outcome ω at the beginning of dispute n. Then the size of the *dispute bond* needed to successfully dispute the current tentative outcome in favor of the new outcome ω during round n is denoted $B(\omega, n)$ and is given by:

$$B(\omega, n) = 2A_n - 3S(\omega, n) \tag{1}$$

The bond sizes are chosen this way to ensure a fixed ROI of 50% for reporters who successfully dispute false outcomes (see Section II D).

The dispute bonds need not be paid in their entirety by a single user. The Augur platform allows participants to crowdsource dispute bonds. Any user who sees an incorrect tentative outcome can dispute that outcome by staking REP on an outcome other than the tentative outcome. If any outcome (other than the tentative outcome) accumulates enough dispute stake to fill its dispute bond, the current tentative outcome will be successfully disputed.

In the case of a successful dispute, the market will either undergo another dispute round, or it will enter the *fork* state (Fig. 2e). If the size of the filled dispute bond is greater than 2.5% of all REP, then the market will enter the fork state. If the size of the filled dispute bond is less than 2.5% of all REP, then the newly chosen outcome becomes the market's new tentative outcome, and the market undergoes another dispute round.

All dispute stake is held in escrow during the dispute round. If a dispute bond is unsuccessful, then the dispute stake is returned to its owners at the end of the dispute round. If no dispute is successful during the 7-day dispute round, the market enters the *finalized* state (Fig. 2f), and its tentative outcome is accepted as its *final outcome*. A market's final outcome is the tentative outcome that passes through a dispute round without being successfully disputed, or is determined via a fork. Augur's contracts treat final outcomes as *truth* and pay out accordingly.

All unsuccessful dispute stake is returned to the original owners at the end of every dispute round. All successful dispute stake is applied to the outcome it championed, and remains there until the market is finalized (or until a fork occurs in some other Augur market). All dispute stake (whether successful or unsuccessful) will receive a portion of the *reporting fee pool*[11] from the current fee window.

9. Fork

The fork state (Fig. 2e) is a special state that lasts up to 60 days. Forking is the market resolution method of

[10] The fact that the dispute rounds coincide with the fee windows is purely a matter of convenience; in principle, dispute rounds and fee window durations could be different.

[11] Any settlement fees and validity bonds collected during a fee window get added to that fee window's reporting fee pool. At the end of the fee window, the reporting fee pool is paid out to users in proportion to the amount of REP they staked during that fee window.

last resort; it is a very disruptive process and is intended to be a rare occurrence. A fork is caused when there is a market with an outcome with a successfully-filled dispute bond of at least 2.5% of all REP. This market is referred to as the *forking market*.

When a fork is initiated, a 60-day[12] *forking period* begins. Disputing for all other non-finalized markets is put on hold until the end of this forking period. The forking period is much longer than the usual fee window because the platform needs to provide ample time for REP holders and service providers (such as wallets and exchanges) to prepare. A fork's final outcome cannot be disputed.

Every Augur market and all REP tokens exist in some *universe*. REP tokens can be used to report on outcomes (and thus earn fees) *only* for markets that exist in the same universe as the REP tokens. When Augur first launches, all markets and all REP will exist together in the *genesis universe*.

When a market forks, new universes are created. Forking creates a new *child universe* for each possible outcome of the forking market (including `Invalid`, as discussed in Section I D 2). For example, a "binary" market has 3 possible outcomes: `A`, `B`, and `Invalid`. Thus, a binary forking market will create three new child universes: universe `A`, universe `B`, and universe `Invalid`. Initially, these newly created universes are empty: they contain no markets or REP tokens.

When a fork is initiated, the *parent universe* becomes permanently *locked*. In a locked universe, no new markets may be created. Users may continue trading shares in markets in locked universes, and markets in a locked universe may still receive their initial reports. However, no reporting rewards are paid out there, and markets in locked universes cannot be finalized. In order for markets or REP tokens in the locked universe to be useful, they must first be migrated to a child universe.

Holders of REP tokens in the parent universe may migrate their tokens to a child universe of their choice. This choice should be considered carefully, because migration is one-way; it cannot be reversed. Tokens cannot be sent from one sibling universe to another. *Migration is a permanent commitment of REP tokens to a particular market outcome.* REP tokens that migrate to different child universes ought to be considered entirely separate tokens, and service providers like wallets and exchanges ought to list them as such.

When a fork is initiated, all REP staked on all non-forking markets is *unstaked* so that it is free to be migrated to a child universe during the forking period.[13]

Whichever child universe receives the most migrated REP by the end of the forking period becomes the *winning universe*, and its corresponding outcome becomes the final outcome of the forking market. Un-finalized markets in the parent universe may be migrated only to the winning universe and, if they have received an initial report, are reset back to the waiting for next fee window to begin phase.

There is no time limit to migrate tokens from the parent universe to a child universe. Tokens may be migrated after the forking period, but they will not count towards the determination of the winning universe. To encourage greater participation during the forking period, all token holders who migrate their REP within 60 days of the start of a fork will receive 5% additional REP in the child universe to which they migrated[14]. This reward is paid for by minting new REP tokens.[15]

Reporters that have staked REP on one of the forking market's outcomes cannot change their position during a fork. REP that was staked on an outcome in the parent universe can be migrated only to the child universe that corresponds to that outcome. For example, if a reporter helped fulfill a successful dispute bond in favor of outcome `A` during some dispute round, then the REP they have staked on outcome `A` can only be migrated to universe `A` during a fork.

Sibling universes are entirely disjoint. REP tokens that exist in one universe cannot be used to report on events or earn rewards from markets in another universe. Since users presumably will not want to create or trade on markets in a universe whose oracle is untrustworthy, REP that exists in a universe that does not correspond to objective reality is unlikely to earn its owner any fees, and therefore should not hold any significant market value. Therefore, REP tokens migrated to a universe which does not correspond to objective reality should hold no market value, regardless of whether or not the objectively false universe ends up being the winning universe after a fork. This has important security consequences, which we discuss in Section II.

10. Finalized

A market enters the finalized state (Fig. 2f) if it passes through a 7-day dispute round without having its tentative outcome successfully disputed, or after completion of a fork. The outcome of a fork cannot be disputed and is always considered final at the end of the forking period.

---

[12] Forking periods can be less than 60 days: a forking period ends when either 60 days have passed, or more than 50% of all genesis REP is migrated to some child universe.

[13] The only exception is the REP staked by the initial reporter when they made the initial report. That REP remains staked on the initial reported outcome and is automatically migrated to the child universe that wins the fork.

[14] This occurs even when the forking period has ended early due to more than 50% of all REP being migrated to some child universe.

[15] The effect of this addition to the money supply of REP is small. For example, if 20% of all existing REP is migrated during the forking period of a fork, this bonus would result in a 1% increase in the money supply of REP. Moreover, forks are expected to be exceedingly rare events.

Once a market is finalized, traders can settle their positions directly with the market. When a market enters the finalized state, we refer to its chosen outcome as the *final outcome*.

D. Market Settlement

A trader can close their position in one of two ways: by selling the shares they hold to another trader in exchange for currency, or by settling their shares with the market. Recall that every share comes into existence as part of a complete set when a total of 1 ETH has been escrowed with Augur.[6] To get that 1 ETH out of escrow, traders must give Augur either a complete set or, if the market has finalized, a share of the winning outcome. When this exchange happens we say traders are *settling with the market contract*.

For example, consider a non-finalized market with possible outcomes `A` and `B`. Suppose Alice has a share of outcome `A` that she wants to sell for 0.7 ETH and Bob has a share of outcome `B` that he wants to sell for 0.3 ETH. First, Augur matches these orders and collects the `A` and `B` shares from the participants. Then Augur gives 0.7 ETH (minus fees) to Alice and 0.3 ETH (minus fees) to Bob.

As a second example, consider a finalized market whose winning outcome is `A`. Alice has a share of `A` and wants to cash it in. She sends her share of `A` to Augur and in return receives 1 ETH (minus fees).

1. Settlement Fees

The only time Augur levies fees is when market participants are settling with the market contract. Augur levies two fees during settlement: the creator fee, and the reporting fee. Both of these fees are proportional to the amount being paid out. So, in the pre-finalized settlement example above, where Alice receives 0.7 ETH and Bob receives 0.3 ETH, Alice would pay 70% of the fees while Bob would pay 30%.

The creator fee is set by the market creator during market creation, and is paid to the market creator upon settlement. The reporting fee is set dynamically (see Section II C) and is paid to reporters who participate in the reporting process.

2. Settlement of Invalid Markets

In the event that a market resolves as `Invalid`, traders who settle with the market contract receive an equal amount of ETH for shares of each outcome. If the market had N possible outcomes (not including the `Invalid` outcome), and the cost of a complete set of shares was C ETH, then traders will receive C/N ETH for each share settled with the market contract.[16]

3. Reputation Redistribution

If a market finalizes without initiating a fork, all REP staked on any outcome other than the market's final outcome is forfeited and distributed to the users who staked on the market's final outcome in proportion to the amount of REP they staked. The dispute bond sizes are chosen such that anyone who successfully disputes an outcome in favor of the market's final outcome is rewarded with a 50% ROI on their dispute stake.[17] This is a strong incentive for reporters to dispute false tentative outcomes.

II. INCENTIVES AND SECURITY

There is a strong relationship between the market cap of REP and the trustworthiness of Augur's forking protocol. If the market cap of REP is large enough[18], and attackers are economically rational, then the outcome that wins the fork should correspond to objective reality. In fact, it would be possible for Augur to function properly without using designated reporters and dispute rounds. Using *only* the forking process, the oracle would report truthfully.

However, forks are disruptive and time consuming. A fork takes up to 60 days to resolve a single market, and can resolve only one market at a time. During the 60 days in which the forking market is being resolved, all other non-finalized markets are put on hold.[19] Service providers have to update, and REP holders have to migrate their REP to one of the new child universes. Therefore, forks should be used only when they are absolutely necessary. Forking is the nuclear option.

Fortunately, once it has been established that forks can be trusted to determine truth, incentives can be used to encourage participants to behave honestly without having to actually initiate a fork. *It is the credible threat of a fork, and the belief that the fork will resolve correctly, that are the cornerstones of Augur's incentive system.*

Next, we discuss the conditions under which the forking system can be trusted to determine truth. We then discuss the incentive system and how it encourages quick and correct resolution of all markets.

---

[16] Trades cannot simply be unwound if a market resolves as `Invalid` due to technical limitations. Shares of outcomes are just tokens, which can be traded directly between users; the ETH and shares are thus not under Augur's control and cannot be given back to the original owner if the market finalizes as `Invalid`.

[17] See Theorem 3 in Appendix A.

[18] See Section II A for details.

[19] Traders can continue trading on those markets, but those markets cannot finalize until after the forking period.

A. Integrity of the Forking Protocol

Here we discuss the reliability of the forking process and the conditions under which it can be trusted. For ease of discussion, when referring to forks, we will refer to the child universe that corresponds to objective reality as the `True` universe, and any other child universe as a `False` universe. We will refer to the child universe which receives the most REP migration during the forking period as the winning universe and all other child universes as losing universes.

Naturally, we always want the `True` universe to be the winning universe, and the `False` universes to be the losing universes. We say that the forking protocol has been successfully attacked whenever a `False` universe ends up being the winning universe of a fork – thus resulting in the forking market (and, potentially, all non-finalized markets) being paid out incorrectly.

Our approach to securing the oracle is to arrange matters such that the maximum benefit to a successful attacker is less than the minimum cost of performing the attack. We formalize this below.

1. *Maximum Benefit to an Attacker*

An attacker who successfully attacks the oracle would cause all non-finalized Augur markets to migrate to a `False` universe. If the attacker controls the majority of REP in the `False` universe, the attacker can then force all non-finalized markets to resolve however she wants. In the most extreme case, she would also be able to capture all funds escrowed in all of those markets.[20]

Definition 1. We define, and denote by I_a, Augur's *native open interest* as the value of the sum of all funds escrowed in unfinalized Augur markets.[21]

Definition 2. We define a *parasitic market* as any market that does not pay reporting fees to Augur, but does resolve in accordance with the resolution of a native Augur market.

Definition 3. We define, and denote by I_p, the *parasitic open interest* as the value of the sum of all funds escrowed in all parasitic markets that resolve in accordance to non-finalized, native Augur markets.

In the most extreme case, an attacker would also be able to capture all funds in all parasitic markets which resolve in accordance to non-finalized, native Augur markets.

Observation 1. The maximum (gross) benefit to an attacker who successfully attacks the oracle is $I_a + I_p$.

[20] This would require the attacker to capture *all* shares of some given outcome, and then force the market to finalize to that outcome.

[21] This includes external markets that pay reporting fees to Augur.

2. *Parasitic Open Interest is Unknowable*

Augur can accurately and efficiently measure I_a. However, I_p cannot be known in general, as there may exist arbitrarily many offline parasitic markets, each with arbitrarily large open interest. Since the maximum possible benefit to an attacker includes the unknowable quantity I_p, one can never be objectively certain that the oracle is secure against economically rational attackers.

However, if we are willing to assert that I_p is reasonably bounded in practice, then we can define conditions under which we may assert that the oracle is secure.

3. *Minimum Cost of a Successful Attack*

Next, consider the cost of attacking the oracle. Let P denote the price of REP. Let ϵ denote one attorep[22]. Let M denote the total amount of REP in existence (the "money supply" of REP). Let S denote the proportion of M that will be migrated to the True universe during the forking period of a fork.

Thus the product SM represents the absolute amount of REP migrated to the True universe during the forking period of a fork, and the product PM is the market cap of REP.

Let P_f denote the price of REP migrated to a `False` universe of the attacker's choosing. Note that if $P \leq P_f$ then the oracle would not be secure against economically rational attackers, because it would be at least as profitable to migrate REP to the False universe as it would be to not migrate at all.

4. *Integrity*

Assumption 1. *Reporters that are not attackers will never migrate REP to a* `False` *universe during a fork.*[23]

By design, a successful attack on the oracle requires more REP to be migrated to some `False` universe than to the `True` universe during the forking period of a fork. By assumption, only the attacker will migrate REP to a `False` universe. The amount of REP migrated to the `True` universe during the reporting period is denoted by SM. Thus, for an attacker to be successful, they must migrate at least $SM + \epsilon$ REP. For simplicity, we will ignore the negligible ϵ, and say that a successful attack requires migrating at least SM REP, which has a value of SMP before the migration, to some `False` universe.

[22] One *attorep* is 10^{-18} REP.

[23] There may be cases where some non-malicious reporters do migrate REP to a `False` universe accidentally or carelessly. However, such behavior is, in practice, indistinguishable from collaborating with an attacker.

If an attacker migrates SM REP during the reporting period of a fork, they will receive SM REP on the child universe to which they migrate.[24] If the attacker migrates to a `False` universe then the value of those coins becomes SMP_f. Thus the minimum cost to the attacker is $(P - P_f)SM$.

Observation 2. The minimum amount of REP a successful attacker must migrate to a `False` universe during a fork is SM, which costs the attacker $(P - P_f)SM$.

Note that if $S > \frac{1}{2}$ then an attack is *impossible* because there does not exist enough REP outside of the `True` universe for any `False` universe to become the winning universe.

Pitted against economically rational attackers, the oracle will resolve to outcomes that correspond to objective reality if the maximum benefit to an attacker is less than the minimum cost of attack. By observations 1 & 2 we can see that this occurs whenever $S > \frac{1}{2}$ or $I_a + I_p < (P - P_f)SM$. This gives us our formal definition of integrity.

Definition 4. (Integrity Property) The forking protocol has *integrity* whenever $S > \frac{1}{2}$ or whenever $I_a + I_p < (P - P_f)SM$.

The above inequality can be solved for PM to see the relationship between forking protocol integrity and the market cap of REP.

Theorem 1. *(Market Cap Security Theorem) The forking protocol has integrity if and only if:*

1. $S > \frac{1}{2}$, *or*
2. $P_f < P$ *and the market cap of REP is greater than* $\frac{(I_a+I_p)P}{(P-P_f)S}$.

Proof. Suppose the forking protocol has integrity. Then, by definition, $S > \frac{1}{2}$ or $I_a + I_p < (P - P_f)SM$. Suppose $I_a+I_p < (P-P_f)SM$. Since $I_a+I_p \geq 0$ and $SM > 0$, we know that $P_f < P$. Then, solving $I_a + I_p < (P - P_f)SM$ for PM, we see that $\frac{(I_a+I_p)P}{(P-P_f)S} < PM$. Thus the first direction is proved.

Now suppose that $S > \frac{1}{2}$, or that $P_f < P$ and $\frac{(I_a+I_p)P}{(P-P_f)S} < PM$. If $S > \frac{1}{2}$, then the forking protocol has integrity by definition. If $P_f < P$ and $\frac{(I_a+I_p)P}{(P-P_f)S} < PM$, then, solving the inequality for $I_a + I_p$, we see that $I_a + I_p < (P - P_f)SM$, and the forking protocol has integrity. □

[24] In practice, the attacker would receive $1.05SM$ REP in the child universe because of the 5% bonus for migrating within 60-days of the start of a fork. We ignore the 5% bonus here for ease of discussion. To see a discussion that includes the 5% bonus, see Appendix C.

B. Our Assumptions and Their Consequences

We believe traders will not want to trade on Augur in a universe where reporters have lied. We also believe that market creators will not pay to create Augur markets in a universe where there are no traders. In a universe without markets or trading, REP does not provide any dividends to those holding it. Therefore, we believe REP sent to a `False` universe will hold no non-negligible market value and we model this by letting $P_f = 0$.

We think it is reasonable to expect at least 20% of existing REP to be migrated to the True outcome during the reporting period of a fork, and we model this by letting $S \geq \frac{1}{5}$. We are also willing to accommodate parasitic open interest as large as 50% of the native open interest, and so we let $I_a \geq 2I_p$.

Under these assumptions, Theorem 1 tells us that the forking protocol has integrity whenever the market cap of REP is at least 7.5 times the native open interest.[25]

C. Market Cap Nudges

Augur gets information about the price of REP in the same way it gets any other information about the real world: through an Augur market. This gives Augur the ability to compute the current market cap of REP. Augur can also measure the current native open interest, and can thus determine what market cap ought to be targeted in order to meet Augur's integrity requirements.

Every universe begins with a default reporting fee of 1%. If the current market cap is below the target, then reporting fees are automatically increased (but will never be higher than 33.3%), putting upward pressure on the price of REP and/or downward pressure on new native open interest. If the current market cap is above the target, then reporting fees are automatically decreased (but will never be lower than 0.01%) so that traders are not paying more than needed to keep the system secure.

The reporting fees are determined as follows. Let r be the reporting fee from the previous window, let t be the target market cap, and let c be the current market cap. Then the reporting fee for the current fee window is given by $\max\left\{\min\left\{\frac{t}{c}r, \frac{333}{1000}\right\}, \frac{1}{10{,}000}\right\}$.

D. Leveraging the Threat of a Fork

As mentioned above, forks are a disruptive and slow way for markets to reach finalization. Rather than using the forking process to resolve every market, Augur leverages the *threat* of a fork to resolve markets efficiently.

[25] See Appendix B for some alternative assumptions and their consequences.

Recall that any stake successfully disputing an outcome in favor of the market's final outcome will receive a 50% ROI on their dispute stake.[26] In the event of a fork, any REP staked on any of the market's false outcomes should lose all economic value, while any REP staked on the market's true outcome is rewarded with 50% more REP in the child universe that corresponds to the market's true outcome (regardless of the outcome of the fork). Therefore, if pushed to a fork, REP holders who dispute false outcomes in favor of true outcomes will always come out ahead, while REP holders who staked on false outcomes will see their REP lose all economic value.

We believe this situation is sufficient to guarantee that all false tentative outcomes will be successfully disputed.

III. POTENTIAL ISSUES & RISKS

A. Parasitic Markets

Recall that a parasitic market is any market that does not pay reporting fees to Augur, but does resolve in accordance with the resolution of a native Augur market. Because parasitic markets do not have any reporters to pay, they can offer the same service as Augur with lower fees. This can have serious consequences for the integrity of Augur's forking protocol.

In particular, if parasitic markets attract trading interest away from Augur, then Augur's reporters will receive less in reporting fees. This would put downward pressure on the market cap of REP. If the market cap of REP falls too low, the integrity of the forking protocol is put in jeopardy (Theorem 1). As a result, parasitic markets have the potential to threaten the long term viability of Augur, and should be vehemently opposed.

Our best defense against parasitic markets is to make trading on the Augur platform as cheap as possible (while still maintaining the integrity of the oracle), in order to minimize the reward for running a parasitic market.

B. Volatility of Open Interest

Large, sudden, and unexpected increases in open interest – like those that may be seen during a popular sporting event – result in rapid increases in the market cap requirement for forking protocol integrity (Theorem 1). When the market cap requirement exceeds the market cap, there is a risk of economically rational attackers causing a fork to resolve incorrectly. While Augur does attempt to nudge the market cap upwards during such situations (see Section II C), these nudges are reactionary and are adjusted only once per 7-day fee window.

It is worth noting, however, that speculators who witness the sudden increase in open interest may buy REP in anticipation of the reactionary market cap nudge, thus driving the market cap of REP up, perhaps to a point where the integrity of the forking protocol is no longer threatened. So the length of time during which the oracle is vulnerable may not be long enough for an attacker to successfully exploit the vulnerability.

C. Inconsistent or Malicious Resolution Sources

During market creation, market creators chose a resolution source that reporters should use to determine the outcome of the event in question. If the market creator chooses an inconsistent or malicious resolution source, honest reporters may lose money.

For example, suppose the market in question has outcomes `A` and `B`, and the market creator, Serena, has chosen her own website, `attacker.com`, as the resolution source. After the market's event end time, Serena – who is also the designated reporter for the market – reports outcome `A`, and updates `attacker.com` to indicate that outcome `B` is the correct outcome. Honest reporters who check `attacker.com` will see that the initial report is incorrect and, during the first dispute round, should successfully dispute the tentative outcome in favor of outcome `B`. Serena would update `attacker.com` to indicate that outcome `A` is the correct outcome, and the market would then enter its second dispute round. Again, reporters who check `attacker.com` will see that the tentative outcome (outcome `B`) is incorrect, and may successfully dispute it. Serena can repeat this behavior until the market resolves. No matter how the market resolves, some honest reporters will lose money.

Several variations of this attack exist. Simply ignoring markets with dubious resolution sources is not sufficient, for in the event that such a market causes a fork, all REP holders will have to choose a child universe to which to migrate their REP. Reporters should remain vigilant against markets with dubious resolution sources. Such markets should be publicly identified so reporters can coordinate to make sure such markets finalize as invalid.

D. Self-Referential Oracle Queries

Markets that trade on the future behavior of Augur's oracle may have undesirable effects on the behavior of the oracle itself [11]. For example, consider a market that trades on the question, "Will any designated reporter fail to submit a report during their three-day reporting period before December 31, 2018?" Bets placed on the `No` outcome of this market may act as a perverse incentive for designated reporters to intentionally fail to report. If a designated reporter can buy up enough `Yes` shares

[26] Measured in REP that exists in a universe that corresponds to the market's final outcome; see Theorem 3 in Appendix A.

at a low enough price to compensate for the loss of the no-show bond, they may intentionally fail to report.

If the market cap of REP is large enough (Theorem 1) then these self-referential oracle queries will not threaten the integrity of the forking protocol. However, they may negatively affect the performance of Augur by causing delays in market finalizations. While markets would still finalize correctly, this sort of behavior is disruptive and undesirable.

E. Uncertain Fork Participation

We cannot know in advance how much REP will be migrated to the `True` universe during the forking period of a fork, thus we cannot know in advance whether the market cap is large enough for the oracle to have integrity (Theorem 1). Our belief in the integrity of the forking protocol can be no stronger than our belief in our assumptions about the lower bound on honest participation during a forking period. We assume that at least 20% of all REP will migrate to the `True` child universe during the forking period of a fork, but we cannot guarantee this.

Augur forks differ from blockchain forks in one important respect: after a blockchain fork, a user who owned a coin on the parent chain will now own a coin on both forks. Ignoring replay attacks, blockchain forks pose little risk to users. After an Augur fork, however, a user who owns a REP token in the parent universe can migrate that coin to only one of the child universes. If the user migrates their token to any universe other than the consensus universe, their token may lose all value. Thus migrating REP during the forking period of a fork, before it is clear which child universe has achieved consensus, exposes the user to risk. That risk may discourage participation during the forking period of contentious forks.

In an effort to compensate for this risk and encourage participation during forking periods, all token holders who migrate their REP within 60 days of the start of a fork will receive 5% additional REP in the child universe to which they migrated (see Section I C 9). However, we cannot know in advance whether this 5% bonus will be enough to compensate for the risk and incentivize participation during a forking period.

F. Ambiguous or Subjective Markets

Only events that have objectively knowable outcomes are suitable for use in Augur markets. If reporters believe that a market is not suitable for resolution by the platform – for example, because it is ambiguous, subjective, or the outcome is not known by the event end date – they should report the market as `Invalid`. If a market resolves as `Invalid`, traders are paid out at equal values for all possible outcomes; for scalar markets, traders are paid out halfway between the market's minimum price and maximum price.

It is possible to imagine markets where some reporters are certain that the outcome is `A` and others are certain that the outcome is `B`. For example, in 2006, TradeSports allowed its users to speculate on whether North Korea would fire a ballistic missile that would land outside of its airspace before the end of July 2006. On July 5, 2006, North Korea successfully fired a ballistic missile that landed outside of its airspace, and the event was widely reported by the world media and confirmed by many U.S. government sources. However, the U.S. Department of Defense had not confirmed the event, as was required by TradeSports' contract. TradeSports concluded that the contract's conditions had not been met, and paid out accordingly.[27]

This is a case where the spirit of the market – to predict the missile launch – was clearly satisfied, but the letter of the market – to predict whether the U.S. Department of Defense would confirm the launch – was not. TradeSports, being a centralized website, was able to unilaterally declare the outcome of the market. If such a situation arises in an Augur market, REP holders may have differing opinions about how the market should resolve, and stake their REP accordingly. In the worst case, this could result in a fork where REP in more than one child universe maintains a non-zero market value.

ACKNOWLEDGMENTS

We thank Abraham Othman, Alex Chapman, Serena Randolph, Tom Haile, George Hotz, Scott Bigelow, and Peronet Despeignes for their helpful feedback and suggestions.

[1] J. Wolfers and E. Zitzewitz. Prediction markets. *Journal of Economic Perspectives*, 18(2):107–126, 2004.
[2] James Surowiecki. *The Wisdom of Crowds*. Anchor, 2005.
[3] R. Hanson, R. Oprea, and D. Porter. Information aggregation and manipulation in an experimental market. *Journal of Economic Behavior & Organization*, 60(4):449–459, 2006.
[4] D.M. Pennock, S. Lawrence, C.L. Giles, and F.A. Nielsen. The real power of artificial markets. *Science*, 291:987–988, 2001.
[5] C. Manski. Interpreting the predictions of prediction markets. *NBER Working Paper No. 10359*, 2004.
[6] J. Wolfers and E. Zitzewitz. Interpreting prediction market prices as probabilities. *NBER Working Paper No. 10359*, 2005.

[27] See `https://en.wikipedia.org/wiki/Intrade#Disputes` for details.

[7] S. Goel, D.M. Reeves, D.J. Watts, and D.M. Pennock. Prediction without markets. In *Proceedings of the 11th ACM Conference on Electronic Commerce*, EC '10, pages 357–366. ACM, 2010.

[8] S. Nakamoto. Bitcoin: a peer-to-peer electronic cash system. *https://bitcoin.org/bitcoin.pdf*, 2008.

[9] V. Buterin. A next generation smart contract and decentralized application platform. *https://github.com/ethereum/wiki/wiki/White-Paper*, 2013.

[10] J. Clark, J. Bonneau, E.W. Felten, J.A. Kroll, A. Miller, and A. Narayanan. On decentralizing prediction markets and order books. In *WEIS '14: Proceedings of the 10th Workshop on the Economics of Information Security*, June 2014.

[11] A. Othman and T. Sandholm. Decision rules and decision markets. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Volume 1 - Volume 1*, AAMAS '10, pages 625–632. International Foundation for Autonomous Agents and Multiagent Systems, 2010.

[12] J. Peterson and J. Krug. Augur: a decentralized, open-source platform for prediction markets. *arXiv:1501.01042v1 [cs.CR]*, 11 2014.

Appendix A: Finalization Time & Redistribution

We begin with some notation, definitions, and observations.

Definition 5. For a given market M, let Ω_M be the outcome space (or set of outcomes) of M.

Definition 6. For $n \geq 1$ and $\omega \in \Omega_M$, let $S(\omega, n)$ denote the total amount of stake on outcome ω at the beginning of dispute round n. This includes all stake from all successful dispute bonds in favor of ω over all previous dispute rounds.

Definition 7. For $n \geq 1$ and $\omega \in \Omega_M$, let $S(\overline{\omega}, n)$ denote the amount of stake on all outcomes in Ω_M *except for* ω at the beginning of dispute round n:

$$S(\overline{\omega}, n) = \sum_{\substack{\gamma \in \Omega_M \\ \gamma \neq \omega}} S(\gamma, n)$$

Definition 8. For $n \geq 0$, let A_n denote the total stake over all outcomes M at the beginning of dispute round n:

$$A_n = \sum_{\omega \in \Omega_M} S(\omega, n)$$

Observation 3. It follows that $A_n - S(\omega, n) = S(\overline{\omega}, n)$.

Definition 9. For $n \geq 1$, let $\hat{\omega}_n$ denote the tentative outcome at the beginning of dispute round n. For example, $\hat{\omega}_1$ is the outcome reported by the initial reporter.

Definition 10. For $n \geq 1$ and $\omega \neq \hat{\omega}_n$, let $B(\omega, n)$ denote the amount of stake required to successfully fill a dispute bond in favor of outcome ω during dispute round n.

Recall that the amount of stake required to successfully fill a dispute bond in favor of outcome ω during dispute round n, where $\omega \neq \hat{\omega}_n$ is given by Eq. 1, $B(\omega, n) = 2A_n - 3S(\omega, n)$.

Observation 4. If a dispute bond is successfully filled in favor of outcome ω during dispute round n, then $S(\omega, n+1) = B(\omega, n) + S(\omega, n)$. That is, the successful dispute stake is the only new stake applied to outcome ω at the end of dispute round n.

Observation 5. For all $\omega \neq \hat{\omega}_n, S(\omega, n-1) = S(\omega, n)$. That is, if a dispute bond is not entirely filled in favor of outcome ω, then no additional stake is added to outcome ω at the beginning of the next dispute round. This is due to the fact that all unsuccessful dispute stake is returned to the users at the end of the dispute round.

Observation 6. For all $n \geq 2, A_n = A_{n-1} + B(\hat{\omega}_n, n-1)$. That is, the total stake over all outcomes at the beginning of a dispute round is simply the total stake from the beginning of the previous dispute round plus the successful dispute stake from the previous dispute round. All other stake is returned to users at the end of the previous dispute round.

Lemma 2. *$S(\hat{\omega}_n, n) = 2S(\overline{\hat{\omega}_n}, n)$, for $n \geq 2$.*

Proof. Suppose a market enters dispute round n, where $n \geq 2$. During dispute round $n-1$, the outcome $\hat{\omega}_{n-1}$ must have been successfully disputed in favor of outcome $\hat{\omega}_n$. According to Eq. 1, the size of that dispute bond is $B(\hat{\omega}_n, n-1) = 2A_{n-1} - 3S(\hat{\omega}_n, n-1)$. Using observation 3, this can be rewritten as

$$B(\hat{\omega}_n, n-1) + S(\hat{\omega}_n, n-1) = 2S(\overline{\hat{\omega}_n}, n-1) \qquad \text{(A1)}$$

We know the dispute bond was successfully filled during round $n-1$. Using observation 4, we see that $B(\hat{\omega}_n, n-1) + S(\hat{\omega}_n, n-1) = S(\hat{\omega}_n, n)$. Observation 5 tells us that the total amount staked on $\overline{\hat{\omega}_n}$ is unchanged from round $n-1$ to n, $2S(\overline{\hat{\omega}_n}, n-1) = 2S(\overline{\hat{\omega}_n}, n)$. Thus, Eq. A1 reduces to $S(\hat{\omega}_n, n) = 2S(\overline{\hat{\omega}_n}, n)$. □

Theorem 3. *Any REP holders successfully disputing an outcome in favor of a market's final outcome will receive a 50% ROI on their dispute stake (measured in REP that exists in a universe that corresponds to the market's final outcome), unless the market is interrupted by some other market causing a fork.*

Proof. During a fork, all users who successfully filled dispute bonds in favor of the forking market's final outcome are given (via coins minted during the fork) a 50% return on their dispute stake when they migrate their dispute stake to the corresponding child universe. Thus, in the case where the market in question has caused a fork, the theorem is immediately true.

Now consider the case where the market in question resolves without causing a fork, and reporting is not interrupted by some other market causing a fork.

Denote the market's final outcome by ω_{Final} and suppose the market resolves at the end of reporting round n, where $n \geq 2$. That means the tentative outcome for round n is ω_{Final}, and that outcome is not successfully disputed during round n. In other words: $\hat{\omega}_n = \omega_{\text{Final}}$. Then by Lemma 2 we know that $S(\omega_{\text{Final}}, n) = 2S(\overline{\omega_{\text{Final}}}, n)$.

Since the market resolves at the end of round n with no further stake added to any outcome, the above equation shows the final amount of stake on the market's final outcome, ω_{Final}, and the sum of all stake on all of the market's other outcomes, $\overline{\omega_{\text{Final}}}$. Note that there is exactly twice as much stake on the market's final outcome as there is on all other outcomes combined.

Augur redistributes all stake on the non-final outcomes to users who staked on ω_{Final}, in proportion to the amount of REP they staked. Therefore the users who successfully filled a dispute bond in favor of ω_{Final} get a 50% ROI on their staked REP. □

Next, consider the maximum number of dispute rounds required to resolve a market. Eq. 1 is minimized when ω is chosen to be the non-tentative outcome that begins the dispute round with the greatest amount of stake. Lemma

2 implies that the non-tentative outcome with the greatest amount of stake is the previous dispute round's tentative outcome. Therefore, the smallest possible dispute bond size that can be successfully filled during dispute round n, where $n \geq 2$, is $B(\hat{\omega}_{n-1}, n)$.

In other words, the dispute bond size grows *slowest* when the same two outcomes are repeatedly disputed in favor of one another. It follows that the number of dispute rounds required for a market to initiate a fork is *maximized* when the same two outcomes are repeatedly disputed in favor of one another. Therefore we can determine the maximum number of dispute rounds that any market may undergo before initiating a fork by finding the maximum number of dispute rounds that can occur in the particular case where the same two market outcomes are repeatedly disputed in favor of on another. We examine that case now.

Suppose that every successful dispute bond is filled in favor of the previous dispute round's tentative outcome. Then the two tentative outcomes that are iteratively disputed in favor of one another other are $\hat{\omega}_1$ and $\hat{\omega}_2$.

Observation 7. In the case where the same two tentative outcomes are repeatedly disputed in favor of one another, $\hat{\omega}_n = \hat{\omega}_{n-2}$ for all $n \geq 3$.

Definition 11. Let d denote the amount of stake placed on $\hat{\omega}_1$ during the initial report. Because the tentative outcome for each round is known in this situation, we can simplify our notation for the dispute bond sizes. Define a shorthand B_n to denote the bond size required for round n, so that $B_1 = 2d$ and $B_n = B(\hat{\omega}_{n-1}, n)$ for all $n \geq 2$. This will make for easier reading and comprehension.

Observation 8. In the case where the same two tentative outcomes are repeatedly disputed in favor of one another, $S(\hat{\omega}_{n-1}, n) = S(\hat{\omega}_{n-1}, n-2) + B_{n-2}$ for $n \geq 3$. (That is, every other successful dispute bond is added to the same outcome.)

Lemma 4. *If the same two tentative outcomes are repeatedly disputed in favor of one another, then for all n where $n \geq 3$:*

1. $S(\hat{\omega}_{n-1}, n) = \frac{2}{3}B_{n-1}$
2. $A_n = 2B_{n-1}$ *and*
3. $B_n = 3d2^{n-2}$

Proof. (By induction on n)

Suppose the same two tentative outcomes are repeatedly disputed in favor of one another.

(Base Case) By definition and Eq. 1 we make the following observations.

- $S(\hat{\omega}_1, 1) = d$, $S(\hat{\omega}_2, 1) = 0$, $A_1 = d$, and $B_1 = 2d$
- $S(\hat{\omega}_1, 2) = d$, $S(\hat{\omega}_2, 2) = 2d$, $A_2 = 3d$, and $B_2 = 3d$
- $S(\hat{\omega}_1, 3) = 4d$, $S(\hat{\omega}_2, 3) = 2d$, $A_3 = 6d$, and $B_3 = 6d$

$S(\hat{\omega}_{3-1}, 3) = S(\hat{\omega}_2, 3) = 2d = \frac{2}{3}(3d) = \frac{2}{3}B_2 = \frac{2}{3}B_{3-1}$, so part 1 of the lemma holds for $n = 3$.

$A_3 = 6d = 2(3d) = 2B_2 = 2B_{3-1}$, so part 2 of the lemma holds for $n = 3$.

$B_3 = 6d = 3d2^{3-2}$, so part 3 of the lemma holds for $n = 3$.

Therefore the lemma, in its entirety, holds true for the base case of $n = 3$.

(Induction) Suppose the lemma is true for all n such that $3 \leq n \leq k$. We want to show that the lemma holds for $n = k + 1$. That is, we want to show that:

(a) $S(\hat{\omega}_k, k+1) = \frac{2}{3}B_k$

(b) $A_{k+1} = 2B_k$ and

(c) $B_{k+1} = 3d2^{k-1}$

First, we prove part (a). By observation 8:

$$S(\hat{\omega}_k, k+1) = S(\hat{\omega}_k, k-1) + B_{k-1}$$

By observation 7 we can rewrite the above as:

$$S(\hat{\omega}_{k-2}, k+1) = S(\hat{\omega}_{k-2}, k-1) + B_{k-1}$$

By the induction hypothesis, we can rewrite $S(\hat{\omega}_{k-2}, k-1)$ as $\frac{2}{3}B_{k-2}$ on the right-hand side to get:

$$S(\hat{\omega}_{k-2}, k+1) = \frac{2}{3}B_{k-2} + B_{k-1}$$

By the induction hypothesis, we can write B_{k-2} as $3d2^{k-4}$ and B_{k-1} as $3d2^{k-3}$:

$$S(\hat{\omega}_{k-2}, k+1) = d2^{k-1}$$

Applying observation 7 to the left-hand side we get:

$$S(\hat{\omega}_k, k+1) = d2^{k-1}$$

Finally, note that by the above equation and the induction hypothesis, $S(\hat{\omega}_k, k+1) = d2^{k-1} = \frac{2}{3}(3d2^{k-2}) = \frac{2}{3}B_k$. This proves part (a).

Next, we prove part (b). By observation 6:

$$A_{k+1} = A_k + B_k$$

By the induction hypothesis, $A_k = 2B_{k-1}$:

$$A_{k+1} = 2B_{k-1} + B_k$$

By the induction hypothesis, $B_{k-1} = 3d2^{k-3}$, so the right-hand side can be simplified to

$$A_{k+1} = 3d2^{k-2} + B_k$$

By the induction hypothesis, $B_k = 3d2^{k-2}$ to rewrite the right-hand side as

$$A_{k+1} = 2B_k,$$

and part (b) is proved.

Finally, we prove part (c). By Eq. 1:

$$B_{k+1} = 2A_{k+1} - 3S(\hat{\omega}_k, k+1)$$

By observation 8, we can write $S(\hat{\omega}_k, k+1)$ as $S(\hat{\omega}_k, k-1) + B_{k-1}$:

$$B_{k+1} = 2A_{k+1} - 3\left(S(\hat{\omega}_k, k-1) + B_{k-1}\right)$$

By observation 7, $\hat{\omega}_k = \hat{\omega}_{k-2}$:

$$B_{k+1} = 2A_{k+1} - 3\left(S(\hat{\omega}_{k-2}, k-1) + B_{k-1}\right)$$

By observation 6, $A_{k+1} = A_k + B_k$:

$$B_{k+1} = 2\left(A_k + B_k\right) - 3\left(S(\hat{\omega}_{k-2}, k-1) + B_{k-1}\right)$$

By the induction hypothesis, $A_k = 2B_{k-1}$ and $S(\hat{\omega}_{k-2}, k-1) = \frac{2}{3}B_{k-2}$:

$$B_{k+1} = 2\left(2B_{k-1} + B_k\right) - 3\left(\tfrac{2}{3}B_{k-2} + B_{k-1}\right)$$

By the induction hypothesis, $B_k = 3d2^{k-2}$, $B_{k-1} = 3d2^{k-3}$ and $B_{k-2} = 3d2^{k-4}$. Making these substitutions and simplifying yields:

$$B_{k+1} = 3d2^{k-1}$$

This proves part (c), and concludes the proof of the lemma. □

Theorem 5. *If not interrupted by some other market causing a fork, a given market may undergo at most 20 dispute rounds before finalizing or causing a fork.*

Proof. Suppose that a given market is not interrupted by some other market causing a fork. Then, as shown above, we know that the number of dispute rounds required for a market to initiate a fork is maximized when the same two outcomes are repeatedly disputed in favor of one another. Part 3 of Lemma 4 tells us that, in this situation, the dispute bond size required for successfully disputing the tentative outcome during round n is given by $3d2^{n-2}$, where d is the amount of the stake placed during the initial report.

We know that forks are initiated after the successful fulfillment of a dispute bond with size at least 2.5% of all existing REP, and we know that there are 11 million REP in existence. Thus a fork is initiated when a dispute bond of size 275,000 REP is filled. We also know that $d \geq 0.35$ REP, because the minimum amount of stake on the initial report is 0.35 REP[28].

Solving $3(0.35)2^{n-2} > 275,000$ for $n \in \mathbb{Z}$ yields $n \geq 20$. Thus, we can guarantee that a market will resolve or cause a fork after at most 20 dispute rounds. □

[28] See appendixes E 2 and E 3

Appendix B: Alternative Assumptions & Consequences

Recall that:

- S is the proportion of total REP that is migrated to the `True` universe during the forking period
- P is the price of REP in the `True` universe
- P_f is the price of REP that has been migrated to a `False` universe of the attacker's choosing
- I_a is Augur's native open interest
- I_p is the parasitic open interest

Augur makes certain assumptions about S, P_f, and I_p in order to arrive at a target market cap. In particular, Augur assumes that at least 20% of all REP will be migrated to the `True` universe during the forking period of a fork, REP migrated to a `False` universe will have no non-negligible value, and parasitic open interest will be at most half of the native open interest. In other words: $S \geq 0.2$, $P_f = 0$, and $I_a \geq 2I_p$. Under these assumptions, Theorem 1 tells us that the forking protocol has integrity whenever the market cap of REP is greater than 7.5 times the native open interest.

You can make your own assumptions about S, P_f, and I_p to arrive at your own conclusions about how large the market cap needs to be for the oracle to have integrity in practice. We list some alternative scenarios here for your convenience.

Scenario 1. More than 50% of existing REP migrates to the `True` universe during the forking period. In this case P_f and I_p do not matter at all. Since $S > \frac{1}{2}$, the forking protocol has integrity no matter what the market cap happens to be. There would not exist enough remaining REP on the market for an attacker to be successful.

Scenario 2. 48% of existing REP migrates to the `True` universe during the forking period, no parasitic markets exist, and REP sent to a `False` universe has no value. In this case $S = 0.48$, $I_p = 0$, and $P_f = 0$. Under these assumptions, the market cap of REP must be greater than about twice the native open interest for the forking protocol to have integrity.

Scenario 3. 20% of existing REP migrates to the `True` universe during the forking period, parasitic open interest is equal to native open interest, and REP migrated to a `False` universe trades at 5% of the value of REP migrated to the `True` universe. In this case $S = 0.2$, $I_p = I_a$, and $P_f = 0.05P$. Under these assumptions, the market cap of REP must be greater than about 10.5 times the native open interest for the forking protocol to have integrity.

Scenario 4. Only 5% of existing REP migrates to the `True` universe during the forking period, parasitic interest is twice as large as native open interest, and REP sent to

a `False` universe trades at 5% of the value of REP sent to the `True` universe. In this case $S = 0.05$, $I_p = 2I_a$, and $P_f = 0.05P$. Under these assumptions, the market cap of REP must be greater than about 63 times the native open interest for the forking protocol to have integrity.

Appendix C: The Effect of the Early Migration Bonus on the Integrity of the Forking Protocol

For ease of discussion, we ignored the 5% early migration bonus and a small term when discussing the integrity of the forking protocol. Here we revisit Theorem 1 taking those two things into consideration.

As before, the amount of REP sent to the `True` universe during the reporting period is denoted by SM. Thus for an attacker to be successful, they must migrate at least $SM + \epsilon$ REP, which has a value of $(SM + \epsilon)P$ before migration, to some `False` universe.

If an attacker migrates $SM+\epsilon$ REP to a `False` universe during the reporting period of a fork, they will receive $1.05(SM + \epsilon)$ REP on the child universe to which they migrated. By definition of P_f, the value of those coins is given by $1.05(SM + \epsilon)P_f$. Thus the minimum cost to the attacker is $(SM + \epsilon)P - 1.05(SM + \epsilon)P_f$, which can be expressed as $(SM + \epsilon)(P - 1.05Pf)$.

As before, the maximum (gross) benefit to an attacker is given by $I_a + I_p$. Thus we would say the forking protocol has integrity whenever $S > \frac{1}{2}$ or:

$$I_a + I_p < (SM + \epsilon)(P - 1.05P_f) \tag{C1}$$

Solving the above inequality for the market cap, PM, we can see that the forking protocol has integrity if and only if:

1. $S > \frac{1}{2}$ or
2. $1.05P_f < P$ and the market cap of REP is greater than $\frac{P(I_a+I_p-\epsilon(P-1.05P_f))}{S(P-1.05P_f)}$

As we can see, the effect of the early migration bonus on the market cap requirement is very small.

Appendix D: The Effect of the Early Migration Bonus on the Minimum Cost of a Fork

To encourage greater participation during a fork, all token holders who migrate their REP within 60 days of the start of a fork will receive 5% additional REP in the child universe to which they migrated. This reward is paid for via currency inflation.

This bonus can become a perverse incentive if the cost of initiating a fork is too low. In particular, if an attacker can gain more value from the 5% REP bonus than she would lose by initiating a fork, then we would expect forks to happen as often as possible. This attack, which we refer to as the *inflation milking attack*, would not result in the oracle reporting incorrectly, but it would result in disruptive forks happening often.

To prevent this behavior Augur needs to make sure that the cost of initiating a fork is greater than the maximum value that can be gained from the 5% inflation bonus. Here, we derive a lower bound on cost of initiating a fork in order to prevent this perverse incentive.

Let P_0 denote the price of REP before the fork and P_1 denote the price of REP after the fork. Let M_0 denote the money supply before the fork and M_1 denote the money supply after the fork. Let S denote the proportion of M_0 migrating to the `True` universe during the forking period of the fork. Let b denote the amount of REP that must be economically burned (that is, staked on a `False` outcome) in order to initiate a fork. We assume $b > 1$.

For the purposes of this section, we make the conservative assumption that all REP that migrates during the forking period is controlled by the attacker. We further assume (because it minimizes the cost of this attack this attack) that all REP that migrates during the forking period is migrated to the `True` universe.

With this notation, SM_0 is the amount of REP migrated during the forking period, while $(1 - S)M_0$ is the amount of REP *not* migrated during the forking period.

$$M_0 = SM_0 + (1 - S)M_0 \tag{D1}$$

When a total of SM_0 REP is migrated during the forking period, a total of $0.05SM_0$ REP is created via inflation:

$$M_1 = 1.05SM_0 + (1 - S)M_0 \tag{D2}$$

Focusing only on the effects of inflation, and for the sake of simplicity, we are assuming that the market cap after the fork will be the same as the market cap before the fork[29]:

$$P_0M_0 = P_1M_1 \tag{D3}$$

Substituting D1 and D2 into D3 and simplifying gives us:

$$P_1 = \frac{20P_0}{20 + S} \tag{D4}$$

The (gross) benefit to the attacker for initiating a fork and taking advantage of the early migration bonus is the value of her migrated REP after migration minus the value of her migrated REP before migration:

$$1.05SM_0P_1 - SM_0P_0 \tag{D5}$$

Substituting D4 into D5 we get an alternative expression for the (gross) benefit to the attacker:

$$1.05SM_0\frac{20P_0}{20 + S} - SM_0P_0 \tag{D6}$$

[29] We think this is conservative. In practice, we expect the market cap to decrease after a fork.

Recall that b is the amount of REP that must be economically burned in order to initiate a fork. Thus, the cost of initiating a fork is bP_0. Therefore, paying the cost of initiating a fork in order to take advantage of the early migration bonus is worthwhile whenever the following inequality is satisfied:

$$0 < 1.05SM_0\frac{20P_0}{20+S} - SM_0P_0 - bP_0 \tag{D7}$$

Observing that $P_0 > 0$, and $S \neq -20$, we solve for b and see that the attack is profitable when:

$$b < \frac{21M_0S}{S+20} - M_0S \tag{D8}$$

In order to prevent the perverse incentive, Augur must arrange matters such that:

$$b \geq \frac{21M_0S}{S+20} - M_0S \tag{D9}$$

Noting that S is restricted to the interval $[0,1]$, we see that the value of the right-hand side of inequality D9 is maximized when $S = 2\sqrt{105} - 20 \approx 0.4939$. That is, this attack is most profitable for the attacker when about 49.39% of all existing REP is migrated during the forking period. Being conservative, we use this value for S.[30]

Substituting $S = 0.4939$ into D9 we get $b \geq 0.012197M_0$. Therefore, if the cost to initiate a fork is at least 1.2197% of existing REP then the inflation milking attack is not profitable.

Recall that a fork is initiated only after a successful dispute bond is filled that is greater than 2.5% of existing REP. Suppose that such a dispute bond were filled in favor of outcome ω and a fork were initiated. Outcome ω is either true or false.

If outcome ω is false, then at least 2.5% of existing REP was staked on a false outcome, and thus economically burned. So inflation milking is not profitable when ω is false.

If outcome ω is true, then Lemma 2 tells us that at least 1.25% of existing REP (in total) is staked on false outcomes, and thus economically burned. So inflation milking is also not profitable when ω is true.

It is for this reason that fork initiation requires successfully filling a dispute bond that is at least 2.5% of existing REP.

Appendix E: Bond Size Adjustments

The validity bond, the no-show REP bond, and the designated reporter stake are dynamically adjusted based on the behavior of participants during the previous fee window. Here we describe how we adjust those values.

We define the function $f : [0,1] \to [\frac{1}{2}, 2]$ by:[31]

$$f(x) = \begin{cases} \frac{100}{99}x + \frac{98}{99} & \text{for} \quad x > \frac{1}{100} \\ 50x + \frac{1}{2} & \text{for} \quad x \leq \frac{1}{100} \end{cases} \tag{E1}$$

The function f is used to determine the multiple used in these adjustments, as described in the subsections below. In brief, if the undesirable behavior occurred exactly 1% of the time during the previous fee window, then the bond size remains the same. If it was less frequent, then the bond size will be reduced by as much as half. If it was more frequent, then the bond size will be increased by as much as a factor of 2.

1. Validity Bond

During the very first fee window after launch, the validity bond will be set at 0.01 ETH. Then, if more than 1% of the finalized markets in the previous fee window were invalid, the validity bond will be increased. If less than 1% of the finalized markets in the previous fee window were invalid, then the validity bond will be decreased (but will never be lower than 0.01 ETH).

In particular, we let ν be the proportion of finalized markets in the previous fee window that were invalid, and b_v be the amount of the validity bond from the previous fee window. Then the validity bond for the current window is $\max\left\{\frac{1}{100}, b_v f(\nu)\right\}$.

2. No-Show REP Bond

During the very first fee window after launch, the no-show REP bond will be set at 0.35 REP. As with the validity bond, the no-show REP bond is adjusted up or down, targeting a 1% no-show rate with a floor of 0.35 REP.

Specifically, we let ρ be the proportion of markets in the previous fee window whose designated reporters failed to report on time, and we let b_r be amount of the no-show REP bond from the previous fee window. The the amount of the no-show REP bond for the current fee window is $\max\{0.35, b_r f(\rho)\}$.

3. Designated Reporter Stake

During the very first fee window after launch, the amount of the designated reporter stake will be set at 0.35 REP. The amount of the designated reporter stake

[30] In practice, the attacker cannot prevent other participants from migrating their own REP during the forking period, and thus cannot guarantee that S would not exceed her ideal value of about 0.4939. However, since we are defending against the worst case scenario, we use $S = 0.4939$.

[31] This formula may change once empirical data from live markets is obtained.

is dynamically adjusted according to how many designated reports were incorrect (failed to concur with the final market outcome) during the previous fee window.

In particular, we let δ be the proportion of designated reports that were incorrect during the previous fee window, and we let b_d be the amount of the designated reporter stake during the previous fee window, then the amount of the designated reporter stake for the current window is $\max\{0.35, b_d f(\delta)\}$.

Appendix F: Design Changes

We arrived at the current design of Augur after three years of research and iteration. The design that emerged from this process differs substantially from the vision laid out in our old whitepaper [12]. Here, we discuss three significant changes as well as the rationale for the changes.

1. Reporting Fees

In the old design, the market creator would set a trading fee which would be split 50/50 with reporters. In the current design, the fees for the market creator and the reporters are independent, and the reporters' fees are tuned dynamically by Augur itself to keep the system secure.

The fees paid to reporters impact the price of REP, which has a direct effect on the security of the forking protocol (Theorem 1). If the fees paid to reporters are too low, then the integrity of the oracle is at risk. If the fees paid to reporters are too high, then the threat of parasitic markets increases. Thus, it is important that the fees paid to reporters be adjusted dynamically to maintain Augur's security, rather than being decided arbitrarily by market creators.

Decoupling reporters' fees from the choices of market creators also ensures that reporters (and thus, forking protocol integrity) are not harmed by competition among market creators to create markets with the lowest fees. Quality markets and quality reporting should be measured and rewarded separately. Competition should be allowed to drive market creator fees towards zero, without dragging the fees paid to reporters down as well.

2. Trading Fees

In the old design, fees were collected from traders on every trade. In the new design, fees are collected from traders only when settling directly with market contracts. This change was made, in part, because Augur cannot police offline trading. Shares of market outcomes are simply tokens, which can be traded freely between users. Since collecting fees on every trade is infeasible, Augur instead collects fees only when traders settle directly with the Augur market contracts. An added benefit of this approach is that it reduces the average fees paid by traders, which should make Augur more competitive.

3. Universes

In the old design, there was only one "version" of REP, and its total supply was fixed. In the current design, REP can fork into many different versions (universes), each of which can end up with more or less total REP than the original version. If a fork is contentious, the REP supply in each child universe might be only a fraction of the total supply in the parent universe. In a non-contentious fork, the early migration bonus to fork participants could result in a child universe that has more total REP than its parent universe.

The new versions of REP spawned by a fork are all different tokens, each with its own price and total supply, and service providers should treat them as such. When Augur first launches, there will be a single universe (the genesis universe) and a single version of REP, just as it exists now. However, as soon as a fork occurs, the single version of REP will split into many versions: for example, a forking market with outcomes `A` and `B` would spawn new tokens REP-`A`, REP-`B`, and REP-`Invalid`. Wallets and exchanges that support REP would now have four different versions of REP which they could (in theory) support – REP-`genesis` (the original version of REP, which would now be locked), REP-`A`, REP-`B`, and REP-`Invalid`.[32]

The total supply of REP in each child universe depends on how much REP migrated to it, and when that migration occurred. Migrating REP during a fork, before it is clear which child universe has achieved consensus, exposes the user to a small (but non-zero) amount of risk (see Section III E), which may discourage participation during the forking period of contentious forks. In order to encourage participation during a fork, users must be compensated for the risk.

Users who do not participate during the forking period of a fork could be penalized by losing some portion of their REP holdings. In fact, the old design had a "use it or lose it" mechanism that penalized non-participants as if they were reporters who reported incorrectly. However, punishing users who do not participate creates significant usability problems. Punishing users who do not participate is problematic for wallets and exchanges who are the custodians of their customers' REP. In the event of a fork, exchanges would need to migrate their customer's REP to some child universe during the forking period, or lose some portion of their REP holdings.[33]

---

[32] As a practical matter, service providers may find it easiest (and least disruptive to their users) to encourage their users to participate in the fork, and then to simply support the winning universe once the fork has resolved.

[33] We also found, as a practical matter, that the smart contract code needed to implement forking rewards only using redistribution was

Instead of penalizing non-participants, fork participants who migrate during the forking period are rewarded by minting a 5% bonus in the child universe to which they are migrating. If 4.762% of REP (or more) migrates to a losing universe – of which 1.25% to 2.5% has already been committed as dispute stake – then all universes will have a smaller total supply of REP than the parent universe.

inordinately complex. Contract code complexity is itself a security risk, so we have tried to simplify the implementation wherever possible.