Faythe Chain

Abstract

The Faythe blockchain provides arbitrators to facilitate decision making and reward distribution involving real-life actors. So far, the utility of chain based contracts has been restricted so far to on-chain activities - involving an off-chain party relinquished the proof of security and reliability and has not been feasible before. A provided theoretical solution is the Trust protocol, which assumes that an ultimately trusted oracle can interface with the chain. Historically the Trust protocol has been an elusive target because attempts for implementation assumed the existence of ultimately reliable oracles. We at Faythe believe that new AI-based fraud detection mechanisms coupled with the distributed ledger of a blockchain are enough to provide a platform that can handle contractual tasks without the involvement of a perfect oracle. In this paper, we present Faythe Chain, a decentralized Oracle Network. We'll define how we envision the use of real-world oracles and actuators to ultimately create a bridge between the analog and digital world through the execution abilities and temper proof ledger of the blockchain.

Rationale

When the Ethereum smart contract solution went "mainstream" in the blockchain industry, we got a sneak peek on the possible future and the power of Smart Contracts. Unfortunately, the Ethereum chain turned out to be a child playground where **everyone** could deploy a smart contract confined in the digital world. Ethereum got so bloated with useless copy-paste smart contracts no one uses that it has been through some severe congestion issues lately. Since Ethereum went online, many other crypto projects tried to mimic its success by creating different smart contract solutions. Apart from their arguable success, most of them lack what we think the blockchain world lacks: a real-world interface and, consequently, a use-case. In this paper, we present FaytheChain, a decentralized Oracle Network. We'll define how we envision the use of real-world **oracles** and **actuators** to ultimately create a bridge between the *Analog* and *Digital* world through the power of Blockchain technologies. We'll briefly cover incentives and the involved Game theory that creates an enough-balanced system to discourage bad actors. We introduce a two-tiered version of smart contracts called Arbitration Templates and Arbitrators. Thanks to the recent boost and democratization of **machine learning** algorithms and technology, we believe we could use **AI**, and **BlackBox testing** to cluster right behaving actors and quickly **identify outliers**.

$

Main Concepts / Definitions

Before discussing the arbitration concept in-depth, we would like to get the reader to familiarize themselves with the terminology and the concepts of the Faythe chain.

On-chain participants

Arbitrator Arbitration templates are the core of the Faythe chain. They codify an agreement between multiple parties to define the stakes of everyone involved and provide a mechanism for the decision of the outcome. E.g., one or more parties pay money in and based on the outcome of selected oracles, they get defined payouts, or activators get activated. We start referring to a template as an Arbitrator after its instantiation with a determined set of parameters. As a comparison to object-oriented programming concepts - the template is a class, and an Arbitrator is an object. Anyone can submit an Arbitration template to the chain, which can then be instantiated by any user or users of the chain.

Components of an Arbitration template

- List of possible interfaces (actors and actuators)
  - Either as an enumeration
  - Or as a blueprint definition with a minimal accepted Trust score
- The oracles' voting strategy for consensus
- Possible outcomes and payout strategies per each outcome
- Actions for each possible scenario
- Time-based constraints (penalty or dividend or interest)

Overseer The Overseer is the trust manager algorithm that is responsible for assigning and maintaining the trust level of interfaces and validators. It is unsupervised AI-based fraud detection and risk management algorithm built into the chain itself that monitors transactions and participants in the Arbitrators. It provides an overlay layer of trust by assigning scores to oracles and actuators through unsupervised learning. It clusters interfaces (based on their blueprints) and contracts (based on their interface clusters and decision logic) - it monitors the expected behavior and warns when outliers are happening - even on brand new contracts. Over the last decade, machine learning technologies made a huge leap forward. Thanks to the work of big tech giants such as Google, AI "algorithms" and machine learning has wide adoption and is now reasonably easy to use. In Faythe Chain, we would like to exploit a subset of machine learning called **unsupervised learning**.

> **Unsupervised learning** is a self-organized Hebbian learning that helps find previously unknown patterns in data set without pre-existing labels. It is also known as self-organization and allows modeling probability densities of given inputs.....

Wikipedia, Thanks to *" cluster analysis "* techniques, we can implement some anomaly detection, also called *outlier detection.* The Faythe Chain Overseer is bundled within the core code and monitors transactions and participants in the Arbitrators. Thanks to the above mentioned unsupervised learning, the Overseer could assign a reliability score to Arbitration templates (for example) by clustering similar ones together and check for anomalies in the expected output. Since Neural Networks become better and better with more data, they ingest we could reasonably say that this mechanism matures with the chain, eventually deprecating other safety mechanisms introduced in this paper (Eg: TrustScore). If this holds the FaytheChain's Overseer would probably become the first neural network of its kind as, to the best of our knowledge, there is no other blockchain project leveraging modern ML techniques to secure a public blockchain.

Validator The validator is a black box tester of off-chain connectors (oracles and actuators). It creates test Arbitrators that are indistinguishable from real ones. It includes several highly trusted interfaces to provide ground truth and invites some untested interfaces to measure their correctness. The evaluated interfaces are rewarded or penalized in the trust level dimension on how closely they match the ground truth. Validators also verify each other to avoid the possibility of a malicious scoring mechanism. The validator nodes are also the ones signing transactions, and thus the miners of Faythe.

Wallets Wallets are addresses on the chain who have certain funds and can instantiate Arbitrator templates. Arbitrators, users, interfaces, and arbitration templates all have their addresses.

Users A user is a person who can own one or more wallets. The chain works anonymously as well, so the chain's complete functionality is available without having users. The user concept is adding a layer of authenticity to individual wallets or Arbitration templates or interfaces.

Off-chain participants

Oracles A blockchain oracle is a third-party information source whose only function is to provide data into the blockchain, which reacts to the real-world information. Since blockchain technologies are, by design, trustless, blockchain oracles try to fill the gap between *on* and *off* chain. An example is worth more than 100 words. A Weather Oracle could provide weather information for a specific geographic area. An Arbitrator might want to use weather information to pay tourists back a chunk (or the totality) of their deposits in case of bad weather on the day of the hike.

Actuators In Faythe chain, we believe that just pulling the data from the real world won't just provide much innovation to the blockchain ecosystem. We believe that a two-way communication channel with the real world is needed, and that's where actuators shine. Actuators "do things." It's as easy as that! An Actuator is something that can interface with the real world. Just like any other component in the Faythe project, actuators can have a varying degree of trustworthiness (more on that later). A simple actuator could be a vending machine that releases a coke or a bank actuator, which initiates a bank transfer upon the completion of an Arbitrator.

Real-world Interfaces Blueprints Both *Oracles* and *Actuators* are "Real World Interfaces." To improve chain flexibility and promote healthy competition amongst both oracles and actuators, we need to introduce a new concept: Blueprints. Blueprints set an interface of needed input and outputs. For example, a blueprint for a *Weather Oracle* might be:

Inputs:

1) Latitude, Longitude
2) timestamp
3) requested information. Ex: rain_probability, humidity, pressure, or other weather data

Outputs:

1) Float value with different meanings based on input#3
2) (Optional) Accuracy

Achieving the "Trustless" property

"*Trustless*" is one quality any blockchain project should have. In Faythe chain, we operate with real-world interfaces (Oracles and Actuators). While there is no real way to prevent bad actors from joining the open blockchain network Faythe Chain, we can introduce incentives to encourage good behaving actors. Both Oracles and Actuators could ask for an operating fee. The *Weather Station* from the example above has some operational costs (Such as maintenance and electricity) that need to be covered by the fees. When instantiating an Arbitration Template, a user can either.

1) Specify one or multiple oracles that are allowed to provide data.
2) Specify a fee pot.

Oracles List Specification

If the user trusts a specific one or more oracles for an Arbitrator resolution such as (but not limited to): * Government-issued weather stations, * Popular sport betting company, * A reliable source of flight information to provide info about delays.

Arbitrator Fee Pot

There is a way to specify an Arbitration template where instead of enumerating the possible oracles, the oracles are invited based on a *blueprint*. Participating oracles send their output to be included and signed within the Arbitration resolution. Participating oracles that were in the majority share the pot. Starting with 50% as the default value, trust score that matures along with their participation and confidentiality. Ultimately the *Fee Pot* is dynamically distributed to all participating oracles. Even *"wrong"* oracles could get a slice of the pot, if the Arbitration template distrobution formular implements so. A simple *Fee Pot* distribution formula might use the TrustScore as weighting parameter like so:

$$\Xi(v_x) = \frac{pot}{\tau(v_x)} \sum_{i=0}^{n} \tau(v_i)\,.$$

As a result, new oracles can still participate in the network and gains τrust.

Arbitration template trust

Anyone can write arbitration templates, but most users would use certified Arbitration templates, which they can customize for their deal. Users don't have to review the code or be able to code, and they can instantiate any template of the chain and know what's expected to happen based on the description and the trust score assigned to it or the instantiated Arbitrators of the same kind. So when Bob hears about the Faythe chain and wants to bet Joe that by tomorrow midday, the price of oil is higher than 100$ / barrel, he may not need to implement a full Arbitration template from scratch, choose a commodities product from a financial service online. Anyone can write Arbitration templates, but it is preferable to choose a battle-tested and, more importantly, SIGNED template over a hardly used and anonymous one. A possible incentive to use the less popular (lower Trust score) template is that it may provide the service cheaper, although the lack of sufficient validation may include a higher risk.

Trust Score retargeting

Given that trust plays a vital role in the Faythe chain operations, its retargeting is a delicate matter. At every Arbitrator resolution with more than 1 oracles participating providing their outputs, each Oracle sees their input only. While this is still a research topic in FaytheChain, we think that τ could be retargeted using the following formula.

$$\tau^{`}(v_x) = \tau(v_x) + \lambda \times \delta$$

Where δ is the minimum weight:

$$\delta = \min 0.1, \tau(v_0), \tau(v_1), ..., \tau(v_n) \ \forall \ v_x$$

And λ is either +1 or -1 depending if v_x was amongst the winning or losing a set of all oracles. Using this straightforward and straight forward formula, we leverage the following outcomes:

- Only Arbitration templates with multiple oracles have trust score retargeting,
- Two colluding trusted oracles are not able to set up Arbitrators with the only purpose of gaining more trust,
- All participating oracles benefit the same amount reaching fairness.

Mitigating bad behaving actors

Along with trust score retargeting and the economic incentive of behaving correctly by providing valuable and correct data to Arbitrators, some bad actors might still arise and try to cheat. To mitigate even further, we need to: 1) Decide the winning party amongst all participating oracles 2) Introduce the "minTrustScore" safety mechanism

1. Deciding the winning party Flexibility is another crucial factor in Faythe Chain. Just like security, we need to make sure we future proof the project, and setting workflows and interfaces in stone is a no-go. Much like ETH has no features but instead gives developers the ability to specify their logic, deciding the winning party should be flexible. Smart Contracts can set their logic on how they would like to handle oracles output. Along with the *Fee Pot distribution", Arbitrators can decide how they read oracles data. For instance, some oracles might provide a non-discrete output leaving the Arbitration template's code the flexibility to set the rules that produce the Arbitrator output based on the oracles-input.

2. minTrustScore While having an open network is a nice feature to have, in FaytheChain, we also account for the user's security. That's why we introduce here the *minTrustScore* option users might want to specify to filter relatively new oracles as long as untrusted bad-behaving interfaces. Users instantiating the Arbitration templates need to specify a minimum amount of trust using one of the UI (Eg: FaytheWallet) that they desire.

Arbitrator resolution, τrust score, and economics of oracles

Oracles registration fees

Registering an Oracle to the Faythe Chain is not feeless. Fees are determined by simply looking at how much market for such oracle there might be: How? By observing the number of similar oracles created and used in the past N blocks. To provide a more formal definition of the oracles registration fees, we need to set some definitions: * $\omega_x(h_l, h_h)$ = the total collected Fees by oracles adhering the blueprint x within blocks having height $h_l \leq x \leq h_h$, * $\xi_x(h_l, h_h)$ = the total amount of oracles that registered within height $h_l \leq x \leq h_h$. * H = the current height constant * N = the number of blocks in the past we want to take into account Hence:

$$\Phi_{x,H} = \frac{\omega_x(0, H - N)}{\xi_x(0, H - N)} \times 0.3 + \frac{\omega_x(H - N, H)}{\xi_x(H - N, H)} \times 0.7$$

This simple formula mitigates any attempt of raising Arbitrator pricing by taking into account the collected fees since creation with 30% weight factor.

Note: When $\Phi_{x,H}$ is ≤ 0 then we use baseFee instead.

Oracles Trust score

Upon registration, Oracles are in an unknown trust state. Oracles can prove the chain that they are trusted by providing data, almost feeless, to Arbitrators having open participation for a certain amount of blocks. TrustScore discourages even more bad operators from spamming the chain with new useless oracles. Once Oracles are fully functional, it's essential to notice that Oracles might provide conflicting information with each other. That's why τ is important. When more than one oracle participate to an Arbitrator the weight of any possible outcome x will be calculated using the following formula:

$$\omega(x) = \sum_{i=0}^{\infty} \tau_i \quad \forall i \ who \ voted \ for \ x$$

So that calculating the winning party is trivial. If there is uncertainty about the winning party for any reason, the Arbitrator should be responsible for handling that case and decide the outcome. Please note that the Arbitration template implementation identifies the winning party. While this might sound scary and unnecessary, we don't want Faythe chain to allow discrete output oracles.E.g.: consider a weather station that provides a non-discrete value for "raining probability," it'll be up to the Arbitrator to decide what to do with the given value.

Interface availability

When *invoked* oracles and actuators should provide a result that is ultimately used by the Arbitrator. Failing in doing so ends up in the Arbitrator call to return an error and unexpectedly close the Arbitrator. Arbitrator errors waste both hashing power and real-world user's time. That's why we believe that *Interface Availability* should also be considered as a fundamental value in the Faythe chain economics and game theory. Building such alternative reputation system is somehow trivial and requires the nodes to account the following information: * α_x = `#` of assigned requests, * β_x = `#` of completed requests. Since both interfaces unavailability could be considered disruptive for the network with different levels of magnitude, we present the $\frac{\beta_x}{\alpha_x}$ ratio very clearly to the user making sure he poses the right attention to the availability topic and its decision implications. While we can always integrate the ratio as mentioned above within the chain, we let this as an eventual future core code improvement.

Interface participation quota

High reputation interfaces have the advantage of participating in arbitrations. All of the previous paragraphs and chapters outline precisely how we plan to discourage bad behaving actors. While we can anticipate a virtuous circle in which good behaving (and always available) oracles/actuators develop a good τrust score, we also want to discourage newcomers to quickly get a nice τ by simply creating Arbitration templates (and their instances) on which they'll participate throwing away some fees for building an excellent trust+availability reputation. As a preemptive safety mechanism, we disallow oracles to participate in Arbitrators during their cooldown period. To put this in simple words, we define the cooldown period (defined with "ζ") as the block-based time frame in which interfaces are not allowed to "write" to the blockchain (unless directly invoked).

ζ builds up with their reputation and scale down in case they prove themselves untrustworthy. Given that τ's domain is $\lfloor 0, \infty \rfloor$, the participation quota, defined in several blocks, is lower, the more significant τ gets.

Example use-cases

As the Faythe chain is a versatile arbitration protocol with real-world interfaces, its possibilities are endless. We want to provide a few possible use cases to showcase the system's capabilities.

Authenticated release of physical goods

A doctor prescribes some rare medicine with high potential for peruse. He may create a release token into a pharmacy Arbitration template with my trusted certificate, which signs the identifier (in an arbitrary form, printed QR code, patients DNA, fingerprint), and assign it to the patient. The patient goes to the pharmacies who are trusted oracles. Their oracle is (an RFID reader, DNA sequencer, fingerprint reader) checks the Arbitrator where the doctors signed release code is, matches it with the customer's unique key, releases the medicine. Payment happens instantly to everyone involved in the exchange.

Betting

Betting is an area where digitalized oracles are already familiar (sports sites with APIs, for example) and is easy to verify. An arbitration template may use several betting data sources, and the users of the template can also choose from many several odds calculation mechanisms. The betting parties can select the betting Arbitration template-based. They like to create an Arbitrator with the specific oracles. The Arbitrator holds the money until the underlying event is finished, and the oracles achieve their consensus. Upon completion, the money is transferred based on the payout mechanism to the participants. Also, the template owner and the oracles/actuators get their fee.

Airbnb on the chain

Airbnb may have an Arbitration template, where the renter puts up its house as the product, its bank's oracle to verify payments, and also its apartment door's NFC reader as an oracle. The customer's transfer (with a reservation code in the comment field) and her phone's NFC are the input for the two oracles to fulfill the Arbitrator. The actuator is the door opener in the apartment. The bank can be an escrow account's bank that releases the funds to the renter upon successful entry to the apartment.

Tracking of physical items or locations - gamified example

We're using a "capture the flag" Arbitration template to organize a bike race. The oracles are RFID tag readers, placed around town with a defined sojourn plan. The race participants register their RFID tags into the Arbitrator and submit the fee into the "pot," which is the Arbitrator wallet. The winner gets the pot.

Conclusion

We've introduced FaytheChain, a decentralized oracle and actuators blockchain network for mediator free arbitration to securely interact with the real world. In this paper, we have outlined the main concepts and incentive mechanisms that provide the fundamentals of the chain's functionality.

With our relentless work, we are striving for a more transparent and fair future. In this future, there is no need to rely on formal establishments or centralized authorities, which have often shown to be corrupt or unreliable. Our goal is to create a transparent and fair system available to anyone so that trust becomes a shared resource and reliability.