Surenet: A Distributed Compliance Network for Taboow

Taboow Team

version 1.2, September 17, 2018

Abstract — Surenet is a Decentralized Compliance Network (DCN) that validates the compliance of transactions to all type of sources: any piece of software as a third party might mine (research) information regarding an address at any given moment for proving its integrity.

It is designed to work on a distributed ledger with a native protocol token called a Taboow coin or simply a TABU, which miners — called analysts— earn through a *research and claim* (RC) of the "Know Your Customer's Customers" (KYCC) and through further analysis for clients among usual due diligence activities. On the other hand, clients spend TABU to pay analysts for their RC work and analysts, while working out their reputation, compete to mine blocks with rewards, though Surenet mining power is limited by their performance in terms of reliability.

There are thus incentives for analysts to complete their work reliably, and perhaps honestly, to protect their reputations and to not to deceive the network. Surenet assigns RC tasks to analysts for the sake of efficiency and scalability of the network, while dealing properly with attacks. The pricing of RC tasks is predictable so that clients have the freedom to balance certainty and prices by allowing them to determine how many analysts will work on their RC tasks.

Surenet enables us to build a decentralized, immutable, censorship resistant and long term archive of the relevant digital data of virtual currency investments that is aimed at ensuring that identity, scoring or related knowledge will remain verifiable indefinitely for the sake of the best compliance, ever while protecting legality and privacy of the corresponding activities.

I. INTRODUCTION

IN THE RECENT YEARS, Bitcoin, Ethereum, Monero, and several other projects of virtual currencies have raised attention of the possibilities of decentralized transaction ledgers, that incorporate any kind of smart-contract-like functions that afford virtual currencies the capability to be programmed what is key to add some degree of intelligence to the money. The distributed ledger technologies (DLT) enable decentralized economies through which individuals, companies, and in a future, machines, worldwide transact safely without trusted third parties as intermediaries. On the other hand, DLT are so far self-contained in their supporting ledgers and have very little to no capacity to interact with other ledgers, the Internet or the rest of the world. In Ethereum there is the proposal of *oracle* as a trusted entity as a way to feed outer information into smart contracts from the real world. In our proposal, oracles are developed as algorithms, and will be called *analysts*. Oracles enable smart contracts to react to events occurring everywhere out and within the ledger world. Anyway, given that this approach places trust in a witnessing, attesting or checking third party (oracles and analysts), it cannot be considered reliable or tamper-resistant, leaving room for contestation and repudiation. A problem to solve in the adoption of analysts is that any source might represent a point of failure that creates opportunities for external malicious actors to remove, rewrite, or insert facts by breaching one system or the network. There is no way for DLT to deliver in their decentralization promises until decentralized oracles can be workable, supported by analysts, that do not blindly trust on third parties but instead rely on digital versions of the wisdom of the crowd.

Additionally, **analysts** as oracles use multiple algorithms to detect fraud, and this can work according to promising evidence: the more claims of fraud there are, the more likely processing via confirmation transfers ICU and URI mechanisms is to occur, as it is explained in section B in this introduction.

A. General Aspects and Presentation

The analysis proposed here is based on **built reputation** rather than aprioristic trust, as in other classical approaches. In fact, the claims of the analysts that will be considered to be true are developed by combining a number of likely claims drawn from a diversity of players that do not know each other and have their own interests. The proposed solutions might work like decentralized prediction market as Augur [2] which based on the notion that market prices can indicate what a crowd thinks the probability of an event is, in the sense that outcomes of checks, researches, and attestations are derived from claims that various miners unknowingly to one another have voted or claimed for.

The analysts (miners) are designed to make an automatic work, and so in most of the cases, they are computers. Some new computational tools based on machine learning have been developed; a short explanation of the mathematical arguments

[1] This whitepaper is preliminary work focused on algorithmic approaches to compliance tasks executed in relation to virtual currency transactions.

in which they are constructed is explained at the end of the paper. So, the "analysts" uses the software with the information that they receive without intervention of the owner of the computer. Therefore, Surenet is built on a Blockchain, initially Ethereum and Bitcoin, with its own token (the TABU). These tokens are used for the KYCC analysis, in relation to pseudo-identities (Ethereum account addresses), and scoring which is delivered back to them. Analysts and oracles can make tokens for fulfilling such work.

So, our solution is featured as:

- Analysts and oracles might collect tasks considering their performance, worked reliability and honesty, or reputation. The more an analyst or oracle's reputation, the more likely a research task will be assigned to it and the more its claims will be taken into account. To mitigate compromised nodes risks, without limiting the distributed computing efficiency, a Proof of Authority (PoA) model might be combined with a quorum-based technique with a random peer selection process.

- Analysts gain reputation in a democratic way. The better their claims match the claims of the majority the better reputation marks will be. On the other hand, the numerical marks of reputation decrease when the "opinion" of the analyst do not follow the main stream.

Obviously, there would be some cases where the procedure might not follow exactly this scheme. The above described rules will be modified depending on the particular context in which the procedure is developed. For example, a purely democratic system could be modulated in case (some of the) analysts might have some kind of conflict of interest.
On the other hand, oracles can be human-powered or partly automated or supervised nodes that are non-standard of the DCN protocol and can be individuals, whistle-blowers, and other entities.

B. More Checking Mechanisms

As analysts detect many situations, and thus in Surenet a library will be installed at every node (miner), however, we add two core algorithms that will facilitate the KYCC process:

The First Algorithm is the "ICU" or "I See You" algorithm (i.e., "I trust you") whereby X acknowledges account Y that is trustable. X can also be a BCA-registered corporation or individual with reputable scoring capacities. X is asked by Surenet to send a concrete amount of money Xa that must be immediately returned by Y. Both pay a fee for their two transactions in the virtual currency of Surenet coins that they are using at a significant discount to inform Surenet of the execution of ICU - KYCC transactions.

The Second algorithm is the "URI" or "You are I" algorithm (i.e., "these two addresses are of the same owner") whereby you, X, acknowledge another account, Y, that is yours. This involves a double ICU whereby instructions are only given to X rather than once to X and then to Y. In the first ICU, X is asked to send an amount of Xa to Y that is immediately returned by Y. The same fees of the first algorithm apply. A second ICU is then executed by asking X to make Y send another amount Ya that X must immediately return in a subsequent transaction where fees apply.

The amounts of Xa and Ya are proposed in such a way that transactions are easily identifiable and in any case may be faked without the proposition of Surenet, and the same occurs for fees.
Thus, Surenet applies an ICU and URI KYCC to assign scores and when applicable BCA trusted certification propagation. The amounts of Xa and Ya may also be modulated from smaller amounts to greater amounts to start modulating the magnitude of trust, rendering addresses that prove an ICU of higher amounts more trustworthy. The rationale behind this is that one might trust to another when one risks an enormous amount of money to prove it. As well, when the ICU is of a maximum level it may be a candidate for a URI without proving that it is a URI.

ICU and URI mechanisms can be activated at any time by the requests of analysts. When requested by clients, clients pay fees (or at least a majority of them). Other algorithms for analyst mining are presented in section VI.

The rest of this document is structured as follows. Section II will define a DCN and Surenet is presented in Section III as a type of DCN. Section IV presents the protocol, and Section V presents the reputation model. Section VI describes the algorithms deployed in the first alpha for analysts.

II. A DECENTRALIZED COMPLIANCE NETWORK

A DCN is network composed of computers, usually called nodes. They all use the same software, which will be used for the verification tasks that will be necessary for the proposed work. The information comes from external sources and is entered in the network in order to be processed. Communication between the pairs that make up each node is also guaranteed, and the information circulates through the network, allowing it to be checked. Thus, a DCN is a system that processes Research & Claim (RC) requests and URI/ICU actions for inspecting the compliance of investors with ICO and all type of crowd sales.

In particular, based on some new machine learning tools (together with the supervision of some experts if needed), the Surenet protocol is an incentivized and verifiable DCN construction built on several innovations. It works over the abstraction for a network of independent providers implemented as algorithms -called "analysts"- to offer RC services:

- *Reputation-Based Mining protocol (RBM)*: Miners, as analyzers or witnesses, do not need to spend time on nonsense computations to mine blocks and instead must

fulfil task assignments, as done in [12]. A Proof of Reputation as a combined Proof of Work and Proof of Authority is proposed.

- *Reputation-Based Task Assignment protocol (RBTA)*: it is an algorithm based on e-auctions and citation auctions that lets the network assign tasks to analysts in a decentralized, fair, uniform, and unpredictable yet deterministic way, similarly to our previous works [10] [15] and [19].

A. Research & Claim

RC requests contain information on how an address must be researched/investigated and acknowledged as a reliable or white money source.

The two elements of an RC request are defined as follows:

- *Research*: to check and agree on the veracity of retrieved information or knowledge that is internal or external to the network and that is calculated from algorithms.
- *Claim*: to supply the conclusions to the client that emitted the RC request, preferably with a degree of certainty.

When a DCN client sends an RC request to the network, nodes process it by retrieving their own information and checking that they all have copies of the information, making calculations or mining, and finally checking or testing what algorithms are analogous and perform identical calculations and making them available for the requester client, similarly the smartcontracts in Ethereum work.

When an RC request is assigned to a DCN node, this is converted into an RC task. Native tokens can be send along the RC requests to reward nodes, analysts, and oracles, for their work and encourage people to run nodes in a DCN. This is inspired by how miners are rewarded in Bitcoin for their combining transactions into blocks [4], for witnessing [11], or by how miners are compensated for their contributions to visual recognition [12]. Said that, when a DCN node is assigned a RC task, it is rewarded with tokens attached to the analyst's request, honored with the reward as long as its claim matches a majority of other participants' claims.

To save costs of performing RC tasks, a DCN can incorporate a PoR (Proof of Reputation) that to assign a PoW task to the less trustworthy/reputable participants instead of PoW for all.

In the end, DCN nodes, the analysts, may experience conflicts of interest when executing RC tasks. In anticipation of this, a DCN implements the said *RBTA* that allows participants to be assigned to more tasks based on their past degree of honesty. Moreover, when a DCN node decides to tamper with RC tasks or makes false, biased or completely fictitious claims, such claims will likely contradict those of reputable participants that pretend to work for Surenet for the long term, and therefore it will miss its opportunity to collect rewards.

III. The Structure of Surenet

The protocol proposed for Surenet contains strategies to keep up the quality of the result of RC requests and eliminate chances of manipulation by detecting and penalizing collusion. Additionally to the incentives by the use of token rewards that encourage participants to behave and not to feed false o careless claims into the DCN, several of the techniques here applied are inspired by several papers like the Sztorc consensus algorithm contained in the Truthcoin whitepaper [1] and the Augur consensus algorithm [2], which as well is based on Sztorc; the wit currency of the witness consensus whitepaper [11]; rewards for crowdfunded processes [13]; studies on Reputation-Based voting and on how it is affected by collusion attacks [8] [9], prestige and necessity aggregation algorithms for source fusion developed by de la Rosa [21] and Aciar [19]; other approaches to collusions in voting [16]; the witcoin.io project presented in the witcoin paper for acknowledging useful knowledge named as wits [20]; and reputation Page-rank-like approaches to reputation as described by Szymanski et al. [15] or those based in open innovation approaches developed by de la Rosa [7]. These mechanisms allow to control the action of possible cheaters, since they operate on the way rewards and penalties work in the system.

Another relevant control mechanism for ensuring that the measures proposed in the previous references are effective, is based on the need that the analysts have no way to contact one another, in order to prevent agreements to gain rewards. It may happen that an agreement of a big amount of analysts communicates the rest that they will sign for a (false) claim, forcing the others to follow them. As said in [11], the effect is the same when the entity announces a vote for a false claim, promises a bribe greater than the reward for telling the truth to whoever votes for the same, and causes most analysts take it. In fact, our protocol intends to grasp the same benefit promised by [11] by rendering "gambits completely useless by not giving participants the chance to reveal or prove the actual value of claims [that] they vote for". As Sanchez et al. claim next [11] "even when a participant accepts a bribe, it can still tell the truth to the DCN and lie to the briber and earn both the reward and the bribe. This is the most profitable of options available and is therefore the most likely to occur". For its part, he asserts it is not for granted that the participants will keep their word, and concludes that as participants, they are incentivized to act for the good of the DCN, to tell the truth to deceive the briber, it is likely that none of the bribed participants will vote for a false claim. Thus, any bribe attempt will lead the briber to waste its own resources.

Based on the analysis of Internet contents, Surenet focuses on the research and attestation or claim of DLT network content. Foundational to a DLT network is the fact that such a network keeps open: New users join daily, and existing ones often create new accounts. From this dynamism, the software in charge of performing cluster network retrieval must be capable of interpreting the DLT to find clusters and outliers. With regards to all of these considerations, Surenet miners use

a DLT capable of reading to perform the network analysis and clustering.

Surenet implements a public ledger that keeps a record of all transactions occurring in the network. Regarding the participation of miners, a rather standard method will be used, with some specific relevant properties. First, and to encourage the early adoption of Surenet, miners might also get some early benefit from Taboow: some amount of tokens proceeding from Taboow will be reserved for this aim. However, this kind of reward will decrease with time, being substituted by fees only. Transactions are recorded as in many open DLT, but to find a solution to a meaningless problem using costly algorithms is not necessary in the Surenet network. A reputation protocol is designed instead for choosing the adequate miners.

Regarding the possible group of miners, any person can become an analyst just by running the adequate network node. Thus, they will earn tokens for the work as explained above. On the other side of the coin, clients pay tokens to have accounts checked by analysts and oracles through the DCN. Depending on the research task complexity –number of analysts needed, the use of ICUs and URIs-, the cost will be determined.

B. Illustrative Case

As an illustration of the protocol, let us think of Andrew as a client who wants a score of a funding address **A** researched and claimed, which is going to send money through his ICO.

Andrew sends an RC request to the network as a Surenet transaction. He might attach few tokens to the task. This amount might depend on the complexity of the research and the claim's certainty that he expects which will condition the number of analysts that will be employed. The steps are:

Step one, *the task of solving Andrew's request is assigned to a set of analysts who are elected through the RBTA protocol. Each of these miners will:*

- Research the network cluster in the ledger regarding the address specified in the request. The resulting value is so-called a **claim**.
- Calculate a hash of the claim which will be different for each analyst because it will be derived from the claim itself, from the miner's public key, and from the hash of the latest block.
- Send the said hash to the network as a commitment to disclose the actual claim when the remaining analysts in the set would have made their own commitments. This is called *the pledge*. The analysts conduct a test so that Surenet determines whether they behave analogously in groups of 3, 5 or 7, proportionally to the reward included in the request. The analysts' commitment transactions are a pledge for a share of the tokens in the RC request. Requiring from 3 analogous algorithms multiplies by 3 times the number of necessary claims to be condensed.

Step two, *once all designated analysts in the set have made their pledges*, reward tokens are divided and attached to the RC request for the analysts with a *valid* pledge for such a request.

Step three, *analysts who made their pledges will disclose* the researches by means of commitment transactions and collect reward tokens.

Step four, *for the next block claim*s from the several analysts in the set are compared and the winning claim is selected by applying a trust aware consensus algorithm. The analysts who knew or guessed the truth, those who achieved consensus, earn reputation points and transactions redeeming their share of the reward are accepted. Otherwise, the analysts who made incorrect claims, those who did not achieve consensus, lose part of their reputation points and cannot redeem their share of the reward.

Step five, *the result of the claim is public* and available to Andrew, and to any other participant of the network as it is in the DCN ledger.

IV. THE SURENET PROTOCOL

The operations performed by clients, the Network and different types of miners/analysts are described in this section.

The role of the clients

They can ask for research and claim (RC-request) of scores by paying analysts in TABU tokens. A client submits an RC client request transaction to the network, and determine the number of analysts to be assigned to the RC task by a replication factor in their requests that is set to 3 by default, and this is multiplied by the lowest number of checked algorithms, which is 3, so the number of analysts involved is from 9. A factor of approximately 6 (that means we will have 18 claims) should be enough in most cases while keeping costs under control. Clearly, more redundancy generates more certainty and confidence in the claim.

The client reads the results of RC requests (RC-read). At the same time that analysts apply the trust aware consensus algorithm, a verdict pointing to the researched and claimed addresses emerges. Transactions with the value of researched and claimed content (the solution) are immediately included in the block being mined in the Surenet ledger. As clients can read the ledger at any time, when the block with the consensed claim is broadcasted to the network, it might be tracked back to the related RC-request transaction. Therefore, the RC-Request can be run locally at no cost of transaction by any client with an up-to-date copy of the Surenet ledger.

The role of the analysts

They do the four actions in every epoch:

- *Read RC-Requests*. When an RC-Request appears in the system, all analysts have what they need to work on the given research.
- *Discover research task assignments:* A task assignment protocol is run for this aim by the analysts. In particular, analysts run the RBTA protocol on RC-Requests broadcast in an epoch to find out which RC tasks they have been proposed to. When pending RC requests are made with an expiring time lock precondition then the task assignment protocol will check them for assignment.
- *Research and commit-pledge*: Analysts executes the research component of their assigned RC tasks and pledge to disclose their results. Now analysts start to fulfill their assigned RC tasks, and when they have the queries, they insert commit-pledge transactions as a sort of broadcast, wherein they cite the information they used to build up their claims to be available for check by anyone else.
- *Disclose:* Analysts disclose the results of their assigned RC tasks and fulfil rewards.

When commit-pledge transactions are included in a block, then each pledging analyst disclose its claims and provides what is necessary to prove that the hashes that it committed in its commit-pledge transactions were obtained from the disclosed claims as well as they are the actual authors of the pledges they want to fulfil.

V. REPUTATION

Surenet employs reputation points, which, together with its TABU reward system and the fact that claims are stored in the ledger, it uses 3 reinforcement mechanisms for considering Surenet a type of reinforcement learning machine.

The total number of reputation points included in the system is a fixed quantity. Holding reputation points entitles a Surenet analyst to be elected to perform RC tasks and to mine new blocks. Therefore, when an analyst's reputation increases, the likelihood of working on CR tasks and receiving token rewards also increases.

Reputation points in Surenet, might be earned and wasted depending on how reliably the analyst votes with the consensus work similar to ArtTRUST [3] [6], to source selection mechanisms [19], to trust aggregation [21], to Truthcoin's Votecoins [1], to Augur's REP [2], and to a lesser extent to Filecoin's Power [17].

Reputation points as defined in [3] [6] and [19] are preferred to be applied to Surenet, and in any of the examples this is a DLT of their implementations. Having a fixed number of total reputation points provides immunity to sybil attacks and at the time provide the network an effective means to penalize miners showing a lack of initiative. The network is designed to not depend on the number of participants. All of its implicit economic models operate the same regardless of whether a single large actor assumes most reputation points or whether numerous smaller actors carry the same number of reputation points.

Reputation points might be affected by demurrage[2] as [11] proposes: analysts reputation points deplete when they store their points rather using them to have a say in the outcomes of RC requests. Therefore, reputation points are both an asset and a liability, as their owners put them to proper use or to lose them altogether when they do not [15].

A. Reputation Procedure

The same welcome score will be assigned for all new entering nodes/analysts. Reputation points, once created never leave the network and cannot be destroyed unless the respective analysts is shut off.

[2] Demurrage is defined as the loss or decay of a property through time, related to the cost of holding currency over a given period, and is used in complementary currencies like Chiemgauer.

Analysts earn reputation points as long as they agree with most other designated analysts on resulting claims of the RC tasks. On the other hand, reputation points are lost when the analysts contradict or fail to agree with most other ones.

At epoch checkpoints, few reputation points are lost by all network analysts at once as demurrage. The points are rewarded to reliable analysts that fulfill RC tasks.

In every epoch, the sum of reputation points earned by reliable analysts is equal to the sum of points deducted from unreliable ones plus the points deducted from all analysts in the network.

However, no analyst will lose its reputation if during an epoch every analyst is reliable, except for demurrage which will still apply to all, and as said deducted points are evenly distributed among analysts fulfilling RC tasks in that epoch.

In a way, in Surenet, being designated to fulfill RC tasks works as a lottery in which reputation points serve as lottery tokens: The more reputation a participant has, the greater chance of the right to collect block and task rewards.

With the goal that most reputable participants also bear more liability, the demurrage being it in percentage of the total reputation, causes the score of the most reputable ones to decline more rapidly than of the smallest ones which is left nearly intact. It is a version of progressive **demurrage** as a means to fight concentration.

In Surenet, reputation is *public and verifiable* by reading the ledger because anyone can calculate the reputation of each analyst/oracle at any point in time so that one's reputation is improved and compromised by performing RC tasks with outcomes that are publicly available and anyone can check the outcome of such tasks and determine if the reputation claimed by an analyst is correct by reading the ledger. And finally, reputation is *balanced*: Analysts earn reputation points by fulfilling RC tasks reliably or lose otherwise or ignore them. This is how reputation points go from the analysts not contributing to the system to those who do contribute.

B. Trust Aware Consensus

Trust aware consensus or similarly Truth-By-Consensus methods used in Surenet's system as proposed by de Pedro and Levi [11] or a more page-rank minded approach of Zsymanski, Krishnamoorthy and de la Rosa [15] as well as other approaches based on data source fusion such as those described by Aciar and de la Rosa [19] and Montaner et al. [6], or modern versions of question propagation through trust [14] may be applied here to improve the protocol for comparing and finding an agreed upon trust out of potentially conflicting claims brought about by independent participants or sources of the network.

The trust aware consensus algorithm chosen is similar as Truthcoin [1] that is based on Singular Value Decomposition (SVD) using the statistical technique of Principal Component Analysis (PCA) while it introduces weighting to take reputations into account.

The SVD analyzes a matrix of all claims made during an epoch and to disclose and sort their effects by influencing, detecting, and removing outliers and colluding analysts. To measure coordination as potential source of collusion, SVD uses the first score calculated from a weighted PCA. This column contains the analysts' claims differences from those of a maximal representative of the covariance across all analysts and their claims.

Reputation points are redistributed every epoch among all analysts designated to fulfill RC tasks and offered their claims on time. When claims are unanimous, reputation scores do not change before the demurrage applies.

Reputation redistribution turns up to be of high computational complexity along the number of network nodes. To relieve it, the protocol can be implemented in lazy way that scores are only updates once in few epochs, recomputation period which should be kept as short as possible for early detecting and not delaying the penalties to unreliable analysts because such measures would lose efficacy. Please note that this period is essential of the network consensus, as reputation scores have key implications for the consensus of the analyst algorithms protocol described in section VI.

Trust aware consensus incentivize the network participants to become analysts, to develop their assigned RC tasks reliably and to make true claims for revenue maximization. All participants are incentivized to achieve strong reputations as their potential incomes are dependent on their scores.

As noted earlier, Surenet offers an incentive for analysts for the secrecy of their claims until they all have their commitments disclosed, representing a *double-agent incentive* as coined by Sztorc [1] so that if someone attempted to coordinate analysts outside of Surenet, analysts would lie to the coordinator and stick to their reputation.

As predicted by [11], as long as more than half of the participants in the network are reliable, cartels or pools are not attractive as analysts are expected to get rid of dishonest and unreliable fellows' involvement, as they compete for a share of the same rewards acting as rivals.

C. Analyst Fees

Analyst fees calculations are in function of the computational complexities in the RC and the quality of the claims by requiring a replication factor (R). There is possibility to appraise the costs derived from R so the client might know in advance how expensive a RC request will be for the network to fulfil. The actual cost required for analysts to perform the tasks that they are assigned must be marginal and virtually negligible when compared to the rewards they receive. The only significant costs involved are miner fees that they need to pay to broadcast their commit-pledges and disclose transactions and to eventually collect their rewards.

Each analyst's reward will always be equal to each request's analyst fee split by the number of committed analysts, which in turn shall be at least the replication factor. As long as an analyst has spare computational power and the reward is more than a threshold, there is no reason why an analyst node should ignore tasks it was designated, as such behavior is penalized by the reputation *demurrage*. This way, Surenet's reputation model turns analysts' fees into bonuses for only

reliable, honest ones to be eligible. As a result, the profit that analysts make for their work not only comes directly from the fees that clients pay but also indirectly from earned reputation points, which eventually afford them a more chance of getting tasks and mining a block. Theoretically, this could even lead analysts to accept tasks at a loss for the sake of long term rewards that will compensate and exceed the loss.

VI. Analyst Algorithm Protocol

Analysts run any algorithm existing in Surenet such that they mine the blockchain and additional information to check whether they can achieve a consensus on results and claims. Thus, analysts are run on nodes and retrieve their reward in TABUS as described above. However, as opposed to having reputation points assigned to the node completing an analyst code from the Surenet Analysts Library, the reward is an important component assigned to the owner of the node while a final fraction (perhaps 20%) goes to the creator or designer of the code. This introduces an incentive to continuously enhance the code for mining and conducting research and to comply to offer the most varied, reliable and useful code for analysts of a newly created market where nodes may also be encouraged to use or support specific analyst codes over others.

In this paper, a first set of analyst codes is proposed to be launched along with the very first version of Surenet.

To detect various forms of financial fraud, a metric learning model is proposed as a common analytical feature of reinforcement learning. From different applications of these ideas we have developed several algorithms focused on particular aspects of fraud detection. Some current developments that can be found in the scientific literature are closely related to our setting (see for example [22], [23], [24]; see also [25], in which a section describing existing techniques on metric learning of graphs is given).

Another relevant concept must be taken into account. As we are analyzing processes that are in some sense directed (either time dependent processes or causal directed graphs), sometimes is necessary to use non symmetric topologies. That is, the "distance" from point a to point b does not coincide with that from b to a. Technically, this means that we sometimes alter our metrics based on the so-called quasimetrics for our approach (see [26]).

These are as follows:

A. Metric graph analysis for fraud detection. Heuristic methods of fraud detection rely heavily on programs based on databases structured as graphs. In the same vein we propose automatic procedures and algorithms that from a graph-type database run analytical rules for addressing special aspects of fraud (see [27], [28]). Distances between the nodes of a graph can be measured to analyze data that can reveal situations potentially involving fraud. The following steps summarize the process.

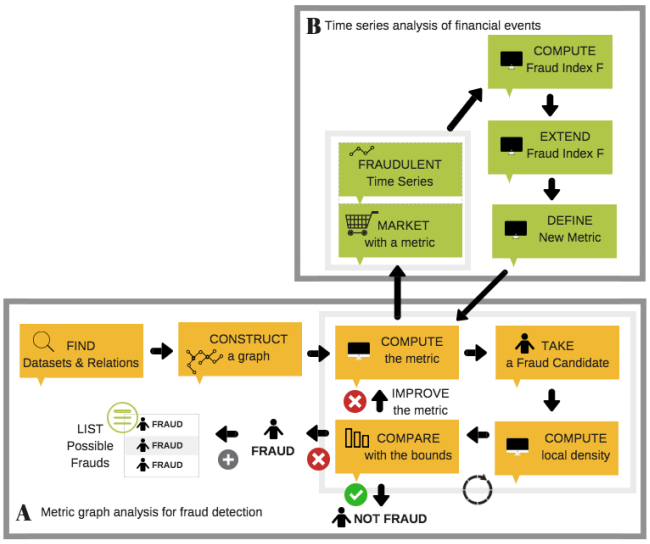
a) Data related to financial processes (nodes = sets of invoices, contracts…) are organized under a graph structure (edges = relations between invoices, related contracts, …).

b) From a canonical graph-type distance, information on the properties of nodes and edges is used to define an adapted metric. It is assumed that the metric measures how far a node is positioned from any other node in the graph.

c) Information contained in the original graph together with other information can be used to improve the metric defined in the graph. To do this, reinforcement learning methods are applied.

d) The procedure "suspects" that the neighbor of a given node that has been proven to be at the center of a fraud process is also a fraud point. This basic rule allows one to apply mathematical tools for detecting fraud processes from the graph. The main principle is stated as follows: large variations in the local density of a region of the graph denote the presence of fraud risk; that is, very high (or low) values of density in the neighbors of a node denote the potential for fraud to occur.

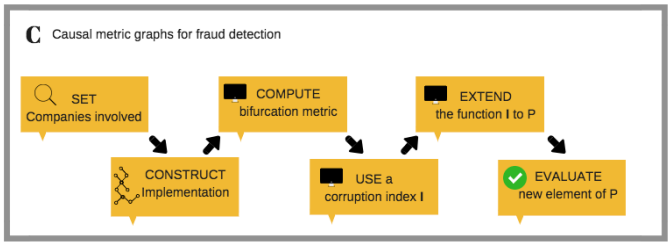
B. Time series analysis of financial events. Dynamics of data searches for uniform patterns can be examined to detect fraud. Once some classical fraud patterns are detected, they can be used for the analysis of general time series. Some families of metrics are often used in metric learning to address such problems (see for example the role of Mahalanobis distances in [29]).



a) Suppose that we have a description of the dynamics of time dependent magnitudes representing a financial process (e.g., the different values of several products offered within a time dependent marketplace).

b) Some recurrent behaviors of sequences of states of all products of the market are detected as indicators of the presence of some form of financial fraud (e.g., correlations among completely different products would indicate the existence of some form of collusion among different companies).

c) The set S of these particular time series (in which the existence of fraud is assumed) is used to define a real function representing an index F of the probability of having some form of fraud appearing. For example, some cases of collusion and how they affect dynamics of prices would be used.
d) An adequate metric is defined to construct a metric space in which elements are sequences of states of the market. Function F is Lipschitz in S, and so its extension E to the whole metric space containing all potential time series is ensured as a consequence of the McShane-Whitney Theorem.
Extension E is used to approximate the probability of any other sequence as a "symptom" of fraud, creating a fraud indicator of, for example, cartel recognition.
e) Extension E, together with new information obtained for the system can be used to improve the metric of the space of sequences. This creates a procedure of distance learning. Thus, such tools can be improved by means of reinforcement learning methods for pattern recognition.



C. Causal metric graphs for fraud detection.

The dynamics and relations of events that should be relevant for detecting financial fraud are not necessarily time dependent. In certain cases, causal connections among events should be considered instead. For example, in a series of contracts, connections among pre- and post-situations would indicate the presence of financial fraud (e.g., the existence of a cartel). Under adequate requirements, causality relations among events can be represented by means of a directed graph. Again, sequences of connected nodes of the graph are considered as elements of a metric space for a suitable distance. This generates a different fraud detection algorithm that can be schematically described as follows.

a) A graph with relevant events as nodes, together with vertices defined by means of causality relations among them, is defined.
b) An appropriate distance among paths in the graph is considered. For instance, the bifurcation metric among sequences is a good candidate. It is defined as follows: for two given paths, the maximum path length minus the number of nodes shared at the beginning of the process gives the desired distance.
c) Again, some set S of paths already known as prototypes of fraud situations together with typically fraud-free cases are used to define a kernel for the definition of oracle (Lipschitz) function F.
d) As in the previous case we can extend this via the McShane-Whitney method (offering explicit formulas) to obtain a final evaluation/oracle function E where the domain is the space of all possible paths in the graph.
e) This generates a first approximation. From the simulation of new situations involving function E (that we call "dreams"), we can enrich the original dataset S to improve E.
f) Finally, this E is used to improve the bifurcation distance (the one started with), applying in this way a new reinforcement learning method in the context of distance learning.

Information on mathematical tools developed to support the algorithms described above can be found in [30], [31], [32] and [33].

ACKNOWLEDGEMENTS

We thank *Sciechain* and contributors from the Universitat de Girona and Universitat Politècnica de València for their insights and contributions as a seed community for the development of the proposed platform, and Daniele Levi for proactive insights regarding witnet [11].

REFERENCES

[1] Sztorc P. (2015), Truthcoin: Peer-to-peer oracle system and prediction marketplace, 2015. http://www.truthcoin.info/papers/truthcoin-whitepaper.pdf
[2] Peterson J. and Krug J., (2015) Augur: a decentralized, open-source platform for prediction markets, *CoRR*, vol. abs/1501.01042, 2015. http://arxiv.org/abs/1501.01042
[3] del Acebo E., Hormazábal N., and de la Rosa JL. (2007): Beyond Trust. Fuzzy Contextual Corrective Filters for Reliability Assessment in MAS. Application to the ART Testbed, *Sixth International Joint Conference on Autonomous Agents and Multiagent Systems* (AAMAS'07): Conference & ART Competition, Fecha: May 14-18, 2007, Hawaii, the USA
[4] Carrillo, C., de la Rosa J. Ll, and Canals A (2007): Towards a knowledge economy. *International Journal of Community Currency* Research 11.1 (2007): 84-97.
[5] S. Nakamoto (2008): Bitcoin: A peer to peer electronic cash systems, 2008, https://bitcoin.org/bitcoin.pdf
[6] Montaner, M., López, B., de la Rosa, J. Ll. (2002): Opinion-based Filtering through Trust, *Cooperative Information Agents IV*, pp: 164-178, septiembre, 2002, Ed Klush, M., Ossowvski S. (eds), Springer Verlag, Lecture Notes in Computer Science (Artificial Intelligence) 2446.
[7] de la Rosa JL, Gibovic D., Torres-Padrosa V. (2017): A Preliminary Work on Virtual Currencies for Open Innovation, IV International Conference on Social and Complementary Currencies, Barcelona, May 22-24, 2017.
[8] Bendahmane A., Essaaidi M., Moussaoui A. E., and Younes A. (2015): The effectiveness of reputation-based voting for collusion tolerance in large-scale grids, IEEE *Transactions on Dependable and Secure Computing*, vol. 12, pp. 665–674, Nov 2015.
[9] Watanabe K., Funabiki N., Nakanishi T., and Fukushi M.

(2012): Modeling and performance evaluation of colluding attack in volunteer computing systems, in *Proc. of the Intl. MultiConf. of Engineers and Computer Scientists*, vol. 2, 2012

[10] Suárez S. B., Christian G. Quintero M., and de la Rosa JL (2010): A Real Time Approach for Task Allocation in a Disaster Scenario, ISSN:1615-3871 (Print) 1860-0794, 1860-0794 (Online) *Advances in Practical Applications of Agents and Multiagent Systems, Book Series Advances in Soft Computing*, Vol. 70/2010 pp: 157-162, ISBN: 978-3-642-12383, DOI 10.1007/978-3-642-12384-9, Publisher Springer April 2010.

[11] de Pedro, A. S., Levi, D., & Cuende, L. I. (2017): Witnet: A Decentralized Oracle Network Protocol. *arXiv preprint arXiv:1711.09756*.

[12] de la Rosa J, L., El-Fakdi, A., Torres, V., & Amengual, X. (2017): Logo Recognition by Consensus for Enabling Blockchain Implementations. *In Recent Advances in Artificial Intelligence Research and Development: Proceedings of the 20th International Conference of the CCAI*, October 25-27, 2017 (Vol. 300, p. 257). IOS Press

[13] Moreno A., de la Rosa "Human" JL, Szymanski BK, and Bárcenas JM (2009). Reward System for Completing FAQs, *Artificial Intelligence Research and Development* ISSN 0922-6389, Vol:202, pp:361-370, Nov 2009, Ed: IOS Press

[14] Trias A. and de la Rosa JL. (2011): Propagation of Question Waves by Means of Trust in a Social Network, In: H. Christiansen et al. (Eds.), Proc. 9th Int. Conf. Flexible Question Answering Systems, FQAS'11 LNAI 7022, 2011, pp. 186–197.

[15] Szymanski B., de la Rosa J., and Krishnamoorthy (2012), An Internet Measures of the Value of Citations, ISSN 0020-0255, *Information Sciences*, Elsevier, INS, Vol.185 (1): 18-31, February 15, 2012. http://www.sciencedirect.com/science/article/pii/S002002551100418X

[16] Araujo F., Farinha J., Domingues P., Silaghi G. C., and Kondo D. (2011): A maximum independent set approach for collusion detection in voting pools, *Journal of Parallel and Distr. Computing*, Oct 2011. https://doi.org/10.1016/j.jpdc.2011.06.004

[17] J. Benet, N. Greco, D. Dalrymple, M. Zumwalt, E. Miyazono, et al. (2017): Filecoin: A decentralized storage network, 2014-2017. http://filecoin.io/filecoin.pdf

[18] de la Rosa JL el-Fakdi A., Torres-Padrosa V., Amengual X. (2017), A Logo Recognition by Consensus for enabling Blockchain Implementations, *20 th International Conference of the Catalan Association for Artificial Intelligence*, CCIA 2017, November 18, Barcelona.

[19] Aciar S., de la Rosa JL, López Herrera J. (2007): Information Sources Selection Methodology for Recommender Systems Based on Intrinsic Characteristics and Trust Measure, *Intelligent Techniques for Web Personalization and Recommender Systems in E-Commerce*. 2007 AAAI Workshop, July, 2007, Vancouver, British Columbia, Canada.

[20] The witcoin.io whitepaper (2017)

[21] de la Rosa JL, Aguilar J., and Serra I. (1994), Heuristics for Cooperation of Expert Systems. Application To Process Control, ISBN 84-605-0275-9 0, 1994, Ed : PIAR – UdG, Barcelona.

[22] Von Luxburg, U. and Bousquet O. Distance-based classification with Lipschitz functions. *Journal of Machine Learning Research* 5. Jun (2004): 669-695.

[23] Cao, Q., Ying Y. and Li P. (2012), Distance metric learning revisited. En *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, Berlin, Heidelberg, 2012. p. 283-298.

[24] Dong, M., Yang, X., Wu, Y., and Xue, J. H. (2018), Metric Learning via Maximizing the Lipschitz Margin Ratio. *arXiv preprint arXiv:1802.03464*.

[25] Jia, Hong, Yiu-ming Cheung, and Jiming Liu (2016), A new distance metric for unsupervised learning of categorical data" *IEEE transactions on neural networks and learning systems* 27.5 (2016): 1065-1079.

[26] N'Guyen, Steve, Clément Moulin-Frier, and Jacques Droulez. (2013), Decision making under uncertainty: a quasimetric approach, *PloS one 8.12 (2013): e83411*.

[27] Shaw, Blake; Huang, Bert; Jebara, Tony (2011), Learning a distance metric from a network. *In Advances in Neural Information Processing Systems* 2011. p. 1899-1907.

[28] Kyng, K., Rao, A., Sachdeva S., Spielman, D.A. (2015), Algorithms for Lipschitz Learning on Graphs. *JMLR (Journal of Machine Learning Research): Workshop and Conference Proceedings* vol 40:1-34, 2015.

[29] Asadi, Kavosh, Dipendra Misra, and Michael L. Littman (2018): Lipschitz Continuity in Model-based Reinforcement Learning. *arXiv preprint arXiv:1804.07193, 2018*

[30] J.M. Calabuig, H. Falciani and E.A. Sánchez-Pérez, Quasi-Pseudo-Metrics and Extension of Lipschitz-Type Functions in Machine Learning. http://jmcalabu.blogs.upv.es/files/2018/05/QuasiMetricLipsCompMath4mayo18.pdf

[31] J.M. Calabuig, H. Falciani, L.M. Garcia-Raffi and E.A. Sánchez-Pérez, Graph Quasi-Distances and Extension of Semi-Lipschitz Functions in Machine Learning. http://jmcalabu.blogs.upv.es/files/2018/05/GraphQuasiMetricLips4mayo18.pdf

[32] J.M. Calabuig, H. Falciani and E.A. Sánchez-Pérez, Dreaming Machine Learning: Graph Quasi-Distances and Lipschitz-Extensions for Modeling Financial Processes. http://jmcalabu.blogs.upv.es/files/2018/05/QPmetricHerStrategy4mayo18.pdf

[33] J.M. Calabuig, H. Falciani, A. Ferrer-Sapena, L.M. Garcia-Raffi and E.A. Sánchez-Pérez, Graph Distances for Determining Inter-Entities Relations: A Topological Approach to Fraud Detection. http://jmcalabu.blogs.upv.es/files/2018/05/PathDistFraudTh1-5-18.pdf