Reputation Systems for Human-Computer Environments

Anton KOLONIN
Aigents, Novosibirsk State University
Novosibirsk, 630090, Russia
SingularityNET, Stitching
Amsterdam, 1083 HN, Netherlands

ABSTRACT

Understanding the principles of consensus in communities and finding ways to find solutions to the optimal community as a whole becomes crucial as the speeds and scales of interaction in modern distributed systems increase. Such systems can be both socially-information computer networks that unite the masses of people, and multi-agent computing platforms, including peer-to-peer systems such as blockchains, operating on the basis of distributed ledger. Finally, it is now becoming possible for hybrid ecosystems to emerge, which include both humans and computer systems using artificial intelligence. We propose a new form of consensus for such systems, based on the reputation of the participants, calculated according to the principle of "fluid democracy". We expect that such a system will be more resistant to social engineering and reputation manipulation than the existing systems. In this article, we discuss the basic principles and options for implementing such a system, and also present preliminary practical results.

Keywords: Collective Intelligence, Consensus, Distributed Systems, Liquid Democracy, Peer-to-Peer, Reputation, Social Computing.

1. INTRODUCTION

Problem of reliable democratic governance is critical for survival of any community, and it will become more important for communities powered with computer networks speeding up social communications [1,2]. Moreover, for such networks being powered with of Artificial Intelligence (AI) or Artificial General Intelligence (AGI) systems, upon their emergence, this will become even more important. Actually, for systems powered with AI and AGI, it will be much more critical than it applies for human societies – because of speed and scale of electronic communications and low latency in system response not being manageable by human perception capabilities anymore. However, even in human communities, no reliable form of reaching truly democratic consensus is invented over history of human race which is causing lots of harm to human communities as well as dangers for peace worldwide [3,4,5].

The first form of consensus relying on brute force known since ancient societies can be serving to the minority having the access to the force – this problem is now replicated in modern distributed computing system based on Proof-of-Work (PoW) in blockchain environments [6,7]. More advanced form of consensus mostly usable by human race by now is reached on basis of financial capabilities of members of community, and it is known to lead the situation when "reacher become richer" and gain more and more power – this problem is also replicated nowadays in latest developments of distributed and peer-to-peer computing system based on Proof-of-Stake (PoS) in blockchain. In the some of recently introduced blockchain systems, the commonly suggested solution called Delegated Proof-of-Stake (DPoS), which effectively mean that rule on basis of financial capabilities is implemented indirectly, by means of manually controlled voting process to select delegates to conduct the governance of the system – this can be only limited improvement and can nor be implemented in communities employing AI operating at high speeds not controllable by means of limited human capabilities.

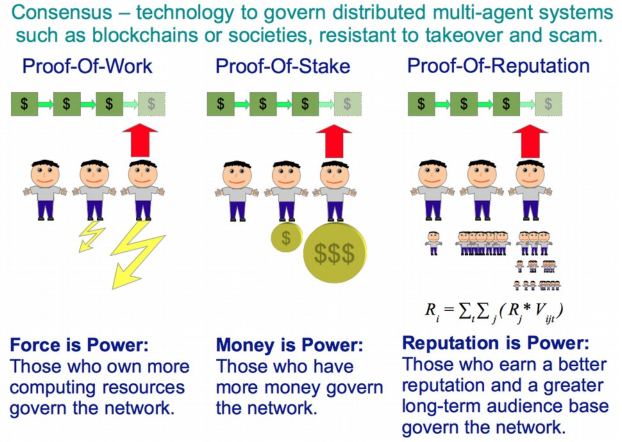

Consensus – technology to govern distributed multi-agent systems such as blockchains or societies, resistant to takeover and scam.
Proof-Of-Work
Proof-Of-Stake
Proof-Of-Reputation
$
$
$
$$$
$R_i = \sum_t \sum_j (R_j * V_{ijt})$
Force is Power:
Those who own more computing resources govern the network.
Money is Power:
Those who have more money govern the network.
Reputation is Power:
Those who earn a better reputation and a greater long-term audience base govern the network.

Fig.1. Types of consensus in distributed systems such as Proof-of-Work, Proof-of-Stake and Proof-of-Reputation.

The described situation leads to the danger that consensus in any AI-powered community may be quickly took over by some AI system or conspired cooperation of such systems hostile to majority of AI systems or humans that are supposed to be served by given community. It may be either because of emergent hostility of an AI system in respect to humans ("Transformative AGI" scenario) or because of particular circle of people intentionally managing given AI system in favor of human minority damaging majority ("Swiss Army Knife AGI" scenario).

To solve the problem, we suggest distributed AI/AGI systems to be based on Reputation Consensus [8,9,10] implementing Proof-of-Reputation (PoR) principle, opposing power of brute force (PoW), power of money (PoS or DPoS). The Proof-of-Reputation makes it possible to implement system of Liquid Democracy opposing and different forms of representative democracy, affected by power of money as we know in human history and direct democracy, not possible for implementation technically at scale of modern real-world communities, human or artificial. The Reputation Consensus principle states that governing power of member of a human or artificial society depends on Reputation of the member computed on basis of the following principles.

1) The first key principle is **liquid nature** of the Reputation values so the Reputation may be computed by means of different measures performed by all members of community in respect to one who Reputation is being computed for, **with account to Reputations of all of the other members** themselves.
2) The second key principle is **temporal scoping of the Reputation**, so reputation measures collected by member in the past are less contributing to current Reputation of the member than the latest ones, which make more impact.
3) The third key principle if **openness of all Reputations of all members** and the measures that they perform so audit of Reputations and the historical measures over the history can be analyzed in order to prevent Reputation cheating and gaming.
4) The fourth key principle is **precedence of human measures over artificial**, so measures provided by human participants of a hybrid communities have unconditional precedence over measures provided by AI systems, if they are also capable to contribute to evaluation of humans and artificial entities in a community.

Multiple measures contributing to evaluation of Reputation may be considered, depending on implementation of a given Reputation system. Applicability of the measures may depend on accuracy and reliability that they may provide as well as resistance to attack vectors targeting takeover of the consensus by means of reputation cheating and gaming. Primarily, we consider such measures as the following: a) members staking financial values on other members; b) members providing explicit ratings in respect to transactions committed with other members; c) the financial values of transactions between the members considered as implicit ratings; d) textual, audial and video reviews made by members in respect to other members or transactions between them.

We consider the problem of Consensus for distributed community appears so important because centralized and even decentralized solutions involving AI/AGI systems may happen to be targeting to fulfill interests of limited group of powerful and resourceful humans eventually, and therefor unlikely be serving interests of humanity as a whole.

2. COMPUTATIONAL APPROACH

Overall computational framework that we suggest to implement is describe in the earlier works [8,9,10]. In this paper we would consider two specific case studies described below.

Iterative Liquid Rank

This version of the Reputation computation algorithm may be applied to relatively small graphs of interactions in temporarily scoped time frame, so the reputation state can be updated for entire set of known interactions iteratively, according the following formula, being simplified case of the one used in the earlier work [10].

$$S_{i,k+1} = \Sigma_j(S_{j,k} * R_{i,j}) \tag{1}$$

In the formula above, reputation rank S is being updated iteratively across all agents i in a multi-agent system for every other agent j that have any ratings $R_{i,j}$ issued, having the rank of the latter agent $S_{j,k}$ known from the previous iteration k so the ranks of every agent in the system is updated on iteration $k+1$. If the ratings are implicit, like financial values $F_{i,j}$ of the transactions recorded in the system, logarithmic non-negative ratings can be used, as follows. The use of logarithm is justified by that real amounts of financial transactions may have significant spread of values and then few random highly valuable transactions can make all other transactions negligible, so logarithm can make impact of absolute value less significant compared to the fact of the transaction.

$$S_{i,k+1} = \Sigma_j(S_{j,k} * log_{10}(1+F_{i,j})) \tag{2}$$

At the end of the each iteration, the obtained new set of reputations may be normalized accordingly to partial normalization, assuring all reputations are residing in range from *0.0* to *1.0*.

$$S'_{i,k+1} = S_{i,k+1}/MAX_i(S_{i,k+1}) \tag{3}$$

Optionally, full normalization may be employed stretching the range of possible reputation values to the range between *0.0* and *1.0*, so minimum values are guaranteed to stay at value of *0.0*.

$$S'_{i,k+1} = (S_{i,k+1}-MIN_i(S_{i,k+1})) /(MAX_i(S_{i,k+1})-MIN_i(S_{i,k+1})) \tag{4}$$

Under this implementation pattern, the iterative process starts with some default reputations assigned to all members, as configured by system parameters. The process continues with standard deviation between previous and latest reputation values computed till the standard deviation gets below the configured threshold.

Incremental Liquid Rank

This version of the Reputation algorithm may be applied to large graphs operating in real-time environments when the time constraints are critical. Also, this version may organically incorporate decay of past Reputation values over the time. In this version, the formula similar to (1) or (2) computes not projected iteration for iteration k, but incremental update of the reputation for the time period between previous moment of time t and current moment of time $t+1$.

$$dS_{i,t+1} = \Sigma_j(S_{j,t} * R_{i,j}) \tag{5}$$

$$dS_{i,t+1} = \Sigma_j(S_{j,t} * log_{10}(1+F_{i,j})) \tag{6}$$

Further, the incremental reputation $dS_{i,t+1}$ can be normalized accordingly to either (3) or (4) to $dS'_{i,t+1}$. After then, the incremental reputation for time $t+1$ may be further blended with the earlier one at time t, based on parameter C called "conservatism" in range *0.0* to *1.0* exclusively, so values close to *1.0* would mean that previously earned reputations would decay slower. That means, C defines how long perviously earned reputation stay relevant, being gradually updated with more recent ratings.

$$S_{i,t+1} =dS_{i,t+1}*(C-1)+S_{i,t}*C) \tag{7}$$

Specific simplified implementation of the algorithm may assume the values of the ratings do not depend on the raters, so the liquid nature is dismissed. In such case, $S_{j,t}$ in formulae 5 and 6 can be considered equal to *1.0*.

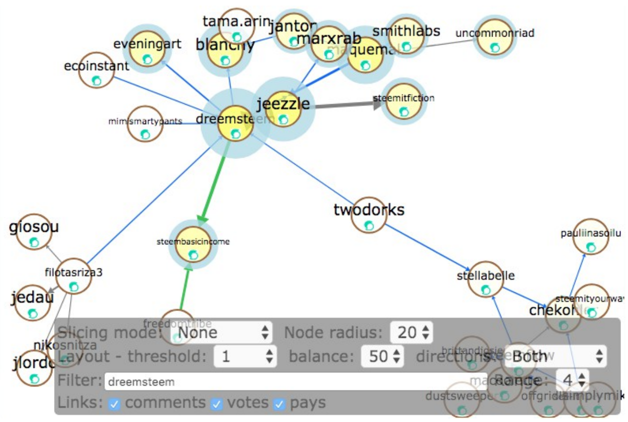
3. CASE STUDIES

The following two case studies were performed based on data from public blockchain Steemit, serving social network https://steemit.com/, serving human authors as well as automated bot accounts. There are multiple interaction types between alive users and automated bots available, with few of them used as ratings for computation of network connectivity graphs [3] and respective reputations [10]: a) textual comments on the posts as well as comments on comments, with logarithm of number of characters in the post used as implicit rating value; b) vote for post or comment of one user by another user, with logarithm of the voting power used as explicit rating value; c) financial transaction between wallets owned by users with logarithm of transaction value used as implicit rating.

The implementation of reputation system in both cases has been done with open source platform for personal social analytics called Aigents (https://github.com/aigents/aigents-java).

Case 1: Social Sub-Graph Querying and Rendering

In this case, **Iterative Liquid Rank** reputation algorithm has been used for two purposes – selecting subgraphs from the entire network-wide interaction using the "graph query" and performing stable deterministic arrangement of nodes on the visual graph representation.



Slicing mode: None
Node radius: 20
Layout - threshold: 1
balance: 50
directions: Both
Filter: dreemsteem
Range: 4
Links: ☑ comments ☑ votes ☑ pays

Fig.2. Subgraph of Steemit social graph computed based on all types of interactions (comments, votes and financial transactions) selected from "seed" node in the center, "hop limit" 4 and top 40 highly-reputable (within given subgraph) nodes selected, with color of the nodes and positions of nodes on the graph corresponds to reputation level. Relative value of the ranking relationship is indicated by link width.

The first problem takes place for the Aigents graph querying when subgraph is extracted based on basis of few "seed" nodes, types of links connecting the nodes and number of graph "hops" from the seed nodes along the links. This is the typical case when amount of links in the graph resulting query bursts exponentially with number of "hops", and either rendered graph becomes unreadable or, if it is being rendered with help of JavaScrip client on Web browser side, the browser hangs because of memory overuse. For such case, it is useful to limit the resulting graph with only the topmost important nodes to be retrieved and using reputation for computing the node importance is one of the option, which has been implemented.

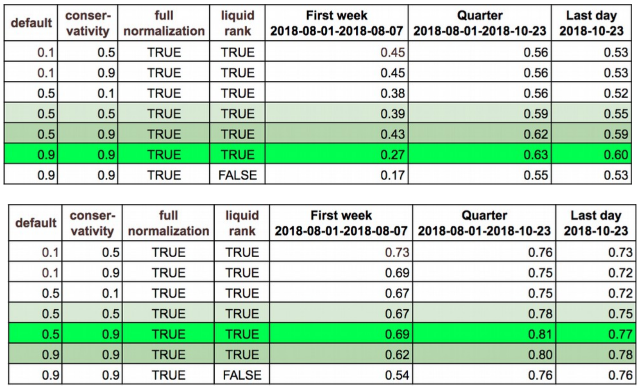
Once the subgraph is retrieved from the Aigents server, it is rendered in Aigents Web browser client and the same ranking algorithm is used to compute saturation of the node color and its relative placement on the graphical view layout, so the more highly-reputable nodes are generally placed higher to the top of the graph, as shown on the Fig.2.

Using such interactive graph querying interface combined with interactive graph visualization has appeared to be useful tool to study society and communication structure of online community on Aigents Web site https://aigents.com/.

Case 2: Computing System-wide Reputation Levels

In this case, **Incremental Liquid Rank** reputation algorithm has been used to explore possibility of its use for detection of either highly useful accounts that can be considered as either top option leaders to be followed or rather spam and trolling sources that should be avoided.

The period of almost three months has been chosen for study – in range from 2018-08-01 and 2018-10-23 and entire network activity in respect to commenting, voting and cryptocurrency transfers has been used as implicit or explicit ratings for the reputation system. To make the experiment measurable, the control sample of top *500* "highly reputable" Steemit accounts has ben created from https://steemwhales.com/ resource and another recent *500* "low reputable" banned and blacklisted highly likely spam accounts compiled from https://steemit.com/buildawhale/@buildawhale/wnk6b-buildawhale-blacklist-update account blog. For control purposes, the "highly reputable" accounts were assigned reputation values of *1.0* and the "low reputable" ones were given *0.0*.



default	conser- vativity	full normalization	liquid rank	First week 2018-08-01-2018-08-07	Quarter 2018-08-01-2018-10-23	Last day 2018-10-23
0.1	0.5	TRUE	TRUE	0.45	0.56	0.53
0.1	0.9	TRUE	TRUE	0.45	0.56	0.53
0.5	0.1	TRUE	TRUE	0.38	0.56	0.52
0.5	0.5	TRUE	TRUE	0.39	0.59	0.55
0.5	0.9	TRUE	TRUE	0.43	0.62	0.59
0.9	0.9	TRUE	TRUE	0.27	0.63	0.60
0.9	0.9	TRUE	FALSE	0.17	0.55	0.53

default	conser- vativity	full normalization	liquid rank	First week 2018-08-01-2018-08-07	Quarter 2018-08-01-2018-10-23	Last day 2018-10-23
0.1	0.5	TRUE	TRUE	0.73	0.76	0.73
0.1	0.9	TRUE	TRUE	0.69	0.75	0.72
0.5	0.1	TRUE	TRUE	0.67	0.75	0.72
0.5	0.5	TRUE	TRUE	0.67	0.78	0.75
0.5	0.9	TRUE	TRUE	0.69	0.81	0.77
0.9	0.9	TRUE	TRUE	0.62	0.80	0.78
0.9	0.9	TRUE	FALSE	0.54	0.76	0.76

Fig.3. Pearson correlation (above) and accuracy (below) measures computed for system-wide reputation ranks evaluated with incremental liquid rank algorithm against control list of 100 accounts created manually.

The reputations were computed incrementally for every day of the entire period with different reputation system parameters, such as default reputation, conservatism, using full normalization or not and using current reputations of raters ("liquid rank") or not. Three network-wide distributions of reputation values were created: a) average reputations for the first week of the period, average reputations for the entire period and latest reputations at the end of the period. These distributions were compared against list of *1000* control accounts, and the two two correspondence metrics were computed across the lists. First, Pearson correlation was evaluated between the lists for every matching account pair. Second, assuming value above system-wide reputation average can be considered "highly reputable" one while the value below can be thought as "low reputable one", the accuracy measure was computed. Both evaluations are presented on Fig.3. The results are showing that the best reputation matching is achieved with default reputation *0.5-0.9*, conservatism *0.9* (called "conservativity" on Fig.3.) and use of the "liquid rank", with accuracy as high as *0.81* and highest positive Pearson correlation at *0.63*. As it have been expected, more close matching has been found closer to the end of the exploration period, when the distribution of reputation is stabilized after initial assignments of default values.

In order to study temporal dynamics of the reputation values, we have also studied how the reputation changes over time for accounts of different types, assuming every account starts with default reputation of *0.5* which may get changed to higher or lower in the same very first day and keep changing over time, as it is shown on Fig,4. The interesting feature of the dynamics is that "expectedly highly reputable" accounts are given longer "tails" spanning over time so reputation either does not decay or decay slower. On the opposite, the "expectedly low reputable accounts" are present with fast reputation value decay. Should be noted, that highly reputable accounts do not necessarily have to get reputation decayed closer to the end of the period – it just have happened that all random accounts selected for the chart were losing reputation to some extent by the end of given time period.

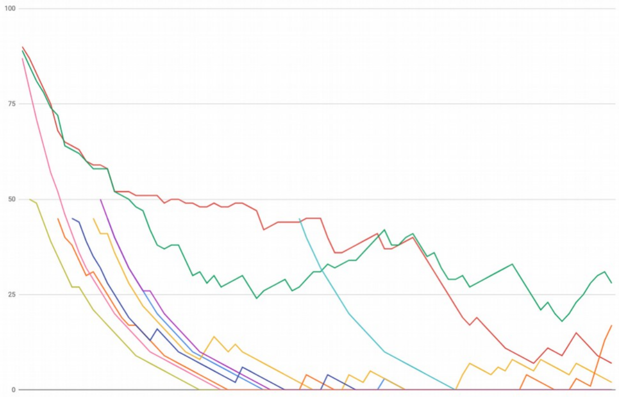


Fig.4. Temporal dynamics of reputation values for randomly selected *5* "expectedly highly reputable" accounts and *5* "expectedly low reputable" ones. Horizontal axis corresponds to time period of *3* months from left to right, vertical axis indicates reputation value in range from *0.0* to *1.0*, labeled on scale from *0* to *100* on the chart.

4. CONCLUSIONS

The presented approach and algorithmic framework appears to be practically usable for explored use cases – interactive graph querying and visualization of local social subgraphs and for studying of the large social networks state and dynamics at whole.

There is a range of possible applications that can benefit using proposed reputation system: social networks and human resource management for search for better interpersonal connections, recommendation systems for services, products and goods, marketplaces for connecting suppliers and consumers, collaborative decision making support systems based on collective intelligence, electronic democracy for achieving consensus based on "liquid democracy" in online and offline communities, multi-agent system employing AI or AGI for safer and controllable consensus within such systems, hybrid human-computer ecosystems involving humans as well as agents powered by AI/AGI to ensure safe consensus based on human-driven valuations.

In the future work, we are going to study how different types of interactions considered separately can make reputation structure and dynamics more precise and useful in specific applications. The other promising possibility of the Aigents technology employed in this work is use context-specific studies so that votes and comments associated with specific tag or textual context identifying particular are of interest or subject domain. Further, more kinds of social network environments may be involved in the study and more precise fine-tuning of the reputation system parameters may get required for each of these.

We believe that practical development of such principles for automation of decision making processes and reaching consensus in distributed systems would become necessary condition for the emergence of collective intelligence in hybrid human-computer ecosystems involving humans as well as AI/AGI components. Such principles can be also seen as a core component of the next generation of computational systems employing social computing.

5. ACKNOWLEDGMENTS

Our acknowledgments are going to Alexey Muravev (https://steemit.com/@dark.sun) who has initiated, enabled and supported the work on social analytics applied to Steemit and colleagues at SingularityNET – Ben Goertzel, Cassio Pennachin, Simone Giacomelly, Mitchell Loureiro, Marco Argentieri, Debora Duong and Nejc Znidar being collaborators in the work on reputation system for SinglularityNET platform and marketplace.

6. REFERENCES

[1] G. Swamynathan , K.Almeroth, B.Zhao , **The design of a reliable reputation system**. Electron Commer Res 10: 239–270, DOI 10.1007/s10660-010-9064-y, 31 August 2010, pp.239-270.

[2] M. Gupta, P. Judge, M. Ammar, **A Reputation System for Peer-to-Peer Networks**. NOSSDAV'03, June 1–3, 2003, Monterey, California, USA, ACM 1-58113-694-3/03/0006, 2003.

[3] E. Garin, R. Mescheriakov, **Method for determination of the social graph orientation by the analysis of the vertices valence in the connectivity component**. Bulletin of the South Ural State University, Ser. Mathematics. Mechanics. Physics., 2017, vol.9, no.4, 2017, pp.5-12.

[4] A. Kolonin, D. Shamenkov, A. Muravev, A. Solovev, **Personal analytics for societies and businesses with Aigents online platform**. 2017 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON) - Conference Proceedings, 2017.

[5] A. Kolonin, **Assessment of personal environments in social networks**. Data Science and Engineering (SSDSE), Siberian Symposium Proceedings, INSPEC Accession Number: 17262023, DOI: 10.1109/SSDSE.2017.8071965, Publisher: IEEE, 2017.

[6] L. Lamport, R. Shostak, M. Pease, **The Byzantine Generals Problem**. ACM Transactions on Programming Languages and Systems, Vol. 4, No. 3, July 1982, 1982, pp.382-401.

[7] C. Cachin, M. Vukolić, **Blockchain Consensus Protocols in the Wild**. 31st International Symposium on Distributed Computing (DISC 2017), Editor: Andréa W. Richa; Article No. 1; pp. 1:1–1:16, Leibniz International Proceedings in Informatics, 10.4230/LIPIcs.DISC.2017.1, 2017.

[8] Aigents, **Proof of reputation as Liquid Democracy for Blockchain.** Steemit 2017. https://steemit.com/blockchain/@aigents/proof-of-reputation-as-liquid-democracy-for-blockchain

[9] A. Kolonin, **Reputation System Design for SingluarityNET**. Medium, 2018. https://medium.com/@aigents/reputation-system-design-for-singularitynet-8b5b61e8ed0e

[10] A. Kolonin, Ben Goertzel, Deborah Duong, Matt Ikle, **A Reputation System for Artificial Societies**, arXiv:1806.07342v1, 2018.