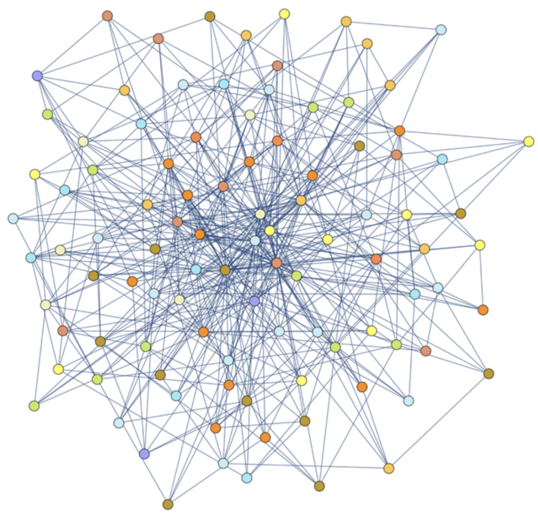
# Simulating the Quality of Community Decisions
# DF4B-RFP2
# Simulation of Voting Methods
# Milestone 3 Report




Principal Investigator: Kenric Nelson

Project Manager: Juana Attieh

Photrek, LLC

admin@photrek.io


28 June 2024

## Executive Summary

## Milestone Description and Deliverables

Simulation of Voting Methods
Budget: $ 11,080

### Milestone Description:

- Quadratic Voting and No-Cost Grading implementation in computer simulations.
- Comparison of the voting dynamics and outcomes between with and without quadratic costs for conviction voting.
- Red team review of Photrek's M2 deliverables.

### Deliverable Description:

- Fully implemented simulations of quadratic voting and No-cost Grading.
- Complete and detailed design of the plural voting model capable of realistically simulating multi-agent interactions and social.
- Comparison of Plural Voting with No-Cost Grading.
- Comprehensive evaluation by red team experts of the plural voting prototype design and Monte Carlo modeling, as presented in Milestone 2.

## Milestone Accomplishments

## Voting Simulation Model: Getting insights from deep funding data

We analyze the community review and funding request of 201 proposals of deep funding, including awarded and open proposals for all pools. (We ignored proposals with no community review). The key idea is that the proposals' information and the community ratings influence the rating of other individuals who are reading the proposals. This interaction between community members establishes clusters of sentiments or opinions about the proposals [1-5].

### Proposal funding request distribution

We show the distribution of funding the proposals requested on the figure below. The distribution follows a negative exponential density probability: the majority of the proposals ask for less than 20 thousand dollars, but there is a non-negligible fraction of proposals asking for more than 100 thousand dollars for funding.

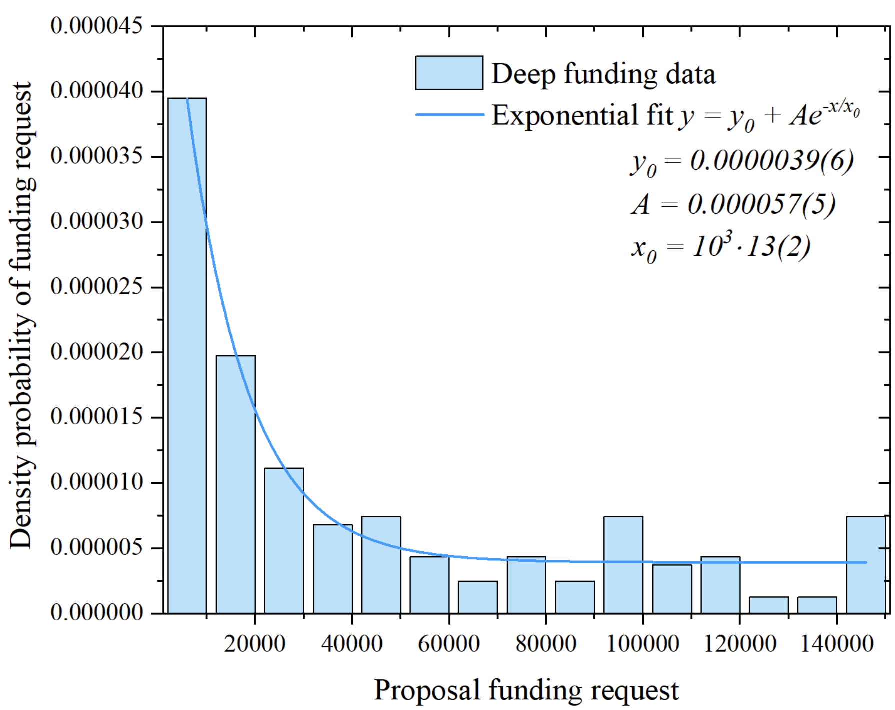



**Figure 1.** Distribution of proposal funding requests using real-world data from the Deep funding platform from rounds 1 to 4. We executed an exponential fitting with this distribution and used the estimated parameters of the exponential function to model the funding request distribution in the simulation.

## Proposal community review distribution

The distribution of community review on proposals follows a Lorenz distribution with a peak around four stars. There are very few proposals with less than 3 stars, probably because people avoid submitting bad proposals. On the other hand, everyone wants to submit excellent proposals, but this is hard, so the majority of proposals have 3.5 to 4.5 stars.

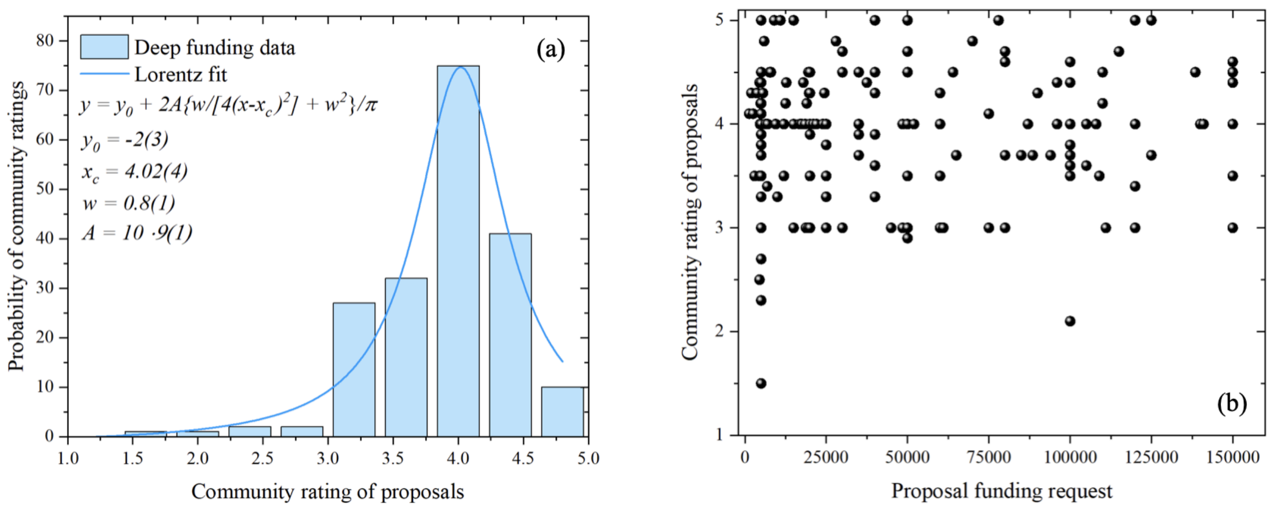



**Figure 2.** In (a), we show the Distribution of the average community rating on the proposals using real-world data from the Deep funding platform from rounds 1 to 4. We executed a Lorenz fitting to model the Distribution. In (b), we show that, in general, the proposal's funding request does not affect the average community review of the proposals. Therefore, in our simulations, the proposal's funding request does not influence how people will vote.

## Relation between Proposal funding request and the community review

How does the funding request the proposal asks affect the community rating on these proposals? The figure below shows that, in general, there is no correlation between the proposal funding request and the community average rating. We see that the majority of the ratings below 3 stars are concentrated on proposals with low funding requests. However, this effect is weak and overall, the proposal funding request does not affect the community review on the proposals.

## Agent's credit distribution

Here we analyze the balance of 182 users of deep funding. The distribution is strongly unequal, with the top wallet holding 86% of the tokens! The top 5 wallets hold 95% of the tokens and the top 16 wallets hold 99% of the total balance.

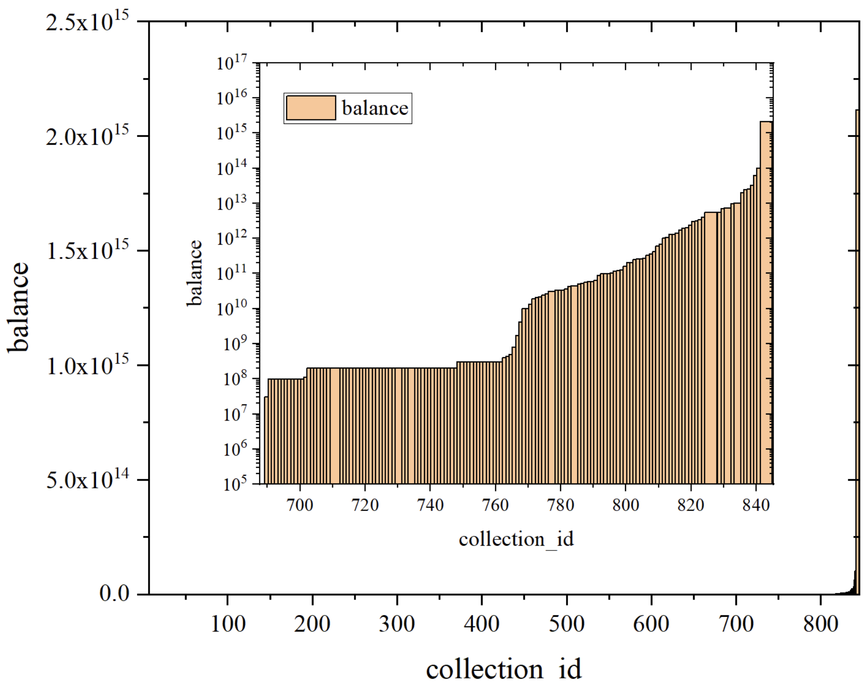



**Figure 3.** Distribution of user balances on the Deep Funding platform. The distribution is highly skewed, with the top wallet holding 86% of all tokens among the collection_ids. The inset shows the same plot with a logarithmic scale on the y-axis for better visualization.

The distribution of tokens balances appears to be formed of 2 independent distributions: the first represents the 39 individuals with zero balance, and the second is a log normal distribution of agents investing their money in the platform, displaying the heavy tail phenomena [6].

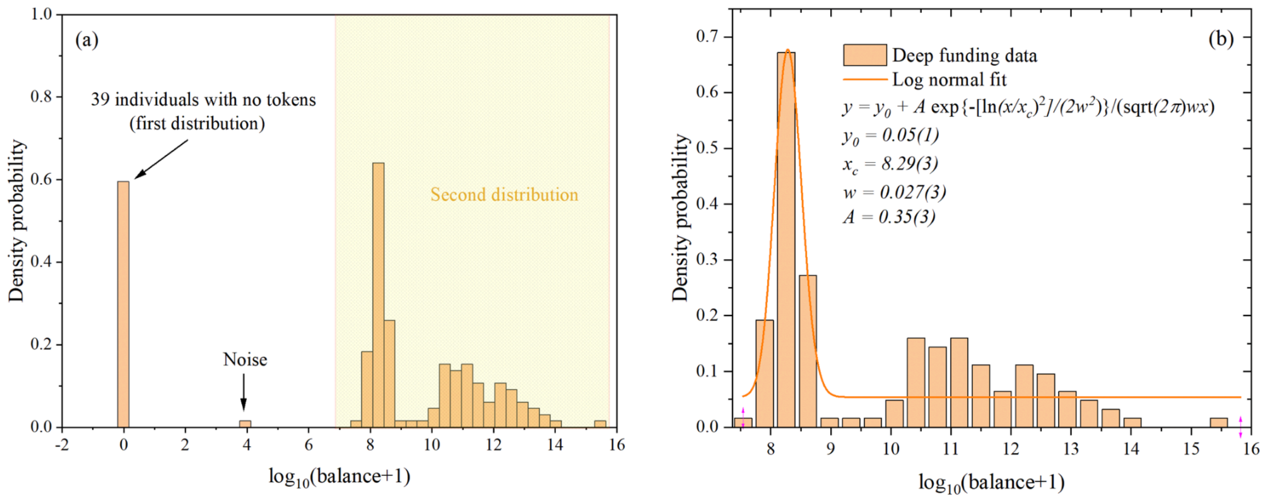



**Figure 4.** Distribution of the balance of each wallet holds using real-world data from the Deep funding platform. We executed a Log normal fitting with this distribution and used the estimated parameters of the Log normal function to model the credits distribution in the simulation.

To represent the entire distribution, we say an agent has a probability 39/182 of having 0 balance; otherwise, his credit is determined by the log-normal distribution fitted above. However, we verified that several people with zero balances actually have a balance but cannot connect their wallets because they cannot vote. Since they cannot connect their wallets, the system shows they have zero balance. Other individuals, however, really have zero balance and voted. Unfortunately, these individuals do not yet understand how the voting system of SingularityNET works. We assume this is a temporary phenomenon/problem. Hence, in the simulation, we will ignore individuals with zero balance.

## How do individuals rate proposals?

When people analyze one proposal, only a small fraction of them publicly record their opinion and ratings. We do not have access to the rating of all users. However, we can construct a model to simulate the interaction between community members and the sentiments of the individuals and their strength of opinions on the projects.

## Voting Simulation Model: Modeling the social dynamics in proposals ratings

We consider a system of N individuals voting on Np proposals. When one proposal is created on deep funding, it has no reviews. One individual may read the proposal with probability 1 - Pskip, and will get a sentiment regarding the proposal. With probability Pposting, the individual rounds his rate to an integer from 1 to 5 stars and posts his rating so the whole community can see his opinion.

Note that since the rating is rounded, there is uncertainty about his true opinion. For example, a rating of 4 stars can represent an individual with an opinion of 3.8 stars or 4.2 stars.

After the first individual posts his opinion regarding the proposal, his opinion will affect the opinion of the next agents that will read the proposal. The next individuals will have a tendency to shift their opinions closer to the average community review due to the herd effect. However, some individuals can have a stronger opinion and display social nonconformity, disagreeing with the community review.

### Inertia effects

Some individuals are more susceptible to social validation than the others. To capture that effect, we introduce the concept of the agent's inertia to model the individual nonconformism or uncertainty about the average community review [7].

For example, consider the agent i read the proposal p and form an opinion u(i, p) (generated at random from a continuum spectrum from 1 to 5 stars) about the value of that proposal. However, he sees that there is an average community review U(n, p) of n individuals of that proposal p. Then, with probability 1- I(i, n), the user i conforms with the average community review, and his opinion u(i, p) get closer to the average community review U(n, p), achieving the final opinion u'(i, p) given by

$$u'(i,\ p)\ =\ u(i,\ p)\ +\ [1\ -I(i,\ n)][U(n,\ p)\ -u(i,p)].$$

On the other hand, with probability I(i, n), the individual i disagree with the average community, and his opinion u(i, p) goes away from the average community review U(n, p), achieving a final opinion u'(i, p) given by

$$u'(i,\ p)\ =\ u(i,\ p) -\ I(i,\ n)/[U(n,\ p)\ -u(i,p)].$$

Note that if the inertia of the individual i is 1, the agent ignores the average community review. On the other hand, if the inertia of the agent is 0, the agent ultimately consents to the previously posted opinions and his final opinion will be equal to the average community rating U(n, p). Most agents might have an inertia between 0 and 1 and will stochastically agree or disagree with the community in each proposal. Note that the final opinion u'(i, p) is also bounded between 1 and 5 stars.

### Social pressure

Now, 10 ratings of 5 stars are far more convincing than 2 ratings of 5 stats. To simulate this effect, we introduce a social pressure parameter Sp and model the decayment of the agent's i initial inertia $I_o$ in function of the quantity n of ratings:

$$I(i, n) = I_o(i)(1 - S_p)^n.$$

For example, if the social pressure is 1%, then the inertia of agent i decreases 1% for each new posted rating, increasing the social validation phenomena.

### Cost distribution of proposals and credit distribution of agents

To each proposal p, we assign a funding request F(p) distributed using the exponential distribution function fitted with the deep funding data.

Similarly, we generate the balance b(i) of each agent i using the log normal density probability fitted with the deep funding data. This ensures we will use realistic distributions in our simulations.

Finally, we set the round funding equals to the average funding of the rounds 1 from 4 of deep funding, which is $ 1,137,500.

### Inertia distribution

To better understand the distribution of inertia along the community, it is worth reviewing the voting analysis articles from Jan Horlings about real deep funding voting. In the article entitled "DFR1 – Voting Analysis Part 1 – Number Crunching" [8], Jan Horlings concluded that "[...] What is interesting is that the smaller wallets seem to be more forgiving or optimistic, since the number of eligible projects is much larger. The difference in outcomes is especially clear in Pool A. One possible explanation is that, with higher stakes, the ‘business model’ for buying tokens just for the sake of voting might also be more interesting(!).".

Therefore, we see that, in general, users with more balance display more nonconformity and are more exigent than individuals with smaller wallets. Indeed, if someone sees that his opinion is not well represented by the average community review, he/she may feel the necessity to invest more tokens in the platform and have a higher voting weight, so that his opinion can be represented. In other words, nonconformity can lead to higher balances.

To model this, we assign to agent *i* an initial inertia value $I_o(i)$ and consider the 'business model' where the competition between agents to have a higher voting weight is an indication of higher inertia. Hence, we calculate the business effect on the initial inertia of each agent i as

$$I_o'(i) = I_o(i)[1 + k\,N(b < b(i))/(N - 1)],$$

Where N(b < b(i)) is the number of agents with less balance than user i and k is a proportionality constant. Observe that this initial inertia value $I'_o(i)$ might also decrease over time due to the social pressure effect described previously.

## Voting Simulation Model: Voting mechanisms.

Every agent i expresses his strength of opinion by assigning a grade to each proposal based on his rating on that proposal. We map the grade ranging from 1-10 from the stars ranging from 1-5 in a linear fashion. For example, 1 star is equal to grade 1, 5 stars is equal to grade 10 and 3.7 stars is equal to grade 7:

$$grade(i, p) = round[(9star(i) - 5)/4].$$

With that, we can calculate the average grade of each proposal p for each voting mechanism. Eligible proposals to be awarded must have received votes from at least 1% the total number of wallets and received at least 1% of the total community balance under quadratic voting. Additionally, the project needs a minimum average grade of 6.5 and must fit the pool's remaining budget after higher voter proposals are funded.

## SingularityNET square root weight

In the singularityNET system, the grade each user gives to each proposal has a weight given by the square root of the user balance. The average grade of one proposal p is then calculated by a weighted arithmetic mean:

$$average\ grade(p) \ = \ \sum_i grade(i,\ p)\sqrt{b(i)}/ \sum_i \sqrt{b(i)}.$$

## One Wallet, One vote (1W1V)

In this voting mechanism, each user has the same weight and the average grade of the proposal p is simply the arithmetic mean of the grades of the users:

$$average\ grade(p) \ = \ \sum_i grade(i,\ p)/N.$$

## One coin, One vote (1C1V)

This system is similar to the SingularityNet voting rule, but without the square root:

$$average\ grade(p) \ = \ \sum_i grade(i,\ p)b(i)/ \sum_i b(i).$$

## Plurality voting with quadratic costings

In this system, the users pay a quadratic costing to vote on each proposal. Each voter concentrates their influence on proposals of interest rather than evaluating all proposals. We consider each agent to invest a fraction of his tokens in each proposal that is proportional to the cost of the grade the user gives to that proposal. We consider grades to cost more as they become more extreme. For instance, the set of grades {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} has a cost {5, 4, 3, 2, 1, 1, 2, 3, 4, 5}. The proposals of interest must have grade at least equal to 7, and each voter can consider the top T proposals with higher grade as his set of proposals of interest.

Hence, the total number of votes of a proposal p is given by

$$total\ votes(p) \ = \ \sum_i \sqrt{Cost\ grade(i,\ p)b(i)/\sum_P Cost\ grade(i, P)}.$$

Where $Cost\ grade(i,\ p)$ is the cost of the grade the user i gave to proposal p, and the sum $\sum_P Cost\ grade(i, P)$ runs along proposals of interest to i and the average grade of the proposal is just the arithmetic mean of the grades each user i gave to each proposal p.

## Voting Simulation Model: Voters' Aggregate welfare.

To measure if the election results are good for the overall voter, we calculate the aggregate voters' welfare of the system [9-12]. To capture the level of satisfaction of an individual i with a funded proposal p, we calculate the utility of i considering the grade of i to proposal p and how much tokens i invested in the platform. Hence, we define the welfare of an agent i as

$$w_i = \sqrt{grade(i,\ p)b(i)/[\sum_P grade(i, P) \sum_i b(i)]}\,. \qquad (1)$$

We note that in this first iteration of the simulation we are assuming that voter's sentiment about a proposal matches what they submit as a grade, therefore this will favor SingularityNET's current voting method; nevertheless, this simple model is helpful to test the simulation methods. To compute the aggregate voters' welfare, we can calculate the arithmetic mean of the individual welfares:

$$W = \sum_{i=1}^{Np} w_i /N. \qquad (2)$$

Alternatively, we aggregate the welfare using the geometric mean so that the aggregate welfare is highly sensible to the worst individual welfare:

$$W = \sqrt[\sum_{p=1}^{Nfp} N_v(p)]{\left(\prod_{p=1}^{Nfp} \prod_{i=1}^{N} w_i\right)}\,. \qquad (3)$$

Where $\prod_{p=1}^{Nfp}$ and $\prod_{i=1}^{N}$ runs along the funded proposals and the agents, respectively and $\sum_{p=1}^{Nfp} N_v(p)$ is the sum of the number of people who voted on each funded proposal.

Figure 5 compares the aggregate voters' welfare of the SingularityNET weight, 1C1V, 1W1V and plurality voting with quadratic costs. In 5(a), we show the arithmetic welfare and see that all voting systems' aggregate welfare undergoes a second-order phase transition. However, the SingularityNET method and 1C1V also show a recovery phase where the welfare increases again before stabilizing. In this scenario, 1C1V provides the most robust aggregate welfare.

However, let's consider the geometric mean in (b). The community welfare has a peak curve for all voting mechanisms, with a maximum value around the optimum inertia value of 0.1 in this specific scenario. Hence, welfare would be maximized for some community balance between agreeableness and social nonconformity. For low inertia, the behavior of the systems is similar. Still, as nonconformity increases, the square root weighting method from SingularityNET displays more robustness than the other systems, yielding significantly higher aggregate welfare.

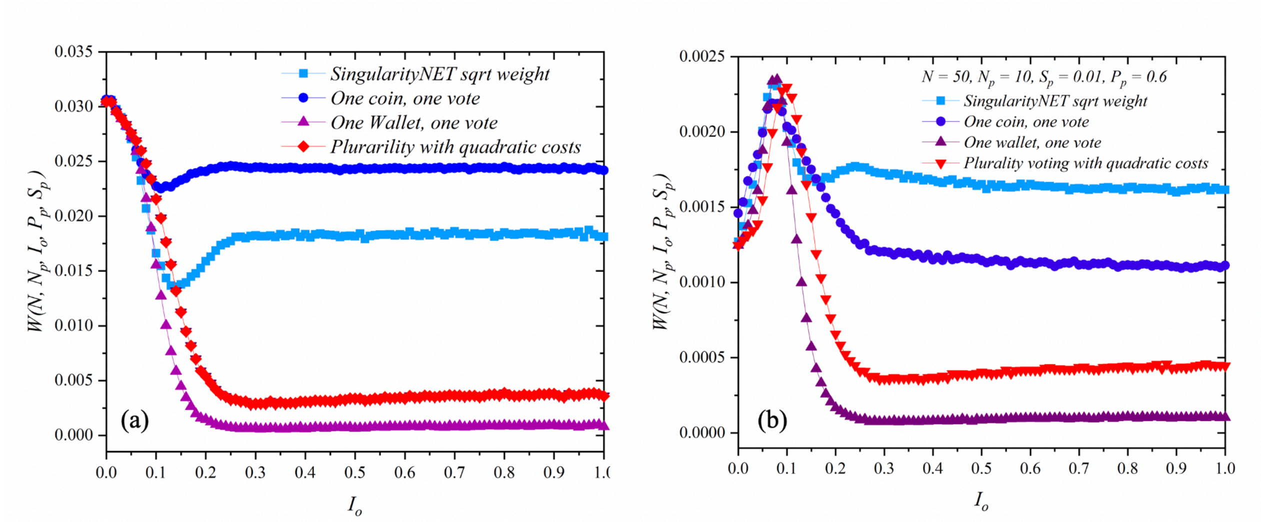



**Figure 5.** Here we show how the aggregate voters' welfare of several voting systems changes with the social inertia value of the voters. For low nonconformity values, the behavior of all systems is similar, but as the inertia increases, each voting system yields different aggregate welfare values. In (a) we compute the arithmetic welfare, and in (b) the geometric aggregate welfare. No conclusions should be drawn from this initial result, since there was no attempt to explore the impact of different voting strategies.

Here we used a community with 50 voters, 10 proposals, social pressure of 1% and Probability of posting of 60%. We shall investigate how these variables can also affect the aggregate community welfare of the voting systems and their robustness against social nonconformity.

## Research Highlights

We began our analysis by defining the aggregate voters' welfare using Equations (1) to (3). However, this approach may not be the most effective for calculating and comparing welfare across different voting systems and strategies. For instance, the results might reflect that the SNET voting aligns with the welfare function rather than the expected behavior. The higher performance of 1C1V and the low robustness of 1W1V might arise from considering the agents' balance when computing welfare, favoring 1C1V over 1W1V.

Therefore, we should explore alternative methods for measuring aggregate welfare that best capture the individuals' welfare rather than the welfare of the coins or wallets. We must establish a proper universal aggregate welfare function or use distinct welfare expressions tailored to each voting system. If we adopt different methods for calculating welfare, ensuring they remain comparable across the various voting systems is crucial.

## Voting Simulation Code

The code is a simulation of a social voting and funding model. It simulates the behavior of individuals rating and voting on proposals with a variety of factors such as social inertia, social pressure, and funding distribution. The simulation aims to model how different voting mechanisms and social factors affect the funding and rating of proposals in a community. It uses a mix of predefined parameters and random distributions to simulate realistic variations in individual behavior and proposal success.

The code and his description can be accessed in the Community Decisions Simulation GitHub repository link by clicking here and accessing the folder "Milestone 3".

Below we reproduce a detailed description of the code and its functionalities:

**Main Simulation Workflow**
The overall simulation workflow involves:

- Initializing system parameters and distributions.
- Simulating proposal dynamics and individual ratings.
- Casting votes using different methods.
- Calculating results, determining the most voted and funded proposals and calculating the voters' aggregate welfare.

**System Parameters**
The code defines several constants that are used throughout the simulation:

- Number of individuals (N).
- Number of experiments (Nxps).
- Backup rate (backUpRate). (minimum number of experiments to write data in a text file)
- Number of stages for threading (Nstages).
- Social inertia parameters (AverageInertiai, AverageInertiaf, dAverageInertia).

- Inertia proportional constant (InertiaProp).
- Probability of posting a rating (Pposting).
- Social pressure parameter (socialPressure).
- Number of proposals available for voting (Nproposals).
- Probability of skipping a proposal (Pskip).
- Maximum number of proposals to vote on using plurality voting (NmaxProposalsToVote).
- Funding parameters (funding money, yoL, xc, w, AL, yo, A, xo, minFunding), variables fitted using real data to generate realistic distributions of credit and proposal funding requests.

**Global System Variables**

Several arrays and variables are defined globally to store the state of the simulation, such as:

- Inertia, rating, proposalRating, NumberRatings, IndividualVotes, money, funding, proposalVotes, Impact
- Variables for sums and averages for calculations (WsqrtSum, WoneCoinSum, WoneWalletSum, WpluralitySum, WLinearpluralitySum)
- Variables for recording results (favoriteProposals, proposalsToVote, MostVotedProposals, NfundedProposals, fundedProposals)

**Random Number Generation**

A custom random number generator (xor64()) and functions for generating log-normal and exponential distributions are defined to simulate randomness in the model using the Inverse Transform Sampling Method.

Functions for Distributions

Functions are defined as distributing money and funding among individuals and proposals using the log-normal and random exponential numbers generators.

Proposal Dynamics

The ProposalDynamics() function simulates the process of individuals rating proposals on the Deep Funding platform, socially influenced by previous ratings from other users. This influences how they will grade and vote on the proposals.

**Voting Methods**

Different methods of casting votes are implemented:

- Square root tokens voting (CastVotesSqrtTokens())

- One coin, one vote (CastVotesOneCoinOneVote())
- One wallet, one vote (CastVotesOneWalletOneVote())
- Plurality voting with square root weight (CastVotesPluralityVoting())
- Plurality voting with linear weight (CastVotesLinearPluralityVoting())

**Decision Making**

Functions to determine which proposals to vote on and which are the most favored in plurality voting when there is a cost to grade the proposals:

```
void GetFavoriteProposals();
void DecideWichProjetcsToVote();
```

**Results Calculation**

Functions to get the most voted and awarded proposals:

```
void GetMostVotedProposals();
void GetAwardedProposals();
void GetPluralityAwardedProposals();
```

**Stage and Points**

Structures and functions to manage simulation stage threads:

```
struct STAGE {
  float INITIAL;
  float FINAL;
};
struct STAGE POINTS(float initial_point, float final_point, float increment, int number_stages);
```

## Next steps

Here we focused on modeling honest and responsible voters that try to give a proper grade to the proposals. We aim to consider dishonest voting behavior, for example, considering that a fraction (say 20%) of them are biased/extreme voters who vote 1 or 10 on the projects.

We want to focus in extreme cases to assess the simulation and the welfare metric properly defined the benefits of the outcome and how that affects community voting. With that, we may explore how the simulation parameters affect the outcome for each voting system and the impact of extreme/biased voting on these scenarios. We may also investigate efficiency of reputation metrics as the voting entropy in enhancing the system performance against dishonest voting behavior.

## Red Team Reviews

Deborah Duang

Review of Milestone 2.

This review is written with the assumption that the most current purpose of the project is was revealed by the contractor in an email to red team:

For this first project, we are focusing on the difference between:

a) square-root weighting of wealth with free 1-10 ranking of proposals

b) square-root weighting of wealth with preference allocation with square costs

c) (optional) linear weighting of wealth with preference allocation with square costs

We assume that a) represents the current SNET deep funding rules and b) the proposed change. This Milestone 2 review is of the setup of the problem, based on the literature references in the Milestone 2 report.

First, we look at the question of the appropriateness of the proposed alternative voting schemes to test given the literature. The square root weighting of wealth that appears in both options was proposed in the reference "Quadratic Voting as Efficient Corporate Governance" , written by one of the original inventors of Quadratic Voting, as follows :"we propose as a slightly more modest alternative a variant of QV that we will call Square-Root Voting (SRV). SRV provides simply that only shareholders vote, and that shareholders have the right to vote the square root of the number of shares they own. A shareholder who owns 1 share gets 1 vote; a shareholder who owns 4 shares gets 2 votes; a shareholder who owns 10 shares gets 3.2 votes; and so on. SRV is formally almost identical to QV." I believe this passage means that the singularity net option A is considered to be "almost identical to Quadratic Voting, and thus we need not assume that the Snet implementation is incorrect from the point of view of the inventors of Quadratic Voting. There already exists an equivalency between buying AGIX/ASI and buying votes.

The original QV proposal however focused on a the issue of the tyranny of the majority, and spoke about giving all shareholders an equal number of chances to vote, called vote credits, but charging multiple votes for the same proposal exponentially, so that the the chance of influencing the outcome of the vote rises linearly with the cost. Thus it makes sense to, in the case where there are many proposals, trade off influence for things one cares less about for influence over what one cares more about. This results in better utility over all, given that one may have a passion say, to right an injustice, and the proposal issue may mean more personally to those suffering that injustice. The SRV proposal by the inventor of QV might capture that difference in that a passionate stakeholder could buy enough AGIX to make their extreme votes have more effect than the extreme votes of those with fewer AGIX. That is, it doesn't matter that there is an option to vote less than 10 or negative 10 to the utility if the rational voter will always vote with weight ten. Because the quadratic rating is taken twice in the proposed alternative solution b), it may even be less the “original” quadratic voting than snet’s option A is.

This brings up the rule that in order to be informative the simulation must concentrate on the foreseen failures of the techniques. For example, it must compare rational voters to be fair, rather than pulling from 1-10 ratings randomly in voting. It must foresee possible failures when more realistic scenarios, such as uncertainty in the utility of the vote in the reference “Balancing Power in Decentralized Governance: Quadratic Voting under Imperfect Information”. In this work, it is theorized that QV works against specialization in that it incentivizes one to maximize the power of vote more broadly when the reason one might be voting narrowly is that that is the area of expertise, and with expertise properly applied the utility of all voters could grow. It would be useful to test option c above, linear voting, against a and b, in order to test for effects on expertise.

### Developer Comments

We thank Deborah Duang for the positive evaluation of our work and suggestions. We are honored to have you as a member of the Red Team reviewing our project's milestone progress and deliverables. Your expertise will improve the applicability of our work within the Singularity NET (SNET) Deep Funding initiative.

We confirm that the method used by SNET is valid and yields adequate outcomes and agree with your emphasis on focusing on potential extreme results in our simulation. We aim to obtain critical information regarding voting dynamics while laying the foundations of exploratory analysis and enabling the development of alternative methods and potential enhancements in the current system.

The simulation in Milestone 3 accounts for rational voting behavior, social validation, and nonconformity in agents' interactions through the Deep Funding platform. Nonconformity is modeled by an inertia parameter, representing uncertainty or disbelief against the average community review of the proposals, which is linked to the perceived benefit of the proposal to the community.

To assess the simulation work, we may contrast rational, responsible, and honest voting behavior with dishonest voting, where an individual gains an advantage by not reporting their true sentiment about a proposal and instead uses extreme 1 or 10 ratings on the projects.

One simple test could be to assume that a) there is no cost to voting and b) all majority projects will be funded. Then, an individual's best strategy is to give a maximum grade to all the projects from which they receive any benefit, however large or small, and to give a minimum score to all the projects with a negative benefit. Although assumption b) is unrealistic, it simplifies the scenario for analysis. Without b), there's some nuance regarding how much to weigh a proposal. Although unrealistic, it might be a helpful test to ensure we've properly defined the benefits of the outcome and how that affects community voting.

One of the main goals of this simulation project is to answer the following question: Is there a way to design the voting system such that people are incentivized to report the truth about how much value the project benefits or costs them?

In Milestone 3, we focus the simulation on modeling honest voters. Although they experience social validation and nonconformity, they all try to give a proper grade to the proposals. We may now consider that a fraction (say 20%) of them are biased/extreme voters who vote 1 or 10 on the projects. We can assume that they have a set of genuine and fake grades.

For example, let's suppose there are four proposals, and agent 9 is biased for project 1. The true grades of agent 9 are {6, 9, 5, 3}, where the genuine grade for proposal 1 is 6, the true grade for proposal 2 is 9, and so on. But since agent 9 is biased for project 1, he votes with a set of fake grades given by {10, 1, 1, 1}. While Agent 9 votes on proposals with his fake grades, we compute his welfare with his true grades. This will also affect other agents' welfare, primarily when funding poor-quality proposals.

We hypothesize that since there is no cost to the grades in the SingularityNET square root weighting method, this method could be highly susceptible to extreme voting (especially if we have whales participating). In contrast, in plurality voting (alternative solution b), the effects of collusion would be limited since extreme grades such as 1 or 10 would cost more. Hence, it is possible that with a more realistic set of voters, the aggregate voter community welfare of plurality voting can outperform the current SingularityNET method.

Instead of considering the quadratic rating taken twice in the alternative solution b), we propose allocating credits with quadratic costs, where extreme grades cost more. For example, we consider the set of grades {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} with cost weights {5, 4, 3, 2, 1, 1, 2, 3, 4, 5}. For instance, if an agent has 100 AGIX and grades {9, 7, 8} on proposals 1, 2, and 3, he would allocate 4/(4+2+3) = 0.44 of his credits to voting on proposal 1. Thus, the effective number of votes he casts on proposal 1 would be sqrt(4*100/(4+2+3)) = 6.67.

Considering these approaches will allow us to compare different voting scenarios accurately, foresee possible failures, and assess the simulation precision in modeling the aggregate community welfare and incentivizing people to report the truth about the project's benefits to them.

Indeed, in quadratic voting, voters may find it more cost-effective to distribute their votes across several issues rather than concentrating them on one to maximize their overall influence. This can lead to voters participating in decisions outside their primary areas of expertise, potentially diluting the quality of their input on specialized topics. We will be sure to include the linear voting option (c) or One Coin, One Vote (1C1V) in our simulations to help us understand its effects on specialization and expertise compared to options (a) and (b).

Walter Karshat
Review of Milestone 2.
Designation

Granting Agency: Singularity NET Deep Funding
Funding Round: 4-Beta
Pool: RFP
Proposal: **Simulating The Quality Of Community Decisions**
Proposer: Photrek
Code: DFR4B-RFP2

### Introduction

It is a distinct honor for me to have been selected as a second member of the Red Team to review for the benefit of the distinguished Proposer the milestone progress and deliverables [Add A] of an important project within the Singularity NET Deep Funding initiative, Simulating The Quality Of Community Decisions. This will afford me an opportunity to assist in matching the theoretical aspects of state-of-the-art research and simulations to the applications of mechanism design and specifically of quadratic voting arithmetic to the process and the outcomes of participants submitting their preferences toward joint decision-making in community or public benefit open organizations.

My personal experiences with Red Team efforts and voting mechanisms are predominantly informed by hands-on efforts within real-world deployments [Add B], and my sociopolitical attitudes have been shaped by individual life history. However, for this engagement, I will focus predominantly on reviewing specific project work items, as interpreted and delivered by the Proposer. The Milestone 2 Voting Prototyping report, already accepted by the Deep Funding staff, is the subject of this round of my reviews.

### Developer Comments

We thank Walter Karshat for the positive evaluation of our work and suggestions. We are honored to have you as a member of the Red Team reviewing our project's milestone progress and deliverables. Your expertise will improve the applicability of our work within the Singularity NET (SNET) Deep Funding initiative.

### Literature Search

Without a doubt, key participants in this initiative are deeply cognizant of the state of the art in the fields of mechanism design, in voting generally, and the variants of quadratic arithmetic specifically. Similarly, they understand stochastic techniques for modeling complex interactions approximating social phenomena to perfection. The lists of References in the Milestone 2 Report and in the two referenced articles authored by André Vilela enumerate dozens of sources of top quality.

### Developer Comments

We thank the referee for his commendations. We are proud to have the opportunity to contribute to this high-quality challenge.

### Voter Model

I'm certain that a worthwhile codebase will be developed or adopted to implement a stochastic computation repeatedly executing the model as described, with several function value lookups inside a multiply / accumulate loop iterated repeatedly. Not yet familiar enough with other implementations, I cannot judge which portions of the proposed model or the capabilities it affords are distinctive or innovative, but hoping to close the gap later in the review process.

To me, the main challenge of this effort will reside in mapping the insights from the model-based investigation to the Deep Funding decision space and its somewhat peculiar arithmetic. For example, the vote-buying metaphor commonly used in theoretical excursions to substantiate the search for equilibria is particularly a stretch with this security. It sports over a $ Billion valuation, resulting in an overwhelming capital at risk requirement to entirely control the outcomes of distributing $1.5M, and in chunks of at most $250K.

### Developer Comments

We thank you for raising these aspects. Motivated by original investigations on quadratic voting, we are developing a state-of-the-art stochastic simulation. We have shared the GitHub link containing the C commented code for Milestone 3, which simulates individual sentiments about proposals and indicates voting behavior across scenarios, assessing impacts on community aggregate welfare.

We based our approach on actual deep funding data, which involved modeling the distribution of user credits and funding requests. For users' inertia distribution, we created a reasonable approach based on voting analysis articles by Jan Horlings, the Chief Deep Funding Officer at SingularityNET, who has analyzed voting behavior from past funding rounds.

We agree with your concerns regarding the complexity of such implementation and hope our simulations offer critical insights on efficiently implementing the system while matching the high standards of the SNET community.

### Communicating Agents

After seeing the term in use, I was looking for modeling of inter-agent communication or other coordination, perhaps to represent small world networks arising from prior social or economic groupings within Singularity NET or Cardano ecosystems. Also, the teams of participants in present and prior proposals, and their friends and family are very likely to group and align their choices.

Note:

*Quadratic literature assumes a negative posture toward voter coordination, even if only detecting some correlation in their choices, and rushes to label it collusion. Meanwhile, communication and alignment are key features of any mutually beneficial human interaction, required for community cohesion and longevity. A method of investigating these interactions and exploring the group configurations and alignment rulesets that result in positive, acceptable, or pathological outcomes would be quite meaningful, even if it falls outside the scope of this specific project.*

### Developer Comments

Thank you for your insightful comments. In Milestone 2, we designed an agent-based interacting model that did not rely on static topological structures. Our interaction network considers an initial user who gives his opinion (rating a proposal), and the following users consider this opinion when deciding theirs. A tunable probability influences how users vote or skip voting based on the proposal's rank, while another probability parameter regulates the chance of a user to post or not their opinion, influencing other agents. The opinion dynamics is based on "growth mechanics" (new users influenced by old users).

We were inspired by the deep funding interaction platform where users can comment, react to comments, and review and rate proposals. The social dynamics in the "Discussion" and "Review and Rating" sections affect the decision-making of the individuals who are evaluating the proposals. Your suggestion to model inter-agent coordination using small-world networks within the SingularityNET and Cardano ecosystems is compelling, and we agree on considering such a possibility.

We agree that communication and healthy alignment levels are natural and essential aspects of any voting community. As part of Milestone 4, we shall develop a proposal plan for how correlation measurements can be incorporated into the simulation, considering your observations on collusion. Thus, we could explore how these groups' coordinations can result in acceptable or adverse outcomes for the community.

### Credit Distribution Function

I wonder whether the optimality of the SQRT voting power adjustment relies on certain assumptions regarding the function P, the distribution of credits. Perhaps an upcoming expanded set of model documentation will cover this relationship, and we can probe the token amount distribution across some subsets of AGIX wallets.

### Developer Comments

That is an excellent point. In Milestone 3, we will illustrate an analysis of credit distribution based on the data extracted from the Deep Funding. We are exploring welfare functions and voting credit distribution to investigate possible impacts on the outcomes.

### Aggregate Welfare

Optimizing for W, defined as a simple sum of the welfare number for each participant, likely comes from considering popular political voting, where the entire population is supposed to benefit, and the outlook of the voters is representative of the total. With Deep Funding proposals, the total welfare of the entirety of the token holders and affiliated community members could be argued to represent the ecosystem’s best interest.

However, the totality of all accounts casting DF proposal votes and all the program applicants and proposal team members are a very small fraction of the entire community, and unlikely its representative selection.

### Developer Comments

We agree and acknowledge that only a tiny fraction of the Deep Funding community participates in voting. In this project, we developed a welfare function that relates to voters expressing their preferences on a poll. We highlight that it does not capture the complexity and amplitude of the implications of a voting round but offers a measure of voter satisfaction with the outcome of a voting effort. Now aware of your highlight, we propose defining W as the aggregate voters' welfare.

Concerned that a simple sum of individual welfare might not accurately represent the aggregate voter welfare, we are redefining the calculation to use a geometric mean of individual welfare for each proposal. This approach makes the voters' welfare measure sharper to the lowest individual welfare scores, providing a more authentic representation of the ecosystem's overall well-being.

### TL;DR

### Web3 Population Segments

Web3 token and digital security holdings and governance protocols have certain peculiarities, historical and systemic.

There are several fairly distinct segments in a typical post-ICO token holder population, and their attitudes and voting patterns likely differ. The common justification for decreasing the voting power of the dreaded whales is by comparing them to the long-term core community contributors, who worked hard to earn their few tokens, but supposedly have the best interest of the project and the ecosystem in their hearts and are eager to vote accordingly.

Meanwhile, the project founders and the institutional pre-ICO investors are particularly well-informed and invariably support the project's long-term prospects for growth, while controlling very large token positions. The droves of nickel-and-dime retail investors all too often know little and care less about a particular project, only looking for a quick valuation pop, and may even arrive as part of some social media orchestrated pump-and-dump clique. They are relegated to the despised speculator category.

In reducing the voting power of large token holders with the aim of protecting core *contributors* from the *whales*, one can decrease the voice of the *founders* and favor the *speculators*.

### Developer Comments

We recognize that reducing the voting power of whales to protect core contributors can inadvertently reduce the influence of knowledgeable founders and favor less informed speculators. We believe that addressing the governance challenges in Web3 communities involves finding a balance between the influence of different types of token holders.

In the current voting system of singularityNET, there is no cost associated with the grades of the proposals. We are working on a reputation-based voting mechanism alongside quadratic voting to support our investigation of balancing interests and voting power. We aim to enhance the voting power of long-term, continuously engaged core community contributors who have earned their tokens through consistent efforts and have the ecosystem's best interest at heart. This system also incentivizes users to be more engaged, naturally supporting them in being more informed about the proposals and the community, positively contributing to their voting behaviors.

### Voter Turnout

In Crypto or DAO token-based voting the turnout can be just a small fraction of % of all active wallets [accounts]. At times only a few percent of all issued tokens can determine the outcome. The total of tokens may be overstated due to lockups, withholding votes for ethical or liability reasons, custodial accounts, and even lost keys. More importantly, the result may be deceptive, given the prevalence of approvals for proposals with major ecosystem impact, and often by a wide margin.

One can assume strong pre-proposal synchronization between the proposers and a dominant set of voters, often strongly overlapping. Likely, they can tap more large voters to ensure the desired outcome. With the AGIX token merger vote held only two months ago, we can study the turnout and distribution of selections in the decision of uniquely profound importance to all token holders, and compare with what happened during the DF voting last fall.

### Developer Comments

Indeed, the number of accounts casting proposal votes is a tiny fraction of the entire Deep funding community. Thus, only a small percentage of all active wallets determine the outcome.

In the initial model on Milestone 2, we simulated the voter's decisions between 2 proposals, where all voters had the same probability of voting in one proposal or the other.

However, in practice, people experience social validation from other users' comments, reviews and ratings. Since people do not truly vote independently, it is natural that some projects have a more substantial prevalence due to social influence.

In particular, proposals with a positive expert review may yield an even more powerful herd effect, which explains that these proposals generally have an outstanding performance in the voting process.

As you assumed, another possibility is pre-proposal synchronization between the proposers and the voters. This can happen, for example, via propaganda of proposals on the Deep Funding Telegram group, which can influence voters' perspectives. In Milestone 3, we included the users' social validation and nonconformity effects via the Deep Funding platform.

Your suggestion about analyzing how many people voted and how the votes were distributed among the different choices before and after the AGIX merger is also interesting and could reveal patterns in voter turnout, behavior, and outcomes.

### Addendum A: Proposed Milestone 2 Deliverables

- A comprehensive literature review to guide the research design focusing on voting dynamics and individual preferences.
- An Initial prototype of a Plural Voting model designed to uncover voting patterns and facilitate expressions in interactive groups.
- A Monte Carlo computational model using random number sequences to model social voting dynamics.

### Addendum B: Personal Position Statement

*I regard it as completely unimportant who in the party will vote and how, but it is extremely important who will count the votes and how.*

Stalin

My strongly held beliefs have been informed by personal life experiences. Conceptually, I find suspect the notion that some individuals know quite well what the desired outcomes of the key decisions are, which then entitles them to manipulate the process to achieve those.

An intellectually honest approach would be to forgo the pretense of any general population involvement and simply pronounce the steps to enforce such glorious outcomes. The population is likely to disagree, perhaps even violently, rightly questioning the knowledge, the ability to implement, and the motives of such prognosticators.

## Budget & Schedule

| Milestone | Description | Budget | Status |
|---|---|---|---|
| 1 | Contract Signing & Management Reserve | $2,400 | Submitted - Approved |
| 2 | Prototyping of Monte Carlo Simulations | $10,970 | Submitted - 20 May; Approved |
| 3 | Simulation of Voting Methods | $11,080 | Submitted - June |
| 4 | Analysis of the Voting Dynamics | $11,660 | Planned - July |
| 5 | Final Report & Roadmap | $13,890 | Planned - Aug |

## Future Plans & Change Notifications

## Project References


[1] Moe, Wendy W., and Michael Trusov. "The value of social dynamics in online product ratings forums." *Journal of marketing research* 48.3 (2011): 444-456.

[2] Mubarok, Mohamad Syahrul, Adiwijaya Adiwijaya, and Muhammad Dwi Aldhi. "Aspect-based sentiment analysis to review products using Naïve Bayes." *AIP conference proceedings*. Vol. 1867. No. 1. AIP Publishing, 2017.

[3] Sahoo, Nachiketa, Chrysanthos Dellarocas, and Shuba Srinivasan. "The impact of online product reviews on product returns." *Information Systems Research* 29.3 (2018): 723-738.

[4] Constantinides, Efthymios, and Nina Isabel Holleschovsky. "Impact of online product reviews on purchasing decisions." *12th International Conference on Web Information Systems and Technologies, WEBIST 2016*. SCITEPRESS, 2016.

[5] Baccianella, Stefano, Andrea Esuli, and Fabrizio Sebastiani. "Multi-facet rating of product reviews." *Advances in Information Retrieval: 31th European Conference on IR Research, ECIR 2009, Toulouse, France, April 6-9, 2009. Proceedings 31*. Springer Berlin Heidelberg, 2009.

[6] Network Science. Albert-Lazló Barabási. Cambridge University Press, 2016.

[7] Reyhani, Reyhaneh, Mark C. Wilson, and Javad Khazaei. "Coordination via polling in plurality voting games under inertia." *Proceedings of COMSOC* (2012): 359-370.

[8] "Deep Funding Round 1 Voting Analysis - Part 1: Number Crunching." Deep Funding, deepfunding.ai/deep-funding-round-1-voting-analysis-part-1-number-crunching/. Accessed 17 June 2024.

[9]Benhaim, Alon, Brett Hemenway Falk, and Gerry Tsoukalas. "Balancing Power in Decentralized Governance: Quadratic Voting under Imperfect Information." SSRN Scholarly Paper. Rochester, NY, April 3, 2023. https://doi.org/10.2139/ssrn.4416748.

[10] Kelter, Jacob, Andreas Bugler, and Uri Wilensky. "Agent-Based Models of Quadratic Voting." In *Proceedings of the 2020 Conference of The Computational Social Science Society of the Americas*, edited by Zining Yang and Elizabeth Von Briesen, 131–42. Springer Proceedings in Complexity. Cham: Springer International Publishing, 2021. https://doi.org/10.1007/978-3-030-83418-0_8.

[11] Lalley, Steven P., and E. Glen Weyl. "Quadratic Voting." Available at SSRN, 2016. https://www.aeaweb.org/conference/2015/retrieve.php?pdfid=3009&tk=BHDG8H2E.

[12] Lalley, Steven P., and E. Glen Weyl. "Quadratic Voting: How Mechanism Design Can Radicalize Democracy." AEA Papers and Proceedings 108 (May 2018): 33–37. https://doi.org/10.1257/pandp.20181002.