

Analysis and Attacks of decentralized content curation platforms

Orfeas Stefanos Thyfronitis Litos, Andrés Monteoliva Mosteiro, and
Aggelos Kiayias

University of Edinburgh
o.thyfronitis@ed.ac.uk, a.monteoliva@serious.server, akiayias@inf.ed.ac.uk

Abstract. We will attack Steem.

1 Introduction

Steem is not incentive-compatible.

2 Related Work

Many people have done many similar things.

3 Model

1 Notation

- We denote the set of all probability distributions on set A as $\mathcal{D}(A)$.
- We denote the powerset of a set A with 2^A .
- $a||b$ denotes the concatenation of a and b .
- Let $n \in \mathbb{N}^*$. $[n]$ denotes $\{1, 2, \dots, n\}$.

2 Properties of Post Voting Systems

A post voting system has the objective to arrange the posts according to the preferences of the participants. The ideal order is defined based on the likeability matrix for the posts.

Definition 1 (Post). Let $N \in \mathbb{N}^*$. A post is defined as $p = (i, l)$, with $i \in [N], l \in [0, 1]^N$.

- **Author.** The first element of a post is the index of its creator, i .
- **Likeability.** The likeability of a post is defined as $l \in [0, 1]^N$.

Let $M \in \mathbb{N}^*$ the number of posts. Then $\forall j \in [M]$, let $\text{creator}_j \in [N]$, $l_j \in [0, 1]^N$ and $p_j = (\text{creator}_j, l_j)$. The set of all posts is $\mathcal{P} = \bigcup_{j=1}^M \{p_j\}$.

Definition 2 (Ideal Score of a post). Let post $p = (m, l)$. We define the ideal score of p as $\text{idealSc}(p) = \sum_{i=1}^N l_i$.

The ideal score of a post is a single number that represents its overall worth to the community. By using simple summation, we assume that the opinions of all players have the same weight. In an ordered list of posts where higher posts are more visible, the “common interest” would require that a post with higher ideal score appear before another post with a lower score.

Definition 3 (t -Ideal Post Order). Let \mathcal{P} a list of posts. We say that \mathcal{P} is in t -ideal order and that the property $\text{IDEAL}^t(\mathcal{P})$ holds if

$$\forall i < j \in |t|, \text{idealSc}(\mathcal{P}[i]) \geq \text{idealSc}(\mathcal{P}[j]) \quad .$$

Definition 4 (Post-Voting System). A tuple $\mathcal{S} = (\text{INIT}, \text{AUX}, \text{HANDLEVOTE}, \text{VOTE})$ of four algorithms. The four algorithms parametrize the following two ITMs:

$\mathcal{G}_{\text{Feed}}$ is a global functionality that accepts two messages: **read**, which responds with the current list of posts and **vote**, which can take various arguments and does whatever is defined in HANDLEVOTE .

Π_{honest} is a protocol that sends **read** and **vote** messages to $\mathcal{G}_{\text{Feed}}$ whenever it receives (**activate**) from \mathcal{E} .

Algorithm 1 $\mathcal{G}_{\text{Feed}}(\text{INIT}, \text{AUX}, \text{HANDLEVOTE})(\mathcal{P}, \text{initArgs})$

- 1: Initialization:
 - 2: $\mathcal{U} \leftarrow \emptyset$
 - 3: $\text{INIT}(\text{initArgs})$
 - 4:
 - 5: Upon receiving (**read**) from u_{pid} :
 - 6: $\text{aux} \leftarrow \text{AUX}(u_{\text{pid}})$
 - 7: Send (**posts**, \mathcal{P} , aux) to u_{pid}
 - 8:
 - 9: Upon receiving (**vote**, ballot) from u_{pid} :
 - 10: $\text{HANDLEVOTE}(\text{ballot})$
-

Algorithm 2 $\Pi_{\text{honest}}(\text{VOTE})$

- 1: Upon receiving (**activate**) from \mathcal{E} :
 - 2: Send (**read**) to $\mathcal{G}_{\text{Feed}}$
 - 3: Wait for response (**posts**, \mathcal{P} , aux)
 - 4: ballot $\leftarrow \text{VOTE}(\mathcal{P}, \text{aux})$
 - 5: Send (**vote**, ballot) to $\mathcal{G}_{\text{Feed}}$
-

Definition 5 (Post-Voting System Activation Message). We define act_{pid} as the message (**activate**), sent to u_{pid} .

Definition 6 (Execution Pattern). Let $N, R \in \mathbb{N}^*$.

$$\text{ExecPat}_{N,R} = \left\{ (\text{act}_{\text{pid}_1}, \dots, \text{act}_{\text{pid}_{NR}}) : \forall i \in [R], \forall k \in [N], \exists j \in [N] : \text{pid}_{(i-1)N+j} = k \right\} ,$$

*i.e. activation messages are grouped in R rounds and within each round each player is **activated** exactly once. The order of activations is not fixed.*

Let Environment \mathcal{E} that sends messages $\text{msgs} = (\text{act}_{\text{pid}_1}, \dots, \text{act}_{\text{pid}_n})$ sequentially. We say that \mathcal{E} respects $\text{ExecPat}_{N,R}$ if $\text{msgs} \in \text{ExecPat}_{N,R}$. (Note: this implies that $n = NR$.)

Definition 7 ((N, R, M, t) -convergence under honesty). We say that a post-voting system $\mathcal{S} = (\text{INIT}, \text{AUX}, \text{HANDLEVOTE}, \text{VOTE})$ (N, R, M, t) -converges under honesty (or t -converges under honesty for N players, R rounds and M posts) if, for every input \mathcal{P} such that $|\mathcal{P}| = M$, for every \mathcal{E} that respects $\text{ExecPat}_{N,R}$ and given that all protocols execute Π_{honest} , it holds that after \mathcal{E} completes its execution pattern, $\mathcal{G}_{\text{Feed}}$ contains a post list \mathcal{P} such that $\text{IDEAL}^t(\mathcal{P})$ is true.

TODO: Discuss: Is R missing from Steem system? Maybe add "Let $N, R \in \mathbb{N}^*$, ..., controlled by an Environment that respects $\text{ExecPat}_{N,R}$ and the following ...".

Definition 8 (Steem system). The Steem system is the post voting system \mathcal{S} with parameters $\mathbf{SP} \in \mathbb{N}^{*N}$, $a, b, \text{regen} \in [0, 1]$, $\text{attSpan} \in \mathbb{N}^*$ and the following parametrizing procedures:

Algorithm 3 INIT ($\mathbf{SP}, \text{attSpan}, a, b, \text{regen}$)

```
1: Store input parameters as constants
2:  $r \leftarrow 1$ 
3:  $\text{lastVoted} \leftarrow \underbrace{(0, \dots, 0)}_N$ 
4:  $\mathbf{VP} \leftarrow \underbrace{(1, \dots, 1)}_N$ 
5:  $\text{scores} \leftarrow \underbrace{(0, \dots, 0)}_{|\mathcal{P}|}$ 
```

Algorithm 4 AUX

```
1: return ( $\mathbf{SP}, \text{attSpan}, a, b, \text{regen}$ )
```

Algorithm 5 HANDLEVOTE ($\text{ballot}, u_{\text{pid}}$)

```
1: if  $\text{lastVoted}_{\text{pid}} \neq r$  then ▷ One vote per player per round
2:    $\mathbf{VP}_{\text{pid}, r} \leftarrow \mathbf{VP}_{\text{pid}}$  ▷ For proofs
3:    $\mathbf{VP}_{\text{pid}} \leftarrow \max \{ \mathbf{VP}_{\text{pid}} + \text{regen} \cdot (r - \text{lastVoted}_{\text{pid}}), 1 \}$  ▷ TODO: Remove
    $(r - \text{lastVoted}_{\text{pid}})?$ 
4:   if  $\text{ballot} \neq \text{null}$  then
5:     Parse ballot as  $(p, \text{weight})$ 
6:      $\text{cost} \leftarrow a \cdot \mathbf{VP}_{\text{pid}} \cdot \text{weight} + b$ 
7:     if  $\mathbf{VP}_{\text{pid}} - \text{cost} \geq 0$  then
8:        $\text{score} \leftarrow \text{cost} \cdot \mathbf{SP}_{\text{pid}}$ 
9:        $\mathbf{VP}_{\text{pid}} \leftarrow \mathbf{VP}_{\text{pid}} - \text{cost}$ 
10:    else
11:       $\text{score} \leftarrow \mathbf{VP}_{\text{pid}} \cdot \mathbf{SP}_{\text{pid}}$ 
12:       $\mathbf{VP}_{\text{pid}} \leftarrow 0$ 
13:    end if
14:     $\text{scores}_p \leftarrow \text{scores}_p + \text{score}$ 
15:  end if
16:   $\text{lastVoted}_{\text{pid}} \leftarrow r$ 
17: end if
18: if  $\forall i \in [N], \text{lastVoted}_i = r$  then ▷ round over
19:    $\mathcal{P} \leftarrow \text{ORDER}(\mathcal{P}, \text{scores})$  ▷ order posts by votes
20:    $r \leftarrow r + 1$ 
21: end if ▷ TODO: count rounds? simplify with set of voted and check of length?
```

Algorithm 6 VOTE(\mathcal{P} , aux)

```
1: Store aux contents as constants
2: if VOTETHISROUND( $r$ ) = yes then
3:   top  $\leftarrow$  CHOOSETOPPOSTS(attSpan,  $\mathcal{P}$ , votedPosts)
4:    $(i, l) \leftarrow \underset{(i,l) \in \text{top}}{\text{argmax}} \{l_{\text{pid}}\}$ 
5:   votedPosts  $\leftarrow$  votedPosts  $\cup (i, l)$ 
6:   return  $((i, l), l_{\text{pid}})$ 
7: else
8:   return null
9: end if
10:
11: function CHOOSETOPPOSTS(attSpan,  $\mathcal{P}$ , votedPosts)
12:   res  $\leftarrow \emptyset$ 
13:   idx  $\leftarrow 1$ 
14:   while |res| < attSpan & idx  $\leq |\mathcal{P}|$  do
15:     if  $\mathcal{P}[\text{idx}] \notin \text{votedPosts}$  then ▷ One vote per post per player
16:       res  $\leftarrow$  res  $\cup \{\mathcal{P}[\text{idx}]\}$ 
17:     end if
18:     idx  $\leftarrow$  idx + 1
19:   end while
20:   return res
21: end function
22:
23: function VOTETHISROUND( $r, |\mathcal{P}|$ )
24:   if  $R < |\mathcal{P}|$  then
25:     return yes
26:   else if  $\left\lfloor (r-1) \bmod \frac{R-1}{|\mathcal{P}|-1} \right\rfloor = 0$  then ▷ TODO: if Discussion above is
27:     return yes accepted, change  $|\mathcal{P}|$  to  $M$  and don't input  $|\mathcal{P}|$ .
28:   else
29:     return no
30:   end if
31: end function
```

Theorem 1. *The Steem system (N, R, M, t) -converges if and only if SP is constant and $R - 1 \geq (M - 1) \left\lceil \frac{a+b}{\text{regen}} \right\rceil$.*

Discussion

- If players have attention span smaller than the full list and do not have the rounds to vote for every post, make a \mathcal{P} with the best post at the end and it will stay there.

Proof. – (\Leftarrow) Suppose that

$$R - 1 \geq (M - 1) \left\lceil \frac{a + b}{\text{regen}} \right\rceil \quad (1)$$

and $\forall i \in [N], \mathbf{SP}_i = c$.

Let $\text{pid} \in [N]$. In this case it is $R \geq M$ and according to **VOTETHIS-ROUND** in Algorithm 6, u_{pid} votes non-null in rounds (r_1, \dots, r_M) with $r_i = \left\lfloor (i - 1) \frac{R-1}{M-1} \right\rfloor + 1$. Observe that:

$$(1) \Rightarrow \frac{R-1}{M-1} \geq \left\lceil \frac{a+b}{\text{regen}} \right\rceil \xrightarrow[\text{integer}]{\text{rhs}} \left\lfloor \frac{R-1}{M-1} \right\rfloor \geq \left\lceil \frac{a+b}{\text{regen}} \right\rceil , \quad (2)$$

$$\forall i \in [M] \setminus \{1\}, r_i \in \left\{ r_{i-1} + \left\lfloor \frac{R-1}{M-1} \right\rfloor, r_{i-1} + \left\lceil \frac{R-1}{M-1} \right\rceil \right\} . \quad (3)$$

From (2) and (3) we have that $\forall i \in [M-1], r_{i+1} - r_i \geq \left\lceil \frac{a+b}{\text{regen}} \right\rceil$. Thus, according to **HANDLEVOTE** (Algorithm 5, line 3) and keeping in mind that in each round every player sends exactly one vote (real or null) according to $\text{ExecPat}_{N,R}$, we have that

$$\begin{aligned} \mathbf{VP}_{\text{pid}, r_{i+1}} &\geq \mathbf{VP}_{\text{pid}, r_i} + \text{regen} \left(\left\lceil \frac{a+b}{\text{regen}} \right\rceil - 1 \right) \Rightarrow \\ \mathbf{VP}_{\text{pid}, r_{i+1}} &\geq \mathbf{VP}_{\text{pid}, r_i} + a + b - \text{regen} . \end{aligned} \quad (4)$$

But $\mathbf{VP}_{\text{pid}, 1} = 1$ (Algorithm 3, line 4) and $\forall i \in [M] \setminus \{1\}, \mathbf{VP}_{\text{pid}, r_{i+1}} \geq \mathbf{VP}_{\text{pid}, r_i} (1 - a) - b$ (Algorithm 5, lines 6-13), thus $\forall i \in [\text{voteRounds}], \mathbf{VP}_{\text{pid}, r_i} = 1$. This in turn means that at the end of the execution, $\forall p = (i, l) \in$

\mathcal{P} , $\text{scores}_p = c \left(Nb + a \sum_{\text{pid} \in [N]} l_{\text{pid}} \right)$, thus all posts will be ordered according to their ideal scores.

– (**SP** variable \Rightarrow no convergence) Let $\mathcal{P} = ((1, (a_1, \dots, a_N)), (2, (b_1, \dots, b_N)))$ such that the following linear constraints are simultaneously feasible:

$$\begin{aligned} \sum_{i=1}^N a_i &> \sum_{i=1}^N b_i \\ \sum_{i=1}^N \text{SP}_i a_i &< \sum_{i=1}^N \text{SP}_i b_i \end{aligned}$$

I think that's always possible if **SP** is not constant.

– (inequality doesn't hold \Rightarrow no convergence) Consider $R-1 < (M-1) \left\lceil \frac{a+b}{\text{regen}} \right\rceil, \forall i \in [N], \mathbf{SP}_i = c$. Then bullet (1) from the previous holds, but (2) becomes $\left\lfloor \frac{R-1}{M-1} \right\rfloor \geq \left\lceil \frac{a+b}{\text{regen}} \right\rceil$. Thus $\mathbf{VP}_{\text{pid}, r_{i+1}} < \mathbf{VP}_{\text{pid}, r_i} + a + b$ and $\forall i \in [\text{voteRounds}] \setminus \{1\}, \mathbf{VP}_{\text{pid}, r_i} < 1$. Let $\epsilon_i = 1 - \mathbf{VP}_{\text{pid}, r_i}$ and $\epsilon > 0$.

- If $R < M$, then the last $M - R$ posts of \mathcal{P} will not be voted by any player. This in turn means that a \mathcal{P} such that $\forall (i, l) \in \mathcal{P} [1..R], l = (0.5, \dots, 0.5)$ and $\forall (i, l) \in \mathcal{P} [R+1..M], l = (1, \dots, 1)$, the Steem system does not $(N, r, M, 1)$ -converge.
- If $p_{\text{bad}} = (i, l_{\text{bad}}) = \mathcal{P} [1]$ with $l_{\text{bad}} = (1 - \epsilon_M + \epsilon, \dots, 1 - \epsilon_M + \epsilon)$, $p_{\text{good}} = (i, l_{\text{good}}) = \mathcal{P} [M]$ with $l_{\text{good}} = (1, \dots, 1)$ and there exists no other post $p \in \mathcal{P}$ such that $\text{idealSc}(p) \geq \text{idealSc}(p_{\text{good}})$, then $\forall \mathcal{P}' : \mathcal{P}'$ contains the same posts as \mathcal{P} and $\text{IDEAL}^1(\mathcal{P}')$ holds, it is $\mathcal{P}' [1] = p_{\text{good}}$. However, after the execution the score of p_{bad} will be $\text{sc}_{\text{bad}} = Nc(a(1 - \epsilon_M + \epsilon) + b)$, whereas the score of p_{good} will be $\text{sc}_{\text{good}} = Nc(a(1 - \epsilon_M) + b) < \text{sc}_{\text{bad}}$, thus the Steem system does not $(N, r, M, 1)$ -converge.

We place the good posts at the end. Players will vote for them with little voting power and they will not rise to the top.

□

The above result is tight. If the conditions are violated the above theorem is not true.

4 Results

Steem won't achieve high quality posts.

5 Further Work

Posts at any time

6 Conclusion

Keep inventing new decentralized content curation platforms.

7 Acknowledgements

We thank @seriousposter for their invaluable posts analyzing Steem and our mums for the cookies.

References