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 $\mathcal{D}(\mathcal{A})$.
- 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 $\{1, 2, \ldots, n\}$.

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* $\mathrm{creator}_j \in [N], l_j \in [0,1]^N$ *and* $p_j = (\mathrm{creator}_j, l_j)$. *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* $\mathrm{idealSc}(p) = \sum_{i=1}^{N} l_i$.

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* $\textsc{Ideal}^t(\mathcal{P})$ *holds if*

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

Definition 4 (Post-Voting System). *A tuple* $\mathcal{S} = (\textsc{Init}, \textsc{Aux}, \textsc{HandleVote}, \textsc{Vote})$ *of four algorithms. The four algorithms parametrize the following two ITMs:*

$\mathcal{G}_{\mathrm{Feed}}$ *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* $\textsc{HandleVote}$.

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

Algorithm 1 $\mathcal{G}_{\mathrm{Feed}}(\textsc{Init}, \textsc{Aux}, \textsc{HandleVote})(\mathcal{P}, \mathrm{initArgs})$

```
Initialization:
    𝒰 ← ∅
    Init(initArgs)

Upon receiving (read) from u_pid:
    aux ← Aux(u_pid)
    Send (posts, 𝒫, aux) to u_pid

Upon receiving (vote, ballot) from u_pid:
    HandleVote(ballot)
```

Algorithm 2 Π_{honest} (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 $\leftarrow$ VOTE $(\mathcal{P}, \text{aux})$
5: Send (`vote`, ballot) to $\mathcal{G}_{\text{Feed}}$

Definition 5 (Post-Voting System Activation Message). *We defince* $\texttt{act}_{\text{pid}}$ *as the message* (`activate`)*, sent to* u_{pid}*.*

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

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

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* $\text{msgs} = (\texttt{act}_{\text{pid}_1}, \dots, \texttt{act}_{\text{pid}_n})$ *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* $\boldsymbol{SP} \in {\mathbb{N}^*}^N, a, b, \text{regen} \in [0, 1], \text{attSpan} \in \mathbb{N}^*$ *and the following parametrizing procedures:*

Algorithm 3 $\textsc{Init}\,(\mathbf{SP}, \mathrm{attSpan}, a, b, \mathrm{regen})$

Store input parameters as constants
$r \leftarrow 1$
$\mathrm{lastVoted} \leftarrow \underbrace{(0, \ldots, 0)}_{N}$
$\mathbf{VP} \leftarrow \underbrace{(1, \ldots, 1)}_{N}$
$\mathrm{scores} \leftarrow \underbrace{(0, \ldots, 0)}_{|\mathcal{P}|}$

Algorithm 4 $\textsc{Aux}$

return $(\mathbf{SP}, \mathrm{attSpan}, a, b, \mathrm{regen})$

Algorithm 5 $\textsc{HandleVote}\,(\mathrm{ballot}, u_{\mathrm{pid}})$

if $\mathrm{lastVoted}_{\mathrm{pid}} \neq r$ **then** ▷ One vote per player per round
  $\mathbf{VP}_{\mathrm{pid}} \leftarrow \max\{\mathbf{VP}_{\mathrm{pid}} + \mathrm{regen} \cdot (r - \mathrm{lastVoted}_{\mathrm{pid}}), 1\}$
  if ballot $\neq$ **null then**
    Parse ballot as (p, weight)
    $\mathrm{cost} \leftarrow a \cdot \mathbf{VP}_{\mathrm{pid}} \cdot \mathrm{weight} + b$
    if $\mathbf{VP}_{\mathrm{pid}} - \mathrm{cost} \geq 0$ **then**
      $\mathrm{score} \leftarrow \mathrm{cost} \cdot \mathbf{SP}_{\mathrm{pid}}$
      $\mathbf{VP}_{\mathrm{pid}} \leftarrow \mathbf{VP}_{\mathrm{pid}} - \mathrm{cost}$
    else
      $\mathrm{score} \leftarrow \mathbf{VP}_{\mathrm{pid}} \cdot \mathbf{SP}_{\mathrm{pid}}$
      $\mathbf{VP}_{\mathrm{pid}} \leftarrow 0$
    end if
    $\mathrm{scores}_p \leftarrow \mathrm{scores}_p + \mathrm{score}$
  end if
  $\mathrm{lastVoted}_{\mathrm{pid}} \leftarrow r$
end if
if $\forall i \in [N]\ \mathrm{lastVoted}_i = r$ **then** ▷ round over
  $\mathcal{P} \leftarrow \textsc{Order}\,(\mathcal{P}, \mathrm{scores})$ ▷ order posts by votes
  if $r = R$ **then**
    Send $(\texttt{output}, \mathcal{P})$ to $\mathcal{E}$
    Halt
  else
    $r \leftarrow r + 1$
  end if
end if

Algorithm 6 $\textsc{Vote}\left(\mathcal{P}, \text{aux}\right)$

```
Store aux contents as constants
if VoteThisRound(r) = yes then
    top ← ChooseTopPosts(attSpan, 𝒫, votedPosts)
    (i, l) ← argmax_{(i,l)∈top} {l_pid}
    votedPosts ← votedPosts ∪ (i, l)
    return ((i, l), l_pid)
else
    return null
end if

function ChooseTopPosts(attSpan, 𝒫, votedPosts)
    res ← ∅
    idx ← 1
    while |res| < attSpan & idx ≤ |𝒫| do
        if 𝒫[idx] ∉ votedPosts then                ▷ One vote per post per player
            res ← res ∪ {𝒫[idx]}
        end if
        idx ← idx + 1
    end while
    return res
end function

function VoteThisRound(r, |𝒫|)
    Let choices be a vector of length R, with each element in {0, 1}. The vector
    choices is such that, if the player votes only on the rounds R when choices_r = 1 and
    the weight of all votes is 1, then the total player's "influence" will be maximized.
    Or simply allow voting when either voting power is full or in evenly spread out
    moments. (This strategy may actually achieve the above.)
    return choices_r
end function
```

Theorem 1. *The Steem system* (N, R, M, t)*-converges if and only if* $\boldsymbol{SP}$ *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. – ($\Leftarrow$) Let $R-1 \geq (M-1)\left\lceil \frac{a+b}{\text{regen}} \right\rceil$, $\text{pid} \in [N]$ and $\forall i \in [N], \mathbf{SP}_i = c$. In this case and according to $\textsc{VoteThisRound}$, u_{pid} will vote in rounds $r \in \text{voteRounds} = \left\{ r \in [R] : \left\lfloor (r-1) \mod \frac{R-1}{M-1} \right\rfloor = 0 \right\}$. Let

$(r_1, \ldots, r_M)$ be the successive rounds when u_{pid} votes ($r_1 \in \text{voteRounds}, \forall i \in [|\text{voteRounds}|] \setminus \{1\}, r_i \in \text{voteRounds} \wedge r_i > r_{i-1}$). It holds that:

1. $\forall i \in [|\text{voteRounds}|] \setminus \{1\}, r_i \in \left\{r_{i-1} + \left\lfloor \frac{R-1}{M-1} \right\rfloor, r_{i-1} + \left\lceil \frac{R-1}{M-1} \right\rceil \right\}$
2. $R - 1 \geq (M-1) \left\lceil \frac{a+b}{\text{regen}} \right\rceil \Rightarrow \frac{R-1}{M-1} \geq \left\lceil \frac{a+b}{\text{regen}} \right\rceil \overset{\text{rhs integer}}{\Rightarrow} \left\lfloor \frac{R-1}{M-1} \right\rfloor \geq \left\lceil \frac{a+b}{\text{regen}} \right\rceil$

From (1) and (2) we have that $\forall i \in [R-1], r_{i+1} - r_i \geq \left\lceil \frac{a+b}{\text{regen}} \right\rceil$. Thus $\mathbf{VP}_{\text{pid},r_{i+1}} \geq \mathbf{VP}_{\text{pid},r_i} + \text{regen} \left\lceil \frac{a+b}{\text{regen}} \right\rceil \Rightarrow \mathbf{VP}_{\text{pid},r_{i+1}} \geq \mathbf{VP}_{\text{pid},r_i} + a + b$. But $\mathbf{VP}_{\text{pid},1} = 1$ and $\forall i \in [\text{voteRounds}] \setminus \{1\}, \mathbf{VP}_{\text{pid},r_i+1} \geq \mathbf{VP}_{\text{pid},r_i} - a - b$, 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.

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

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

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 $\forall (i, l) \in \mathcal{P}[1..R], l = (0.5, \ldots, 0.5)$ and $\forall (i, l) \in \mathcal{P}[R+1..M], l = (1, \ldots, 1)$, the Steem system does not $(N, r, M, 1)$-converge.
  - If $p_{\text{bad}} = (i, l_{\text{bad}}) = \mathcal{P}[1]$ with $l_{\text{bad}} = (1 - \epsilon_M + \epsilon, \ldots, 1 - \epsilon_M + \epsilon)$, $p_{\text{good}} = (i, l_{\text{good}}) = \mathcal{P}[M]$ with $l_{\text{good}} = (1, \ldots, 1)$ 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 $\textsc{Ideal}^1(\mathcal{P}')$ holds, it is $\mathcal{P}'[1] = p_{\text{good}}$. However, after the execution the score of p_{bad} will be $\text{sc}_{bad} = Nc(a(1 - \epsilon_M + \epsilon) + b)$, whereas the score of p_{good} will

be $\mathrm{sc}_{good} = Nc\,(a\,(1-\epsilon_M)+b) < \mathrm{sc}_{bad}$, 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