

Analysis and Attacks of decentralized content curation platforms

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

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

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 (Post score). Let post $p = (m, l)$. We define the score of p as $\text{sc}(p) = \sum_{i=1}^N l_i$.

The 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 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

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

Definition 4 (Post-Voting System). A tuple $\mathcal{S} = (\mathcal{G}_{\text{Feed}}, \Pi_{\text{honest}})$ of two ITMs, parametrized by INIT , AUX , HANDLEVOTE and VOTE . The two ITMs have to implement the following API:

$\mathcal{G}_{\text{Feed}}$ 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 can do whatever the functionality wants. Eventually, the functionality sends a message (**output**, \mathcal{P}) to \mathcal{E} and halts.¹

Π_{honest} is a protocol that sends **read** and **vote** messages to $\mathcal{G}_{\text{Feed}}$ whenever activated by \mathcal{E} .

¹ Alternatively \mathcal{E} decides when to stop, but seems dirtier

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

```

1: Initialization:
2:    $\forall i \in |\mathcal{P}|$ , Parse  $\mathcal{P}[i]$  as  $(\text{creator}_i, l_i)$ 
3:   Assert( $\forall i \neq j \in |\mathcal{P}|, |l_i| = |l_j|$ )
4:    $N \leftarrow |l_1|$ 
5:    $\mathcal{U} \leftarrow \emptyset$ 
6:    $\text{INIT}^{\text{Feed}}(\text{initArgs})$ 
7:
8: Upon receiving (read) from  $u_{\text{pid}}$ :
9:    $\text{aux} \leftarrow \text{AUX}(u_{\text{pid}})$ 
10:  Send (posts,  $\mathcal{P}$ ,  $\text{aux}$ ) to  $u_{\text{pid}}$ 
11:
12: Upon receiving (vote,  $\text{ballot}$ ) from  $u_{\text{pid}}$ :
13:    $\mathcal{U} \leftarrow \mathcal{U} \cup u_{\text{pid}}$ 
14:   if  $|\mathcal{U}| > N$  then
15:     Abort
16:   end if
17:    $\text{HANDLEVOTE}(\text{ballot})$ 

```

Algorithm 2 $\Pi_{\text{honest}} \left(\text{INIT}^{\text{honest}}, \text{VOTE} \right) (\text{initArgs})$

```

1: Initialization:
2:    $\text{INIT}^{\text{honest}}(\text{initArgs})$ 
3:
4: Upon receiving (activate) from  $\mathcal{E}$ :
5:   Send (read) to  $\mathcal{G}_{\text{Feed}}$ 
6:   Wait for response (posts,  $\mathcal{P}$ ,  $\text{aux}$ )
7:    $\text{ballot} \leftarrow \text{VOTE}(\mathcal{P}, \text{aux})$ 
8:   Send (vote,  $\text{ballot}$ ) to  $\mathcal{G}_{\text{Feed}}$ 

```

Definition 5 (t -convergence under honesty). *We say that a post-voting system $\mathcal{S} = (\mathcal{G}_{\text{Feed}}, \Pi_{\text{honest}})$ t -converges under honesty if, for every valid input \mathcal{P} , for every \mathcal{E} and given that all protocols execute Π_{honest} , eventually $\mathcal{G}_{\text{Feed}}$ sends a single message (**output**, \mathcal{P}) to \mathcal{E} such that $\text{IDEAL}^t(\mathcal{P})$ holds.*

Definition 6 (Steem system). *The Steem system is the post voting system \mathcal{S} with parameters $\mathbf{SP} \in \mathbb{N}^{*N}$, $R \in \mathbb{N}^*$, $a, b, c \in \mathbb{R}_+$, $\text{attSpan} \in \mathbb{N}^*$ and the following parametrizing procedures:*

Algorithm 3 $\text{INIT}^{\mathcal{G}_{\text{Feed}}}(\mathbf{SP}, R, a, b, c)$

```
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 $\text{INIT}^{\text{honest}}(\mathbf{SP}, \text{attSpan}, \text{pid}, R, a, b, c)$

```
1: Store input parameters as constants
2:  $\text{votedPosts} \leftarrow \emptyset$ 
```

Algorithm 5 **AUX**

```
1: return ▷ TODO: DISCUSS: Maybe should send votes?
```

Algorithm 6 $\text{HANDLEVOTE}(\text{ballot}, u_{\text{pid}})$

```
1: if  $\text{lastVoted}_{\text{pid}} \neq r$  then ▷ One vote per player per round
2:    $\mathbf{VP}_{\text{pid}} \leftarrow \max\{\mathbf{VP}_{\text{pid}} + c \cdot (r - \text{lastVoted}_{\text{pid}}), 1\}$ 
3:   if  $\text{ballot} \neq \text{null}$  then
4:     Parse ballot as  $(p, \text{weight})$ 
5:      $\text{score} \leftarrow a \cdot \mathbf{VP}_{\text{pid}} \cdot \mathbf{SP}_{\text{pid}} \cdot \text{weight} + b$ 
6:      $\mathbf{VP}_{\text{pid}} \leftarrow \mathbf{VP}_{\text{pid}} - (a \cdot \mathbf{VP}_{\text{pid}} \cdot \text{weight} + b)$ 
7:      $\text{scores}_p \leftarrow \text{scores}_p + \text{score}$ 
8:   end if
9:    $\text{lastVoted}_{\text{pid}} \leftarrow r$ 
10: end if
11: if  $\forall i \in [N] \text{ lastVoted}_i = r$  then ▷ round over
12:    $\mathcal{P} \leftarrow \text{ORDER}(\mathcal{P}, \text{scores})$  ▷ order posts by votes
13:   if  $r = R$  then
14:     Send  $(\text{output}, \mathcal{P})$  to  $\mathcal{E}$ 
15:     Halt
16:   else
17:      $r \leftarrow r + 1$ 
18:   end if
19: end if
```

Algorithm 7 VOTE (\mathcal{P})

```
1: if VOTETHISROUND( $r, R$ ) = yes then
2:   TODO: DISCUSS NOTATION: add attSpan, votedPosts to inputs?
3:   CONT: but complex for  $\Pi_{\text{honest}}$ 
4:   top  $\leftarrow$  CHOOSETOPPOSTS(attSpan,  $\mathcal{P}$ , votedPosts)
5:    $p \leftarrow \underset{(i,l) \in \text{top}}{\text{argmax}} \{l\}$ 
6:   votedPosts  $\leftarrow$  votedPosts  $\cup p$ 
7:   return ( $p, l_{\text{pid}}$ )
8: else
9:   return null
10: end if
11:
12: function CHOOSETOPPOSTS(attSpan,  $\mathcal{P}$ , votedPosts)
13:   res  $\leftarrow \emptyset$ 
14:   idx  $\leftarrow 1$ 
15:   while |res| < attSpan & idx  $\leq |\mathcal{P}|$  do
16:     if  $\mathcal{P}[\text{idx}] \notin \text{votedPosts}$  then ▷ One vote per post per player
17:       res  $\leftarrow$  res  $\cup \{\mathcal{P}[\text{idx}]\}$ 
18:     end if
19:     idx  $\leftarrow$  idx + 1
20:   end while
21:   return res
22: end function
23:
24: function VOTETHISROUND( $r, R$ )
25:   TODO
26: end function
```

Theorem 1. *The Steem system t -converges under honesty, assuming the following conditions on $\mathbf{SP}, R, a, b, c, \text{attSpan}$: TODO: continue*

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