# SkyHash: a Hash Opinion Dynamics Model

Houwu Chen and Jiwu Shu*
Department of Computer Science and Technology
Tsinghua University
{chenhw11@mails., shujw@}tsinghua.edu.cn



***Abstract*—This paper proposes the first hash opinion dynamics model, named SkyHash, that can help a P2P network quickly reach consensus on hash opinion. The model consists of a bit layer and a hash layer, each time when a node shapes its new opinion, the bit layer is to determine each bit of a pseudo hash, and the hash layer is to choose a hash opinion with minimum Hamming distance to the pseudo hash. With simulations, we conducted a comprehensive study on the convergence speed of the model by taking into account impacts of various configurations such as network size, node degree, hash size, and initial hash density. Evaluation demonstrates that using our model, consensus can be quickly reached even in large networks. We also developed a denial-of-service (DoS) proof extension for our model. Experiments on the SNAP dataset of the Wikipedia who-votes-on-whom network demonstrate that besides the ability to refuse known ill-behaved nodes, the DoS-proof extended model also outperforms Bitcoin by producing consensus in 45 seconds, and tolerating DoS attack committed by up to 0.9% top influential nodes.**

## I. Introduction

*Opinion dynamics* is a field which utilizes computational tools or mathematical-and-physical models to explore the dynamical processes of the diffusion and evolution of opinions in a society, where individuals constantly shape their opinions based on the opinions they receive from a subset of the society [1]. Existing study show that opinion dynamics can be used for sybil-proof consensus in P2P networks, without the disadvantage of best known approach based on computational challenge which can't resist adversary with dominant compute power [2], [3].

Generally, to commit transactions in P2P networks such as cryptocurrencies, objects to agree at are the hashes calculated from the transactions [4]. However, even a plethora of opinion dynamics models are proposed for binary opinion(e.g. majority rule, voter and Sznajd), continuous opinion(e.g. Deffuant and Hegselmann-Krause) and vector opinion(e.g. Axelrod) [5], no model for hash opinion exists.

This paper proposes the **first** hash opinion dynamics model named *SkyHash*. Each node in a P2P network shapes its opinion by rounds, in each round it receives opinions, applies the SkyHash model to determine its new opinion, and then shares the new opinion. The SkyHash model consists of a bit layer and a hash layer, where for each round of a node, the bit layer is to determine each bit of a pseudo hash, and the hash layer is to choose a hash opinion with minimum Hamming distance to the pseudo hash. Simulations indicate that the number of rounds needed for full convergence increases at the speed of $log_D N$, where $D$ is average node degree and $N$ is network size. Simulations also show that convergence performance increases with hash size as well as initial hash density. Our evaluation demonstrates that using our model, consensus can be quickly reached even in large networks. For example, for a homogeneous network with 20000 nodes, average node degree is 33 and 256-bit hash size, consensus can be reached within only 14 rounds. To tolerate denial of service (DoS) attack which prevents a P2P network to agree at *well* hashes, we developed a DoS-proof extension for the model. Experiments on the SNAP dataset of the Wikipedia who-votes-on-whom network [6] demonstrates that besides the advantage of opinion dynamics based consensus to refuse known ill-behaved nodes [3], the DoS-proof extended model also outperforms Bitcoin by producing consensus in 45 seconds, and tolerating DoS attack committed by up to 0.9% top influential nodes.

## II. Problem and Datasets

P2P networks are assumed to be constructed by trust relationships. As shown in Fig. 1, when node $A$ trust a node $E_i$, $E_i$ is a **followee** of node $A$ whereas $A$ is a **follower** of $E_i$. Opinions flow from followees to followers unidirectionally. In this way, the network can be abstracted to a directed graph where each trust relationship is a directed edge.

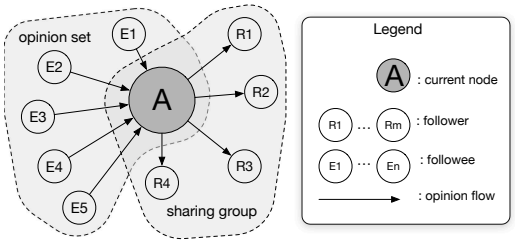



Fig. 1. Nodes relationships

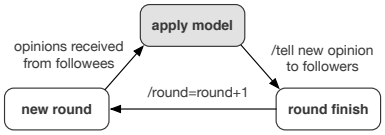



Fig. 2. Node State

During an opinion dynamics process, a node $A$ shapes its opinion **by rounds** as shown in Fig. 2. In each round, $A$ receives opinions from its followees, and those opinions

together with the opinion of $A$ itself form an *opinion set* as shown in Fig. 1. $A$ then applies the opinion dynamics model to determine its new opinion according to the opinion set, and tell its new opinion to its followers by sharing the new opinion to a *sharing group* (a group that consists of its followers, a.k.a. swarm) in Fig. 1. After that, $A$ enters the next round.

The proposed model is applied in the gray state in Fig. 2 to **produce the convergence of hash opinions** in the whole network. We use the terms **convergence** and **consensus** interchangeably in this paper.

Our strategy is to analyze the opinion dynamics model in a synchronous process where each round for each node takes exactly 1 unit of time, and it takes no time for opinions to determinately flowing from followees to followers. We then implement the model with practical asynchronous time assumption. Such an initial synchronous phase is sometimes called a conciliator [6].

Our model is evaluated on the SNAP dataset of Wikipedia who-votes-on-whom network [7], which presents trust relationships in the form of votes for administration and is named the *wiki* dataset in this paper. To ensure connectivity, we constrain that each well-behaved node in the P2P client has $indegree >= 10$, thus all nodes with less than 10 followees are removed. Parameters of such as network are shown in Table I, and the cumulative distribution of indegrees and outdegrees is shown in Fig. 3.

Table I
DATASETS PARAMETERS

| Name | Wiki |
|---|---|
| Nodes Counts | 998 |
| Average Degree | 33.33 |
| Diameter | 5 |
| Average Path Length | 2.34 |
| Density | 0.033 |
| Average Clustering Coefficient | 0.183 |
| Eigenvector Centrality Sum Change | 0.029 |

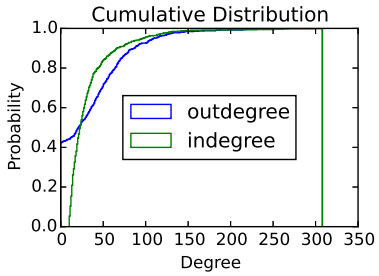



Fig. 3. Degree distribution of the wiki dataset

To reveal the impact of network size, we also run simulations on homogeneous networks by varying the network size to 100, 1000, 5000 and 20000 nodes. In each configuration each node has the same indegree as the average degree of the *wiki* dataset, but they are connected to each other randomly. Those datasets are named *uniform-l00*, *uniform–1k*, *uniform–5k* and *uniform–20k* respectively.

## III. THE SKYHASH MODEL

The hash opinion dynamics model can be abstracted to a function $F$ applied to an opinion set $H$ to produce a hash value $H_x$, s.t. $H_x = F(H)$, where the opinion set $H = \{H_0, H_1, H_2, \cdots H_n\}$, $H_0$ is the hash opinion of the current node, and $H_i$ for $i \in [1, n]$ is the hash opinion of the $i$-th of $n$ followees.

The SkyHash model we proposed consists of a bit layer and a hash layer as shown in Fig. 4.

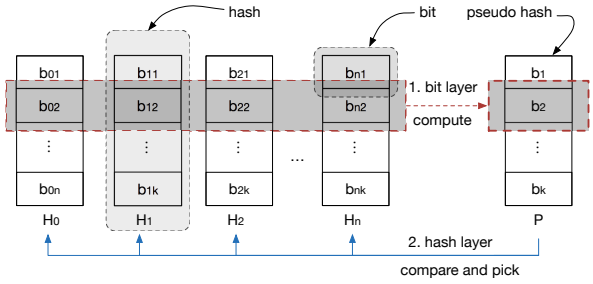



Fig. 4. The SkyHash model

1: **function** BITSKY($\{b_{0j}, b_{1j}, \cdots b_{ij}\}$)
2: $n_0 \leftarrow$ count 0 in $\{b_{0j}, b_{1j}, \cdots b_{ij}\}$
3: $n_1 \leftarrow$ count 1 in $\{b_{0j}, b_{1j}, \cdots b_{ij}\}$
4: **return** randomly pick in {BITMR($n_0$, $n_1$), BITSA($n_0$, $n_1$)}
5: **function** BITMR($n_0, n_1$)
6: **if** $n_0 > n_1$ **then**
7: **return** 0
8: **else if** $n_1 > n_0$ **then**
9: **return** 1
10: **return** randomly pick in $\{0, 1\}$
11: **function** BITSA($n_0, n_1$)
12: $n \leftarrow n_0 + n_1$
13: **if** $n_0 > n * 0.8$ **then**
14: **return** 0
15: **else if** $n_1 > n * 0.8$ **then**
16: **return** 1
17: $test \leftarrow$ randomly pick in $[0, n]$
18: **if** $test < n_0$ **then**
19: **return** 0
20: **else if** $test > n_0$ **then**:
21: **return** 1
22: **return** randomly pick in $\{0, 1\}$

Fig. 5. Bit layer algorithm

### A. Bit Layer

Bit layer is applied at each bit position $j$ for $j \in [1, k]$, where $k$ is the hash size. First, for each hash $H_i$, bit at position $j$ of $H_i$ marked as $b_{i,j}$ is extracted. Then a value $b_j$ is determined on the bit set $\{b_{0j}, b_{1j}, \cdots b_{ij}\}$ according to the *Sky* bit layer model [3], which is a mix of a majority

rule model and a simulated annealing model. Bit layer model can be implemented as the function *BITSKY* in Fig. 5, where *BITMR* is the majority rule model which mainly picks the majority opinion, and *BITSA* is the simulated annealing model which mainly picks an opinion with probability corresponding to the density of the opinion.

### B. Hash Layer

After applying the bit layer at each bit position $j$ to determine a bit $b_j$, a *pseudo hash* $P$ can be constructed as $P = b_0 b_1 b_2 \cdots b_k$. This layer then computes the Hamming distance between each $H_i$ and $P$, and picks from H the hash Hx with minimum hamming distance, as shown by function *HASH* in Fig. 6.

```
1: function HASH({H_0, H_1, H_2, ··· H_n}, P)
2:     min_d ← k + 1                          ▷ k is hash size
3:     for all H_i in {H_0, H_1, H_2, ··· H_n} do
4:         test ← bitwisely apply xor between H_i and P
5:         d ← count 1 in all bits of test
6:         if d < min_d then
7:             H_x = H_i
8:             min_d = d
9:     return H_x
```

Fig. 6. Hash layer algorithm

### C. Simulations

Impacts of various factors on convergence performance are revealed by simulation results exhibited in Fig. 7 .

Fig. 7(a) demonstrates the impact of *network size* on convergence performance. Simulations are executed on various datasets where hash size is 256-bit, and initially each node hold an random hash opinion. The horizontal axis of Fig. 7(a) is the round of the network, where all nodes are always at the same round as stated in section II. The vertical axis of Fig. 7(a) is the density of the **top hash** which is the hash opinion hold by the most number of nodes in the whole network. The figure unfolds the following results:

- For homogeneous networks with the same average degree (e.g. all the *uniform-** datasets), while the number of rounds needed increases with the network size, the speed of increase is slow. e.g., when network size increase from 100 to 20000, round to converge needed only increases from 6 to 14.
- For heterogeneous network, e.g, the *wiki* dataset, convergence performance decreases remarkably comparing to the homogeneous network with same size and average node degree. Similar result is also observed in existing studies which shows that community strength in a heterogeneous network impacts the performance significantly [8], [9].
- Convergence increases quickly at the intermediate rounds for all datasets, however, for slower simulations, it takes more time to escape from disorder when convergence is near 0 and to full order when convergence is near 1.

Fig. 7(b) demonstrates the impact of *average node degree* on convergence performance. Simulations are executed on homogeneous networks with 1000 nodes, but with various average node degree respectively. Also, each node holds a random hash opinion initially, and the hash size is 256-bit. The horizontal and vertical axises are same with Fig. 7(a). The figure shows that for fixed sized homogeneous networks, convergence performance increases with degree.

Data in Fig. 7(a) and Fig. 7(b) for homogeneous networks is presented again in Fig. 7(c), where horizontal axis is $log_D N$, $D$ is average node degree and $N$ is network size. The figure shows that the round needed for full consensus linearly increases with $log_D N$.

Fig. 7(d) demonstrates the impact of *hash size* on convergence performance. Simulations are executed on the uniform–1k dataset with various hash sizes while each node hold a random hash opinion initially. The horizontal and vertical axises are same with Fig. 7(a). The figure shows convergence performance increases with hash size for a given dataset.

Fig. 7(e) demonstrates the impact of *initial hash density* on convergence. Simulations are executed on the uniform–1k dataset with 256-bit hash size and various numbers of initial opinions hold by all the nodes evenly. For example, for the 2 initial opinions case, there are two hash opinions in the whole network, and each hash opinion is hold by half number of the nodes. The figure shows that for a given dataset, convergence performance decrease with initial opinions count.

## IV. The DoS-proof Extension

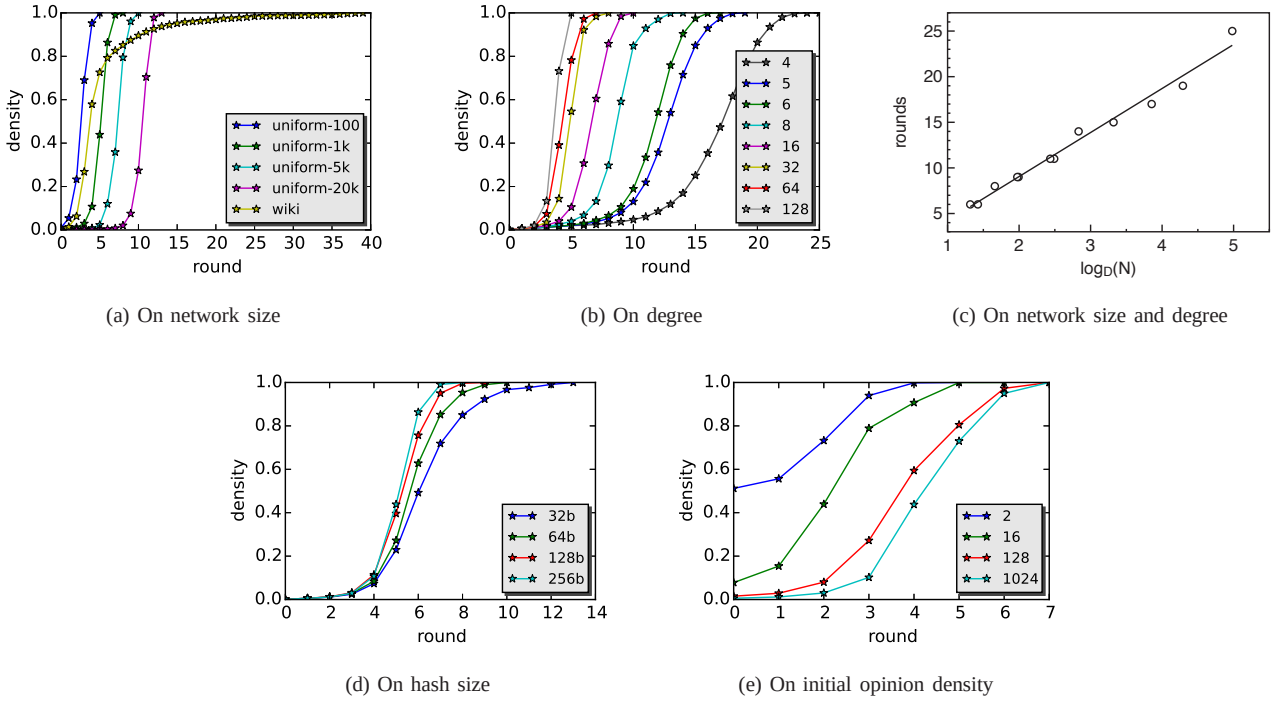
### A. Denial of Service Attack

Nodes in a P2P network may be ill-behaved, and they do not obey the proposed model or even collude with other nodes. However, behaviors of ill-behaved nodes are constrained to ensure consistency between hash and the corresponding data, or they will be identified by well-behaved nodes. However, existing studies show that if ill-behaved nodes collude together to keep telling other nodes a fixed opinion disregarding the opinions of their followees, even a small number of such nodes may control the opinion of the whole network. Such nodes are usually called **stubborn agents** or **committed minorities** [10], [11].

Simulations (omitted in this paper due to space limitation) reveal that even 0.5% of such nodes can prevent the whole network to agree at **well hashes** proposed by well-behaved nodes, and only **ill hashes** proposed by ill-behaved nodes are agreed at. Such a case is named denial of service (DoS) attack.

### B. The Extended Model

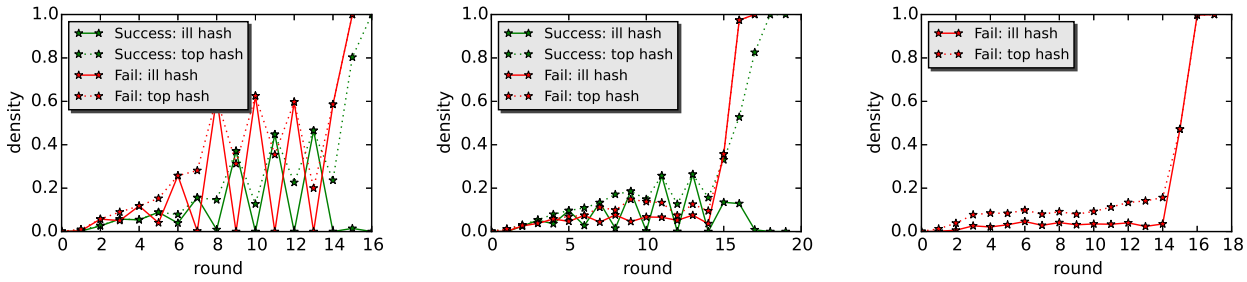
Based on the observation that the higher density of well top opinion, the stronger to tolerate attack [3], we proposed a DoS-proof extension consisting of two phases: a *reverse phase* and a *normal phase*. The extension can be implemented as algorithm described in Fig. 9, where *HASH* (the normal phase) is exactly the one in Fig. 6, and *RHASH* (the reverse phase) is almost same as *HASH* except it picks the hash $H_x$ with the **maxium** Hamming distance.




(a) On network size (b) On degree (c) On network size and degree

(d) On hash size (e) On initial opinion density

Fig. 7. Simulation on various factors




(a) DoS attack by 11% nodes (b) DoS attack by 15% nodes (c) DoS attack by 20% nodes

Fig. 8. Simulation of DoS-proof extension

### C. Simulations

With round threshold $R = 15$, simulations on the uniform–1k dataset with DoS attack from 11%, 15% and 20% nodes respectively are shown in Fig. 8. The horizontal and vertical axises are same with Fig. 7(a). Green lines are the cases that for each case the network succeeds to tolerate the DoS attack, where all well-behaved nodes agrees at a well hash. Red lines are the cases that for each case the network fails to tolerate the DoS attack, where all well-behaved nodes agrees at the ill hash. Solid lines are the density of the ill hash, and dashed lines are the density of the **top hash** (may be well hash or ill hash), which is the hash opinion hold by the most number of nodes in the whole network.

The network is able to survive DoS attack by less than 15% nodes, but 50% of the runs agree on ill hashes, thus the throughput will decrease by 50%. Fig. 8(a) and Fig. 8(b) demonstrate the oscillation of the density of the ill hash, and show that the heavier the attack the smaller range the oscillation, until the oscillation is insignificant and in all runs the network always agrees on the ill hash as in Fig. 8(c).

1: **function** HASHDOS($\{H_0, H_1, H_2, \cdots H_n\}$, $P$, r) ▷ $r$ is the round number
2: **if** $r < R$ **then** ▷ $R$ is a given threshold
3: **return** RHASH($\{H_0, H_1, H_2, \cdots H_n\}$, $P$)
4: **return** HASH($\{H_0, H_1, H_2, \cdots H_n\}$, $P$)

Fig. 9. DoS-roof extension algorithm

## V. Implementation and Experiments

### A. Implementation

To implement the model, each node publishes a public key as its identity, and each opinion the node shares is signed by its private key. A sharing group (known as a "swarm") is identified by the public key of a node, and its followers join the swarm by finding the public key in a distributed hash table (DHT).

For each node, a **failure detector** is utilized to deal with the FLP impossibility problem in asynchronous system [12], [13]. As shown in Fig. 10, the failure detector maintains an **active** followees list as well as a **suspect** followees list (as state $E$ and $G$). A followee is initially in the active list, it is moved to the suspects list (as action in $G \rightarrow H$) if no up-to-date opinion is received when **timeout**(as $C \rightarrow G$), and moved back to active list if a new up-to-date opinion is received (as $E \rightarrow F$).

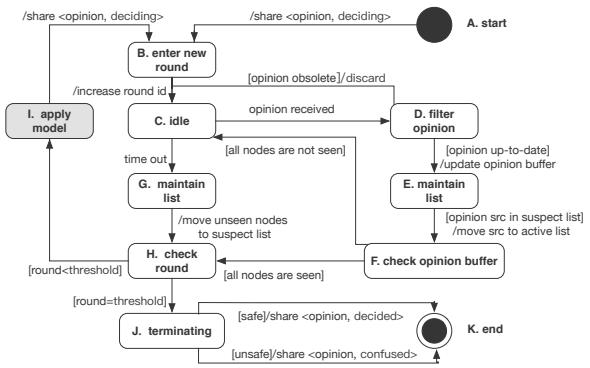



Fig. 10. Node state diagram

Each opinion is attached with a flag in {*deciding, decided, confused*} denoting the current **status** of the corresponding node. A node starts with *deciding* (as action in $A \rightarrow B$), and keep this flag (as action in $I \rightarrow B$) until a given round threshold is reached (as condition in $H \rightarrow J$) and then the node is terminating the current consensus process (as state $J$). It then share its opinion with flag *decided* if over 2/3 of its active followees share the same opinion (as $J \rightarrow K$ with condition *safe*), or share its opinion with flag *confused* (as $J \rightarrow K$ with condition *unsafe*).

An **opinion filter** (as state $D$) is utilized by each node to check whether an opinion received is up-to-date or not. When an opinion is shared by a node, it also attaches the current **round** number. An opinion is considered to be up-to-date by a follower if $opinion.round \geq follower.round$ or $opinion.status \in \{decided, confused\}$.

Each node also maintains an **opinion buffer**, and for each of its followees only the newest single opinion is kept in the buffer. As a result each time a node receives an up-to-date opinion, it then check the opinion from the same followee in the opinion buffer, and if the newly received opinion is attached with a greater round number, it is saved in the opinion buffer and the original opinion from the same followee is discarded (as shown in $D \rightarrow E$).

If a node sees absence of opinions from its active followees in the buffer, it continues to wait (as condition of $F \rightarrow C$). Otherwise (as condition of $F \rightarrow H$), the current round of the node finishes (as state $H$), according to the of the magnitude relation between node's round number and the given threshold (as $H \rightarrow J$ or $H \rightarrow I$), the node either terminates (as state J) or applies the model (as state $I$), share new opinion (as $I \rightarrow B$), and enters the new round (as state $B$).

### B. Experiments

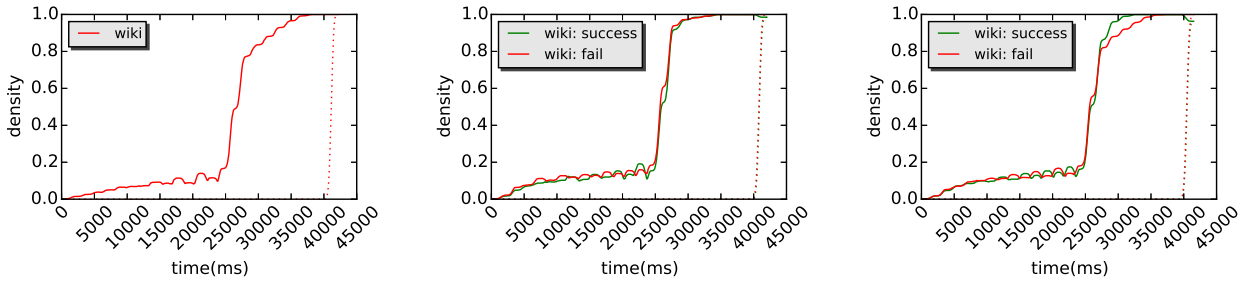
According to existing studies, latencies between peers in DHT are mostly between $50$ to $1000$ ms [14]. Our experiments employ a simply latency model that the times to deliver opinions conforms gauss distribution of ($\mu = 500$, $\sigma = 500$) with lower cutoff of 50 and no upper cutoff which means an opinion may be lost in a small probability. Additionally, $timeout = 2000$ and round threshold $R = 15$.

Fig. 11 exhibits the experiment results on the wiki dataset. The horizontal axis of each sub figure is time in unit of millisecond. The vertical axis of each sub figure is the density of the **top hash** which is the hash opinion hold by the most number of nodes in the whole network. Green lines are the cases that for each case the network succeeds to tolerate the DoS attack, where all well-behaved nodes agrees at a well hash. Red lines are the cases that for each case the network fails to tolerate the DoS attack, where all well-behaved nodes agrees at the ill hash. Solid lines are the density of the ill hash, and dashed lines are the density of the **top hash** (may be well hash or ill hash), which is the hash opinion hold by the most number of nodes in the whole network. In each sub figure, solid line is for all well-behaved nodes, while dashed line is for all nodes with opinions **decided** only.

The wiki dataset can survive DoS attack committed by 7% random nodes (as shown in Fig. 11(b)) or 0.9% top influential nodes defined as the top 0.9% nodes by sorting all nodes in descendant order on the count of a node's followees (as shown in Fig. 11(c)). However, the throughput will decrease 50% even when the network survives. In all the cases that the network survives, well-behaved nodes can always reach consensus within 45 seconds without well-behaved nodes agree at different values, however 1.5% and 4% nodes are *confused* respectively when under DoS attack by 7% random nodes or 0.9% top influential nodes. In contrast, Bitcoin produces consensus in about 10 minutes and it can not survive when even one single node has dominant compute power, and more severely, well-behaved nodes has no means to tolerate the power [4], whereas in our opinion dynamics based approach known ill-behaved nodes is harmless after being unfollowed by well-behaved nodes [3].

## VI. Related Work

Systematization of knowledge on opinion dynamics is introduced based on the viewpoint of statistical physics, and popular models including the voter model, majority rule




(a) Without DoS attack (b) DoS attack by 7% random nodes (c) DoS attack by 0.9% top influential nodes

Fig. 11. Experiments on the wiki dataset

model, models based on social impact theory, the Sznajd model, bounded confidence models and other models are briefly discussed in [1]. [5] gives a multidisciplinary review on opinion dynamics, and brief comparison of the various models is also given by categories. [3] is the first work to bring opinion dynamics into P2P network. However, no hash opinion dynamics model is introduced at present.

As the source of DoS attack exhibited in this paper, the presence of stubborn agents (another name is committed minorities) in opinion dynamics is also studied in [10], [11], [3], but their primary focus is the impact of those stubborn agents thus no countermeasure is provided.

Similar to our observation on performance decrease in heterogeneous network comparing to homogeneous network with same parameters, [8] reveals that the convergence time decays exponentially with increasing community strength. [15] points out that strongly coupled nodes within the same community synchronizing their opinions faster than other nodes. [16] further indicates a transition at a value of the interconnectivity parameter, and communities reach consensus or opposite opinions when above or below the value respectively.

## VII. Conclusion and Future Work

To sum up, using the *SkyHash* model, consensus can be quickly reached even in large P2P networks. The DoS-proof extension of the model is effective to tolerate DoS attack at the cost of throughput reduction. Experiments show that the model outperforms Bitcoin besides it has the ability to refuse known ill-behaved nodes. To the best of our knowledge, it's the first hash opinion dynamics model.

To circumvent the impact of communities on the convergence performance, we are developing a ground truth community aware opinion dynamics model which leverages known member inclusion information of communities (e.g. chat rooms). Preliminary simulations of the model exhibit vast improvement on convergence performance.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 61433008), the National High Technology Research and Development Program of China (Grant No. 2013AA013201), and Project of science and technology of Beijing City (Grant No. D151100000815003).

## References

[1] C. Castellano, S. Fortunato, and V. Loreto, "Statistical physics of social dynamics," *Rev. Mod. Phys.*, vol. 81, no. 2, pp. 591–646, 2009.

[2] J. Aspnes, C. Jackson, and A. Krishnamurthy, "Exposing computationally-challenged Byzantine impostors," Tech. Rep. YALEU/DCS/TR-1332, Yale University Department of Computer Science, July 2005.

[3] C. Houwu and S. Jiwu, "Sky: Opinion dynamics based consensus for p2p network with trust relationships," in *Proc. of 15th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP)*, Springer, Nov. 2015. (to appear).

[4] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies," pp. 104–121, IEEE, May 2015.

[5] H. Xia, H. Wang, and Z. Xuan, "Opinion Dynamics: A Multidisciplinary Review and Perspective on Future Research," *Int. J. Knowl. Syst. Sci.*, vol. 2, no. 4, pp. 72–91, 2011.

[6] J. Aspnes, "A Modular Approach to Shared-memory Consensus, with Applications to the Probabilistic-write Model," in *Proceedings of the 29th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, PODC '10, (New York, NY, USA), pp. 460–467, ACM, 2010.

[7] J. Leskovec and A. Krevl, "SNAP Datasets: Stanford large network dataset collection." http://snap.stanford.edu/data, June 2014.

[8] W. Ru and C. Li-Ping, "Opinion Dynamics on Complex Networks with Communities," *Chinese Physics Letters*, vol. 25, p. 1502, Apr. 2008.

[9] F. Gargiulo and S. Huet, "Opinion dynamics in a group-based society," *EPL (Europhysics Letters)*, vol. 91, p. 58004, Sept. 2010.

[10] J. Xie, S. Sreenivasan, G. Korniss, W. Zhang, C. Lim, and B. Szymanski, "Social consensus through the influence of committed minorities," *Physical Review E*, vol. 84, no. 1, p. 011130, 2011.

[11] E. Yildiz, A. Ozdaglar, D. Acemoglu, A. Saberi, and A. Scaglione, "Binary Opinion Dynamics with Stubborn Agents," *ACM Trans. Econ. Comput.*, vol. 1, no. 4, pp. 19:1–19:30, 2013.

[12] M. J. Fischer, N. A. Lynch, and M. S. Paterson, "Impossibility of distributed consensus with one faulty process," *J. ACM*, vol. 32, no. 2, pp. 374–382, 1985.

[13] T. D. Chandra and S. Toueg, "Unreliable failure detectors for reliable distributed systems," *J. ACM*, vol. 43, no. 2, pp. 225–267, 1996.

[14] F. Dabek, J. Li, E. Sit, J. Robertson, M. F. Kaashoek, and R. Morris, "Designing a DHT for low latency and high throughput," in *Proceedings of the 1st Conference on Symposium on Networked Systems Design and Implementation - Volume 1*, NSDI'04, (Berkeley, CA, USA), pp. 7–7, USENIX Association, 2004.

[15] R. Ghosh and K. Lerman, "The Impact of Network Flows on Community Formation in Models of Opinion Dynamics," *The Journal of Mathematical Sociology*, vol. 39, no. 2, pp. 109–124, 2015.

[16] R. Lambiotte and M. Ausloos, "Coexistence of opposite opinions in a network with communities," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2007, pp. P08026–P08026, Aug. 2007.