A Fast, Secure and Distributed Consensus Mechanism for Energy Trading Among Vehicles using Hashgraph

Gaurang Bansal, Ashutosh Bhatia
Birla Institute of Technology and Science, Pilani (BITS Pilani)

Abstract

Energy trading among inter vehicles (V2V) offers an efficient response to most of the problems presented by future electricity supply. V2V is often envisioned as a peer-to-peer (P2P) model of electric mercantilism for electric vehicle (EV) merchandising. With security vulnerability increasing, confidence in secured third parties is declining. Blockchain is becoming increasingly common as it provides a system for privacy conservation and efficient agreement without the need for trusted third parties. However, all operations in such a scheme are restricted by memory, time, computing capital, energy etc. and it is quite obvious that the mechanism for blockchain agreement is not sufficient to address all of them. In this paper, we present an alternative to the blockchain using Hashgraph which is scalable, fast, fault-tolerant and fair. It is efficient, inexpensive, and DoS resistant fulfilling the requirement of V2V energy trading.

Motivation for Distributed Consensus

The characteristics of the distributed network involve:
1) Independent Failure
2) No guaranteed in-order delivery of messages
3) No global clocks
4) No shared memory
5) Concurrent transactions can happen

Our objective is not just consensus, but to achieve security and privacy in distributed networks such that all parties can reliably come to an agreement, or consensus, about the state of the system, while maintaining resilience against bad actors

Contributions

1. We propose a scalable, fast, and secure distributed ledger based on the hashgraph for V2V communications
2. We also present a formal security proof of consensus of Hashgraph algorithm. This paper also provides security analysis to different distributed ledger technologies
3. Through experiments we demonstrate substantial improvement of hashgraph based communication over state-of-the-art approaches.

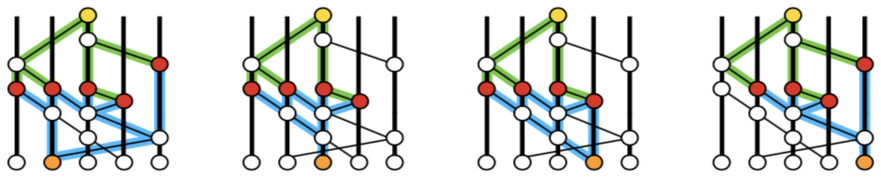
Hashgraph Overview

Gossip about gossip is called the history of how such occurrences are linked together by their parent hash. This history is expressed as a sort of acyclic (DAG) graph or hash graph. The hashgraph tells the story of the communication between the participants. All participants maintain a local copy of the hashgraph, which continues to be synchronized as participants.

- The first event for a member's node is that node's witness. Witness events are only responsible for exchanging virtual votes.
- A node X strongly sees node Y is if they are connected by multiple directed paths. It proves that if A and B strongly see each other, they are both able to calculate C's virtual vote.

A witness is called famous in a round if and only if it is seen by many other witnesses in round r+1. If Witness A is seen by the participants of the following round, they count as a vote in favor of Witness A. If a witness can not see Witness A in the next round, then that decision of a witness is that Witness A is said to be not famous. In order to make Witness A famous, Witness A is known to at least 2/3 of the voting participants.

Virtual Voting



For Hashgraph, there exists yellow event (Y) at the top, Red event (R) in middle and orange event (O) at bottom of each hashgraph. In fig. (d), O is an ancestor of each of 4, R intermediate events. Each of these R events is an ancestor of Y. So Y sees O. It holds true for all hashgraphs where Y sees O through 4 R's. If all 4 O's and parents of Y form a round. then Y is created in next round or round + 1 as it strongly sees more than 2/3 of total witnesses in a round.

Results

Hashgraph Security Evaluation: Performance comparison on multiple intrinsic and extrinsic security parameters. Overall, we observe that Hashgraph outperforms all the existing distributed ledger technologies.

Features →	Immutability	DoS Resistance	Fair Ordering	Fair Time Stamps
Central Server	✗	✗	✗	✗
Economy Based	✓	✗	✗	✗
Leader Based	✓	✗	✗	✗
Voting Based	✓	✓	✗	✗
Proof of Work Based	✓	✓	✗	✗
Our Model	✓	✓	✓	✓

Algorithm

Algorithm 1 Divide Rounds algorithm

Divide RoundsEvent x**while** $x \neq null$ **do**
if number of round of parents of x == none **then**
  $r \leftarrow 1$
else
  $r \leftarrow$ max (round of first parent of x, round of second parent of x)
end if
if x sees more than $2n/3$ witnesses in round r **then**
  $x.round = r + 1$
else
  $x.round = r$
end if
x.witness = (x.parent == null) ∨ (x.round > x.parent.round)
end while

- Algorithm 1 gives description of rounds are divided in hashgraph..
- Algorithm 2 gives an overview of fame calculation.

Ordering is achieved through the collection and median timelines of the earliest antecedents of the famous witnesses. Thus taking the median timestamp of those gathered events.

Algorithm 2 Fame Calculation algorithm

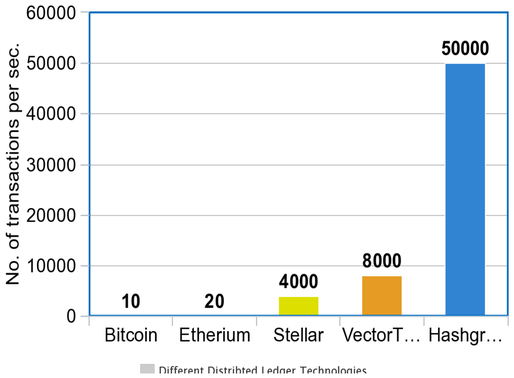
Deciding Fame**for** each event x **do**
for each event y **do**
  if (x.witness ∧ y.witness) ∧ (y.round−x.round> 0) **then**
    d ← difference between round of y and x
    s ← witness set in round y.round-1
    v ← majority vote in s
    t ← number of events with vote v in witness set in round y.round-1
    if d == 1 **then**
      if y can see x **then**
        y.vote ← TRUE
      else
        y.vote ← FALSE
      end if
    end if
    if t > 2*n/3 **then**
      x.famous ← v
      y.vote ← v
    else
      y.vote ← v
    end if
  end if
end for
end for

Formal Proof (Lemma)

An event X created by an honest has probability of 1 to be assigned in total ordering.

Proof. Using the gossip protocol, since majority of nodes are honest, all honest node will get the information regarding event X. Two honest nodes will definitely communicate resulting in all famous witness which are unique belong to event X. Hence event X will receive its timestamp and its position in consensus. Moreover, there does not exist any event Y that has round greater than X, and comes before event X in position of consensus order because then round of Y would be less than round of X.

Speed Comparison



The program is executed on Macbook Air with 1.8GHz dual-core Intel Core i5 processor equipped with 8 GB of RAM. The comparison of number of transactions per second in presented which depicts Hashgraph achieves speed of 50,000 transactions per second better than any other DLT.

Contact

Webpage: http://gaurang-bansal.github.io
Email: *h20140128@Pilani.bits-pilani.ac.in*