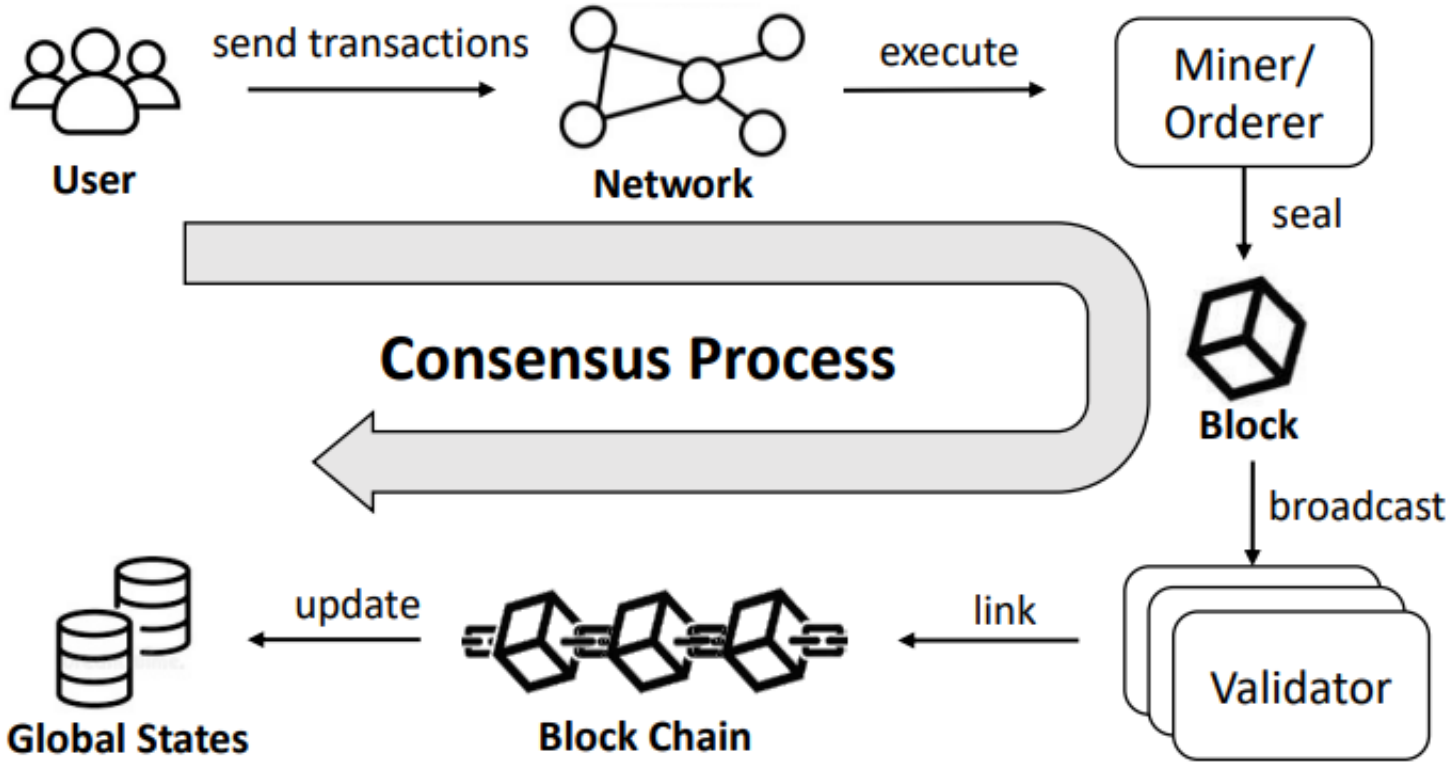
# Tyr: Finding Consensus Failure Bugs in Blockchain System with Behaviour Divergent Model

Yuanliang Chen, Fuchen Ma, Yuanhang Zhou, Yu Jiang, Ting Chen, and Jiaguang Sun

School of Software, Tsinghua University, KLISS, BNRist, Beijing, China

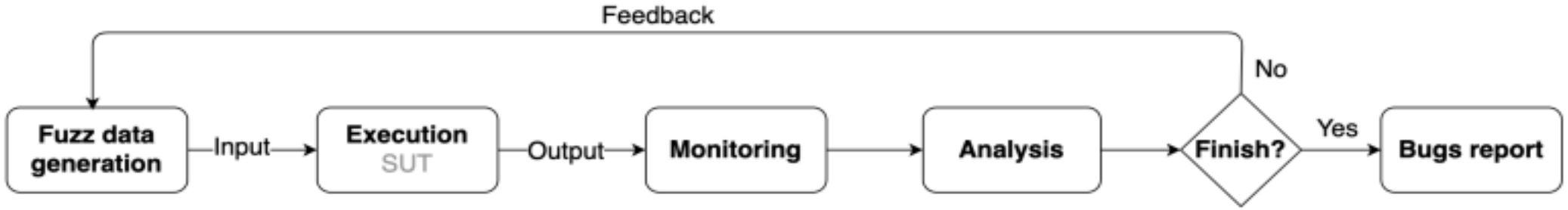
University of Electronic Science and Technology of China, Chengdu, China

# Blockchain Consensus




- Public Blockchain: Ethereum、Bitcoin.
  - Nakamoto Consensus: POW & POS
- Consortium Chain: Fabric、FISCO-BCOS、Quorum、Diem.
  - Committee-based Consensus: PBFT
- Message
  - Node Communication

# Fuzzing Technology




generate a with 2^n

```
1. a= Input()
2. if a=1:
3.         func1()
4. else if a=2:
5.         func2()
6. ...
7. ...
8. else if a=1024:
9.         bug()
```

No Bug

report Bug

- Metric:
  - Find bug&Flaw
  - branch coverage

# Previous Work

- **Fluffy**[ SOSP‘21]
  - Differential testing is used to detect formula vulnerabilities in blockchain clients of different progrem languages. ( e.g.,Ethereum geth(golang)，Parity（rust).)
  - Fabric , FISCOBCOS have only one implementation.

- **Peach**[industry]
  - Mutating network protocol field.
  - Only detect the target program exits normally.

- **Twins**[industry]
  - Detects byzantine behaviors in a mock environment.
  - Ignoring the runtime behaviors

# Contributions

- **Four consensus property** are constructed to detect bugs of the consensus system.
  - Liveness、Safety、Integrity、Fairness.
- Construct the **behavior divergent model** to diverge the behaviors of nodes and trigger the bug oracles.
- Tyr Framwork
  - Compatible with **six blockchain platforms**.
  - High branch coverage
  - 20 serious previously unknown bugs(**5 CVEs**)

# Main Idea

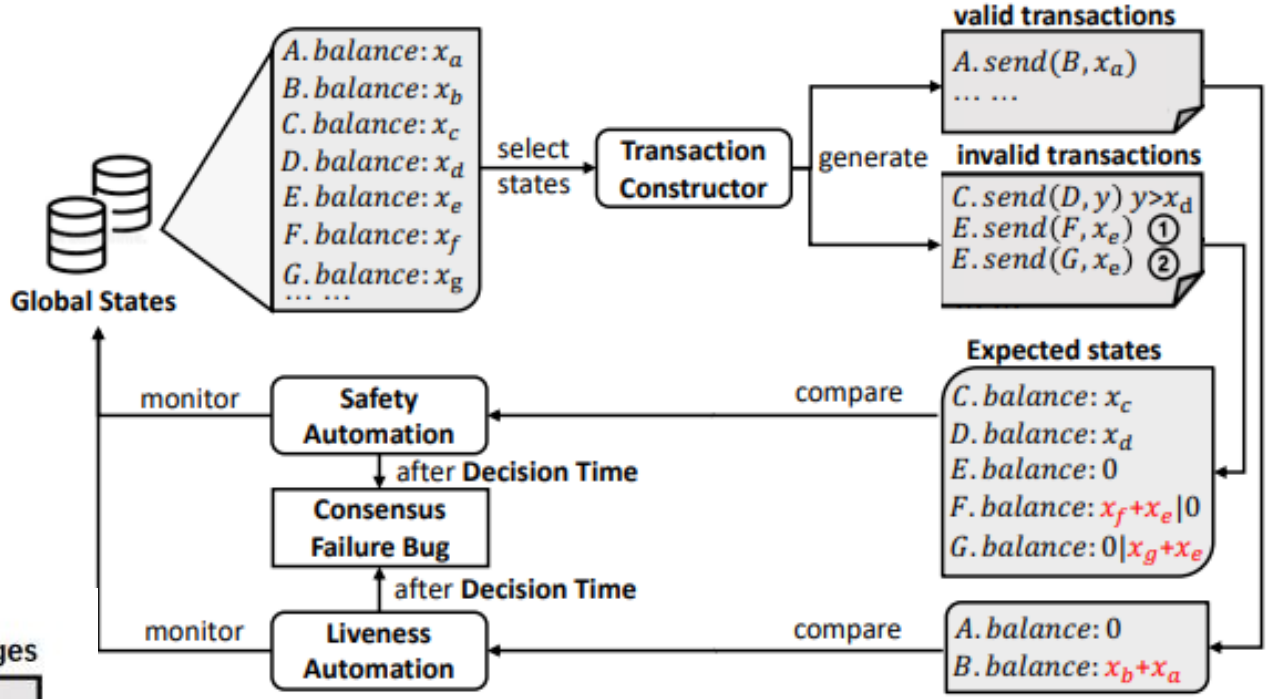
- **Four consensus property**
  - Proposed the rules of anomaly detection for fuzzer.

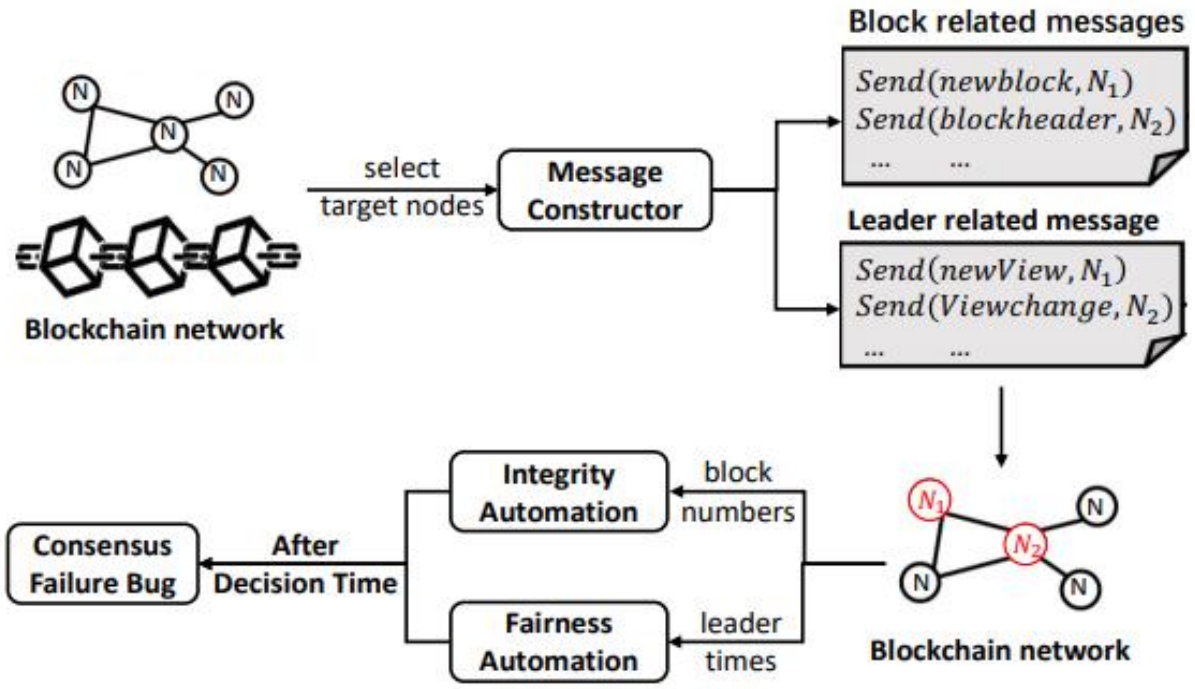
- **behavior divergent model**
  - Guide the blockchain system to violate the consensus property.

# Design

- **Four consensus property**
  - **Liveness**
    - all valid transactions must be executed, committed and stored in a specific block eventually.
  - **Safety**
    - Any invalid transactions are not allowed to be executed, committed, or stored in any blocks.
  - **Integrity**
    - any block with the same block height should be equivalent to each other in all nodes.
    - block syncing mechanism should work normally. There is no node isolation in this network.
  - **Fairness.**
    - all nodes should have a fair possibility to be elected as the leader node or miner node

Safety and Liveness




Integrity

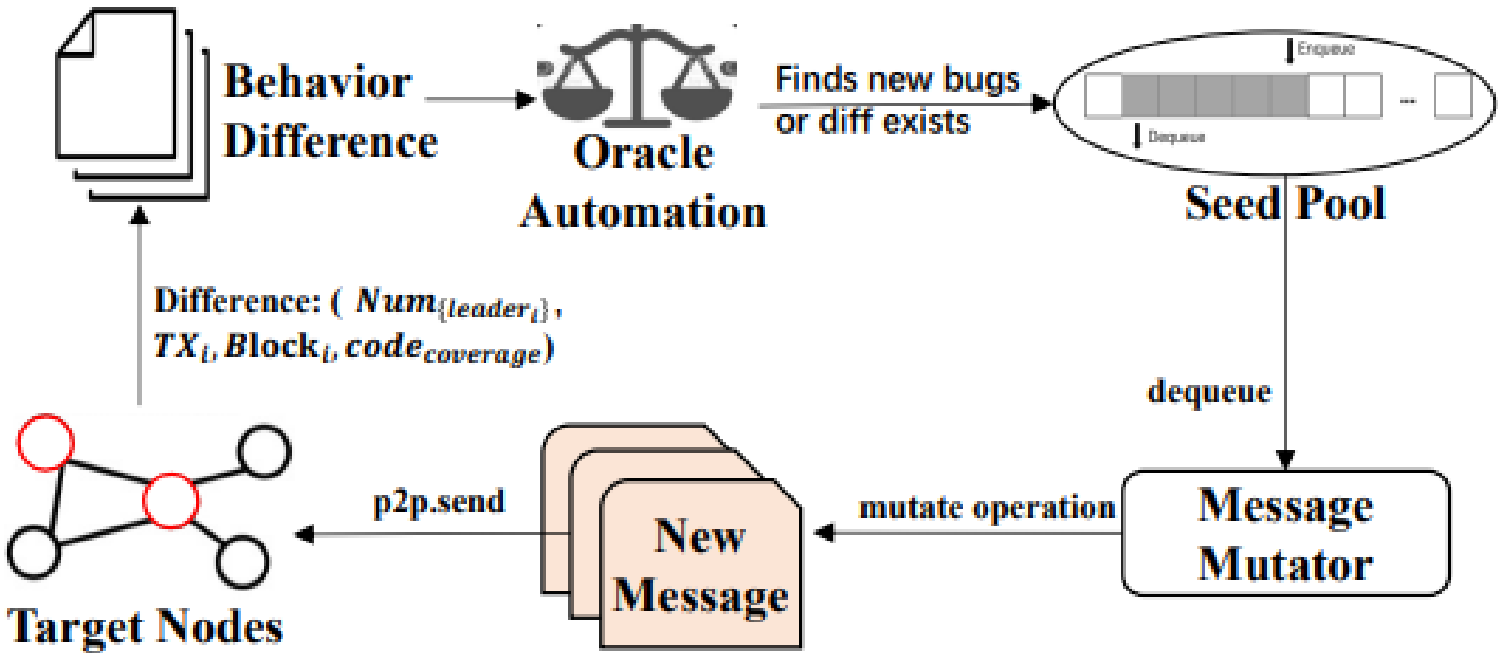
$$(diff_{ij} = |Height_{block_i} - Height_{block_j}|/Height_{block_i})$$

Fairness

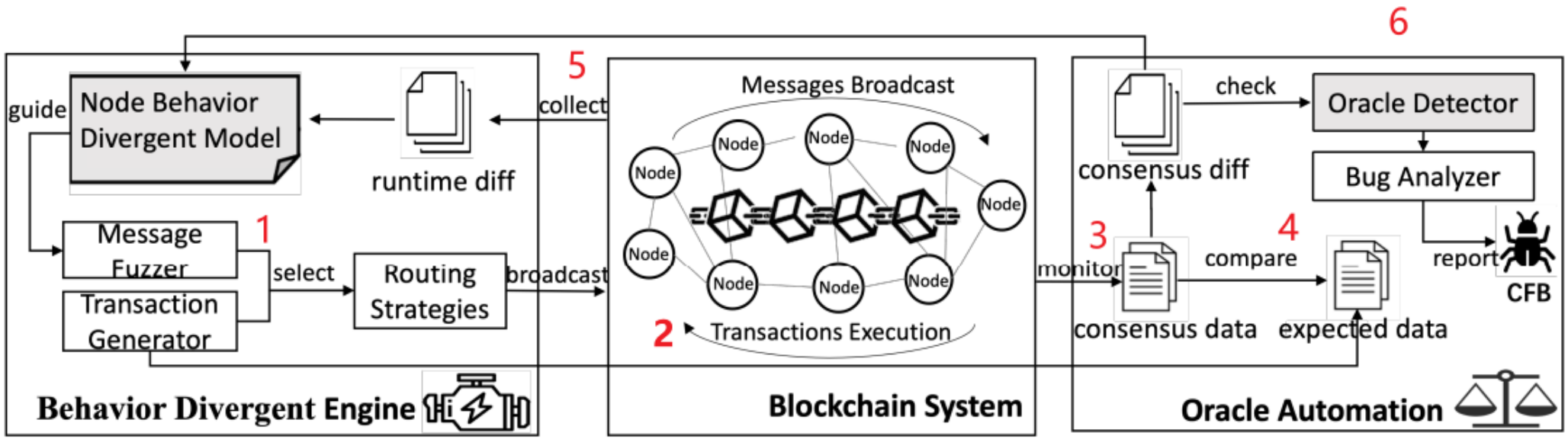
$$(diff_{ij} = |Num_{leader_i} * P_j - Num_{leader_j} * P_i|/Num_{leader_i} * P_j)$$

# Behavior Divergent Engien

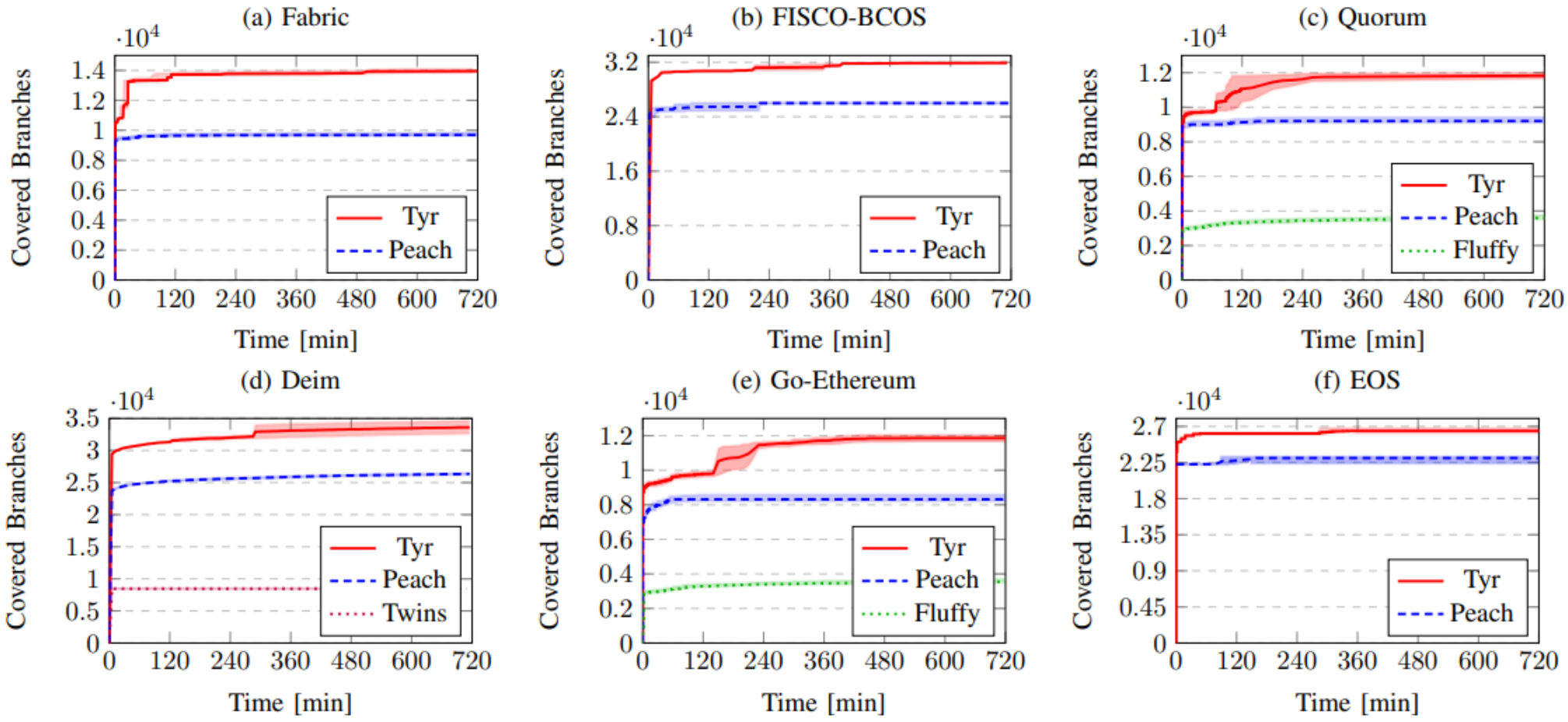
- heuristic insight
  - the ultimate consensus failure is the cumulative result of many transient inconsistencies in the consensus process.



# Implemataion



# Expriment

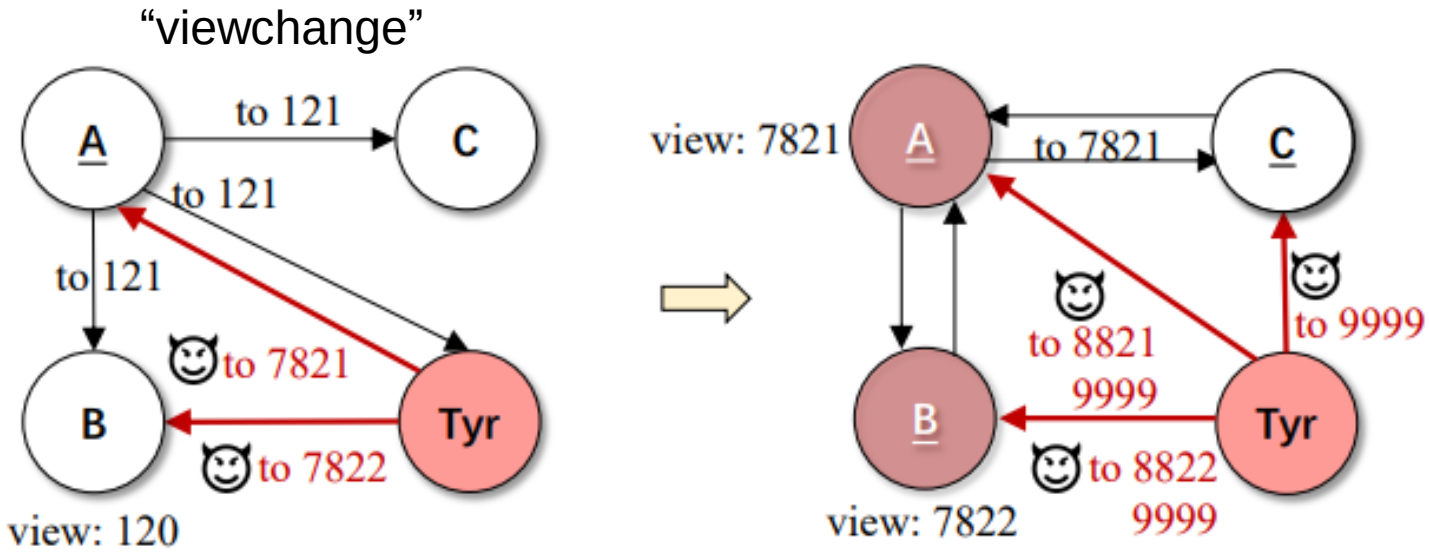

(a) Fabric
(b) FISCO-BCOS
(c) Quorum
(d) Deim
(e) Go-Ethereum
(f) EOS
Covered Branches
Time [min]
·10^4
Tyr
Peach
Fluffy
Twins

# Expriments

| # | Platform | Bug Type | Bug Description | Identifier |
|---|---|---|---|---|
| 1 | Fabric | Integrity | Missing Deletion of in-flight when syncing past the in-flight sequence. | CVE-2022-26297 |
| 2 | Fabric | Safety | Asynchronous sync procedures cause some proposals to be double processed. | CVE-2022-26298 |
| 3 | Fabric | Integrity | Repeat malicious consensus messages makes some honest nodes to be disconnected. | Bug#18167 |
| 4 | Fabric | Fairness | Various viewchange message sequences make some nodes always skip leader. | Bug#17950 |
| 5 | Fabric | Liveness | Random newView causes abnormal high-frequent viewchange and chaos in the network. | Bug#17875 |
| 6 | FISCO-BCOS | Liveness | The nodes change view frequently and stop generating blocks. | CVE-2022-26534 |
| 7 | FISCO-BCOS | Liveness | Transaction handling process is stuck after confusing nodes with different transaction headers. | Bug#2206 |
| 8 | FISCO-BCOS | Liveness | Multi-thread bugs cause some transactions cannot to be executed anymore. | Bug#2204 |
| 9 | FISCO-BCOS | Liveness | Some transactions cannot be processed correctly due to a deadlock. | Bug#2133 |
| 10 | FISCO-BCOS | Liveness | Lack of the verification of the packet header and the view-change is continuously triggered. | Bug#2448 |
| 11 | FISCO-BCOS | Safety | A malicious leader may fake a proposal's header and transactions cannot be processed . | Bug#2307 |
| 12 | FISCO-BCOS | Fairness | A malicious node can always be the leader, thus stop producing new blocks.. | CVE-2022-28937 |
| 13 | Quorum | Liveness | Transactions get stuck in a pending state after receiving incorrect gas from a malicious node. | Bug#1371 |
| 14 | Quorum | Integrity | Serial of malicious sync messages cause repeated "Full sync failed ", isolate normal node. | Bug#1107 |
| 15 | Diem | Fairness | Malicious nodes affect the QC commit and the leader's reputation and cause unfair leader selection. | Bug#10362 |
| 16 | Go-Ethereum | Integrity | Geth nodes no longer sync with Parity nodes after keep receiving malicious sync messages. | Bug#25243 |
| 17 | Go-Ethereum | Integrity | Keep rejecting blocks and stopping the block syncing procedure, leading to node isolation. | Bug#24448 |
| 18 | Go-Ethereum | Liveness | The client stopped transaction processing after receiving plenty of re-connection requests. | Bug#24832 |
| 19 | EOS | Liveness | The producer node crashes when generating a test account through the txn_test_gen_plugin. | CVE-2022-26300 |
| 20 | EOS | Integrity | Isolation occurs when multiple nodes produce blocks with the same index at the same time. | Bug#11063 |

# CVE-2022-26534

- Platform: FISCO-BCOS
- Implication: Dos to make node off line.




```
uint64_t greaterViewWeight = 0;
ViewType viewToReach = 0;
for (auto const& it : m_viewChangeCache) {
    ...
    // check the viewchange weight
    auto viewChangeCache = it.second;
    for (auto const& cache : viewChangeCache){
        auto fromIdx = cache.first;
        auto nodeInfo = m_config->
            getConsensusNodeByIndex(fromIdx);
        if (!nodeInfo){ continue;
// BUG: weight in cache should be clear to 0.
        greaterViewWeight += nodeInfo->weight();
    }
}
if (greaterViewWeight <
     (m_config->maxFaultyQuorum() +1)) return 0;
if (m_config->toView()>=viewToReach) return 0;
if (viewToReach > 0)
{  m_config->setToView(viewToReach - 1)};
```

# Conclusion

- These paper proposes Tyr, an automatic testing tool for detecting Consensus bug in blockchain systems based on the behavior divergent model.
- Tyr designs four properties to analyzing the violation of consensus.
- Fabric, FISCO-BCOS, Quorum, Diem, Go-Ethereum, and EOS.
  - Tyr is compatible with six platforms, has higher code coverage, and finds 20 consensus bugs (5 CVEs).