Towards Mechanised Probabilistic Blockchains

Kiran Gopinathan, Ilya Sergey

UCL, NUS

Blockchains & Security

- Blockchain systems have become commonplace.
- Hundreds of public Blockchain systems deployed to date.
- History of bugs and exploits:
  - 92 billion BTC underflow in 2010.
  - 5 successful 51% attacks in 2018.

Prior Work

- **Formalisations**
  - Bitcoin Backbone Protocol
  - Blockchain in Asynchronous networks
- **Mechanisations**
  - Toychain

Mechanising Blockchain Consensus

George Pîrlea
University College London, UK
george.pirlea.15@ucl.ac.uk

Ilya Sergey
University College London, UK
i.sergey@ucl.ac.uk

Abstract

We present the first formalisation of a blockchain-based distributed consensus protocol with a proof of its consistency mechanised in an interactive proof assistant.

Our development includes a reference mechanisation of the *block forest* data structure, necessary for implementing provably correct per-node protocol logic. We also define a model of a network, implementing the protocol in the form of a replicated state-transition system. The protocol's executions are modeled via a small-step operational semantics for asynchronous message passing, in which packages can be rearranged or duplicated.

In this work, we focus on the notion of global system safety, proving a form of eventual consistency. To do so, we provide a library of theorems about a pure functional implementation of block forests, define an inductive system invariant, and show that, in a quiescent system state, it implies a global agreement on the state of per-node transaction ledgers. Our development is parametric *wrt.* implementations of several security primitives, such as *hash*-functions, a notion of a *proof object*, a *Validator Acceptance Function*, and a *Fork Choice Rule*. We precisely characterise the assumptions, made about these components for proving the global system consensus, and discuss their adequacy. All results described in this paper are formalised in Coq.

CCS Concepts • **Theory of computation → Program verification**; • **Networks → Formal specifications**;

Keywords blockchain, consensus, protocol verification, Coq

ACM Reference Format:
George Pîrlea and Ilya Sergey. 2018. Mechanising Blockchain Con-

1 Introduction

The notion of decentralised blockchain-based consensus is a tremendous success of the modern science of distributed computing, made possible by the use of basic cryptography, and enabling many applications, including but not limited to cryptocurrencies, smart contracts, application-specific arbitration, voting, *etc.*

In a nutshell, the idea of a distributed consensus protocol based on *blockchains*, or *transaction ledgers*,[1] is rather simple. In all such protocols, a number of stateful nodes (participants) are communicating with each other in an asynchronous message-passing style. In a message, a node (a) can announce a *transaction*, which typically represents a certain event in the system, depending on the previous state of the node or the entire network (we intentionally leave out the details of what can go into a transaction, as they are application-specific); a node can also (b) create and broadcast a *block* that contains the encoding of a certain vector of transactions, created locally or received via messages of type (a) from other nodes. Each recipient of a block message should then *validate* the block (*i.e.*, check the consistency of the transaction sequence included in it), and, in some cases, append it to its local ledger, thus, extending its subjective view of the global sequence of transactions that have taken place in the system to date. The process continues as more messages are emitted and received.

In order to control the number of blocks in the system, distributed ledger protocols rely on certain cryptographic primitives, such as a hash-function *hash* defined both on transactions and blocks, a notion of a *proof object* necessary for defining the validity of a block, and an implementation of a *Validator Acceptance Function* (VAF) that is used to ensure

Blockchain Protocol

set of transactions $\{tx_1, tx_2, tx_3, tx_4, tx_5\}$

Blockchain Protocol

set of transactions

$$\{tx_1, tx_2, tx_3, tx_4, tx_5\}$$

$$\Big\downarrow \text{ blockchain protocol}$$

global ordering

$$tx_1 \rightarrow tx_2 \rightarrow tx_3 \rightarrow tx_4 \rightarrow tx_5$$

Blockchain Protocol

set of transactions $\{tx_1, tx_2, tx_3, tx_4, tx_5\}$

$$\downarrow$$

$$[] \rightarrow [tx_1] \rightarrow [tx_2, tx_3] \rightarrow [tx_4, tx_5]$$

$$\downarrow$$

global ordering $tx_1 \rightarrow tx_2 \rightarrow tx_3 \rightarrow tx_4 \rightarrow tx_5$

Blockchain Protocol

set of transactions

$$\{tx_1, tx_2, tx_3, tx_4, tx_5\}$$

$$\downarrow$$

$$[] \leftarrow [tx_1] \leftarrow [tx_2, tx_3] \leftarrow [tx_4, tx_5]$$

$$\downarrow$$

global ordering

$$tx_1 \rightarrow tx_2 \rightarrow tx_3 \rightarrow tx_4 \rightarrow tx_5$$

Probabilistic Difficulty

Global Consensus

Node	0
Hash	———-
Data	
POW	29

Node	1
Hash	010110
Data	tx_1
POW	10

Node	2
Hash	111010
Data	tx_2 tx_3
POW	421

Node	3
Hash	100111
Data	tx_4 tx_5
POW	772

Probabilistic Difficulty

Global Consensus

Node	0
Hash	———-
Data	
POW	29

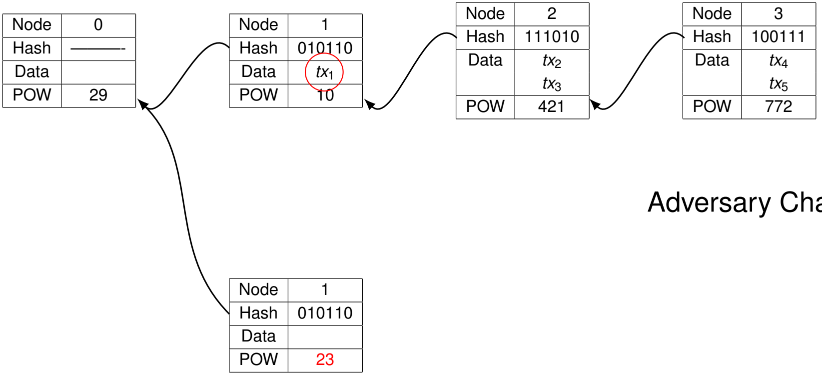
Node	1
Hash	010110
Data	tx_1
POW	10

Node	2
Hash	111010
Data	tx_2
	tx_3
POW	421

Node	3
Hash	100111
Data	tx_4
	tx_5
POW	772

Probabilistic Difficulty

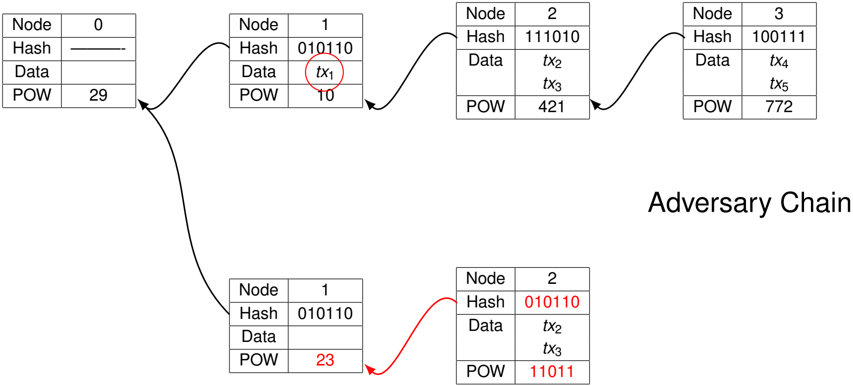
Global Consensus



Adversary Chain

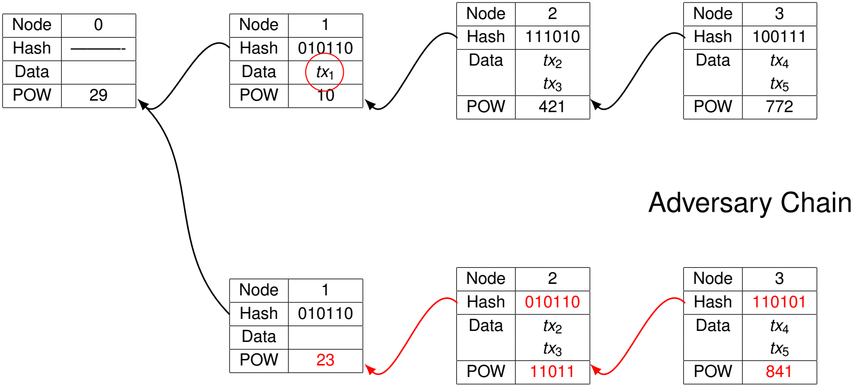
Probabilistic Difficulty

Global Consensus



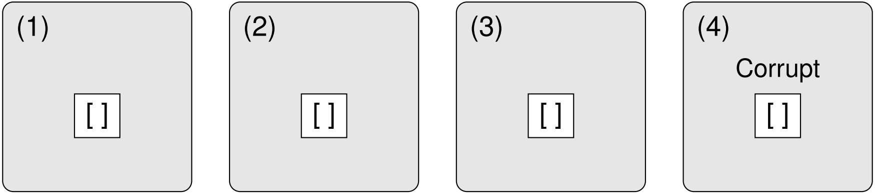
Probabilistic Difficulty

Global Consensus

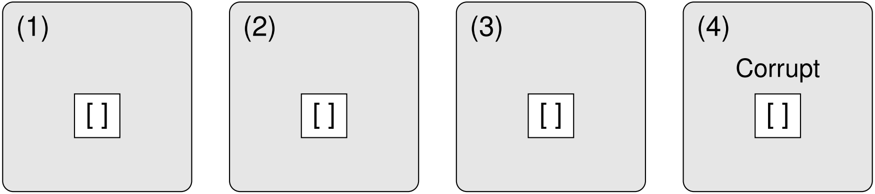


Network Model

Network Model



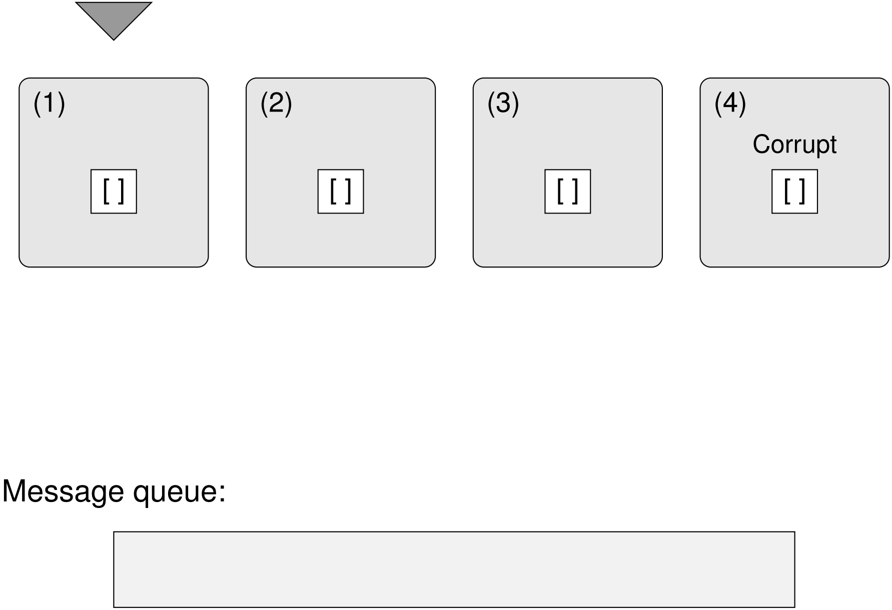
Network Model



Message queue:

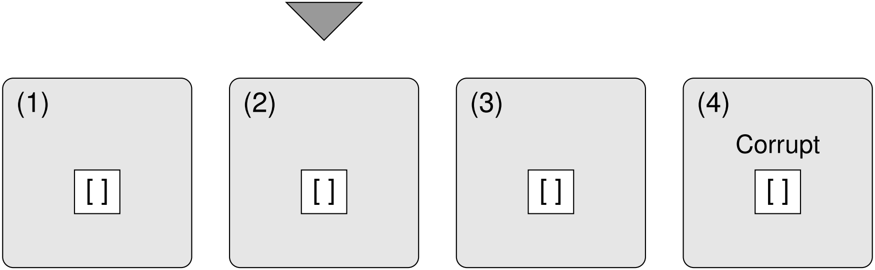
Network Model

Round 1



Network Model

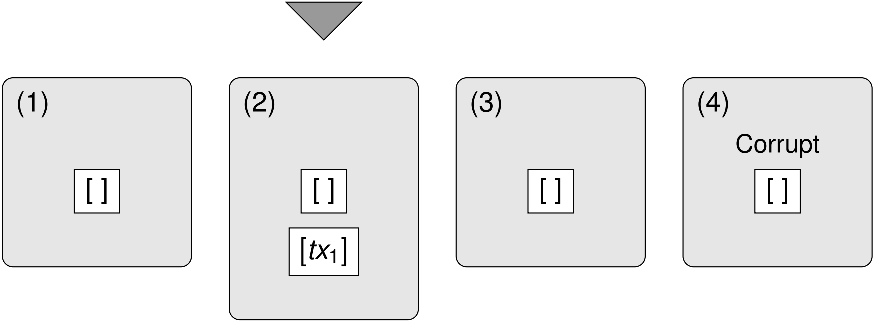
Round 1



Message queue:

Network Model

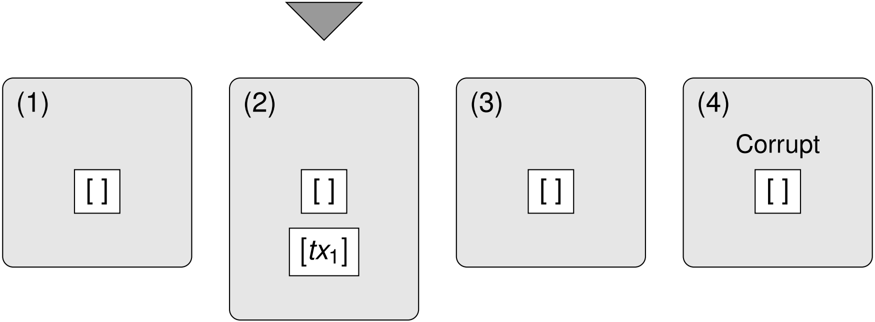
Round 1



Message queue:

Network Model

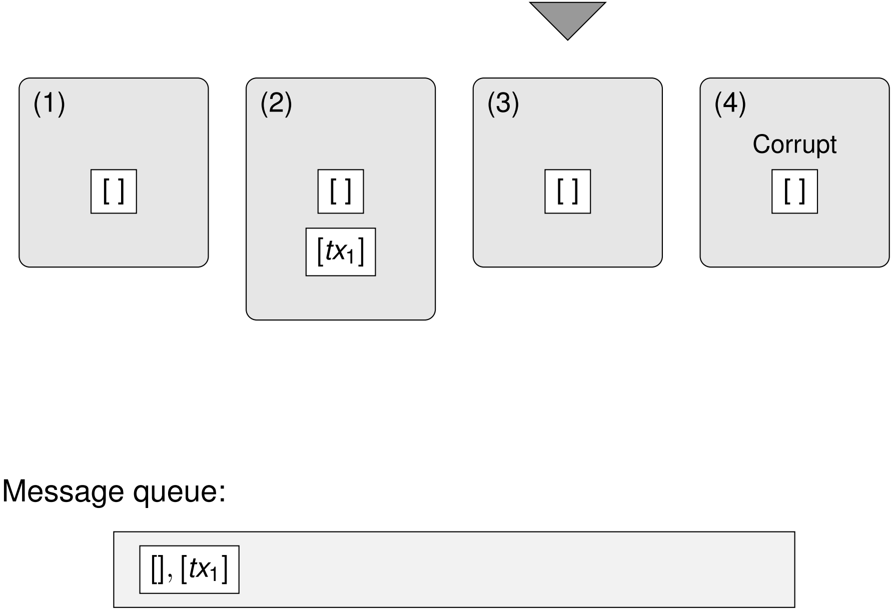
Round 1



Message queue:

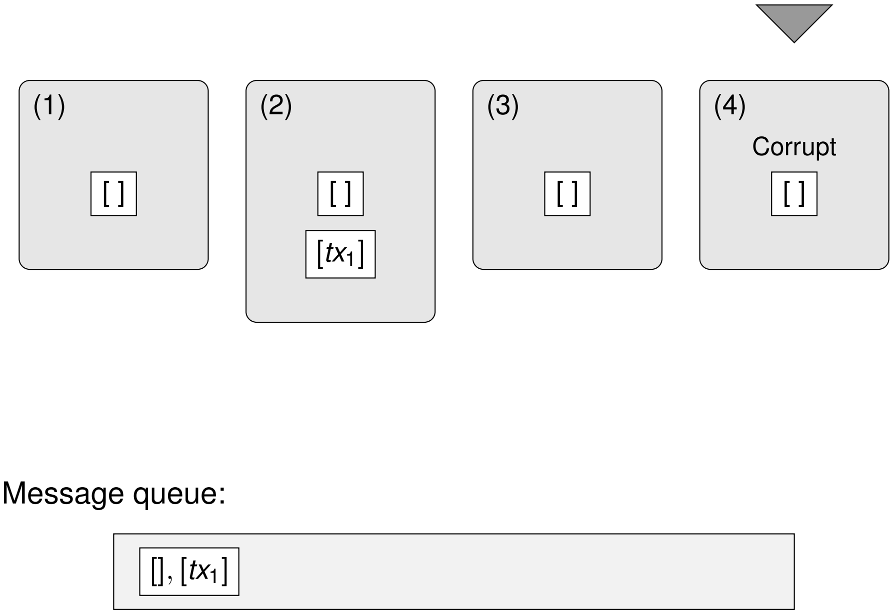
Network Model

Round 1



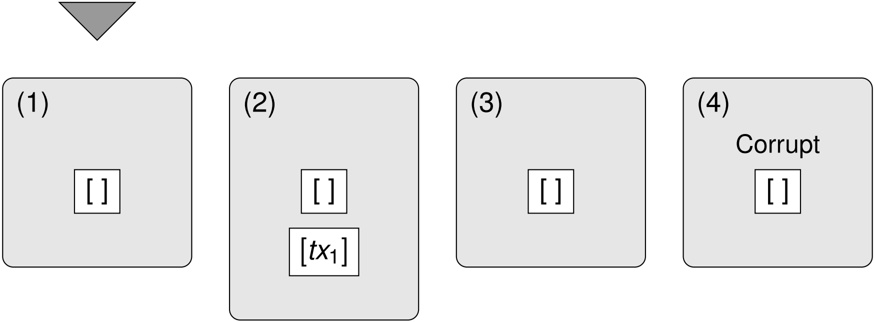
Network Model

Round 1



Network Model

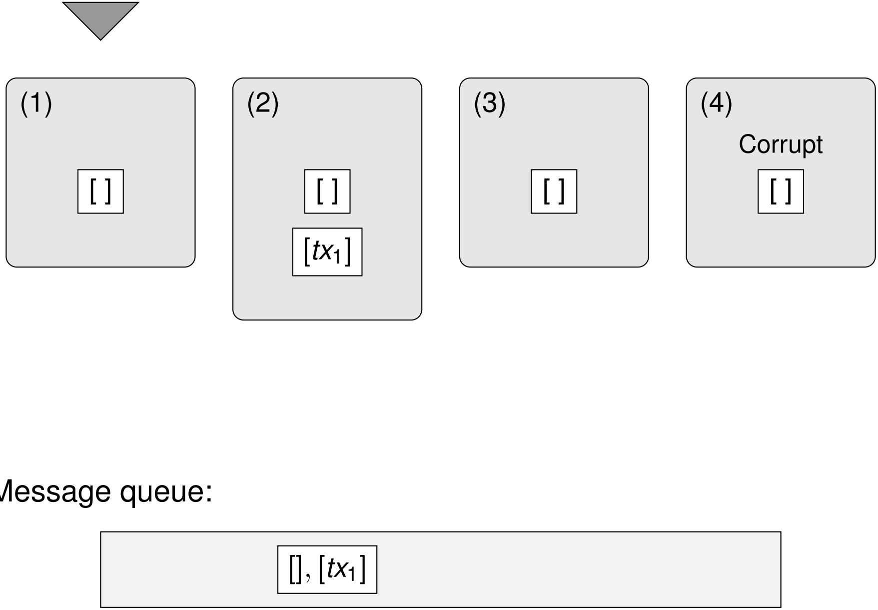
Round 2



Message queue:

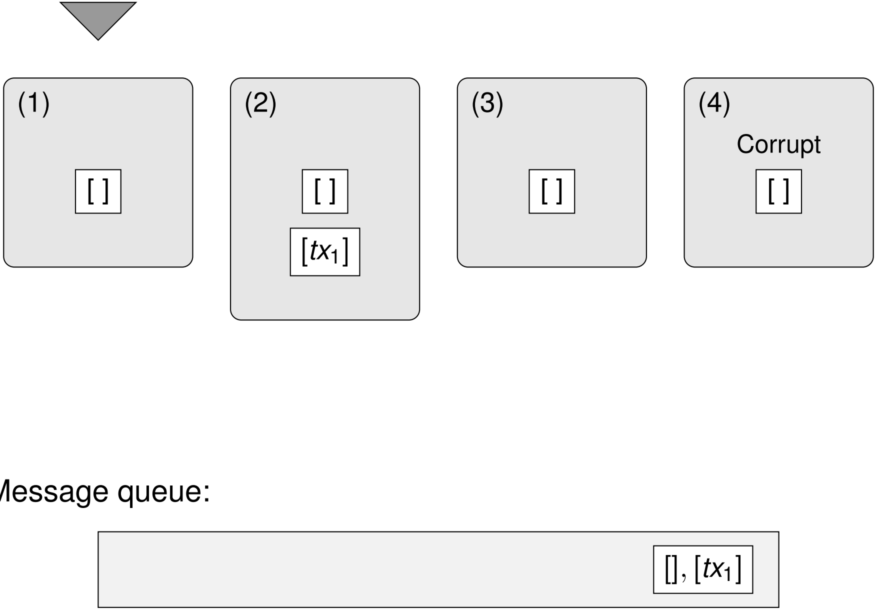
Network Model

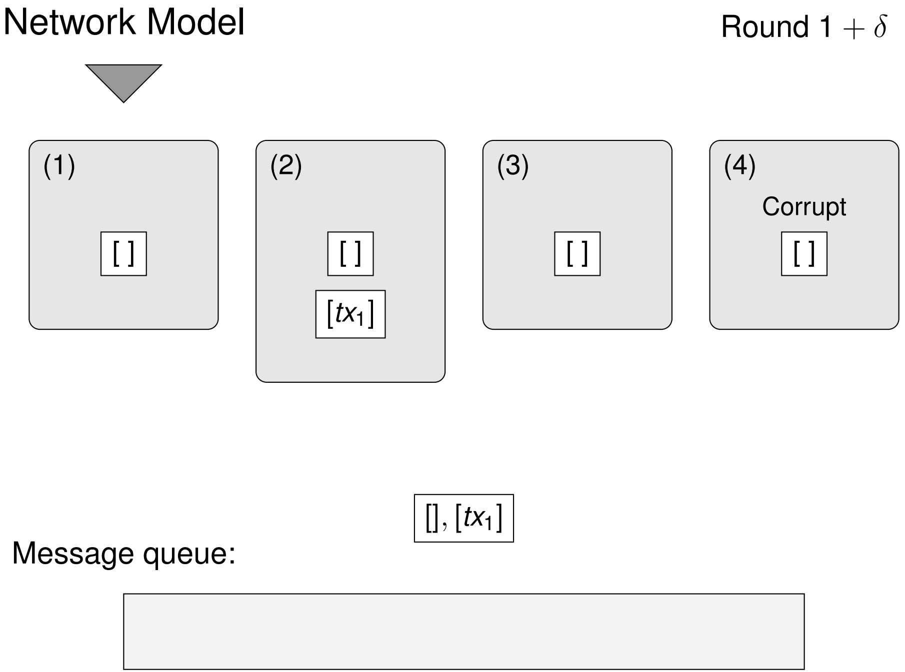
Round 2

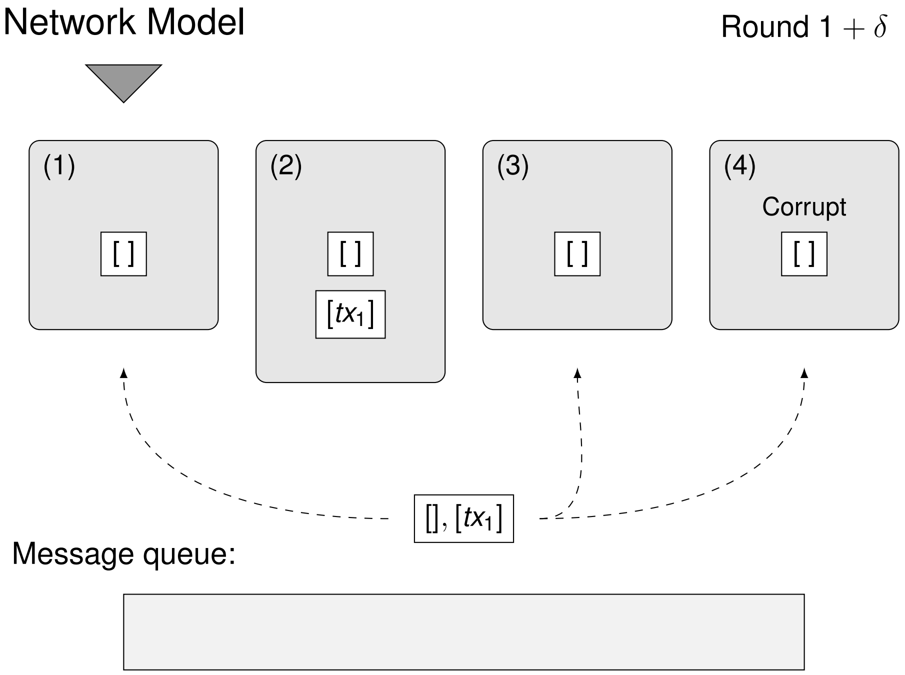


Network Model

Round $1 + \delta$

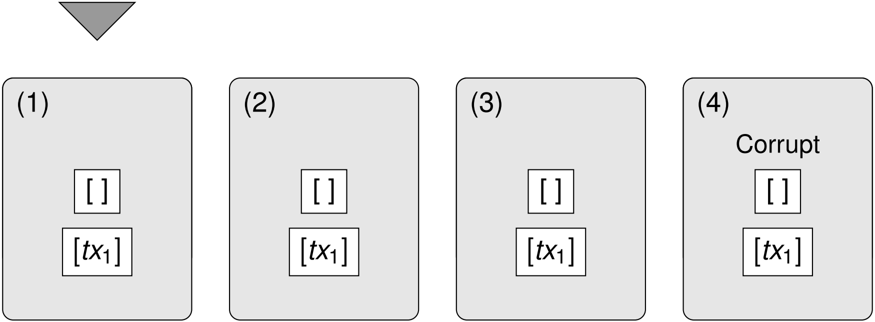



Network Model
Round $1 + \delta$
(1)
[ ]
(2)
[ ]
$[tx_1]$
(3)
[ ]
(4)
Corrupt
[ ]
$[], [tx_1]$
Message queue:


Network Model
Round $1 + \delta$
(1)
[ ]
(2)
[ ]
$[tx_1]$
(3)
[ ]
(4)
Corrupt
[ ]
$[], [tx_1]$
Message queue:

Network Model

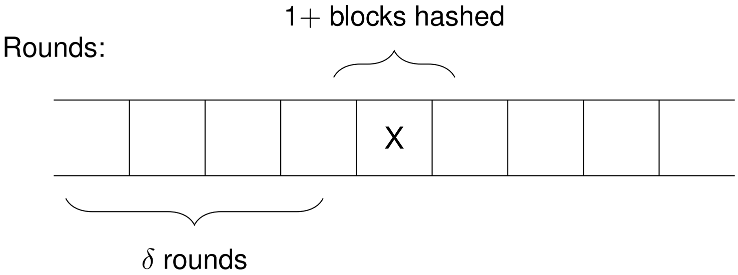
Round $1 + \delta$



Message queue:

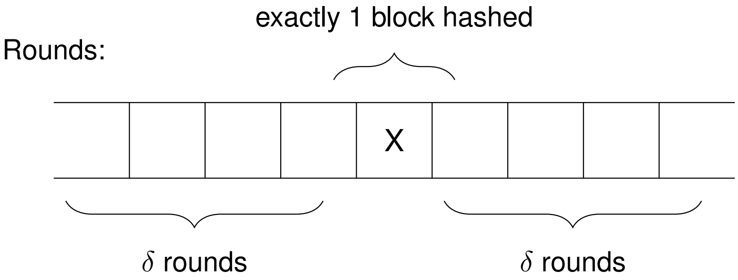
Typical Execution Property

- Bounded Successful Rounds - X'
- Uniquely Bounded Successful Rounds - Y'
- Number of Adversarial Blocks - Z'



Typical Execution Property

- Bounded Successful Rounds - X'
- Uniquely Bounded Successful Rounds - Y'
- Number of Adversarial Blocks - Z'



Typical Execution Property

- Bounded Successful Rounds - X'
- Uniquely Bounded Successful Rounds - Y'
- Number of Adversarial Blocks - Z'

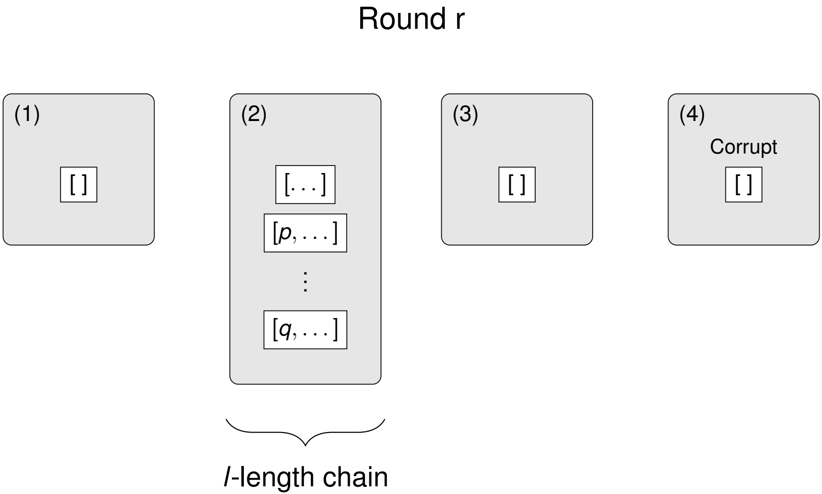
Rounds:

Z'_0	Z'_1	Z'_2	Z'_3	Z'_4	Z'_5	Z'_6	Z'_7	Z'_8

where

$Z'_i \sim$ # blocks hashed by adversary in round i

Chain Growth Property



Chain Growth Property

Round $s \geq r + \delta$

(1)

$[\ldots]$

$[a, \ldots]$

$\vdots$

$[b, \ldots]$

(2)

$[\ldots]$

$[p, \ldots]$

$\vdots$

$[q, \ldots]$

(3)

$[\ldots]$

$[x, \ldots]$

$\vdots$

$[y, \ldots]$

(4)

Corrupt

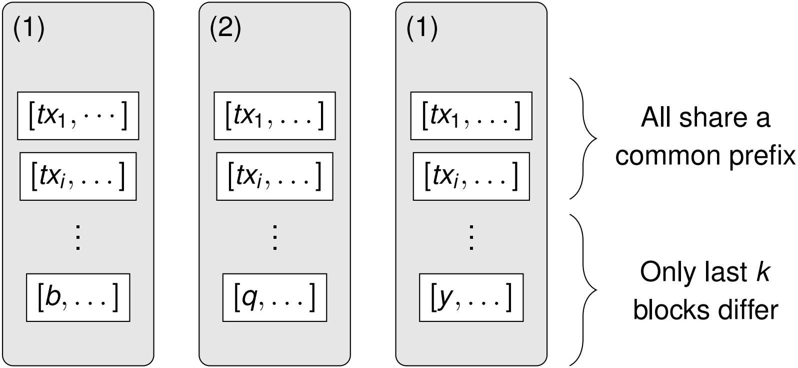
$[\,]$

all have chain lengths

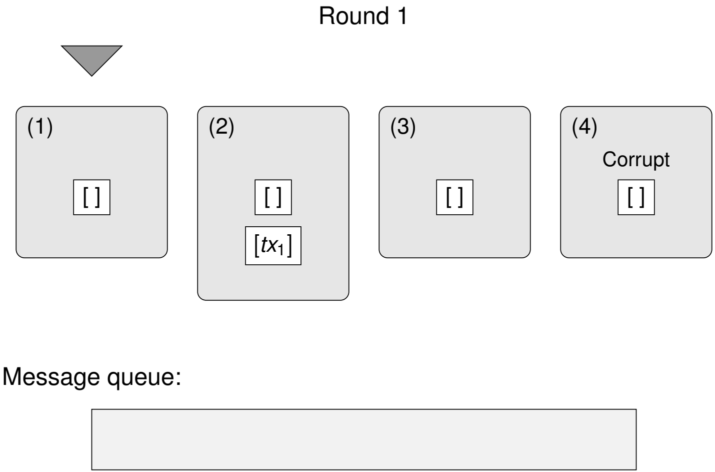
$$l' \geq l + \sum_{i=r}^{s-\delta} X'_i$$

Common Prefix Property

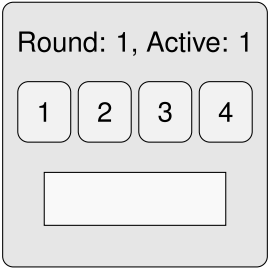
At all rounds,



Mechanical Semantics

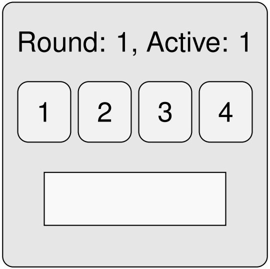


Mechanical Semantics

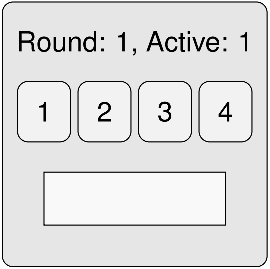


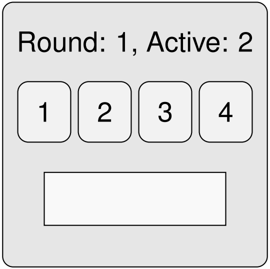
World

Mechanical Semantics

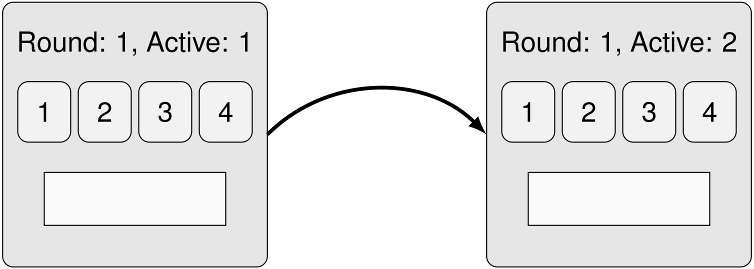


Mechanical Semantics



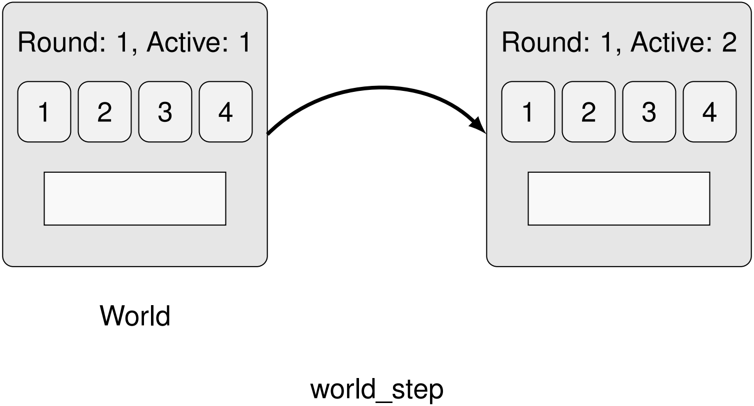


Mechanical Semantics

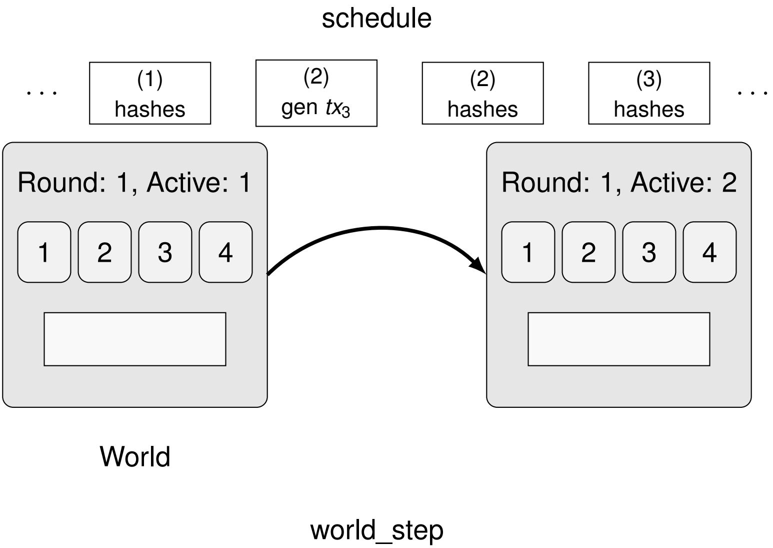


world_step

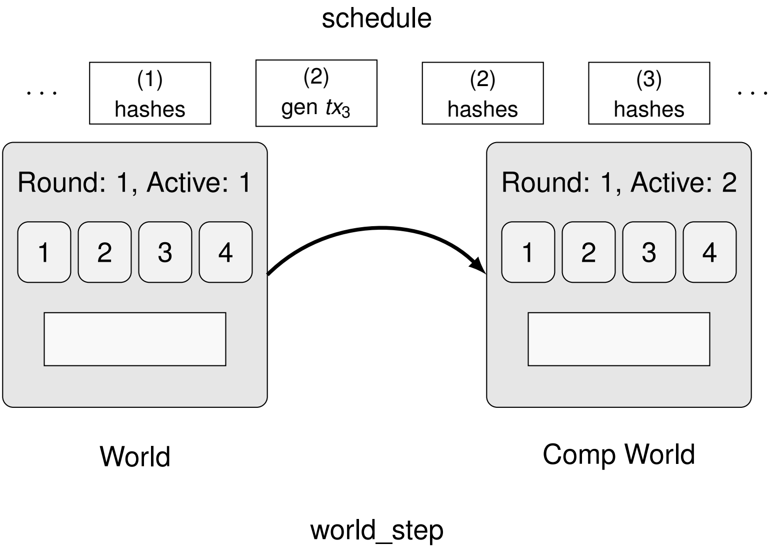
Mechanical Semantics



Mechanical Semantics



Mechanical Semantics



Encoding Probability

hash(x) = ???

Encoding Probability

$$\text{hash}(x) = \begin{array}{c} \vdots \\ \left\{ \begin{array}{c} 18 \\ 17 \\ 16 \\ 15 \\ 14 \\ 13 \\ 12 \end{array} \right. \\ \vdots \end{array}$$

Encoding Probability

$$\text{hash(x)} = \begin{cases} \vdots \\ 18 \\ 17 \\ 16 \\ 15 & : P[\,hash(x) = 15\,] \\ 14 \\ 13 \\ 12 \\ \vdots \end{cases}$$

Encoding Probability

$$\text{hash} : A \rightarrow (B \rightarrow \mathbb{R})$$

Encoding Probability

$$\text{hash} : A \to \text{dist } B$$

Encoding Probability

- Probability monad defined by Affeldt and Hagiwara.

$$\begin{aligned}\text{bind} &: \text{dist } A \rightarrow (A \rightarrow \text{dist } B) \rightarrow \text{dist } B\\ \text{ret} &: A \rightarrow \text{dist } A\end{aligned}$$

Encoding Probability

- We extend it to probabilistically execute the system.

$$\text{eval\_dist} : \text{Comp } A \rightarrow \text{ dist } A$$

Encoding Probability

- To allow stating properties about probable worlds.

$$\forall sc, \forall w, \text{eval\_dist (world\_step } w_0 \ sc) \ w > 0 \implies F \ w$$

$$\forall sc, \text{P[(world\_step } w_0 \ sc) \ \triangleright \ F \ ] = 1$$

Key lemmas

- **Typical Execution Assumption**

$$\mathrm{P}[\,\text{world\_step}\ sc \triangleright \mathrm{TEP}_{\varepsilon}\ sc\,] = 1 - e^{-\Omega(\kappa)}$$

- **Chain Growth Property**

$$\mathrm{P}[\,\text{world\_step}\ sc\ w_0 \triangleright \mathrm{CGP}\,] = 1$$

- **Common Prefix Property**

$$\mathrm{P}[\,\text{world\_step}\ sc\ w_0 \triangleright (\mathrm{CPP}_k \mathbin{\dot\wedge} \mathrm{TEP}_{\varepsilon}\ sc)\,] = \mathrm{P}[\,\text{world\_step}\ sc\ w_0 \triangleright \mathrm{TEP}_{\varepsilon}\ sc\,]$$

Key lemmas

- **Typical Execution Assumption**

$$\mathrm{P}[\ \text{world\_step}\ sc \rhd \mathrm{TEP}_{\varepsilon}\ sc\ ] = 1 - e^{-\Omega(\kappa)}$$

- **Chain Growth Property**

$$\mathrm{P}[\ \text{world\_step}\ sc\ w_0 \rhd \mathrm{CGP}\ ] = 1$$

- **Common Prefix Property**

$$\mathrm{P}[\ \text{world\_step}\ sc\ w_0 \rhd (\mathrm{CPP}_k \mathbin{\dot\wedge} \mathrm{TEP}_{\varepsilon}\ sc)\ ] = \mathrm{P}[\ \text{world\_step}\ sc\ w_0 \rhd \mathrm{TEP}_{\varepsilon}\ sc\ ]$$

Key lemmas

- **Typical Execution Assumption**

$$\mathrm{P}[\ \text{world\_step}\ sc \rhd \mathrm{TEP}_{\varepsilon}\ sc\ ] = 1 - e^{-\Omega(\kappa)}$$

- **Chain Growth Property**

$$\mathrm{P}[\ \text{world\_step}\ sc\ w_0 \rhd \mathrm{CGP}\ ] = 1$$

- **Common Prefix Property**

$$\mathrm{P}[\ \text{world\_step}\ sc\ w_0 \rhd (\mathrm{CPP}_k \mathbin{\dot\wedge} \mathrm{TEP}_{\varepsilon}\ sc)\ ] = \mathrm{P}[\ \text{world\_step}\ sc\ w_0 \rhd \mathrm{TEP}_{\varepsilon}\ sc\ ]$$

Key lemmas

- **Typical Execution Assumption**

$$\mathrm{P}[\ \mathrm{world\_step}\ sc \triangleright \mathrm{TEP}_{\varepsilon}\ sc\ ] = 1 - e^{-\Omega(\kappa)}$$

- **Chain Growth Property**

$$\mathrm{P}[\ \mathrm{world\_step}\ sc\ w_0 \triangleright \mathrm{CGP}\ ] = 1$$

- **Common Prefix Property**

$$\mathrm{P}[\ \mathrm{world\_step}\ sc\ w_0 \triangleright (\mathrm{CPP}_k \mathbin{\dot\wedge} \mathrm{TEP}_{\varepsilon}\ sc)\ ] = \mathrm{P}[\ \mathrm{world\_step}\ sc\ w_0 \triangleright \mathrm{TEP}_{\varepsilon}\ sc\ ]$$

Main Contributions

- Implemented a mechanised probabilstic blockchain model based on the Bitcoin Backbone Protocol (BBP) by Garay et al.
- Proved several preliminary lemmas.
- Formulated the main BBP lemmas within this model.

Future work

- Completing proofs of the key properties.
- Elevating the Typical Execution Assumption to a lemma.
- Extracting the system to an executable implementation.

Take away

- Blockchain security properties inherently require probabilistic considerations.
- 2 key properties:
  - **Chain growth property**
  - **Common prefix property**
- Working on a mechanisation of the Bitcoin backbone protocol.
  - Mechanised protocol model.
  - Formulated several key lemmas.

https://github.com/certichain/probchain