University of BRISTOL

M0019 Lecture 17: Blockchain & Cryptocurrencies, part 1

Seth Bullock & Dave Cliff

Outline

Today:

- Bitcoin
- Blockchain
- Recent Developments
- A wider context

Outline

Today:

- Bitcoin
- Blockchain
- ~~Recent Developments~~
- ~~A wider context~~

University of BRISTOL

Bitcoin in the News…

TECH & SCIENCE

The End of Dumb Money

BY KEVIN MANEY 1/15/14 AT 12:40 PM

Newsweek

BUSINESS DAY

Apparent Theft at Mt. Gox Shakes Bitcoin World

By NATHANIEL POPPER and RACHEL ABRAMS FEB. 25, 2014

THE TIMES

Technology

News | Opinion | Business | Money | Sport | Life | Arts | Puzzles | Papers | Irish new

Welcome to your preview of The Times

Bitcoin and other virtual currencies coul
be real threat to economy, says Ban

Nov 13, 2015 COMMENT

Bitcoin — and a scam easy to spot

The trouble with the currency is that it is accountable to no one, writes Izabella Kaminska

FT FINANCIAL TIMES

Most Recently…

The R3 Consortium explores Blockchain for banks…

University of BRISTOL

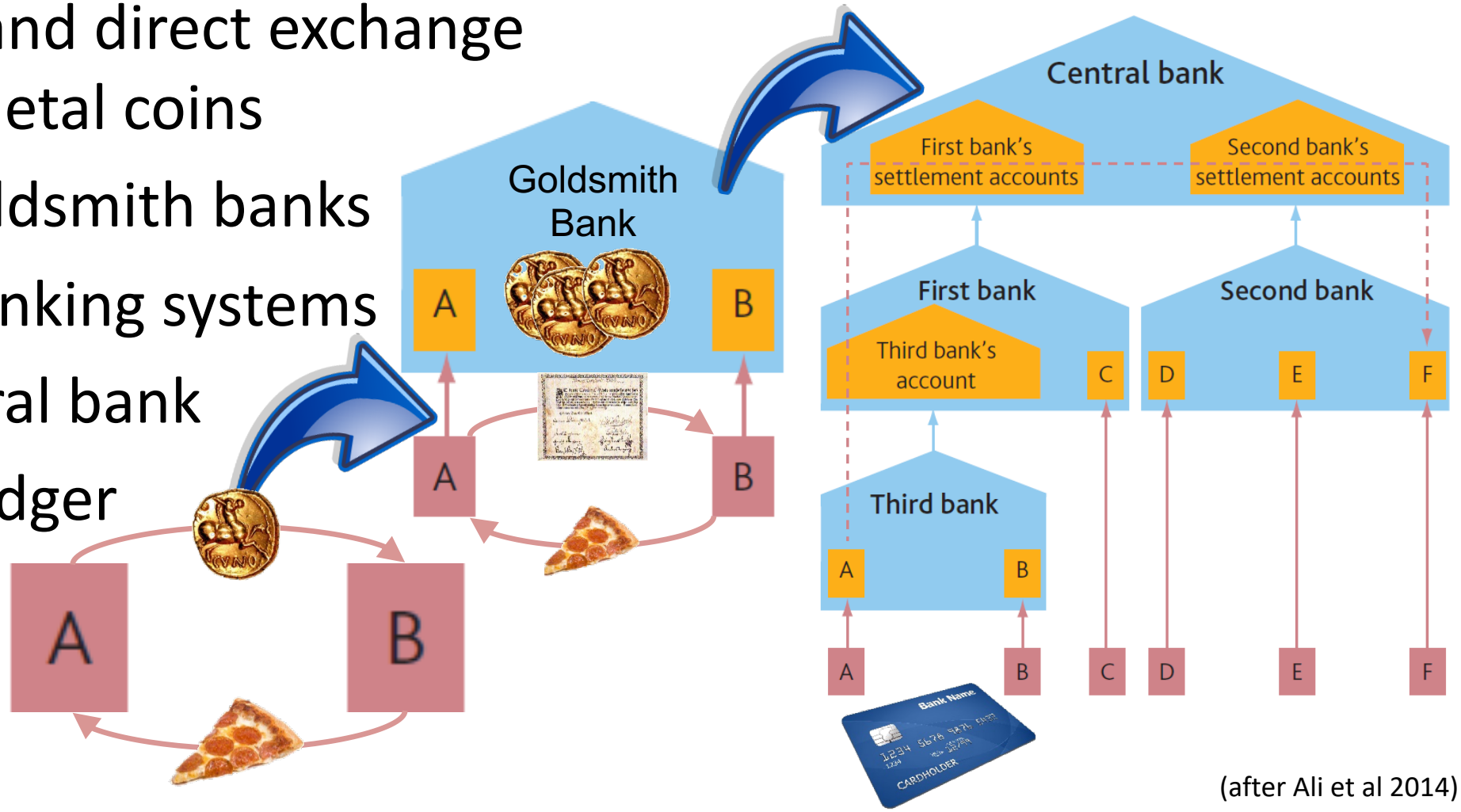
https://www.etoro.com/markets/btc/chart

COMS

l.ac.uk

The Evolution of Payment

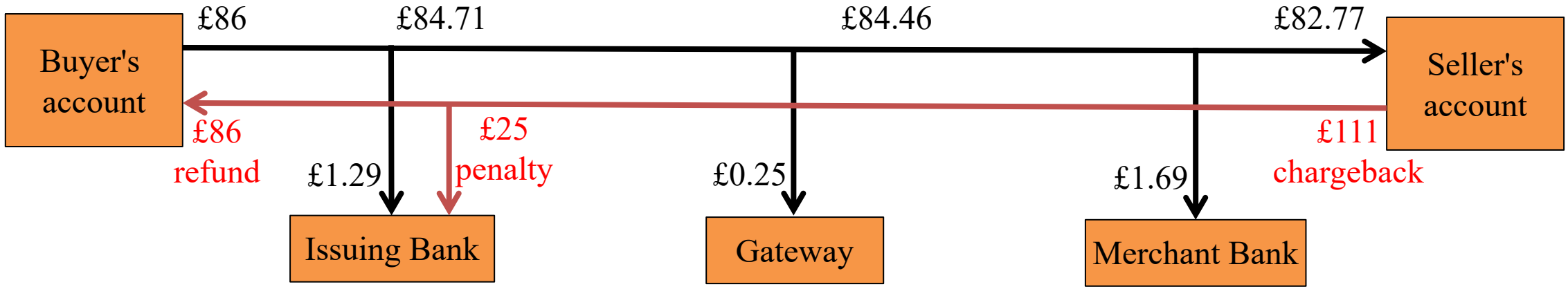
- From barter and direct exchange of precious metal coins
- …to single goldsmith banks
- …to tiered banking systems
- …with a central bank
- & a central ledger



(after Ali et al 2014)

Flow of Money

- Intermediaries between buyer and seller mean transaction costs…



- …and the risk that intermediaries take is offset by imposing penalties on the seller when transactions go bad: "chargeback"
  - E.g., technical (insufficient funds…), clerical (fat finger…), quality (lousy service ,merchandise undelivered…), fraud (stolen card…), etc.

Exposure

- Centralised banks also expose customers to risk:

- … and to scrutiny:

Silk Road

anonymous marketplace

Transactions

- The easiest way to understand Bitcoin (and Blockchain) is to walk through the steps involved in a Bitcoin transaction.
- Imagine that Amelia earned some bitcoin by doing some coding
- She wants to use some of it to buy a car from Bea
- The following steps take place as they make their transaction

Bitcoin Transaction Steps

1. Agree The Transaction
2. Create the Transaction Message
3. Sign the Transaction Message
4. Broadcast the Transaction Message
5. Verify the Transaction Message
6. Success!

Bitcoin Transaction Steps

1. Agree The Transaction
2. Create the Transaction Message
3. Sign the Transaction Message
4. Broadcast the Transaction Message
5. Verify the Transaction Message
6. Success!

University of BRISTOL

Step 1: Agree the Transaction

- Amelia and Bea agree that the car is only worth about £1000.
- By current exchange rate (1Bc ≈ £5,555): £1000 ≈ 0.18 Bitcoin
- Amelia has earned 10 Bitcoin from coding so she can afford it
- Amelia knows that adding a small "fee" to her payment makes it more likely to be processed quickly…
- …because the fee will be "earned" by the miner that mines the block that Amelia's transaction is included in.
- So she's going to spend a fraction of a bitcoin on this fee.
- The fee is not necessary, but it is an incentive for bitcoin miners.

Bitcoin Transaction Steps

1. Agree The Transaction
2. Create the Transaction Message
3. Sign the Transaction Message
4. Broadcast the Transaction Message
5. Verify the Transaction Message
6. Success!

Step 2: Create the Transaction Message

- Amelia creates a *message* that includes three components:
  1. A *reference* to the earlier transaction that paid Amelia the 10 bitcoin that she is going to use to pay for the car (the *input* to the transaction).
  2. A *list of addresses* of recipients of the payment, i.e., Bea's address and also Amelia's address, since the transaction pays her the change back.
  3. A *list of amounts* to pay to the addresses (the transaction *outputs*).
- Amelia will *split* her 10 bitcoin input between two addresses: i) Bea, who gets paid for the car; ii) Amelia, who gets change back
- If amounts (i)+(ii) < 10, the remainder is the "fee" for a miner. But we don't know who will get it yet...

Bitcoin Transaction Steps

1. Agree The Transaction
2. Create the Transaction Message
3. Sign the Transaction Message
4. Broadcast the Transaction Message
5. Verify the Transaction Message
6. Success!

Step 3: Sign the Transaction Message

- Amelia “signs” the message by encrypting it using her private key.

- Everyone else can confirm that the message was made by Amelia, because they can use her public key to quickly decrypt it.
- But no-one can encrypt a message like Amelia without stealing her private key – so no-one else can spend Amelia’s bitcoin.
- The encryption doesn’t keep the message secret – it confirms who the message was authored (encrypted) by.
- Like a fancy royal seal – hard to make, but easy to recognise.

Bitcoin Transaction Steps

1. Agree The Transaction
2. Create the Transaction Message
3. Sign the Transaction Message
4. Broadcast the Transaction Message
5. Verify the Transaction Message
6. Success!

Step 4: Broadcast the Transaction Message

- Amelia then *broadcasts* the encrypted message along with her public key (which others can use for decrypting the message).
- Amelia initiates the broadcast by spreading the message to Amelia's immediate peers in the peer-2-peer network
- Before broadcasting it to their peers, they verify the message by checking that it meets many constraints:
  - inputs ≥ outputs; any transactions referenced exist in the Blockchain; etc.
- Amelia's transaction is now one of a number of transactions on the network that haven't yet been included in the Blockchain.

Bitcoin Transaction Steps

1. Agree The Transaction
2. Create the Transaction Message
3. Sign the Transaction Message
4. Broadcast the Transaction Message
5. Verify the Transaction Message
6. Success!

Step 5: Mining (Verifying the Transaction)

- Miners now compete to be the first to “mine” a “block” containing transactions that haven’t yet been added to the Blockchain.
- The winner will get to keep the “fee” from each transaction and also gets to include a payment to themselves in the block (12.5Btc).
- (These 12.5 bitcoin are new bitcoin currency in the economy.)
- This payment incentivises the work that the miners carry out.
- Mining is hard work – it takes many compute cycles to win the competition. This is important, because “proof of work” is what keeps the Blockchain consistent and reliable.

Important Interlude: Anatomy of a Block

- Each block in the Blockchain has a *header* formed of:
  - Timestamp and version number
  - A hash of the *previous* block header – this links new blocks into a chain!
  - A *merkle root* summarising the new transactions that the block contains – this means every block will contain a history of all bitcoin transactions!
  - A *nonce value* chosen by the miner such that when the header is hashed using $SHA256^2$ the resulting 256-bit hash has at least *x* leading 0s
- After the header comes a list of the transactions that are recorded by the block (and summarised in the block's header), including the transaction that pays the miner his or her fee.

University of BRISTOL

Important Interlude: Nonce Values & Difficulty...

- SHA256 hashes any data, *D*, into an arbitrary 256-bit string, *S*.
- (SHA256^2 just applies SHA256 twice.)
- If you wanted SHA256^2 to hash *D* into *S*=000…011, you could add an arbitrary "nonce" value to the end of *D* and keep changing this value until SHA256^2 gave you a bit string, *S*, that evaluated to 3.
- If you are only happy with one specific 256-bit string you will be waiting a long, long, long time before you find the right nonce.
- But if you just want SHA256^2 to give you a hash *S* such that $S<2^{200}$ things are easier. If you're happy with $S<2^{250}$ then it's even easier.

Important Interlude: …Are A Bit Like Sudoku

- Checking whether a nonce "works" is *very quick*
  - It requires one execution of $SHA256^2$ on *D* and a comparison operation: is *S* satisfactory. Easy.
  - This is like *checking* whether a Sudoku solution is correct – pretty easy even if the Sudoku is very large.
- But finding a nonce that works is *very slow*
  - Many executions of $SHA256^2$ depending on how small the subset of allowable hash strings is
  - This is like *solving* a Sudoku puzzle –very difficult esp. if the puzzle is large or initially quite empty.

4	5	7	3	8	1	2	6	9
1	6	2	5	4	9	8	7	3
9	3	8	2	7	6	4	5	1
3	7	4	8	6	2	1	9	5
8	2	5	9	1	7	3	4	6
6	1	9	4	3	5	7	2	8
2	4	1	6	5	8	9	3	7
5	8	3	7	9	4	6	1	2
7	9	6	1	2	3	5	8	4

	5				1		6	
1			5			8		3
				7			5	
	7		8	6				5
		5	9		7	3		
6				3	5		2	
	4			5				
5		3			4			2
	9		1				8	

Important Interlude: …But Also Different…

- Some things are true of Sudoku but not SHA256[2]
  - One can get better at Sudoku through practice
  - Some people can just solve puzzles faster than others
- But the only way to improve the chance of finding the right nonce value is to *check more possibilities*
  - There's no "trick". There's no "optimal search".
  - Brute force is the only "strategy"
- This means that miners can't know which of them will be successful – they just have to do the work, and a correct nonce is their Proof-of-Work (PoW)

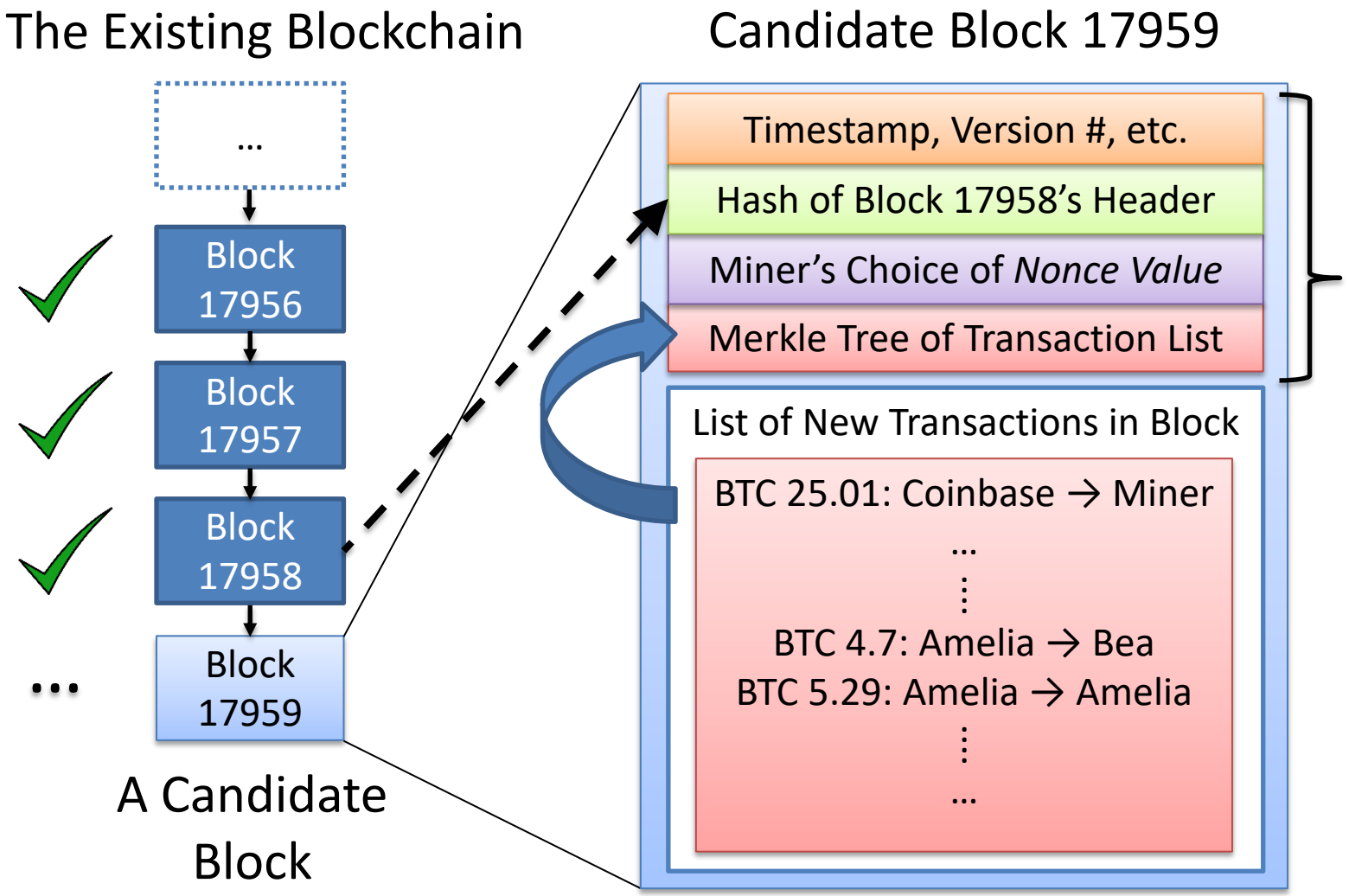
4	5	7	3	8	1	2	6	9
1	6	2	5	4	9	8	7	3
9	3	8	2	7	6	4	5	1
3	7	4	8	6	2	1	9	5
8	2	5	9	1	7	3	4	6
6	1	9	4	3	5	7	2	8
2	4	1	6	5	8	9	3	7
5	8	3	7	9	4	6	1	2
7	9	6	1	2	3	5	8	4

	5				1		6	
1			5			8		3
				7			5	
	7		8	6				5
		5	9		7	3		
6				3	5		2	
	4			5				
5		3			4			2
	9		1				8	

Bitcoin Transaction Steps

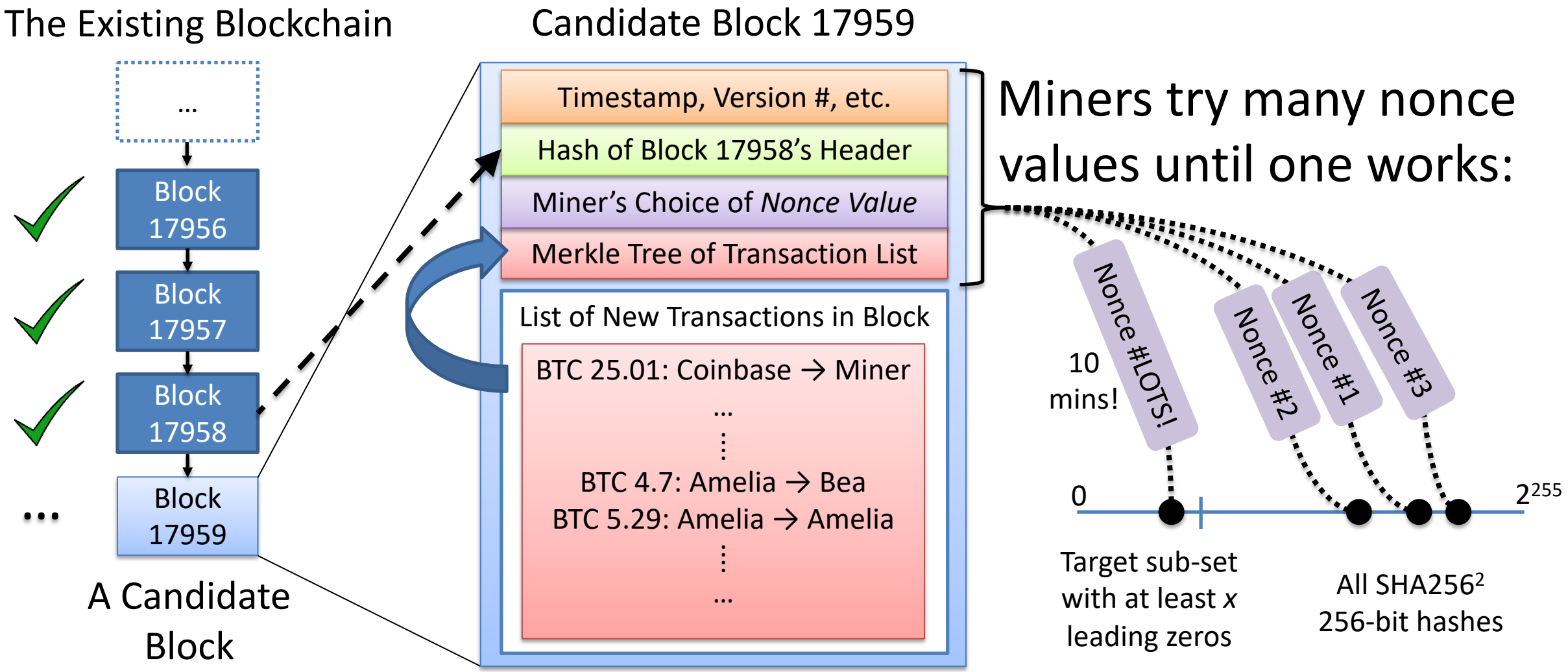
1. Agree The Transaction
2. Create the Transaction Message
3. Sign the Transaction Message
4. Broadcast the Transaction Message
5. Verify the Transaction Message
6. Success!

Step 5: Mining (Verifying the Transaction)



- The Block's Header
- The miner hashes the new header using $SHA256^2$
- Does the resulting hash start with at least *x* zeros?

Step 5: Mining (Verifying the Transaction)



Bitcoin Transaction Steps

1. Agree The Transaction
2. Create the Transaction Message
3. Sign the Transaction Message
4. Broadcast the Transaction Message
5. Verify the Transaction Message
6. Success!

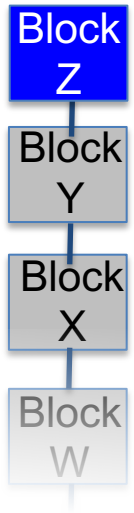
Step 6: Success!

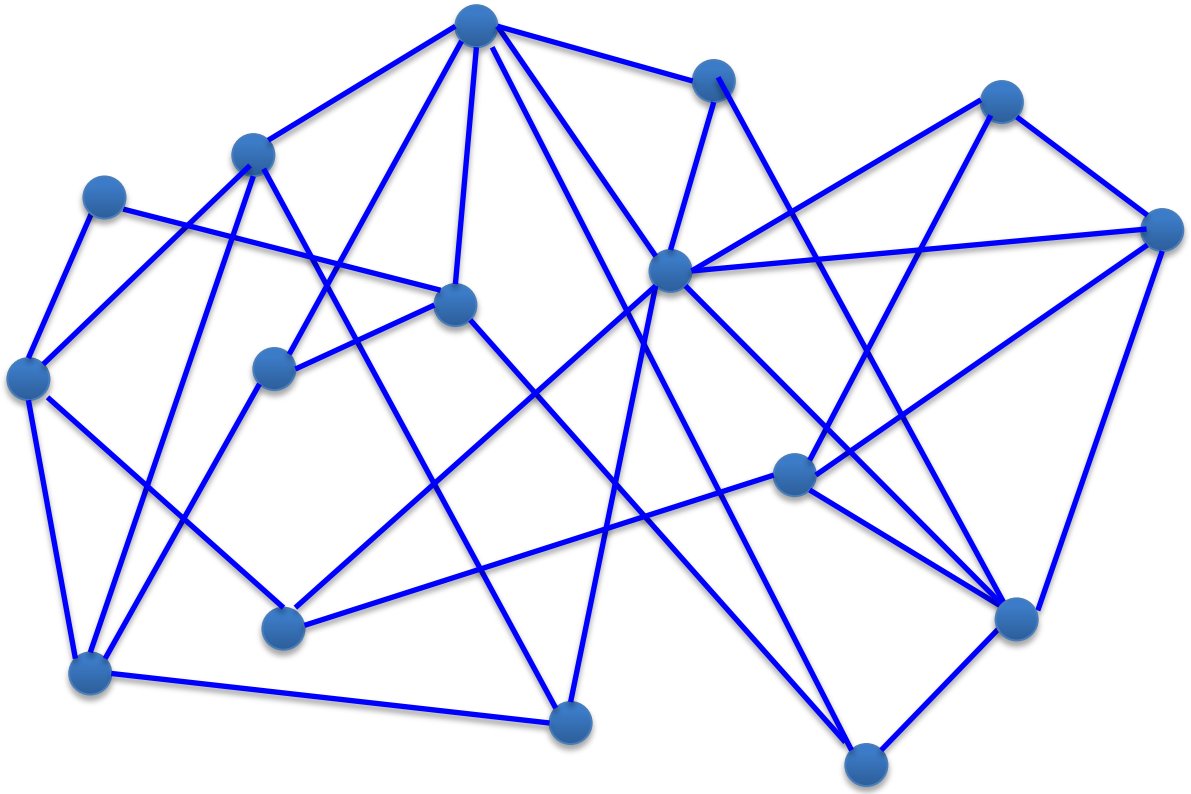
- After 11mins miner *M* mines a block with Amelia's transaction in it
- *M* gets 12.5 new bitcoins and also the transaction fees in the block
- *M* broadcasts the new block to the peer-to-peer network.
- Nodes all add the new block to their copies of the Blockchain...
- …and miners start to mine the next new block of transactions.
- Bea can check that the transaction is verified in the Blockchain...
- …and hand Amelia the keys to her car
- Bea can now spend her new bitcoin…

Double Spend Problem

- A digitised song can be shared multiple times by making copies…
- What stops someone spending their digital bitcoin twice?
  - The Blockchain stores a single agreed history of every transaction
  - Nodes check to see if a bitcoin has already been spent
  - If so, the transaction isn't included in the block being mined
  - If a coin is spent twice simultaneously on different parts of the P2P network, a "fork" in the Blockchain can occur temporarily...
  - …but Blockchain will identify and correct such a fork quickly and without recourse to centralised control or co-ordination

Blockchain Handling A Fork

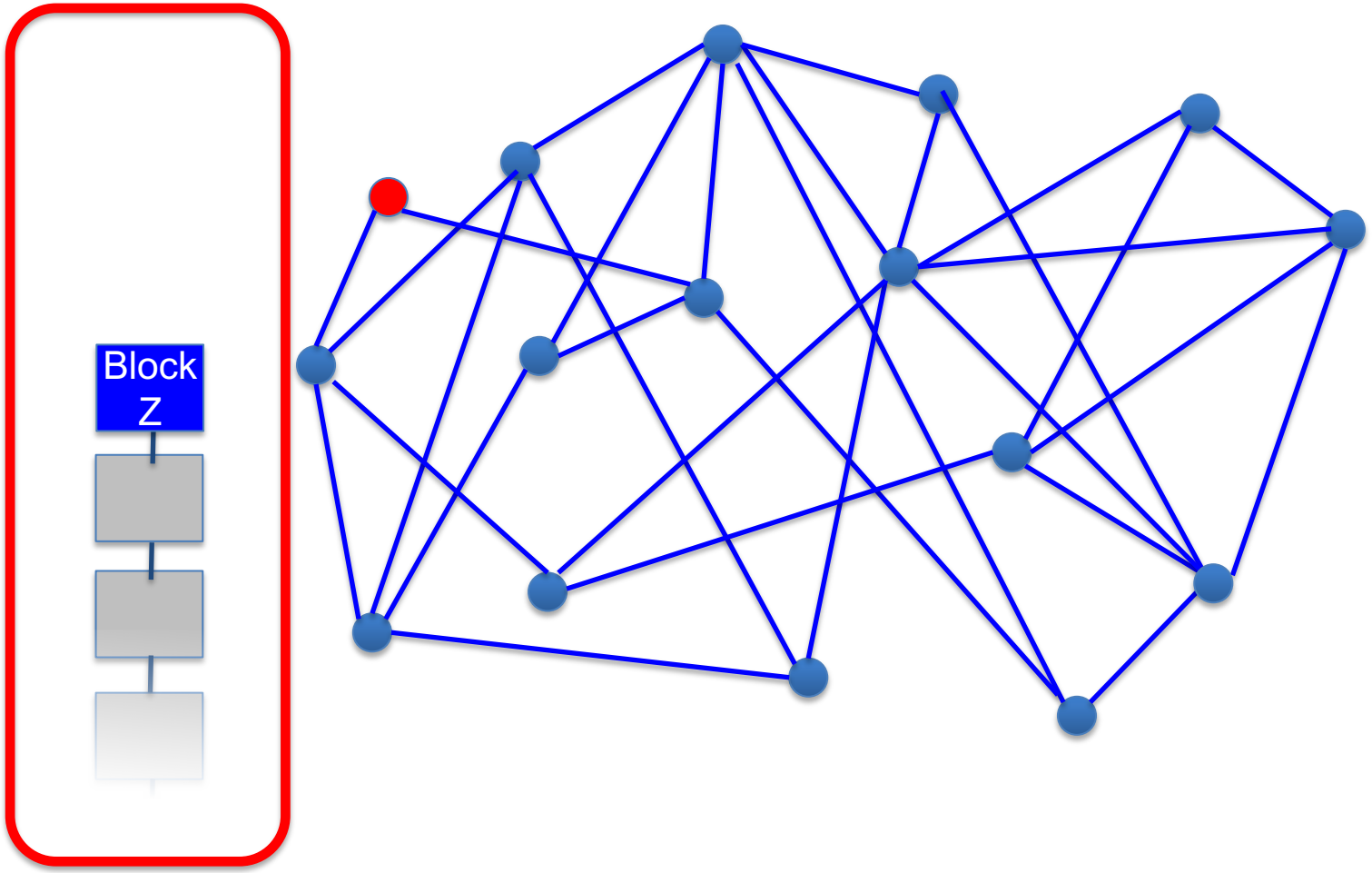




- Initially: the most recent block mined and added to the chain is Block Z.
- Every node in the network of miners recognize this: their view is **consistent**
- The miners are busy working on a block for a new set of transactions…

(after Antonopolous 2014, pp.202ff)

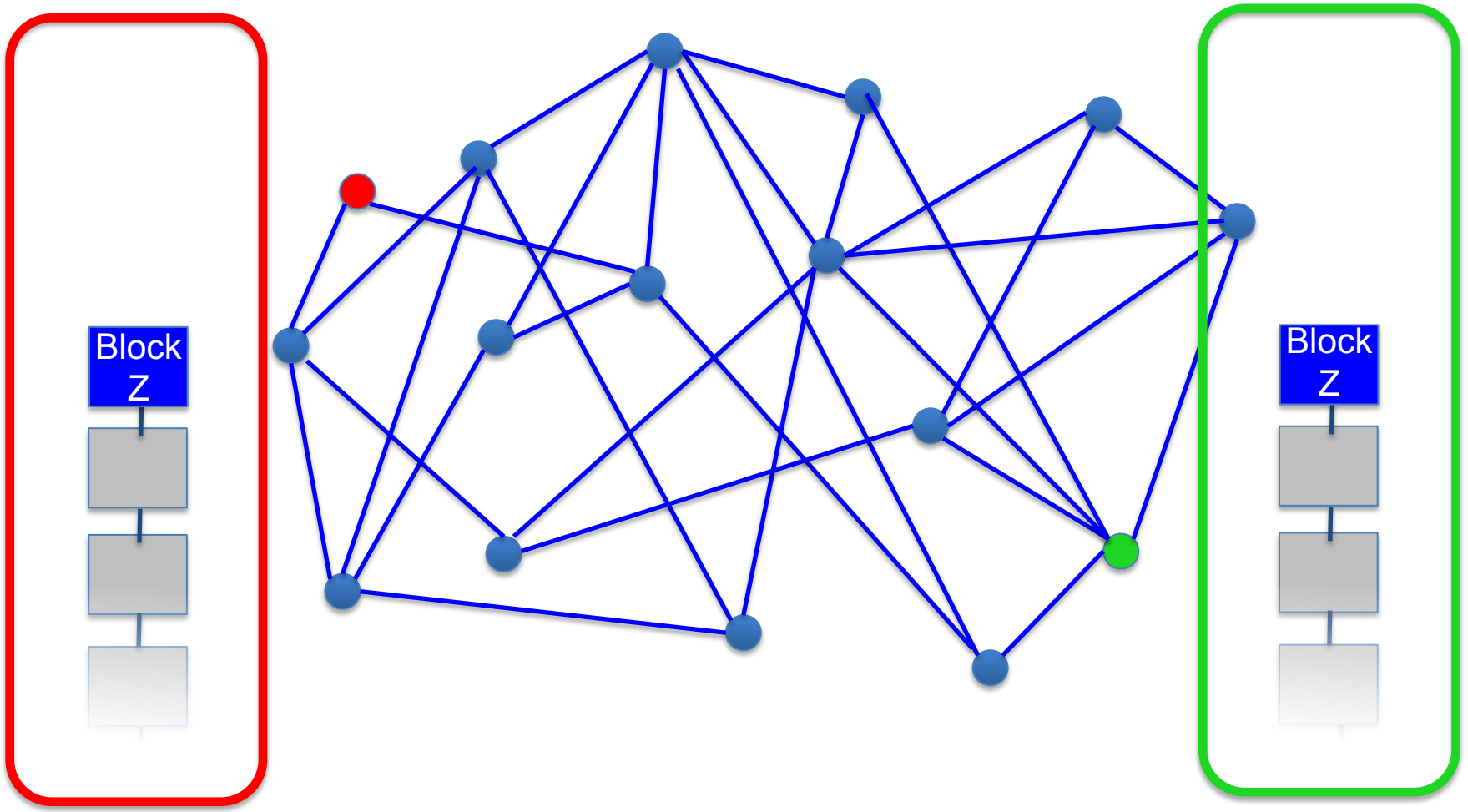
Blockchain Handling A Fork



- Now that consistency is about to break down, so let's keep a track of the red node's view of the Blockchain, on the left hand side of the diagram…

(after Antonopolous 2014, pp.202ff)

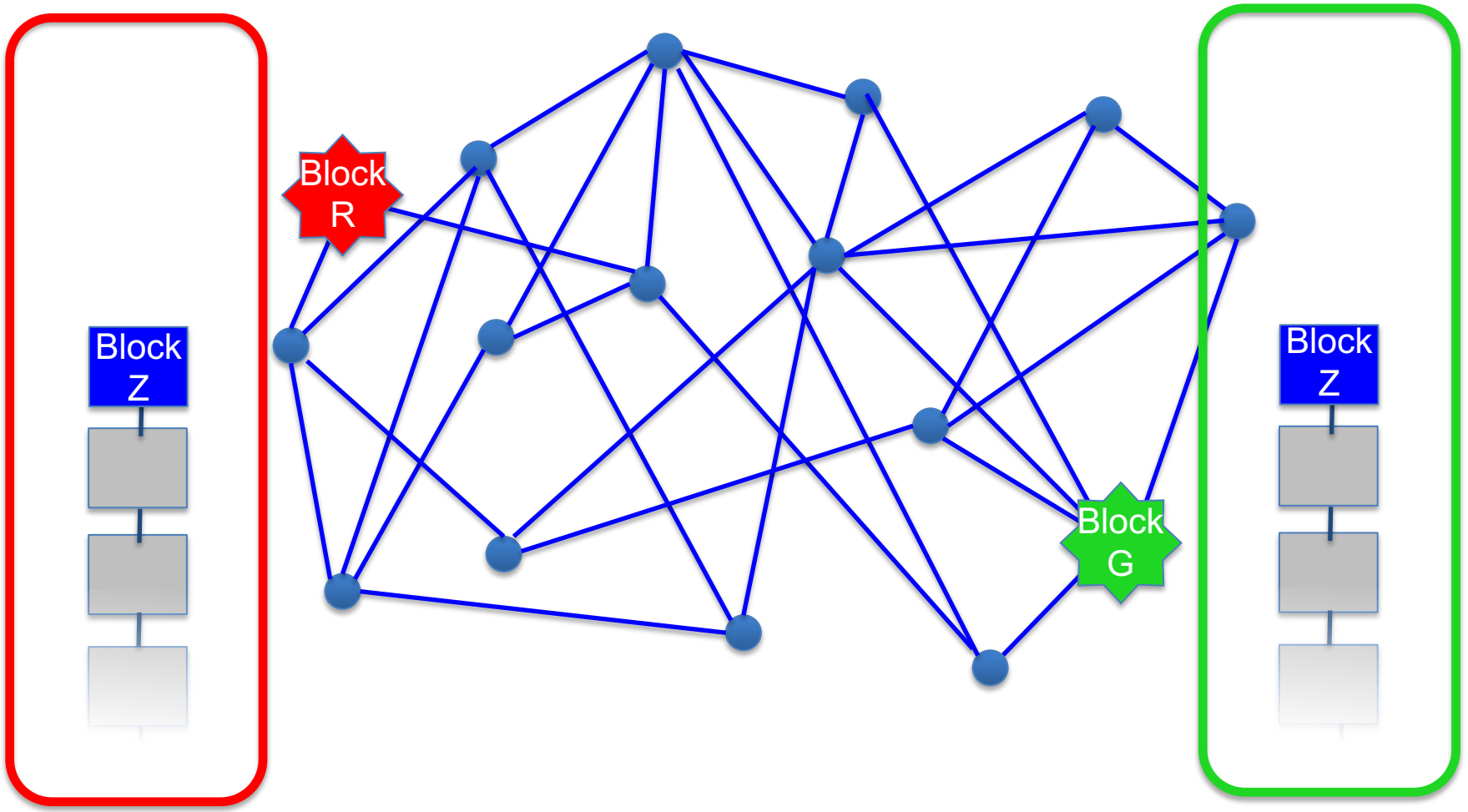
Blockchain Handling A Fork



- …and let's keep track of the green node's view of the Blockchain on the right-hand side

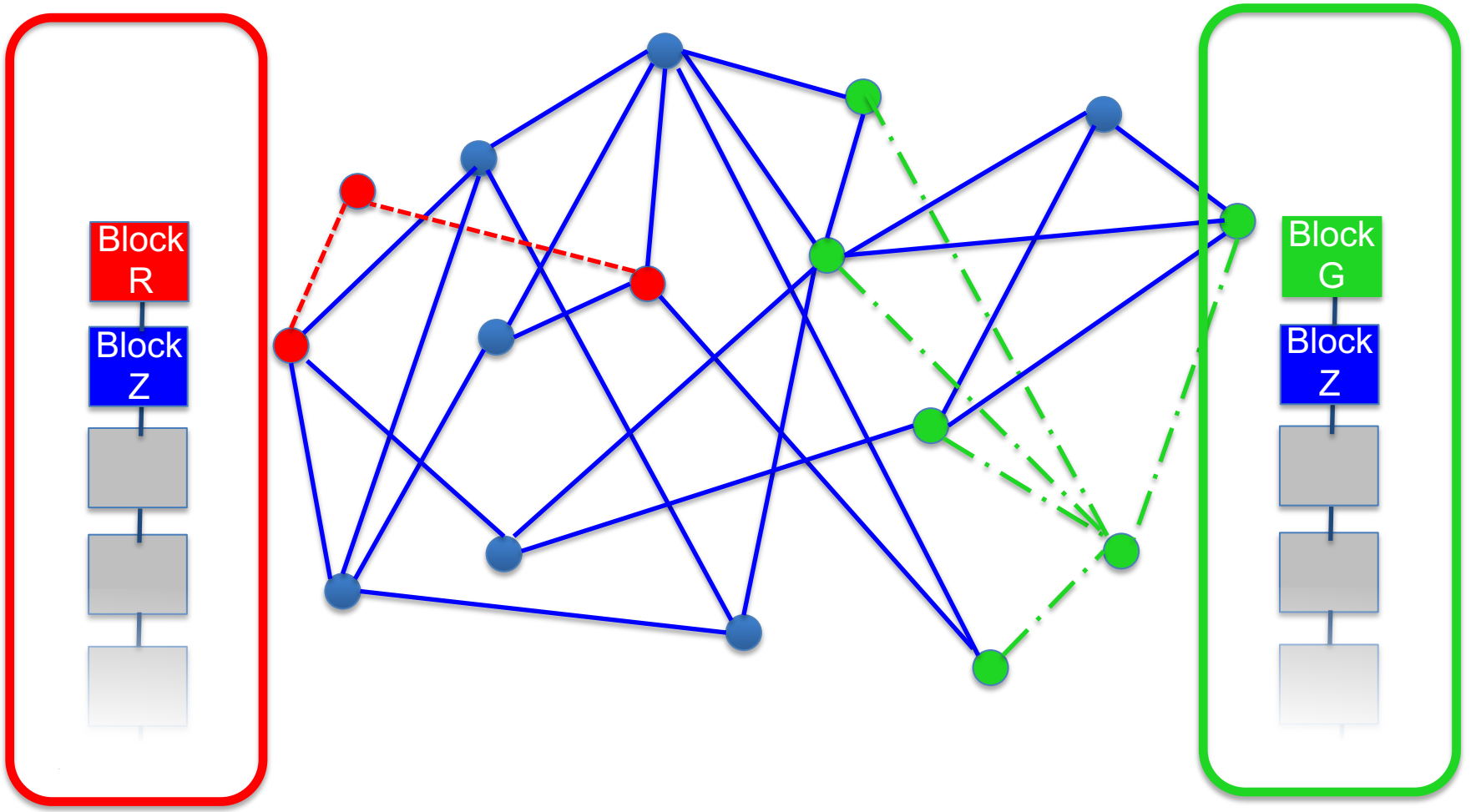
(after Antonopolous 2014, pp.202ff)

Blockchain Handling A Fork



- Now the red node mines a new block!
- And at almost the same time the green node also mines a new block!
- …maybe for conflicting transactions!

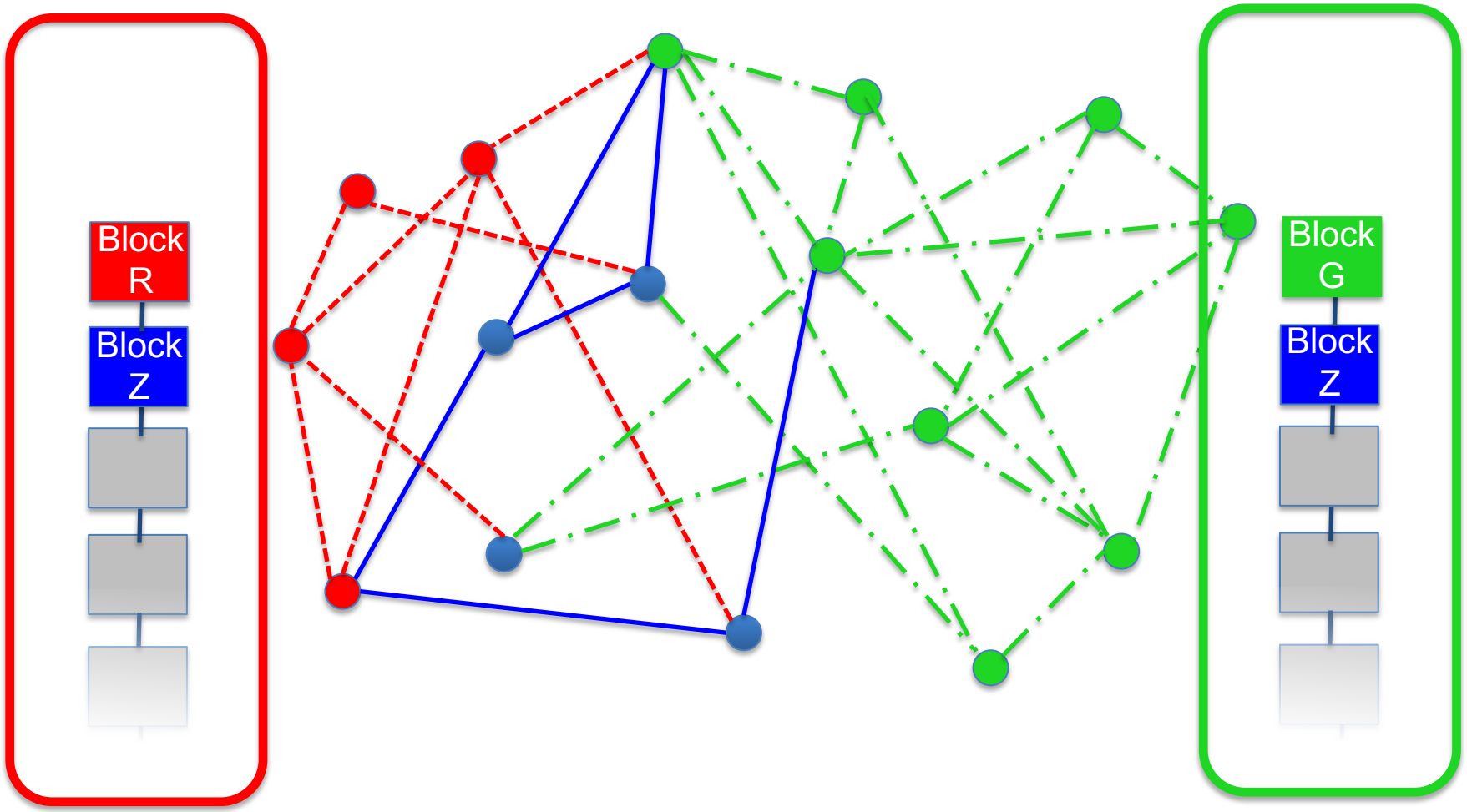
Blockchain Handling A Fork



- Each node propagates the new extended Blockchain to its neighbour nodes

(after Antonopolous 2014, pp.202ff)

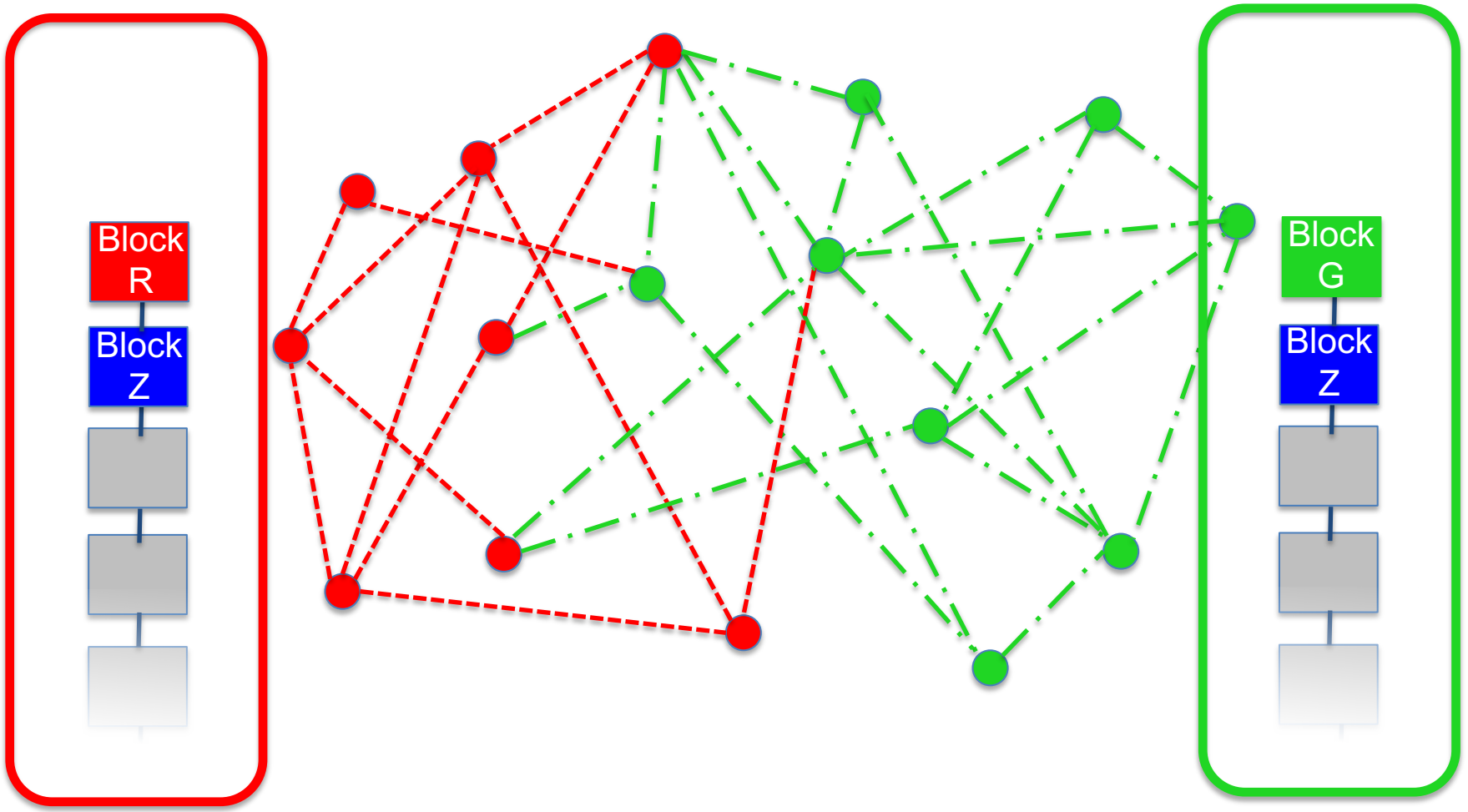
Blockchain Handling A Fork



- The propagation continues

(after Antonopolous 2014, pp.202ff)

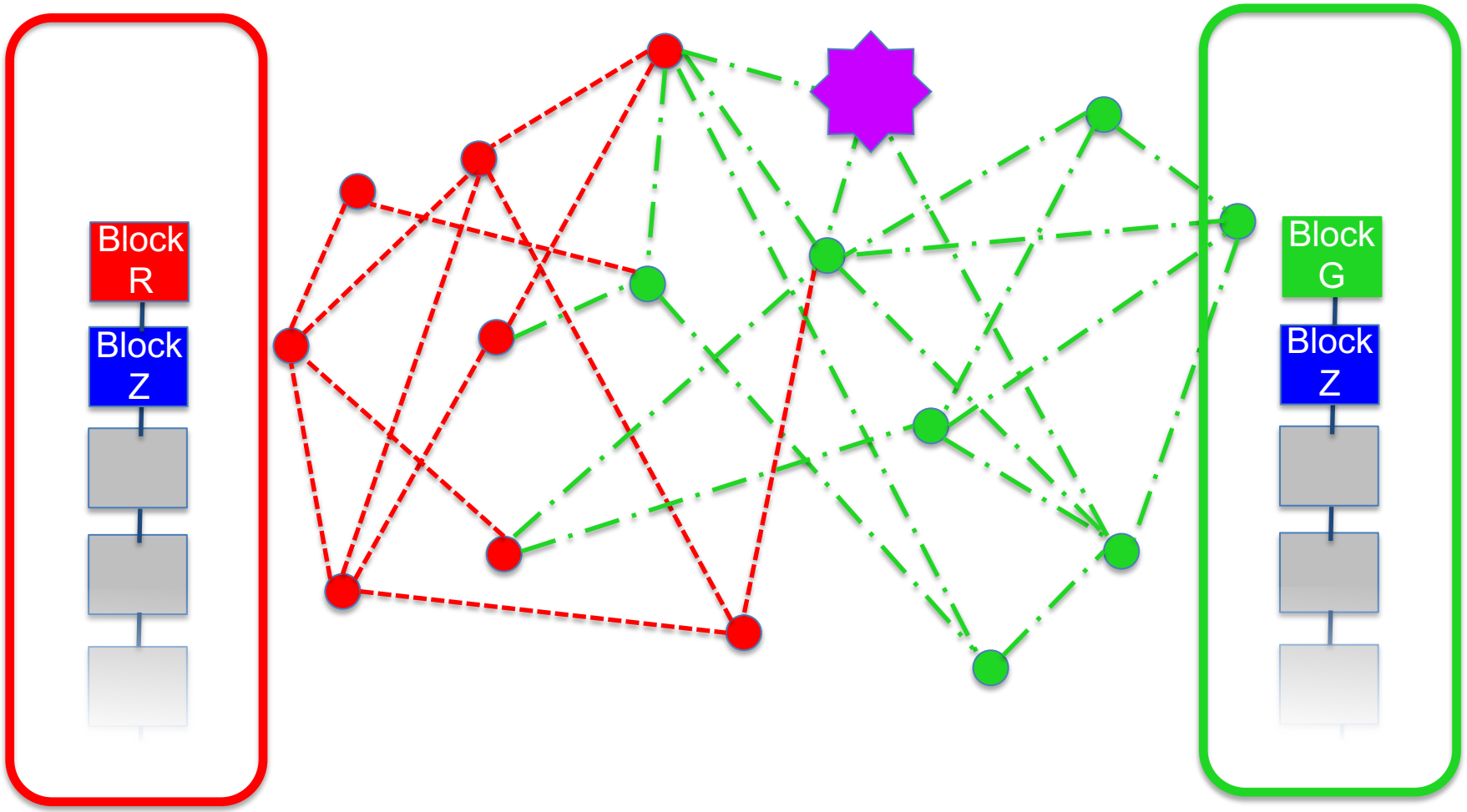
Blockchain Handling A Fork



- All nodes eventually update their record of the Blockchain… but there are now two inconsistent versions

(after Antonopolous 2014, pp.202ff)

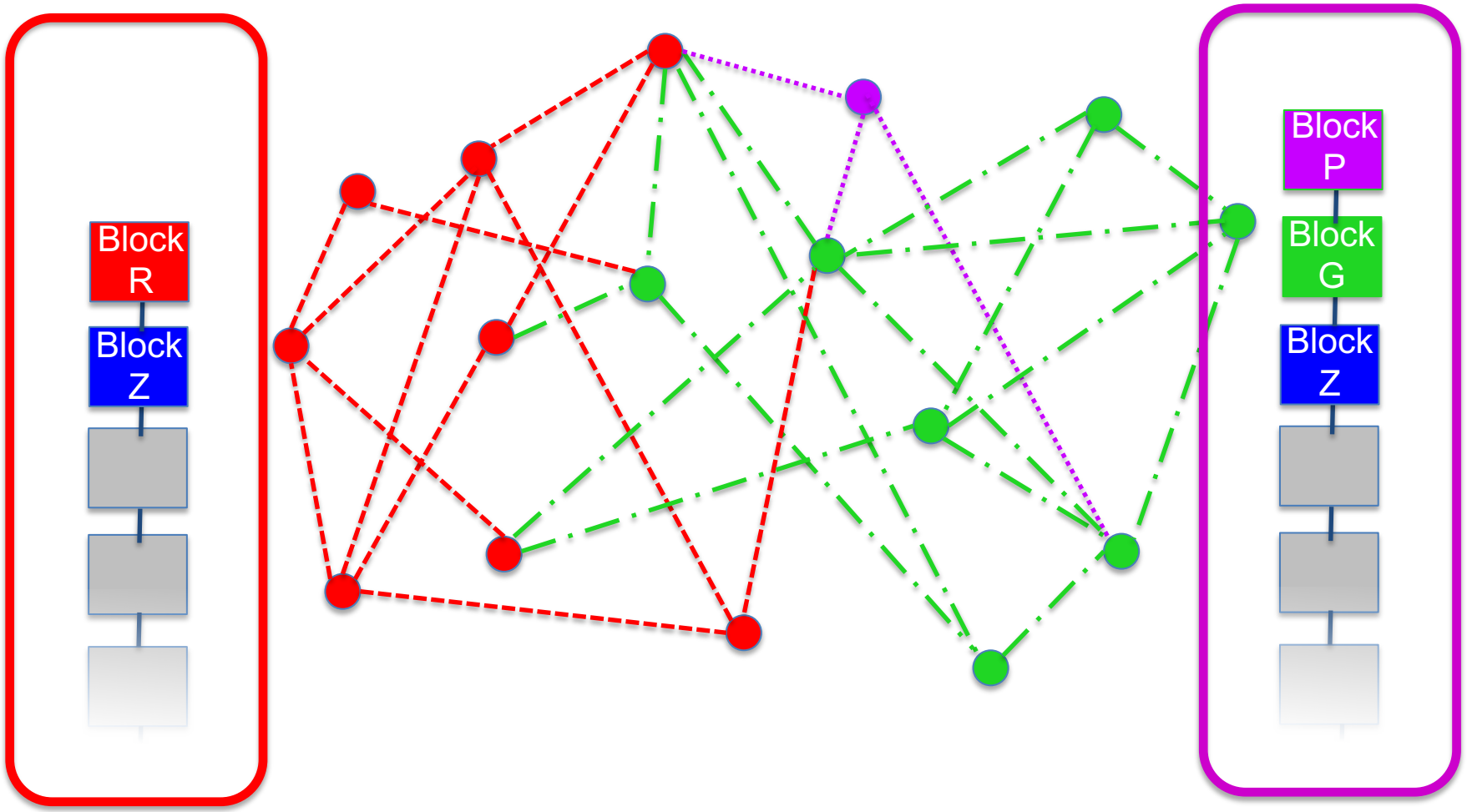
Blockchain Handling A Fork



- Then, say that the next block mined is generated by one of the green nodes who know about Block G...

(after Antonopolous 2014, pp.202ff)

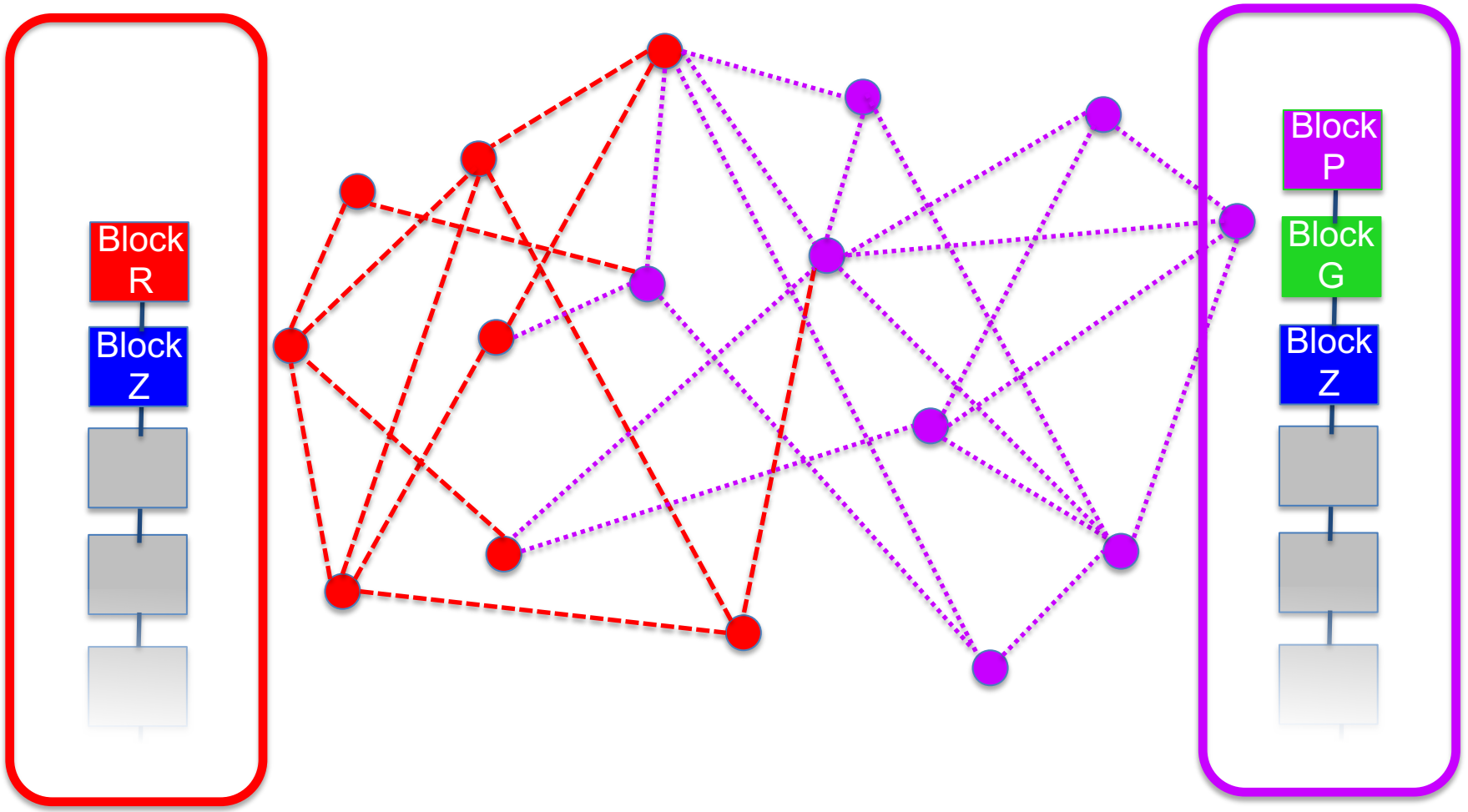
Blockchain Handling A Fork



- That then propagates across the network

(after Antonopolous 2014, pp.202ff)

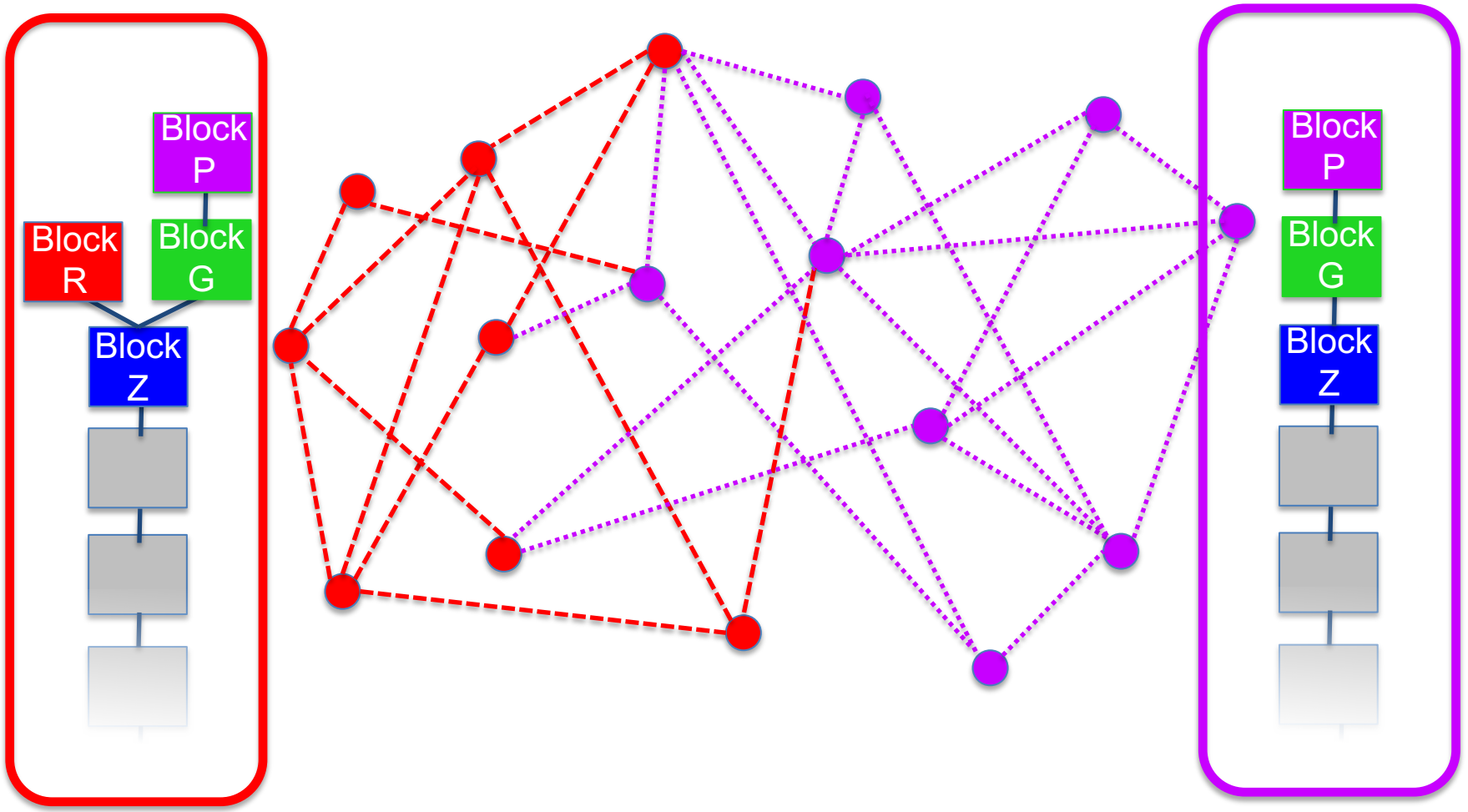
Blockchain Handling A Fork



- Nodes that had Block G in their block chain see Block P as an extension of the chain, validating Block G's position in the chain.

(after Antonopolous 2014, pp.202ff)

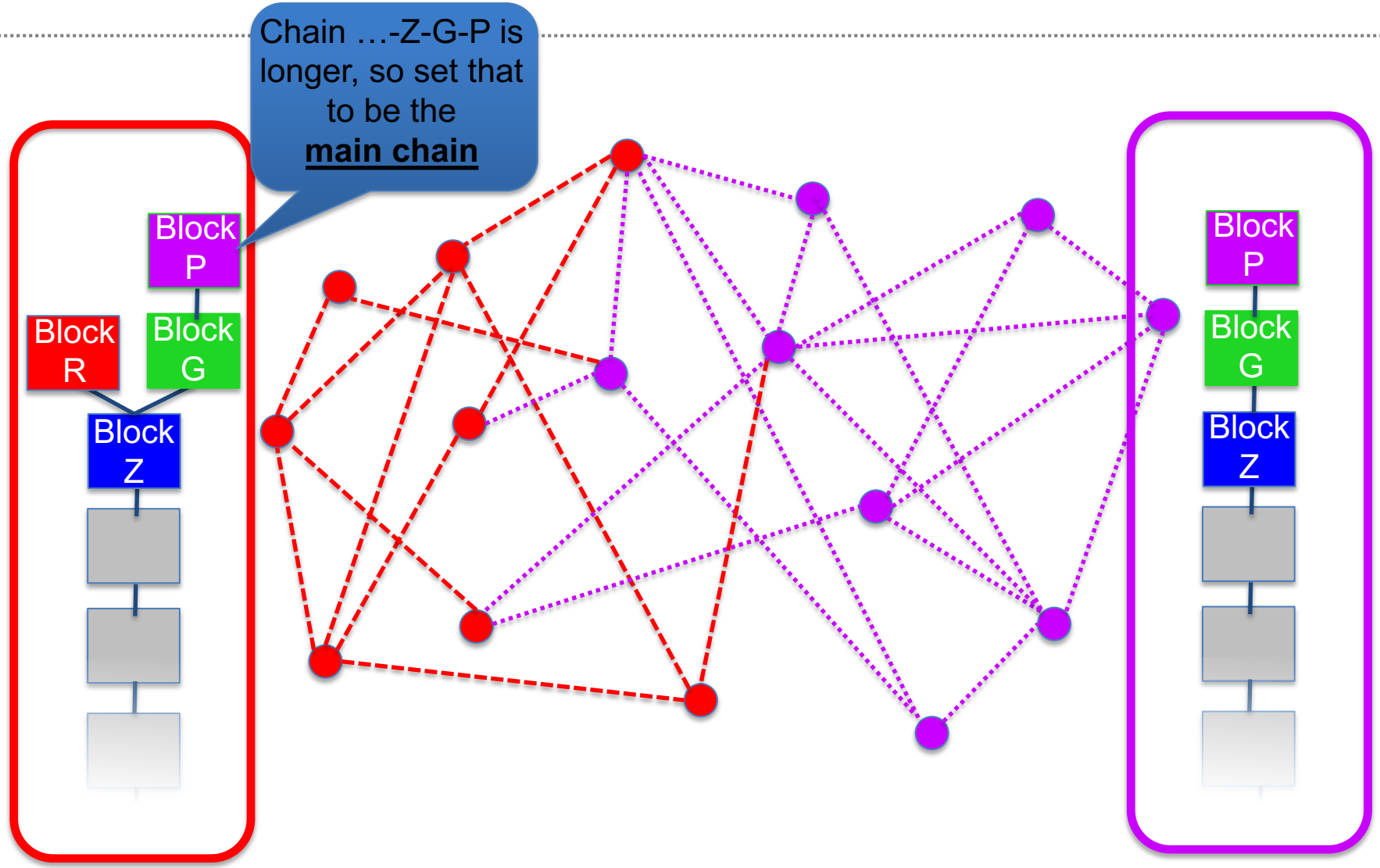
Blockchain Handling A Fork



- But the nodes that had previously seen Block R, and added it into their chain now see **two** chains

(after Antonopolous 2014, pp.202ff)

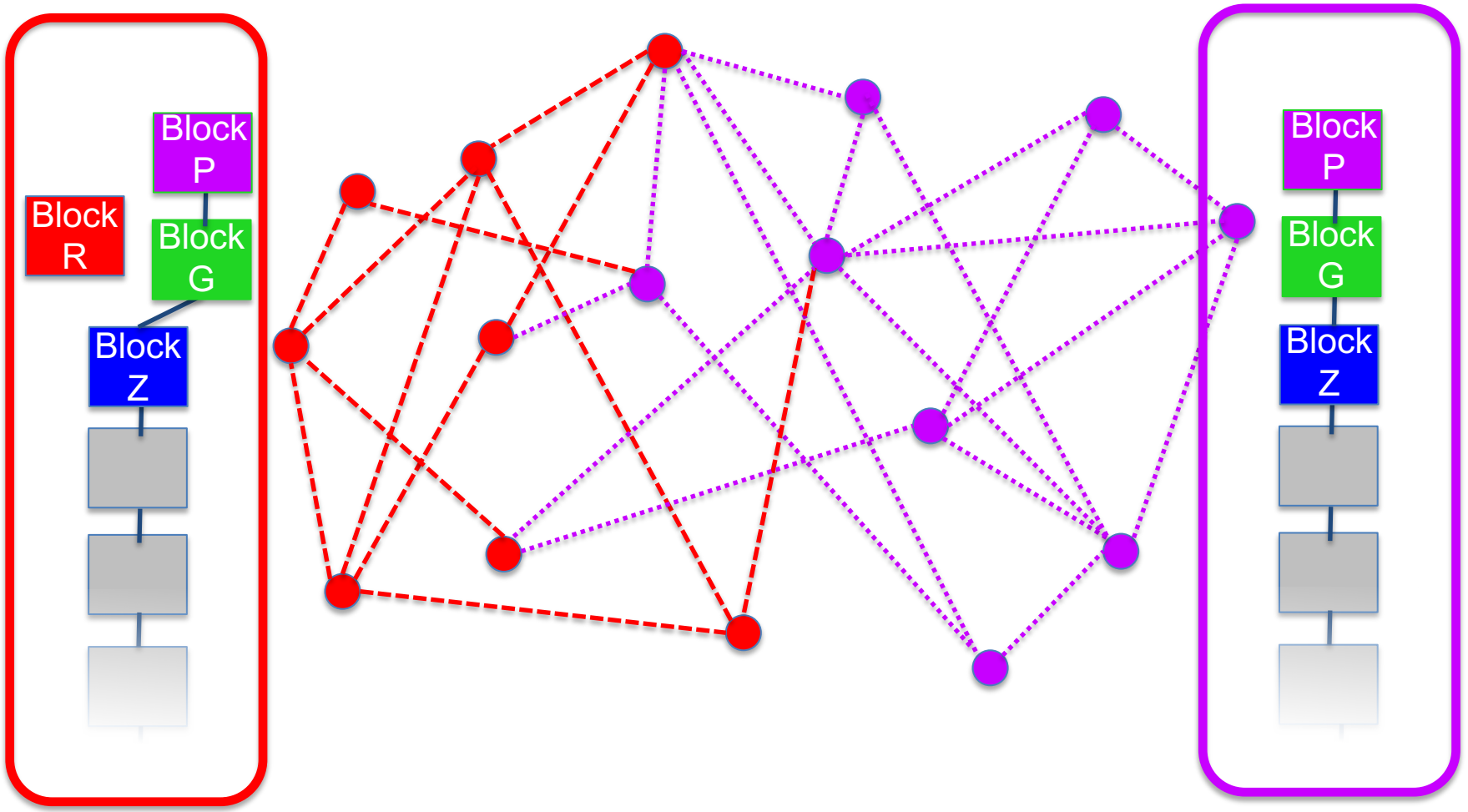
Blockchain Handling A Fork



- But the nodes that had previously seen Block R, and added it into their chain now see **two** chains

(after Antonopolous 2014, pp.202ff)

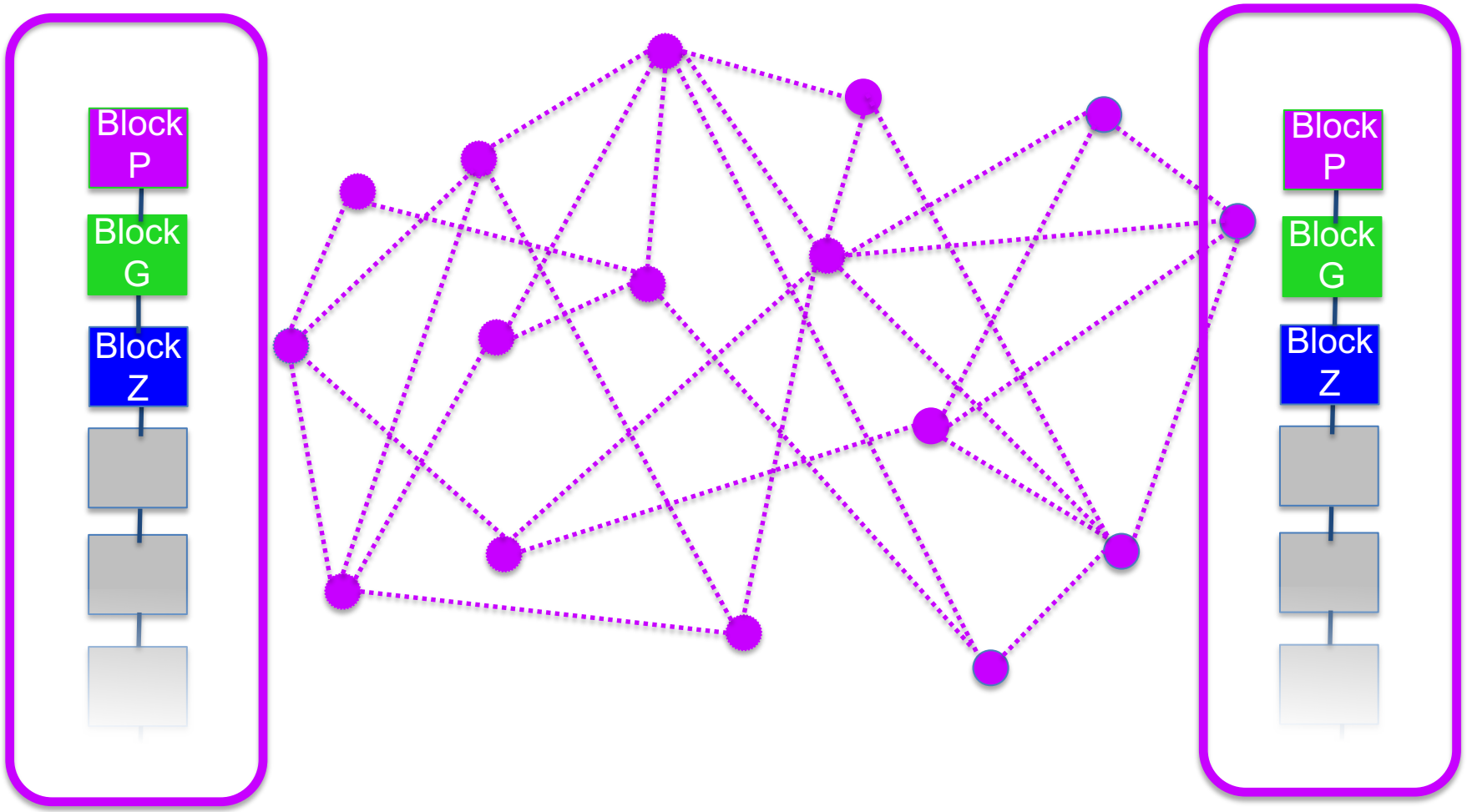
Blockchain Handling A Fork



- Block R is now an “orphan” block and its transactions will need to be re-verified…

(after Antonopolous 2014, pp.202ff)

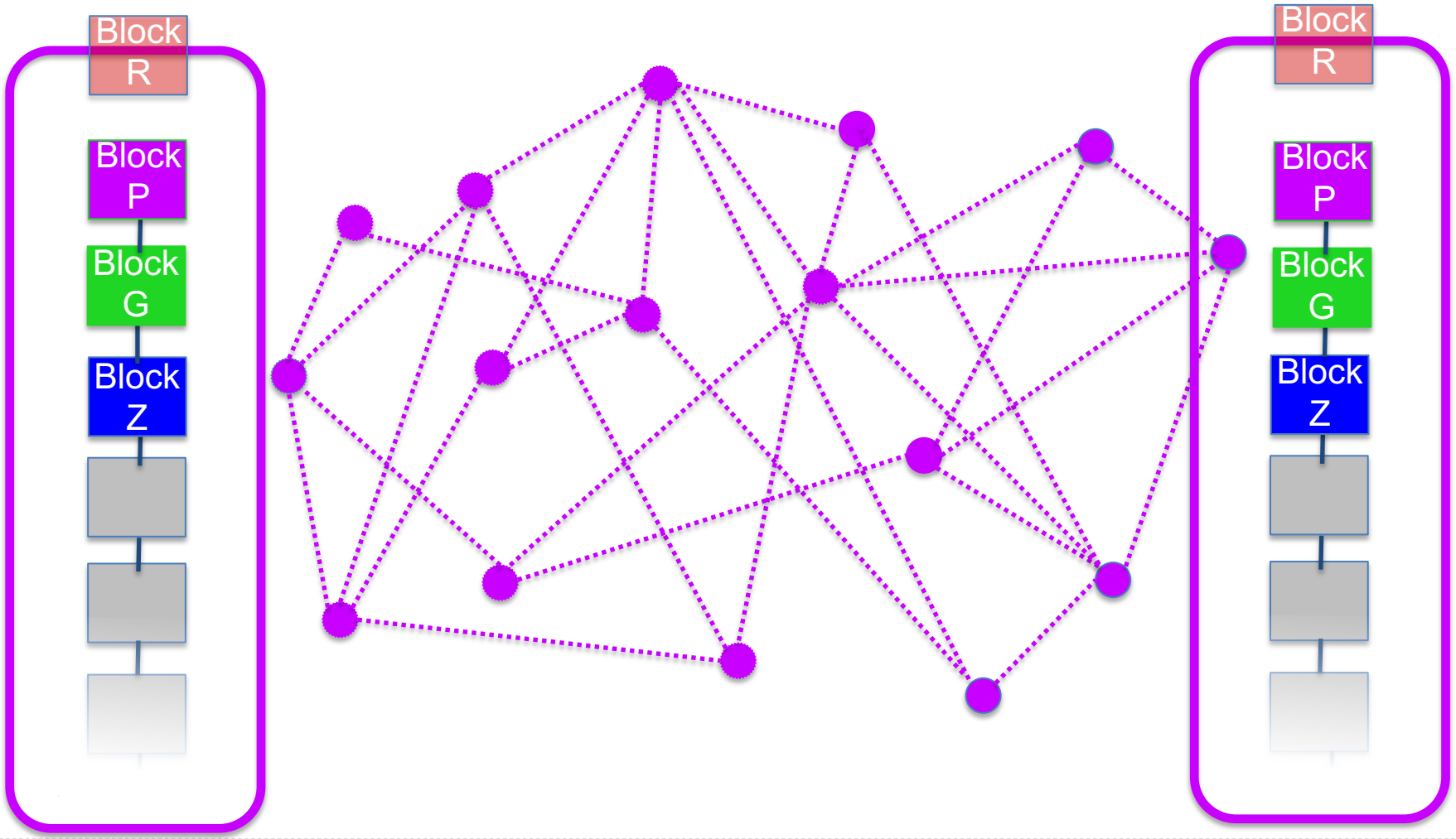
Blockchain Handling a Fork



- Any miners working on extending the …-Z-R chain will stop when they are made aware of a longer chain (…Z-G-P).

(after Antonopolous 2014, pp.202ff)

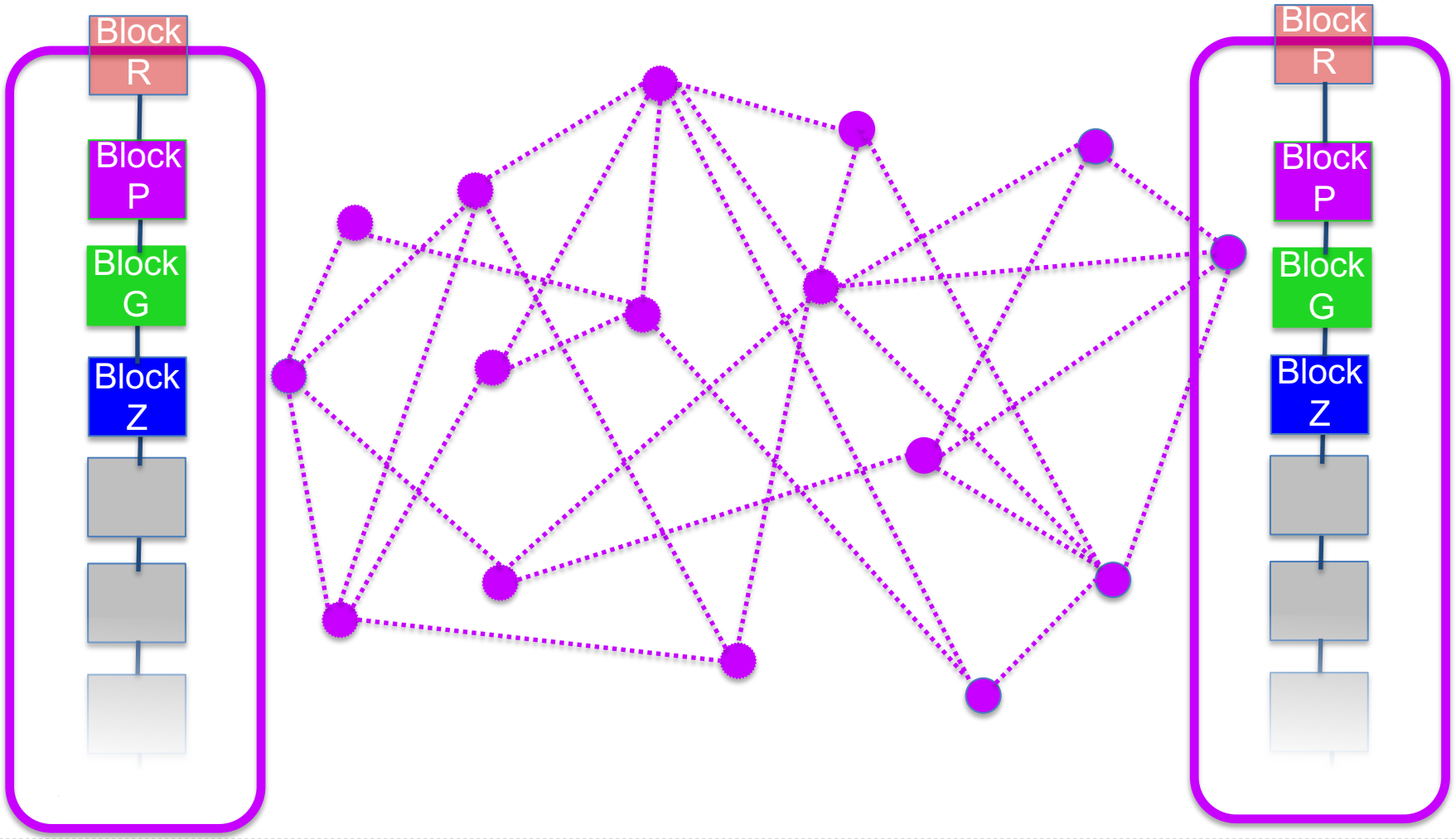
Blockchain Handling A Fork



- Transactions that were in Block R are queued up for inclusion in the next block
- (Block R is no longer in the main chain)

(after Antonopolous 2014, pp.202ff)

Blockchain Handling A Fork



- The fork has been dealt with.
- Note that the fact that it takes *time* to mine a block is crucial
- This is the crux of the "Proof of Work" idea…

(after Antonopolous 2014, pp.202ff)

Attack of the Forking Miners

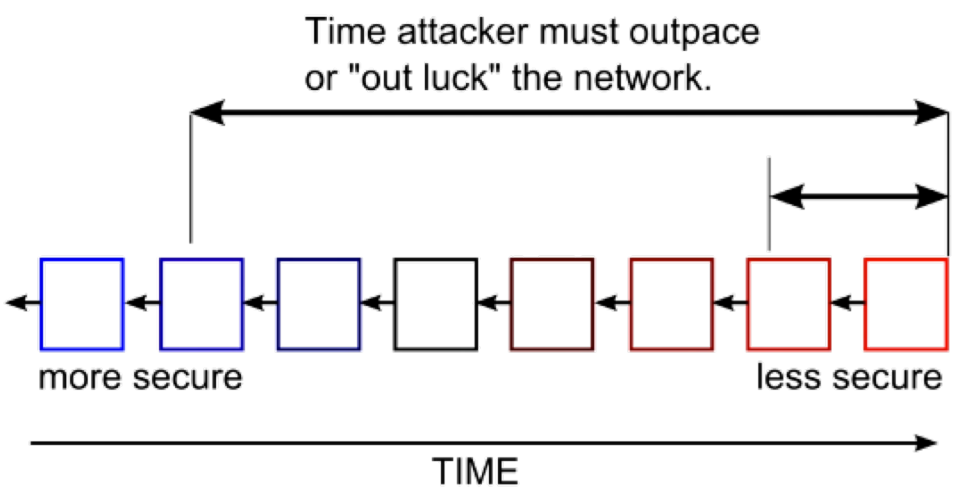
What if the forks that occur aren't just accidents but are *malicious*?

What if a gang of rogue miners are trying to make money from forks? They could:

- Reverse transactions (steal money back)
- Prevent transactions from gaining any confirmations (block payments)

But they need >50% of the compute power on the whole network in order to remain in control of the Blockchain:

- Otherwise other miners will fix the forks



Decentralised Ledger Revolution

- The R3 CEV LLP (or just R3) Consortium of 20+ banks includes:
  - Goldman Sachs, JPMorgan, Credit Suisse, UBS, Barclays, RBS, etc.
- They are not so interested in Bitcoin as a rogue currency…
- …but they are interested in *Blockchain* as a revolutionary way of…
  - handling the problem of maintaining a consensus…
  - in a complex system of agents working in parallel…
  - without needing to rely on a central bottleneck.
- R3 are exploring the creation of their *own* Blockchain for recording and auditing inter-bank transactions.

University of BRISTOL

D… dger Revolution

- The R3 CEV LLP (… ks includes:
  - Gol… RBS, etc.
- They a… ency…
- …but th… nary way of…
  - handlir…
  - in a com…
  - without …
- R3 are expl… ions.
  Blockchain …

Blockchain

More trouble in blockchain-land

44

AUGUST 22 2019 By: Izabella Kaminska

By any measure, R3 was supposed to be one of the surest of the enterprise blockchain consortium bets. Headed by financial-platform industry veteran David Rutter, of ICAP and Prebon Yamane heritage, it was one of the earliest of the blockchain projects to pull in serious backing from the banking community.

And yet, just as Digital Asset is also finding out, it turns out that attracting funding for blockchain-based development is far easier than delivering an actual viable and usable product to market. The rollout of the company's flagship "Corda Enterprise" system has allegedly been plagued by delays, frustrating insiders.

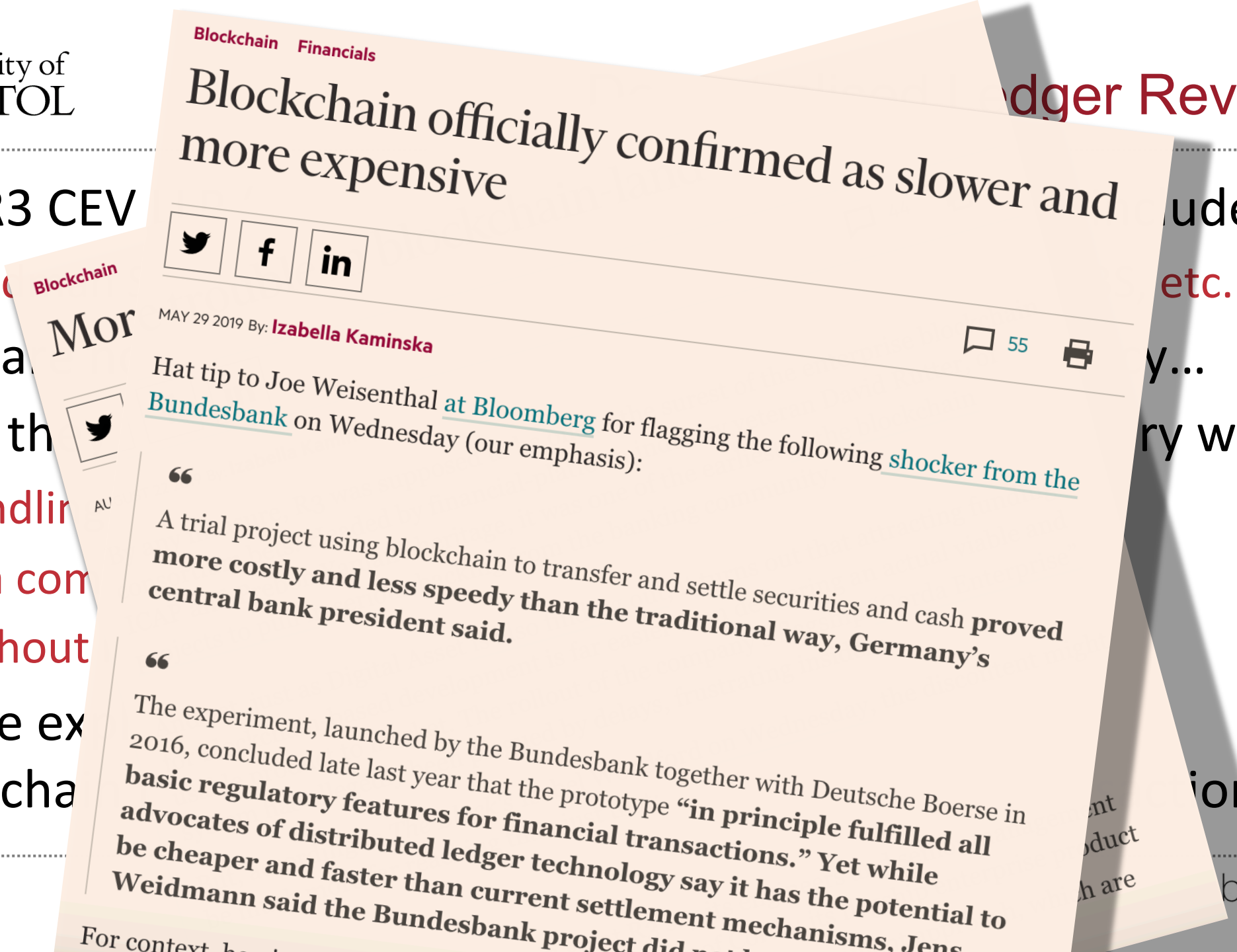
But, according to *The Block*'s Isabel Woodford on Wednesday, the discontent might be much more far reaching than just that:

…giant R3 say its engineers and senior management …e of Corda; its flagship enterprise product …ment approach, which are

University of BRISTOL

dger Revolution

- The R3 CEV udes:
  - Gol etc.
- They a y…
- …but th ry way of…
  - handlir
  - in a com
  - without
- R3 are ex ions.
  Blockcha



Blockchain Financials

Blockchain officially confirmed as slower and more expensive

MAY 29 2019 By: Izabella Kaminska

55

Hat tip to Joe Weisenthal at Bloomberg for flagging the following shocker from the Bundesbank on Wednesday (our emphasis):

> A trial project using blockchain to transfer and settle securities and cash **proved more costly and less speedy than the traditional way, Germany's central bank president said.**

> The experiment, launched by the Bundesbank together with Deutsche Boerse in 2016, concluded late last year that the prototype **"in principle fulfilled all basic regulatory features for financial transactions." Yet while advocates of distributed ledger technology say it has the potential to be cheaper and faster than current settlement mechanisms, Jens Weidmann said the Bundesbank project did**

For context, h

Not Just Banks That Are Interested…

- Land registry in Honduras by Factcom
- Stock exchange trades at NASDAQ
- Diamond registry at Everledger
- Password substitute at Onename
- All or nothing crowd-sourcing
- Self-Owning self-driving cars on Ethereum

Blockchains

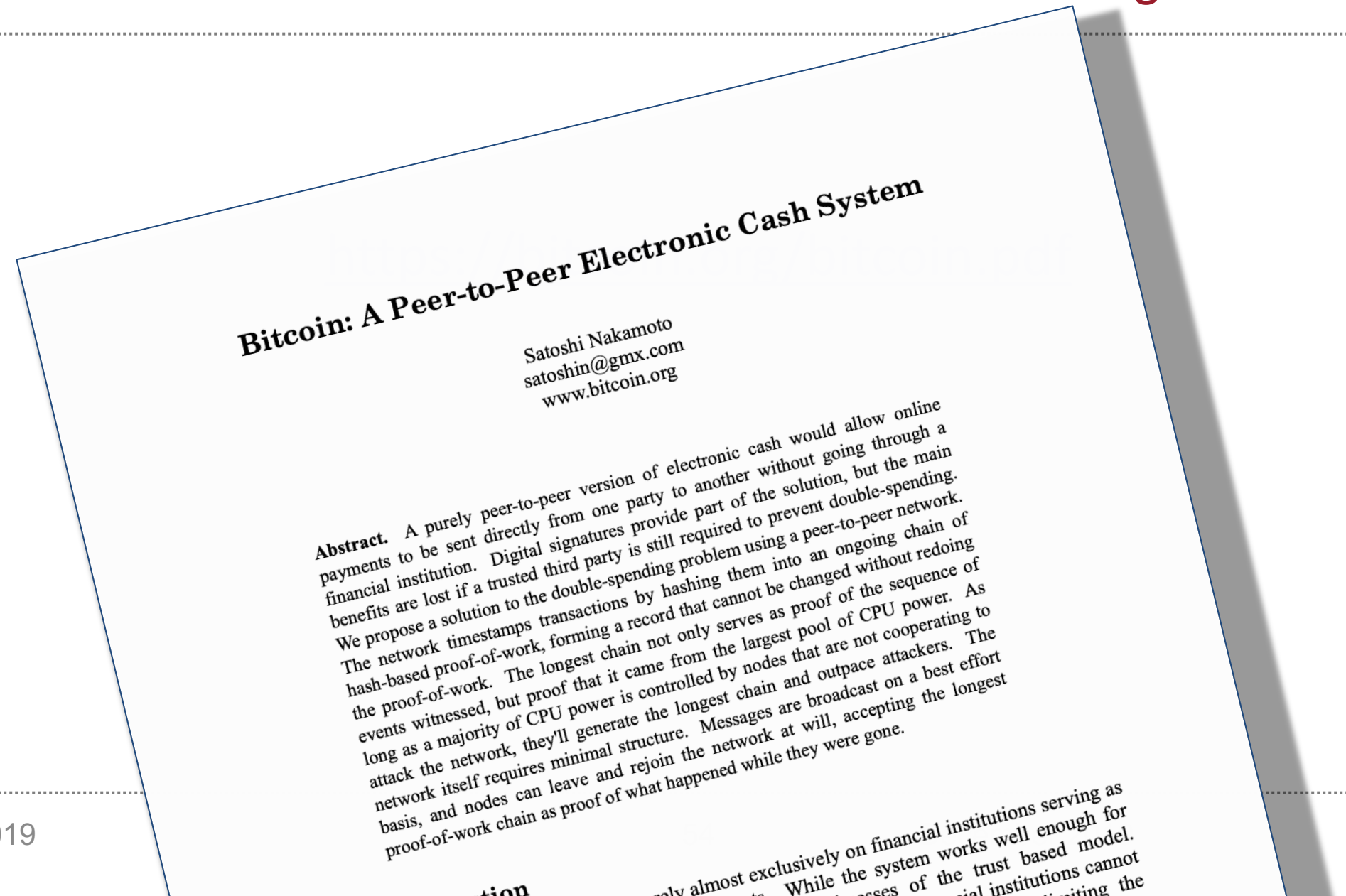
The great chain of being sure about things

The Economist

Link to Satoshi Nakamoto's Original Whitepaper

https://bitcoin.org/bitcoin.pdf

Link to Satoshi Nakamoto's Original Whitepaper



Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

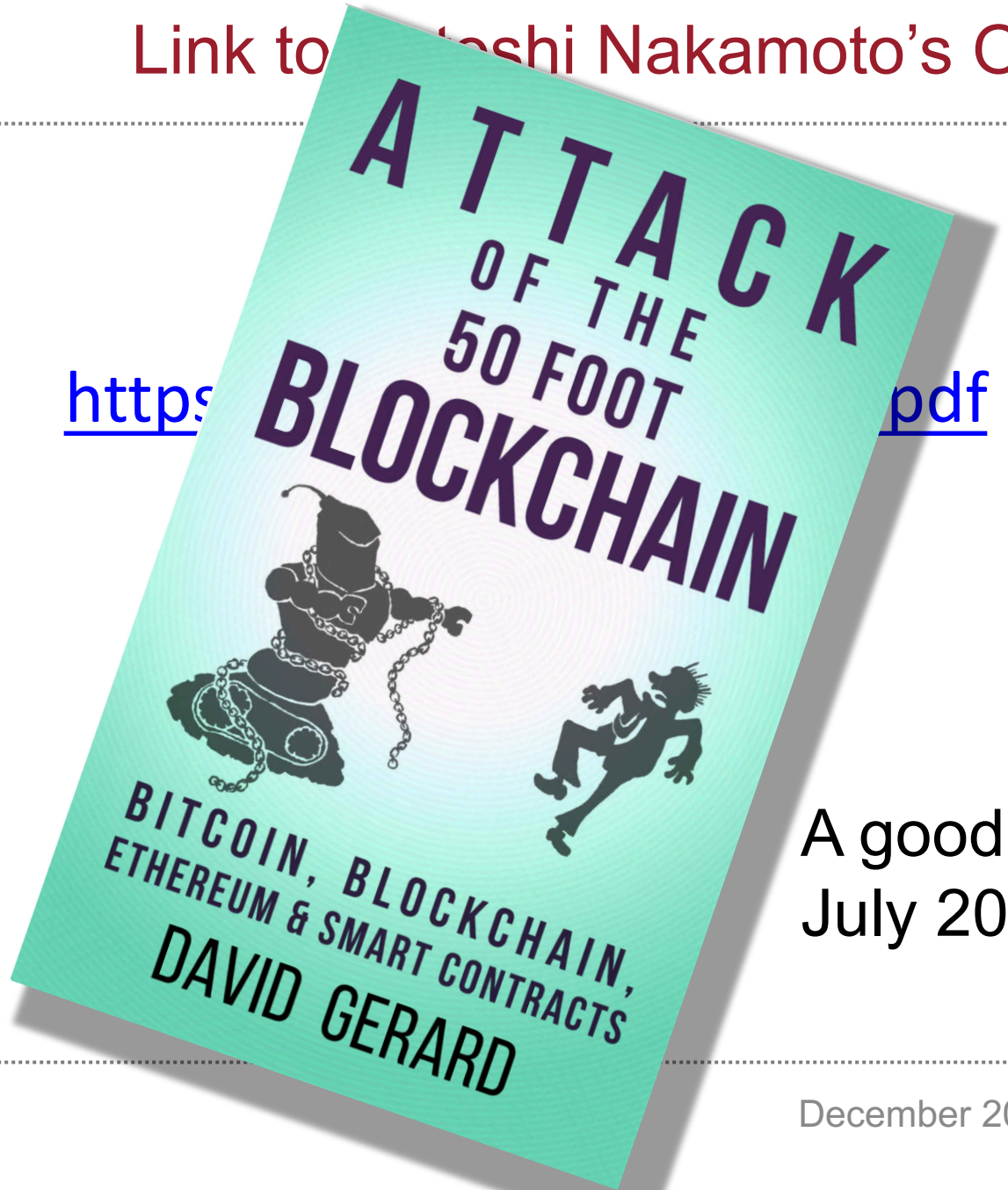
Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

University of BRISTOL

Original Whitepaper

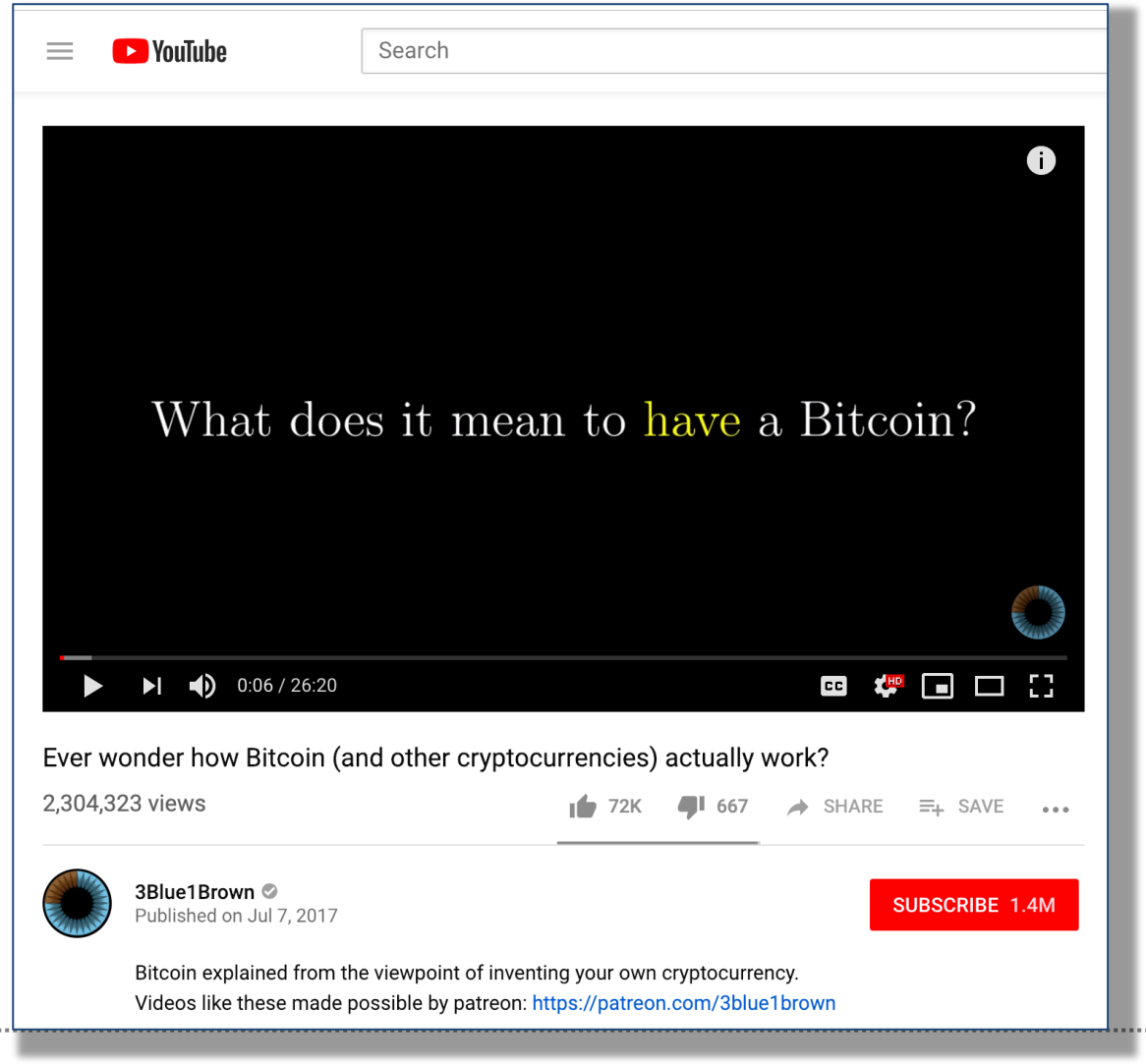
Link to [illegible]toshi Nakamoto's Original Whitepaper

https[illegible]pdf



A good read (non-academic)
July 2017

https://www.youtube.com/watch?v=bBC-nXj3Ng4



A really great video explanation
July 2017

The End