SCALING BITCOIN WORKING PAPER

Subchains Facilitate On-chain Scaling and Fast Transaction Verification

Peter R. Rizun[†]

20 December 2015
Extended & Revised 8 August 2016

Abstract. Orphaning risk for large blocks limits Bitcoin's transactional capacity while the lack of secure instant transactions restricts its usability. Progress on either front would help spur adoption. This paper considers a technique for using fractional-difficulty blocks (weak blocks) to build subchains bridging adjacent pairs of full-difficulty blocks (strong blocks). Subchains both reduce orphaning risk by propagating block contents over the entire block interval, and add security to zero-confirmation transactions due to the weak blocks built above them. Miners are incentivized to cooperate building subchains in order to process more transactions per second (thereby claiming more fee revenue) without incurring additional orphaning risk. The use of subchains also diverts fee revenue towards network hash power rather than dripping it out of the system to pay for orphaned blocks. By nesting subchains, weak block confirmation times approaching the theoretical limits imposed by speed-of-light constraints would become possible with future technology improvements. As subchains are built on top of the existing Bitcoin protocol, their implementation does not require any changes to Bitcoin's consensus rules.

KEY WORDS

1. On-chain scaling. 3. Weak blocks. 4. Network security. 5. Instant transactions. 6. Fractional confirmations.

1. Introduction

Bitcoin's performance as a payment network is hardly impressive. In 2015, it processed an average of 1.4 transactions per second[1] while merchants waited on average eight minutes to receive initial verification from a miner.[2] In contrast, the Visa network processed over 2,000 transactions per second, [3] and—with chip-and-PIN technology—merchants received authorization and PIN-verification in under a second.[4] Unlike Visa, Bitcoin's transactional capacity is (partly) limited due to miners' hesitation to produce blocks containing large volumes of new transactions.[5] Such blocks propagate across the network slowly,[6] increasing the chances that the block is orphaned and the miner's reward is lost. Also unlike Visa, the initial verification of a transaction by a miner is delayed because blocks are propagated on average only every ten minutes,[7] rather than at a rate dynamically tuned to the bandwidth and latency of the network. In this paper, we present a scaling technique called *subchains* to build blocks layer-by-layer—at a small fraction of Bitcoin's ten-minute block time—thereby reducing both orphaning risk and the wait-time for the first verification of a transaction by a miner.

Throughout this paper, we make certain simplifying assumptions. In particular, we assume that:

[†] P. R. Rizun (peter_r@gmx.com) is a physicist and entrepreneur from Vancouver, Canada.
The original PDF version of this document has been time-stamped in the Blockchain.
1BWZe6XkGLcf6DWC3TFXiEtZmcyAoNq5BW

- The network consists of both *honest miners* who reliably follow the agreed-upon protocol and *unscrupulous miners* who will deviate from the protocol to facilitate attacks if doing so is in their short-term-greedy best interest.
- The network hash rate is constant over a given block interval.[8]
- Block information propagates with a well-defined impedance measured in time per bytes propagated.[6,9,10]
- The free-market equilibrium block size is smaller than the protocol-enforced block size limit (if such a limit exists).

In Section 3, we describe the subchain technique,[11] which is a practical application of weak blocks[12,13,14,15] that provides incentives for miners to cooperate for the mutual benefit of the network. Its implementation requires neither a hard nor soft fork—but it does require participation from a significant fraction of the network hash power in order to be useful. A significant advantage of this technique is revealed in Section 4, when we show how subchains considerably reduce the probability of orphan races.

We move onto transaction fee market economics in Section 5. We show that although a miner can include all of the subchain's transactions in his block candidate—and thus all of the subchain's fees—without incurring orphaning risk, he still incurs orphaning risk for *new* transactions included in his block candidate. This property drives a fee market and economically restricts the rate of subchain growth.

Certain investigators have argued that fees that result from orphaning risk do not contribute to network security.[16] With a simple diagram, we prove this line of reasoning false in Section 6 by showing that the fees already included in the subchain contribute *directly* to network security in the same way that the block reward does. Only the fees in the new (marginal) transactions added on top of the subchain go to cover the (marginal) orphaning risk for those transactions.

In Section 7 we review the security of zero-confirmation transaction under standard block propagation rules, and explain why the lower bound on security is zero. We then show that double-spending a transaction verified in a subchain has an objectively-measurable cost on the order of the total fees that have accumulated in the subchain above the transaction an attacker is attempting to double spend (Section 8). In Section 9, we illustrate how subchains can be nested, creating a fractal-like blockchain structure where transactions are processed almost continuously. Let us begin by defining the symbols we use.

2. List of Symbols

For the remainder of this manuscript, the following symbols have the specified meanings.

F	fees	R	orphaning risk
ΔF_i	fees included in the ith Δ-block	ΔR_i	orphaning risk of the ith Δ-block
Δf	bribe offered by attacker	T	block interval (10 min target)
i	index of Δ-block	ΔT	Δ-block interval
m	number of subchain verifications	t	time
n	Δ -block index for double-spend	X	subchaining factor
P_{orphan}	probability of an orphan race event	z	propagation impedance (time per bytes propagated)

Q	block size or block space in bytes	χ	fraction of hash power controlled by unscrupulous miners
Q_i	subchain size up to the ith Δ-block	τ	propagation time
ΔQ	size of Δ-block	τ_0	propagation latency
		$\Delta\tau$	propagation time minus latency

The symbol $ refers to *US dollars*; price conversions between bitcoin and US dollars assume that 1 Ƀ = $400.

3. Subchains

To append a new block to the Blockchain, a miner must find a valid proof-of-work. This entails finding a nonce that when hashed together with the previous block's hash and the root hash for the block's transactions, results in an integer less than the difficulty target.[17] We define a *weak block* as a block that satisfies the weaker requirement

hash(previous hash, nonce, root hash) < weak target,

where the weak target is larger than the difficulty target. By sharing these weak blocks, miners can cooperate to build *subchains* (Fig. 1).

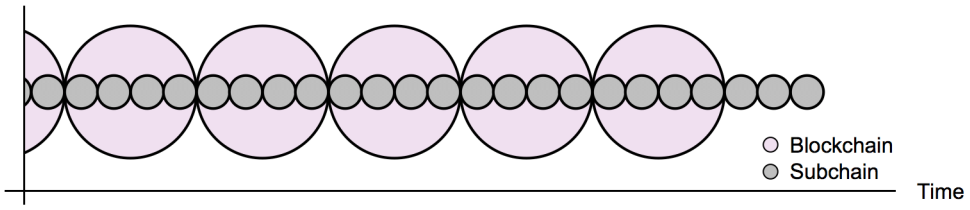

Fig. 1. Miners cooperate to build subchains in order to process more transactions and claim more fees without incurring additional orphaning risk. This illustration visualizes ¼-difficulty "idealized" subchains (also referred to as 4x subchains); in reality, some strong blocks will contain more than four weak blocks and some will contain less.

Upon accepting a (strong) block, miners begin working on creating the next block in the chain by using the hash of the accepted block as the previous hash (Fig. 2a). When a miner finds a proof-of-work that satisfies the weak target, he broadcasts the weak block to the network. After verifying the weak block, each miner modifies the coinbase reward, appends additional transactions to the block if desired, computes the new root hash, and then continues scanning for a valid nonce (Fig. 2b). We will refer to the new information as the miner's *Δ-block* (Fig. 2f). If again a miner finds a proof-of-work that satisfies the weak target, he broadcasts the new weak block by sending only his Δ-block and the hash of the previous weak block. In this manner, miners can cooperate to build the subchain by transmitting only the new information and a fixed-byte-size reference to the subchain's tip.

When a miner finds a proof-of-work that meets the strong target (Fig. 2d), he broadcasts it in the same manner he would for a weak block (*i.e.*, by sending only his Δ-block and the hash of the previous weak block). Nodes recognize this as a valid (strong) block, retain the nonce

and coinbase transaction, and close the subchain. The process of constructing a subchain on top of this latest block begins anew (Fig. 2e).

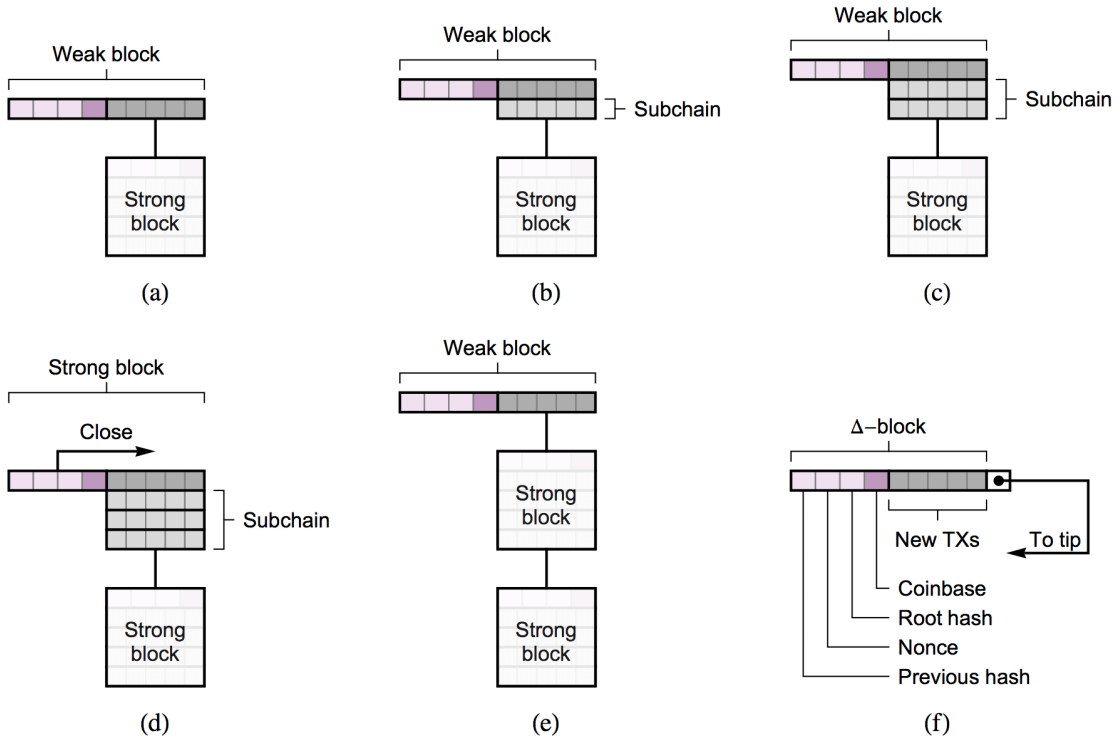

Fig. 2. Miners build subchains layer by layer (a – c), where each layer corresponds to the solution of a weak block. To propagate blocks (weak or strong), miners need only send their Δ-block and a reference to the subchain's tip (f), reducing the quantity of transmitted bytes. When a nonce that satisfies the strong target is found, the subchain is closed thereby becoming a strong block (d), and miners begin working on a new subchain (e).

If more than a single subchain exists, miners build off the *longest* subchain. In cases where two subchains of equal length exist, miners work on the one they knew about first, switching to the other if it becomes longer. For conflicting (double-spent) transactions, the transaction verified in a subchain has priority over one only in mempool. Note that this behavior represents a departure from the Satoshi protocol where miners will only replace transactions in mempool if a conflict is included in strong block (subchains extends this behavior to weak blocks too). The departure is necessary so that miners, under normal conditions, converge upon a single subchain. For the remainder of this paper, *mempool* is defined as the set transactions that have been neither confirmed in a strong block nor verified in a weak block.

4. Reduced Orphan Risk

Subchains reduce orphan risk by reducing the information propagated the moment the proof-of-work is solved. If a block takes time τ to propagate, the probability the network finds another block during the propagation interval[18] $0 < t < \tau$ is given by

$$P_{\text{orphan}} = \int_0^{\tau} \frac{1}{T} e^{-\frac{t}{T}} dt = 1 - e^{-\frac{\tau}{T}},$$

where $\frac{1}{T} e^{-\frac{t}{T}}$ is of course the probability distribution for the arrival time of a valid proof-of-work. Miners cooperate to build subchains in order to pre-propagate much of the block contents, thereby minimizing this propagation time and the chances of an orphan race.

Assuming that miners produce equal-sized Δ-blocks, each Δ-block is scaled down by the subchain factor, $\frac{T}{\Delta T}$, such that $\Delta Q = \frac{\Delta T}{T} Q$. The propagation time is thus $\tau = z\Delta Q + \tau_0$, from which it follows that

$$P_{\text{orphan}} = 1 - e^{-\frac{\tau_0}{T}} e^{-\frac{zQ\Delta T}{T^2}}.$$

This equation is plotted in Fig. 3 for various subchain factors and using recent estimates for the network propagation constants ($z = 17$ s/MB and $\tau_0 = 10$ s).[6,9,19] A subchain with $\frac{T}{\Delta T} = X$ would permit approximately X times more transactions per second at the same level of orphaning risk as without the subchain. The minimum useful subchain verification time is limited, however, because the network cannot come to consensus regarding the subchain faster than the network's latency (which, regardless of technology advancements, is limited by the product of the network diameter[20] and the speed of light to approximately 0.1 s).

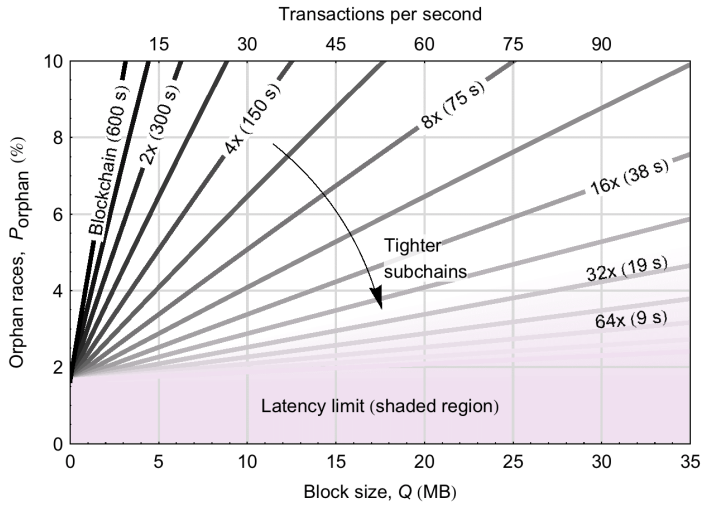

Fig. 3. Subchains help scale Bitcoin by reducing orphaning risk for larger block sizes. This chart is based on recent estimates for the network propagation constants ($z = 17$ s/MB and $\tau_0 = 10$ s).[6,9] For example, a subchain with 38-second verifications would permit approximately 16 times more transactions per second at the same level of orphaning risk as without the subchain. The minimum subchain verification time is limited, however, due to network latency (shaded region).

5. Existence of a Fee Market

With conventional block propagation, a miner must balance the additional fee revenue he earns by making his block bigger, with the decreased orphaning risk he enjoys by making his block smaller. The same economics extends to the scenario where subchains are the default

mechanism to build and propagate blocks but with one difference: a miner can include all of the subchain's transactions in his block candidate—and thus all of its fees—without incurring additional orphaning risk. The reason this is possible is because the miner can reference the entirety of the subchain with a single hash; the propagation time for that hash does not depend on the size of the subchain that the hash references. The fees in each propagated Δ-block thus add to the subchain's "pot," increasing the effective block reward as indicated by the black points in Fig. 4a at Q_1, Q_2 and Q_3.

A miner does, however, incur orphaning risk for the *new* transactions included in his Δ-block. The larger he makes his Δ-block, the slower it will propagate across the network, and—in the case where he finds a strong block—the greater the risk he incurs of having his block orphaned and losing the block reward. This risk (as a function of block size) is depicted by the superlinear curve[21] marked "supply" in Fig. 4a. We will use the symbol R to represent it. The sublinear curve marked "demand" represents the fees available from transactions in mempool; we will use the symbol F to represent it. The miner's expected profit is greatest at the block size that maximizes the difference between these two curves, which is the point where the marginal orphaning risk, dR/dQ, is equal to the marginal fee revenue, dF/dQ (*cf.* Fig. 4b).[5,22,23]

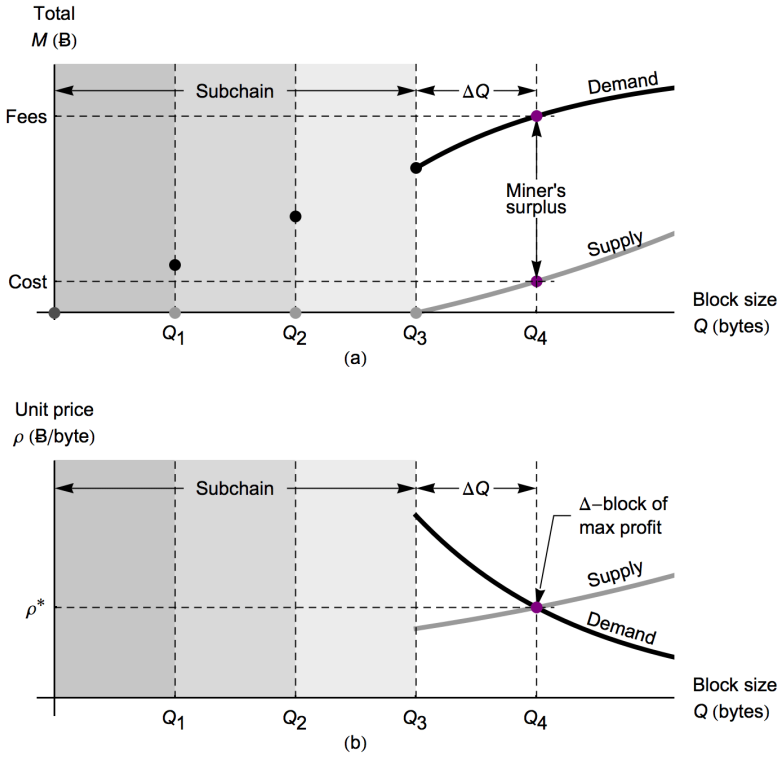

Fig. 4. The fees in a subchain increase each time a Δ-block gets added, effectively "growing the pot". Since a miner incurs no orphaning risk by including the contents of the subchain in his block candidate, he is incentivized to build his Δ-blocks on top of the highest-fee subchain. However, a miner incurs normal orphaning risk for any *new* transactions he chooses to add. Because of this, he will only include new transactions that pay more in fees per byte than the marginal cost of the additional block space.

Miners are naturally incentivized to share each Δ-block they find, as doing so reduces the orphaning risk of their candidate block.

6. Proof-of-Work Security From Fee Revenue

It is simple to show that fees contribute to proof-of-work security (in the absence of a block size limit). Fig. 5 is a modification of Fig. 4a that considers all of the miner's revenues and costs, including the block reward and electricity for hashing. In a competitive market, the profits for marginal miners will trend to zero. To reconcile this fact with Fig. 5, the total production costs for block space must increase such that the two points marked in purple move closer together. That is, if industry profits were large, miners would tend to deploy more hash power to compete for this profit, thereby shifting the entire production cost curve upwards, increasing hashing costs and decreasing profits. As shown in Fig. 5, the fee revenue is significantly greater than the orphan risk; this fee revenue—captured Δ-block-by-Δ-block in the subchain—acts no differently than an increase in the block reward would: it serves to increase the network hash rate.

One subtlety to note is that a miner with revenue and costs as depicted in Fig. 5 would not start to mine until the subchain contained sufficient fees to make the expectation value of his profit positive. Presently, fees are such a small fraction of the block reward that most miners are profitable regardless of fees. However, when total fees are no longer small compared to the block reward, we would expect the instantaneous hash rate to increase every time a new Δ-block (and its fees) is added, as miners with marginally higher electricity costs turn on their machines. Further discussion of this phenomenon is beyond the scope of this paper.

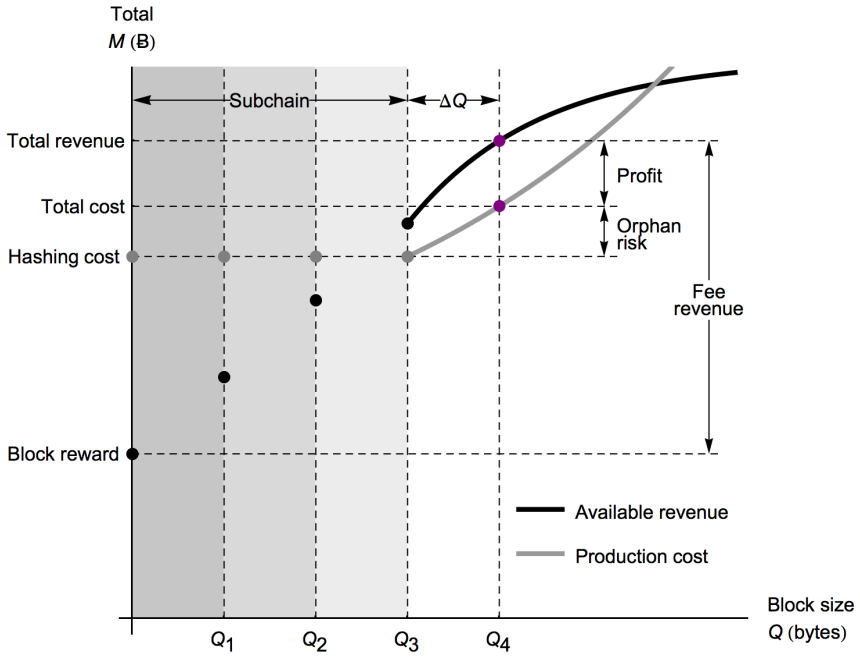

Fig. 5. If excess mining profits are available due to high fees, miners will deploy more hash power, raising network difficulty, resulting in higher hashing costs. This has the effect of shifting the production cost curve upwards, thereby reducing industry profits. Fees thus contribute directly to proof of work security. Total orphaning risk is small compared to total fee revenue, although marginal orphaning risk remains equal to marginal fee revenue.

7. Zero-Confirmation Security With Standard Blocks

With standard block propagation, it is well known that an attacker can—with probability χ—double-spend a zero-confirm transaction with negligible cost if unscrupulous miners control a fraction χ of the hash power and disobey the protocol when doing so is profitable.

The Satoshi protocol specifies that a miner must accept only the first-seen version of a transaction into his mempool. He should never replace a transaction from his mempool with a conflicting version (unless that conflicting version was included in a solved block). However, an unscrupulous miner may from time to time have the opportunity to earn a greater profit by "cheating"—replacing an earlier version of a transaction with one that pays Δf more in fees (and one that presumably reverses the payment made in the original version of the transaction). Since cheating remains profitable as $\Delta f \to 0$, it is sometimes said that the lower bound on the security of zero-confirmation transactions in Bitcoin is zero. Next we will show how the use of subchains would change this property.

8. Zero-Confirmation Security With Subchains

A similar misalignment of incentives as describe in Section 7 exists with subchains, although when subchains are used "cheating" no longer has zero cost. To see this, consider the situation where an attacker submits a double-spent version of a transaction—that pays Δf more in fees—after the original version of the transaction has received 0, 1, … , m subchain verifications and in the presence of unscrupulous miners. For what values of Δf will the group of unscrupulous miners participate in the attack? (To communicate the intuition behind this concept without becoming bogged down in math, we assume that the fraction of the hash power controlled by the unscrupulous miners is small compared to the honest miners and that the attack succeeds if the unscrupulous group finds a strong block. We leave a more rigorous treatment as future work).

Zero subchain verifications—Since aiding in the attack has no short-term cost to the miner (a miner can simple replace the first transaction with the second), the attack will succeed with a probability χ for all values of $\Delta f > 0$. This case is equivalent to that of a standard unconfirmed transaction.

One subchain verification—The original transaction has *increased* security compared to the previous case. To aid in the attack, the group of unscrupulous miners must build off an earlier Δ-block (otherwise their blocks would be invalid). A short-term greedy miner will not take part in such an attack unless the double-spent transaction pays more in additional fees than the marginal orphaning risk of the pre-propagated transactions he forfeits: $\Delta f > R(\Delta Q_n)$ were ΔQ_n is the size in bytes of the Δ-block containing the transaction the miner is attempting to double spend. Since R is superlinear in Q, this represents a lower bound (*i.e.*, the actual fee required to entice a short-term greedy miner may be greater).

Arbitrary subchain verifications—Building on the logic from the previous case, each Δ-block added to the subchain reduces the unscrupulous miners' expected profit by an amount equal to or greater than the sum of each forfeited Δ-block's orphaning risk. Making the approximation that the fees in each Δ-block are equal to the Δ-block's marginal orphaning risk, participating in the double-spend attack costs the miner an amount on the order of the total fees that have accumulated in the subchain above the transaction in question. Denoting ΔF_i as the fees in *i*th Δ-block, the inequality

$$\Delta f > R(\Delta Q_i) + \cdots + R(\Delta Q_{i+m}) \approx \sum_{i=m}^{n+m} \Delta F_i$$

thus provides a lower bound on the price that will entice an unscrupulous miner to participate in the attack as a function of the number of subchain verifications m. The probability of success is simply the fraction of the hash power that behaves unscrupulously.

9. Nested Subchains

Miners must agree on the weak target in order to cooperate building subchains. At first glance, it appears there is a trade-off: too strong a target leads to higher orphan risk and slower subchain verifications while too weak a target may result in weak blocks found at such a high rate that convergence on a single subchain is not possible. One possible solution is to use nested subchains (Fig. 7).

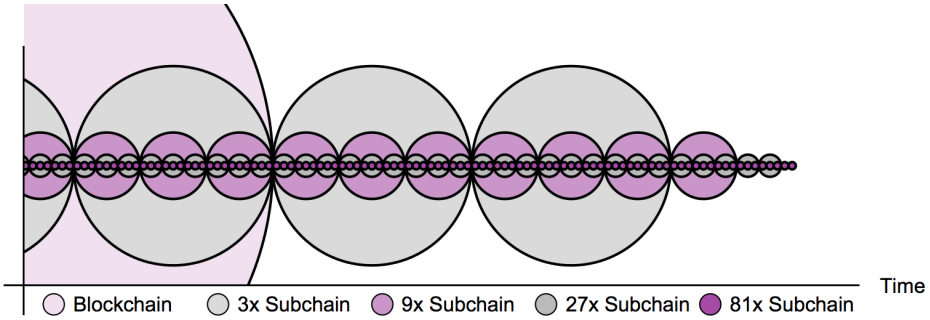

Fig. 7. Subchains can be nested to avoid the tradeoff between verification time and security. This image shows trinary nesting of Δ-blocks (represented by the circles).

A nested subchain is simply a subchain within a subchain. When a miner finds a block that satisfies the subchain difficulty, the deeper subchain is closed and a new subchain at the deeper level begins. Miners build from the highest-fee subchain at a given nesting depth but ignore higher-fee subchains at deeper nestings. With nesting, the subchain verification time can be reduced as miner connectivity improves, fundamentally limited only by network latency due to the time is takes light to travel across the network.

10. Conclusion

We presented a scaling technique called subchains to reduce orphaning risk for large blocks and improve the security of zero-confirmation transactions. Subchains are formed as a series of weak blocks, with the next weak block building a new layer of transactions (what we called a Δ-block) upon the previous weak block. Miners transmit blocks (both weak and strong) by sending only the latest Δ-block and a fixed byte-size reference to the subchain's tip.

Miners cooperate to extend a single subchain in order to maximize that subchain's total fees, as those fees can be included in each miner's candidate block without incurring orphaning risk. Interestingly, subchains only have a small effect on the marginal transaction cost. Although the technique reduces orphan rates for large blocks considerably, it does not result in a corresponding reduction in the total fees required by miners to produce such blocks.

The result is that transaction fees pay for proof-of-work security, rather than paying for orphans.

Transactions included in a subchain are quickly secured by the transactions stacked above them, encouraging additional uses for Bitcoin that require fast verifications. A lower-bound on the cost to double-spend a transaction included in a subchain is on the order of the total fees that have accumulated in the subchain above the transaction in question. Subchains can be nested to avoid the tradeoff between verification time and convergence inherent in a non-nested subchain. As the use and complexity of subchains grow, the idea of "blocks" and "confirmations" could be abstracted away from the user—the Blockchain may one day appear as a continuous stream of transactions, where the new transactions added each moment serve to secure the ones that came before them.

Neither a hard nor soft fork is required to implement subchains; however, the technique is only useful if a significant fraction of the network hash power participates. Network-wide support for subchains would add significant transactional capacity and improve the user experience, helping to further advance the adoption of Bitcoin.

Acknowledgement

The author gratefully acknowledges the kind review and suggestions of Gavin Andresen and "awemany." He also acknowledges Gregory Maxwell for the motivation to investigate fee market dynamics in scenarios where information is pre-propagated, and he thanks the many thoughtful and intelligent individuals at the Bitcoin Forum (bitco.in) for hours of pleasant and productive discussion.

Notes and References

[1] "Total Number of Transactions" chart. *Blockchain.info* (13 December 2015) https://blockchain.info/charts/n-transactions-total

[2] "Median Transaction Confirmation Time (With Fee Only)" chart. *Blockchain.info* (13 December 2015) https://blockchain.info/charts/avg-confirmation-time

[3] "Scalability." *Bitcoin Wiki* (13 December 2015) https://en.bitcoin.it/wiki/Scalability

[4]Murdoch, S. J., Drimer, S., Anderson, R., Bond, M. "Chip and PIN is Broken." *2010 IEEE Symposium on Security and Privacy*, Oakland, California (16 May 2010) http://www.unibank.org/toposign/chip_and_pin_is_broken.pdf

[5]Rizun, P. R. "A Transaction Fee Market Exists Without a Block Size Limit." No Publisher (2015) https://dl.dropboxusercontent.com/u/43331625/feemarket.pdf

[6] Stone, G. A. "An Examination of Bitcoin Network Throughput Via Analysis of Single Transaction Blocks." No Publisher (2015) http://www.bitcoinunlimited.info/1txn

[7] Barski, C., and Wilmer, C. *Bitcoin for the Befuddled*. San Francisco: No Starch Press (2014)

[8] Carlsten, M., Kalodner, H., Narayanan, A. "Mind the Gap: Security Implications of the Evolution of Bitcoin Mining." *Scaling Bitcoin Montreal* (12 September 2015)

[9] "Bitcoin Network Capacity Analysis – Part 6: Data Propagation." *Tradeblock Blog* (23 June 2015) https://tradeblock.com/blog/bitcoin-network-capacity-analysis-part-6-data-propagation

[10] Decker C. and Wattenhofer R. "Information Propagation in the Bitcoin Network." *13th IEEE International Conference on Peer-to-Peer Computing*, Trento, Italy, September 2013

[11] Pseudonymous ("rocks"). Comment in "Gold Collapsing. Bitcoin UP." *Bitcoin Forum.* (12 November 2015) https://bitco.in/forum/threads/gold-collapsing-bitcoin-up.16/page-99#post-3585

[12] Andresen, G. "[Bitcoin-development] Weak block thoughts…" *Bitcoin-development* (23 September 2015) http://lists.linuxfoundation.org/pipermail/bitcoin-dev/2015-September/011157.html

[13] Pseudonymous ("TierNolan"). "Decoupling transactions and POW." *Bitcointalk* (18 April 2013) https://bitcointalk.org/index.php?topic=179598.0

[14] Andresen, G., Comment in "Faster blocks vs bigger blocks." *Bitcointalk* (3 July 2014) https://bitcointalk.org/index.php?topic=673415.msg7658481#msg7658481

[15] Rosenbaum, K., Russell, R. "IBLT and Weak Block Propagation Performance." *Scaling Bitcoin Hong Kong* (6 December 2015)

[16] Maxwell, G. "[Bitcoin-development] Block Size Increase." *Bitoin-development* 7 May 2015 (accessed 13 December 2015) https://lists.linuxfoundation.org/pipermail/bitcoin-dev/2015-May/007880.html

[17] BitFury Group. "Proof of Stake versus Proof of Work." No Publisher (13 September 2015) http://bitfury.com/content/4-white-papers-research/pos-vs-pow-1.0.2.pdf

[18] Andresen, G. "Back-of-the-envelope calculations for marginal cost of transactions." No Publisher (2013) https://gist.github.com/gavinandresen/5044482.

[19] These estimates are probably conservative (*i.e.*, the latency and propagation impedance are both likely smaller) as the methodology used by Stone includes other effects such as the time to construct a new block candidate from mempool, and the methodology used by Tradeblock measured propagation to nodes rather than to hash power.

[20] Pseudonymous ("awemany"). Comment in "Block Space as a Commodity." *Bitcoin Forum* (26 September 2015) https://bitco.in/forum/threads/block-space-as-a-commodity-a-transaction-fee-market-exists-without-a-block-size-limit.58/page-4#post-1409

[21] The derivative of this curve is a monotonically-increasing function of Q via the law of supply, thus this curve is necessarily superlinear in Q.

[22] Pinna, D. "On the Nature of Miner Advantages in Uncapped Block Size Fee Markets." No Publisher (2015) http://www.scribd.com/doc/276849939/On-the-Nature-of-Miner-Advantages-in-Uncapped-Block-Size-Fee-Markets

[23] BitFury Group. "Incentive Mechanisms for Securing the Bitcoin Blockchain." No Publisher (2015) http://bitfury.com/content/4-white-papers-research/bitfury-incentive_mechanisms_for_securing_the_bitcoin_blockchain-1.pdf