Close Latency–Security Trade-off for the Nakamoto Consensus

Jing Li, Dongning Guo, and Ling Ren
JingLi2015@u.northwestern.edu, dGuo@northwestern.edu, renling@illinois.edu

***Abstract*—Bitcoin is a peer-to-peer electronic cash system invented by Nakamoto in 2008. While it has attracted much research interest, its exact latency and security guarantees have not been rigorously established. Previous analyses of Bitcoin either focus on specific attacks or provide asymptotic bounds that are too loose for practical use. This paper describes a continuous-time model for blockchains and develops a rigorous analysis that yields very close latency (or confirmation time) and security bounds. For example, when the adversary controls 10% of the total mining power and the block propagation delays are within 10 seconds, a Bitcoin block is secured with less than 10^{-3} error probability after 5 hours 20 minutes of confirmation time, or with less than 10^{-10} error probability after 12 hours 15 minutes. These confirmation times are close to lower bounds due to a simple private attack. To establish the tight results, the mining of some special blocks are shown to be renewal processes. Moment generation functions of the inter-arrival times of those processes are derived in closed form. The general results are used to study the latency–security trade-off of several well-known proof-of-work longest-chain cryptocurrencies. Guidance is also provided on how to set parameters for different purposes.**

I. Introduction

Bitcoin was invented by Nakamoto [1] in 2008 as a peer-to-peer electronic cash system. It builds a distributed ledger commonly referred to as a blockchain. The Bitcoin blockchain is a growing sequence of transaction-recording blocks which begins with a genesis block, and chains every subsequent block to a parent block using a cryptographic hash. Producing a new block requires proof-of-work *mining*: a nonce must be included such that the block's hash value satisfies a difficulty requirement. An honest miner follows the longest-chain rule, i.e., it always tries to mine a block at the maximum height. Miners form a peer-to-peer network to inform each other of newly mined or received blocks.

A block (and the transactions within) cannot be immediately confirmed upon its inclusion in the blockchain due to a phenomenon called *forking*. Different blocks may be mined and published at around the same time, so different honest miners may extend different blockchains. Forking may also occur as a result of adversarial miners deviating from the longest-chain rule. Forking presents opportunities for double spending, which may happen if a transaction buried in a longest fork at one time is not included in another fork that later overtakes the former fork. In fact, unlike classic Byzantine fault tolerant protocols, the Bitcoin protocol only admits probabilistic guarantees. The latency (or confirmation time) of a block in Nakamoto-style consensus protocols (Nakamoto consensus for short) depends on the desired security level. The goal of this work is to obtain tight bounds on the latency of the Nakamoto consensus as a function of the target security level.

While the Nakamoto consensus protocol is simple and elegant, a rigorous analysis for the latency–security trade-off is very challenging. The original Bitcoin white paper [1] only analyzed a single specific attack, called *private mining attack*, which is to mine an adversarial fork in private. Nakamoto showed that the probability the adversary's private fork overtakes the main blockchain vanishes exponentially with the latency.

It is not until six years later that Garay et al. [2] provided the first proof that the Nakamoto consensus is secure against all possible attacks. One major limitation of [2] is that their *round-based lock-step synchrony* model essentially abstracts away block propagation delays. Several follow-up works [3]–[6] have extended the analysis to the Δ-synchrony model in which the rounds in which different honest miners observe the same block may differ by up to a known upper bound Δ.

So far, existing analyses against all possible attacks [2]–[7] (including a few concurrent and follow-up works [8]–[11]) focus on establishing asymptotic trends using the big $O(\cdot)$ or big $\Omega(\cdot)$ notation. These results are not concrete bounds for any given security level because the latency implied could be hours, days, or even years depending on the unknown constants. If one works out the constants in these asymptotic results, the latency upper bounds will be several orders of magnitude higher than the best known lower bounds [12], [13]. Thus, despite their theoretical value, existing analyses of the Nakamoto consensus provide little guidance on the actual confirmation time, security guarantees, or parameter selection in practice.

In this paper, we explicitly and closely characterize the trade-off between latency and security for Nakamoto-style proof-of-work consensus protocols. The latency results we prove are within a few hours to simple lower bounds due to the private attack. The gap remains relatively constant at different security levels, and is hence insignificant for high security levels but can be significant at low security levels. For example, with a 10% adversary mining power, a mining rate of one block every 10 minutes, and a maximum block propagation delay of 10 seconds, a block in the Nakamoto consensus is secured with 10^{-3} error probability after 5 hours 20 minutes, or with 10^{-10} error probability after 12 hours 15 minutes. As a reference, due to the private attack, one must wait for at least 1 hour 30 minutes or 8 hours 5 minutes before confirming for 10^{-3} and 10^{-10} security levels, respectively. In contrast, the best bounds prior to this work put the latency guarantees at thousands of hours or more under the same settings [6], [12], [13].

Since Bitcoin's rise to fame, numerous altcoins and Bitcoin hard forks have adopted the Nakamoto consensus protocol with very different parameters. Those parameters are mostly determined in an ad-hoc or empirical manner. This paper provides theoretical and quantitative tools to reason about the effects and trade-offs of these parameters on different metrics in the Nakamoto consensus including confirmation time, throughput, and fault tolerance. We use these new tools to analyze and compare various altcoins and offer new insights and recommendations for setting parameters.

Some new techniques developed in this paper may be of independent interests. Assuming all block propagation delays are under Δ units of time, we show the arrivals of several species of honest blocks form renewal processes. That is, the inter-arrival times of such a process are independent and identically distributed (i.i.d.). We show that the adversary must match the so-called double-laggers in order to succeed in any attack. We derive the moment generating functions of the inter-arrival times. This allows us to calculate quite accurately the probability that more double-laggers are mined than adversarial blocks in any time interval, which leads to a close latency–security trade-off. As aforementioned, prior work based on Chernoff-type bounds can be hundreds of times looser in terms of the latency guarantee.

We note that several existing proofs in the literature are flawed. A recurrent subtle mistake is to presume memorylessness of the mining process over a time interval defined according to some miners' views and actions. The boundaries of such an interval are in fact complicated *stopping times*. In general, when conditioned on a stopping time, the mining processes are no longer distributed as the original ones without conditioning. In this paper, we carefully circumvent this issue to develop a fully rigorous analysis.

Our main contributions include: 1) We provide an explicit formula for the security guarantee as a function of the latency. This is equivalent to an upper bound on the latency that guarantees any desired security level. 2) By means of numerical analysis, the latency upper bound is shown to be close to a lower bound due to the private attack. 3) We quantify how the block propagation delay bound, mining rates, and other parameters affect the latency–security trade-off. 4) We analyze and compare the performance and security of several prominent proof-of-work longest-chain protocols.

The remainder of this paper is organized as follows. Section II reviews the Nakamoto consensus and describes a continuous-time model for it. Section III presents the main theorems about the latency–security trade-off and a brief discussion. Section IV develops a technical proof for the latency–security guarantee. Section V calculates a bound due to the private attack. Section VI is devoted to numerical results and discussions. Section VII concludes.

II. Preliminary and Model

In this section, we first give an overview of the Nakamoto consensus protocol in Section II-A and then formally describe how we model it in Section II-B.

A. Review of the Nakamoto Consensus Protocol

The Nakamoto consensus centers around the proof-of-work (PoW) mechanism and the "longest-chain-win" rule. The gist of the protocol can be described succinctly: At any point in time, every honest miner attempts to mine a new block that extends the longest blockchain in that miner's view; once a new block is mined or received, its miner publishes it through the peer-to-peer mining network.

If an honest miner mines a block at time t, the block must extend the miner's longest blockchain immediately before t. The honest miner will also immediately publish the block through a peer-to-peer network. Under the Δ-synchrony model, where Δ is a known upper bound on network delay, all other miners will receive the block by time $t + \Delta$. Note that Δ upper bounds the end-to-end delay between every pair of miners regardless of the number of hops between them.

In contrast, if an adversarial miner mines a block at time t, the block may extend any blockchain mined by time t and may be presented to individual honest miners at any time after t, subject to the delay bound.

B. Formal Model

Definition 1 (Block and mining). *A genesis block, also referred to as block 0, is mined at time 0. Subsequent blocks are referred to as block 1, block 2, and so on, in the order they are mined. Let T_b denote the time when block b is mined.*[1]

Definition 2 (Blockchain and height). *Every block has a unique parent block that is mined strictly before it. We use $f_b \in \{0, 1, ..., b-1\}$ to denote block b's parent block number. The sequence $b_0, b_1, \dots, b_n$ defines a blockchain if $b_0 = 0$ and $f_{b_i} = b_{i-1}$ for $i = 1, \dots, n$. It is also referred to as blockchain b_n since b_n uniquely identifies it. The height of both block b_i and blockchain b_i is said to be i.*

Throughout this paper, "by time t" means "during $(0, t]$".

Definition 3 (A miner's view). *A miner's view at time t is a subset of all blocks mined by time t. A miner's view can only increase over time. A block is in its own miner's view from the time it is mined.*

Definition 4 (A miner's longest blockchain). *A blockchain is in a miner's view at time t if all blocks of the blockchain are in the miner's view at time t. A miner's longest blockchain at time t is a blockchain with the maximum height in the miner's view at time t. Ties are broken in an arbitrary manner.*[2]

[1] A block in a practical blockchain system is a data structure that contains a unique identifier, a reference to its parent block, and some application-level data. The block number and mining time are tools in our analysis and are not included in the block. The probability that two blocks are mined at the same time is zero in the continuous-time model. Nevertheless, for mathematical rigor, we can break ties in some deterministic manner even if they occur.

[2] The Bitcoin protocol favors the earliest to enter the view.

Definition 5 (Honest and adversarial miners)**.** *Each miner is either honest or adversarial. A block is said to be honest (resp. adversarial) if it is mined by an honest (resp. adversarial) miner. An honest block mined at time t must extend its miner's longest blockchain immediately before t.*

We assume all block propagation delays are upper bounded by Δ units of time in the following sense.

Definition 6 (Block propagation delay bound Δ)**.** *If a block is in any honest miner's view by time t, then it is in all miners' views by time $t + \Delta$.*

The adversary is allowed to use arbitrary strategy subject to the preceding constraints. Specifically, an adversary can choose to extend any existing blockchain. Once an adversarial block is mined, its miner can determine when it enters each individual honest miner's view subject to the delay bound Δ (Definition 6).

This treatment cannot be fully rigorous without a well-defined probability space. At first it appears to be intricate to fully described blockchains and its probability space. One option (adopted in [10]) is to define blockchains as branches of a random tree that depend on the adversary's strategies as well as the network topology and delays. Other authors include in their probability space the random hashing outcomes, which also depend on the adversary strategies. For our purposes it turns out to be sufficient (and most convenient) to include no more than the mining times of the honest and adversarial blocks in the probability space. Under a typical event in this probability space, blockchain consistency is guaranteed under all adversarial strategies and network schedules (under Δ-synchrony). Thus, the adversary's strategies and the network schedules do not have to be included in the probability space.

Definition 7 (Mining processes)**.** *Let H_t (resp. A_t) denote the total number of honest (resp. adversarial) blocks mined during $(0, t]$. We assume $(H_t, t \geq 0)$ and $(A_t, t \geq 0)$ to be independent homogeneous Poisson point processes with rate α and β, respectively. The total mining rate of honest (resp. adversarial) miners is thus α (resp. β) blocks per unit of time.*

Note that the adversary can in principle regulate their mining effort at will (i.e., mine at reduced rates). But such strategies can be modeled as discarding selected adversarial blocks. Hence, Definition 7 is without loss of generality.

In lieu of specifying the number of honest and adversarial miners, the proposed model only defines their respective aggregate mining rates (they remain constant in time). This is in the same spirit as the permissionless nature of the Nakamoto consensus.

III. Main Results

Theorem 8 (Latency–security upper bound)**.** *Let α and β denote the total mining rates (in blocks per unit time) of all honest and adversarial miners, respectively. Let network delays be upper bounded by Δ units of time. Suppose*

$$\beta < \alpha e^{-2\alpha\Delta}. \tag{1}$$

For every $s > 0$, barring an event $E_{s,t}$ with probability $P(E_{s,t}) \leq e^{-\Theta(t)}$, a block that is mined by time s and included in some honest miner's longest blockchain at time $s+t$ is also included in all honest miners' longest blockchains at all later times. To be precise, let

$$\psi(v) = 1 - \frac{\beta(\alpha - v)}{v^2 - \alpha v - \alpha v e^{(v-\alpha)\Delta} + \alpha^2 e^{2(v-\alpha)\Delta}}. \tag{2}$$

Let θ be the smallest positive number that satisfies $\psi(\theta) = 0$. Then $P(E_{s,t})$ is upper bounded by

$$\min_{v \in (0,\theta)} \left(1 + \frac{v + \beta - v\psi^2(v)}{\beta\psi(v)}\right)^2 (1 + \psi(v))^{\frac{2}{\psi(v)}} e^{2\Delta v - \psi(v) v t}. \tag{3}$$

As the central result in this paper, Theorem 8 upper bounds the probability of consistency violation under a given confirmation time. Equivalently, it upper bounds the required confirmation time under a desired security level. In more detail, the theorem states that, at any time, in an honest miner's longest blockchain, all blocks that are received t units of time earlier (hence, mined at least that much earlier) will always remain in all miners' longest blockchains in the future, except for a probability exponentially small in t. Unlike existing asymptotic results, Theorem 8 gives a concrete latency–security upper bound.

To understand the tightness of the result, we compare it to a lower bound due to the following private attack.

Definition 9 (Private attack)**.** *A private attack strategy on block b, denoted as ζ_b, is described as follows:*

- *As soon as b is mined, the adversary starts to mine a private adversarial blockchain that extends block f_b; and*
- *Starting from (including) block b, every newly mined honest block enters all other honest miners' view exactly Δ units of time thereafter.*

As alluded to in Section II, the adversary is given the advantage of manipulating block propagation times subject to the delay bound (Definition 6). We say the private attack on block b is successful at time t if the privately mined adversarial blockchain at time t is at least as long as one t-credible blockchain (so that the adversarial blockchain can be published to make some honest miners reverse block b). We derive the following bound as a function of the confirmation time and the maximum block propagation delay.

Theorem 10 (Latency–security lower bound)**.** *Given $s > 0$ and $t > \Delta$, under an event $B_{s,t}$, a block b mined at time s that is included in an honest miner's longest blockchain at time $s + t$ will not be in some honest miner's longest blockchains at some later time under the private attack ζ_b, where*

$$P(B_{s,t}) = \sum_{i=1}^{\infty} e^{-\beta t} \frac{(\beta t)^i}{i!} \left(1 - F\left(\frac{t}{\Delta} - i - 1, i, \alpha\Delta\right)\right) \tag{4}$$

where $F(\cdot, n, a)$ denotes the cumulative distribution function (cdf) of the Erlang distribution with shape parameter n and rate parameter a.

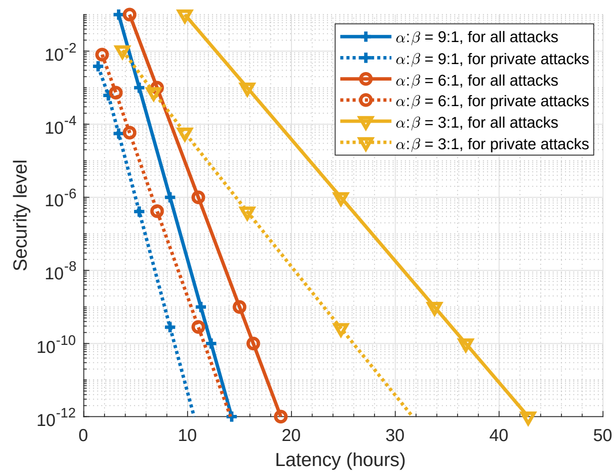

Fig. 1. Bitcoin's latency–security trade-off with $\alpha + \beta = 1/600$ blocks per second and $\Delta = 10$ seconds.

Remarks. First, we note that most previous analyses on the Nakamoto consensus assume a finite lifespan of the protocol [2], [6], that is, a maximum round number is defined, at which round the protocol terminates. The probability of consistency depends on the maximum round number.

In contrast, this paper does not assume a finite lifespan. Theorem 8 states that, barring a small probability event, a confirmed block remains permanently in all miners' longest blockchains into the arbitrary future.

Second, for technical convenience, we regard a block in a miner's longest blockchain to be confirmed after a certain amount of *time* elapses since the block is mined or enters the miner's view. Nakamoto [1] originally proposed confirming a block after it is sufficiently *deep* in an honest miner's longest blockchain. We believe both confirmation rules are easy to use in practice. But the two confirmation rules imply each other in probability (see Appendix A for further discussion).

Numerical results. The latency–security trade-off under several different sets of parameters is plotted in Figure 1. The mining rate is set to Bitcoin's one block per 600 seconds, or $\alpha + \beta = 1/600$ blocks/second. The propagation delay bound is assumed to be $\Delta = 10$ seconds. The latency upper (resp. lower) bounds are computed using Theorem 8 (resp. Theorem 10). In Figure 1, all bounds appear to be exponential in latency (this is also rigorously established by Theorem 8.)

It is instructive to examine concrete data points in Figure 1: If the adversarial share of the total network mining rate is 10%, then a confirmation time of 5 hours 20 minutes is sufficient to achieve 10^{-3} security level, and 12 hours 15 minutes achieve 10^{-10} security level. These results are within 4 hours of the corresponding lower bounds due to the private attack. If the adversarial share of the mining rate increases to 25%, then 16 hours 20 minutes and 37 hours 20 minutes of confirmation times achieve 10^{-3} and 10^{-10} security levels, respectively, and the gap between the upper and lower bounds is about 12 hours. The gap is essentially constant under at security levels (a pair of corresponding curves are almost parallel in the figure). The gap is relatively insignificant at high security levels but can be significant at low security levels.

IV. Proof of Consistency

A. Overview

The proof of consistency is done in two main steps: In Section IV-B, we identify a typical event, which is a sufficient condition for a block to be *permanent* (or irreversible) once confirmed. In Section IV-D, we upper bound the probability that the typical event does not occur. Combining the two yields an upper bound on the probability of consistency violation.

In slightly more detail, the typical event involves a special type of honest blocks called *loners*. A loner is an honest block that is not mined within Δ units of time of other honest blocks. Let b be a block mined at time s and included in some honest miner's longest blockchain at time $s+t$. Section IV-B proves that, if for all $a \le s$ and $b \ge s+t$, more loners than adversarial blocks are mined in the time interval $(a, b]$ (this is the typical event), then block b is permanent.

Section IV-D upper bounds the probability that the typical event does not occur. The probability of distribution of loners is difficult to analyze. We define another specie of honest blocks called *double-laggers*. Double laggers have one-to-one correspondence with loners but are easier to analyze since they can be shown to form a renewal process. We derive the moment generating function of double-laggers' inter-arrival times. This allows to tightly bound the probability of the typical event.

For convenience and better intuition, we specifically choose the time unit to be equal to the block propagation delay bound in this proof. Hence Δ units of time in Theorem 8 becomes one (new) unit of time here. This obviously normalizes the block propagation delay bound to 1 under the new unit. Consequently, the mining rate, aka the expected number of blocks mined *per new unit of time*, is equal to the expected number of blocks mined per maximum delay. With slight abuse of notion, we still use α and β as the mining rates under the new time unit. At the end of the analysis we will recover Theorem 8 with an arbitrary time unit.

B. Consistency under the Typical Event

Definition 11 (Publication). *A block is said to be published by time t if it is included in at least one honest miner's view by time t. A blockchain is said to be published by time t if all of its blocks are published by time t.*

Definition 12 (t-credibility). *A blockchain is said to be t-*credible *if it is published by time t and its height is no less than the height of any blockchain published by time $t-1$. If there is no need to specify t explicitly, the blockchain is simply said to be credible (in context).*

Once a block is published, it takes no more than 1 unit of time to propagate to all miners. Hence at time t, an honest miner must have seen all blockchains published by $t-1$. It

α	collective honest mining rate
β	collective adversarial mining rate
Δ	upper bound on network delay
f_b	the block number of block b's parent
T_b	the time block b is mined
$A_{s,t}$	total number of adversarial blocks mined during time $(s,t]$
$H_{s,t}$	total number of honest blocks mined during time interval $(s,t]$
$X_{s,t}$	total number of laggers mined during time interval $(s,t]$
$Y_{s,t}$	total number of loners mined during time interval $(s,t]$
$V_{s,t}$	total number of double-laggers mined during time interval $(s,t]$

TABLE I
NOTATIONS

follows that every honest miner's longest blockchain must be t-credible. As we shall see, it is unnecessary to keep tabs of individual miner's views as far as the fundamental security is concerned. Focusing on credible blockchains allows us to develop a simple rigorous proof with minimal notation.

There can be multiple t-credible blockchains, which may or may not be of the same height.

Definition 13 (Lagger)**.** *An honest block mined at time t is called a* lagger *if it is the only honest block mined during $[t-1,t]$. By convention, the genesis block is honest and is regarded as the 0-th lagger.*

Definition 14 (Loner)**.** *An honest block mined at time t is called a* loner *if it is the only honest block mined during $[t-1,t+1]$.*

Lemma 15. *A loner is the only honest block at its height.*

Proof. Suppose block b mined at time t is a loner. By definition, no other honest block is mined during $[t-1,t+1]$. By Definitions 3 and 6, block b is in all honest miners' views by time $t+1$. Thus, all honest blocks mined after $t+1$ must have heights at least $h(b)+1$. Similarly, if an honest block is mined before $t-1$, its height must be smaller than $h(b)$; otherwise, block b's height would be at least $h(b)+1$. Hence, no other honest block is at the same height as block b. $\square$

Suppose $0 \le s < t$. Let $H_{s,t} = H_t - H_s$ denote the total number of honest blocks mined during time interval $(s,t]$. Let $X_{s,t}$ denote the total number of laggers mined during $(s,t]$. Let $Y_{s,t}$ denote the total number of loners mined during $(s,t]$. Let $A_{s,t}$ denote the total number of adversarial blocks mined during $(s,t]$. By convention, $H_{s,t} = X_{s,t} = Y_{s,t} = A_{s,t} = 0$ for all $s \ge t$. Table I illustrates frequently used notations.

Lemma 16. *Suppose $t \le r$. Let s denote the mining time of the highest honest block shared by a t-credible blockchain and an r-credible blockchain. Then*

$$Y_{s+1,t-1} \le A_{s,r}. \tag{5}$$

Proof. Let block e denote the highest honest block shared by r-credible blockchain d and t-credible blockchain d' with $T_e = s$. Let block b denote the highest block shared by blockchains d and d'. Blocks b and e may or may not be the same block. The relationship between these blocks is illustrated as follows:

$$\begin{array}{ccccccc} & & \boxed{} & - \cdots - \cdots - & \boxed{\text{d}} & & \\ & & | & & \text{time } r & & \\ \boxed{\text{e}} & - \cdots - & \boxed{\text{b}} & - \cdots - & \boxed{\text{d'}} & & \\ \text{time } T_e = s & & & & \text{time } t & & \end{array} \tag{6}$$

If $t - s \le 2$ or no loner is mined during $(s+1, t-1]$, obviously $Y_{s+1,t-1} = 0 \le A_{s,r}$. Otherwise, consider loner c mined during $(s+1, t-1]$. We next show that c can be paired with an adversary block mined during $(s,r]$.

Since blockchain e is s-credible and block c is mined after time $s+1$, we have $h(c) \ge h(e)$. Since blockchain d is r-credible and blockchain d' is t-credible, we have $h(c) \le \min\{h(d), h(d')\}$. Consider the following two only possible cases:

1) If $h(e) < h(c) \le h(b)$, there exists at least one adversarial block at height $h(c)$ because all blocks between block e (exclusive) and block b (inclusive) are adversarial by definition.
2) If $h(b) < h(c) \le \min\{h(d), h(d')\}$, there is at least one adversarial block at height $h(c)$, because two diverging blockchains exist but loner c is the only honest block at its height by Lemma 15.

Thus, for every loner mined during $(s+1, t-1]$, at least one adversarial block must be mined during $(s,r]$ at the same height. In particular, the adversarial block must be mined before r because it is published by r. Hence (5) must hold. $\square$

We now define some "typical events" alluded to at the beginning of this section.

Definition 17. *For all $s, t \ge 0$ and $\epsilon \in (0,1)$, let*

$$F^{\epsilon}_{s,t} = \bigcap_{a\in[0,s],b\in[t,\infty)} \{Y_{a+1,b-1-\epsilon} > A_{a,b}\}. \tag{7}$$

We fix arbitrary $\epsilon \in (0,1)$ for now. We will send $\epsilon \to 0$ later.

Lemma 18. *Suppose block b is mined by time s and is included in a t-credible blockchain. Then, under event $F^{\epsilon}_{s,t}$, block b is included in all r-credible blockchains for all $r \ge t$.*

Proof. We first establish the result for $r \in [t, t+\epsilon]$ and then prove the lemma by induction.

Fix arbitrary $r \in [t, t+\epsilon]$. Let block e denote the highest honest block shared by an r-credible blockchain and a t-credible blockchain that includes block b. We have

$$Y_{T_e+1,r-1-\epsilon} \le Y_{T_e+1,t-1} \tag{8}$$
$$\le A_{T_e,r} \tag{9}$$

where (9) is due to Lemma 16. Under $F^{\epsilon}_{s,t}$,

$$Y_{a+1,r-1-\epsilon} > A_{s,r} \tag{10}$$

holds for all $a \in [0,s]$. Hence for (9) to hold, we must have $T_e > s$. Since $s \ge T_b$ by assumption, block b must be included in blockchain e, which implies that block b must also be included in the r-credible blockchain.

Suppose the lemma holds for $r \in [t, t+n\epsilon]$ for some positive integer n. We show the lemma also holds for $r \in [r, t+(n+1)\epsilon]$ as follows: Let $t' = t + n\epsilon$. Because $F^{\epsilon}_{s,t'}$ occurs under $F^{\epsilon}_{s,t}$ and that block b is included in a t'-credible blockchain, a repetition of the $r \in [t, t+\epsilon]$ case with t replaced by t' implies that block b is included in all r-credible blockchains with $r \in [t', t'+\epsilon]$. Hence lemma holds also for $r \in [t, t+(n+1)\epsilon]$.

The lemma is then established by induction on n. □

Lemma 18 guarantees that a block with some confirmation time is permanent/irreversible under the typical event $F^{\epsilon}_{s,t}$. It remains to lower bound the probability of $F^{\epsilon}_{s,t}$ as a function of the confirmation time $t-s$.

C. Some Moment Generating Functions

We start by deriving the moment generating functions (MGF) of several types of inter-arrival times. These will be useful in the analysis of $P\left(F^{\epsilon}_{s,t}\right)$.

Lemma 19. *Let W be an exponential random variable with mean $1/\alpha$. Let the MGF of W conditioned on $W \le 1$ be denoted as $\Phi_0(u)$. Then*

$$\Phi_0(u) = \begin{cases} \frac{\alpha(1-e^{u-\alpha})}{(1-e^{-\alpha})(\alpha-u)} & \text{if } u \neq \alpha \\ \frac{\alpha}{1-e^{-\alpha}} & \text{if } u = \alpha. \end{cases} \tag{11}$$

Proof. The probability density function (pdf) of W conditioned on $W \le 1$ is simply

$$\frac{\alpha}{1-e^{-\alpha}} e^{-\alpha w} 1_{\{0<w<1\}} \tag{12}$$

where $1_{\{\cdot\}}$ represents the indicator function which takes the values of 1 or 0 depending on whether the condition in the braces holds or not. The conditional MGF is thus

$$\Phi_0(u) = \mathbb{E}\left[e^{uW} | W \le 1\right] \tag{13}$$

$$= \int_0^1 \frac{\alpha}{1-e^{-\alpha}} e^{-\alpha w} e^{uw} \mathrm{d}w \tag{14}$$

which becomes (11). □

Lemma 20. *Let W be an exponential random variable with mean $1/\alpha$. Let the MGF of W conditioned on $W > 1$ be denoted as $\Phi_1(u)$. Then*

$$\Phi_1(u) = \frac{\alpha e^u}{\alpha - u} \tag{15}$$

where the region of convergence is $u \in (-\infty, \alpha)$.

Proof. Conditioned on $W > 1$, the pdf of W is given by

$$e^{\alpha(-w+1)} 1_{\{w>1\}}. \tag{16}$$

The conditional MGF is thus:

$$\Phi_1(u) = \mathbb{E}\left[e^{uW} | W > 1\right] \tag{17}$$

$$= \int_1^{+\infty} e^{\alpha(-w+1)} e^{uw} \mathrm{d}w. \tag{18}$$

The integral converges if and only if $u < \alpha$, where the result is given by (15). □

Recall the genesis block is the 0-th lagger. For $i = 1, 2, \dots$, let X_i denote the time elapsed between the mining times of the $(i-1)$-st and the i-th lagger. Let K_i denote the number of honest blocks mined between the $(i-1)$-st lagger (excluded) and the i-th lagger (included).

Lemma 21. $(X_1, K_1), (X_2, K_2), \dots$ *are i.i.d.*

Lemma 21 is proved in Appendix B.

Double-laggers. The loner process is not easy to characterize since whether a block mined at time t is a loner depends not only on the past but also on future blocks (in $(t, t+1]$). In order to count loners, we examine a tightly-related specie of honest blocks defined as follows.

Definition 22 (Double-lagger)**.** *The first honest block mined after a loner is called a double-lagger.*

Note that a loner is also a lagger. So whenever two laggers are mined in a row, the former one is a lagger and the latter one is a double-lagger. As such, there is a one-to-one correspondence between loners and double-laggers. We prove the independence of inter-double-lagger times, and derive their MGFs, thus establishing the arrivals of double-laggers as a renewal process.

Let $V_{s,t}$ denote the total number of double-laggers mined during $(s, t]$. Let V_1 denote the time the first double-lagger arrives. Let J_1 be the number of laggers after the genesis block until the first double-lagger (included). For $i > 1$, let V_i denote the time elapsed between the $(i-1)$-st and the i-th double-lagger. Let J_i be the number of laggers between the $(i-1)$-st double-lagger to the i-th double-lagger.

Lemma 23. *For all $0 \le s \le t$,*

$$Y_{s,t} \ge V_{s,t} - 1. \tag{19}$$

Proof. Because loners and double-laggers appear in consecutive pairs, all but the first double-lagger mined during $(s, t]$ corresponds to a loner mined during $(s, t]$. □

Lemma 24. $(V_1, J_1), (V_2, J_2), \dots$ *are i.i.d.*

Lemma 24 is proved in Appendix C.

Lemma 25. *The time from a lagger to the next double-lagger follows the same distribution as the inter-double-lagger time.*

Proof. Let blocks b, c, d be consecutive honest blocks. Evidently, $T_c - T_b$ and $T_d - T_c$ are i.i.d. exponential random variables. Let Q be the time elapsed from block d to the next double-lagger. If d is a lagger, then Q does not depend on whether c is a lagger. Thus, for all x,

$$\begin{aligned} &P(Q \le x \,|\, T_d - T_c > 1) \\ &\qquad = P(Q \le x \,|\, T_d - T_c > 1, T_c - T_b > 1). \end{aligned} \tag{20}$$

The left hand side is the cdf of the time between a lagger and the next double-lagger; the right hand side is the cdf of inter-double-lagger time. Hence, the lemma is proved. □

For convenience we define the following function $h_\alpha(u)$:

$$h_\alpha(u) = u^2 - \alpha u - \alpha u e^{u-\alpha} + \alpha^2 e^{2(u-\alpha)}. \quad (21)$$

Evidently $h_\alpha(u) > 0$ if $u \le 0$ and $h_\alpha(\alpha) = 0$. Also, $h_\alpha(u)$ is differentiable with bounded derivative on $[0, \alpha]$. From now on, let u_0 denote the smallest zero of $h_\alpha(\cdot)$, i.e.,

$$h_\alpha(u_0) = 0 \quad (22)$$

and $h_\alpha(u) \neq 0$ for all $u \in [0, u_0)$. We must have $0 < u_0 \le \alpha$.

Lemma 26. *Let V denote an inter-double-lagger time. The MGF of V is*

$$\Phi(u) = 1 + \frac{\alpha u - u^2}{u^2 - \alpha u - \alpha u e^{(u-\alpha)} + \alpha^2 e^{2(u-\alpha)}} \quad (23)$$

where the region of convergence is $(-\infty, u_0)$.

Lemma 26 is proved in Appendix D. The key is to study a Markov process: The initial state is a lagger. With a known probability a double-lagger follows immediately to terminate the process. With the remaining probability we visit a non-lagger state a geometric number of times until we return to the initial lagger state. This allows us to write a recursion for the MGF of the inter-double-lagger time (aka the time till the double-lagger terminal state), the solution of which is (23).

D. Bounding the Probability of the Typical Event $F^\epsilon_{s,t}$

It is hard to directly calculate the probability of the event $F^\epsilon_{s,t}$, which is defined as the intersection of uncountably many events. To circumvent this difficulty, we lower bound the probability of a "smaller" event $G^{\epsilon,q}_{s,t}$, which is the intersection of a countable number of simple events.

Definition 27. *For all $\epsilon \in (0, 1)$, $q > 0$, and $0 \le s < t$, let*

$$G^{\epsilon,q}_{s,t} = \bigcap_{\substack{m \in \{0,1,\dots,\lceil \frac{s}{q} \rceil\},\\ n \in \{0,1,\dots\}}} \{Y_{s-mq+q+1,t+nq-q-1-\epsilon} > A_{s-mq,t+nq}\}. \quad (24)$$

Lemma 28. *For all $\epsilon \in (0, 1)$, $q > 0$, and $0 \le s < t$,*

$$G^{\epsilon,q}_{s,t} \subset F^\epsilon_{s,t}. \quad (25)$$

Proof. Without loss of generality, we assume $G^{\epsilon,q}_{s,t}$ is not empty. For every $0 \le a \le s < t \le b$, let m be the smallest integer satisfying $s - mq \le a$ and n be the smallest integer satisfying $t + nq \ge b$. Evidently $m, n \ge 0$. Then we have,

$$\begin{aligned} Y_{a+1,b-1-\epsilon} &\ge Y_{s-mq+q+1,t+nq-q-1-\epsilon} && (26)\\ &> A_{s-mq,t+nq} && (27)\\ &\ge A_{a,b} && (28)\end{aligned}$$

where (27) is because $G^{\epsilon,q}_{s,t}$ occurs. As a consequence, $F^\epsilon_{s,t}$ occurs. Hence the proof of this Lemma. ☐

With some foresight, we define

$$f(u) = e^{(2q+2+\epsilon)u}\Phi^2(u) \quad (29)$$

and

$$\Psi(u) = \frac{1}{u}(\beta + u - \beta\Phi(u)) \quad (30)$$

for $u \in [0, u_0)$. By convention, we let $\Psi(0) = 1$, so that Ψ is continuous on $[0, u_0)$.

Lemma 29. *For all $\epsilon \in (0, 1)$, $q > 0$, and $0 \le s < t$,*

$$P(Y_{s+q+1,t-q-1-\epsilon} \le A_{s,t}) \le f(u)e^{-u\Psi(u)(t-s)} \quad (31)$$

for all $u \in (0, u_0)$.

Proof. Suppose $u \in (0, u_0)$. By independence of the Y and Z processes, we have

$$\begin{aligned} &P(Y_{s+q+1,t-q-1-\epsilon} \le A_{s,t})\\ &\le \sum_{j=0}^{\infty} P(A_{s,t} = j) \cdot P(Y_{s+q+1,t-q-1-\epsilon} \le j) && (32)\\ &\le \sum_{j=0}^{\infty} P(A_{s,t} = j) \cdot P(V_{s+q+1,t-q-1-\epsilon} \le j+1) && (33)\end{aligned}$$

where (33) is due to Lemma 23.

If no more than $j+1$ double-laggers are mined during time interval $(s+q+1, t-q-1-\epsilon]$, then counting from time $s+q+1$, the $(j+2)$-nd double-lagger must be mined after time $t-q-1-\epsilon$. Due to Lemma 24, we have

$$\begin{aligned} &P(V_{s+q+1,t-q-1-\epsilon} \le j+1)\\ &\qquad \le P\left(\sum_{m=1}^{j+2} V_m \ge t-s-2q-2-\epsilon\right) \end{aligned} \quad (34)$$

where $V_1, V_2, \dots$ are i.i.d. inter-double-lagger times. Using Markov's inequality and Lemma 24, for all $u \in (0, u_0)$:

$$\begin{aligned} &P\left(\sum_{m=1}^{j+2} V_m \ge t-s-2q-2-\epsilon\right)\\ &\le \mathbb{E}\left[\exp\left(u\Big(\sum_{m=1}^{j+2} V_m - (t-s-2q-2-\epsilon)\Big)\right)\right] && (35)\\ &= e^{-u(t-s-2q-2-\epsilon)}\Phi^{j+2}(u). && (36)\end{aligned}$$

Using (33)–(36), we have for $u \in (0, u_0)$

$$\begin{aligned} &P(Y_{s+q+1,t-q-1-\epsilon} \le A_{s,t})\\ &\le \sum_{j=0}^{\infty} e^{-(t-s)\beta}\frac{((t-s)\beta)^j}{j!}e^{-u(t-s-2q-2-\epsilon)}\Phi^{j+2}(u) && (37)\\ &= e^{-(\beta+u)(t-s)+(2q+2+\epsilon)u}\Phi^2(u)\sum_{j=0}^{\infty}\frac{((t-s)\beta\Phi(u))^j}{j!} && (38)\\ &= e^{(2q+2+\epsilon)u}\Phi^2(u)e^{-(t-s)(\beta+u-\beta\Phi(u))} && (39)\end{aligned}$$

which becomes (31). ☐

Lemma 30. *If $\beta < \alpha e^{-2\alpha}$, then there exists some positive number $u^* < u_0$ such that $u\Psi(u) > 0$ for all $u \in (0, u^*]$.*

Proof. By definition, $u\Psi(0) = 0$ and the right derivative

$$
\begin{aligned}
&(u\Psi(u))'_+|_{u=0} \\
&= 1 - \beta(\Phi(u))'_+|_{u=0} &(40)\\
&= -\Bigg[\frac{-\alpha - \alpha e^{-\alpha+u} + 2u^2 e^{2(u-\alpha)} + 2u - \alpha u e^{u-\alpha}}{(\alpha^2 e^{2(u-a)} - \alpha u - \alpha u e^{u-a} + u^2)^2} \\
&\quad \cdot \beta(-\alpha u + u^2) \\
&\quad + \frac{\beta(-\alpha + 2u)}{\alpha^2 e^{2(u-a)} - \alpha u - \alpha u e^{u-a} + u^2} + 1\Bigg]\Bigg|_{u=0} &(41)\\
&= 1 - \frac{\beta}{\alpha e^{-2\alpha}}. &(42)
\end{aligned}
$$

If $\beta < \alpha e^{-2\alpha}$, we have $(u\Psi(u))'_+|_{u=0} > 0$. By continuity, there must exist a $u^* < u_0$ such that $(u\Psi(u))' > 0$ and $u\Psi(u) > 0$ for all $u \in (0, u^*]$. □

Let $u_1 > 0$ be the smallest positive number such that $\Psi(u) = 0$. Note that as $u \to u_0$, we have $\Phi(u) \to \infty$ and $u\Psi(u) \to -\infty$. By Lemma 30, u_1 exists and $u^* < u_1 < u_0$.

Lemma 31. *For all $\epsilon \in (0,1)$, $q > 0$, and $0 \le s < t$,*

$$
P\left((F^\epsilon_{s,t})^c\right) \le \min_{0<u<u_1} \left(1 + \frac{u + \beta - u\Psi^2(u)}{\beta\Psi(u)}\right)^2 \times (1+\Psi(u))^{\frac{2}{\Psi(u)}} e^{(2+\epsilon)u - \Psi(u)u(t-s)} \quad (43)
$$

where u_1 is the smallest number such that $u_1\Psi(u_1) = 0$.

Proof. Let $k = \lceil \frac{s}{q} \rceil$. By Lemmas 28 and 29 and using the union bound, we have for $u \in (0, u_1)$:

$$
\begin{aligned}
&P\left((F^\epsilon_{s,t})^c\right) \\
&\le P\left((G^{\epsilon,q}_{s,t})^c\right) &(44)\\
&\le P\left(\bigcup_{\substack{m\in\{0,1,\dots,k\},\\ n\in\{0,1,\dots\}}} \{Y_{s-mq+q+1,t+nq-q-1-\epsilon} \le A_{s-mq,t+nq}\}\right) &(45)\\
&< \sum_{\substack{m\in\{0,1,\dots,k\},\\ n\in\{0,1,\dots\}}} f(u) e^{-u\Psi(u)(t+nq-s+mq)} &(46)\\
&= f(u) e^{-u\Psi(u)(t-s)} \left(\sum_{m=0}^{k} e^{-u\Psi(u)mq}\right)\left(\sum_{n=0}^{\infty} e^{-u\Psi(u)nq}\right) &(47)\\
&< \frac{f(u)}{\left(1 - e^{-u\Psi(u)q}\right)^2} e^{-u\Psi(u)(t-s)}. &(48)
\end{aligned}
$$

For a given u, to minimize (48), we set the derivative with respect to q to zero to obtain the optimal choice:

$$
q^* = \frac{\log(1+\Psi(u))}{u\Psi(u)}. \quad (49)
$$

Note that by (29), we have

$$
\Phi(u) = 1 + \frac{u - u\Psi(u)}{\beta}. \quad (50)
$$

Setting q to q^* and plugging (29), (49), and (50) into (48), we have for $u \in (0, u_1)$

$$
\begin{aligned}
&P\left((F^\epsilon_{s,t})^c\right) \\
&\le \frac{e^{(2q^*+2+\epsilon)u}\Phi^2(u)}{\left(1 - \frac{1}{1+\Psi(u)}\right)^2} e^{-u\Psi(u)(t-s)}. &(51)\\
&= \frac{\left(1 + \frac{u - u\Psi(u)}{\beta}\right)^2}{\left(\frac{\Psi(u)}{1+\Psi(u)}\right)^2} e^{2q*} e^{-u\Psi(u)(t-s)+(2+\epsilon)u} &(52)\\
&= \left(1 + \frac{u + \beta - u\Psi^2(u)}{\beta\Psi(u)}\right)^2 (1+\Psi(u))^{\frac{2}{\Psi(u)}} e^{(2+\epsilon)u - \Psi(u)u(t-s)}. &(53)
\end{aligned}
$$

This completes the proof of Lemma 31. □

Lastly, we recover Theorem 8 for the original arbitrary time unit. In particular, the block propagation delays are bounded by Δ time units. To reintroduce Δ into the result, we let $\tau = t\Delta$, $\sigma = s\Delta$, $a = \alpha/\Delta$, $b = \beta/\Delta$, and $v = u/\Delta$. These new variables and parameters are then defined under the original time unit. We define

$$
\phi(v) = \Phi(\Delta v) \quad (54)
$$
$$
\psi(v) = \Psi(\Delta v). \quad (55)
$$

Plugging in (23) and (30), we have

$$
\begin{aligned}
\phi(v) &= 1 + \frac{av - v^2}{v^2 - av - ave^{(v-a)\Delta} + a^2 e^{2(v-a)\Delta}} &(56)\\
\psi(v) &= \frac{b + v - b\phi(v)}{v} &(57)\\
&= 1 - \frac{b\,(a - v)}{v^2 - av - ave^{(v-a)\Delta} + a^2 e^{2(v-a)\Delta}}. &(58)
\end{aligned}
$$

Suppose a block is mined at time σ and is included in a τ-credible blockchain. Applying Lemma 18, the block is included in all future credible blockchains under event $F^\epsilon_{\frac{\sigma}{\Delta},\frac{\tau}{\Delta}}$. We then apply Lemma 31 with the conversion. For $0 \le \sigma < \tau$, we define

$$
E_{\sigma,\xi} = \left(\bigcup_{\epsilon>0} F^\epsilon_{\frac{\sigma}{\Delta},\frac{\sigma+\xi}{\Delta}}\right)^c. \quad (59)
$$

Using Lemma 31 and letting $\epsilon \to 0$, an upper bound of $E_{\sigma,\tau-\sigma}$ is given by

$$
\min_{v\in(0,v_1)} \left(1 + \frac{v + b - v\psi^2(v)}{b\psi(v)}\right)^2 (1+\psi(u))^{\frac{2}{\psi(v)}} e^{2v\Delta - \psi(v)(\tau-\sigma)v} \quad (60)
$$

where v_1 is the smallest positive number such that $\psi(v_1) = 0$. Then every block mined before σ and is in a τ-credible blockchain must be included in all credible blockchains thereafter barring event $E_{\sigma,\tau-\sigma}$, whose probability is upper bounded by (60). This conclusion is equivalent to the main theorem (with minor abuse of notation we still use α and β to replace a and b, respectively, to follow some convention). Thus, Theorem 8 is proved.

V. Blockchain Liveness and Private Attack

In this section, we analyze blockchain growth and the private attack in Definition 9 to establish a minimum confirmation time needed for a given security level. The key is to compare the growth of a blockchain made of only honest blocks and a privately mined adversarial blockchain.

Definition 32 (Jumper)**.** *A block is called a jumper if it is the first honest block mined at its height.*[3]

Without loss of generality, we assume that honest miners always mine on top of the earliest longest blockchain in case multiple longest blockchains are in an honest miner's view. If an honest block is not a jumper, then once its propagation is subject to the maximum delay, it loses to some other honest block at the same height and hence does not make into any later credible blockchain even if the adversary takes no action at all. Hence, without loss of generality, we assume the adversary attacks a jumper block b which is referred to as the 0-th jumper in this section.

For $i = 1, 2, \dots$, let M_i denote the time elapsed between the $(i-1)$-st jumper and the i-th jumper. As in much of Section IV, we use Δ as the time unit for convenience. Let $M_{s,t}$ denote the number of jumpers mined during time interval $(s, t]$. For simplicity, we also assume that individual honest miners have infinitesimal mining power, so that almost surely no individual miner mines two consecutive jumpers in a row.

Lemma 33. *The inter-jumper times $M_1, M_2, \dots$ are i.i.d. and $M_i - 1$ follows an exponential distribution with mean $1/\alpha$.*

Proof. Let $b_0 = b$ and let block b_i denote the i-th jumper after block b. Until $T_{b_i} + 1$ when block b_i is in all honest miners' views, all honest miners except block b_i's miner (who is negligible) are mining at heights no higher than $h(b_i)$. From $T_{b_i} + 1$, it takes exponential time with mean $1/\alpha$ to mine the next jumper. Due to the memoryless nature of the honest mining process, $M_1, M_2, \dots$ are i.i.d. (the jumpers form a renewal process). □

Using Lemma 33, it is straightforward to establish the following result concerning the height of longest (or credible) blockchains over time.

Lemma 34 (Blockchain growth)**.** *For all $s, t \geq 0$, every honest miner's longest blockchain at time $s + t$ must be at least n higher than every honest miner's longest blockchain at time s with probability no less than*

$$F(t - 1 - n, n, \alpha). \tag{61}$$

Proof. If $t \leq 1$ then $F(t - 1 - n, n, \alpha) = 0$, so the lemma holds trivially. We assume $t > 1$. All jumpers mined during $(s, s + t - 1]$ must be in every honest miner's views by $s + t$, where the first jumper is higher than the miner's longest blockchain at time s and the last jumper is no higher than the miner's longest blockchain at time $s+t$. Hence the probability of interest is no less than $P\left(M_{s,s+t-1} \geq n\right)$. The event that n or more jumpers are mined on $(s, s + t - 1]$ is the same as that n inter-jumper times can fit in a duration of $t - 1 - s$, i.e.,

$$P(M_{s,s+t-1} \geq n) = P(M_1 + \cdots + M_n \leq t - 1) \tag{62}$$

where $M_1, \dots, M_n$ are i.i.d. inter-jumper times. This probability is equal to

$$\begin{aligned} P((M_1 - 1) + \cdots + (M_n - 1) < t - 1 - n) \\ = F(t - 1 - n, n, \alpha) \end{aligned} \tag{63}$$

where (63) is because $M_1 - 1, \dots, M_n - 1$ are i.i.d. exponential random variables whose sum has the Erlang distribution with shape parameter n and rate α. □

Lemma 35. *If the adversary performs the private attack ζ_b on a jumper b and publishes no block during $[T_b - \Delta, T_b + t]$ with $t > 0$, then the height of the i-th jumper mined during $(T_b, T_b + t)$ is $h(b) + i$. Also, the height of all $(T_b + t)$-credible blockchains is no greater than $h(b) + M_{T_b, T_b+t}$.*

Proof. By definition every jumper is higher than the one that precedes it. Beginning from block b, the jumpers mined during $[T_b, T_b + t)$ have consecutive heights because no adversarial blocks are published on those heights by $T_b + t$. The height of a $(T_b + t)$-credible blockchain is no heigher than that of the last jumper mined before $T_b + t$. Hence the proof of the lemma. □

Proof of Theorem 10. Using the normalized time unit, we convert the original condition $t > \Delta$ to $t > 1$. If block b is not a jumper, it will not be included in any honest miner's longest blockchain at $s + t$ when the adversary takes no action, so there is nothing to prove in this case.

If block b is a jumper, the adversary performs the private attack ζ_b and begins to mine blocks from height $h(b)$. The private attack is successful at time $s + t$ if the adversary mines more blocks than the number of competing jumpers. Because jumpers are subject to one unit of propagation delay, only jumpers mined until time $s + t - 1$ are competitive. Specifically, the private attack is successful under this event:

$$B_{s,t} = \{A_{s,s+t} \geq M_{s,s+t-1} + 1\}. \tag{64}$$

Again, let $M_1, M_2, \dots$ denote i.i.d. inter-jumper times. The probability of success can be lower bounded by

$$\begin{aligned} P\left(B_{s,t}\right) &= \sum_{i=1}^{\infty} P(A_{s,s+t} = i) P(M_{s,s+t-1} \leq i - 1) && (65) \\ &= \sum_{i=1}^{\infty} e^{-\beta t} \frac{(\beta t)^i}{i!} P(M_1 + \cdots + M_i > t - 1) && (66) \\ &= \sum_{i=1}^{\infty} e^{-\beta t} \frac{(\beta t)^i}{i!} (1 - F(t - i - 1, i, \alpha)). && (67) \end{aligned}$$

Once converted to the original time unit, i.e., with α, β, and t replaced by $\alpha\Delta$, $\beta\Delta$, and t/Δ, respectively, this result becomes (4). □

[3] Ties are broken in a deterministic manner.

With the preceding techniques, it is also straightforward to establish the following probabilistic bound for blockchain quality or liveness.

Lemma 36 (Blockchain liveness)**.** *Let $s > 0$ and $t > 1$. In every honest miner's longest blockchain at time $s + t$, the probability that n or more of those blocks are honest blocks mined during $(s, s+t)$ is lower bounded by*

$$\sum_{i=0}^{\infty} e^{-\beta t}\frac{(\beta t)^i}{i!}F(t-i-n-1, i+n, \alpha). \tag{68}$$

Proof. Since $M_{s,s+t-1}$ jumpers are mined during $(s, s+t-1]$ (with different heights), and at most $A_{s,s+t}$ of them are matched by adversarial blocks, the number of surplus jumper blocks lower bounds the number of honest blocks in any honest miner's longest blockchain that are mined during $(s, s+t]$. The said probability is thus lower bounded by

$$\begin{aligned} &P\left(M_{s,s+t-1} - A_{s,s+t} \geq n\right) \\ &= \sum_{i=0}^{\infty} P(A_{s,s+t} = i)P(M_{s,s+t-1} \geq i+n) \qquad (69) \\ &= \sum_{i=0}^{\infty} e^{-\beta t}\frac{(\beta t)^i}{i!}P(M_1 + \cdots + M_{i+n} \leq t-1) \qquad (70) \end{aligned}$$

which is equal to (68). □

VI. Analysis of Existing Systems

A. Methodology

Metrics. The performance metrics of a Nakamoto-style protocol include latency for a given security level, throughput, and fault tolerance (the upper limit of the fraction of adversarial mining in a secure system). This section numerically computes the trade-off between different performance metrics of popular Nakamoto-style systems (Bitcoin Cash, Ethereum, etc.) and discuss their parameter selections.

We remark that the throughput metric can be defined in a few different ways, ranging from the "best-case" throughput where the adversarial miners follow the protocol, to the "worst-case" throughput where the adversarial miners not only mine empty blocks but also use a selfish mining type of attack [2], [14] to push honest blocks out of longest blockchains. In this paper, we choose to focus on the "best-case" throughput, which is the throughput under normal operation and is perhaps what protocol designers have in mind when setting parameters.

Block propagation delay. The above metrics crucially depend on the block generation rate (or the total mining rate in the system), maximum block size, and block propagation delay. The former two are explicitly specified in the protocol. The block propagation delay, however, depends on network conditions. Block propagation delays in the Bitcoin network have been measured in [15]–[17]. Such measurements are in general lacking for other systems. It is observed in [15] that there is a linear relationship between propagation delays and block size.

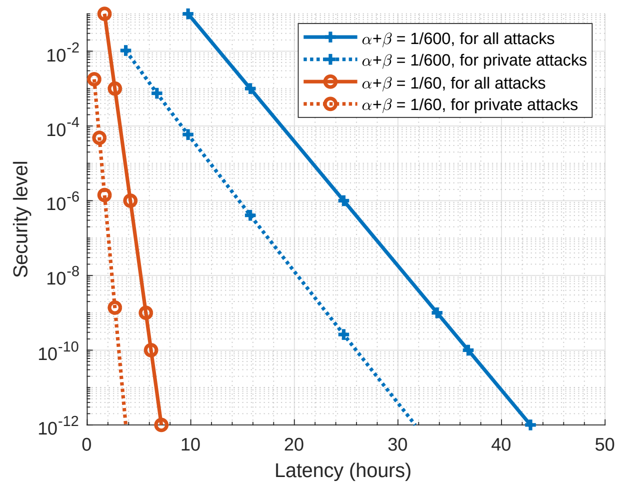

Fig. 2. Latency–security trade-off with $\Delta = 10$ seconds and 25% percentage of adversarial mining.

In this section, we let the maximum block propagation delay be determined by the block size S (in KB) according to the following formula:

$$\Delta = aS + b. \tag{71}$$

We determine the coefficients a and b using propagation delay data from Bitcoin and Ethereum monitoring websites. In Bitcoin, the block size is about 1 MB. The propagation delay of Bitcoin blocks fluctuates over the years with an overall decreasing trend [18]; the 90th percentile of block propagation is 4 seconds on average as of July 2020. Since Δ in our model needs to be an upper bound on propagation delay, we assume $\Delta = 10$ seconds for a 1 MB Bitcoin block. According to [19], the 90th percentile of Ethereum block propagation is 1.75 seconds for an average block size of 25 KB. We round it up to 2 seconds for an upper bound. Using these data points, we estimate $a = 0.008$ and $b = 1.79$.

B. Confirmation time

The latency–security trade-off has already been shown in Figure 1 for the Bitcoin protocol parameters. Figure 2 illustrates how the trade-off changes if the block generation rate increases by 10 folds (to 1 block per minute), with everything else held the same. It is not surprising to see that the latency is much shorter under the higher block generation rate in this particular case.

Figure 3 illustrates the confirmation time needed for different security levels, block generation rates, and block propagation delays. As expected, the latency is larger with longer block propagation delay and/or stronger security level requirement. Interestingly, increasing the block generation rate first reduces latency but eventually causes the latency to rise without bound. From the graph, with the Bitcoin block propagation delay around 10 seconds, a sweet spot for block generation rate is between 40 and 100 blocks per hour in terms of optimizing latency.

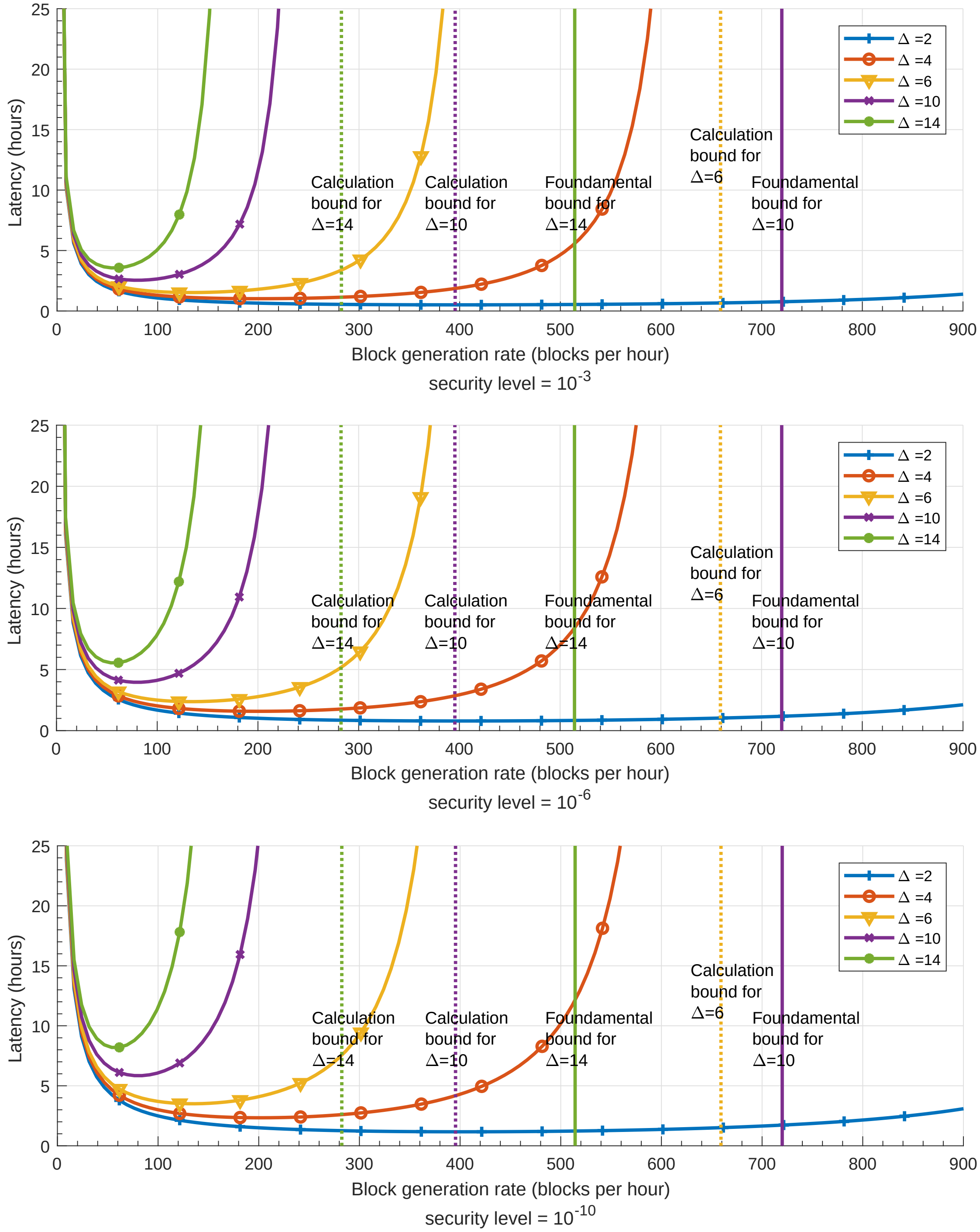

Fig. 3. Latency required for different propagation delays. The percentage of adversarial mining is 25%.

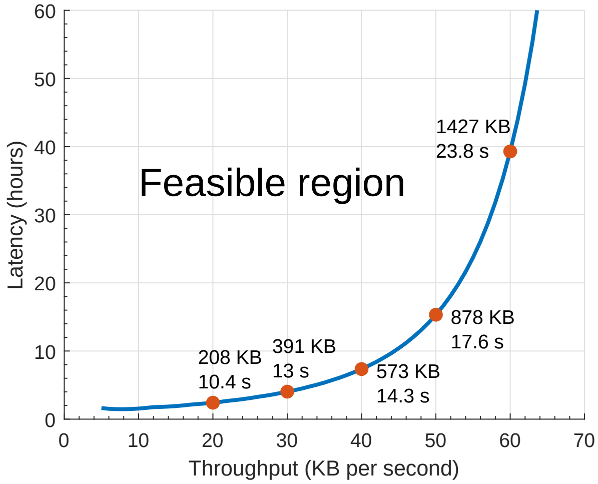

Fig. 4. Feasible latency for different throughput. The security level is 10^{-10}. The percentage of adversarial mining power is 25%.

We recall that Theorem 8 requires the honest-to-adversarial mining ratio to be bounded by

$$\frac{\beta}{\alpha} < e^{-2\alpha\Delta}. \tag{72}$$

This is because $\alpha e^{-2\alpha\Delta}$ is the exact rate that loners are mined. Beyond the ratio in (72), this paper provides no security guarantee. This is marked as the "calculation bound" in Figure 3. Note that (72) is a sufficient but not necessary condition for the consistency of the Nakamoto consensus. The sufficient and necessary condition is given in [10], [11]:

$$\frac{\beta}{\alpha} < \frac{1}{1+\alpha\Delta}. \tag{73}$$

This is marked as the fundamental bounds in Figure 3.

C. The latency–throughput trade-off

A larger block size may benefit throughput by carrying more transactions. On the other hand, the larger block size increases the propagation delay, which causes longer latency. A protocol designer may want to find a sweet spot that leads to the most desirable latency and throughput.

Figure 4 illustrates the minimum latency required to achieve given throughput according to Theorem 8 (it is actually a latency upper bound). For several target throughput numbers (20, 30, 40, 50, and 60 KB per second), we also mark the corresponding block sizes (in KB) and block generation rates (in seconds per block) to achieve the best latency bound. For example, the best latency bound is 2 hours and 25 minutes for a target throughput of 20 KB per second (at 10^{-10} security level, 25% percentage adversarial mining). This latency can be achieved by setting the block size to 208 KB and the block generation rate to one block every 10.4 seconds.

D. Case Studies in the Current Ecosystem

Some proof-of-work protocols attempt to better Bitcoin by inflating the block size (Bitcoin Cash, Bitcoin SV) or increasing the block generation rate (Litecoin). This subsection discusses the performance of these Bitcoin-like protocols. Table II describes the parameters, estimated propagation delay, and performances of the aforementioned protocols.

BCH. Bitcoin Cash (BCH) is a hard fork of Bitcoin from 2017. BCH aims to increase the throughput by increasing the maximum block size to 8 MB while remaining the same block generation rate as Bitcoin [20]. As a result, the latency is increased from around 37 hours to 50 hours for 10^{-10} security level. Had BCH increase the block generation rate (instead of the block size) by eight times, it would have obtained the same eight fold throughput improvement while at the same time shortened the latency by a factor of eight or so.

BSV. Bitcoin SV (BSV) was created in 2018 by forking BCH. BSV intended to reduce transaction fees by adjusting the protocol with even larger block sizes upper bounded by 2 GB [21]. However, the 2 GB block size will cause very long propagation delay. According to the fundamental fault tolerance bound (73), we see that BSV's security can not be guaranteed unless the adversary controls less than 3.6% of the total mining power. In reality, the only reason BSV has not observed a problem because it has low interests and the blocks its miners produce are nowhere close to 2 GB. However, when BSV starts to operate at its intended capacity, its 3.6% fault tolerance will become a major issue.

Litecoin. Litecoin is also a fork of the Bitcoin Core client that dates back to 2011. Litecoin decreases the block generation time from 10 minutes to 2.5 minutes per block [22]. For Litecoin, the latency is 10 hours 55 minutes, and the current throughput is 6.7 KB per second (for 10^{-10} security level and 25% percentage of adversarial mining power). From Figure 4, one can see that a latency less than 2 hours can be achieved with a throughput of 6.7 KB per second. This can be achieved by increasing the block generation rate and decreasing the block size.

Zcash. Proposed in 2016, Zcash is aims to provide enhanced privacy features. In 2017, Zcash doubled the maximum block size from 1 MB to 2 MB [23]. Zcash also decreased the block interval from 10 minutes to 1.25 minutes [24]. Similar to Litecoin, ZCash can be improved by increasing the block generation rate (higher throughput) and/or decreasing block size (shorter latency).

Ethereum. The second largest cryptocurrency platform Ethereum has the block generation rate of 15 seconds per block [25], [26]. The maximum gas consumption for each Ethereum block is 12.5×10^6. Given that 21000 gas must be paid for each transaction and 68 gas must be paid for each non-zero byte of the transaction [27], we estimate the maximum block size of an Ethereum block is 183 KB. Ethereum increases the block generation rate and decrease the block size. From Figure 4, for the throughput of around 12 KB per second, the latency bound is around 1 hour and 40 minutes, which is close to the current confirmation time of 1 hour and 55 minutes. The parameters of Ethereum seem to be well-chosen.

Protocol	Maximum block size	Generation rate	Propagation delay (seconds)	Latency: 10^{-3} security level	Latency:10^{-6} security level	Latency: 10^{-10} security level	Throughput (KB/second)	Fault tolerance
Bitcoin	1 MB	6	10	< 15h 45m	< 24h 45m	< 37h 50m	1.7	49.7%
BCH	8 MB	6	67.4	< 21h 55m	< 34h 10m	< 50h 30m	13.3	48.0%
BSV	2000 MB	6	1.6×10^4	N/A	N/A	N/A	N/A	3.6%
Litecoin	1 MB	24	10	< 4h 40m	< 7h 15m	< 10h 55m	6.7	48.8%
Zcash	2 MB	48	18.2	< 4h 40m	< 7h 15m	< 10h 40m	26.7	45.8%
Ethereum	0.183 MB	240	3.3	< 50m	< 1h 20m	< 1h 55m	12.2	46.2%

TABLE II

PARAMETERS AND PERFORMANCES OF NAKAMOTO-STYLE PROTOCOLS. THE PERCENTAGE OF ADVERSARIAL MINING POWER IS 25%. IN FORMULA (71), A = 0.008 AND B = 1.79.

Summary. In general, most of Nakamoto-style cryptocurrencies start with Bitcoin as the baseline and aim to improve its throughput. Since Bitcoin has a very low block generation rate, the best option according to a principled method is to increase its block generation rate. Additional improvements can be obtained by *decreasing* the block size and *further increasing* the block generation rate. This will not only increases throughput but also shortens the latency. Unfortunately, almost all the systems we looked at went in the opposite direction to increase the block size, partly due to a lack of principled methodology. The only exception is Etheruem; Etheruem's parameters are very close to the optimal ones recommended in Figure 3.

VII. CONCLUSION AND FUTURE DIRECTIONS

This work has derived a concrete latency–security trade-off for the Nakamoto consensus and applied the results to analyze existing proof-of-work longest-chain cryptocurrencies. When the mining rate is low (compared to the block propagation delay), the obtained upper bounds are close to the lower bounds from private mining.

When the block generation rate is high, however, our method does not give very tight results. Recent works [10], [11] have established the tight fault tolerance under high mining rate but tight bounds on latency remain open. Another direction is to analyze the Nakamoto consensus with dynamic participation and/or difficulty adjustment. Only asymptotic bounds exist in this direction [7], [28] and it is interesting future work to establish concrete latency–security bounds.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Available online: http://bitcoin.org/bitcoin.pdf*, 2008.

[2] J. Garay, A. Kiayias, and N. Leonardos, "The bitcoin backbone protocol: Analysis and applications," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 281–310, Springer, 2015.

[3] R. Pass, L. Seeman, and A. Shelat, "Analysis of the blockchain protocol in asynchronous networks," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 643–673, Springer, 2017.

[4] R. Pass and E. Shi, "Rethinking large-scale consensus," in *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pp. 115–129, IEEE, 2017.

[5] L. Kiffer, R. Rajaraman, and S. Abhi, "A better method to analyze blockchain consistency," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pp. 729–744, ACM, 2018.

[6] V. Bagaria, S. Kannan, D. Tse, G. Fanti, and P. Viswanath, "Prism: Deconstructing the blockchain to approach physical limits," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pp. 585–602, 2019.

[7] J. Garay, A. Kiayias, and N. Leonardos, "The bitcoin backbone protocol with chains of variable difficulty," in *Annual International Cryptology Conference*, pp. 291–323, Springer, 2017.

[8] V. Bagaria, A. Dembo, S. Kannan, S. Oh, D. Tse, P. Viswanath, X. Wang, and O. Zeitouni, "Proof-of-stake longest chain protocols: Security vs predictability," *arXiv preprint arXiv:1910.02218*, 2019.

[9] J. Niu, C. Feng, H. Dau, Y.-C. Huang, and J. Zhu, "Analysis of nakamoto consensus, revisited," *arXiv preprint arXiv:1910.08510*, 2019.

[10] A. Dembo, S. Kannan, E. N. Tas, D. Tse, P. Viswanath, X. Wang, and O. Zeitouni, "Everything is a race and nakamoto always wins," *arXiv preprint arXiv:2005.10484*, 2020.

[11] P. Gazi, A. Kiayias, and A. Russell, "Tight consistency bounds for bitcoin," *preprint*, 2020.

[12] J. Li and D. Guo, "On analysis of the Bitcoin and Prism backbone protocols in synchronous networks," in *Proceedings of the 57th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, 2019.

[13] J. Li and D. Guo, "Liveness and consistency of Bitcoin and Prism blockchains: The non-lockstep synchronous case," in *IEEE International Conference on Blockchain and Cryptocurrency*, Online, 2020.

[14] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," in *International conference on financial cryptography and data security*, pp. 436–454, Springer, 2014.

[15] C. Decker and R. Wattenhofer, "Information propagation in the bitcoin network," in *IEEE P2P 2013 Proceedings*, pp. 1–10, IEEE, 2013.

[16] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer, *et al.*, "On scaling decentralized blockchains," in *International conference on financial cryptography and data security*, pp. 106–125, Springer, 2016.

[17] A. E. Gencer, S. Basu, I. Eyal, R. Van Renesse, and E. G. Sirer, "Decentralization in bitcoin and ethereum networks," in *International Conference on Financial Cryptography and Data Security*, pp. 439–457, Springer, 2018.

[18] "Propagation evolution." http://bitcoinstats.com/network/propagation/.

[19] https://ethstats.net/.

[20] Y. Kwon, H. Kim, J. Shin, and Y. Kim, "Bitcoin vs. bitcoin cash: Coexistence or downfall of bitcoin cash?," in *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 935–951, IEEE, 2019.

[21] "Bitcoin sv removes block sizes limits via its 'genesis' hard fork." https://www.cryptoglobe.com/latest/2020/02/bitcoin-sv-removes-block-sizes-limits-via-

[22] "Litecoin mining: A helpful guide." https://www.genesis-mining.com/litecoin-mining-guide.

[23] "Zcash protocol specification." https://github.com/zcash/zips/blob/b34edb5e4090e9abb2

[24] "Network information." https://z.cash/upgrade/.

[25] D. Vujičić, D. Jagodić, and S. Ranđić, "Blockchain technology, bitcoin, and ethereum: A brief overview," in *2018 17th international symposium infoteh-jahorina (infoteh)*, pp. 1–6, IEEE, 2018.

[26] "Ethereum average block time chart." https://etherscan.io/chart/blocktime.

[27] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project yellow paper*, 2014.

[28] T. H. Chan, N. Ephraim, A. Marcedone, A. Morgan, R. Pass, and E. Shi, "Blockchain with varying number of players," 2020.

Appendix A
Converting Confirmation Depth to Confirmation Time

Let

$$\kappa(\lambda, \epsilon) = \min\left\{k :\ e^{-\lambda} \sum_{i=k}^{\infty} \frac{\lambda^i}{i!} \le \epsilon\right\}. \tag{74}$$

Then for every $\tau > 0$, the probability that $\kappa((\alpha+\beta)\tau, \epsilon)$ or more blocks are mined in τ units of time is no greater than ϵ. For example, we learn from Figure 1 that 5.58 hours of latency guarantees a security level of 0.0005. Using (74), we obtain that $\kappa(5.58 \times 6, 0.0005) = 55$. Hence if one counts 55 confirmation blocks, it implies that at least 5.58 hours have elapsed with error probability 0.0005. In all, 55 confirmation blocks guarantees 10^{-3} security level, assuming that at most 10% of the total mining power is adversarial.

At 10% adversarial mining power, Nakamoto [1] estimated that confirming after six blocks beats private attack at least 99.9% of the time. In contrast, 55 confirmation blocks guarantees the same security level regardless of what attack the adversary chooses to employ. We also note that while on average six blocks take only one hour to mine, with probability 10^{-3} it takes 2.75 hours or more to mine.

Appendix B
Proof of Lemma 21

For $i = 1, 2, \dots$, let W_i denote the inter-arrival time between the i-th honest block and the $(i-1)$-st honest block throughout the appendices. They are i.i.d. exponential random variables with mean $1/\alpha$.

For convenience, we introduce the following shorthand notation within the bounds of this proof:

$$L_n = K_1 + \cdots + K_n \tag{75}$$

$$l_n = k_1 + \cdots + k_n \tag{76}$$

for $n = 1, 2, \dots$. By convention, we let $L_0 = l_0 = 0$. It is easy to see that

$$X_i = W_{L_{i-1}+1} + \cdots + W_{L_i} \tag{77}$$

holds for $i = 1, 2, \dots$. Also, the event that $K_i = k_i$ is equivalent to the event that

$$W_{L_{i-1}+1} \le 1, \dots, W_{L_{i-1}+k_i-1} \le 1, W_{L_{i-1}+k_i} > 1. \tag{78}$$

Given $K_1 = k_1, \dots, K_i = k_i$, the event $X_i \le x$ is equivalent to the event that

$$W_{l_{i-1}+1} + \cdots + W_{l_i} \le x. \tag{79}$$

For all positive integers $n, k_1, k_2, \dots, k_n$ and real numbers $x_1, x_2, \dots, x_n$, we have

$$\begin{aligned}
&P(X_1 \le x_1, K_1 = k_1, \dots, X_n \le x_n, K_n = k_n)\\
&= P(W_1 + \cdots + W_{l_1} \le x_1,\\
&\qquad W_1 \le 1, \dots, W_{l_1-1} \le 1, W_{l_1} > 1,\\
&\qquad \cdots,\\
&\quad W_{l_{n-1}+1} + \cdots + W_{l_n} \le x_n,\\
&\qquad W_{l_{n-1}+1} \le 1, \dots, W_{l_n-1} \le 1, W_{l_n} > 1) \qquad (80)\\
&= P(W_1 + \cdots + W_{k_1} \le x_1,\\
&\qquad W_1 \le 1, \dots, W_{k_1-1} \le 1, W_{k_1} > 1)\\
&\quad \times \cdots \times\\
&\quad P(W_{l_{n-1}+1} + \cdots + W_{l_n} \le x_n,\\
&\qquad W_{l_{n-1}+1} \le 1, \dots, W_{l_n-1} \le 1, W_{l_n} > 1) \qquad (81)
\end{aligned}$$

which is a product of n probabilities, where (81) is because $W_1, W_2, \dots$ are i.i.d. Let us define a two-variable function

$$f(x, k) = P(X_1 \le x, K_1 = k) \tag{82}$$

for all $x \in (-\infty, \infty)$ and $k \in \{0, 1, \dots\}$. The i-th probability on the right hand side of (81) can be reduced as follows:

$$\begin{aligned}
&P(W_{l_{i-1}+1} + \cdots + W_{l_i} \le x_i,\\
&\qquad W_{l_{i-1}+1} \le 1, \dots, W_{l_i-1} \le 1, W_{l_i} > 1)\\
&= P(W_1 + \cdots + W_{k_i} \le x_i,\\
&\qquad W_1 \le 1, \dots, W_{k_i-1} \le 1, W_{k_i} > 1) \qquad (83)\\
&= f(X_1 \le x_i, K_1 = k_i) \qquad (84)
\end{aligned}$$

for all $i = 1, \dots, n$, where (83) is because $W_1, W_2, \dots$ is stationary. Applying (84) to (81) yields

$$\begin{aligned}
&P(X_1 \le x_1, K_1 = k_1, \dots, X_n \le x_n, K_n = k_n)\\
&= P(X_1 \le x_1, K_1 = k_1) \cdots P(X_1 \le x_n, K_1 = k_n) \qquad (85)\\
&= f(x_1, k_1) \cdots f(x_n, k_n). \qquad (86)
\end{aligned}$$

Hence the joint probability distribution of $(X_i, K_i)_{i=1}^n$ decomposes and each term takes exactly the same form. Thus Lemma 21 is established.

Appendix C
Proof of Lemma 24

This proof takes the same form as the proof of Lemma 21. For convenience, we introduce the following shorthand within the bounds of this proof:

$$M_n = J_1 + \cdots + J_n \tag{87}$$

$$m_n = j_1 + \cdots + j_n \tag{88}$$

for $n = 1, 2, \dots$. By convention, we let $M_0 = m_0 = 0$. It is easy to see that

$$V_i = X_{M_{i-1}+1} + \cdots + X_{M_i} \tag{89}$$

holds for $i = 1, 2, \dots$. Also, the event $J_i = j_i$ is equivalent to the event that

$$K_{M_{i-1}+1} > 1, \dots, K_{M_{i-1}+j_i-1} > 1, K_{M_{i-1}+j_i} = 1. \tag{90}$$

Given $J_1 = j_1, \dots, J_i = j_i$, the event $V_i \le v$ is equivalent to

$$X_{m_{i-1}+1} + \cdots + X_{m_i} \le v. \tag{91}$$

For all positive integers $n, j_1, \dots, j_n$ and real numbers $v_1, \dots, v_n$, we have

$$\begin{aligned}
&P(V_1 \le v_1, J_1 = j_1, \dots, V_n \le v_n, J_n = j_n) \\
&= P(X_1 + \cdots + X_{j_1} \le v_1, \\
&\qquad K_1 > 1, \dots, K_{j_1-1} > 1, K_{j_1} = 1, \\
&\qquad \dots, \\
&\quad X_{m_{n-1}+1} + \cdots + X_{m_n} \le v_n, \\
&\qquad K_{m_{n-1}+1} > 1, \dots, K_{m_n-1} > 1, K_{m_n} = 1) \qquad (92)\\
&= P(X_1 + \cdots + X_{j_1} \le v_1, \\
&\qquad K_1 > 1, \dots, K_{j_1-1} > 1, K_{j_1} = 1) \\
&\quad \times \cdots \times \\
&\quad P(X_{m_{n-1}+1} + \cdots + X_{m_n} \le v_n, \\
&\qquad K_{m_{n-1}+1} > 1, \dots, K_{m_n-1} > 1, K_{m_n} = 1) \qquad (93)
\end{aligned}$$

which is the product of n probabilities, where (93) is due to Lemma 21, i.e., $(X_1, K_1), (X_2, K_2), \dots$ are i.i.d. Moreover, the i-th probability on the right hand side of (93) can be reduced as:

$$\begin{aligned}
&P(X_{m_{i-1}+1} + \cdots + X_{m_i} \le v, \\
&\qquad K_{m_{i-1}+1} > 1, \dots, K_{m_i-1} > 1, K_{m_i} = 1) \\
=\ &P(X_1 + \cdots + X_{j_i} \le v, \\
&\qquad K_1 > 1, \dots, K_{j_i-1} > 1, K_{j_i} = 1)
\end{aligned} \tag{94}$$

for all $i = 1, \dots, n$. Applying (94) to (93) yields

$$\begin{aligned}
&P(V_1 \le v_1, J_1 = j_1, \dots, V_n \le v_n, J_n = j_n) \\
&= P(V_1 \le v_1, J_1 = j_1) \cdots P(V_1 \le v_n, J_1 = j_n).
\end{aligned} \tag{95}$$

Hence the joint probability distribution of $(V_i, J_i)_{i=1}^n$ decomposes and each term takes exactly the same function form. Thus Lemma 24 is established.

APPENDIX D
PROOF OF LEMMA 26

By Lemma 24, it suffices to consider V_1, the arrival time of the first double-lagger starting from time 0. Let K denote the number of honest blocks until (including) the first lagger and let $b_1, \dots, b_K$ denote that sequence of blocks. Then blocks $b_1, \dots, b_{K-1}$ are non-laggers, and block b_K is a lagger (it may or may not be a double-lagger).

With probability $e^{-\alpha}$, $W_1 > 1$. In this case, block b_1 is a double-lagger since the genesis block is a lagger. We know $K = 1$ and $V_1 = W_1$.

With probability $1 - e^{-\alpha}$, $W_1 \le 1$. Then block b_1 is not a lagger. We have $W_1 \le 1, \dots, W_{K-1} \le 1, W_K > 1$. Let V' denote the time from lagger b_K to the next double-lagger. Then we can write

$$V_1 = \begin{cases} W_1 & \text{if } W_1 > 1 \\ W_1 + \cdots + W_K + V' & \text{if } W_1 \le 1 \end{cases} \tag{96}$$

where V' follows the same distribution as V_1 by Lemma 25. Thus the MGF of V_1 can be calculated as

$$\begin{aligned}
&\mathbb{E}\left[e^{uV_1}\right] \\
&= (1 - e^{-\alpha})\mathbb{E}\left[e^{u(W_1+\cdots+W_K+V')}\middle| W_1 \le 1\right] \\
&\qquad + e^{-\alpha}\mathbb{E}\left[e^{uW_1}\middle| W_1 > 1\right] \qquad (97)\\
&= (1 - e^{-\alpha})\mathbb{E}\left[e^{u(W_1+\cdots+W_K)}\middle| W_1 \le 1\right]\mathbb{E}\left[e^{uV'}\right] \\
&\qquad + e^{-\alpha}\mathbb{E}\left[e^{uW_1}\middle| W_1 > 1\right] \qquad (98)\\
&= (1 - e^{-\alpha})\mathbb{E}\left[e^{u(W_1+\cdots+W_K)}\middle| W_1 \le 1\right]\mathbb{E}\left[e^{uV_1}\right] \\
&\qquad + e^{-\alpha}\mathbb{E}\left[e^{uW_1}\middle| W_1 > 1\right] \qquad (99)
\end{aligned}$$

where (98) is because V' and W_is are independent, and the fixed-point equation (99) is because V' is identically distributed as V_1.

If

$$(1 - e^{-\alpha})\mathbb{E}\left[e^{u(W_1+\cdots+W_K)}\middle| W_1 \le 1\right] < 1 \tag{100}$$

rearranging (99) yields

$$\mathbb{E}\left[e^{uV_1}\right] = \frac{e^{-\alpha}\mathbb{E}\left[e^{uW_1}\middle| W_1 > 1\right]}{1 - (1 - e^{-\alpha})\mathbb{E}\left[e^{u(W_1+\cdots+W_K)}\middle| W_1 \le 1\right]}. \tag{101}$$

We shall revisit the condition (100) shortly.

Note that

$$P(K = k | W_1 \le 1) = (1 - e^{-\alpha})^{k-2} e^{-\alpha}, k = 2, 3, \dots. \tag{102}$$

Hence

$$\begin{aligned}
&\mathbb{E}\left[e^{u(W_1+\cdots+W_K)}\middle| W_1 \le 1\right] \\
&= \sum_{k=2}^{\infty} P(K = k | W_1 \le 1) \times \\
&\qquad \mathbb{E}\left[e^{u(W_1+\cdots+W_k)}\middle| K = k, W_1 \le 1\right] \qquad (103)\\
&= \sum_{k=2}^{\infty} (1 - e^{-\alpha})^{k-2} e^{-\alpha} \times \\
&\quad \mathbb{E}\left[e^{u(W_1+\cdots+W_k)}\middle| W_1 \le 1, \dots, W_{k-1} \le 1, W_k > 1\right] \qquad (104)\\
&= \sum_{k=2}^{\infty} (1 - e^{-\alpha})^{k-2} e^{-\alpha} \times \mathbb{E}\left[e^{uW_1} | W_1 \le 1\right] \times \cdots \times \\
&\qquad \mathbb{E}\left[e^{uW_{k-1}} | W_{k-1} \le 1\right] \times \mathbb{E}\left[e^{uW_k} | W_k > 1\right] \qquad (105)\\
&= \sum_{k=2}^{\infty} (1 - e^{-\alpha})^{k-2} e^{-\alpha}\, \Phi_0^{k-1}(u) \Phi_1(u) \qquad (106)\\
&= e^{-\alpha} \Phi_0(u) \Phi_1(u) \sum_{k=0}^{\infty} (1 - e^{-\alpha})^k \Phi_0^k(u) \qquad (107)
\end{aligned}$$

where (105) is due to mutual independence of inter-arrival times and (106) is due to Lemmas 19 and 20.

If

$$(1 - e^{-\alpha})\Phi_0(u) < 1, \tag{108}$$

then the series sum converges to yield

$$\mathbb{E}\left[e^{u(W_1+\cdots+W_K)}\middle|W_1 \le 1\right] = \frac{e^{-\alpha}\Phi_0(u)\Phi_1(u)}{1-(1-e^{-\alpha})\Phi_0(u)}. \tag{109}$$

Let us examine the conditions (100) and (108). Note that

$$(1 - e^{-\alpha})\Phi_0(u) = \begin{cases} \frac{1-\frac{e^u}{e^\alpha}}{1-\frac{u}{\alpha}} & \text{if } u \neq \alpha, \\ \alpha & \text{if } u = \alpha. \end{cases} \tag{110}$$

It is clear that (110) is less than 1 for $u \le 0$. For $u > 0$, because e^u/u is monotone decreasing on $(0, 1)$ and monotone increasing on $(1, +\infty)$, there must exist $u \le \alpha$ that satisfies $(1 - e^{-\alpha})\Phi_0(u) = 1$. Hence the region of convergence must be a subset of $(-\infty, \alpha)$.

Let $h_\alpha(u)$ be defined as in (21). Using (109), it is straightforward to show that $(1-e^{-\alpha})\mathbb{E}\left[e^{u(W_1+\cdots+W_K)}\middle|W_1 \le 1\right] = 1$ is equivalent to $h_\alpha(u) = 0$. Let $u_0 > 0$ be the smallest number that satisfies $h_\alpha(u_0) = 0$. Then u_0 exists and $0 < u_0 \le \alpha$ according to the discussion in Section IV-C. Also, $u < u_0$ implies (100). Therefore, the region of convergence for the MGF is $(-\infty, u_0)$.

For $u < u_0$, we have by (109) and Lemma 20:

$$\mathbb{E}\left[e^{uV_1}\right] = \frac{e^{-\alpha}\Phi_1(u)(1-(1-e^{-\alpha})\Phi_0(u))}{1-(1-e^{-\alpha})\Phi_0(u) - e^{-\alpha}(1-e^{-\alpha})\Phi_0(u)\Phi_1(u)} \tag{111}$$

which becomes (23).