# Random Walks and Concurrent Zero-Knowledge*

Anand Aiyer, Xiao Liang, Nilu Nalini, Omkant Pandey

Stony Brook University, Stony Brook, USA
{aaiyer,liang1,omkant}@cs.stonybrook.edu

**Abstract.** The established bounds on the round-complexity of (black-box) concurrent zero-knowledge (c$\mathcal{ZK}$) consider adversarial verifiers with complete control over the scheduling of messages of different sessions. Consequently, such bounds only represent a *worst* case study of concurrent schedules, forcing $\widetilde{\Omega}(\log n)$ rounds for *all* protocol sessions. What happens in "average" cases against random schedules? Must all sessions still suffer large number of rounds?

Rosen and Shelat first considered such possibility, and constructed a c$\mathcal{ZK}$ protocol that adjusts its round-complexity based on existing network conditions. While they provide experimental evidence for its average-case performance, no provable guarantees are known.

In general, a proper framework for studying and understanding the average-case schedules for c$\mathcal{ZK}$ is missing. We present the first theoretical framework for performing such average-case studies. Our framework models the network as a stochastic process where a new session is opened with probability $p$ or an existing session receives the next message with probability $1-p$; the existing session can be chosen either in a first-in-first-out (FIFO) or last-in-first-out (LIFO) order. These two orders are fundamental and serve as good upper and lower bounds for other simple variations.

We also develop methods for establishing provable average-case bounds for c$\mathcal{ZK}$ in these models. The bounds in these models turn out to be intimately connected to various properties of one-dimensional random walks that reflect at the origin. Consequently, we establish new and tight asymptotic bounds for such random walks, including: expected rate of return-to-origin, changes of direction, and concentration of "positive" movements. These results may be of independent interest.

Our analysis shows that the Rosen-Shelat protocol is highly sensitive to even moderate network conditions, resulting in a large fraction of non-optimal sessions. We construct a more robust protocol by generalizing the "footer-free" condition of Rosen-Shelat which leads to significant improvements for both FIFO and LIFO models.

**Keywords:** Concurrent Zero-Knowledge, Optimistic Protocols, Average Case, Random Walks

## 1 Introduction

Concurrent zero-knowledge (c$\mathcal{ZK}$) [DNS98] protocols are a generalization of the standard notion of zero-knowledge ($\mathcal{ZK}$) [GMR85]. In settings where many protocol instances may be running simultaneously, c$\mathcal{ZK}$-protocols maintain their security whereas $\mathcal{ZK}$ protocols may become completely insecure [FS90, GK90].

The adversarial model for c$\mathcal{ZK}$ considers the "worst-case" situation where an adversarial verifier interacts with many provers and has complete control over the scheduling of messages of different sessions. The round complexity of c$\mathcal{ZK}$ in the worst-case is now largely understood — $\widetilde{\Theta}(\log n)$ rounds are necessary and sufficient for black-box simulation [CKPR01, PRS02] and constant rounds for non-black-box simulation (though current constructions for the latter require non-standard assumptions [CLP13b, PPS15, CLP15]).

* Research supported in part by NSF grant 1907908, the MITRE Innovation Program, and a Cisco Research Award. The views expressed are those of the authors and do not reflect the official policy or position of the funding agencies.

In contrast, the *average-case* complexity of c$\mathcal{ZK}$ has not received sufficient attention. Is it possible for c$\mathcal{ZK}$ sessions to terminate quickly in the average case? This question was first considered by Rosen and Shelat [RS10] who formulate an appropriate model for studying such protocols. They consider protocols that are aware of existing network conditions, and exploit them to adjust their round complexity. Two protocol sessions may thus have different number of rounds depending upon the network conditions at the time of their execution.

More specifically, the Rosen-Shelat model provides the prover algorithm full information about the scheduling of messages on the network so that it can decide to terminate early (if doing so will not harm the zero-knowledge property). If the conditions are not favorable, some sessions may still need as many rounds as the worst case solution. Such protocols are called *optimistic*, following the terminology of [Lam06]. Such prover models in c$\mathcal{ZK}$ were first considered by Persiano and Visconti [PV05], and a constant round solution was first given by Canetti et al. [CJP14]). However, all of these works require large communication that depends on the number of concurrent sessions. In contrast, Rosen and Shelat seek solutions where rounds and communication are both independent of the number of concurrent sessions.

Rosen and Shelat demonstrated that in the average-case, it is indeed possible for some sessions to terminate early while provably maintaining the c$\mathcal{ZK}$ property. More specifically, they construct a c$\mathcal{ZK}$ protocol that has the same canonical structure as [RK99, KP01, PRS02] — it consists of a preamble stage with many "slots" and a proof stage. The prover of each sessions examines the schedule to check for a critical condition called *footer-free slot*; if the condition is satisfied, the prover can terminate the session early by directly moving to the proof stage. In particular, it does not have to execute any remaining slots of the preamble stage.

While Rosen-Shelat do not provide any provable bounds, they include experimental evidence in [RS10] to demonstrate the effectiveness of their protocol. They implement the **1-Slot version** of their protocol over their local network, and find that of the 122681 TCP sessions, only 26579 did not satisfy the footer-free condition; i.e., over 79% sessions terminated after only 1 slot despite high degree of concurrency where there were 57161 or 46.5% instances of one session overlapping with another.

**This work.** The experiments in [RS10] demonstrate that the average-case schedules for c$\mathcal{ZK}$ are qualitatively different from the worst-case schedule. It seems that the worst-case situations that require large number of slots in the preamble occur only occasionally in the experiments. However, a proper framework for studying the average-case schedules for c$\mathcal{ZK}$ and developing effective strategies for them with provable bounds, is lacking.

This work initiates a rigorous study of average-case schedules for c$\mathcal{ZK}$ by first laying the framework to formally capture the "average-case network" as a stochastic process and then developing methods to prove rigorous performance bounds for candidate c$\mathcal{ZK}$ protocols in this framework. We demonstrate our approach by developing provable bounds for the Rosen-Shelat protocol.

A central observation emerging from our approach is that complexity of average-case schedules is inherently connected to properties of one-dimensional random walks that have a reflection boundary at the origin. As a result, we also establish new and tight asymptotic bounds on various properties of such random walks. This includes: the expected rate of return-to-origin as a function of walk length, changes of direction (a.k.a. "peak points"), and concentration of "positive" movements. To the best of our knowledge, these bounds are not known or follow from known results, and may be of independent interest.

Our analysis shows that the Rosen-Shelat protocol is too sensitive to the parameters of the stochastic process; in particular, it becomes almost completely ineffective even for reasonably small parameters (details provided shortly). This leads us to look for alternative protocols that are more robust to minor changes in average-case schedules. By generalizing the "footer-free" condition of Rosen-Shelat, we construct a new protocol which performs strictly better, and in some cases, optimally. We now discuss our contribution in more detail.

### 1.1 Our Contribution

**Modeling the Network.** To measure the average-case performance, the first non-trivial task is to formulate reasonable network conditions. It may be quite non-trivial – and not the subject of this work – to come up with stochastic models for networks of interest to us. We take a slightly different approach and focus on stochastic processes which are simple enough to analyze but provide useful insights into average-case schedules for c$\mathcal{ZK}$.

Towards this goal, we start with a stochastic network analogous to the *binary symmetric channel* in coding theory. More specifically, for $p \in [0, 1]$, the process opens a new session with probability $p$ and sends the next message of an existing session $s$ with probability $q = 1 - p$ (unless there are no active sessions, in which case it simply opens a new session). Depending upon how $s$ is chosen leads to models with different properties. As a starting point, the following, two fundamental cases attract our attention:

- $p$-FIFO: choose $s$ on a *first-in first-out* basis.
- $p$-LIFO: choose $s$ on a *last-in first-out* basis.

Despite their simple definition, proving bounds in these models already turns out to be highly non-trivial. The models reveal many important characteristics of the Rosen-Shelat protocol and its sensitivity to the parameter $p$. Other models for choosing $s$ can be viewed as a simple combination of these two fundamental cases; in particular, bounds for these models serve as good lower and upper bounds for other selection models.

**Analyzing Rosen-Shelat Protocol.** We proceed to prove rigorous bounds on the effectiveness of Rosen-Shelat under these models. First, we consider a simpler setting where the protocol is stopped after exactly 1-slot. This allows us to do away with some unnecessary details; note that this is also the model used by Rosen-Shelat for their empirical study. We also show that the bounds for the 1-slot model serve as a lower bound for the full protocol where all slots are allowed to continue if necessary. Our analysis proves that, in expectation, the fraction of sessions that terminate after 1-slot for Rosen-Shelat protocol after $t$ steps in the $p$-FIFO model is at most:

$$\begin{cases} \frac{1-2p}{1-p} + O\left(\frac{1}{t^{1/4}}\right) & 0 < p < 0.5 \\ 0 + O\left(\frac{1}{t^{1/4}}\right) & 0.5 \leq p < 1 \end{cases}$$

except with negligible probability in $t$. Exploiting the same approach, we can derive that the fraction for $p$-LIFO model is at most:

$$1 - p + O\left(\frac{1}{t^{1/4}}\right) \quad p \in (0, 1)$$

This is pretty bad news since, for example, the fraction for $p$-FIFO model approaches 99% quickly as $p$ increases; for $p = 0.5$ almost all sessions are already *sub-optimal*, i.e., require more than one slot (see Section 5).

**Connection to Random Walks.** As mentioned above, we prove these bounds by establishing a connection between the number of optimal sessions in 1-slot $p$-FIFO with the number of *returns to origin* in a one-dimensional biased random walk with parameter $p$. In fact, we need a slightly modified version of the standard random walk where the walk always stays on the positive side of the number line (or equivalently, contains a reflection boundary at the origin). Likewise, the bounds for the $p$-LIFO model are shown to be connected to the number of times the walk changes direction (a.k.a. "peak points"). Consequently, we establish bounds on the expected rate of returns to origin for such modified random walks as well as peak points; we also need a concentration bound for total positive moves made by the walk to bound the fraction of optimal sessions. We obtain the concentration bounds by proving that the Doob's Martingale defined over the sum of positive movements is *bounded* and hence Azuma's inequality can be applied. To the best of our knowledge, these results are new and of independent interest (see Section 4).[1] In the special case when $p < 0.5$, if we limit the number of maximum open sessions, we can also estimate the number of returns to origin using a finite state Markov chain as $t \to \infty$. This approach is somewhat simpler although it only works for $p < 0.5$ (see Section 5.2).

**Our Protocol.** Since performance of Rosen-Shelat for average-case schedules deteriorates quickly as $p$ increases, we look for alternative protocols that are not so sensitive to $p$. In designing such protocols, we must be careful to not "tailor" the construction to $p$-FIFO or $p$-LIFO models, but instead look for general principles which would be helpful in other situations too. Towards this goal, we construct a new black-box optimistic c$\mathcal{ZK}$ protocol by generalizing the key idea in Rosen-Shelat protocol, namely *nested footers*. We show that by generalizing the nested-footer condition to "depth-$d$" sessions for constant values of $d$ maintains polynomial time simulation without decreasing the optimal sessions in *any* model. At a high level, a depth $d$ session contains a fully nested session of depth $d-1$ and so on; such sessions are easy to simulate in time $O(n^d)$ (see Section 6 for more details). More interestingly, by changing values of $d$ we can control the performance of the protocol in any model. For example, by setting $d = 1$ all sessions of our protocol terminate optimally in the $p$-FIFO model; furthermore, the protocol also does extremely well for the $p$-LIFO model with very moderate values of $d$, e.g., $d = 5$ (see Section 7).

### 1.2 Related work

Early works on concurrent zero-knowledge rely on "timing constraints" on the network [DNS98, DS98, Gol02] to obtain feasibility results. These constructions are constant rounds but require large delays; these delays were later significantly improved in [PTV10]. The lower bound of [CKPR01] on the round complexity of black-box c$\mathcal{ZK}$ builds upon [KPR98, Ros00], and the $\widetilde{O}(\log n)$ protocol of [PRS02] builds upon prior work in [RK99, KP01]. Several other setup assumptions have been used to obtain constant round c$\mathcal{ZK}$ constructions with minimal trust, most notably the bare-public key model [CGGM00, CPV04, SV12] and the global hash model [CLP13a].

Using non-black-box simulation, a constant round construction for *bounded* c$\mathcal{ZK}$ was first obtained in [Bar01], with further improvements in [PV05, CJP14] who consider the client-server model of c$\mathcal{ZK}$ as in this work, [GJO+13] who assume a bound on the number of players rather than the total sessions in c$\mathcal{ZK}$. Constant round constructions can also be obtained by using 'knowledge assumptions" [Dam91, HT98, GS14] but without an explicit simulator. Constant round c$\mathcal{ZK}$ with

[1] We were not able to find these results, or derive them as simple corollaries of known results, in any standard texts on probability such as [Fel68].

explicit simulator can be achieved using non-black-box simulation under new assumptions such as strong $P$-certificates [CLP13b], public-coin indistinguishability obfuscation [PPS15, IPS15], and indistinguishability obfuscation [CLP15, BGI+01, GGH+13].

## 2 Preliminaries

We use standard notation and assume familiarity with standard cryptographic concepts such as commitment schemes, interactive proofs, zero-knowledge, and so on. We use $x$, $n = |x|$, and $\mathbb{N}$ to denote the NP instance, the security parameter, and the set of natural numbers. Notation $\langle P, V \rangle$ denotes an interactive proof with $P, V$ as prover and verifier algorithms and $\text{view}^P_{V^*}(x)$ denotes the view of algorithm $V^*$ in an interaction with $P$ on common input $x$. The transcript of the interaction between two parties contains the messages exchanged between them during an execution of the protocol.

### 2.1 Optimistic Concurrent Zero-Knowledge

We now recall the setting for optimistic concurrent zero-knowledge from [RS10]. The setting for optimistic c$\mathcal{ZK}$ is syntactically identical to the standard c$\mathcal{ZK}$ where we consider an adversarial verifier $V^*$ interacting with many provers concurrently; $V^*$ controls the message scheduling of all sessions as described by Dwork, Naor, and Sahai [DNS98].

However, in optimistic c$\mathcal{ZK}$ all parties are allowed to learn relevant information about scheduling of network messages (such as the presence of other sessions and even the scheduling itself). This is necessary to allow the provers to terminate the protocol earlier if favorable network conditions are present. Following [RS10], we consider a concurrent $V^*$ that interacts with a *single* prover $P$ proving the same instance $x$ in many concurrent sessions. For such a $V^*$, $\text{view}^P_{V^*}(x)$ denotes the *entire* view, including $x$, the randomness of $V^*$, and the messages it exchanges with $P$ in *all* sessions in the order they are sent/received.

**Definition 1 (Concurrent Zero-Knowledge).** *Let $\langle P, V \rangle$ be an interactive proof system for a language $L$. We say that $\langle P, V \rangle$ is* concurrent zero-knowledge *(c$\mathcal{ZK}$), if for every probabilistic strict polynomial-time concurrent adversary $V^*$ there exists a probabilistic polynomial-time algorithm $S_{V^*}$ such that the ensembles $\{\text{view}^P_{V^*}(x)\}_{x \in L}$ and $\{S_{V^*}(x)\}_{x \in L}$ are computationally indistinguishable.*

### 2.2 Random Walks in One Dimension

We now recall some basic definitions and facts about random walks in one dimension. We follow the convention from [Fel68, Chapter 3]. Consider a sequence of coin-tosses $(\epsilon_1, \epsilon_2, \epsilon_3, ...)$ where each $\epsilon_i$ takes values $+1$ or $-1$ with probability $p \in (0, 1)$ and $q = 1 - p$ respectively. We imagine a particle on the number line at initial position $s_0 \in \mathbb{N}$, and moves one step to its right or left depending upon the coin toss $\epsilon_i$. Note that the position of the particle at any step $t \in \mathbb{N}$ is given by the partial sum $s_t = s_0 + \sum_{i=1}^{t} \epsilon_i$.

The sequence of partial sums, $S = (s_0, s_1, s_2, ...)$, is called a *random walk*. If $s_0 = 0$, we say that the walk starts at the *origin* (or *zero*); if $s_t = 0$, the walk is said to *return to the origin (or "hit zero")* at step $t \geq 1$. Unless stated otherwise $s_0 = 0$ for all random walks in this paper. Such walks have been extensively studied [Fel68]. The probability that the walk returns to the origin at step

$t$ is denoted by $u_t$ where $u_t = 0$ for odd $t$ and $u_t = \binom{t}{\frac{t}{2}}(pq)^{\frac{t}{2}}$ otherwise. The generating function corresponding to the sequence $\{u_t\}_{t=0}^{\infty}$ is given by:

$$U(s) = \frac{1}{\sqrt{1-4p(1-p)s^2}} = \sum_{t=0}^{\infty} u_t \cdot s^t \tag{1}$$

Another important quantity is the probability of *first* return to the origin. Let $f_t$ be the probability that the walk returns to the origin at step $t$ for the *first* time, i.e., $s_1 > 0, \ldots, s_{t-1} > 0, s_t = 0$). The generating function for the sequence $\{f_t\}_{t=0}^{\infty}$ is given by:

$$F(s) = 1 - \sqrt{1-4p(1-p)s^2} = \sum_{t=0}^{\infty} f_t \cdot s^t \tag{2}$$

It can be seen that $f_{2t} = \frac{1}{2t-1} \cdot u_{2t}$ and $f_{2t-1} = 0$ for all $t \geq 1$. Furthermore, for *unbiased (i.e., $p = 0.5$)* random walks, if we use $f_i^*, u_i^*$ to denote $f_i, u_i$ (where $*$ is to insist that $p = 0.5$), then: $f_{2t}^* = u_{2t-2}^* - u_{2t}^*$, and $\sum_{i=1}^{t} f_{2t}^* = 1 - u_{2t}^*$.

### 2.3 Azuma's Inequality

**Theorem 1 (Azuma Inequality).** *If $\{B_i\}_{i=1}^{t}$ is a Martingale (i.e., for every $i \in [t]$, $E[B_i|B_1, \ldots, B_{i-1}] = B_{i-1}$) and $|B_i - B_{i+1}| \leq c_i$, then for any real $\varepsilon$:*

$$\Pr\left[\left|B_t - B_0\right| \geq \varepsilon\right] \leq 2 \cdot \exp\left(-\frac{\varepsilon^2}{2 \cdot \sum_{i=1}^{t} c_i^2}\right).$$

### 2.4 Canonical Protocol and Slots

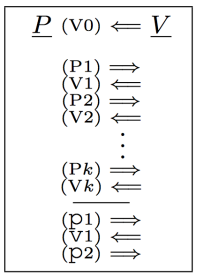


**Fig. 1.** $k$-round preamble in Rosen-shelat model

We specify some important (though standard) terminology in this section. A *canonical* c$\mathcal{ZK}$ protocol has two stages (see Figure 1): a *preamble* stage (or stage-1) and a *proof* stage (or stage-2). The preamble stage consists of messages denoted by $(V0), (P1), (V1), \ldots, (Pk), (Vk)$ where $k = k(n)$ is a protocol parameter. Every pair $(Pj, Vj)$ for $j = 1, \ldots, k$ if called **slot**. All messages of the preamble are completely independent of the common input $x$. Sometimes, the protocol may also have an initial prover message $(P0)$; however pair $(P0, V0)$ is not a slot and only serves as the initialization step of the protocol. The proof stage of the protocol consists of a canonical 3-round proof denoted by $(p1), (v1), (p2)$.

When dealing with a concurrent schedule consisting of many sessions, if we wish to identify a particular message of a session $A$, it will have $A$ as the superscript; e.g., the $j$-th slot of $A$ is denoted as $(P_j^A, V_j^A)$. Furthermore, for c$\mathcal{ZK}$ protocols of the canonical form (as in [RS10, PRS02]),

the second stage messages of a session pose no difficulty in simulation once the underlying trapdoor has been extracted from the preamble phase. Due to this, without loss of generality, we adopt the convention that when the second stage message $(p1)^A$ of a session $A$ is sent, it is immediately followed by all other messages of that stage, namely $(v1)^A, (p2)^A$. Messages $(V0)$ and $(p2)$ are often called the **first** and **last** messages of the session; however note that due to our convention of sending all second stage messages together, we will sometimes call $(p1)$ also as the **last** message.

## 3 Modeling the Network

To analyze the average-case performance of optimistic protocols, we propose a simple stochastic network model called $p$-FIFO where FIFO stands for *first-in first-out*. The model is analogous to a *binary symmetric channel* in coding theory and described below.

First, we describe this model for a general protocol and then later consider a simpler version for the case of *canonical* protocols. We assume w.l.o.g. that the first message of each session is sent by the verifier.[2] Furthermore, honest provers send their next message immediately after receiving the corresponding verifier message; the sequence of protocol messages is denoted by $\{(V0), (P1), (V1), (P2), (V2), \ldots\}$. In the sequel, all sessions are an instance of the *same* protocol.

**$p$-FIFO model.** Let $0 \leq p \leq 1$ be a parameter. The $p$-FIFO model samples a concurrent schedule sch as follows. We view sch as an ordered list of messages belonging to different concurrent sessions. sch is initially empty; messages are added to sch as follows. At each time step $t \in \mathbb{N}$, an independent coin $X_t \in \{-1, +1\}$ is tossed such that $\Pr[X_t = +1] = p$.

1. If $X_t = +1$, a new session $s$ is added to the list by adding the first message of that session, denoted $(V0)^s$ to sch; due to our convention the next prover message of $s$, denoted $(P1)^s$, is also added to sch.
2. Otherwise, let $s'$ be the *oldest* active session in sch; i.e., $s'$ is the *first* session in sch whose last message does not appear in sch up to and including time step $t-1$.
   (a) If no such $s'$ exists, open a new session $s$ as in step (1).
   (b) Else, add the next verifier message of session $s'$, denoted $(Vj)^{s'}$, to sch. Due to our convention, the corresponding prover message $(Pj)^{s'}$ is also added to sch.

**$p$-LIFO model.** Identical to $p$-FIFO except that in step (2), sessions $s'$ is chosen to be the *last* active session in sch.

*Remark 1.* Due to step 2(a), a new session is opened with probability 1 if there are no active sessions in sch. Therefore, the schedule continues to evolve forever. This allows us to study the asymptotic effectiveness of the optimistic protocols. It is possible to formulate interesting variations of these models. E.g., we can restrict the number of active sessions to not grow beyond a maximum value $N$, or allow $p$ and $N$ to change as a function of $t$.

### 3.1 Optimal Termination and the 1-Slot Model

The fastest possible termination of a canonical protocol (including the Rosen-Shelat protocol) occurs if the protocol terminates after only one slot.

[2] For canonical protocols, we can allow an inconsequential first message from the prover (see Section 2.4).

**Definition 2 (Optimal Session).** *An execution of a canonical* c$\mathcal{ZK}$ *protocol is said to terminate* optimally *if the preamble stage of the execution ends after the first slot* $(P1, V1)$*. A session that terminates optimally is called an* optimal session*.*

**Restricting to one slot.** We will primarily be interested in optimal sessions. Due to this it suffices to work with a simpler model in which each canonical protocol is terminated after exactly 1 slot. If this termination is not optimal, then the entire sessions will not be optimal no matter what happens in the rest of the slots. On the other hand, if it is optimal, the protocol will end after this slot any way. This model is called the "1-slot $p$-FIFO" model.

– **1-Slot $p$-FIFO model.** The 1-slot $p$-FIFO model is identical to the $p$-FIFO model where the underlying protocol is a canonical protocol with exactly *one slot* (i.e., $k = 1$) in the preamble phase.

We can define **1-Slot** $p$-LIFO analogously. Note that the 1-Slot restriction is also used by Rosen-Shelat in their empirical study. Our primary model of investigation will be the $p$-FIFO and $p$-LIFO models with 1 slot when working with canonical protocols.

## 4 Random Walks with Reflection at the Origin

As stated in the introduction, we analyze the round complexity of average-case c$\mathcal{ZK}$ protocols by establishing a connection to random walks with reflection at the origin. In this section, we present a formal treatment for this process. We will first give the formal definition and then establish various results about characteristics of such random walks. To the best of our knowledge, these results are not known and may be of independent interest.

**A Road Map.** In this section, one important goal is to bound the expected number of times the walk returns to the origin (Theorem 2). Toward this goal, we first show two lemmata in Section 4.1, which will help in understanding such random walks. In Section 4.2, we then establish the asymptotic characterization of the expected fraction of returns to origin. In Section 4.3, we prove concentration bounds for the number of movements to the "right" (called "positive movements"). These bounds play a control role in deriving the results of round complexity analysis in Section 5. We start with the following definition.

Recall that a random walk is defined by a sequence of partial sums $S = (s_0, s_1, s_2, ...)$ over variables $\epsilon_1, \epsilon_2, \ldots$. A random walk with reflection at the origin is a random walk with the additional constraint that whenever the partial sum $s_t$ reaches 0, the next coin toss $\epsilon_{t+1}$ must be $+1$.

**Definition 3 (Random Walk with Reflection at Origin).** *A random walk with reflection at the origin is defined by the partial sum process* $S = (s_0, s_1, s_2, ...)$ *where* $s_0 \in \mathbb{N}$ *is the starting point of the walk,* $s_t = \sum_{i=1}^{t} \epsilon_i$*, and* $\epsilon_i \in \{-1, +1\}$ *for all* $i, t \in \mathbb{N}$ *such that:* $\Pr[\epsilon_{t+1} = 1 \,|s_t = 0] = 1$ *and* $\Pr[\epsilon_{t+1} = 1 \,|s_t \neq 0] = p$*, where* $p \in (0, 1)$ *is a parameter of the random walk. If* $s_0 = 0$*, we say that the walk starts the origin.*

### 4.1 Two Lemmata of Generating Functions

In this part, we prove two lemmata of interest for the random walks with reflection at origin (defined in Definition 3), all of which are assumed to start at the origin $s_0 = 0$.

Recall that in a standard random walk without reflection (Section 2.2) the probability of first return to origin at time $t$ is denoted by $f_t$, and just a return is denoted by $u_t$. When we want to be explicit about the parameter of the random walk, we will sometimes use the notation $u_t(p), f_t(p), \ldots$ etc.

**Lemma 1.** *In a random walk with reflection at zero let $g_t$ denote the probability that walk returns to the origin for the* first *time at step $t$. Then:*

$$f_t(p) = 2p \cdot g_t(p), \quad \forall p \in (0,1), \forall t \in \mathbb{N}$$

In the sequel, the parameter $p$ will be the same for both functions $f_t, g_t$ and hence we will drop it from the notation and simply write:

$$f_t = 2p \cdot g_t. \tag{3}$$

*Proof.* First, for all odd number step $2t+1$, in both models, the particle cannot return to the origin. So we have $f_{2t+1} = g_{2t+1} = 0$, thus $f_{2t+1} = 2p \cdot g_{2t+1}$. We then only need to show the case for even-number steps, namely $f_{2t} = 2p \cdot g_{2t}$.

In a random walk model, denote $f_{2t}^+$ as the probability that a particle only traveling positive part of the number line returns to the origin for the first time at step $2t$, i.e. $s_0 = s_{2t} = 0$ and $s_i > 0$ for $i = 1, 2, ..., 2t-1$; denote $f_{2t}^-$ as the probability that a particle only traveling negative part of the number line returns to the origin for the first time at step $2t$, i.e. $s_0 = s_{2t} = 0$ and $s_i < 0$ for $i = 1, 2, ..., 2t-1$; These two cases constitute all the possibilities for returning to zero for the first time at step $2t$. Also, these two cases are "symmetric", thus have the same probability. Therefore we have:

$$\begin{cases} f_{2n} = f_{2n}^+ + f_{2n}^- \\ f_{2n}^+ = f_{2n}^- \end{cases} \tag{4}$$

In a model having the same "going right" probability $p$ but with reflection at zero, $g_{2t}$ is exactly the same as $f_{2t}^+$ except the first step: $g_{2t}$ paths "go right" in the first step with probability 1, while $f_{2t}^+$ paths "go right" in the first step with probability $p$. So we have the relation:

$$f_{2t}^+ = p \cdot g_{2t} \tag{5}$$

From the above three equations, we get:

$$f_{2t} = 2p \cdot g_{2t}$$

□

Let $G(s)$ be the generating function for $\{g_t\}$. Combining Lemma 1 with the generating function $F(s)$ (see Section 2.2), we get:

$$G(s) = \frac{F(s)}{2p} = \frac{1}{2p} \cdot [1 - \sqrt{1 - 4p(1-p)s^2}] = \sum_{t=0}^{\infty} g_t \cdot s^t \tag{6}$$

Next, for random walks with reflection at zero, we want to compute the probability of visiting the origin at time step $t$. Let $v_t$ denote this probability and $p$ be the parameter of the random walk. We prove the following.

**Lemma 2.** *In a random walk with reflection at the origin, let $v_t$ denote the probability that the walk is at the origin at time step $t$, $p$ be the parameter of the random walk, and $V(s)$ be the generating function for sequence $\{v_t\}_{t=0}^{\infty}$. Then,*

$$V(s) = \frac{2p}{2p - 1 + \sqrt{1 - 4p(1-p)s^2}} \tag{7}$$

*and*

$$\begin{cases} v_{2t} = 1 - \frac{p}{1-p}\left(\sum_{i=0}^{t} g_{2i}\right) \\ v_{2t+1} = 0 \end{cases} \qquad t \in \mathbb{N} \tag{8}$$

Note that in the equations above, the parameter for function $g_{2i}$ is $p$ (as before).

*Proof.* By definition of random walk with reflection, returning to zero cannot happen in odd steps. This proves $v_{2t+1} = 0$.

For all the paths returning to 0 at some even step $2t$, we can partition them by their first visiting to 0. This gives us the following relation:

$$v_{2t} = g_0 \cdot v_{2t} + g_2 \cdot v_{2t-2} + g_4 \cdot v_{2t-4} + ... + g_{t-2} \cdot v_2 + g_{2t} \cdot v_0 \tag{9}$$

From the above equation, we know:

$$V(s) = 1 + V(s)G(s)$$

Then with the expression of $G(s)$, the expression for $V(s)$ can be derived as follows:

$$V(s) = \frac{1}{1 - G(s)} = \frac{2p}{2p - 1 + \sqrt{1 - 4p(1-p)s^2}}$$

The remaining task is to derive the expression for $v_{2t}$. To do this, we transform the expression for $V(s)$ so that we can relate it to known quantities, such as $F(s)$. For succinctness, we use $q = 1 - p$.

$$\begin{aligned} V(s) &= \frac{2p \cdot (2p - 1 - \sqrt{1 - 4pqs^2})}{(2p-1)^2 - (1 - 4pqs^2)} = \frac{1}{2} \cdot \frac{(2p - 1 - \sqrt{1 - 4pqs^2})}{(p-1) + qs^2} \\ &= \frac{1}{2q}\left(\frac{q-p}{1-s^2} + \frac{\sqrt{1 - 4pqs^2}}{1-s^2}\right) = \frac{q-p}{2q} \cdot \frac{1}{1-s^2} + \frac{1}{2q} \cdot \frac{1 - F(s)}{1-s^2} \\ &= \frac{1}{1-s^2} - \frac{1}{2q} \cdot \frac{F(s)}{1-s^2} \end{aligned}$$

Again, since a random walk cannot return to 0 at odd steps, $f_{2t+1} = 0$. Therefore, we know that $F(s) = \sum_{t=0}^{\infty} f_t \cdot s^t = \sum_{t=0}^{\infty} f_{2t} \cdot s^{2t}$. We also know the expansion $\frac{1}{1-s^2} = \sum_{t=0}^{\infty} s^{2t}$. Thus the expression for $V(s)$ can be expanded as:

$$\begin{aligned} V(s) = \frac{1}{1-s^2} - \frac{1}{2q} \cdot \frac{F(s)}{1-s^2} &= \sum_{t=0}^{\infty} s^{2t} - \frac{1}{2q} \cdot \sum_{t=0}^{\infty}(\sum_{i=0}^{t} f_{2i}) s^{2t} \\ &= \sum_{t=0}^{\infty}(1 - \frac{1}{2q} \cdot \sum_{i=0}^{t} f_{2i}) \cdot s^{2t} \end{aligned}$$

Substituting $q$ back to $(1-p)$, we get the expression for $v_{2t}$:

$$v_{2t} = 1 - \frac{1}{2(1-p)} \cdot \sum_{i=0}^{t} f_{2i} = 1 - \frac{p}{1-p} \cdot \sum_{i=0}^{t} g_{2i}$$

□

### 4.2 Expected Number of Returns to Origin

We are now ready to establish the expected number of returns to the origin (also called "equalizations") which plays an important role when we try to analyze the round complexity of c$\mathcal{ZK}$ in $p$-FIFO model later.

Let $h_t$ denote the expected number of equalizations in a random walk of length $t$ with reflection at the origin. We define the corresponding generating function by:

$$H(s) = \sum_{t=0}^{\infty} h_t s^t \tag{10}$$

Now we establish the expression for $H(s)$ from $G(s)$ or $V(s)$ using convolutions.

**Lemma 3.** *Generating function $H$ is given by:*

$$H(s) = \frac{G(s)}{(1-s)\cdot(1-G(s))} = \frac{V(s)-1}{1-s} \tag{11}$$

*and*

$$h_{2t} = h_{2t+1} = \sum_{i=0}^{t} v_{2i} - 1 \qquad t \in \mathbb{N} \tag{12}$$

*Proof.* A similar argument as in the derivation for equation (9) can be used. For paths visiting 0 for the first time at step $i$, this visiting gives 1 equalization. For the remaining part (i.e. from step $i+1$ to step $t$), the expected number of equalizations is just $h_{t-i}$. So we have the following recursive relation:

$$h_t = \sum_{i=0}^{t} g_i \cdot (1 + h_{t-i}) = \sum_{i=0}^{t} (g_i + g_i \cdot h_{t-i})$$

From this recursive format, we get the relation between $H(s)$ and $G(s)$:

$$H(s) = \frac{G(s)}{1-s} + G(s)\cdot H(s)$$

Then we have the expression for $H(s)$:

$$H(s) = \frac{G(s)}{(1-s)\cdot[1-G(s)]} = \frac{V(s)}{1-s} - \frac{1}{1-s} \tag{13}$$

Note that the last equality sign comes from the relation $V(s) = \frac{1}{1-G(s)}$.

In equation (13), we know the expansion for both $V(s)$ and $\frac{1}{1-s}$, so the expansion of $H(s)$ can be calculated:

$$H(s) = \sum_{t=0}^{\infty} \left( \sum_{i=0}^{t} v_i - 1 \right) \cdot s^t \tag{14}$$

This expansion tells that: $h_t = \sum_{i=0}^{t} v_i - 1$.

Note the fact that for any odd number step $2t+1$, we have $v_{2t+1} = 0$. Thus:

$$h_{2t+1} = \sum_{i=0}^{2t} v_i + v_{2t+1} - 1 = \sum_{i=0}^{t} v_{2i} - 1 = h_{2t}$$

□

We are now ready to establish the main result of this section, which shows the asymptotic behavior of $h_t$ in relation to $t$. The following theorem captures this behavior. Since it may be of independent interest, we state the result as a rate of return to origin in a random walk with reflection at the origin. We highlight both, the asymptotic behavior in big-$O$ notation as well as the limit behavior.

**Theorem 2 (Expected Rate of Returns to Origin).** *In a random walk with reflection at the origin, for $p \in (0,1)$, $q = 1-p$, and every positive $t$, the rate of return to the origin is given by:*

$$\frac{h_t}{t} = \begin{cases} \frac{1}{2}\left(1 - p/q\right) + O\left(1/t\right) & p < 0.5 \\ O\left(1/\sqrt{t}\right) & p = 0.5 \\ O\left(1/t\right) & p > 0.5 \end{cases}$$

*Furthermore,* $\lim_{t\to\infty} \frac{h_t}{t} = \frac{1}{2}\left(1 - \frac{p}{q} \cdot G(1)\right)$ *which equals 0 for* $p \geq 0.5$ *and* $\frac{1}{2}\left(1 - \frac{p}{q}\right)$ *for* $p < 0.5$.

*Proof.* First, note that

$$\begin{aligned} h_{2t} &= \sum_{k=0}^{t} v_{2k} - 1 = \sum_{k=0}^{t}\left(1 - \frac{p}{q}\sum_{i=0}^{k} g_{2i}\right) - 1 = t + 1 - \frac{p}{q}\sum_{k=0}^{t}\sum_{i=0}^{k} g_{2i} - 1 \\ &= t - \frac{p}{q} \cdot \left[t \cdot g_2 + (t-1) \cdot g_4 + ... + 2 \cdot g_{2t-2} + 1 \cdot g_{2t}\right] \end{aligned}$$

If $p = \frac{1}{2}$, we have:

$$\begin{aligned} \frac{h_{2t}}{2t} &= \frac{t - [t \cdot g_2 + (t-1) \cdot g_4 + ... + 2 \cdot g_{2t-2} + 1 \cdot g_{2t}]}{2t} \\ &= \frac{t - [t \cdot \sum_{i=1}^{t} g_{2i} - \sum_{i=1}^{t}(i-1)g_{2i}]}{2t} \\ &= \frac{1}{2} \cdot \left(1 - \sum_{i=1}^{t} g_{2i} + \sum_{i=1}^{t} \frac{(i-1)}{t} g_{2i}\right) = \frac{1}{2} \cdot \left(1 - \sum_{i=1}^{t} g_{2i} + \sum_{i=1}^{t} \frac{(i-1)}{t} f_{2i}\right) \end{aligned}$$

Observe that for $p = 0.5$, $g_{2i} = f^*_{2i}$ so that (a) $1 - \sum_{i=1}^{t} g_{2i} = u^*_{2t} = O\left(1/\sqrt{t}\right)$ (see last few lines of Section 2.2); (b) furthermore, by using the facts that (i) $f_{2i} = f^*_{2i} = O(1/i\sqrt{i})$ for $p = 0.5$, and (ii) $\sum_{i=1}^{n} \frac{1}{\sqrt{i}} < 2\sqrt{n}$, we get:

$$\sum_{i=1}^{t} \frac{(i-1)}{t} f_{2i} = O\left(\frac{1}{\sqrt{t}}\right).$$

Combining observations (a) and (b), we see that for $p = 0.5$, $h_t/t = O(1/\sqrt{t})$.

If $p \neq \frac{1}{2}$, let us first observe (simply consider the difference between $G(s)$ and its differentiation $G'(s)$ w.r.t. $s$, details omitted):

$$\sum_{i=1}^{\infty}(i-1) \cdot g_{2i} = \frac{1}{2}G'(1) - G(1).$$

Then we have:

$$\begin{aligned}
\frac{h_{2t}}{2t} &= \frac{1}{2} \cdot \left[1 - \frac{p}{q}\left(\sum_{i=1}^{t} g_{2i} - \sum_{i=1}^{t} \frac{(i-1)}{t} g_{2i}\right)\right] \\
&\leq \frac{1}{2} \cdot \left[1 - \frac{p}{q}\left(\sum_{i=1}^{\infty} g_{2i} - \sum_{i=1}^{\infty} \frac{(i-1)}{t} g_{2i}\right)\right] \\
&= \frac{1}{2} \cdot \left[1 - \frac{p}{q}\left(G(1) - \frac{\frac{1}{2}G'(1) - G(1)}{t}\right)\right] = \frac{1}{2} \cdot \left[1 - \frac{p}{q} \cdot G(1) + O\left(\frac{1}{t}\right)\right] \\
&= \frac{1}{2} \cdot \left(1 - \frac{p}{q} \cdot G(1)\right) + O\left(\frac{1}{t}\right)
\end{aligned}$$

This establishes that for $p \neq \frac{1}{2}$:

$$\frac{h_{2t}}{2t} \quad = \quad \frac{1}{2} \cdot \left(1 - \frac{p}{q} \cdot G(1)\right) + O\left(\frac{1}{t}\right)$$

Using $s = 1$ in the expression of $G(s)$, we see that $G(s) = 1$ for $p \leq 0.5$ and $G(s) = q/p$ for $p > 0.5$. Using these values gives us the bounds for cases $p < 0.5$ and $p > 0.5$ for all values of the form $2t$, or equivalently all even values of $t$.

For odd values, observe that $h_{2t} = h_{2t+1}$, so that:

$$\frac{h_{2t+1}}{2t+1} = \frac{h_{2t}}{2t+1} \leq \frac{h_{2t}}{2t}$$

Thus the bound holds for odd $t$ as well.

To get the claim about the limit behavior, the derivation for $\frac{h_{2t}}{2t}$ involving $G(1)$ requires some minor manipulations, but nevertheless follows easily. The details are omitted. □

*Remark 2.* We notice that in the work of Essifi and Peigné [EP15] (and its precursor [Lal95]), similar results were obtained using measure-theory techniques. But their results are not applicable for our purpose for the following reasons. Their work does not capture the (most important) case of $p < 0.5$. Even for other cases ($p = 0.5$ and $p > 0.5$), they only consider the "limit" behavior when $t$ tends to infinity; in contrast, we provide a "Computer-Science flavor" result which shows direct dependence on $t$.

### 4.3 Concentration Bounds for Positive Movements

To measure the true number of optimal sessions in terms of total sessions, we need to know the distribution of total sessions in the 1-slot $p$-FIFO model. This is related to the total number of movements to the "right" (also called "positive movements" since it corresponds to variables $\epsilon_t = +1$). We prove that the total number of positive movements is sharply concentrated around its expectation.

It is tempting to think that we can obtain these bounds using some form of Chernoff-Hoeffding in the limited dependence setting. Unfortunately, all of our attempts to use this approach were unsuccessful. Instead, we rely on Martingales.

In fact, we are able to prove a stronger result. We show that the Doob Martingale defined for, roughly speaking, the sum of coin-tosses of the random walk is *bounded*. The proof relies on the properties of the random walk. This allows us to apply Azuma's inequality, but is of independent interest.

**Theorem 3.** *Let $S = (s_0 = 0, s_1, s_2, \ldots)$ be a random walk with reflection at the origin, defined over binary random variables $(\epsilon_1, \epsilon_2, \ldots)$. For all positive $i$, let*

$$X_i = \frac{1+\epsilon_i}{2} = \begin{cases} 1 & \text{if } \epsilon_i = 1 \\ 0 & \text{if } \epsilon_i = -1 \end{cases}$$

*Then, random variable $M_t = \sum_{i=1}^{t} X_i$ counts the number of positive movements in the walk. Furthermore, if $B_i := E_{X_{i+1}, X_{i+2}, .., X_t}[M_t | X_1, X_2, ..., X_i]$ for $i \in \{1, \ldots, t-1\}$ then $\{B_i\}_{i=1}^{t-1}$ is a Martingale for all $t \in \mathbb{N} \setminus \{0\}$ such that:*

$$|B_i - B_{i+1}| \leq 1.$$

*Proof.* Observe that the variables $X_i$ correspond to the movements on right, and since negative movements are discarded by setting $X_i = 0$, the sum $M_t$ indeed represents the total positive movements. Furthermore, the sequence $\{B_i\}$ is the standard Doob's Martingale so that $E[B_i | B_1, \ldots, B_{i-1}] = B_{i-1}$ (see, e.g., [AS04, Chap. 7]).

The main task is now to show that the martingale $\{B_i\}_i$ is indeed bounded by 1. The proof is somewhat tedious and relies on certain characteristics of random walks with reflection. The proof of this bound is given in Section A. □

**Corollary 1.**

$$\Pr\left[\left|\sum_{i=1}^{t} X_i - E\left[\sum_{i=1}^{t} X_i\right]\right| \geq \varepsilon\right] \leq 2 \cdot \exp\left(-\frac{\varepsilon^2}{2 \cdot t}\right) \tag{15}$$

*Proof.* Consider the Doob's Martingale $\{B_i\}$ from Theorem 3. Observe that $B_0 = E[M_t] = E[\sum_{i=1}^{t} X_i]$ and $B_t = E[M_t | X_1, X_2, ..., X_t] = M_t = \sum_{i=1}^{t} X_i$. Furthermore, since $|B_i - B_{i+1}| \leq 1$, we can set $c_i = 1$ for all $i$ in Azuma's inequality (Theorem 1) to get stated bound. □

**Note:** We prefer this form since it makes it easier to see that we are comparing the sum of $X_i$ with its expectation. However, in future, we will freely substitute $M_t$ for the sum $\sum_{i=1}^{t} X_i$ for succinctness.

## 5 Analysis of Rosen-Shelat Protocol

We are now ready to analyze the effectiveness of Rosen-Shelat protocol against an average-case network, as modeled by the 1-Slot $p$-FIFO process described in Section 3.1. We also establish bounds for 1-Slot $p$-LIFO.

We start by recalling the Rosen-Shelat protocol (see Protocol 1). The protocol relies on the notion of a "nested footer" recalled below:[3]

[3] The statement of this definition in [RS10] actually has $(Vk)$ instead of $(p1)$ as $A$'s nested message. However, we believe that it is a typo and by $(Vk)$ authors really mean the presence of second stage messages; this is guaranteed by having $(p1)$ in the definition but not by $(Vk)$. Indeed, many nested protocols may terminate without ever reaching $(Vk)$. If $(Vk)$ is used in the definition, the simulator in [RS10] will run in exponential time even for the simple concurrent schedule described in [DNS98] (and shown in red in Fig. 1 in [RS10]).

**Protocol 1** Rosen-Shelat Protocol [RS10]

**Common Input:** $x \in \{0,1\}^n$, security param. $n$, round param. $k \in \omega(\log n)$.
**Prover's Input:** a witness $w$ such that $R_L(x,w) = 1$
**Stage 1:**
P $\to$ V ($P0$): Send first message of perfectly hiding commitment **Com.**
V $\to$ P ($V0$): Using the commitment **Com**, commit to random $\sigma \in \{0,1\}^n$, $\{\sigma^0_{i,j}\}^k_{i,j=1}, \{\sigma^1_{i,j}\}^k_{i,j=1}$ such that $\sigma^0_{ij} \oplus \sigma^1_{ij} = \sigma$ for all $i,j$.
**Slot** $j \in [k]$**:**
P $\to$ V ($Pj$): Send a random challenge $r_i = r_{1,j}, \cdots, r_{k,j}$.
V $\to$ P ($Vj$): Upon receiving a message $r_i$, decommit to $\sigma^{r_{1,j}}_{1,j}, \cdots, \sigma^{r_{k,j}}_{k,j}$.
P $\to$ V: If any of the decommitments fails verification, abort.
If slot $j$ is footer-free **or** $j = k$ move to **stage 2**.
If slot $j$ is not footer-free and $j < k$ move to slot $j+1$.
**Stage 2:**
$P$ and $V$ engage in Blum's 3-round Hamiltonicity protocol using challenge $\sigma$:
1. P $\to$ V ($p1$): Use witness to produce first message of Ham protocol
2. V $\to$ P ($v1$): Decommit to $\sigma$ and to $\{\sigma^{1-r_{i,j}}_{ij}\}^k_{i,j=1}$.
3. P $\to$ V ($p2$): If decommitments are valid and $\sigma^0_{ij} \oplus \sigma^1_{ij} = \sigma$ for all $i$,$j$, answer $\sigma$ with third message of Ham protocol. Otherwise abort.

**Definition 4 (Nested Footer).** *Slot $j$ of session $B$ is said to have a* nested footer *of session $A$ within it if session $A$'s ($p1$) message occurs between messages $(Pj), (Vj)$ of session $B$. A slot is said to be* footer free *if it has no nested footer.*

### 5.1 Bounding Optimal Sessions

We measure the effectiveness of Rosen-Shelat protocol by counting the number of *optimal sessions* as the schedule evolves over time $t$ according to the *1-slot* $p$-FIFO process. Since $t$ does not represent the actual number of total sessions, we will also bound the expected *ratio* of optimal sessions w.r.t. total sessions.

We start by proving the following key proposition. It states that the number of optimal sessions in 1-slot $p$-FIFO are equal to the number of returns to the origin in a random walk defined over the coin-tosses of $p$-FIFO.

**Proposition 1.** *Let $\mathbf{X} = (X_1, X_2, \ldots)$ be the sequence of coin tosses defining the 1-Slot $p$-FIFO process. Let $S = (s_0 = 0, s_1, s_2, \ldots)$ be the partial sum process defined over $\mathbf{X}$. Then, $S$ is a random walk with parameter $p$ and reflection at the origin. Furthermore, for any finite time step $t \in \mathbb{N}$, the number of optimal sessions in $\mathbf{X}$ up to and including $t$ is equal to the number of returns to the origin in the random walk $S$.*

*Proof.* Note that return to the origin at step $t$ is denoted by $s_t = 0$.

We first show that every return to the origin gives an optimal session. If $s_t = 0$, there is no session remaining active when step $t$ is finished. Then a new session $A$ will be opened at step $t+1$. By the 1-Slot $p$-FIFO rule, every session opened later will be closed after $A$'s closing. Thus $A$ is an optimal session.

Then we show that for every optimal session, there is a corresponding return to zero (or $s_t = 0$). Given an optimal session $A$ which is opened at step $t+1$. If we assume $s_t \neq 0$, there must be some session $B$, which is opened before $A$ and still active up to step $t$. By 1-Slot $p$-FIFO rule, $B$ has to be closed before $A$'s closing. So $A$ contains $B$'s footer, thus cannot be optimal. Therefore, we must have $s_t = 0$.

Combining the above two claims together completes the proof. □

According to Proposition 1, we can compute the expected fraction of optimal session for 1-Slot $p$-FIFO model by analyzing the behavior of returns to the origin in a random walk. With the notations defined in Section 4, the following theorem gives the asymptotic bounds for the Rosen-Shelat protocol.

**Theorem 4.** *Let* $\mathsf{OPT}_{\mathsf{RS}}(p,t)$ *denote the expected fraction of optimal sessions for the Rosen-Shelat protocol in the 1-slot* $p$-FIFO *model. Then, except with probability* $\delta_t := 2 \cdot \exp\left(-\frac{\sqrt{t}}{2}\right)$,

$$\mathsf{OPT}_{\mathsf{RS}}(p,t) = \left(1 - \frac{p}{q} \cdot G(1)\right) \pm O\left(\frac{1}{t^{1/4}}\right)$$

*where* $q = 1-p$, $p \in (0,1)$, *and* $t \in \mathbb{N}$. *Furthermore,* $\lim_{t\to\infty} \mathsf{OPT}_{\mathsf{RS}}(p,t) = 1 - \frac{p}{q} \cdot G(1)$, *which equals 0 for* $p \geq 0.5$ *and* $(1-p/q)$ *otherwise.*

*Proof.* This proof is based on Theorem 2 and Corollary 1.

To get the ratio of optimal sessions with total sessions, we first need a concentration bound for the total sessions. Using notation from Section 4.3, the total sessions are represented by the variable $M_t = \sum_{i=1}^{t} X_i$ so that

$$E[M_t] = \sum_{i=1}^{t} E[X_i] = t \cdot p + (1-p) \cdot \sum_{i=1}^{t} v_{i-1} = t \cdot p + q \cdot (h_{t-1} + 1).$$

Let $\varepsilon = t^{\frac{3}{4}}$ and apply inequality (15) (Corollary 1); we get that except with probability $\delta_t = 2 \cdot \exp\left(-\frac{\sqrt{t}}{2}\right)$,

$$M_t \in \Big[E[M_t] - \varepsilon,\ E[M_t] + \varepsilon\Big]. \tag{16}$$

Now, let $z_t$ denote the actual number of optimal sessions after $t$ steps. By definition, $E[z_t] = h_t$. Using the range bound for $M_t$ above, we conclude that except with probability $\delta_t$, the fraction $z_t/M_t$ of optimal sessions satisfies:

$$\frac{z_t}{M_t} \in \left[\frac{z_t}{E[M_t] + \varepsilon},\ \frac{z_t}{E[M_t] - \varepsilon}\right] \tag{17}$$

Substituting the value of $E[M_t]$,

$$\begin{aligned} \frac{z_t}{M_t} &\in \left[\frac{z_t}{tp + q(h_{t-1}+1) + \varepsilon},\ \frac{z_t}{tp + q(h_{t-1}+1) - \varepsilon}\right] \\ \implies E\left[\frac{z_t}{M_t}\right] &\in \left[\frac{E[z_t]}{tp + q(h_{t-1}+1) + \varepsilon},\ \frac{E[z_t]}{tp + q(h_{t-1}+1) - \varepsilon}\right] \end{aligned}$$

We now make a few observations. First, note that $\mathsf{OPT}_{\mathsf{RS}}(t,p) = E[z_t/M_t]$ and $E[z_t] = h_t$. Furthermore, $h_{t-1}/t = h_t/t$ asymptotically. If we define $\gamma_t = h_t/t = h_{t-1}/t$, $\varepsilon_1 = (\varepsilon + q)/t = O(t^{-1/4})$, and $\varepsilon_2 = (\varepsilon - q)/t = O(t^{-1/4})$, the above range equation simplifies to:

$$\mathsf{OPT}_{\mathsf{RS}}(t,p) \in \left[\frac{\gamma_t}{p + q\gamma_t + \varepsilon_1},\ \frac{\gamma_t}{p + q\gamma_t - \varepsilon_2}\right] \tag{18}$$

To complete the proof, simply plugin the value of $\gamma_t$ from Theorem 2 and observe that $\varepsilon_1, \varepsilon_2$ are small enough to be sucked into the $O$-notation. Specifically, (1) if $p < 0.5$, $\gamma_t = \frac{1}{2}(1 - p/q) + O(1/t)$ and $(p + q\gamma_t) = \frac{1}{2} + O(1/t)$. Note that the $O(1/t)$ term will also be absorbed into $\varepsilon_1$ or $\varepsilon_2$, which then gives the claimed bound; (2) if $p \leq 0.5$, $\gamma_t$ grows slower than $\varepsilon_1$ and $\varepsilon_2$ so that both sides of range become $O(t^{-1/4})$ which is also the bound for $\mathsf{OPT}_{\mathsf{RS}}$ since $G(1) = q/p$ when $p \geq 0.5$.

For the limit behavior, we simply use the claim from Theorem 2 regarding limit behavior of $h_t/t$. □

### 5.2 Markov Chain Approach

In the case $p < 0.5$, a simpler analysis is possible by using Markov chains for a slightly modified model where the total number of sessions are not allowed to grow beyond some fixed bound, say $n$. This is equivalent to having a reflection boundary at time step $n$ in the random walk model so that walk always stays between 0 and $n$. Without this bound, or when $p \geq q$, the resulting Markov chain may not be finite.

To analyze the expected number of returns to the origin when $p < 0.5$, consider a Markov chain $M$ with $n$ states marked from '0' to '$n-1$'. The transition probabilities to capture the $p$-FIFO model are as follows. If the chain is in state '0', it goes to state '1' with probability 1. Likewise, if it is in state '$n-1$' it returns to state '$n-2$' with probability 1. For any other state '$i$' the chain goes to state '$i+1$' with probability $p$ and '$i-1$' with probability $q = 1-p$. Let $\pi = (\pi_0, \ldots, \pi_{n-1})$ denote the state steady distribution where $\pi_i$ is the probability that the chain is in state '$i$' for $i \in [0, n-1]$. The steady state equations for this chain are:

$$\begin{aligned}
\pi_0 &= q \times \pi_1, \\
\pi_1 &= \pi_0 + q \times \pi_2, \\
\pi_2 &= p \times \pi_1 + q \times \pi_3, \\
&\ldots \\
\pi_{n-2} &= p \times \pi_{n-3} + \pi_{n-1}, \\
\pi_{n-1} &= p \times \pi_{n-2}
\end{aligned}$$

Using $\sum_{i=0}^{n-1} \pi_i = 1$ and solving for $\pi_0$, we get:

$$\pi_0 = \left(1 + \left(\frac{1 - (p/q)^{n-2}}{q - p}\right) + \left(\frac{p}{q}\right)^{n-2}\right)^{-1}$$

Observe that every time the walk returns to the origin, the chain would be in state 0. Therefore, the expected number of returns to the origin in a walk of length $t$ can be estimated as $\pi_0 t$ for sufficiently large $t$.

**When $p < 0.5$ and $n$ is large**, we can ignore the term $\left(\frac{p}{q}\right)^{n-2}$ since $p < q$. This yields:

$$\pi_0 \approx \frac{q - p}{2q} = \frac{1}{2}\left(1 - \frac{p}{q}\right)$$

This is indeed the same asymptotic behavior we proved about the rate of return to origin. This analysis does not hold for $p \geq q$ or without the bound $n$ since the chain may not have a stationary distribution. Nevertheless, this approach can be useful when dealing with more complex distributions such as the Poisson distribution.

### 5.3 Deriving the Bounds for LIFO Model

In this section, we use the approach developed in Section 5.1 to measure the effectiveness of Rosen-Shelat protocol under the set 1-Slot $p$-LIFO setting. First, we give a proposition to relate the optimal session in $p$-LIFO process to a so-called "peak point" in the random walk with reflection at the origin. Then by analyzing the frequency of peak points, we derive the bounds for the expected fraction of optimal sessions.

In a random walk, if the particle moves left immediately after a right movement, we say that it forms a "peak point". A formal definition using our standard notations follows. Let $S = (s_0 = 0, s_1, s_2, ...)$ be a random walk defined over binary random variables $(\epsilon_1, \epsilon_2, ...)$. The pair $(\epsilon_i, \epsilon_{i+1})$ forms a peak point if and only if $\epsilon_i = 1$ and $\epsilon_{i+1} = -1$.

The following proposition shows the relation between peak points and optimal sessions in $p$-LIFO process.

**Proposition 2.** *Let $\mathbf{X} = (X_1, X_2, \ldots)$ be the sequence of coin tosses defining the 1-Slot $p$-LIFO process. Let $S = (s_0 = 0, s_1, s_2, \ldots)$ be the partial sum process defined over $\mathbf{X}$. Then, $S$ is a random walk with parameter $p$ and reflection at the origin. Furthermore, for any finite time step $t \in \mathbb{N}$, the number of optimal sessions in $\mathbf{X}$ up to and including $t$ is equal to the number of peak points in the random walk $S$.*

*Proof.* Consider a peak point $(X_i = 1, X_{i+1} = -1)$. In the protocol, it means a new session $s$ is opened at step $i$, and an active session is closed at step $i+1$. Due to the LIFO strategy, the session closed at step $i+1$ must be the newestly opened session, namely $s$. Therefore, session $s$ does not contain a footer, thus it is an optimal session. And one can easily check that this is the only case when the 1-slot $p$-LIFO setting gives an optimal session. So the proposition follows. □

We now present a theorem of the expected fraction of peak points, which plays a similar role as Theorem 2 in $p$-FIFO model. It will later help us to investigate the behavior of optimal sessions in $p$-LIFO process.

**Theorem 5 (Expected Rate of Peak Points).** *In a random walk with reflection at origin, for $p \in (0, 1)$, $q = 1 - p$, and every position $t$, denote the expected number of peak points as $\ell_t$. The rate of peak points is given by:*

$$\frac{\ell_t}{t} = \begin{cases} \frac{q}{2} + O\left(\frac{1}{t}\right) & p < 0.5 \\ p \cdot q + O\left(\frac{1}{\sqrt{t}}\right) & p = 0.5 \\ p \cdot q + O\left(\frac{1}{t}\right) & p > 0.5 \end{cases}$$

*Proof.* Let $X_i$ be the random variable which takes the value 1 if the particle moves right at step $i$ and then moves left at step $i+1$; otherwise, $X_i = 0$. The number of peak points up to step $t$ would be $\sum_{i=0}^{t-2} X_i$. Also, it is easy to see $E[X_i] = v_i \cdot q + (1 - v_i) \cdot p \cdot q$ if we divide the event $X_i = 1$ into tow cases by whether the particle is at the origin at step $t$. We then have:

$$\ell_t = E[\sum_{i=0}^{t-2} X_i] = \sum_{i=0}^{t-2} E[X_i] = \sum_{i=0}^{t-2} [v_i \cdot q + (1 - v_i) \cdot p \cdot q]$$
$$= q^2 \cdot \left(\sum_{i=0}^{t-2} v_i\right) + (t-1) \cdot p \cdot q = q^2 \cdot (h_{t-2} + 1) + (t-1) \cdot p \cdot q$$

Divide both sides by $t$ and notice that $(h_{t-2}+1)/t = h_t/t$ asymptotically. We then have:

$$\frac{\ell_t}{t} = q^2 \cdot \frac{h_t}{t} + p \cdot q - \frac{p \cdot q}{t} = q^2 \cdot \frac{h_t}{t} + p \cdot q - O\left(\frac{1}{t}\right)$$

Plugging the previous result of $h_t/t$ into the above equation gives us the claimed bound in the theorem. □

Now we are ready to bound the expected fraction of the number for optimal sessions under the $p$-LIFO setting.

**Theorem 6.** *Let* $\mathsf{OPT}_{\mathsf{RS}}^{\mathsf{LIFO}}(p,t)$ *denote the expected fraction of optimal sessions for the Rosen-Shelat protocol in the 1-slot* $p$-LIFO *model. Then, except with probability* $\delta_t := 2 \cdot \exp\left(-\frac{\sqrt{t}}{2}\right)$,

$$\mathsf{OPT}_{\mathsf{RS}}^{\mathsf{LIFO}}(p,t) = q \pm O\left(\frac{1}{t^{1/4}}\right)$$

*where* $q = 1-p$, $p \in (0,1)$, *and* $t \in \mathbb{N}$.

*Proof.* The proof is almost the same as the one for Theorem 4, except the $\gamma_t$ in the numerator of interval (18) should be substituted by $\gamma'_t = \ell_t/t$. Namely:

$$\mathsf{OPT}_{\mathsf{RS}}^{\mathsf{LIFO}}(t,p) \quad \in \quad \left[\frac{\gamma'_t}{p+q\gamma_t+\varepsilon_1},\ \frac{\gamma'_t}{p+q\gamma_t-\varepsilon_2}\right] \tag{19}$$

Going through the same argument as in the proof of Theorem 4 with the result for $\ell_t/t$ from Theorem 5 will complete the proof. □

## 6 Our Protocol and Simulator

We now present our modification to the Rosen-Shelat protocol. Our modification simply replaces the footer-free condition with a slightly more complex condition that we call "depth $d$" slots. This results in increasing the expected running time of the simulator to $\mathsf{poly}(n^d)$, which remains polynomial if $d$ is chosen to be a constant, but does not change anything else. By setting $d$ appropriately, one can improve the overall performance of the protocol.

At a high level, a "depth $d$" slot is a generalization of a slot with a nested-free where the slot is allowed to contain nested sessions so long as the total *recursive depth* of all the nested sessions is at most $d$. Such sessions can be solved easily in exponential time in $d$ (in expectation) using naïve recursive rewinding. We start with a few definitions regarding depth of nested sessions and slots.

**Definition 5 (Session Nested in a Slot).** *We say that a session* $B$ *is nested in slot* $j$ *of session* $A$ *if* both $(V0)^B$ *and* $(p1)^B$ *(i.e., the first and the last messages of session* $B$*) appear after* $P_j^A$ *but before* $V_j^A$ *in the schedule.*

Note that if $(p1)^B$ appears in a slot of $A$ then by our convention all second-stage messages of $B$ occur in that slot. Therefore, the above definition simply says that slot $j$ of session $A$ contains the entire session $B$ (except possibly $(P0)$ which is irrelevant). Next, we define slots with increasing levels of nesting. This is done by defining the depth of a session and a slot recursively. The definition

**Protocol 2** Our Protocol

**Common Input:** $x \in \{0,1\}^n$, sec. param. $n$, round param. $k = \omega(\log n)$, degree $d$.
**Prover's Input:** a witness $w$ such that $R_L(x, w) = 1$
**Stage 1:**
P $\to$ V ($P0$): Send first message of perfectly hiding commitment **Com.**
V $\to$ P ($V0$): Using the commitment **Com**, commit to random $\sigma$.

**Slot** $j \in [k]$**:**
P $\to$ V ($Pj$) : Send a random challenge
V $\to$ P ($Vj$): Upon receiving a message $r_i$, decommit.
P $\to$ V : If any of the decommitments fails verification, abort.
If $\mathsf{depth}_j \leq d$ **or** $j = k$, move to **stage 2**.
If $\mathsf{depth}_j > d$ **and** $j < k$, move to slot $j + 1$.
**Stage 2:**
$P$ and $V$ engage in Blum's 3-round Hamiltonicity protocol using challenge $\sigma$.
1. P $\to$ V ($p1$): Use witness to produce first message of Ham protocol
2. V $\to$ P ($v1$): Decommit to $\sigma$
3. P $\to$ V ($p2$): If decommitments are valid, answer $\sigma$ with third message of Ham protocol. Otherwise abort.

below states that the depth of a slot is 0 if it does not contain any nested sessions; otherwise, it is 1 more than the depth of the session that is nested in the slot and has the maximum depth of all sessions nested in that slot. The depth of a session is equal to the depth of the slot(s) with maximum depth.

**Definition 6 (Slot Depth and Session Depth).** *For a session $A$ and index $j \in [k]$, let $F_j^A$ denote the set of all sessions $B$ such that $B$ is nested in slot $j$ of session $A$. Then, the depth of slot $j$ of session $A$, denoted $\mathsf{depth}_j^A$, is defined recursively as follows:*

$$\mathsf{depth}_j^A = \begin{cases} 0, & F_j^A = \emptyset \\ 1 + \max_{B \in F_j^A}\{\mathsf{depth}^B\}, & F_j^A \neq \emptyset \end{cases}$$

*where $\mathsf{depth}^B$ (without any subscript) denotes the depth of session $B$, which in turn, is simply the depth of its highest nested slot; i.e.,*

$$\mathsf{depth}^B = \max_{i \in [k]}\{\mathsf{depth}_i^B\}.$$

If $\mathsf{depth}_j^A = d$ we say that slot $j$ of session $A$ is a depth-$d$ slot; likewise, $A$ is a depth-$d$ session if $\mathsf{depth}^A = d$. When we do not need to be explicit about the the session, we will write $\mathsf{depth}_j$ to refer to the depth of the $j$-th slot of some underlying session.

**Our protocol.** Our new protocol is obtained by simply replacing the footer-free condition in Rosen-Shelat protocol with the condition that the depth of the slot is at most $d$. For completeness, we give the description in Protocol 2.

The completeness and soundness of this protocol follow from that of Rosen-Shelat. The proof of zero-knowledge property is given in Section 6.2.

### 6.1 Bounding Optimal Sessions for Our Protocol

Bounding optimal sessions for our protocol in the $p$-FIFO model turns out to be trivial. Actually, the $p$-FIFO model is the best case scenario where *all* sessions are optimal with just $d = 1$. Due to this, it does not matter if $p$-FIFO stops after 1 slot and result holds for arbitrary $k$-slots.

**Proposition 3.** *All sessions of our protocol in the $p$-FIFO model are optimal if the depth parameter $d \geq 1$, for all values of $p$ and number of slots $k$.*

*Proof.* Assume that there exist a session $A$ whose first slot has a depth more than 0. Then there must be some session $B$ nested between messages $P_1^A$ and $V_1^A$. That means $B$ is opened after $A$, but its last message is scheduled before that of $A$. This contradicts the FIFO order of closing the slots. Thus, every session must be optimal. □

We note that for our protocol, $p$-LIFO provides more insight into protocol's performance than the $p$-FIFO model. This can be seen from the experimental simulations we perform and provide in Section 7.

### 6.2 Proof of Zero-Knowledge Property of Our Protocol

In this section, we prove the zero-knowledge property of our protocol by describing an expected polynomial-time simulator and proving corresponding indistinguishability claim regarding its output. We will first present a **subroutine** to simulate depth-$d$ schedules, and then show the full simulator based on the subroutine.

**Simulating Depth-$d$ Schedules.** If a concurrent adversary $V^*$ is guaranteed to never produce schedules of depth $> d$, then its view can be simulated in $\mathsf{poly}(n^d)$ time using the naïve rewinding strategy which solves each slot (of each session) by rewinding it immediately after it is closed; if a previously solved session gets rewound past its first message, the simulator just solves it again as needed. This is presented as the subroutine naïve_recurse below; the description contains additional details (to make it compatible with our full simulator presented later).

In the sequel, we only focus on preamble messages and timely extraction of the simulation trapdoor for each session; for standard details on how to handle second stage messages see [Ros04, PRS02].

**Subroutine** $\mathsf{naïve\_recurse}(d, \mathsf{view}, V_j^s, \mathsf{aux})$**.** The input to the subroutine consists of a maximum depth parameter $d$, a partial view of the adversary $V^*$ containing $V_j^s$ as the last message in the transcript so far denoting the closure of slot $j$ of some sessions $s$, and a list of auxiliary inputs aux containing the simulation trapdoors $\sigma_{s'}$ for every session $s'$ not nested in slot $(P_j^s, V_j^s)$ (where $j, s, P_j^s$ are implicit in view). The goal of the subroutine is to extract simulation trapdoor $\sigma_s$ for session $s$ and append it to aux.

Let $\mathsf{st}_j$ denote the state of $V^*$ before message $P_j^s$ is sent. Proceed as follows:

1. Rewind $V^*$ to state $\mathsf{st}_j$ and send a freshly sampled prover message $P_j'^s$ for slot $j$ of sessions $s$.
2. If $V^*$ sends a second-stage message of an already solved session, send the next message of that session (by using trapdoors from aux).
3. If $V^*$ opens a new session $s'$, send next message of $s'$ honestly.
4. If $V^*$ closes a slot $j'$ of an unsolved session $s' \neq s$ by sending $V_{s'}^{j'}$:[4]
   - if $\mathsf{depth}_{j'}^{s'} < d$, call:

     $$\mathsf{naïve\_recurse}(\mathsf{depth}_{j'}^{s'}, \mathsf{view}', V_{s'}^{j'}, \mathsf{aux}')$$

     where $\mathsf{view}', \mathsf{aux}'$ are current values of variables view and aux.

[4] Note that by our restriction on aux, if $s'$ is not solved, it must have started after the slot $j$ of $s$ was already opened.

– else **go to** step 1.

5. If $V^*$ halts without sending a valid $V_j^{s'}$, **go to** step 1.
6. Otherwise, extract $\sigma_s$ from values $(P_j^s, V_j^s, P_j'^s, V_j'^s)$ and append it to $\mathsf{aux}$.[5]

This completes the description of $\mathsf{naïve\_recurse}$.

**Lemma 4.** *The expected running time of* $\mathsf{naïve\_recurse}(d, \mathsf{view}, V_j^s, \mathsf{aux})$ *is*

$$\frac{\mathsf{poly}(n^d)}{\zeta_{j,s,\mathsf{view}}}$$

*where* $\zeta_{j,s,\mathsf{view}}$ *denotes the probability that interaction with* $V_{j,s}^*$ *results in successful closing of* $j$*-th slot of session* $s$ *with depth at most* $d$*, and* $V_{j,s}^*$ *is algorithm* $V^*$ *starting from the state immediately before slot* $j$ *of* $s$ *is opened in* $\mathsf{view}$*.*

*Proof.* The proof is straightforward but some care is needed since the slot may contain many nested sessions. The lemma holds for $d = 0$ since in this case the slot cannot contain any nested sessions so that all steps are strict polynomial time, repeated $\frac{1}{\zeta_{j,s,\mathsf{view}}}$ times in expectation.

Consider the case for $d = 1$. The slot can contain several nested sessions $a_1, a_2, \ldots$, each of depth at most 0. The messages of these sessions can overlap with each other but no $s_i$ is (fully) nested in $a_j$ since otherwise they will not be of depth 0. Now let us consider the running time of a single execution attempt (i.e., without executing the **go to** steps) to complete this slot. Either the slot will be successfully closed or the execution will decide to reach the go to step. The running time of a single execution is thus the sum of the running times for each internal depth 0 sessions $a_1, a_2, \ldots$ (polynomially many). Note that the trapdoors $\sigma_{a_i}$ for each internal session $a_i$ only needs to be extracted if $a_i$ is successfully completed with degree at most 0. We now apply the standard "$p \times 1/p$" argument: the expected time for completing each of sessions $a_i$ along with extracting trapdoors $\sigma_{a_i}$ is polynomial (since the lemma holds for $d = 0$). Therefore, the expected running time to complete a *single* run is at most $\mathsf{poly}(n)$.

Now, the total number of runs, in expectation, are $\frac{1}{\zeta_{j,s,\mathsf{view}}}$, and these runs are independent of each other. So we can just multiply the two expectations to get the total expected time to extract $\sigma$ for $d = 1$.

Assume by induction that the claim is true for all depths less than $d$. We now simply repeat the above argument. Specifically, the transcript of a depth $d$ slot can be viewed as consisting of several sessions $A_1, A_2, \ldots$, each of depth at most $d-1$. The expected time to simulate each of these sessions is $\mathsf{poly}(n^{d-1})$ due to the induction assumption and the "$p \times 1/p$" argument as above. The expected time for a single run for depth $d$ slot is the sum of values $\mathsf{poly}(n^{d-1})$ for $\mathsf{poly}(n)$ times, which is at most $\mathsf{poly}(n^d)$. And since there are $\frac{1}{\zeta_{j,s,\mathsf{view}}}$ independent runs, we get the bound in the lemma. □

**Full Simulator.** Our full simulator is identical to the simulator for Rosen-Shelat except that we use the subroutine $\mathsf{naïve\_recurse}$ described above instead of the nested-footer strategy used by them. All claims also follow easily without any significant changes (except the $\mathsf{naïve\_recurse}$ related changes). At a high level, this is sufficient because the Rosen-Shelat simulator relies on following two key conditions:

[5] We assume wlog that $\sigma_s$ can always be extracted from such values; in rare cases (e.g.,when $P_j^s = P_j'^s$) the trapdoor can be extracted in exponential time by a higher level simulator without affecting the expected running time overall.

1. When the PRS-based recursive simulation reaches the bottom level of recursion (i.e., recursion depth $\ell = 1$), the naïve "rewind until successful" strategy (used for footer-free slots) contributes only polynomial amount in expectation to the running time of the full simulator. We can replace this with essentially any other procedure that maintains the polynomial contribution condition, e.g., the naïve_recurse procedure if $d$ is a constant.
2. When employing the naïve rewinding strategy for footer-free sessions, the trapdoors for all sessions that start before the slot have already been extracted (either by the PRS-strategy or by the naïve one). This invariant is also maintained for our simulator and reflected in the condition for input aux. Indeed, a depth-$d$ slot considers sessions that are fully nested in it with their first and last messages. If a session starts in the slot but ends outside, its trapdoor is not needed during the slot; if it starts before the slot but ends in the slot, its trapdoor must have been extracted either by naïve_recurse or the PRS-strategy.

For completeness, we now present the full simulator. It is taken almost verbatim from [RS10]; we only incorporate changes corresponding to the $\ell = 1$ case in the recursion.

We construct a black-box simulator $S$ to demonstrate that the c$\mathcal{ZK}$ property holds. Given a dishonest verifier $V^*$ that acts as the adversary in our concurrent scenario, $S$ will rewind the interaction with $V^*$ and examine the behavior of inputs and outputs. We define SOLVE procedure that supplies the simulator with challenges from $V^*$ before it reaches **stage 2** in the protocol. This is done by rewinding the interaction with $V^*$ while trying to achieve two "different" answers to some $(Pj)$ message using the naïve_recursesubroutine.

The timing of the rewinds performed by SOLVE depends on the number of **stage 1** verifier messages received so far and on the size of the schedule. Whenever SOLVE encounters a situation where the slot in the session is $\mathsf{depth}^s_j \leq d$, it adaptively assumes this is a case where it can solve that session using the naïve "rewind until successful" strategy to extract the simulation trapdoor.

SOLVE subroutine splits the **Stage 1** messages passed as input to it into two halves and invokes itself recursively twice for each half (completing the two runs of the first half before proceeding to the two runs of the second half). At the top level of the recursion, the messages that are about to be explored consist of the entire schedule, whereas at the bottom level the procedure explores only a single message. It may read a message multiple times via rewinding depending on the depth. SOLVE outputs a message only once after the first encounter with it.

The input to SOLVE consists of a triplet $(\ell, \mathsf{hist}, \mathsf{T})$. The parameter $\ell$ corresponds to the total number of messages from the verifier, hist is a string of messages in the "first-visited" history of interaction, and T is a table containing the contents of all the messages explored upto that point. The messages stored in T are used in order to determine $\sigma$ according to answers $(Vj)$ to different $(Pj)$. They are kept relevant by constantly keeping track of the sessions that are rewound past their initial commitment. That is, whenever SOLVE rewinds past the $(V0)$ message of a session, all messages belonging to this session are deleted from T.

The analysis takes advantage of the fact that no naı"vely rewound slot has $\mathsf{depth} > d$, building on the assumption that a slot with $\mathsf{depth} \leq d$ is an event of non-negligible probability (or otherwise it would not have occurred). Repeated rewinding means that the simulator will obtain an execution of this slot with $\mathsf{depth} \leq d$ again pretty soon in expectation. This will enable it to successfully solve the session even though the sessions may have less than $k$ slots.

The SOLVE procedure is described below (Figure 2).

**Procedure** SOLVE $(\ell, \mathsf{hist}, \mathsf{T})$**:**

BOTTOM LEVEL $(\ell = 1)$:

1. For each $s \in \{1, \cdots, m\}$, if the initial commitment, $(V0)$, of session $s$ does not appear in $\mathsf{hist}$, delete all session $s$ messages from $\mathsf{T}$.
2. Run $\beta \leftarrow V^*(\mathsf{hist}, p)$. If $\beta$ is of the form $(\mathsf{recv}, V, \alpha, t)$, then continue to the next step. Else if it is $(\mathsf{send}, V, t)$, then uniformly choose a first stage prover message $p$, append it to the transcript at time $t$, and repeat this step. If $t$ or $\alpha$ are invalid, then halt the simulation and output the current transcript.
3. Let
   - $(p_1, v_1, \cdots, p_t, v_t) = (\mathsf{hist}, p, v)$
   - $i$ be the session number to which $v$ corresponds. $v_t$ is the $i^{th}$ message from verifier to prover in some session $s$, namely $v_t = V_i^s$.
4. If there exists a pair of indices $(a, b)$ such that $a \in [t]$ and $b = t$ for which:
   - $v_b \neq \mathsf{ABORT}$
   - both $v_b$ and $p_a$ belong to session $s$ and $(p_a, v_b)$ construct slot $i$ of some session $s$, namely $(p_a, v_b) = (P_i^s, V_i^s)$.
   - and, $\mathsf{depth}_i^s \leq d$.

   Then solve such $(a, b)$ using $\mathsf{naïve\_recurse}(\mathsf{depth}_i^s, \mathsf{view}, V_i^s, \mathsf{T})$;
   it stores all the necessary information for this sessions in $\mathsf{T}$, including the second successful run of the slot, denoted $(p, v)$.
5. output $\mathsf{T}$, $(p,v)$.

RECURSION $(\ell > 1)$:

1. Set $\mathsf{T}_1, (p_1, v_1, \cdots, p_{\frac{\ell}{2}}, v_{\frac{\ell}{2}}) \leftarrow \mathsf{SOLVE}\ (\frac{\ell}{2}, \mathsf{hist}, \mathsf{T})$.
2. Set $\mathsf{T}_2, (\widetilde{p}_1, \widetilde{v}_1, \cdots, \widetilde{p}_{\frac{\ell}{2}}, \widetilde{v}_{\frac{\ell}{2}}) \leftarrow \mathsf{SOLVE}\ (\frac{\ell}{2}, \mathsf{hist}, \mathsf{T}_1)$.
3. Set $\mathsf{T}_3, (p_{(\frac{\ell}{2}+1)}, v_{(\frac{\ell}{2}+1)}, \cdots, p_\ell, v_\ell) \leftarrow \mathsf{SOLVE}\ (\frac{\ell}{2}, (\mathsf{hist}, p_1, v_1, \cdots, p_{\frac{\ell}{2}}, v_{\frac{\ell}{2}}), \mathsf{T}_2)$.
4. $\mathsf{T}_4, (\widetilde{p}_{(\frac{\ell}{2}+1)}, \widetilde{v}_{(\frac{\ell}{2}+1)}, \cdots, \widetilde{p}_\ell, \widetilde{v}_\ell) \leftarrow \mathsf{SOLVE}\ (\frac{\ell}{2}, (\mathsf{hist}, p_1, v_1, \cdots, p_{\frac{\ell}{2}}, v_{\frac{\ell}{2}}), \mathsf{T}_3)$.
5. Output $\mathsf{T}_4, (p_1, v_1, \cdots, p_\ell, v_\ell)$.

**Fig. 2.** The SOLVE Procedure

The analysis of the simulator is almost identical to that of [RS10] except that we use the bound $\frac{\mathsf{poly}(n^d)}{\zeta_{j,s,\mathsf{view}}}$ for the running time of $\mathsf{naïve\_recurse}$ instead of $\frac{1}{\zeta_{a,b}}$ in the calculation of the expected running time of the simulator. The details are omitted.

## 7 Experimental Simulations

In this section we display some empirical results of simulation to show the performance of our protocol as well as Rosen-Shelat protocol in various models.

Figure 3 shows the average fraction of non-optimal sessions on 1-slot $p$-FIFO and $p$-LIFO. In the $p$-FIFO setting, all sessions in our protocol are optimal, just as we proved in Proposition 3. In addition, it is clear in this plot that the empirical result agrees with the theoretical bound we derived for 1-Slot $p$-FIFO and $p$-LIFO earlier. In 1-Slot $p$-LIFO setting, our model performs the same as Rosen-Shelat. We expect it to be so because, in this setting, our model is the same as Rosen-Shelat's model in terms of optimal sessions. Again, this plot shows that the empirical results coincide with our theoretical bound.

Next, we consider the simulation for higher number slots, e.g., 10 slots. Note that even with 10-slots, in the $p$-FIFO model our protocol will always have all sessions to be good (due to Proposition 3). Therefore, we only generate the plot for the $p$-LIFO model. This plot is appears in figure 4

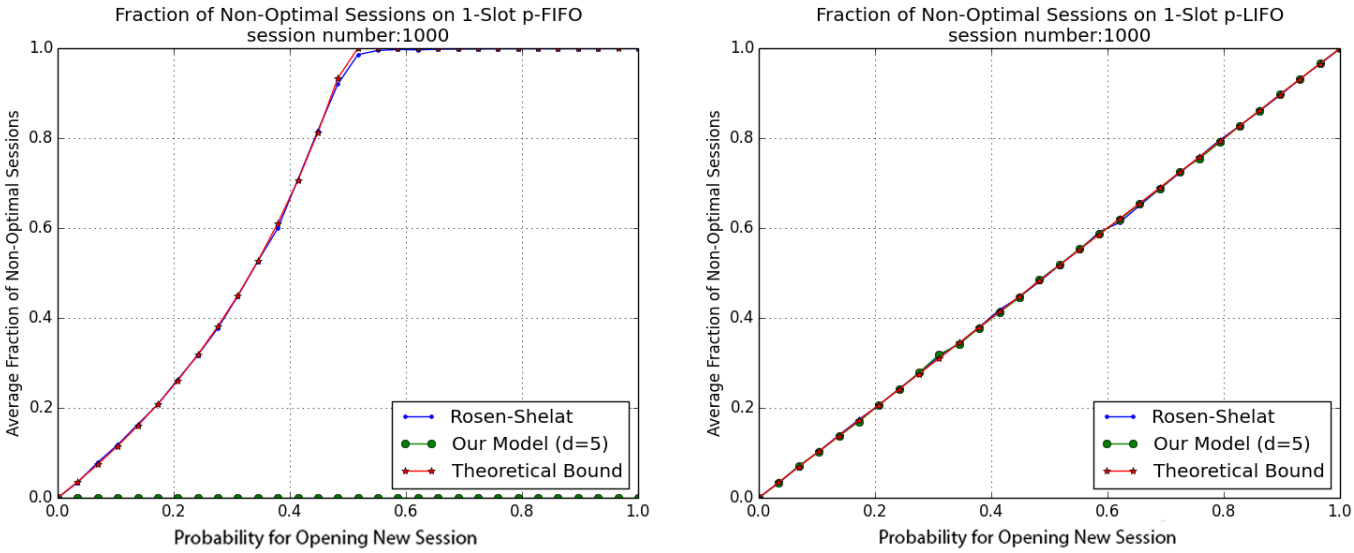


**Fig. 3.** Comparison for Fraction of Non-Optimal Sessions in 1-Slot Setting

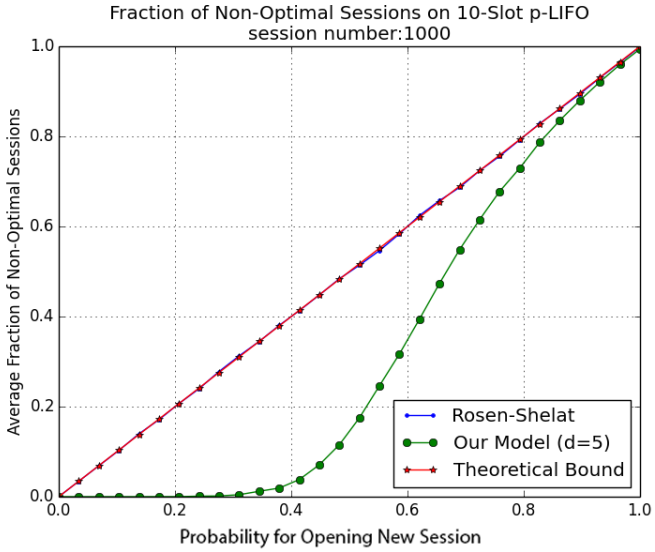


**Fig. 4.** Comparison for Fraction of Non-Optimal Sessions in 10-Slot Setting

and shows that our protocol performs significantly better than the Rosen-Shelat protocol (even for moderate values of the depth parameter, such as 5). By picking a higher constant for depth we can expect to see a higher fraction of optimal sessions for our model.

## References


AS04. Noga Alon and Joel H Spencer. *The probabilistic method*. John Wiley & Sons, 2004.

Bar01. B. Barak. How to go beyond the black-box simulation barrier. In *Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science*, FOCS '01, pages 106–, Washington, DC, USA, 2001. IEEE Computer Society.

BGI+01. B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S. Vadhan, and K. Yang. On the (im)possibility of obfuscating programs. In *CRYPTO*, pages 1–18, 2001.

CGGM00. Ran Canetti, Oded Goldreich, Shafi Goldwasser, and Silvio Micali. Resettable zero-knowledge. In *STOC*, pages 235–244, 2000.

CJP14. Ran Canetti, Abhishek Jain, and Omer Paneth. Client-server concurrent zero knowledge with constant rounds and guaranteed complexity. In *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part II*, pages 337–350, 2014.

CKPR01. Ran Canetti, Joe Kilian, Erez Petrank, and Alon Rosen. Black-box concurrent zero-knowledge requires\tilde {$\Omega$}(log n) rounds. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 570–579. ACM, 2001.

CLP13a. Ran Canetti, Huijia Lin, and Omer Paneth. Public-coin concurrent zero-knowledge in the global hash model. In *TCC*, pages 80–99, 2013.

CLP13b. Kai-Min Chung, Huijia Lin, and Rafael Pass. Constant-round concurrent zero knowledge from p-certificates. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 50–59. IEEE, 2013.

CLP15. Kai-Min Chung, Huijia Lin, and Rafael Pass. Constant-round concurrent zero-knowledge from indistinguishability obfuscation. In *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, pages 287–307, 2015.

CPV04. Giovanni Di Crescenzo, Giuseppe Persiano, and Ivan Visconti. Constant-round resettable zero knowledge with concurrent soundness in the bare public-key model. In *CRYPTO*, pages 237–253, 2004.

Dam91. Ivan Damgård. Towards practical public key systems secure against chosen ciphertext attacks. In *CRYPTO*, pages 445–456, 1991.

DNS98. Cynthia Dwork, Moni Naor, and Amit Sahai. Concurrent zero-knowledge. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, pages 409–418, New York, NY, USA, 1998. ACM.

DS98. Cynthia Dwork and Amit Sahai. Concurrent zero-knowledge: Reducing the need for timing constraints. In *CRYPTO*, pages 442–457, 1998.

EP15. Rim Essifi and Marc Peigné. Return probabilities for the reflected random walk on n0. *Journal of Theoretical Probability*, 28(1):231–258, 2015.

Fel68. William Feller. *An Introduction to Probability Theory and its Applications: Volume 1*. 1968.

FS90. Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, pages 416–426, 1990.

GGH+13. Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *FOCS*, 2013.

GJO+13. Vipul Goyal, Abhishek Jain, Rafail Ostrovsky, Silas Richelson, and Ivan Visconti. Concurrent zero knowledge in the bounded player model. In *TCC*, pages 60–79, 2013.

GK90. Oded Goldreich and Hugo Krawczyk. On the composition of zero-knowledge proof systems. In *Automata, Languages and Programming, 17th International Colloquium, ICALP90, Warwick University, England, July 16-20, 1990, Proceedings*, pages 268–282, 1990.

GMR85. S Goldwasser, S Micali, and C Rackoff. The knowledge complexity of interactive proof-systems. In *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*, STOC '85, pages 291–304, New York, NY, USA, 1985. ACM.

Gol02. Oded Goldreich. Concurrent zero-knowledge with timing, revisited. In *STOC*, pages 332–340, 2002.

GS14. Divya Gupta and Amit Sahai. On constant-round concurrent zero-knowledge from a knowledge assumption. In *Progress in Cryptology - INDOCRYPT 2014 - 15th International Conference on Cryptology in India, New Delhi, India, December 14-17, 2014, Proceedings*, pages 71–88, 2014.

HT98. Satoshi Hada and Toshiaki Tanaka. On the existence of 3-round zero-knowledge protocols. In *Annual International Cryptology Conference*, pages 408–423. Springer, 1998.

IPS15. Yuval Ishai, Omkant Pandey, and Amit Sahai. Public Coin Differing-Inputs Obfuscation. In *TCC*, 2015.

KP01. Joe Kilian and Erez Petrank. Concurrent and resettable zero-knowledge in poly-logarithmic rounds. In *STOC'01 Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing*, pages 560–569. ACM, 2001.

KPR98. J. Kilian, E. Petrank, and C. Rackoff. Lower bounds for zero knowledge on the Internet. In *FOCS*, pages 484–492, 1998.

Lal95. Steven P Lalley. Return probabilities for random walk on a half-line. *Journal of Theoretical Probability*, 8(3):571–599, 1995.

Lam06. Leslie Lamport. Fast paxos. *Distributed Computing*, 19(2):79–103, 2006.

PPS15. Omkant Pandey, Manoj Prabhakaran, and Amit Sahai. Obfuscation-based non-black-box simulation and four message concurrent zero knowledge for np. In *Theory of Cryptography Conference*, pages 638–667. Springer, 2015.

PRS02. Manoj Prabhakaran, Alon Rosen, and Amit Sahai. Concurrent zero knowledge with logarithmic round-complexity. In *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium on*, pages 366–375. IEEE, 2002.

PTV10. Rafael Pass, Wei-Lung Dustin Tseng, and Muthuramakrishnan Venkitasubramaniam. Eye for an eye: Efficient concurrent zero-knowledge in the timing model. In *TCC*, pages 518–534, 2010.

PV05. Giuseppe Persiano and Ivan Visconti. Single-prover concurrent zero knowledge in almost constant rounds. In *Automata, Languages and Programming, 32nd International Colloquium, ICALP 2005, Lisbon, Portugal, July 11-15, 2005, Proceedings*, pages 228–240, 2005.

RK99. Ransom Richardson and Joe Kilian. On the concurrent composition of zero-knowledge proofs. In *Proceedings of the 17th International Conference on Theory and Application of Cryptographic Techniques*, EUROCRYPT'99, pages 415–431, Berlin, Heidelberg, 1999. Springer-Verlag.

Ros00. Alon Rosen. A note on the round-complexity of concurrent zero-knowledge. In *CRYPTO*, pages 451–468, 2000.

Ros04. Alon Rosen. *PhD Thesis : The Round-Complexity of Black-Box Concurrent Zero-Knowledge*. PhD thesis, The Weizmann Institute of Science, Israel, 2004.

RS10. Alon Rosen and Abhi Shelat. Optimistic concurrent zero knowledge. *Advances in Cryptology-ASIACRYPT 2010*, pages 359–376, 2010.

SV12. Alessandra Scafuro and Ivan Visconti. On round-optimal zero knowledge in the bare public-key model. In *EUROCRYP*, pages 153–171, 2012.

## Appendix

## A Proof for $|B_i - B_{i+1}| < 1$

(All symbols in this proof are inherited from Section 4.)

We start with two lemmas about some general truth of the random walk with reflection at zero.

**Lemma 5.** *In a random walk with reflection at zero, at any step $t$, for any length $L$ and any current position $a \in \mathbb{N}$, the following holds:*

$$0 \leq E[\sum_{i=1}^{L} X_{t+i} | S_t = a] - E[\sum_{i=1}^{L-1} X_{t+i} | S_t = a] \leq 1$$

*Remark 3.* The proof is omitted for this lemma since it is obvious once we use the linearity of the expectation operator. But note that $S_t$ means the position at step $t$ ($S_t = \sum_{i=1}^{t} \epsilon_i$), while $\sum_{i=1}^{L} X_{t+i}$ is the number of "going right" from step $t+1$ to step $t+L$.

The following corollary follows immediately from Lemma 5.

**Corollary 2.** *In a random walks with reflection at zero, at any step $t$, for any length $L$ and any $a \in \mathbb{N}$, the following holds:*

$$0 \leq E[\sum_{i=1}^{L} X_{t+i} | S_t = a] - E[\sum_{i=1}^{L-1} X_{t+1+i} | S_{t+1} = a] \leq 1$$

*Proof.* Substitute $t+1$ by $t$ in $E[\sum_{i=1}^{L-1} X_{t+1+i} | S_{t+1} = a]$ dose not change its value since this value only related to $L$, $a$ and $p$. Thus we have

$$E[\sum_{i=1}^{L} X_{t+i} | S_t = a] - E[\sum_{i=1}^{L-1} X_{t+1+i} | S_{t+1} = a] = E[\sum_{i=1}^{L} X_{t+i} | S_t = a] - E[\sum_{i=1}^{L-1} X_{t+i} | S_t = a]$$

Then according to Lemma 5, we proved the bounds in this Corollary

Here comes the second lemma, which is critical to our main theorem.

**Lemma 6 (Diff-1 Lemma).** *Given a random walks with reflection at zero $\{X_i\}$, at any step $t$, for any length $L$ and any $a \in \mathbb{N}$ , we have:*

$$0 \leq E[\sum_{i=1}^{L} X_{t+i} | S_t = a] - E[\sum_{i=1}^{L} X_{t+i} | S_t = a+1] \leq 1$$

*Proof.* Construct two sets of events: $K = \{k_0, k_1, k_2, ..., k_L\}$ and $K' = \{k'_0, k'_1, k'_2, ..., k'_L\}$.

$k_j$ ($j \neq 0$) denote the event that starting from $S_t = a$, our first visit to the origin (position 0) happens at step $t+j$. $k_0$ means we do not visit position 0 up to step $t+L$.

Let $k'_j$ ($j \neq 0$) denote the event that starting from $S_t = a+1$, we our first visit to position 1 happens at step $t+j$. $k'_0$ means we do not visit position 1 up to step $t+L$.

$$E[\sum_{i=1}^{L} X_{t+i}|S_t = a] = E_K\left[E[\sum_{i=1}^{L} X_{t+i}|S_t = a, K]\right]$$
$$= \sum_{k_j \in K} E[\sum_{i=1}^{L} X_{t+i}|S_t = a, K = k_j] \cdot P(K = k_j|S_t = a)$$

Similarly,

$$E[\sum_{i=1}^{L} X_{t+i}|S_t = a+1] = \sum_{k'_j \in K'} E[\sum_{i=1}^{L} X_{t+i}|S_t = a+1, K' = k'_j] \cdot P(K' = k'_j|S_t = a+1)$$

By definition, it is obvious that

$$P(K = k_j|S_t = a) = P(K' = k'_j|S_t = a+1).$$

To finish the prove, now we only need to show:

$$0 \leq E[\sum_{i=1}^{L} X_{t+i}|S_t = a] - E[\sum_{i=1}^{L} X_{t+i}|S_t = a+1] \leq 1$$

Further, we know that:

$$E[\sum_{i=1}^{L} X_{t+i}|S_t = a] - E[\sum_{i=1}^{L} X_{t+i}|S_t = a+1]$$
$$= \sum_{j=0}^{L}\left(E[\sum_{i=1}^{L} X_{t+i}|S_t = a, K = k_j] - E[\sum_{i=1}^{L} X_{t+i}|S_t = a+1, K' = k'_j]\right) \cdot P(K = k_j|S_t = a)$$

Then once we show the following holds for all $j$, we are done with the proof:

$$0 \leq E[\sum_{i=1}^{L} X_{t+i}|S_t = a, K = k_j] - E[\sum_{i=1}^{L} X_{t+i}|S_t = a+1, K' = k'_j] \leq 1 \tag{20}$$

For $j = 0$, this model is equivalent to a segment of pure random walk, because we never visit 0 thus no "bumping up" can happen. The above inequality holds because

$$E[\sum_{i=1}^{L} X_{t+i}|S_t = a, K = k_0] - E[\sum_{i=1}^{L} X_{t+i}|S_t = a+1, K' = k'_0] = p \cdot L - p \cdot L = 0$$

Now let us focus on the case that $j \neq 0$:

$$E[\sum_{i=1}^{L} X_{t+i}|S_t = a, K = k_j] = E[\sum_{i=1}^{j} X_{t+i} + \sum_{i=j+1}^{L} X_{t+i}|S_t = a, K = k_j]$$
$$= E[\sum_{i=1}^{j} X_{t+i}|S_t = a, K = k_j] + E[\sum_{i=j+1}^{L} X_{t+i}|S_t = a, K = k_j]$$

The first term in the above step means the expected number of "going right" starting from $S_t$ and end at step $t+j$, which is the first time we reach 0 after step t. Actually this expectation is a constant, which can be calculated easily. It is $\frac{j-a}{2}$.

For the second term in the above step, note that we must bump right at step $t+j+1$ since we have already reached 0 at step $t+j$. Further, since we now are already at step $S_{t+j+1}$ and the expectation is taken on $\{X_{t+j+2}, X_{t+j+3}, ..., X_{t+L}\}$, so the previous steps dose not matter once we condition it on the current position. That means we can change the condition $(S_t = a, K = k_j)$ to $(S_{t+j+1} = 1)$ now. So

$$E[\sum_{i=j+1}^{L} X_{t+i} | S_t = a, K = k_j] = 1 + E[\sum_{i=j+2}^{L} X_{t+i} | S_{t+j+1} = 1]$$

Pull them together, we have:

$$E[\sum_{i=1}^{L} X_{t+i} | S_t = a, K = k_j] = \frac{j-a}{2} + 1 + E[\sum_{i=j+2}^{L} X_{t+i} | S_{t+j+1} = 1] \tag{21}$$

Now consider the case for $S_t = a+1$. A similar argument gives us:

$$E[\sum_{i=1}^{L} X_{t+i} | S_t = a+1, K' = k'_j] = \frac{j-a}{2} + E[\sum_{i=j+1}^{L} X_{t+i} | S_{t+j} = 1] \tag{22}$$

Equation (21) minus (22) gives us:

$$\begin{aligned} &E[\sum_{i=1}^{L} X_{t+i} | S_t = a, K = k_j] - E[\sum_{i=1}^{L} X_{t+i} | S_t = a+1, K' = k'_j] \\ &= 1 - \left( E[\sum_{i=j+1}^{L} X_{t+i} | S_{t+j} = 1] - E[\sum_{i=j+2}^{L} X_{t+i} | S_{t+j+1} = 1] \right) \end{aligned}$$

By Corollary 2, we know that:

$$0 \leq \left( E[\sum_{i=j+1}^{L} X_{t+i} | S_{t+j} = 1] - E[\sum_{i=j+2}^{L} X_{t+i} | S_{t+j+1} = 1] \right) \leq 1$$

thus Equation (20) holds. Therefore we proved this Lemma.

With these two lemmas above, we are now ready to prove $|B_i - B_{i+1}| \leq 1$. To shorten the notation, we will use $E^t_{i+1}[M_t | X_1 : X_i]$ to substitute

$$E_{X_{i+1}, X_{i+2}, .., X_t}[M_t | X_1, X_2, ..., X_i]$$

By definition,

$$\begin{aligned} B_i - B_{i+1} &= E^t_{i+1}[M_t | X_1 : X_i] - E^t_{i+2}[M_t | X_1 : X_i, X_{i+1}] \\ &= E_{X_{i+1}} \left\{ E_{X_{i+2:X_t}}[M_t | X_1 : X_i, X_{i+1}] | X_1 : X_i \right\} - \\ &\quad E^t_{i+2}[M_t | X_1 : X_i, X_{i+1}] \qquad (23) \\ &= P(X_{i+1} = 0 | X_1 : X_i) \cdot E^t_{i+2}[M_t | X_1 : X_i, X_{i+1} = 0] + \\ &\quad P(X_{i+1} = 1 | X_1 : X_i) \cdot E^t_{i+2}[M_t | X_1 : X_i, X_{i+1} = 1] - \\ &\quad E^t_{i+2}[M_t | X_1 : X_i, X_{i+1}] \qquad (24) \end{aligned}$$

*Remark 4.* At step (23) in the above, we take advantage of the property of expectation operator that $E[X|A] = E_B\{E[X|A,B]|A\}$. It following from the linearity of expectation and Law of Total Probability.

Note that $B_i - B_{i+1}$ is indeed a function on random vector $(X_1, X_2, ..., X_{i+1})$. So we can use the notation $f(X_1, X_2, ..., X_{i+1}) := B_i - B_{i+1}$. We want to prove that $|B_i - B_{i+1}| < c_i$, which is the same as: Find some constant $c_i$ such that $|f(X_1, X_2, ..., X_{i+1})| < c_i$ holds for all points in the sample space of random vector $(X_1, X_2, ..., X_{i+1})$. Now let us partition the sample space of this random vector by its last element and discuss the corresponding behavior.

For $\boldsymbol{x} \in \{(X_1, X_2, ..., X_{i+1})|X_{i+1} = 1\}$, equation (24) becomes:

$$\begin{aligned}
f(\boldsymbol{x}) &= P(X_{i+1} = 0|X_1 : X_i) \cdot E^t_{i+2}[M_t|X_1 : X_i, X_{i+1} = 0] + \\
&\quad P(X_{i+1} = 1|X_1 : X_i) \cdot E^t_{i+2}[M_t|X_1 : X_i, X_{i+1} = 1] - \\
&\quad E^t_{i+2}[M_t|X_1 : X_i, X_{i+1} = 1] \\
&= P(X_{i+1} = 0|X_1 : X_i) \cdot E^t_{i+2}[M_t|X_1 : X_i, X_{i+1} = 0] + \\
&\quad (P(X_{i+1} = 1|X_1 : X_i) - 1) \cdot E^t_{i+2}[M_t|X_1 : X_i, X_{i+1} = 1] \\
&= P(X_{i+1} = 0|X_1 : X_i) \cdot \\
&\quad \big(E^t_{i+2}[M_t|X_1 : X_i, X_{i+1} = 0] - E^t_{i+2}[M_t|X_1 : X_i, X_{i+1} = 1]\big) \\
&= P(X_{i+1} = 0|X_1 : X_i) \cdot \\
&\quad \left(E^t_{i+2}[\sum_{j=i+1}^{t} X_j|X_1 : X_i, X_{i+1} = 0] - E^t_{i+2}[\sum_{j=i+1}^{t} X_j|X_1 : X_i, X_{i+1} = 1]\right) \\
&= -P(X_{i+1} = 0|X_1 : X_i) + P(X_{i+1} = 0|X_1 : X_i) \cdot \\
&\quad \left(E^t_{i+2}[\sum_{j=i+2}^{t} X_j|X_1 : X_i, X_{i+1} = 0] - E^t_{i+2}[\sum_{j=i+2}^{t} X_j|X_1 : X_i, X_{i+1} = 1]\right) \qquad (25)
\end{aligned}$$

For $\boldsymbol{x} \in \{(X_1, X_2, ..., X_{i+1})|X_{i+1} = 0\}$, similar derivation shows that:

$$\begin{aligned}
f(\boldsymbol{x}) &= P(X_{i+1} = 1|X_1 : X_i) \cdot \\
&\quad \left(E^t_{i+2}[\sum_{j=i+1}^{t} X_j|X_1 : X_i, X_{i+1} = 1] - E^t_{i+2}[\sum_{j=i+1}^{t} X_j|X_1 : X_i, X_{i+1} = 0]\right) \\
&= P(X_{i+1} = 1|X_1 : X_i) + P(X_{i+1} = 1|X_1 : X_i) \cdot \\
&\quad \left(E^t_{i+2}[\sum_{j=i+2}^{t} X_j|X_1 : X_i, X_{i+1} = 1] - E^t_{i+2}[\sum_{j=i+2}^{t} X_j|X_1 : X_i, X_{i+1} = 0]\right) \qquad (26)
\end{aligned}$$

From Equation (25) and (26), we know that if we could show the following bounds for some positive number $c$

$$0 \leq E^t_{i+2}[\sum_{j=i+2}^{t} X_j|X_1 : X_i, X_{i+1} = 0] - E^t_{i+2}[\sum_{j=i+2}^{t} X_j|X_1 : X_i, X_{i+1} = 1] \leq c \qquad (27)$$

then by denoting $p^* = P(X_{i+1} = 1|X_1 : X_i)$ and $q^* = P(X_{i+1} = 0|X_1 : X_i)$, we will have

$$\begin{cases} \forall \boldsymbol{x} \in \{(X_1, X_2, ..., X_{i+1}) | X_{i+1} = 1\}, & -q^* \leq f(\boldsymbol{x}) \leq (c-1) \cdot q^* \\ \forall \boldsymbol{x} \in \{(X_1, X_2, ..., X_{i+1}) | X_{i+1} = 0\}, & (1-c) \cdot p^* \leq f(\boldsymbol{x}) \leq p^* \end{cases}$$

That is to say for any point $\boldsymbol{x}$ in the sample space of $(X_1, X_2, ..., X_{i+1})$, $f(\boldsymbol{x}) = B_i - B_{i+1}$ is bounded by

$$\min\{-q^*, (1-c) \cdot p^*\} \leq B_i - B_{i+1} \leq \max\{(c-1) \cdot q^*, p^*\}, \tag{28}$$

Now if we can show that $c = 2$ satisfies Inequality (27), we are done with the proof. This is because setting $c = 2$ in Inequality (28) gives us:

$$-1 \leq \min\{-p^*, -q^*\} \leq B_i - B_{i+1} \leq \max\{p^*, q^*\} \leq 1, \tag{29}$$

which means $|B_i - B_{i+1}| \leq 1$.

The next lemma tells us that $c = 2$ indeed satisfies Inequality (27). So this lemma finishes our proof for $|B_i - B_{i+1}| \leq 1$.

**Lemma 7.**

$$0 \leq E^t_{i+2}[\sum_{j=i+2}^{t} X_j | X_1 : X_i, X_{i+1} = 0] - E^t_{i+2}[\sum_{j=i+2}^{t} X_j | X_1 : X_i, X_{i+1} = 1] \leq 2 \tag{30}$$

*Proof.* It is obvious that for any point $(x_1, x_2, ..., x_i)$ in the sample space of random vector $(X_1, X_2, ..., X_i)$, we have

$$\begin{cases} E^t_{i+2}[\sum_{j=i+2}^{t} X_j | X_1 : X_i, X_{i+1} = 1] = E^t_{i+2}[\sum_{j=i+2}^{t} X_j | S_{i+1} = a+1] \\ E^t_{i+2}[\sum_{j=i+2}^{t} X_j | X_1 : X_i, X_{i+1} = 0] = E^t_{i+2}[\sum_{j=i+2}^{t} X_j | S_{i+1} = a-1] \end{cases},$$

where $a = S_i = \sum_{j=1}^{i} \epsilon_j$.

Then by Lemma 6, we have

$$\begin{cases} 0 \leq E^t_{i+2}[\sum_{j=i+2}^{t} X_j | X_1 : X_i, X_{i+1} = 0] - E^t_{i+2}[\sum_{j=i+2}^{t} X_j | S_{i+1} = a] \leq 1 \\ 0 \leq E^t_{i+2}[\sum_{j=i+2}^{t} X_j | S_{i+1} = a] - E^t_{i+2}[\sum_{j=i+2}^{t} X_j | X_1 : X_i, X_{i+1} = 1] \leq 1 \end{cases}$$

Adding them together proves the inequality in this lemma